Hindawi Publishing Corporation
Discrete Dynamics in Nature and Society
Volume 2014, Article ID 756891, 9 pages
http://dx.doi.org/10.1155/2014/756891

Research Article

A Scatter Search Approach for Multiobjective Selective Disassembly Sequence Problem

Xiwang Guo[1,2] and Shixin Liu[1]

[1] State Key Laboratory of Synthetical Automation for Process Industries, College of Information Science and Engineering, Northeastern University, Shenyang 110819, China
[2] College of Computer and Communication Engineering, Liaoning Shihua University, Fushun 113001, China

Correspondence should be addressed to Xiwang Guo; x.w.guo@163.com

Received 27 July 2014; Revised 29 September 2014; Accepted 4 November 2014; Published 18 November 2014

Academic Editor: Tinggui Chen

Disassembly sequence has received much attention in recent years. This work proposes a multiobjective optimization of model for selective disassembly sequences and maximizing disassembly profit and minimizing disassembly time. An improved scatter search (ISS) is adapted to solve proposed multiobjective optimization model, which embodies diversification generation of initial solutions, crossover combination operator, the local search strategy to improve the quality of new solutions, and reference set update method. To analyze the effect on the performance of ISS, simulation experiments are conducted on different products. The validity of ISS is verified by comparing the optimization effects of ISS and nondominated sorting genetic algorithm (NSGA-II).

1. Introduction

In order to protect environment and promote reuse and remanufacturing of used products, it is essential that disassembly of used products should be performed. One of the important topics is disassembly modeling and planning. Disassembly modeling methods mainly include an undirected graph, directed graph, AND/OR graph and Petri nets. Moore et al. [1] proposed Petri net approach to describe the logical relationship of product assembly and presented the AND/OR graph search method to obtain the optimal disassembly sequence. Hui et al. [2] adopted disassembly feasible information diagram to represent the product disassembly sequence and operation information and proposed genetic algorithm to search the possible set of disassembly sequence. Min et al. [3] developed the disassembly feasibility analysis method in which all the disassembly sequences were transformed into AND/OR graph to derive an efficient disassembly sequence. Moore et al. [4] and Glover [5] proposed the disassembly precedence matrix method to determine the optimal feasible solution. Gungor and Gupta [6, 7] generated the optimal disassembling sequence based on the geometry precedence relationship of components. They also proposed the evaluation method pertaining to the efficiency disassembly. Srinivasan and Gadh [8] used wave propagation method to solve the objective components disassembly problem and disassembled the least number of components as evaluation index to generate the optimal disassembly sequence. Chung and Peng [9] considered the bulk disassembly of components, and reachability of tools, and proposed an integrated selective disassembly method, aimed at products disassembly to produce the best disassembly method. Kara et al. [10] proposed a contact map to establish the priority rules to generate disassembly sequences, containing the objective component of all of the feasible disassembly sequence finally by screening to obtain the best optimal disassembly sequence. Wang et al. [11] developed geometrical reasoning methods to get disassembly components and used ant colony algorithm starting from the initial disassembly of components until disassembly of the objective component. Yi et al. [12] established general machinery components basis on CAD model and proposed one kind selective disassembly planning algorithm of disassemble wave components to obtain the best optional disassembly sequence. Li et al. [13–15] used hybrid map to analyze objective disassembly sequence and used genetic algorithm to get disassembly sequence. Wang et al. [16] proposed an ant colony optimization algorithm for intelligent selective disassembly. The number of initial disassembly components as the number of ants, multiple ants individual cooperative to search objective components, until disassembly objective components was found [17] used direct method and constraint release method to research the generation of objective sequence, for each disassembly sequence by searching the objective component to delete repeat sequences, and finally get the disassembly sequence of objective components. Tian et al. proposed a chance constrained programming approach to determine optimal disassembly sequence [18–20]. A Petri net is a graphic modeling method, which is widely used in modeling and analyzing discrete event systems such as semiconductor manufacturing, recommender system, and automated manufacturing system [21–24]. Tang and Grochowski employed a machine learning approach based on a disassembly Petri net and a hybrid Bayesian network. It models the disassembly process and predicts the outcome of each disassembly action by examining the probabilistic relationships among the different aspects of the disassembly process [25, 26].

The above overviewed results show that the prior researches mainly focused on disassembly sequence optimization with a single disassembly target, that is, disassembly time or cost. However, in an actual disassembly process, a decision maker may want to maximize the disassembly profit, as well as minimize the disassembly time. To handle it, a multiobjective optimization issue needs to be addressed. To do so, this work proposes a selective disassembly sequence optimization problem to establish a multiobjective optimization model for product disassembly. Namely, the aim of this work is to find a new way to determine the optimal disassembly sequence by taking multiple objectives into full account. In addition, a scatter search (SS) is adapted to solve the proposed model based on its characteristics. Compared with existing researches, we have three contributions: (1) this work extends the current disassembly information models and makes it better to describe actual conditions; (2) to solve multiobjective disassembly sequence optimization problem, a scatter search algorithm is designed, which includes diversification generation of initial solutions, crossover combination operator, and reference set update method; solutions improved operator is proposed to improve the quality of new solutions; (3) via experiments on multigroup randomly generated instances and in comparison with nondominated sorting genetic algorithm (NSGA-II), this work validates the effectiveness and feasibility of the proposed model and algorithm in solving multiobjective disassembly optimization problem.

The rest of this paper is organized as follows. Section 2 describes a disassembly problem and establishes its mathematical model. Section 3 describes SS algorithm. Section 4 presents the solutions to several cases. Finally, Section 5 concludes our work.

2. Problem Assumptions and Notations

2.1. Assumptions. The selective disassembly problem is to find an optimal disassembly sequence for single or multiple target components. The optimal sequences concern the geometric constraints of the assembly and precedence reorientations of the assembly during the disassembly process.

This work makes the following assumptions.

(1) The end-of-life product is composed of different types of components.

(2) Disassembly direction is as follows. Each component is disassembled within 3D space and can be disassembled in special directions, which are indexed as $+x$, $+y$, $+z$, $-x$, $-y$, and $-z$.

(3) Disassembly method is as follows. According to the disassembly method for each component, the process consists of destructive disassembly (D), which focuses on materials, rather than component recovery, and nondestructive disassembly (N), which focuses on components rather than material recovery.

(4) The disassembly sequence must satisfy predefined precedence relationship.

(5) Disassembly recovery profit and cost is dependent on the disassembly methodology.

2.2. Notations

Problem Parameters

n is the number of components composed of the EOL product.

j is index of component, $j = 1, 2, \ldots, n$.

P_j is components set of component p_j.

m_{ij} is penalty value due to the disassembly method change between components p_i and p_j (0: if method change is not required and 1: if method change is required).

d_{ij} is penalty value due to direction change between components p_i and p_j (0: if direction change is not required, 1: if 90° change is required, and 2: if 180° change is required).

t_j^0 is needed time for destructive disassembly component p_j.

t_j^1 is needed time for nondestructive disassembly component p_j.

C_j^0 is cost value for nondestructive disassembly component p_j.

C_j^1 is cost value for destructive disassembly component p_j.

R_j^0 is revenue value when recovering component p_j in a nondestructive manner.

R_j^1 is revenue value when recovering component p_j in a destructive manner.

R_j^N is revenue value when recovering component p_j is not disassembled.

Decision Variables

$\Pi = \{\pi_1, \pi_2, \ldots, \pi_n\}$ is disassembly sequence set.

$Y = \{y_1, y_2, \ldots, y_n\}$ is binary decision vector for the disassembly category (disassembled or not). $y_j = 0$; component p_j is disassembled; $y_j = 1$; component p_j is not be disassembled.

$X = \{x_1, x_2, \ldots, x_n\}$ is binary decision vector for the disassembly method. $x_j = 0$ denotes a nondestructive method to disassemble component p_j; $x_j = 1$ denotes a destructive method to disassemble component p_j.

Γ is the set of vectors Y.

Ω is the set of vectors X.

$\prod$ is satisfying disassembly precedence relations between n components of all the feasible set of permutations, and $\prod = \{\pi_1, \pi_2, \ldots, \pi_{|\prod|}\}$

2.3. Mathematical Modelling for Multiobjective Selective Disassembly. This paper considers the selective disassembly model where multiple objectives are considered. The first objective is disassembly time from the view of recovery efficiency. The second objective is recovery profit from the view of economic income. Most of the literatures address primarily single objective disassembly problem and the models with multiobjectives are transformed into a single objective for simplicity. However, the single objective model may not capture the real world characteristics in which multiobjectives often arise from the conflicting problem [27]. Based upon the above concept, the objective function depends on the disassembly sequence, which may include the penalty from the change of disassembly directions and disassembly methods. Let $X = \{x_1, x_2, \ldots, x_n\}$ be a disassembly sequence and x_i is the component in the ith position of such a disassembly sequence. There is another decision variable Y. The total disassembly time and overall recovery profit are simultaneously determined. Accordingly, a multiobjective optimization model can be described as follows:

$$\min Z_1 = \sum_{i=1}^{n} \mathrm{DT}_j \cdot (1 - y_i) + \sum_{i=1}^{n-1} \left(m_{ij} + d_{ij}\right)(1 - y_i), \quad \pi \in \Pi, \quad y \in \Gamma; \quad (1)$$

$$\max Z_2 = \left(R_j^D - C_j^D\right)\left(1 - y_j\right) + \sum_{j=1}^{n} R_j^N y_j, \quad y \in \Gamma.$$

Detaching component j is defined in DT_j:

$$\mathrm{DT}_j = t_j^0 \left(1 - x_j\right) + t_j^1 x_j. \quad (2)$$

Total disassembly revenue can be calculated by the following formula:

$$R_j^D = \sum_{j=1}^{n} R_j^0 \left(1 - x_j\right) + \sum_{j=1}^{n} R_j^1 x_j, \quad x \in \Omega. \quad (3)$$

Total disassembly cost is acquired by the following expression:

$$C_j^D = \sum_{j=1}^{n} C_j^0 \left(1 - x_j\right) + \sum_{j=1}^{n} C_j^1 x_j, \quad x \in \Omega. \quad (4)$$

Since this paper involves minimizing disassembly time of an EOL product and maximizing disassembly profit for two-objective optimization, which requires a computerized optimization procedure, the original SS is designed to address single objective problem and produce excellent results in shorter running time. Hence, an improved multiobjective SS is developed as an efficient solution methodology for selective disassembly sequence problem.

3. Scatter Search and Improved Design of Multiobjective Scatter Search

Scatter search (SS) as a heuristic algorithm was first proposed by Glover and has been successfully applied to solve hard optimization problem [5]. SS uses the diversification and intensification strategies to provide efficient solution for a variety of optimization problems. The performance of SS is rarely dependent on the randomness process during the search but is rather dependent on the systematic integrated method to solve optimization problem [2, 28–30]. In this paper, an improved SS is adopted to solve the multiobjective selective disassembly sequence problem.

3.1. The Framework of SS Algorithm. Scatter search consists of the following procedures: diversification generation, solution improvement, reference set update method, subset generation, and solution combination method [31]. Figure 1 shows the structure of scatter search algorithm.

The diversification generation method focuses on the systematic generation of diverse solution over randomization. The improvement method tries to update a solution into one or more enhanced solutions. The reference set update method is to build a reference set consisting of some of the best solutions. Subset generation method is designed to operate on the reference set to produce a subset. Solution combination method is to transform the subset of solutions into more combined solutions.

3.2. Solution Encoding Scheme. According to the problem in this research, solution encoding scheme is described as follows. Each individual solution appears the tuple-link structure $I = \{\pi, Y, X\}$, where $\pi = \{p_1, p_2, \ldots, p_n\}$ represents the disassembly sequence, $Y = \{y_1, y_2, \ldots, y_n\}$ represents the corresponding component whether component is performed, and $X = \{x_1, x_2, \ldots, x_n\}$ represents the selection of disassembly method. Basically, three possible direction changes may occur for manual disassembly under the assumption of stabilized product. The first is no direction change (i.e., $+x$ to $+x$, $+y$ to $+y$, $+z$ to $+z$, $-x$ to $-x$, $-y$ to $-y$, and $-z$ to $-z$), which has no penalty. The second direction change is a 90° change (e.g., $+x$ to $+y$, $-x$ to $+z$, and $-z$ to $+x$), which has a penalty of 1. The third direction change is a 180° change (i.e.,

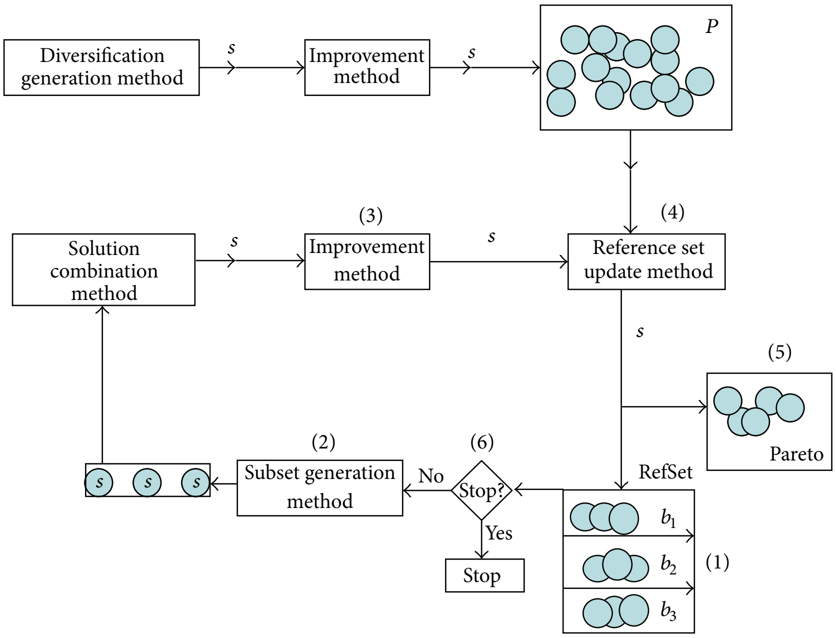


Figure 1: Schematic plot of Scatter search.

Table 1: Problem parameters.

Index	1	2	3	4	5	6	7	8	9	10
P	$\{2,3\}$			$\{9\}$	$\{7\}$	$\{7\}$	$\{9\}$	$\{2,3\}$	$\{2,3\}$	$\{3\}$
d_{ij}	$+y$	$+x$	$+x$	$+z$	$-z$	$-z$	$+y$	$-x$	$-z$	$-y$
t_j^0	1	3	2	2	3	2	1	1	3	2
t_j^1	3	4	3	3	4	3	3	2	4	3
R_j^0	5	7	5	9	6	8	7	5	5	8
R_j^1	4	6	3	5	4	5	6	3	2	6
R_j^N	3	5	4	7	5	6	5	4	4	5
C_j^0		5	3	3	5	7	4	6	4	9

Table 2: Coding and the objective function of individual.

π	3	9	8	2	1	7	4	10	6	5
Y	0	0	1	0	0	0	0	1	0	0
X	0	0	1	0	1	1	1	1	0	0
Z_1					29					
Z_2					16					

$+x$ to $-x$, $+y$ to $-y$, $+z$ to $-z$, $-x$ to $+x$, $-y$ to $+y$, and $-z$ to $+z$), which has a penalty of 2 (see Table 1).

The left side frame door of a simplified aircraft is taken as a case study example to analyze the selective disassembly sequence planning. It contains 10 components and its index is denoted from 1 to 10. Its precedence relationship, disassembly direction, different disassembly method, disassembly time, disassembly cost, and disassembly profit are as shown in Table 1. This paper embodies 6 disassembly directions of component; that is, its left and right movement direction is $-x$ and $+x$, before and after movement direction is $+y$ and $-y$, and up and down movement direction is $+z$ and $-z$.

An example is introduced to illustrate the solution encoding scheme as shown in Table 1. The individual I is encoded as shown in Table 2. The constraints condition must be satisfied as follows: component 9 must be disassembled prior to component 8 and after component 3; component 9's disassembly direction is "$-z$," disassembly time ($t_9 = 3$) is 3 units of time, disassembly direction change time ($d_{39} = 1$) is 1 unit of time, disassembly method change time ($m_{39} = 0$) is 0 unit of time, disassembly profit is 5 yuans ($R_9^0 = 5$), disassembly cost is 4 yuans ($C_9^0 = 4$), according to the objective function formula calculated disassembly time is 29 seconds ($Z_1 = 29$), and disassembly profit is 16 yuans ($Z_2 = 16$).

3.3. Generation of Initial Solution Set P. As the number of components increases the numbers of solutions does so drastically. Some generated solutions/individuals among them may not meet a precedence relationship constraint. They should be regenerated. The initial solution set P is randomly generated through the following procedure. (1) The randomly generated disassembly sequence $\pi = \{p_1, p_2, \ldots, p_n\}$ constructs the first component of individual solution I. In addition, the precedence relationship must be satisfied with the constraints. (2) According to the element in the position of π, the corresponding element is randomly placed into X and Y where $y_i \in \{0,1\}$, $x_i \in \{0,1\}$. (3) Repeat (1)-(2) processes until the initial solution set P is full.

3.4. Solutions to Improve Operator. In SS algorithm, solution improvement method is used to produce new high quality solution by a local search method. The solution improvement process in this paper is introduced as follows: for a given individual $I = (\pi, X, Y)$, exchange the two-position values in the $\pi = \{p_1, p_2, \ldots, p_n\}$ while ensuring that the precedence constraints are kept. Thus, bit values in vectors $Y = \{y_1, y_2, \ldots, y_n\}$ and $X = \{x_1, x_2, \ldots, x_n\}$ are changed by the rule "0" turning to "1" and "1" turning to "0." A domination test procedure is performed after the solutions improvement process which checks whether the improved solutions are dominated or not. If the original solution dominates the newly generated solution, the new one will be abolished. Otherwise, the original solution will be replaced with the new generated solution. If both solutions are nondominated, the new one will be migrated to the external archive and the original one retains to avoid the worst solutions poured into the archive. The proposed improved method has the following advantages. First of all, in the solution improvement process, no nondominated solution is lost during the search process and they were inserted into the external archive. Second, this process enhances the local search ability of scatter search algorithm and solutions diversity is maintained to prevent the premature phenomenon.

3.5. Update Pareto Solution Set. In order to ensure the solving process to produce a high quality Pareto solution set, external archive is adopted [32]. The main purpose of external archive is to store each nondominated individuals during the search process. Whether the new solution is inserted into the external archive is determined by the achieve management process. Precisely speaking, in the solution improvement process or solution combination process, when a new solution is found, the dominant check test is triggered. If this new solution is dominated by the archive solution, this solution will be removed; otherwise, the solution is stored in the archive. If the solution in archive was dominated by new solution, the solution in archive will be replaced, the dominance measurement will stop until reach the maximum count number which is predefined parameter.

3.6. Generation of Reference Set. The proposed multiobjective SS is based on the SS. In order to improve searching performance, some modifications have been incorporated and new reference set updating mechanism has been developed. The reference set plays the significant role in the SS procedure. If all the solutions in the reference set are identical to each other during the search, the SS is subject to becoming struck in the local optimum, even though employing a sophisticated procedure will not probably improve the best solutions found. The reference set is built to maintain a collection of high quality and diversity solutions. In our proposed improved multiobjective SS, the size of reference set is denoted by $b_1 + b_2 + b_3 = |\text{refset}|$, where b_1 best solutions are selected according to the function value of objective 1 and b_2 best solutions are selected according to the function value of objective 2, respectively. The b_3 solutions are added into the reference set due to Euclidean distances D. The distance between two individuals to generate b_3 diversified solutions is as follows:

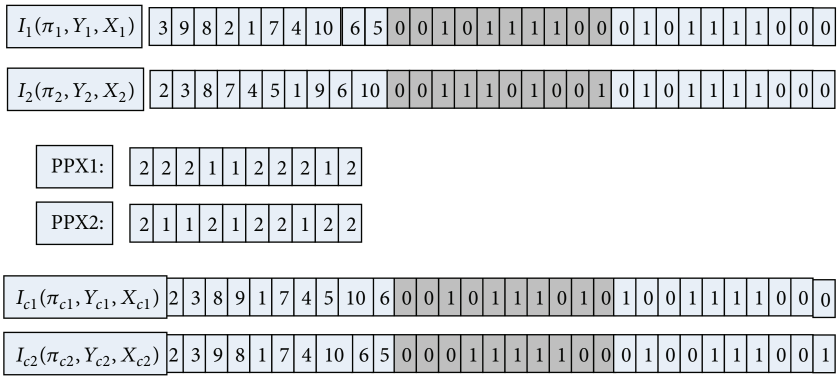
$$D(I_1, I_2) = \sum_{j=1}^{n} \left|\operatorname{pos}\left(p_j^{\pi_1}\right) - \operatorname{pos}\left(p_j^{\pi_2}\right)\right| + \sum_{j=1}^{n} \left|p_j^{y_1} - p_j^{y_2}\right| + \sum_{j=1}^{n} \left|p_j^{x_1} - p_j^{x_2}\right|, \tag{5}$$

where $\operatorname{pos}(p_j^{\pi_i})$ represents the position of component p_j in the disassembly sequence π_i, $p_j^{y_1}$ represents the bit value of component p_j in vector Y_1, and $p_j^{x_1}$ represents bit value of component p_j in vector X_1.

For each diversity solution in reference set, the minimum value of distance to the other solutions in the reference set is calculated. We compute the minimum distance between solutions I_i and I_j in reference set as the following expressions $D(I_i, \text{Ref}) = \min\{D(I_i, I_j) \mid \forall I_j \in \text{Ref}\}$. The b_3 solutions are selected from P with the criterion of maximizing the minimum similarity distance.

3.7. Generating Subsets. Subset generation method in SS generates the subsets of fixed or variable sizes and therefore operates on the reference set. The number of subset greatly affects the execution time of scatter search. In order to simplify the generation of new solutions and improve the efficiency of the algorithm, we adopt all 2-element subsets which consist of all pair-wise combinations of the solution in reference set.

3.8. Solution Combination Operator. Since the precedence relationship constraints exist, we employ the precedence preserved crossover PPX (precedence preservative crossover) as the Combination operator. The PPX crossover operator [33] compared with other crossover combination operators, the advantage of PPX is that precedence relationship constraints of each operation of parents are completely enforced from one generation to next and ensure that each operation sequence of offspring meets all the precedence relationship constraints. PPX operator is employed on the pair-wise sequence of the first segment of parental solutions. The newly generated sequence maintains the precedence relationship and determines the parent gene cross-order. In the process of PPX, first of all, two vectors PPX1 and PPX2 with the same length of component number n are randomly generated which represents the order in which the components are successively drawn from the Parent solution 1 and Parent solution 2. These operations pass on the parental permutations to generate two new offspring while keeping the original precedence relationship. This procedure is repeated until the first segment of two new offspring I_{c1} and I_{c2} is created. For example, consider two solutions I_1 and I_2. Since the offspring rely on the first segment of the parental solutions, the left bits can be obtained with respect to corresponding position value of the disassembling sequence π. Next, the mask arrays PPX1 and PPX2 are randomly created which consist of the numberd 1 and 2 representing the parent number from which the gene



$I_1(\pi_1, Y_1, X_1)$ 3 9 8 2 1 7 4 10 6 5 0 0 1 0 1 1 1 1 0 0 0 1 0 1 1 1 1 0 0 0

$I_2(\pi_2, Y_2, X_2)$ 2 3 8 7 4 5 1 9 6 10 0 0 1 1 1 0 1 0 0 1 0 1 0 1 1 1 1 0 0 0

PPX1: 2 2 2 1 1 2 2 2 1 2

PPX2: 2 1 1 2 1 2 2 1 2 2

$I_{c1}(\pi_{c1}, Y_{c1}, X_{c1})$ 2 3 8 9 1 7 4 5 10 6 0 0 1 0 1 1 1 0 1 0 1 0 0 1 1 1 1 0 0 0

$I_{c2}(\pi_{c2}, Y_{c2}, X_{c2})$ 2 3 9 8 1 7 4 10 6 5 0 0 0 1 1 1 1 1 0 0 0 1 0 0 1 1 1 0 0 1

Figure 2: The precedence relation preserved crossover chart.

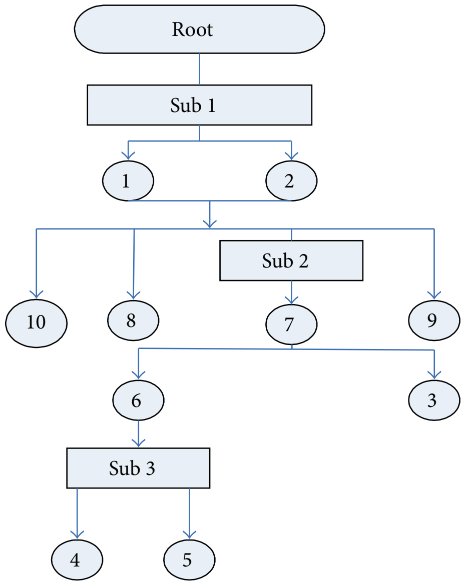


Figure 3: The product structure diagram to be disassembled.

is selected. Then, the first segment of offspring I_{c1} and I_{c2} is created. Figure 2 shows the whole process of combination operator.

3.9. Update Reference Set. Reference set update method uses improved new solutions scheme as updated reference set, so that the reference set ensures maintaining high quality solutions and the diversity solutions. Through improved solution method, the generated solution is superior to the reference set solution regarding the quality or diversity aspect. The generated solutions will be comprised of the reference set, while the solutions in the reference set with worse quality or lack of diversity will be excluded. Presently, there are two kinds of reference set update methods, called the dynamic reference set update method and static reference set update method, respectively. We employ the dynamic reference set update method in this paper.

4. Computational Experiments

To test the effectiveness of the proposed method in this work, a case from Literature [27] is considered to validate in the experiments. The case contains 10 parts and its product structure shown in Figure 3 notes that problem parameters are modified according to the problem definition in this paper. Parameters of running this case are set as follows: needed time for destructive disassembly component p_j is $t_j^0 \in [1, 15]$; needed time for nondestructive disassembly component p_j is $t_j^1 \in [1, 15]$; cost value for nondestructive disassembly component p_j is $C_j^0 \in [1, 10]$; cost value for destructive disassembly component p_j is $C_j^1 \in [1, 10]$; revenue value when recovering component p_j in a nondestructive manner is $R_j^0 \in [1, 10]$; revenue value when recovering component p_j in a destructive manner is $R_j^1 \in [1, 10]$; revenue value when recovering component p_j is not disassembled $R_j^N \in [1, 10]$. We use the Java language implements improved scatter search algorithm, which runs on Windows XP operating system Pentium IV (2.66 GHz/2.0 G RAM) PC. The results of running the experiment once for the product with 10 components (as shown in Figure 3) are shown in Table 3.

To further test the model and algorithm optimization results and time efficiency, we use 7 products with 30, 50, 80, 100, 150, 200, and 300 components, respectively, and relevant parameters are randomly generated as shown in Table 4. To execute ISS and NSGA-II algorithm [34], NSGA-II algorithm includes traditional selection, crossover, mutation, fast nondominated sort strategies, and crowding distance assignment. Note that pc represents crossover probability, pm represents mutation probability, *pop_size* represents the size of population, and *Gen* represents maximum iteration count in SS.

ISS algorithm parameters are set as follows: *pop_size* $=$ 100, $b_1 = b_2 = 5$, $b_3 = 10$, and *Gen* $= 100$. NSGA-II parameters are set as follows: *pop_size* $= 100$, $p_c = 0.6$, $p_m = 0.005$, and *Gen* $= 100$. In order to fairly compare two algorithms, NSGA-II algorithm uses the same encoding and decoding rules, select operator using the roulette wheel method, crossover operator using PPX. Two algorithms to solve the same disassembly sequence problem,

Table 3: Computational results.

Number		Disassembly sequence	Disassembly time	Disassembly profit
	π	1, 2, 8, 7, 6, 4, 5, 3, 10, 9		
1	Y	1, 1, 1, 1, 1, 1, 1, 1, 0, 1	15	83
	X	1, 1, 1, 1, 0, 1, 0, 1, 0, 1,		
	π	1, 2, 7, 6, 8, 5, 4, 3, 9, 10		
2	Y	1, 1, 1, 1, 0, 1, 1, 1, 1, 1	19	95
	X	1, 1, 0, 1, 1, 0, 1, 1, 0, 1,		
	π	1, 2, 9, 7, 8, 10, 6, 4, 5, 3		
3	Y	1, 1, 0, 1, 1, 1, 1, 1, 1, 1,	13	80
	X	1, 0, 1, 0, 1, 1, 1, 0, 1, 1,		

Table 4: Different parameters and value range.

Parameters	t_j^0	t_j^1	R_j^0	R_j^1	R_j^N	C_j^0	C_j^1
Range	[1, 10]	[1, 10]	[1, 15]	[1, 15]	[1, 20]	[1, 10]	[1, 10]

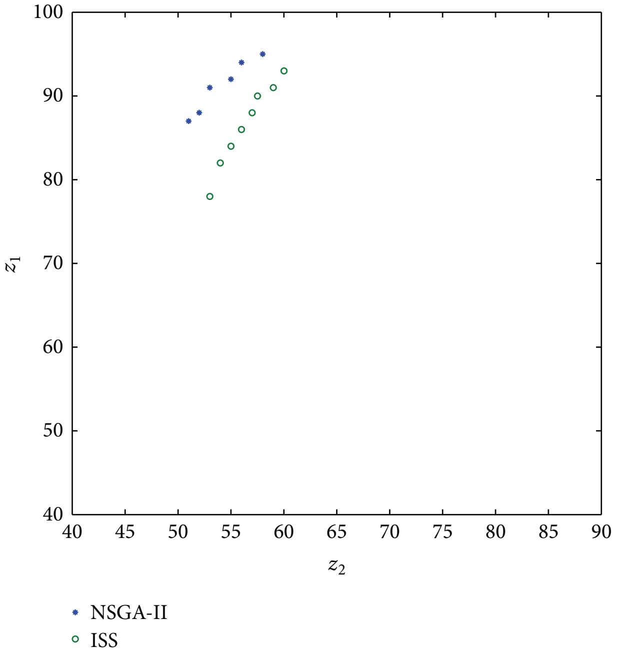


Figure 4: Final evolvable population generated by ISS and NSGAII with 30 components.

author collected the ISS algorithm terminates in the external archive approximate Pareto solutions. Two algorithms to solve disassembly sequence problem, author collected the ISS algorithm terminates when the external archive approximate Pareto solutions, and NSGA-II algorithm terminates in the last population approximate Pareto solutions; then according to collected Pareto solution ISS algorithm and NSGA-II algorithm to generate approximate Pareto solution set, the results are shown in Figures 4 and 5.

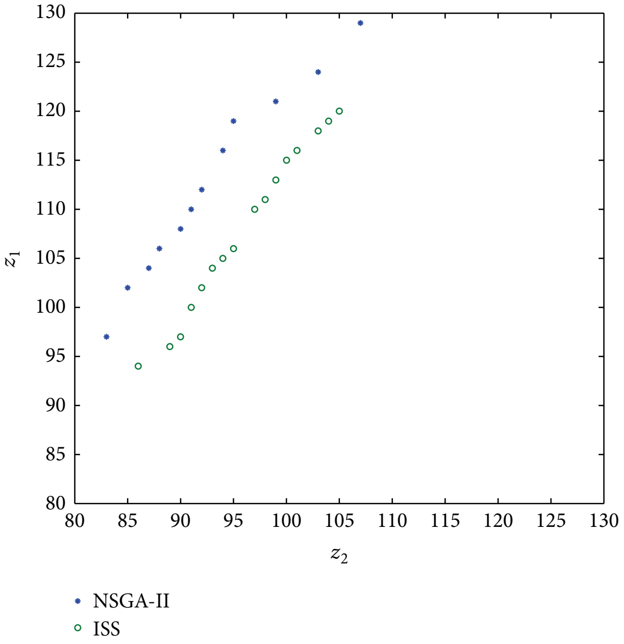


Figure 5: Final evolvable population generated by ISS and NSGA-II with 50 components.

Figures 4 and 5 are the results of solving 30 components and 50 components which use the ISS algorithm and NSGA-II algorithm. The results show that each objective function value, i.e., disassembly revenue or time increases as the number of disassembly components increases. The ISS algorithm generates approximate Pareto solution set, which is superior to compare NSGA-II algorithm for multiobjective selective disassembly sequence problems. Figure 6 represents comparison graph of two algorithms on computing time, which shows that SS algorithm CPU time is shorter than the NSGA-II algorithm CPU time. By analyzing Figures 4, 5, and 6, the following results can be found:

(1) with the increasing scale of the problem, the corresponding target value will increase in the same parameters;

(2) algorithm for different scale of the problem will solve different effective approximate solution set; the solution results by SS algorithm are better than solution results by NSGA-II algorithm;

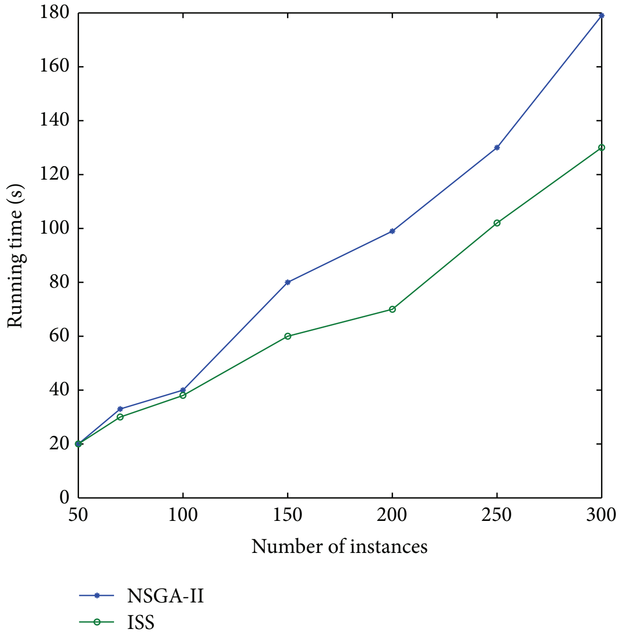


FIGURE 6: Comparison graph of two algorithms on computing time.

(3) ISS algorithm CPU time is shorter than the NSGA-II algorithm CPU time;

(4) the algorithm can obtain more numbers of Pareto solutions; in a word, based on the results, the proposed algorithm is feasible and efficient to solve a multiobjective disassembly sequence optimization model.

5. Conclusions

Disassembly sequence planning is an essential procedure of EOL products recovery which directly affects the effectiveness of maintenance and remanufacturing process in reverse logistics [35, 36]. This work addresses an intelligent selective disassembly approach based on scatter search algorithms for the first time. A multiobjective selective model has been developed to maximize disassembly revenue and minimize disassembly time. We then choose to employ improved multiobjective scatter search method to obtain the solution set of the proposed model. Using this solution methodology, a multiobjective selective disassembly problem can be easily solved. A case study on disassembling a domestic aircraft is adopted to illustrate the efficiency of the proposed model. The results can be used to guide decision makers in making better decisions when a product is disassemblied.

Although the efficacy of the proposed model has been verified, certain limitations exist. First, this work does not use actual disassembly data to validate this method to provide the best decision support for disassembly sequence practice. Another limitation is that the effective measures to describe the quality of the Pareto set may not be applied. Therefore, more effective measures and methods should be studied in the future to identify the best ways to solve the disassembly problems.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors would like to thank the editor and reviewers for the insightful comments that helped us improve this paper. In addition, they would like to thank Professor Guangdong Tian for his valuable comments to improve the presentation of this paper. This research is financially supported by National Natural Science Foundation of China (71171038 and 71021061) and the Fundamental Research Funds for the Central Universities (N100504001).

References

[1] K. E. Moore, A. Güngör, and S. M. Gupta, “Petri net approach to disassembly process planning for products with complex AND/OR precedence relationships,” *European Journal of Operational Research*, vol. 135, no. 2, pp. 428–449, 2001.

[2] W. Hui, X. Dong, and D. Guanghong, “A genetic algorithm for product disassembly sequence planning,” *Neurocomputing*, vol. 71, no. 13–15, pp. 2720–2726, 2008.

[3] S. S. Min, X. J. Zhu, and X. Zhu, “Mechanical product disassembly and/or graph construction,” in *Proceedings of the International Conference on Measuring Technology and Mechatronics Automation*, vol. 5, pp. 627–631, March 2010.

[4] K. E. Moore, A. Güngör, and S. M. Gupta, “Petri net approach to disassembly process planning for products with complex AND/OR precedence relationships,” *European Journal of Operational Research*, vol. 135, no. 2, pp. 428–449, 2001.

[5] F. Glover, “Scatter search and path Re-linking,” *New Methods in Optimization*, vol. 19, no. 1, pp. 297–316, 1998.

[6] A. Gungor and S. M. Gupta, “Disassembly sequence planning for products with defective parts in product recovery,” *Computers & Industrial Engineering*, vol. 35, no. 1-2, pp. 161–164, 1998.

[7] A. Gungor and S. M. Gupta, “An evaluation methodology for disassembly processes,” *Computers & Industrial Engineering*, vol. 33, no. 1-2, pp. 329–332, 1997.

[8] H. Srinivasan and R. Gadh, “A geometric algorithm for single selective disassembly using the wave propagation abstraction,” *Computer Aided Design*, vol. 30, no. 8, pp. 603–613, 1998.

[9] C. Chung and Q. Peng, “An integrated approach to selective-disassembly sequence planning,” *Robotics and Computer-Integrated Manufacturing*, vol. 21, no. 4-5, pp. 475–485, 2005.

[10] S. Kara, P. Pornprasitpol, and H. Kaebernick, “A selective disassembly methodology for end-of-life products,” *Assembly Automation*, vol. 25, no. 2, pp. 124–134, 2005.

[11] J. F. Wang, J. H. Liu, S. Q. Li, and Y. F. Zhong, “Intelligent selective disassembly using the ant colony algorithm,” *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, vol. 17, no. 4, pp. 325–333, 2003.

[12] J. J. Yi, L. Y. Cheng, G. Li, and D. Q. Hu, “Research on the selectable disassembly strategy of mechanical parts based on the

generalized CAD mode," *The International Journal of Advanced Manufacturing Technology*, vol. 37, no. 3, pp. 599–604, 2008.

[13] J. R. Li, S. B. Tor, and L. P. Khoo, "A hybrid disassembly sequence planning approach for maintenance," *Journal of Computing and Information Science in Engineering*, vol. 2, no. 1, pp. 28–37, 2002.

[14] J. R. Li, S. B. Tor, and L. P. Khoo, "A novel representation scheme for disassembly sequence planning," *International Journal of Advanced Manufacturing Technology*, vol. 20, no. 8, pp. 621–630, 2002.

[15] J. R. Li, L. P. Khoo, and S. B. Tor, "An object-oriented intelligent disassembly sequence planner for maintenance," *Computers in Industry*, vol. 56, no. 7, pp. 699–718, 2005.

[16] J. F. Wang, J. H. Liu, S. Q. Li, and Y. F. Zhong, "Intelligent selective disassembly using the ant colony algorithm," *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, vol. 17, no. 4, pp. 325–333, 2003.

[17] X. Y. Pan, G. F. Liu, and Z. F. Liu, "The target of disassembly sequence generation algorithm," *Machine Tool & Hydraulics*, vol. 4, no. 3, pp. 42–43, 2003.

[18] G. Tian, M. Zhou, J. Chu, and Y. Liu, "Probability evaluation models of product disassembly cost subject to random removal time and different removal labor cost," *IEEE Transactions on Automation Science and Engineering*, vol. 9, no. 2, pp. 288–295, 2012.

[19] G. Tian, Y. Liu, Q. Tian, and J. Chu, "Evaluation model and algorithm of product disassembly process with stochastic feature," *Clean Technologies and Environmental Policy*, vol. 14, no. 2, pp. 345–356, 2012.

[20] G. Tian, M. Zhou, and J. Chu, "A chance constrained programming approach to determine the optimal disassembly sequence," *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 4, pp. 1004–1013, 2013.

[21] Z. W. Li and M. C. Zhou, "Elementary siphons of Petri nets and their application to deadlock prevention in flexible manufacturing systems," *IEEE Transactions on Systems, Man, and Cybernetics Part A*, vol. 34, no. 1, pp. 38–51, 2004.

[22] Z. Li and M. Zhou, "Clarifications on the definitions of elementary siphons in Petri nets," *IEEE Transactions on Systems, Man, and Cybernetics A: Systems and Humans*, vol. 36, no. 6, pp. 1227–1229, 2006.

[23] N. Wu, M. Zhou, and Z. Li, "Resource-oriented Petri net for deadlock avoidance in flexible assembly systems," *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans*, vol. 38, no. 1, pp. 56–69, 2008.

[24] Z. Li and M. Zhao, "On controllability of dependent siphons for deadlock prevention in generalized Petri nets," *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans*, vol. 38, no. 2, pp. 369–384, 2008.

[25] Y. Tang, "Learning-based disassembly process planner for uncertainty management," *IEEE Transactions on Systems, Man and Cybernetics A: Systems and Humans*, vol. 39, no. 1, pp. 134–143, 2009.

[26] D. E. Grochowski and Y. Tang, "A machine learning approach for optimal disassembly planning," *International Journal of Computer Integrated Manufacturing*, vol. 22, no. 4, pp. 374–383, 2009.

[27] E. Kongar and S. M. Gupta, "Disassembly sequencing using genetic algorithm," *International Journal of Advanced Manufacturing Technology*, vol. 30, no. 5-6, pp. 497–506, 2006.

[28] K.-K. Seo, J.-H. Park, and D.-S. Jang, "Optimal disassembly sequence using genetic algorithms considering economic and environmental aspects," *International Journal of Advanced Manufacturing Technology*, vol. 18, no. 5, pp. 371–380, 2001.

[29] B. González and B. Adenso-Díaz, "A scatter search approach to the optimum disassembly sequence problem," *Computers and Operations Research*, vol. 33, no. 6, pp. 1776–1793, 2006.

[30] R. Martí, M. Laguna, and F. Glover, "Principles of scatter search," *European Journal of Operational Research*, vol. 169, no. 2, pp. 359–372, 2006.

[31] M. Laguna and R. Martí, *Scatter Search Methodology and Implementations in C*, Kluwer Academic, Dordrecht, The Netherlands, 2003.

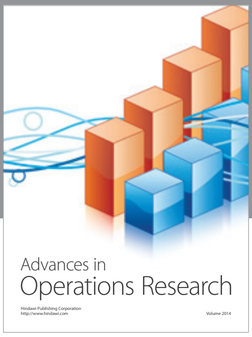
[32] A. J. Nebro, F. Luna, E. Alba, B. Dorronsoro, J. J. Durillo, and A. Beham, "AbYSS: adapting scatter search to multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 4, pp. 439–457, 2008.

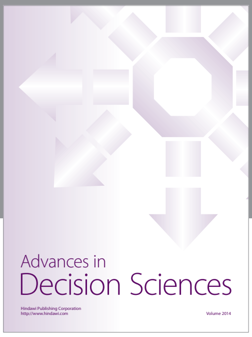
[33] C. Bierwirth and D. C. Mattfeld, "Production scheduling and rescheduling with genetic algorithms," *Evolutionary computation*, vol. 7, no. 1, pp. 1–18, 1999.

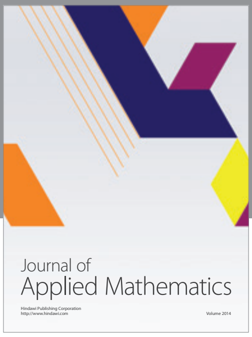
[34] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

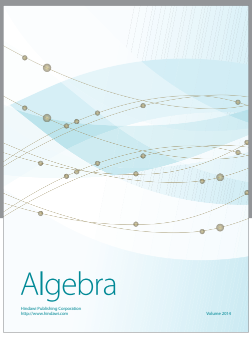
[35] G. Tian, J. Chu, and T. Qiang, "Influence factor analysis and prediction models for component removal time in manufacturing," *Proceedings of the Institution of Mechanical Engineers Part B: Journal of Engineering Manufacture*, vol. 227, no. 10, pp. 1533–1540, 2013.

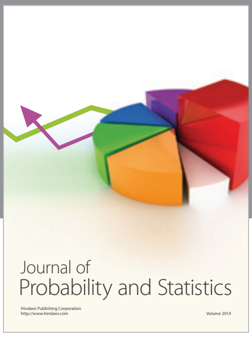
[36] G. D. Tian, J. W. Chu, H. S. Hu, and H. Li, "Technology innovation system and its integrated structure for automotive components remanufacturing industry development in China," *Journal of Cleaner Production*, 2014.


Advances in
Operations Research

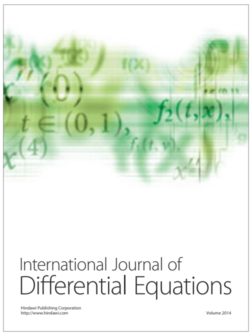

Advances in
Decision Sciences


Journal of
Applied Mathematics

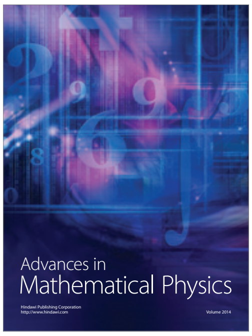

Algebra

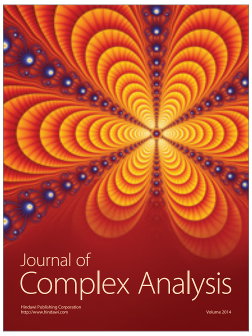

Journal of
Probability and Statistics

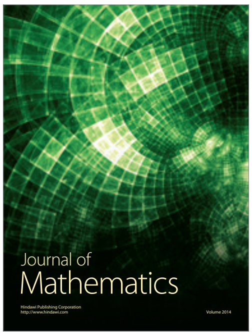
The Scientific
World Journal

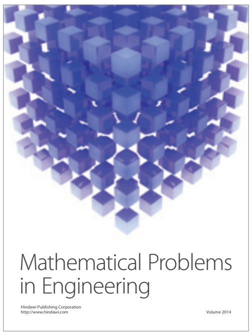

International Journal of
Differential Equations

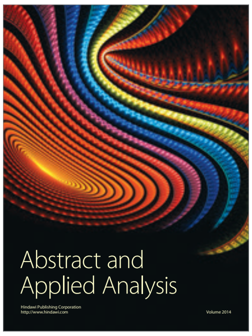
International Journal of
Combinatorics

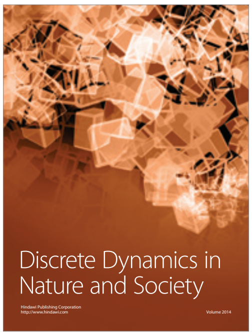

Advances in
Mathematical Physics

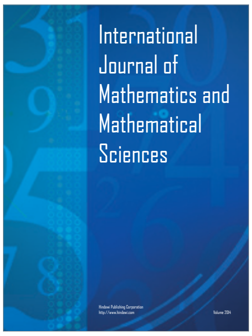

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

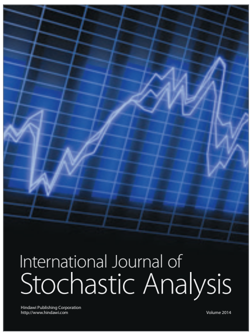

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization