symmetry

MDPI

Article

Human–Robot Collaboration on a Disassembly-Line Balancing Problem with an Advanced Multiobjective Discrete Bees Algorithm

Yanda Shen [1], Weidong Lu [2], Haowen Sheng [2], Yangkun Liu [2], Guangdong Tian [1,2,*], Honghao Zhang [1,*] and Zhiwu Li [3]

[1] School of Mechanical Engineering, Shandong University, Jinan 250061, China
[2] School of Mechanical-Electrical and Vehicle Engineering, Beijing University of Civil Engineering and Architecture, Beijing 100044, China; lwdg9z4@163.com (W.L.); shw010504@163.com (H.S.); liuyangkun249688@163.com (Y.L.)
[3] Institute of Systems Engineering, Macau University of Science and Technology, Macau 999078, China
* Correspondence: author: tiangd2013@163.com (G.T.); honghao_zhang@sdu.edu.cn (H.Z.)

Abstract: As resources become increasingly scarce and environmental demands grow, the recycling of products at the end of their lifecycle becomes crucial. Disassembly, as a key stage in the recycling process, plays a decisive role in the sustainability of the entire operation. Advances in automation technology and the integration of Industry 5.0 principles make the balance of human–robot collaborative disassembly lines an important research topic. This study uses disassembly-precedence graphs to clarify disassembly-task information and converts it into a task-precedence matrix. This matrix includes both symmetry and asymmetry, reflecting the dependencies and independencies among disassembly tasks. Based on this, we develop a multiobjective optimisation model that integrates disassembly-task allocation, operation mode selection, and the use of collaborative robots. The objectives are to minimise the number of workstations, the idle rate of the disassembly line, and the energy consumption. Given the asymmetry in disassembly-task attributes, such as the time differences required for disassembling various components and the diverse operation modes, this study employs an evolutionary algorithm to address potential asymmetric optimisation problems. Specifically, we introduce an advanced multi-objective discrete bee algorithm and validate its effectiveness and superiority for solving the disassembly-line balancing problem through a comparative analysis with other algorithms. This research not only provides innovative optimisation strategies for the product-recycling field but also offers valuable experience and reference for the further development of industrial automation and human–robot collaboration.

Keywords: disassembly-line balancing; human–robot collaboration; bees algorithm; green manufacturing

check for updates

Citation: Shen, Y.; Lu, W.; Sheng, H.; Liu, Y.; Tian, G.; Zhang, H.; Li, Z. Human–Robot Collaboration on a Disassembly-Line Balancing Problem with an Advanced Multiobjective Discrete Bees Algorithm. *Symmetry* **2024**, *16*, 794. https://doi.org/10.3390/sym16070794

Academic Editor: Theodore E. Simos

Received: 19 May 2024
Revised: 12 June 2024
Accepted: 19 June 2024
Published: 24 June 2024

Correction Statement: This article has been republished with a minor change. The change does not affect the scientific content of the article and further details are available within the backmatter of the website version of this article.

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In the face of increasingly scarce resources and severe environmental issues, remanufacturing has emerged as a key solution to these challenges [1]. By transforming end-of-life (EOL) products back into usable ones, remanufacturing not only conserves raw materials but also significantly reduces energy consumption and emissions [2]. In this process, the disassembly stage plays a crucial role. It decomposes discarded products into reusable components and materials, providing the necessary raw materials and resources for remanufacturing, and lays a solid foundation for achieving sustainable-development goals [3]. The design and optimisation of disassembly lines are key to improving disassembly efficiency and are indispensable for promoting the sustainable development of industrial resources. Through careful design and optimisation of disassembly lines, disassembly and recycling work can be systematised and proceduralised, thereby reducing production time and costs [4]. Therefore, the disassembly-line balancing (DLB) problem has become an

important issue that urgently needs to be resolved [5,6]. This problem involves the rational allocation of disassembly tasks to various workstations, aiming to optimise efficiency standards while satisfying multiple constraints, such as task priority [5].

With the continuous advancement of automation technology, robots bring new vitality to the disassembly line. These robotic disassembly systems can efficiently execute repetitive disassembly tasks through pre-programmed paths, significantly improving production efficiency [7]. However, as emphasised in Industry 5.0, the unique value and flexibility of human operators are irreplaceable. Human–robot collaborative disassembly combines the flexibility of humans and the precision of robots, showing great potential. But at the same time, the introduction of robots brings new challenges to the DLB problem, and how to optimise DLB in the context of human–robot collaboration has become a problem that urgently needs to be solved [8]. In addition, in actual operation, due to the participation of robots, different tasks have multiple modes of operation, such as single-person disassembly, single-robot disassembly, and human–robot collaborative disassembly. Therefore, it is necessary to choose the most suitable mode of operation for each task. Ignoring this may lead to an unreasonable or unbalanced use of resources. Therefore, it is necessary to incorporate mode selection into the research of the DLB problem.

For DLB problems, as typical NP-hard problems, metaheuristic algorithms provide an effective way to solve them [8,9]. However, according to the "no free lunch" theorem in the optimisation field, no algorithm can be applied to all problems [10]. In response to this challenge, this paper proposes an efficient discrete bee algorithm to solve the proposed problem. In summary, this paper focuses on the human–robot collaborative DLB problem and constructs its multiobjective DLB model with the idle rate of the disassembly line as the goal. The proposed problem integrates factors such as disassembly-task allocation and mode selection. To efficiently solve the problem, this paper also develops an efficient multiobjective discrete bee algorithm (MOBDA) as an important tool.

The main contributions of this paper are as follows:

i. A multiobjective optimisation model is proposed for the human–robot collaborative DLB problem, aiming to minimise the number of opened workstations and the idle rate of the disassembly line and energy consumption;
ii. The proposed model integrates disassembly-task allocation and operation mode selection;
iii. A highly efficient multiobjective discrete bee algorithm is designed and implemented to solve the proposed problem, and it has been compared with other advanced algorithms, demonstrating its superiority.

The remainder of this paper is organised as follows: Section 2 provides a comprehensive literature review, Section 3 describes the problem discussed in this paper, Section 4 details the steps of the proposed algorithm, Section 5 presents a real-world case study, Section 6 investigates the effectiveness of MODBA through comparison with other state-of-the-art algorithms, and finally, there are the conclusions, limitations, and future research directions of this paper (Section 7).

2. Literature Review

Recently, scholars have delved deeply into the DLB problem. For instance, Dan et al. employed a particle-swarm optimisation (PSO) algorithm to ascertain DLB solutions for complex products [11]. Wang et al. constructed a DLB model that takes into account the harmful index, demand index, and the number of changes in disassembly operation direction and proposed an improved genetic algorithm to solve it [12]. Ren et al., based on the gravitational search algorithm, proposed solutions for a profit-oriented partial DLB problem [13]. Singh et al. introduced a quantum heuristic algorithm to optimise the profitability and workload balance of the disassembly line [14]. Wang et al. aimed to maximise workers' efficiency, increase profits, reduce energy consumption, and balance the workers' load; established a new DLB model based on economic benefits and environmental impact; and proposed a discrete multiobjective artificial bee colony algorithm to solve the proposed

model [15]. Yang et al. proposed a multiobjective fruit fly optimisation algorithm to achieve low carbon emissions, low energy consumption, and reduced costs in the disassembly process. They considered the DLB of old agricultural machinery and, finally, proved the feasibility of their scheme through examples [16]. Zhang et al. proposed using a hybrid graph to express the direct and indirect constraints between components to improve the efficiency and environmental friendliness of the disassembly line. They established a new multiobjective DLB optimisation model with the shortest disassembly time and the least carbon emissions. They also proposed an optimised disassembly sequence genetic algorithm to solve their model and, finally, proved the practicality of the proposed model through the example of a car engine [17]. Tian et al. established a multiobjective optimisation model to minimise the idle rate, cost, and energy consumption during the operation of the disassembly line. They proposed an improved discrete whale optimisation algorithm to solve their problem and compared it with existing algorithms, proving the high performance of the proposed algorithm [18]. Wang et al. proposed a priority graph based on the associated parts to address the issue of a large number of parts in waste electrical and electronic equipment and the complex relationships between parts. They established a DLB problem model that considered efficiency, profit, energy consumption, and risk. They also proposed an improved multiobjective genetic simulated annealing algorithm to solve the model and verified the rationality of the model through examples [19].

With the rise of Industry 5.0, collaborative robots brought new vitality to the efficient operation of the disassembly line. The human–robot collaborative DLB problem received some attention. For example, Wu et al. proposed a human–robot collaborative unit-level disassembly model, specifically for the hazardous and complex nature of the battery assembly process [20]. Xing et al. developed a linear mathematical model aimed at maximising the profit of human–robot collaborative disassembly lines and verified the effectiveness of the model [21]. Liu et al. established a human–robot collaborative model for the DLB problem. They classified and allocated tasks for human–robot collaborative disassembly and proposed an improved discrete bee algorithm (BA) to solve the human–robot collaborative model. They proved the effectiveness of the method through examples [22]. Xu et al. established a human–robot collaborative disassembly information model that considered safety strategies to ensure the safety of operators when robots and operators collaborate to perform disassembly tasks. They proposed an improved discrete BA to solve the human–robot collaborative DLB problem and proved its rationality with examples [8]. Liu et al., in response to the risk of manual disassembly that did not ensure personal safety, proposed a human–robot hybrid disassembly model. They proposed an improved Q-learning algorithm to solve the human–robot hybrid disassembly model [23].

Additionally, the disassembly sequence planning (DSP) problem, which is closely related to the DLB problem, has also been extensively explored. Rehal and Sen proposed an assembly partitioning scheme based on component accessibility, aiming to improve the efficiency of DSP [24]. Bahubalendruni and Varupala [25] proposed an automated optimal disassembly sequence planning method to address the unsafe handling of electronic and electrical equipment and improve disassembly efficiency. To design safer and more reliable EOL product disassembly sequences, Wang et al. introduced a novel DSP method based on spatial matrices [26]. Gulivindala et al. [27] introduced a new environmental risk-reduction model for managing the e-waste generated in the medical sector due to COVID-19, with a particular emphasis on reducing human and environmental toxicity risks through scientific disassembly strategies. The model categorises materials using a scoring system and designs objective functions to disassemble high eco-toxicity and low human toxicity materials in the fewest layers possible. Additionally, it suggests disassembly operations to avoid the direct exposure of workers to toxic materials by assessing threshold limits. Feng et al. proposed an improved DSP method that integrates the remanufacturing characteristics of EOL products, enhanced QICA (quality, cost, delivery time, reliability, flexibility, and environmental impact), and expert decision reliability measurement and updating into DSP. Finally, the feasibility of the procedure was demonstrated through a case study [28].

Xie et al. presented an improved grey-wolf optimiser to address a DSP problem. The effectiveness of the proposed algorithm was validated by applying it to two engineering cases [29]. Anil Kumar et al. [30] proposed a multi-level approach to extract valuable materials from EOL products while separating toxic elements without exceeding threshold limits. The effectiveness of the proposed method was evaluated by comparing it with existing DSP methods.

Based on the aforementioned literature, insights could be drawn:

i. Most current research has considered single-person DLB problems, and human–robot collaborative disassembly has received less attention. Although some researchers have researched human–robot collaborative DLB problems, most of them have been limited to having only one person or robot in a workstation, and less consideration has been given to situations where people and robots are in the same position. And, there has been a lack of research on choosing different operation modes for disassembly tasks at the same workstation. At the same time, human–robot collaborative DLB problems, considering energy consumption, have also been relatively scarce;
ii. Metaheuristic algorithms, with their excellent search capabilities and adaptability, have played an important role in solving DLB and DSP problems. However, according to the "no free lunch" theorem in the optimisation community, there had been no algorithm that could be applied to solve all problems. Therefore, it had been particularly important to develop and improve new algorithms according to the specific characteristics of the problem to improve the applicability and efficiency of the algorithm.

In summary, this paper innovatively proposes a framework for the human–robot collaborative DLB problem. In this framework, not only is the task-assignment optimisation problem considered in depth but also the operation modes of different workstations are comprehensively considered, covering a variety of scenarios such as independent operation by a human, independent operation by a robot, and collaborative operation by a human and a robot. In addition, energy consumption is considered as one of the optimisation objectives. To effectively solve this complex problem, we have designed a novel discrete bee algorithm, which aims to improve the solution efficiency and ensure the solution quality, thus providing a new solution to the DLB problem in practical applications.

3. Problem Description

In this section, we first introduce the representation of disassembly information and the matrix transformation approach (Section 3.1). Then, we discuss the human–robot collaboration DLB problem addressed in this paper (Section 3.2). Finally, we construct the proposed multiobjective model (Section 3.3).

3.1. Disassembly-Precedence Graph

The disassembly-precedence graph is a model specifically used to describe the structure of EOL products [1,2]. It graphically presents the hierarchical relationships and constraints between the disassembly tasks of EOL products in detail. This model not only helps us understand the complexity of product disassembly but also provides an important basis for formulating effective disassembly strategies.

This model contains two core elements: the basic disassembly unit set and the directed edge set. The basic disassembly unit set consists of various independent tasks during the product disassembly process, with each task resulting in the removal of a specific part. The task number corresponds to the part number being removed. The directed edge set clearly expresses the order and dependencies between these disassembly tasks, ensuring the orderliness and rationality of the disassembly process [2].

Figure 1 illustrates a 3D model of a partial structure of a gear transmission device, which comprises four parts. It can be observed that, once the shaft (1) is removed, the key (2) or the flywheel (3) can be taken out. After the flywheel (3) is removed, the counterweight (4) can be disassembled, completing the entire disassembly process. Its

disassembly-precedence graph is shown in Figure 2. In Figure 2, disassembly tasks 1, 2, 3, and 4 constitute the basic disassembly unit set. There are clear precedence constraints between tasks 1 and 2, tasks 1 and 3, and tasks 3 and 4; these relationships are represented by the directed edge set. For example, task 1 must be completed before task 2, the start of task 3 depends on the completion of task 1, and task 4 needs to be carried out after task 3.

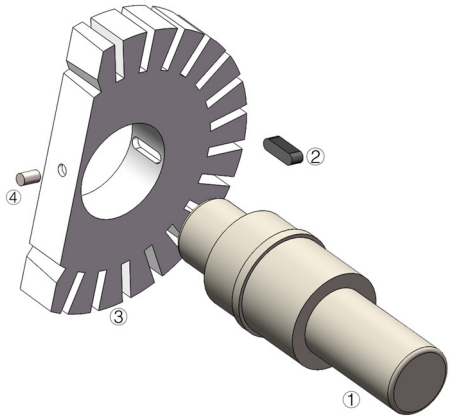


Figure 1. Partial structure of a gear unit.

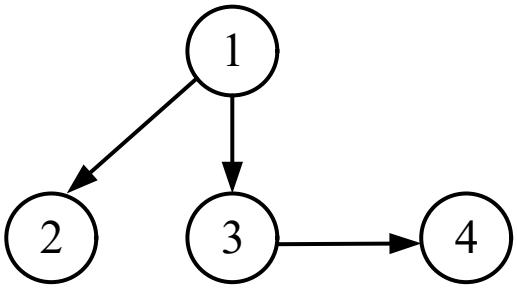


Figure 2. Disassembly-precedence graph.

The structured expression of the disassembly-precedence graph allows us to intuitively understand the interrelationships and sequence dependencies between multiple disassembly tasks of a product, which is crucial for ensuring the smooth progress of disassembly work. This model has an important application value for product disassembly planning. It not only helps to optimise the disassembly process and improve production efficiency but also promotes the maximisation of resource utilisation and reduces waste.

To apply the disassembly-precedence graph in the modelling process, it needs to be converted into a matrix form, which can be defined as the precedence matrix P. The disassembly-precedence graph in Figure 2 can be transformed into the following form:

$$\begin{bmatrix} p & 1 & 2 & 3 & 4 \\ 1 & 0 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 & 0 \\ 3 & 1 & 0 & 0 & 0 \\ 4 & 0 & 0 & 4 & 0 \end{bmatrix}$$

3.2. Human–Robot Collaboration DLB Problem

In the disassembly process, the rational allocation of disassembly tasks to achieve the set DLB objectives is key to improving the efficiency and production capacity of the disassembly line. Therefore, the primary goal of this paper is to rationally allocate disassembly tasks to various workstations. In addition, with the introduction of robots, the operators of the workstations are classified into single-person operators, single-robot operators, and human–robot collaborative combination operators, as shown in Figure 3.

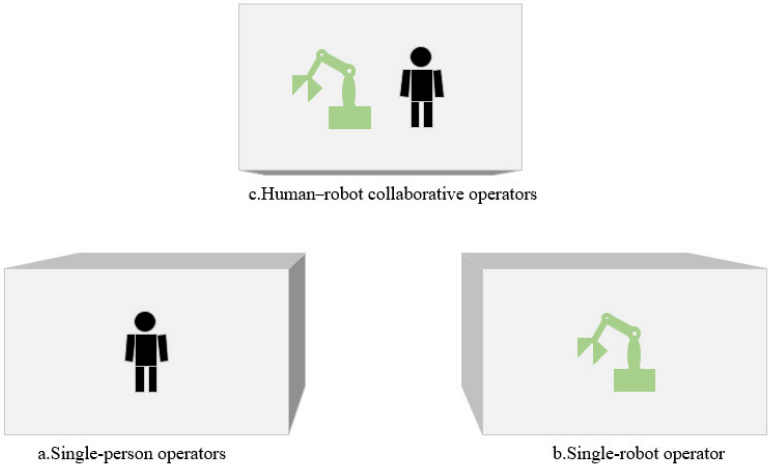


Figure 3. Schematic of the operation models.

(i) Single-person operators have a high degree of flexibility and can adjust the operation mode and steps according to the specific requirements of the disassembly task;

(ii) Single-robot operators have a high degree of automation, do not require human intervention, and are suitable for highly repetitive disassembly tasks;

(iii) Human–robot collaborative combination operators can combine the judgment ability of humans and the automation advantages of robots, effectively reducing disassembly time and improving disassembly efficiency.

For example, in the disassembly process of a specific power battery, the upper casing can be disassembled through three different methods: manual disassembly, robot disassembly, and human–robot collaborative disassembly. Manual disassembly is performed by workers, relying on their skills and experience, and requires a high level of technical proficiency. Typically, this process takes 180 s. Robot disassembly is a highly automated method where robots operate according to pre-set programs and paths, completing the task in just 90 s, significantly enhancing production efficiency. Human–robot collaborative disassembly combines the advantages of both workers and robots. In this mode, robots handle high-precision, repetitive parts, while workers manage parts that require flexible responses. This cooperative approach allows the disassembly of an upper casing in just 70 s. Evidently, this approach is optimal for tasks that have a high potential for automation and pose no harm to humans. This method can significantly improve disassembly speed and better handle complex situations, thereby reducing the error rate during the disassembly process.

The problem studied in this paper is a human–robot collaborative DLB problem, which involves not only implementing task allocation but also selecting the most suitable operation mode for these tasks. Therefore, to assess the automation potential of disassembly tasks (whether they are suitable for execution by robot operators), this paper adopts the method outlined in [31]. Specifically, it employs 10 attributes: five for the necessity to automate the corresponding disassembly operation (*NA*) and five for the technical ability of a disassembly process to be automated (*TAA*). These attributes comprehensively consider the complexity of the disassembly task, the associated hazards, the value of the components, and other multidimensional factors. The weights of these attributes are equal. The detailed scoring criteria are listed in Table 1. The values of *NA* and *TAA* are calculated based on the scores of each attribute, with the specific formulas as follows:

$$NA = 10 \times \sum_{i=1}^{5} NA_i \tag{1}$$

$$TAA = 10 \times \sum_{i=1}^{5} TAA_i \tag{2}$$

where NA_i and TAA_i represent the scores for each criterion, with each criterion's score ranging from $[-2,\ 2]$. The total score range is, therefore, $10 \times \left[\sum_{i=1}^{5} -2,\ \sum_{i=1}^{5} 2\right] = [-100,\ 100]$.

Table 1. *NA* and *TAA* evaluation criteria.

Category	Serial Number	Attribute	Scoring Criteria 2	1	0	−1	−2
NA	1	Number of Actions (People)	Many	Quite a Few	Moderate	Few	Very Few
	2	Manual Disassembly Time (Seconds)	Very Long	Long	Average	Short	Very Short
	3	Weight	≥25 kg	<25 kg	<15 kg	<10 kg	<10 kg
	4	Priority (Value)	Very High	High	Average	Low	Very Low
	5	Hazards (High Voltage Protection, Hazardous Materials)	High Voltage and Chemical Hazards, Sharp Edges	High Voltage or Chemical Hazards, Sharp Edges	High Voltage or Chemical Hazards, No Sharp Edges	No High Voltage or Chemical Hazards, Only Sharp Edges.	No High Voltage or Chemical Hazards, No Sharp Edges
TAA	1	Action Complexity (Robot)	Simple Standard Actions	Moderate Number of Actions	More Complex Actions	Even More Complex Actions	Very Complex Actions
	2	Detection Feasibility	Clear View, Good Contrast	Clear View, Poor Contrast	Partially Hidden, Poor Contrast	Partially Hidden, Shadowed	Completely Hidden
	3	End Effector Access	Completely Open	Open with Size Limitations	Extended End Effector Required	Small Tool Required	No Access
	4	Material Handling	Simple fasteners are collected into a metal box for easy further recycling	Parts are metal only but are small or medium-sized	Different materials cannot be sorted, or the parts are very large	Parts are large and involve different materials, making recycling difficult	Parts are very large, bulky, or involve hazardous materials
	5	End Effector Automation Potential	Multiple Automated Tool Options	Some Existing Automated Tool Options	Few Existing Automated Tool Options	Automation Concept Exists, Not Yet Implemented	No Automation Concept

Based on the calculated NA and TAA scores, each disassembly task is classified and assigned to either workers or robots. When $NA > 0$ and $TAA \geq 50$, the task needs to be automated and is assigned to robots. When $NA \leq 0$ and $TAA \geq 50$, the task can be performed in one of three ways. In other cases, the task has no automation potential and must be executed by humans.

In summary, the problem studied in this paper can be summarised as, under the premise of satisfying various characteristic constraints, rationally allocating each task to the disassembly workstation and selecting the appropriate operation mode for these tasks. Finally, Figure 4 shows a framework for the operation of a human–robot collaborative disassembly line.

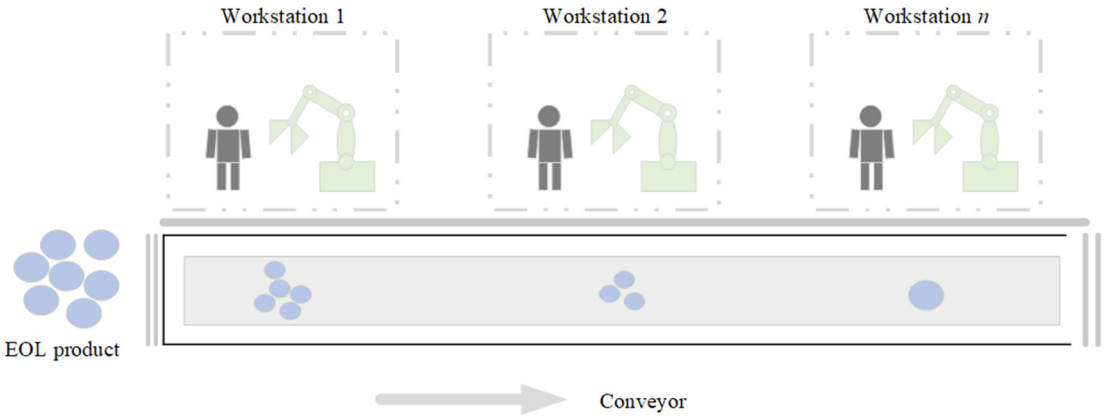


Figure 4. Human–robot collaboration disassembly line.

3.3. Problem Formulation

This paper constructs the proposed human–robot collaborative DLB model based on the following assumptions:

- The product to be disassembled is complete with all parts;
- The time for workers and robots to disassemble each product part is known;
- The product supply is uninterrupted;
- Uncertain factors in the disassembly process are ignored;
- Each disassembly workstation has only one human worker and one robot.

In addition, before formulating the proposed model, the following notations are introduced:

Indices:

m, j	Disassembly-task index, $m, j \in \{1, 2, \ldots, M\}$
n	Workstation index, $n \in \{1, 2, \ldots, N\}$
p	Operation mode index for task execution, $p \in \{1, 2, \ldots, P\}$, p being 1, 2, 3, respectively for single-person operation, single-robot operation, and human–robot collaborative operation

Parameters:

M	Number of disassembly tasks
N	Maximum number of workstations
P	Number of task operation modes
l_{mp}	Execution time of task m in mode p
c_{mp}	Energy consumption of task m in operation mode p
t_m	Operation time of task m
e_w	Fixed energy consumption per second during the operation of the workstation
e_m	Operation energy consumption of task m
a_{mj}	Precedence matrix

Decision variables:

h_{mp}	1, if task m can be executed in mode p, 0 otherwise
w_{mp}	1, if task m is executed in mode p, 0 otherwise
x_{mn}	1, if task m is assigned to workstation n, 0 otherwise
u_n	1, if workstation n is activated, 0 otherwise

Then, the multiobjective model constructed in this paper is as follows.

The first objective is to minimise the number of workstations opened. By reducing the number of workstations opened, the overall operating cost can be reduced [9].

$$minf_1 = \sum_{n=1}^{N} u_n \tag{3}$$

The second objective is to minimise the idle rate of the workstation. By minimising the idle rate, the workstation can be used for production or other tasks as much as possible, thereby improving the overall production efficiency and resource utilisation rate [1].

$$minf_2 = \frac{\sum_{n=1}^{N}\left(C_t u_n - \sum_{m=1}^{M} t_m x_{mn}\right)}{\sum_{n=1}^{N} C_t u_n} \tag{4}$$

$$t_m = \sum_{p=1}^{p} l_{mp} w_{mp} \tag{5}$$

The third objective is to minimise disassembly energy consumption. In the current context of increasing concern over environmental impact and energy costs, minimising energy consumption is a crucial goal in industrial production. By optimising energy usage, companies can achieve significant cost savings, which is particularly important given the rising energy prices. Additionally, minimising energy consumption aligns with global environmental sustainability goals. Lower energy consumption means reduced greenhouse-gas emissions, helping to mitigate climate change. In this article, we have chosen to use power as the metric for measuring energy consumption, which involves two main aspects, namely the energy consumed by the machinery or tools during task execution and the

ongoing energy consumption of the workstation when it is active, including electricity used for lighting, ventilation, equipment standby modes, and monitoring systems.

$$minf_3 = \sum_{n=1}^{N} C_t e_w u_n + \sum_{n=1}^{N} \sum_{m=1}^{M} e_m x_{mn} \tag{6}$$

$$e_m = \sum_{p=1}^{p} c_{mp} w_{mp} \tag{7}$$

The above objectives are subject to the following constraints:
s.t.

$$\lceil \sum_{m=1}^{M} t_m / C_t \rceil \leqslant \sum_{n=1}^{N} u_n \leqslant M \tag{8}$$

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

$$\sum_{p=1}^{P} w_{mp} = 1, \ \forall m \in \{1, 2, \ldots, M\} \tag{10}$$

$$w_{mp} \leq h_{mp}, \forall m \in \{1, 2, \ldots, M\}, \forall p \in \{1, 2, \ldots, P\} \tag{11}$$

$$\sum_{n=1}^{N} (n x_{mn}) \leqslant \sum_{n=1}^{N} (n x_{jn}), \forall a_{mj} = 1 \tag{12}$$

$$\sum_{m=1}^{M} t_m x_{mn} \leq C_t, \forall n \in \{1, 2, \ldots, N\} \tag{13}$$

$$h_{mp}, w_{mp}, x_{mn}, u_n \in \{0, 1\} \tag{14}$$

Constraint set (6) limits the number of open workstations. Constraints (7) and (8) ensure that each task is assigned to a single workstation and can only be executed by one operational mode. Constraint set (9) stipulates that tasks can only be operated in modes in which they are executable. Constraint set (10) maintains the restrictions on task priority order, and constraint (11) ensures that the total task time at each workstation does not exceed the maximum cycle time limit. Finally, constraint set (12) defines the binary variables used in the model, ensuring that all decision variables are either 0 or 1.

4. Proposed Solution Method

The DLB problems are confirmed as NP-hard, presenting significant challenges for both exact and heuristic solution algorithms, particularly when the problem size is large. As a result, researchers turn to metaheuristic algorithms as an effective tool for tackling such complex issues. In this paper, we choose the BA as our solution approach. BA is selected because it simulates the natural foraging behaviour of bees, utilising swarm intelligence to balance local and global searches, effectively exploring the solution space. Its adaptability allows the algorithm to adjust strategies based on feedback from the search process, better adapting to the complexity of the DLB problem. Moreover, BA is easy to implement, has simple parameter adjustments, and is supported by previous research demonstrating its effectiveness in solving similar optimisation problems [32,33]. Therefore, based on these advantages, BA becomes our preferred metaheuristic algorithm for addressing the problem presented.

In this section, we first describe the customised population initialisation process, ensuring the diversity and quality of the initial population, which provides a solid foundation for the algorithm's search process (Section 4.1). We then detail the division of bee roles, drawing on the social behaviour patterns of bees in nature, which effectively guides the search process through a clear division of labour and cooperation (Section 4.2). Further-

more, this paper carefully designs a series of operators, which are key in the solution process of the algorithm and are responsible for improving existing solutions to achieve better optimisation outcomes (Sections 4.3 and 4.4). We follow with an explanation of the constraint correction strategy, which is crucial for ensuring that all solutions meet the constraints (Section 4.5). Next, we discuss the population update method and the algorithm's stopping criteria (Section 4.6). After thoroughly discussing each component of the MODBA algorithm, we provide a comprehensive framework that showcases the algorithm's flow and key components and their interrelationships (Section 4.7).

4.1. Population Initialisation

In metaheuristic algorithms, population initialisation is a key step for ensuring the algorithm's effectiveness [34]. To generate feasible initial populations, we use a dual-layer encoding mechanism to represent the problem's solution. In this mechanism, the first layer of encoding uses integers to represent each task, and the order of these integers clearly defines the execution order of the tasks. For example, the sequence one, two, three, four indicates that the tasks will be disassembled in the order of one, two, three, and four. To meet the precedence constraints between tasks, we first transform the disassembly-precedence graph into matrix P, and then use the following process to generate a feasible disassembly sequence:

Step 1: identify tasks with all zeroes in their rows in matrix P. These tasks have no prerequisite tasks and can be executed immediately;

Step 2: randomly select a task from the executable tasks and add it to the disassembly sequence;

Step 3: remove the row corresponding to the task from matrix P and set all elements in the task's column to zero, indicating that the task has been completed and subsequent tasks no longer depend on it;

Step 4: Repeat steps 1 to 3 until all tasks have been added to the disassembly sequence, forming a complete sequence that follows the precedence constraints.

We then further generate the second layer of encoding, which defines the execution mode for each task. For example, if the second layer of encoding for a task is one, it means that the task is executed manually, as described in Section 3.3. For tasks that have multiple execution modes to choose from, we determine the final mode through a randomised method.

After forming a feasible disassembly sequence and determining the execution mode of its tasks, we proceed to the decoding phase, which assigns disassembly tasks to various workstations and opens new workstations when the remaining time at a workstation is less than the task execution time until all tasks have been allocated. After the decoding is completed, the generation process of a feasible individual is finished.

4.2. Scout-Bee Role Categories

Similar to the traditional BA, the assignment of scout-bee roles is crucial in the initial phase of the proposed MODBA [32]. Faced with the complex challenge of multiobjective optimisation, we first set an initial population consisting of g_{size} scout bees (individuals). These initial individuals then undergo a comprehensive evaluation process that includes fast non-dominated sorting and crowding-distance calculation [35].

Fast non-dominated sorting is a method used in multiobjective optimisation problems to evaluate the quality of individuals, comparing their performances across all objective functions to determine each individual's dominance rank. In this process, individuals are classified based on whether they are dominated by other individuals across all objectives. If an individual performs better than another on at least one objective and is not worse on the others, it is considered to dominate the latter. Fast non-dominated sorting allows us to categorise the g_{size} solutions into different Pareto front levels.

Next, the concept of crowding distance is introduced to assess the diversity of individuals within the same dominance rank. After calculating the crowding distance for each

individual, they are sorted based on the size of their values; a larger crowding distance means that the individual has a more unique position within its vicinity, indicating that the solution provides greater diversity in the objective space.

After fast non-dominated sorting and crowding-distance calculation, we select the top *NS* scout bees as optimal scout bees, the next *ES* scout bees as better scout bees, and the remaining scout bees are classified as random scout bees. Although they have not yet shown a clear advantage, they play an indispensable role in maintaining population diversity and promoting the algorithm's global search capability. Through this hierarchical screening and role-assignment mechanism, the MODBA algorithm can efficiently focus resources on the most promising solutions while maintaining population diversity, laying a solid foundation for finding a balanced set of solutions in multi-objective optimisation problems.

4.3. Search Phase of Optimal Scout Bees

After the scout bees have been assigned roles, MODBA transitions to the search phase for optimal scout bees (forager bees search for nectar around the optimal scout bees). In this phase, each optimal scout bee carries *OF* forager bees, namely, each scout bee conducting *OF* neighbourhood searches. We introduce two main neighbourhood search operators. The first updates the assembly sequence in the original solution (S1), and the second updates the execution mode of each task in the original solution (S2).

(i) Changing S1

In this phase, we use the PPX operator, which efficiently addresses the sequence crossover process with precedence relationships. After two parent individuals perform the PPX operator, the offspring individual obtained does not have any duplicate or missing information. More importantly, the offspring individuals still meet the precedence constraints, eliminating the need for further adjustment. The PPX operator, first, randomly generates an execution sequence, the length of which is equal to the individual, and the elements contain only two values, indicating which parent each gene to be crossed comes from. According to the execution sequence, from front to back (left to right), select the first gene from the specified parent and insert it into the offspring, while clearing the inserted gene from both parents. When the last gene specified by the execution sequence is inserted into the offspring, the PPX crossover process ends, and at this point, the two parent individuals carry no gene information. Figure 5 illustrates a PPX process.

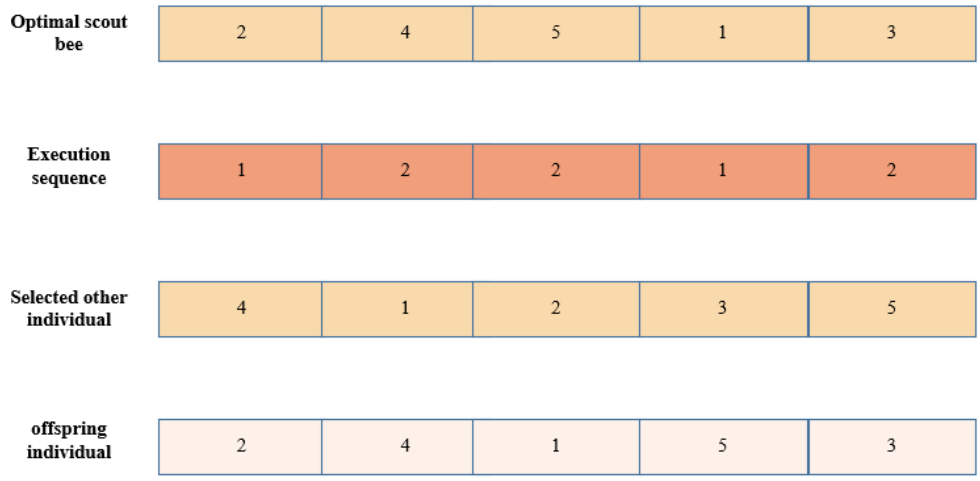


Figure 5. PPX operator.

During the first iteration of MODBA, each optimal scout bee randomly pairs with an individual from the current first Pareto front level to generate new offspring. In subsequent iterations, it changes to randomly selecting an individual from the external archive as the other parent. Details about the external archive are described in Section 4.6.

(ii) Changing S2

S2 involves changing the execution mode of each task. Specifically, we select Q% of tasks that can change their work mode and change their mode of operation to the shortest time required. If the current mode of operation is already the shortest, it remains unchanged.

4.4. Search Phase for Better Scout Bees

Next, MODBA enters the search phase for better scout bees, where in this phase, each better scout bee carries NF forager bees. These forager bees conduct nectar collection (neighbourhood search) around each better scout bee, enabling each better scout bee to generate new solutions. Similarly, this phase establishes two types of neighbourhood search operators.

(i) Changing S1

We randomly select 30–70% of the disassembly tasks. Then, we rearrange them using a wave sorting method and insert them to the far right of the original sequence. Wave sorting is a special sorting method that arranges the elements of an array into a wave pattern. Specifically, for an array $[a_0, a_1, a_2, \ldots, a_{n-1}]$, we make each element at an odd index less than or equal to its adjacent even index, that is, $a_0 \leq a_1, \geq a_2 \leq a_3 \geq a_4 \leq \cdots$. Finally, Figure 6 illustrates this strategy.

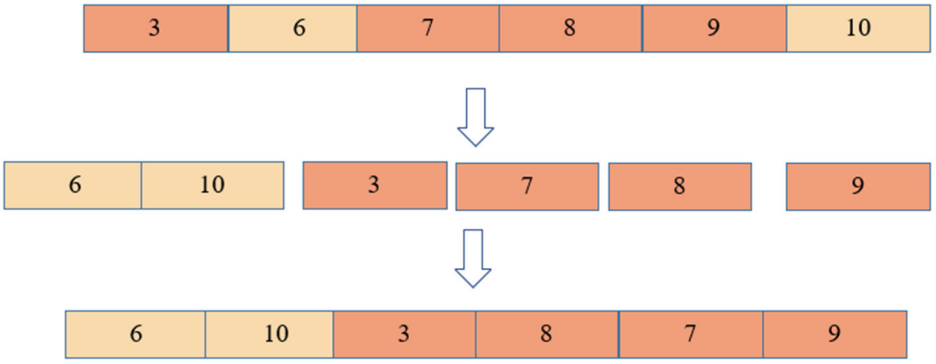


Figure 6. Wave pattern operator.

(ii) Changing S2

In phase S2, we begin by selecting the individual through a crowding-distance mechanism from the current iteration. Subsequently, the execution modes of the tasks for the current better scout bees are adjusted using a uniform crossover method. For instance, if task 1 is executed in mode 1 in the current better scout bee and in mode 2 in the optimal individual, we set a probability threshold p_u. If a random number generated is less than p_u, then we update the execution mode of task 1 in the current better scout bee to mode 2. If the execution modes of the two tasks are already the same, we skip this task and continue with the process until all tasks have undergone the necessary updates.

4.5. Methods for Restraint Correction

New solutions generated by metaheuristic algorithms may violate constraints. MODBA is no exception. In our proposed problem, since S2 has already met the constraints during the update process, we only need to adjust S1. For the newly generated disassembly sequence, we start checking from the first task; if it meets the execution constraints, we proceed to check subsequent tasks. Otherwise, we select an executable task to replace it and continue this process until all tasks have been inspected.

4.6. Population Update and Stop Iteration

Once all the scout bees have completed their search, the newly generated solutions are merged with the original scout-bee population. Then, a fast non-dominated sorting and a crowding-distance calculation are executed to update the scout-bee population and assign new scout-bee roles. Additionally, the non-dominated solution set is stored in an

external archive that is updated after each iteration. After reaching the maximum number of iterations *Maxit* or a set stopping time, the non-dominated solution set in the external archive is output.

4.7. Algorithm Framework

Based on the components mentioned above, the overall process of the MODBA is as follows:

Step 1: execute the population initialisation process, according to the strategy in Section 4.1;

Step 2: perform fast non-dominated sorting and crowding-distance calculation, according to the strategy in Section 4.2, and assign scout-bee roles;

Step 3: execute the optimal scout-bee foraging process according to the strategies in Section 4.3, generating new individuals;

Step 4: execute the better scout-bee foraging process according to the strategies in Section 4.4, generating new individuals;

Step 5: random scout bees perform a global search process, generating entirely new individuals;

Step 6: combine the newly generated individuals with the original scout-bee population, perform fast non-dominated sorting and crowding-distance calculation, and assign new scout-bee roles;

Step 7: Determine whether the algorithm has reached the maximum stopping criterion; if it has, output the final non-dominated solution set. If not, return to step 3.

In addition, we provide a flowchart of MODBA, as shown in Figure 7, which describes in detail the steps from algorithm initialisation to the final stop.

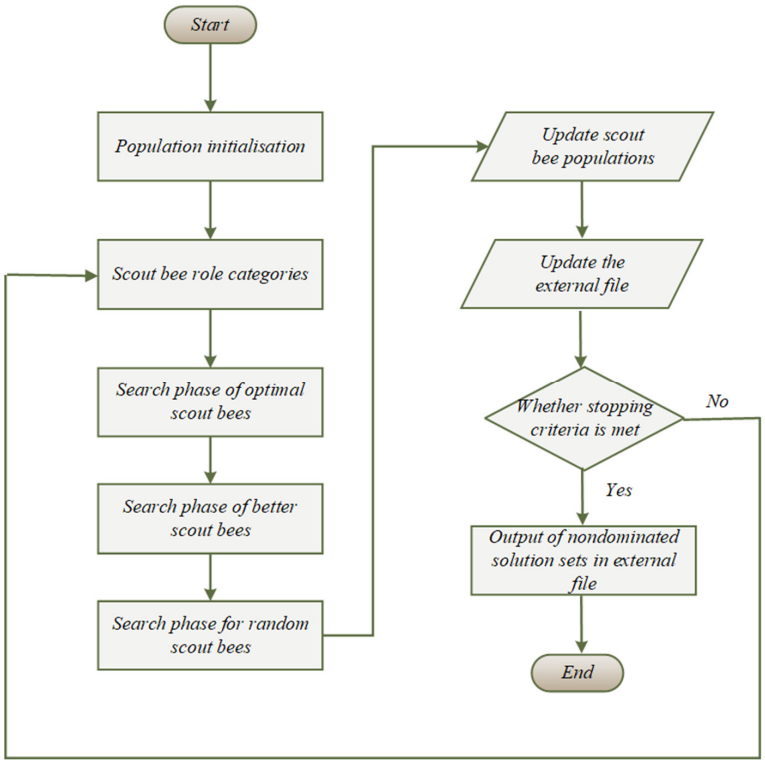


Figure 7. MODBA flowchart.

5. Case Study

To validate the effectiveness of the proposed model and algorithm, this section applies a Tesla Model1s power battery system disassembly example to the human–robot collaborative DLB problem discussed in this paper, and MODBA is used to solve it. Due

to the outdated design and long service life of the Tesla Model1s, the scrapping volume of its power battery system is experiencing explosive growth, thus, it is selected as the research object in this paper. A Tesla Model1s power battery system comprises 16 battery modules, a battery management system, and essential power electronic devices. Figure 8 shows its 3D model. Additionally, based on the product-assembly relationships and spatial position constraints, a disassembly-precedence graph of the Tesla Model1s power battery system tasks can be obtained, as shown in Figure 9, with specific disassembly information presented in Table 2.

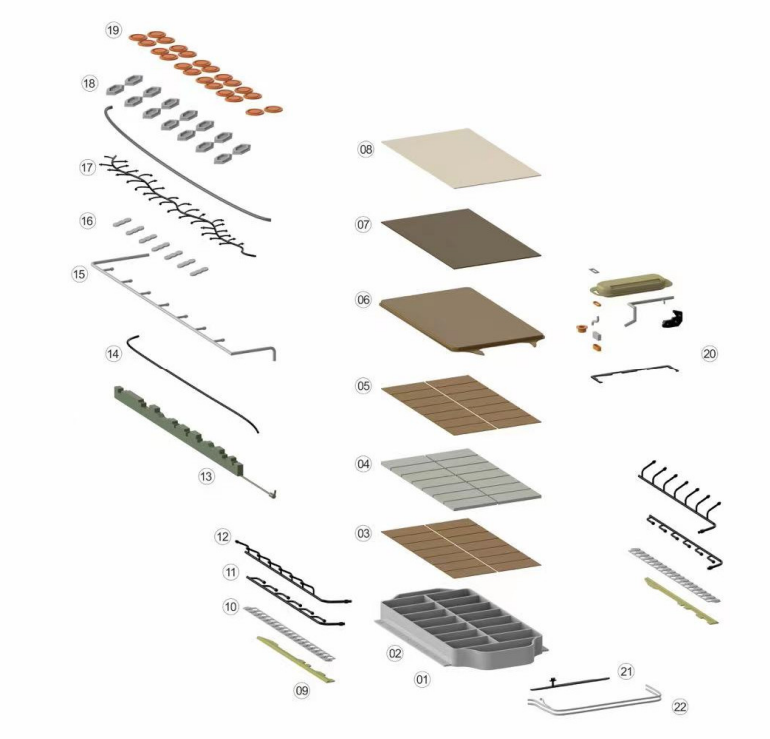


Figure 8. 3D model of the Tesla Model 1s power battery system.

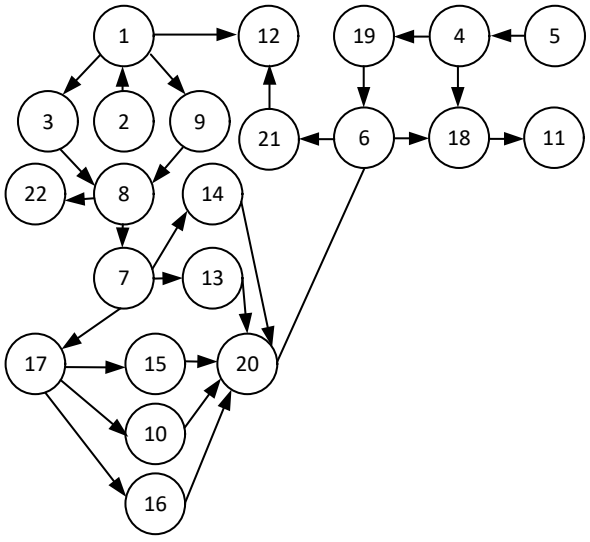


Figure 9. Disassembly-precedence graph of the Tesla Model 1s power battery system.

Table 2. Tesla Model 1s power battery system disassembly information.

Disassembly Task	Description	Quantity	Specifications and Dimensions	e_{m1}	e_{m2}	e_{m3}	l_{m1}	l_{m2}	l_{m3}
1	Upper Shell	1	685 mm × 300 mm × 80 mm	39.60	18.90	31.59	180	90	70
2	Upper Shell Screws	8	M8 × 20 mm	11.22	5.67	9.48	51	27	21
3	Sound Insulation Cotton	1	780 mm × 350 mm × 90 mm	9.90	5.67	9.48	45	27	21
4	Upper Shell of HV Assembly	1	780 mm × 350 mm × 90 mm	6.60	5.04	8.42	30	24	19
5	Shell Screws	12	M10 × 25 mm	9.90	5.67	9.48	45	27	21
6	Fuses	6	Bussmann 170M4415, 550 A 690 V	5.28	3.15	5.27	24	15	12
7	Battery Module	6	3100 mAh, 3.6–3.88 V, 18.6 mm × 65.2 mm	13.20	7.56	12.64	60	36	28
8	Busbar	1	500 A, 310 V	9.90	5.04	8.42	45	24	19
9	Metal Partition	2	500 mm × 300 mm × 2 mm	9.24	6.30	10.53	42	30	23
10	Insulation Pad	6	490 mm × 290 mm × 1 mm	5.94	3.78	6.32	27	18	14
11	Fiberboard	6	485 mm × 285 mm × 1 mm	3.30	1.89	3.16	15	9	7
12	Cooling Pipeline	1	20 mm diameter, 1000 mm length	10.56	6.93	11.58	48	33	26
13	Charging Port Connector	1	Complies with SAE J1772 standard [36]	6.60	4.41	7.37	30	21	16
14	Battery Harness	1	1500 mm length	7.26	5.04	8.42	33	24	19
15	Battery Management System	1	327 mm × 99 mm × 32 mm	3.30	2.76	4.21	15	12	9
16	Battery System Harness	1	1500 mm length	8.58	5.52	8.42	39	24	19
17	Module PCB Board	1	139 mm × 67 mm × 16 mm	7.50	5.52	8.42	30	24	19
18	Fiberboard Screws	8	M6 × 16 mm	6.00	3.45	5.27	24	15	12
19	High Voltage Electric Fuse	1	TE EV200, 2000 A, 320 V	3.75	1.38	2.11	15	6	5
20	Battery Cell	12	NCR 18650, 3100 mAh, 3.6–3.88 V, 18.6 mm × 65.2 mm	21.00	11.04	16.85	84	48	37
21	Coolant	1	5 L	11.25	8.28	12.64	45	36	28
22	Hazardous Battery Pack	1	500 mm × 300 mm × 200 mm	9.00	5.52	8.42	36	24	19

In addition to the information in Table 2, after evaluation, tasks 4, 6, 11, and 22 are mandatory for assignment to the robot to perform.

5.1. MODBA Parameter Calibration

Before executing the MODBA search process, a crucial step is to determine its reasonable parameters. Proper parameter settings can significantly enhance the search efficiency and stability of the algorithm. Based on detailed preliminary experimental results and a literature analysis [32–35], we first set several reference levels for the parameters of MOBDA, as shown in Table 3.

Table 3. Reference levels for MODBA parameters.

Parameters	Level 1	Level 2	Level 3
$Maxit$	50	80	100
g_{size}	30	40	50
NS	6	8	10
ES	3	4	5
OF	4	5	6
NF	4	5	6
Q	60	70	80
P_u	0.7	0.75	0.8

Based on the reference levels in Table 3, conducting a full factorial experiment would require $3^8 = 6561$ experiments. Such a scale of experimentation undoubtedly demands substantial computational resources and time, posing significant challenges to the feasibility and cost-effectiveness of the experiment. To effectively optimise the experimental process and reduce the unnecessary computational burden, we introduced the Taguchi method. This is a statistical approach aimed at reducing the number of experiments and costs through systematic experimental design.

Additionally, we used the relative percentage deviation from the literature [34] as a benchmark to ensure the scientific validity and accuracy of the results. The final experimental results are shown in Table 4.

Table 4. Orthogonal experiment results.

Experiment No.	*Maxit*	g_{size}	*NS*	*ES*	*OF*	*NF*	*Q*	P_u	*RPD*
*L*1	1	1	1	1	1	1	1	1	0.11785
*L*2	1	1	1	2	2	2	3	3	0.21675
*L*3	1	1	1	3	3	3	2	2	0.16925
*L*4	1	2	3	1	2	3	1	2	0.11225
*L*5	1	2	3	2	3	1	3	1	0.13905
*L*6	1	2	3	3	1	2	2	3	0.12565
*L*7	1	3	2	1	3	2	1	3	0.05095
*L*8	1	3	2	2	1	3	3	2	0.10445
*L*9	1	3	2	3	2	1	2	1	0.12915
*L*10	2	1	3	1	3	2	3	2	0.09445
*L*11	2	1	3	2	1	3	2	1	0.11125
*L*12	2	1	3	3	2	1	1	3	0.10525
*L*13	2	2	2	1	1	1	3	3	0.17075
*L*14	2	2	2	2	2	2	2	2	0.13735
*L*15	2	2	2	3	3	3	1	1	0.01555
*L*16	2	3	1	1	2	3	3	1	0.11785
*L*17	2	3	1	2	3	1	2	3	0.11635
*L*18	2	3	1	3	1	2	1	2	0.11415
*L*19	3	1	2	1	2	3	2	3	0.12945
*L*20	3	1	2	2	3	1	1	2	0.13735
*L*21	3	1	2	3	1	2	3	1	0.08335
*L*22	3	2	1	1	3	2	2	1	0.05175
*L*23	3	2	1	2	1	3	1	3	0.17055
*L*24	3	2	1	3	2	1	3	2	0.09905
*L*25	3	3	3	1	1	1	2	2	0.08915
*L*26	3	3	3	2	2	2	1	1	0.11535
*L*27	3	3	3	3	3	3	3	3	0.08275

Subsequently, based on the results in Table 4, we calculated the *RPD* for each reference level of each parameter across all experiments to determine the optimal parameters for MOBDA (with the best values highlighted in **bold**). The final results are shown in Table 5.

Table 5. Mean *RPD* results.

Maxit	g_{size}	*NS*	*ES*	*OF*	*NF*	*Q*	P_u
0.1295	0.1294	0.1304	0.1038	0.1208	0.1227	**0.1044**	**0.0979**
0.1092	0.1136	**0.1065**	0.1387	0.1292	**0.1100**	0.1177	0.1175
0.1065	**0.1022**	0.1084	**0.1027**	**0.0953**	0.1126	0.1232	0.1298

Finally, the parameter settings for MODBA are as shown in Table 6.

Table 6. MODBA parameters setting.

Parameters	Value
Maxit	100
g_{size}	50
NS	8
ES	5
OF	6
NF	5
Q	60
P_u	0.7

5.2. Results and Discussion

After determining the parameters, MODBA successfully generated eight non-dominated solution sets, which are presented in Table 7.

Table 7. Line balancing solutions.

Order	Line Balancing Solutions	Operation Mode	f_1	f_2	f_3
1	[2, 17, 15, 10, 5, 1, 3] → [9, 16, 8, 22, 4, 7, 13] → [14, 19, 20, 6, 18, 11, 21, 12]	[3, 2, 3, 2, 3, 3, 3] → [2, 3, 2, 1, 1, 3, 3] → [2, 3, 3, 1, 2, 1, 3, 2]	3	0.0067	236.27
2	[17, 16, 2, 5, 4, 10, 1] → [3, 15, 9, 8, 7, 13] → [19, 6, 22, 18, 14, 11, 21, 20, 12]	[2, 3, 3, 3, 1, 3, 3] → [3, 2, 2, 2, 2, 3] → [2, 1, 1, 3, 2, 1, 3, 3, 3]	3	0.0050	238.23
3	[5, 2, 4, 19, 1, 6] → [3, 18, 9, 11, 17, 10, 21] → [16, 15, 8, 22, 7, 12, 13] → [14, 20]	[3, 3, 1, 3, 3, 1] → [2, 3, 2, 1, 2, 3, 2] → [3, 2, 3, 1, 3, 2, 2] → [2, 3]	4	0.2425	231.79
4	[2, 5, 1, 4, 17, 16] → [15, 19, 9, 10, 3, 6, 18, 21, 11] → [12, 8, 7, 13, 22, 14, 20]	[3, 3, 3, 1, 2, 3] → [2, 3, 2, 2, 3, 1, 3, 2, 1] → [3, 3, 3, 2, 1, 2, 3]	3	0.0450	232.01
5	[17, 15, 5, 2, 10, 4, 1] → [9, 16, 19, 3, 8, 7, 14, 6] → [13, 22, 18, 11, 21, 20, 12]	[2, 2, 3, 3, 3, 1, 3] → [2, 3, 3, 2, 3, 3, 3, 1] → [3, 1, 3, 1, 2, 3, 2]	3	0.0367	233.72
6	[2, 1, 9, 3, 5, 4] → [8, 7, 19, 13, 6, 14, 22, 17, 10] → [16, 18, 21, 11, 15, 20, 12]	[3, 3, 2, 3, 3, 1] → [3, 3, 2, 3, 1, 2, 1, 2, 3] → [2, 3, 2, 1, 3, 3, 2]	3	0.0133	234.85
7	[5, 4, 2, 17, 15, 19, 16, 10, 6, 18] → [1, 3, 11, 21, 12] → [9, 8, 7, 13, 22, 14, 20]	[3, 1, 3, 3, 2, 3, 2, 3, 1, 2] → [3, 3, 1, 2, 2] → [2, 3, 3, 2, 1, 3, 3]	3	0.0183	234.34
8	[2, 1, 9, 5, 17, 4] → [3, 15, 10, 19, 16, 6, 21, 18, 8] → [22, 12, 11, 7, 13, 14, 20]	[3, 3, 2, 3, 3, 1] → [3, 3, 2, 3, 2, 1, 2, 3, 3] → [1, 2, 1, 3, 3, 2, 3]	3	0.0267	233.75

Based on the results in Table 7, there exists a complex interplay between the objectives. For decision makers determined to minimise the number of operational workstations, all schemes except Scheme 3 show significant advantages. However, if the goal is to minimise energy consumption, Scheme 3 stands out with the lowest energy consumption. It is important to note, though, that Scheme 3 also has the highest number of operational workstations and a relatively high idle rate. For decision makers aiming for the lowest disassembly idle rate, Scheme 2 is particularly prominent. However, it is crucial to note that Scheme 2 does not perform well in terms of energy efficiency, which is a key factor that requires careful consideration and trade-off. Therefore, decision-makers must carefully balance the number of operational workstations, energy consumption, and disassembly idle rate to achieve the best overall benefits. The choice of the optimal scheme should not depend solely on the performance of a single indicator but must comprehensively consider the mutual influences and trade-offs between various objectives to ensure the comprehensiveness and feasibility of the final decision.

6. Algorithm Performance Analysis

In this section, we conduct an in-depth examination of the performance of MODBA, comparing it with other advanced algorithms from the literature that have been proven effective in solving line balancing and other discrete location-based optimisation problems. These include the non-dominated sorting genetic algorithm II (NSGA-II) [35], the multiobjective novel immune clonal algorithm (NICA-II) [37], the MOPSO algorithm [11], the improved gravitational search algorithm (GSA) [13], and the balanced-quantum inspired evolutionary algorithm (QEA) [14]. We perform a comprehensive analysis of the performances of different algorithms using three well-known multiobjective evaluation metrics: the number of Pareto solutions (NPS), inverted generational distance (IGD), and hypervolume (HV). The specific descriptions are as follows [34]:

NPS: This metric measures the total number of Pareto optimal solutions an algorithm can find. In multi-objective optimisation, Pareto optimal solutions refer to solutions that are not dominated by any other solutions in all objectives, and at least one objective is better. A higher NPS indicates that the algorithm can explore more of the Pareto front;

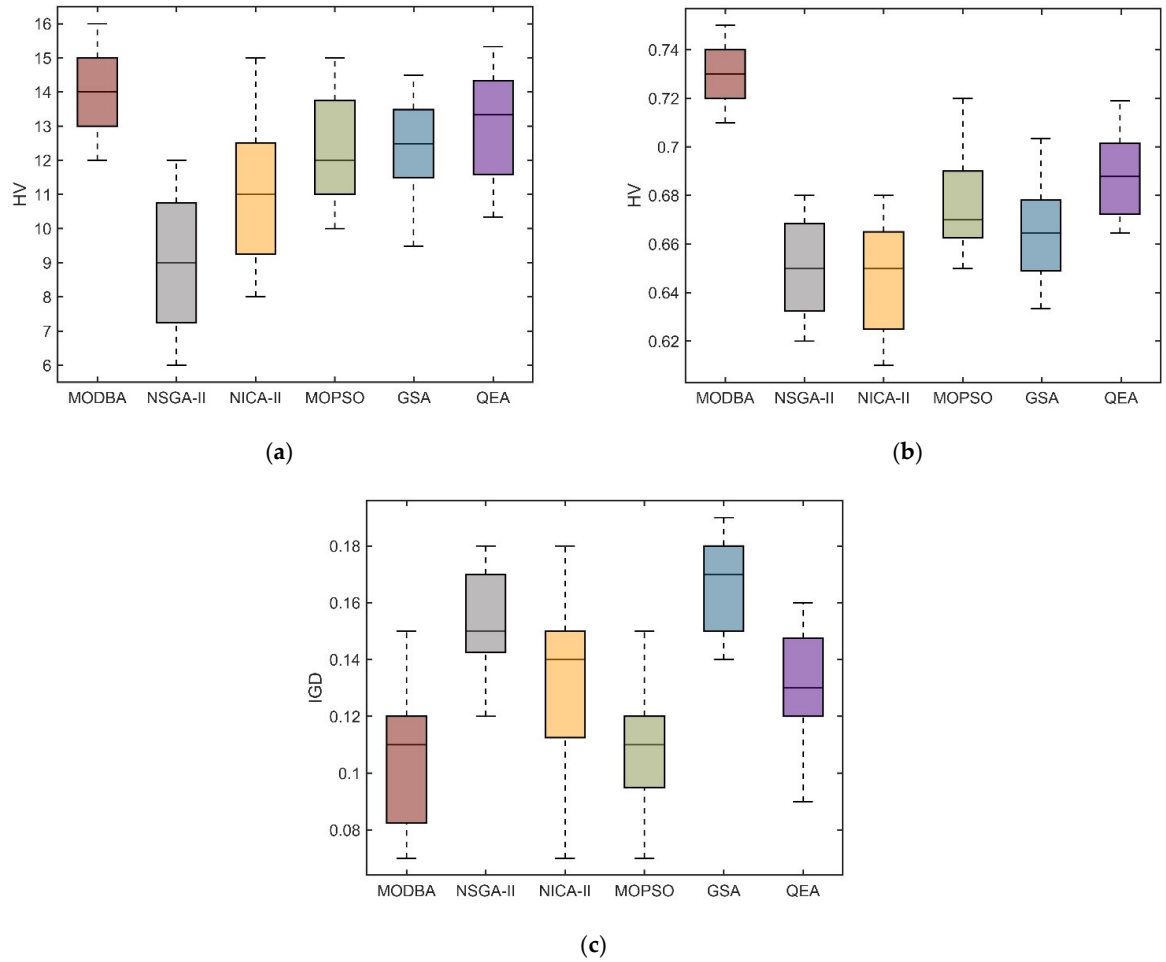
IGD: This metric measures the closeness of the Pareto front generated by the algorithm to the true Pareto front. It is assessed by calculating the sum of the distances from each solution in the algorithm's set to the nearest true Pareto solution. A smaller IGD indicates that the set of solutions generated by the algorithm is closer to the true Pareto front;

HV: The hypervolume metric measures the size of the area covered by the Pareto front generated by the algorithm. It is commonly used to assess the diversity of the algorithm in the objective function space. A larger hypervolume indicates that the set of solutions generated by the algorithm is more widely distributed in the objective function space.

These metrics together provide a comprehensive assessment of the algorithm's performance, including solution quality, diversity, and closeness. To ensure fairness, we set the running time for all algorithms to 120 s, which allows ample time for each algorithm to thoroughly explore the solution space, and the population size to 50, with other parameters set according to the relevant literature. Additionally, we adopt the same decoding and encoding methods. It is worth noting that, to overcome the randomness of heuristic algorithms, we run each algorithm 15 times and report the average values, as shown in Table 8. Furthermore, Figure 10 displays the statistical results using box plots.

Table 8. Algorithm comparison results.

Algorithms	NPS	HV	IGD
MODBA	14.2	0.732	0.116
NSGA-II	8.9	0.658	0.153
NICA-II	10.8	0.656	0.149
MOPSO	12.0	0.675	0.118
GSA	12.5	0.662	0.176
QEA	13.3	0.683	0.132



(**a**) (**b**)

(**c**)

Figure 10. Statistical results of algorithm comparison NPS (**a**), HV (**b**), IGD (**c**).

Based on the data presented in Table 8, MODBA has emerged as a top performer, achieving the best scores for the NPS, HV, and IGD metrics. These results suggest that

MODBA is not only adept at identifying a substantial collection of Pareto optimal solutions but also ensures that these solutions are expansively spread across the objective function space. This widespread distribution is indicative of a robust exploration capability, which is crucial for multi-objective optimisation.

Furthermore, as depicted in Figure 10, MODBA demonstrates commendable stability in its performance with respect to the NPS and HV metrics, which is evidenced by the shortest lengths of its box plots. While the IGD metric shows the greatest stability for the GSA, it is important to note that the average IGD value for the GSA significantly lags behind that of MODBA.

In summary, MODBA has proven itself to be a formidable contender in the realm of disassembly-planning problems. Its strong performance across various metrics—solution quality, diversity, closeness to the true Pareto front, and stability—underscores its potential as an efficient and dependable optimisation tool. MODBA's ability to consistently produce high-quality solutions, coupled with its broad exploration of the solution space and stability of performance, makes it a promising choice for addressing complex optimisation challenges.

7. Conclusions and Future Work

With the growing constraints on resources and the increasing demands for environmental protection, the recycling of EOL products has become essential. Disassembly, as the first step in the EOL product-recycling process, is receiving attention. The advancement of robotics technology has made it imperative to efficiently address the human–robot collaborative DLB problem. However, research on the human–robot DLB problem is limited, and the existing studies on human–robot collaborative DLB focus on scenarios where there is only one human or one robot at a workstation. Additionally, research on energy-saving human–robot collaborative DLB problems is also scarce.

This study has constructed a multiobjective human–robot collaborative DLB optimisation model that takes into account disassembly-task allocation, task operation mode selection, and the integration of collaborative robots. The aim is to minimise the number of active workstations, the idleness rate of the disassembly line, and the energy consumption. We also develop a MODBA algorithm to efficiently solve the proposed problem.

A case study of fuel-cell disassembly demonstrates that the objectives we propose have complex and contradictory relationships. Our research provides decision makers with a broad decision-making space. Furthermore, through comparative analysis with other advanced algorithms, we confirm the effectiveness and superiority of the proposed algorithm for solving the DLB problem. The algorithm is innovative in theory and shows strong adaptability and practicality in its application.

Despite the achievements of this study, there are limitations and future research directions. Further research on the specific requirements of disassembly for different types of products, as well as how to apply the models and algorithms of this study to a broader range of industrial scenarios, will be an important direction for future research. Moreover, dedicating efforts to a more in-depth study of the disassembly-precedence matrix is a highly valuable direction. For instance, incorporating more comprehensive factors based on priority, such as disassembly urgency and difficulty, can lead to a more thorough evaluation and optimisation. Furthermore, introducing randomness or variability in task execution time to better simulate real-world uncertainties is another highly valuable research direction. This could include using methods like Monte Carlo simulations to model the variability of task execution times under different scenarios and assessing their impact on overall task completion time and efficiency [34]. Additionally, conducting a sensitivity analysis to investigate the impacts of various factors (such as operator skills, tool efficiency, and environmental conditions) on task execution time and the overall disassembly process would be beneficial. These approaches will help to gain a more comprehensive understanding and address the complexities and uncertainties encountered in real-world scenarios. Finally, exploring more comprehensive human–robot task-allocation methods is also a valuable

direction, such as employing the Nelder–Mead simplex optimisation approach [38] and incorporating technologies like augmented reality and virtual reality [39–41].

Author Contributions: Conceptualisation, W.L.; Data curation, Y.L.; Formal analysis, Y.S., G.T., and Z.L.; Methodology, G.T. and Z.L.; Resources, G.T.; Validation, W.L.; Writing—original draft, Y.S., W.L., H.S., Y.L., G.T. and H.Z.; Writing—review and editing, H.S. and H.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported in part by the National Natural Science Foundation of China (Grant No. 52075303).

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Lambert, A.J. Disassembly sequencing: A survey. *Int. J. Prod. Res.* **2003**, *41*, 3721–3759. [CrossRef]
2. Hertel, D.; Bräunig, G.; Thürer, M. Towards a green electromobility transition: A systematic review of the state of the art on electric vehicle battery systems disassembly. *J. Manuf. Syst.* **2024**, *74*, 387–396. [CrossRef]
3. Özceylan, E.; Kalayci, C.B.; Güngör, A.; Gupta, S.M. Disassembly line balancing problem: A review of the state of the art and future directions. *Int. J. Prod. Res.* **2019**, *57*, 4805–4827. [CrossRef]
4. Laili, Y.; Li, Y.; Fang, Y.; Pham, D.T.; Zhang, L. Model review and algorithm comparison on multi-objective disassembly line balancing. *J. Manuf. Syst.* **2020**, *56*, 484–500. [CrossRef]
5. Edis, E.B.; Edis, R.S.; Ilgin, M.A. Mixed integer programming approaches to partial disassembly line balancing and sequencing problem. *Comput. Oper. Res.* **2022**, *138*, 105559. [CrossRef]
6. Tuo, Y.; Zhang, Z.; Wu, T.; Zeng, Y.; Zhang, Y.; Liu, J. Multimanned disassembly line balancing optimization considering walking workers and task evaluation indicators. *J. Manuf. Syst.* **2024**, *72*, 263–286. [CrossRef]
7. Zeng, Y.; Zhang, Z.; Yin, T.; Zheng, H. Robotic disassembly line balancing and sequencing problem considering energy-saving and high-profit for waste household appliances. *J. Clean. Prod.* **2022**, *381*, 135209. [CrossRef]
8. Xu, W.; Cui, J.; Liu, B.; Liu, J.; Yao, B.; Zhou, Z. Human-robot collaborative disassembly line balancing considering the safe strategy in remanufacturing. *J. Clean. Prod.* **2021**, *324*, 129158. [CrossRef]
9. Wang, K.; Li, X.; Gao, L.; Garg, A. Partial disassembly line balancing for energy consumption and profit under uncertainty. *Robot. Comput.-Integr. Manuf.* **2019**, *59*, 235–251. [CrossRef]
10. Wolpert, D.H.; Macready, W.G. No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* **1997**, *1*, 67–82. [CrossRef]
11. Shan, F.; Wu, Z.; Li, J.; Li, Z.; Sun, A. Study on the balance problem of lithium battery disassembly line based on improved particle swarm algorithm. *Intern. Combust. Engines Accessories* **2011**, *23*, 4–6.
12. Wang, Y.J.; Wang, N.D.; Cheng, S.M.; Zhang, X.C.; Liu, H.Y.; Shi, J.L.; Ma, Q.Y.; Zhou, M.J. Optimization of disassembly line balancing using an improved multi-objective Genetic Algorithm. *Adv. Prod. Eng. Manag.* **2021**, *16*, 240–252. [CrossRef]
13. Ren, Y.; Yu, D.; Zhang, C.; Tian, G.; Meng, L.; Zhou, X. An improved gravitational search algorithm for profit-oriented partial disassembly line balancing problem. *Int. J. Prod. Res.* **2017**, *55*, 7302–7316. [CrossRef]
14. Singh, R.K.; Singh, A.R.; Yadav, R.K. A balanced-quantum inspired evolutionary algorithm for solving disassembly line balancing problem. *Appl. Soft Comput.* **2023**, *132*, 109840. [CrossRef]
15. Wang, K.; Li, X.; Gao, L.; Li, P.; Sutherland, J.W. A discrete artificial bee colony algorithm for multiobjective disassembly line balancing of end-of-life products. *IEEE Trans. Cybern.* **2021**, *52*, 7415–7426. [CrossRef] [PubMed]
16. Yang, Y.; Yuan, G.; Zhuang, Q.; Tian, G. Multi-objective low-carbon disassembly line balancing for agricultural machinery using MDFOA and fuzzy AHP. *J. Clean. Prod.* **2019**, *233*, 1465–1474. [CrossRef]
17. Zhang, L.; Zhao, X.; Ke, Q.; Dong, W.; Zhong, Y. Disassembly line balancing optimization method for high efficiency and low carbon emission. *Int. J. Precis. Eng. Manuf.-Green Technol.* **2021**, *8*, 233–247. [CrossRef]
18. Tian, G.; Zhang, C.; Zhang, X.; Feng, Y.; Yuan, G.; Peng, T.; Pham, D.T. Multi-objective evolutionary algorithm with machine learning and local search for an energy-efficient disassembly line balancing problem in remanufacturing. *J. Manuf. Sci. Eng.* **2023**, *145*, 051002. [CrossRef]
19. Wang, K.; Li, X.; Gao, L.; Li, P. Modeling and balancing for green disassembly line using associated parts precedence graph and multi-objective genetic simulated annealing. *Int. J. Precis. Eng. Manuf.-Green Technol.* **2021**, *8*, 1597–1613. [CrossRef]
20. Wu, T.; Zhang, Z.; Yin, T.; Zhang, Y. Multi-objective optimisation for cell-level disassembly of waste power battery modules in human-machine hybrid mode. *Waste Manag.* **2022**, *144*, 513–526. [CrossRef]
21. Xing, Z.; Han, Y.; Wang, J.; Qin, S.; Qi, L.; Tan, Y. Improved cuckoo optimization algorithm for human-machine collaborative disassembly line balancing problem. In Proceedings of the 5th International Conference on Computer Information Science and Application Technology (CISAT 2022), Chongqing, China, 29–31 July 2022; SPIE: Bellingham, WA, USA, 2022; pp. 1031–1039.

22. Liu, B.; Xu, W.; Liu, J.; Yao, B.; Zhou, Z.; Pham, D.T. Human-robot collaboration for disassembly line balancing problem in remanufacturing. In Proceedings of the ASME 2019 14th International Manufacturing Science and Engineering Conference, Erie, PA, USA, 10–14 June 2019; American Society of Mechanical Engineers: New York, NY, USA, 2019; p. V001T02A037.
23. Liu, Y.; Zhou, M.; Guo, X. An improved Q-learning algorithm for human-robot collaboration two-sided disassembly line balancing problems. In Proceedings of the 2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Prague, Czech Republic, 9–12 October 2022; IEEE: New York, NY, USA, 2022; pp. 568–573.
24. Rehal, A.; Sen, D. An efficient disassembly sequencing scheme using the shell structure. *Comput.-Aided Des.* **2023**, *154*, 103423. [CrossRef]
25. Bahubalendruni, M.V.A.R.; Varupala, V.P. Disassembly sequence planning for safe disposal of end-of-life waste electric and electronic equipment. *Natl. Acad. Sci. Lett.* **2021**, *44*, 243–247. [CrossRef]
26. Wang, F.; Yang, Y.; Ji, X.; Yang, Q.; Li, Y.; Liu, J. A novel disassembly sequence planning method based on spatial constraint matrices. *Int. J. Adv. Manuf. Technol.* **2023**, *124*, 3001–3010. [CrossRef]
27. Gulivindala, A.K.; Bahubalendruni, M.V.A.R.; Balan P, M.; Eswaran, M. Mechanical disassembly sequence planning for end-of-life products to maximize recyclability. *Sādhanā* **2023**, *48*, 109. [CrossRef]
28. Feng, Y.; Cui, K.; Hong, Z.; Li, Z.; Yan, W.; Tan, J. Disassembly Sequence Planning of Product Structure with an Improved QICA Considering Expert Consensus for Remanufacturing. *IEEE Trans. Ind. Inform.* **2022**, *19*, 7201–7213. [CrossRef]
29. Xie, J.; Li, X.; Gao, L. Disassembly sequence planning based on a modified grey wolf optimizer. *Int. J. Adv. Manuf. Technol.* **2021**, *116*, 3731–3750. [CrossRef]
30. Anil Kumar, G.; Bahubalendruni, M.V.A.R.; Prasad, V.; Sankaranarayanasamy, K. A multi-layered disassembly sequence planning method to support decision making in de-manufacturing. *Sādhanā* **2021**, *46*, 102. [CrossRef]
31. Hellmuth, J.F.; DiFilippo, N.M.; Jouaneh, M.K. Assessment of the automation potential of electric vehicle battery disassembly. *J. Manuf. Syst.* **2021**, *59*, 398–412. [CrossRef]
32. Pham, D.T.; Ghanbarzadeh, A.; Koç, E.; Otri, S.; Rahim, S.; Zaidi, M. The bees algorithm—A novel tool for complex optimisation problems. In *Intelligent Production Machines and Systems, Proceedings of the 2nd I*PROMS Virtual International Conference, Online, 3–14 July 2006*; Elsevier Science Ltd.: Amsterdam, The Netherlands, 2006; pp. 454–459.
33. Hartono, N.; Ramírez, F.J.; Pham, D.T. Optimisation of robotic disassembly plans using the Bees Algorithm. *Robot. Comput.-Integr. Manuf.* **2022**, *78*, 102411. [CrossRef]
34. Zhang, X.; Tian, G.; Fathollahi-Fard, A.M.; Pham, D.T.; Li, Z.; Pu, Y.; Zhang, T. A chance-constraint programming approach for a disassembly line balancing problem under uncertainty. *J. Manuf. Syst.* **2024**, *74*, 346–366. [CrossRef]
35. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [CrossRef]
36. SAE International J1772_201710: SAE Electric Vehicle and Plug in Hybrid Electric Vehicle Conductive Charge Coupler—SAE International. Available online: https://www.sae.org/standards/content/j1772_201710/ (accessed on 18 May 2024).
37. Liu, J.; Zhang, Z.; Gong, J.; Chen, F.; Yin, T.; Zhang, Y. A novel hybrid clonal selection algorithm for the corridor allocation problem with irregular material handling positions. *Comput. Ind. Eng.* **2022**, *168*, 108118. [CrossRef]
38. Inkulu, A.K.; Bahubalendruni, M.V.A.R. Optimal resource allocation for multiple shop floor tasks in collaborative assembly. *Comput. Ind. Eng.* **2023**, *185*, 109695. [CrossRef]
39. Eswaran, M.; Bahubalendruni, M.V.A.R. Challenges and opportunities on AR/VR technologies for manufacturing systems in the context of industry 4.0: A state of the art review. *J. Manuf. Syst.* **2022**, *65*, 260–278. [CrossRef]
40. Inkulu, A.K.; Bahubalendruni, M.V.A.R.; Dara, A.; SankaranarayanaSamy, K. Challenges and opportunities in human robot collaboration context of Industry 4.0-a state of the art review. *Ind. Robot. Int. J. Robot. Res. Appl.* **2021**, *49*, 226–239. [CrossRef]
41. Eswaran, M.; kumar Inkulu, A.; Tamilarasan, K.; Bahubalendruni, M.V.A.R.; Jaideep, R.; Faris, M.S.; Jacob, N. Optimal layout planning for human robot collaborative assembly systems and visualization through immersive technologies. *Expert Syst. Appl.* **2024**, *241*, 122465. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.