Front. Mech. Eng. 2019, 14(4): 474–488
https://doi.org/10.1007/s11465-019-0560-z

RESEARCH ARTICLE

Jiali ZHAO, Shitong PENG, Tao LI, Shengping LV, Mengyun LI, Hongchao ZHANG

Energy-aware fuzzy job-shop scheduling for engine remanufacturing at the multi-machine level

Abstract The rise of the engine remanufacturing industry has resulted in increased possibilities of energy conservation during the remanufacturing process, and scheduling could exert significant effects on the energy performance of manufacturing systems. However, only a few studies have specifically addressed energy-efficient scheduling for remanufacturing. Considering the uncertain processing time and routes and the operation characteristics of remanufacturing, we used the crankshaft as an illustrative case and built a fuzzy job-shop scheduling model to minimize the energy consumption during remanufacturing. An improved adaptive genetic algorithm was developed by using the hormone modulation mechanism to deal with the scheduling problem that simultaneously involves parallel machines, batch machines, and uncertain processing routes and time. The algorithm demonstrated superior performance in terms of optimal value, run time, and convergent generation in comparison with other algorithms. Computational results indicated that the optimal scheduling scheme is expected to generate 1.7 kW·h of energy saving for the investigated problem size. In addition, the scheme could improve the energy efficiency of the crankshaft remanufacturing process by approximately 5%. This study provides a basis for production managers to improve the sustainability of remanufacturing through energy-aware scheduling.

Keywords remanufacturing scheduling, adaptive genetic algorithm, energy efficiency, sustainable remanufacturing, hormone modulation mechanism

Received April 21, 2019; accepted July 12, 2019

Jiali ZHAO
School of Mechanical & Electronical Engineering, Lanzhou University of Technology, Lanzhou 730050, China

Shitong PENG (✉), Tao LI, Mengyun LI, Hongchao ZHANG
Institute of Sustainable Design and Manufacturing, Dalian University of Technology, Dalian 116024, China
E-mail: speng@mail.dlut.edu.cn

Shengping LV
College of Engineering, South China Agricultural University, Guangzhou 510642, China

Hongchao ZHANG
Department of Industrial, Manufacturing & Systems Engineering, Texas Tech University, Lubbock, TX 79409, USA

1 Introduction

The dramatic increase in the demand and consumption of new products places substantial environmental and economic burdens on the original equipment manufacturer. Remanufacturing has been widely adopted as an energy-saving and environmentally benign manufacturing paradigm to recover the residual value of end-of-life products completely [1]. According to statistical data from China Automotive Technology and Research Center, the vehicle population and number of scrapped vehicles in China are projected to reach 1.4 billion and 99.5 million in 2020, respectively [2]. The increasing vehicle consumption in China would inevitably lead to increased amounts of scrapped vehicles, which enhances the need for engine remanufacturing. An environmentally and financially successful remanufacturing process requires careful consideration of energy use. Energy efficiency is the core of any strategic approach to guarantee cost-effective energy conservation and environmental burden reduction [3]. China has been continuing its effort to highlight efficient energy consumption in industries. The central government initiated a mandatory regulation as a part of the 13th Five-Year Plan for a 15% improvement in energy intensity from 2015 to 2020 [4]. With increasing policy pressure and market competition, improvement in energy efficiency has become an important objective for remanufacturers.

As highlighted in prior studies [5,6], scheduling and planning can directly influence the overall performance, such as improved quality, reduced cost, and enhanced efficiency, of remanufacturing systems. Previous studies on energy savings in remanufacturing systems have focused on assessment rather than detailed approaches for energy reduction [7–9]. Apart from technological innovations and process redesign, production scheduling can be used as an energy- and cost-reduction approach for remanufacturing; it requires modest capital investment. Lage Junior and Godinho Filho [10] utilized a dynamic programming method to solve the problem of disassembly scheduling in clutch remanufacturing for cost minimization. Sun et al. [11] optimized the acquisition lot size of returned cores and scheduled the remanufacturing sequences to minimize the average total cost. However, they did not consider the uncertainties of remanufacturing.

The uncertainties of remanufacturing originate from the fact that remanufacturing regards waste products as work blanks; this is the difference between manufacturing and remanufacturing systems. Decommissioned products had experienced varying operation conditions during their service life. Consequently, recycled products exhibit different damage forms and degrees. Stochastic return and quality variation of cores cause difficulty and complexity in the modeling of remanufacturing systems. Uncertainty occurs in the remanufacturing process, such as remanufacturing routes, operation time, remanufacturing rate, and processing cost, due to the uncertainty propagation effect. As stated by Ref. [12], planning and control of remanufacturing operations are more complicated than those of conventional manufacturing due to the high degree of variability. The quantification of uncertainty expressed in the form of a possibility distribution, fuzzy set, or rough set usually relies on large amounts of statistical data. Many previous studies on remanufacturing uncertainty focused on the input side, namely, quality grades [11], job categories [13], and purchase volumes of new components [14]. Another difference between the two manufacturing paradigms is that each machine is generally assumed to handle one job at a time in conventional manufacturing, whereas remanufacturing cleaning equipment allow the simultaneous cleaning of multiple components, thereby enhancing the difficulty of modeling and programming.

Compared with scheduling for sustainability in conventional manufacturing, scheduling for the energy saving of the remanufacturing process under uncertainty has rarely been addressed. To address this deficiency, the present study developed a fuzzy job-shop scheduling method to minimize the energy consumption in the crankshaft remanufacturing process. On-site investigation was conducted at SINOTRUK, Jinan Fuqiang Power Co., Ltd., a large engine remanufacturer in China. The primary uncertainties of interest were the operation time of each technical process and the remanufacturing routes. Uncertainties of processing time were presented in the form of fuzzy numbers. Considering that remanufacturing is usually composed of different sequential subprocesses and involves diverse machines, we focused on the multimachine level, namely, the process chain of crankshaft remanufacturing. Given that the job-shop scheduling problem (JSSP) is a typical combination optimization problem, we developed an improved adaptive genetic algorithm (IAGA) by using a novel adaptive mechanism and examined its effectiveness by comparing it with other algorithms. This study is expected to help engine remanufacturers achieve increased energy saving in manufacturing and additional reductions in power cost and environmental burden.

2 Literature review

Uncertain parameters should be incorporated to form a new type of JSSP, namely, fuzzy JSSP (FJSSP), because using crisp values in practical manufacturing problems is not necessarily feasible. FJSSP, however, is still in its early development, and it can be roughly classified according to the considered uncertain parameters, such as processing time and due date. Introducing fuzzy precedence constraints and variables to FJSSP allows it to approximate real-world situations. However, FJSSP for energy optimization is less investigated compared with conventional JSSP. Another problem is that in the scheduling community, no consensus has been reached regarding the criterion to approximate fuzzy numbers. The rest of this part discusses the state of the art in these two aspects.

2.1 Energy-efficient scheduling under uncertainty

Scheduling for energy efficiency, also referred to as energy-oriented or energy-aware scheduling, typically adopts energy-related issues as optimization objectives. Uncertain parameters in manufacturing systems affect the completion time, idle time, and work loads of machines and further interfere with the overall energy use. FJSSP occurs frequently in uncertain environments and enhances the complexity of scheduling. Therefore, maximizing energy conservation through scheduling under uncertainty deserves an in-depth investigation.

Singh et al. [15] proposed an online non-clairvoyant job scheduling algorithm to minimize flow time and energy consumption. Liu et al. [16] addressed flow JSSP in auto tire manufacturing with state-dependent setup times. Their model considered the uncertainty of processing time and due date to minimize the total energy use and tardiness. They combined the classical genetic algorithm (GA) with the common pattern matching scheme and probabilistic heuristics-based operators to solve the scheduling problem. However, the superiority of the proposed algorithm was not fully demonstrated. Shrouf et al. [17] built a mathematical model for single machine scheduling with the goal of energy cost minimization. Launch time, off-on time, and idle time were determined by GA in consideration of the electricity cost variation during the production process. A comparison of heuristic and analytical solutions proved that the heuristic algorithm is favorable for large-scale problems. However, scheduling at the unit process level merely determines the state consequences of the machine and cannot determine the actual consequence of jobs. Aside from the uncertainty of the process time and due date, uncertain factors, such as stochastic breakdown of machines and new arrival of jobs, were considered in the work of Ref. [18]. The researchers developed a novel algorithm based on particle swarm optimization to search for the Pareto optimal solution of energy consumption and makespan. Numerical experiments were performed to evaluate the performance of the proposed algorithm. Their work focused on algorithm innovation, and the experiments were performed on fictitious case studies rather than real-world manufacturing practices.

The current literature survey indicates that energy-efficient scheduling studies with one or multiple objectives have concentrated on the conventional manufacturing system that possesses specific uncertainties. However, waste product return is usually recognized as an exogenous process in which return time, quantity, and quality are beyond the direct control of remanufacturers. Compared with traditional manufacturing, remanufacturing involves distinct uncertainties with a higher degree derived from stochastic return and quality variation. Although many studies have addressed the scheduling problem for energy efficiency, very few have examined energy-efficient scheduling for remanufacturing in consideration of specific uncertain factors. This deficiency implies the necessity of energy efficiency improvement for remanufacturing. Similar to many discrete manufacturing processes, the crankshaft remanufacturing process is modeled as an FJSSP in consideration of the fuzzy processing time and stochastic processing routes.

2.2 Operations on fuzzy numbers

In most real-world cases, the parameters involved in the production process are not deterministic or simply represented by crisp values. To deal with uncertain conditions, fuzzy set theory and probability theory are generally used with pre-established fuzzy numbers and distribution [16]. According to the summary of Ref. [19], the commonly used methods to represent due date and processing time in FJSSP are double fuzzy numbers, triangular fuzzy numbers (TFN), and trapezoidal fuzzy numbers. Among them, TFN is the most prevalent alternative. Thus, we discuss operations regarding TFN in this section. The most important arithmetic and logical operations of TFN in FJSSP are addition and max. The addition operation is generally applied to calculate the completion time, and the max operation determines the beginning time. The ranking method compares the maximum fuzzy completion time [20]. For two given TFNs, $t=(t_1, t_2, t_3)$ and $s=(s_1, s_2, s_3)$, the three elements in a TFN (cost-type fuzzy parameter) refer to optimistic, most plausible, and pessimistic values, respectively. The addition operation can be presented as follows:

$$s+t=(s_1+t_1,\ s_2+t_2,\ s_3+t_3). \tag{1}$$

The ranking method proposed by Ref. [21] has been widely used in numerous subsequent studies [20,22–25]. The ranking of TFNs s and t adopts the following criteria:

Criterion I: If $c_1(s)=\dfrac{s_1+2s_2+s_3}{4}>c_1(t)=\dfrac{t_1+2t_2+t_3}{4}$, then $s>t$ and vice versa;

Criterion II: If $c_1(s)= c_1(t)$, then let $c_2(s) = s_2$ and $c_2(t) = t_2$. If $c_2(s) > c_2(t)$, then $s > t$ and vice versa;

Criterion III: If $c_2(s) = c_2(t)$, then let $c_3(s) = s_3-s_1$ and $c_3(t)= t_3-t_1$. If $c_3(s) > c_3(t)$, then $s > t$ and vice versa.

The prevalent max operations of TFNs were proposed by Refs. [20,26]. In Sakawa's model, the approximate maximum is essentially a TFN composed of triple values from s and t. The criterion is presented as follows:

$$s\vee t\approx(s_1\vee t_1,\ s_2\vee t_2,\ s_3\vee t_3). \tag{2}$$

Meanwhile, Lei's model [27] is primarily based on the ranking method, and the approximate maximum of TFN follows the criterion:

$$\text{If } s>t, \text{ then } s\vee t=s;\ \text{else } s\vee t=t. \tag{3}$$

Different from Sakawa's criterion that captures values from both TFNs, Lei's criterion results in either one of the two TFNs. A comparison of the two methods performed by Ref. [20] indicated that in most cases, Lei's model generates a smaller or similar approximation error compared with Sakawa's criterion. To mitigate possible errors of the operations of TFNs, Liu et al. [16] recently developed a novel approximate maximum method. A comparison of the three approaches is presented in Fig. 1. The membership functions of fuzzy numbers are piecewise, continuous, and convex; thus, they are suitable for coding implementation. Based on fuzzy set theory, the computer procedure proposed by Ref. [16] results in an accurate fuzzy maximum. However, this method is complex, particularly when used in coding for JSSP. The present study uses Lei's criterion, which has been extensively applied to many previous fuzzy scheduling problems due to its conciseness and effectiveness [22,24].

In several optimization problems, defuzzification for converting a fuzzy number to a crisp number is required for quantitative comparison. A simple procedure to de-fuzzify the fuzzy number uses the mean value of the fuzzy number with TFNs as an example, as indicated in Eqs. (4) [19], (5) [28], and (6) [29]. Equation (7) [30] presents the defuzzification method integrated with a centroid function, which is a physically prevalent defuzzification function.

$$S=(s_1+2s_2+s_3)/4, \tag{4}$$

$$S=[(s_3-s_1)+(s_2-s_1)]/3+s_1, \tag{5}$$

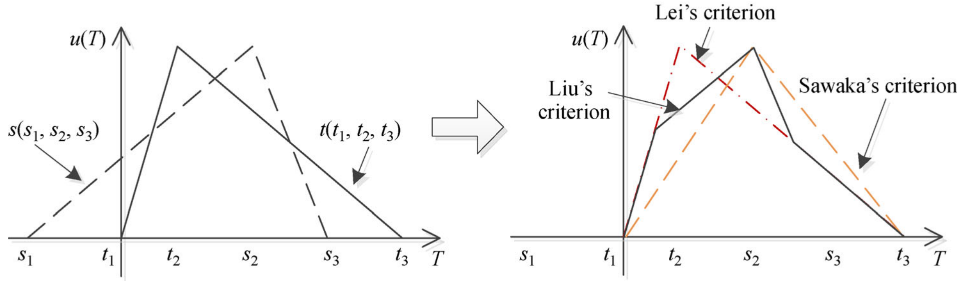

Fig. 1 Comparison of three approximate maximum methods.

$$S = s_2 + [(s_3 - s_2) - (s_2 - s_1)]/4, \tag{6}$$

$$S = \frac{\int u(s) s \mathrm{d}t}{\int u(s) \mathrm{d}t}, \tag{7}$$

where S is the crisp value of TFNs and $u(s)$ is the membership function. An intuitive meaning reflected by Eq. (7) is that the crisp value is the center of the area under the membership function curve.

3 System and problem description

3.1 Remanufacturing system description

Crankshaft remanufacturing generally includes disassembly, cleaning, testing, component reprocessing, and reassembly. After engine disassembly, all of the used components indiscriminately undergo cleaning and testing. However, each type of component has its specific processing route, and even identical types of used components with diverse damage degrees experience different remanufacturing process routes and varying processing times. This condition induces scheduling problems in machine allocation and job sequencing for energy efficiency. This study focuses on the scheduling of the crankshaft reprocessing step. According to our investigation at the engine remanufacturer, two processing routes are applied to returned crankshafts, which are roughly classified as slightly damaged and severely damaged. As stated by Ref. [31], a critical technical barrier in the remanufacturing industry is the lack of technical standards and specifications. The classification of used crankshafts in practice is roughly based on the inspection of the wear dimension. Abrasion over 0.04 mm (for the Φ82 journal) and 0.09 mm (for the Φ100 journal) is regarded as severe damage. Details on these processing routes are displayed in Fig. 2. The restoration of waste crankshafts requires seven processes for the severely damaged parts and five for slightly damaged ones. As shown in Fig. 2, the fourth and seventh processes are cleaning steps and share the same machine (m5), and P3, P5, and P6 have two parallel machines. The buffer capacity before each process is assumed to be infinite. Notably, the cleaning process (P4 and P7) can deal with multiple components simultaneously, which is different from the prior assumption in conventional manufacturing, i.e., one machine at most for one job at a time.

Generally, machines in the system have four states: Starting up, idle, processing, and shutdown. Given that the time duration of starting up and shutdown is short, energy use in these phases is disregarded in this study, which is in line with previous scheduling problems [16]. Table 1 lists the power of machines under idle and operation states and the duration of each process in the form of TFN.

3.2 Problem description

For the crankshaft remanufacturing process, FJSSP comprises a total of n jobs J_i ($i = 1, 2, \ldots, n$), including severely and slightly damaged crankshafts. Considering

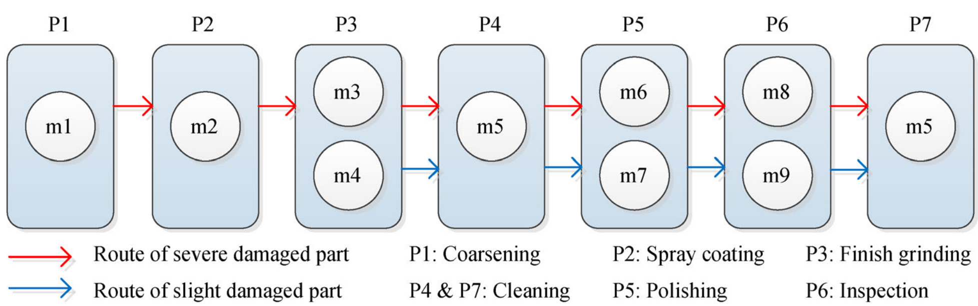

Fig. 2 Configuration of the crankshaft remanufacturing process.

Table 1 Energy-rated information on workshop equipment

Equipment	Processing capability	Operation power/kW	Idle power/kW	Time duration/min
m1	Surface coarsening	1.50	0.9	(1.4, 2, 2.9)
m2	Spray coating	20.00	12.0	(0.8, 1, 1.6)
m3	Grinding	1.50	0.9	(8, 10, 12)
m4	Grinding	2.00	1.2	(7, 8, 11)
m5	Cleaning	42.76	–	(2, 2.5, 3.1); (2.2, 3, 3.5)
m6	Polishing	2.50	1.1	(4.2, 5, 6.1)
m7	Polishing	3.00	1.4	(3.2, 4, 4.9)
m8	Dimension inspection	–	–	(3.9, 5, 6.1)
m9	Dimension inspection	–	–	(3.5, 4, 5.5)

Note: Dimension inspection is implemented manually, and the time durations (two TFNs) of m5 refer to cleaning in P4 and P7.

the different processing routes of crankshafts shown in Fig. 2, we assume that the slightly damaged crankshaft has two virtual processes (P1 and P2) operating on the virtual machine (m10). The relevant operation time and energy consumption are zero. Therefore, each job consists of seven operations, and FJSSP has 10 machines. For processes P3, P5, and P6, the operation o_{ij} ($j = 1, 2, ..., 7$) denoting the jth operation of the ith job can be performed on multiple machines. The TFN $\tilde{t}_{ijk} = (t^1_{ijk}, t^2_{ijk}, t^3_{ijk})$ indicates the processing time of operation o_{ij} on machine M_k ($k = 1, 2, \ldots, 10$).

Given that crankshaft damage is classified into two types, the returned crankshafts can be described by a 0-1 matrix $R_{n\times 2}$, in which the row and column represent the returned crankshaft (or job) and processing route (or damage type), respectively. For example, element "1" in the first column (processing route for severely damaged components) of matrix $\boldsymbol{R}$ indicates that a job should be processed through this route, and "0" means not processed. The ratio of the quantity of "1" in the first column to total jobs n reflects the possibility of severely damaged parts. Determination of possible damages mainly depends on statistical data. As presented in Eq. (8), the matrix $\boldsymbol{H}_{2\times 7}$ shows the machines of operations in the two routes.

$$\boldsymbol{H} = \begin{bmatrix} 1 & 2 & (3{,}4) & 5 & (6{,}7) & (8{,}9) & 5 \\ 10 & 10 & (3{,}4) & 5 & (6{,}7) & (8{,}9) & 5 \end{bmatrix}. \tag{8}$$

The first and second rows of matrix $\boldsymbol{H}$ indicate the operations for severely and slightly damaged crankshafts, respectively. In this matrix, "10" denotes the virtual machine, and the two numbers in parentheses suggest that either of the machines is optional. Matrix $\boldsymbol{JM}$, the product of matrices $\boldsymbol{R}$ and $\boldsymbol{H}$, denotes the operations and corresponding machines for the crankshafts required to be remanufactured.

The following commonly used hypotheses are adopted in this FJSSP: (1) All jobs share the same priority; (2) all machines are available at time 0; (3) an operation cannot be interrupted until it is completed on a machine; and (4) each operation of a job is only processed on no more than one machine at a time. The objective of this study is to minimize the energy consumption E of remanufacturing returned crankshafts. As shown in Eq. (9), the total energy use has two parts, namely, processing energy consumption $E_{\rm pr}$ and idle energy consumption $E_{\rm idle}$.

$$\begin{aligned} E &= E_{\rm pr} + E_{\rm idle} \\ &= \sum_{i=1}^{n}\sum_{j=1}^{7}\sum_{k=1}^{10} x_{ij}^{k} E_{\rm o}(i,j,k) \\ &\quad + \sum_{i=1}^{n}\sum_{j=1}^{7}\sum_{k=1}^{10} E_{\rm id}(i,j,k), \end{aligned} \tag{9}$$

$$E_{\rm o}(i,j,k) = P_{{\rm o},k} t_{ijk}, \tag{10}$$

$$E_{\rm id}(i,j,k) = P_{{\rm id},k}\{\max(ST_{i,j}^{k},\ CT_{i,j-1}^{k'}) - ST_{i,j}^{k}\},$$

$$E_{\rm id}(i,j,k) = P_{{\rm id},k}\{\max(ST_{i,j}^{k},\ CT_{i',j}^{k}) - ST_{i,j}^{k}\}, \tag{11}$$

where $E_{\rm o}(i, j, k)$ and $E_{\rm id}(i, j, k)$ indicate the processing and idle energy consumption of the jth operation of the ith job on the kth machine, respectively, $P_{{\rm o},k}$ and $P_{{\rm id},k}$ denote the operation and idle power, respectively, and x_{ij}^{k} is the engagement indicator of the kth machine. When the jth operation of the ith job is performed on the kth machine, $x_{ij}^{k} = 1$; otherwise, $x_{ij}^{k} = 0$. $ST_{i,j}^{k}$ and $CT_{i,j-1}^{k'}$ mean the starting time of the kth machine (i.e., the available time of the kth machine for the jth operation) and the completion time of the (j–1)th operation on the k'th machine, respectively. The maximization operation in Eq. (11) adopts Lei's criterion. The constraints on machines and jobs in this FJSSP are described in Eqs. (12)–(17).

$$\begin{cases} ST_{ij}^{k} \leqslant CT_{ij}^{k}, \\ CT_{ij_0}^{k} + t_{ij_0k} \leqslant CT_{ij}^{k'}, \end{cases} \forall i,j,j_0,\ i \in [1,\ 2,\ ...,\ n],$$

k and $k' \in [1, 2, ..., 10]$, j and $j_0 \in [1, 2, ..., 7]$,

$$j_0 < j, \tag{12}$$

$$ST_{ij}^{k} + t_{ijk} \cdot x_{ijk} = CT_{ij}^{k}, \quad \forall i,j,k, \ i \in [1, 2, ..., n],$$

$$k \in [1, 2, ..., 10], j \in [1, 2, ..., 7], \tag{13}$$

$$\sum_{k=1}^{10} x_{ij}^{k} = 1, \ \forall \ i,j, \tag{14}$$

$$x_{ij}^{5} = 1, \ y|o_{i4} \text{ or } y|o_{i7}, \tag{15}$$

$$CT_{ij}^{k} - CT_{i'j}^{k} + M \cdot (1 - x_{ij}^{k}) \geqslant t_{ijk}, \ \forall i,j,k \neq 5, \ i \neq i', \tag{16}$$

$$d \geqslant \max(CT_{ij}^{k}), \ \forall i,j,k, \tag{17}$$

where the constraint set Eq. (12) specifies the precedence relationship of two successive operations of a job implemented on the kth and k'th machines. The new task of a machine can be initialized after task completion on this machine, and the new operation of a job can be started after the completion of the previous operation. Interruption of the operation on machines is forbidden, as implied in constraint Eq. (13). Constraint Eq. (14) ensures that at least one machine is available for any operation of a job. The cleaning equipment in the remanufacturing process can deal with multiple waste components simultaneously, which is different from the assumption in previous studies that each machine only processes one job at a time. Constraint Eq. (15) limits the starting condition of the fifth machine, namely, the cleaning process. y is the quantity of crankshafts simultaneously processed in a cleaning batch. The cleaning equipment can be initiated if the amounts of o_{i4} or o_{i7} are evenly divisible by y. Constraint Eq. (16) indicates that all of the machines, except for the fifth one, can process only one operation at a time. M is a large real number. Constraint Eq. (17) shows that the completion time (pessimistic value) of the last job would not exceed due date d.

In the 84 returned crankshaft samples collected at Jinan Fuqiang Power Co., Ltd., the probabilities of severely and slightly damaged components are approximately 0.25 and 0.75, respectively. In this study, we assume that the remanufacturing comprises four severely damaged and eight slightly damaged components and transform the case into a 12×7 (number of jobs × number of operations) FJSSP.

4 Solution algorithm

GA is a widely applied heuristic algorithm for JSSP. However, traditional GA has an inherent drawback, that is, the constant crossover probability and mutation probability fail to regulate the convergent process and result in premature convergence. To efficiently and accurately obtain the solution of this FJSSP, we propose an improved adaptive GA for energy minimization of the crankshaft remanufacturing process. The basic processes are presented in Fig. 3. The following sections describe the main points illustrated in this figure.

4.1 Initial population

The generation of chromosomes or populations in GA depends on the encoding process. In this study, a dual layer representation method is proposed for encoding, as shown in Fig. 4. A schedule can be represented by two integer strings: Job sequencing and machine allocation strings. Matrix $\boldsymbol{JM}_{12\times7}$ indicating all the processing operations of returned components depicted in Eq. (18) is used to understand the encoding process thoroughly. The elements in $\boldsymbol{JM}$ refer to machine numbers, and the two numbers in parentheses in matrix $\boldsymbol{JM}$ suggest two optional machines for relevant operations. Additionally, the columns denote the seven operations, and the first and sixth rows refer to the processing route for severely damaged components.

$$\boldsymbol{JM} = \begin{array}{c} 1 \\ \vdots \\ 6 \\ \vdots \end{array} \begin{bmatrix} 1 & 2 & (3,4) & 5 & (6,7) & (8,9) & 5 \\ \vdots & & & \vdots & & & \\ 1 & 2 & (3,4) & 5 & (6,7) & (8,9) & 5 \\ \vdots & & & \vdots & & & \end{bmatrix}. \tag{18}$$

The job sequencing and machine allocation strings contain 84 elements because the problem size is 12×7. As indicated in Fig. 4, the job sequence determines the operation orders of components. For example, the first three numbers "1", "5", and "6" appearing for the first time pertain to the first operation of the first, fifth, and sixth components, i.e., o_{11}, o_{51}, and o_{61}. The second and third "6" in the job sequencing string refer to the second and third operations of the sixth component, namely, o_{62} and o_{63}. The machine allocation string determines the machine candidate for each operation. For example, given that the first operation is conducted on the unique machine (#1) shown in matrix $\boldsymbol{JM}$, the first "1" in the machine allocation string corresponding to o_{11} refers to machine #1. The "2" in this string corresponding to o_{63} denotes the second candidate, namely, machine #4, because the third operation can be operated on two machines (#3 and #4). Each randomly generated individual in the initial population can be regarded as a feasible solution of FJSSP, and the relevant chromosome is jointly determined by the job sequencing and machine allocation strings.

To generate feasible chromosomes for the scheduling, the encoding procedure is implemented as follows:

Step 1: Generate an indicator vector $\boldsymbol{v}_{1\times12}$ and set all

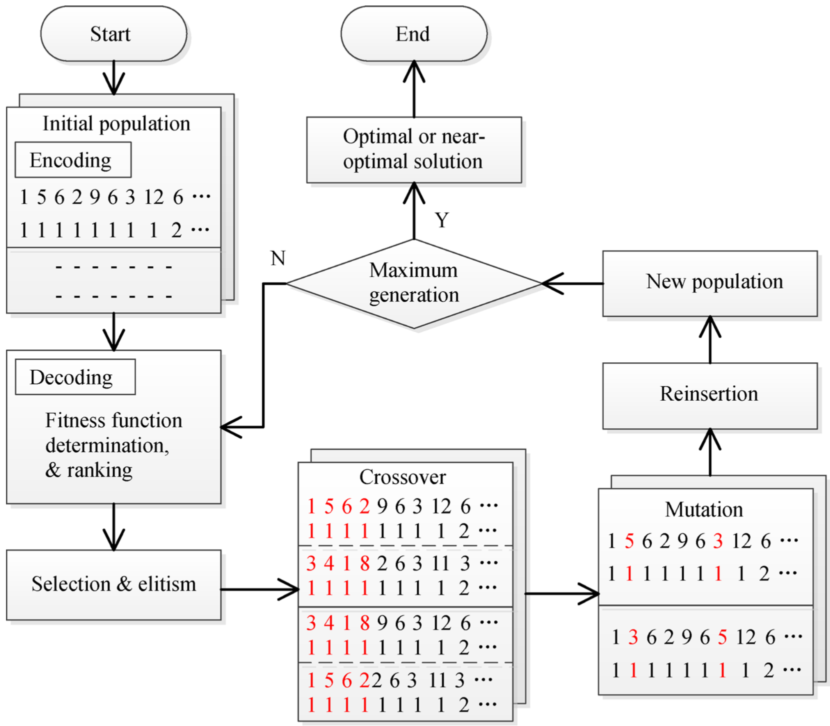

Fig. 3 Implementation procedure of a GA for FJSSP.

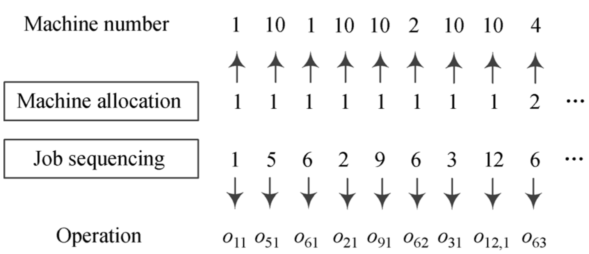

Fig. 4 Encoding method for the generation of the initial population.

elements equal to 7.

Step 2: Randomly create an integer belonging to [1, 12] as a job sequencing element and location sign. The value in vector $\boldsymbol{v}$ located by this integer minus 1. With this integer (component number), the corresponding value (remaining operations) in vector $\boldsymbol{v}$, and matrix $\boldsymbol{JM}$, the amounts of optional machines could be determined.

Step 3: Repeat Step 2 for 84 times until vector $\boldsymbol{v} = \boldsymbol{0}$ then produce a chromosome for scheduling.

Step 4: Repeat Steps 1–3 above for N times to generate the required initial populations.

However, several chromosomes generated by these steps are illegal because the cleaning process (P5) can process multiple components (supposedly two here) at a time, and this necessitates the checking and modification of these chromosomes. Checking and modification are also applied to crossover and mutation, in which illegal chromosomes might be generated. The detailed procedure is as follows:

Step 1: Check and modify the job sequencing string ($S1$) number-wisely.

Step 1.1: Extract a number "x" from $S1$. If the quantities of "1" or "4" ("1" and "4" refer to remanufacturing cleaning) in vector $\boldsymbol{v}$ are less than 2 and $v(x)$= 1 or 4, then shift x to the end of $\boldsymbol{v}$ and continue with Step 1.1.

Step 1.2: If the quantities of "1" and "4" in vector $\boldsymbol{v}$ are not less than 2 and $v(x)$= 1 or 4, then randomly select a location y of "1" and "4" in vector $\boldsymbol{v}$ and assign x and y to new job sequencing string $S2$, $v(x)$= $v(x)$–1. If $v(y)$= $v(y)$–1 assign y to vector $\boldsymbol{pos}$.

Step 1.3: If x is a member of vector $\boldsymbol{pos}$, then assign 0 to the location of x in vector $\boldsymbol{pos}$; otherwise, assign x to the chromosome $S2$, $v(x)$= $v(x)$–1.

Step 1.4: Repeat Steps 1.1–1.3 until all the numbers in the original job sequencing string $S1$ have been extracted.

Step 2: Check and modify the machine allocation string ($A1$) number-wisely.

Step 2.1: Extract a number x from $S2$, find the first x in $S1$, locate its position p, $S1(p)$= $S1(p)$–1, and assign the corresponding machine number in $A1$ to a new machine allocation string $A2$, $A2(p)$= $A1(p)$.

Step 2.2: Repeat Step 2.1 until all the numbers in $S2$ have been extracted.

The chromosomes created by these steps correspond to specific Gantt charts. Figure 5, which is in the form of a fuzzy Gantt chart, reflects the operations and machines in the Fig. 4. Different from the traditional Gantt chart using actual processing time, the fuzzy Gantt chart utilizes TFNs

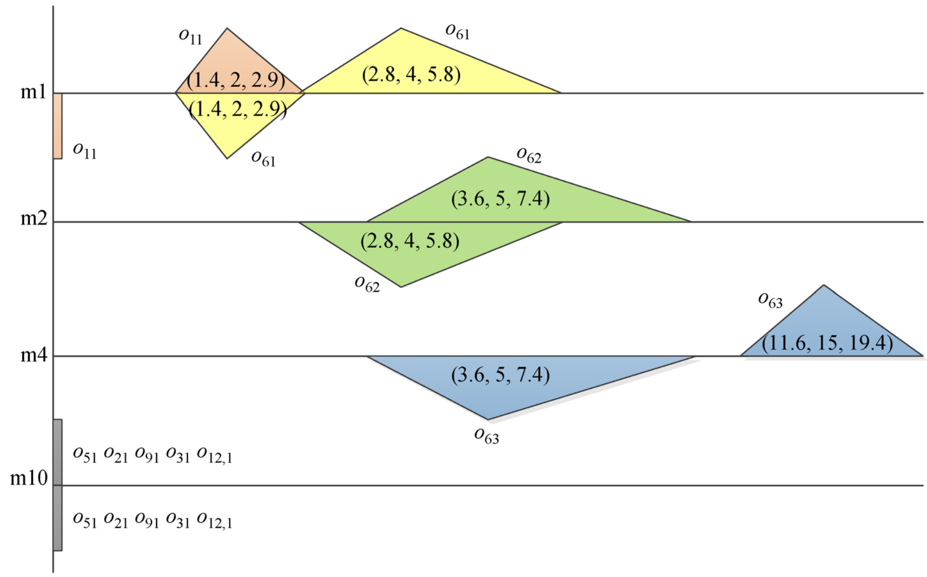

Fig. 5 Fuzzy Gantt chart based on chromosomes.

to represent the time of operations. TFNs under the solid line are the starting time of the operation, and those above the solid line refer to the completion time.

4.2 Decoding and fitness function

The decoding procedure compiles the chromosomes into the energy consumption of remanufacturing the used crankshafts. Similar to the population generation, the remanufacturing process should focus on decoding. The relevant steps for decoding a chromosome are as follows:

Step 1: Extract numbers j and k from the job sequencing and machine allocation strings, respectively, and determine the corresponding job, machine, operation number, available time ($ST^k_{i,j}$) of machine k, and completion time ($CT^{k'}_{i,j-1}$) of the prior operation of job j.

Step 2: If $k \neq 5$ (remanufacturing cleaning equipment), then determine the energy consumption according to Eqs. (9)–(11) and record the completion time of this operation and machine.

Step 3: If $k = 5$, then extract several numbers ($j+1$ and $k+1$) that also experience an identical cleaning process, determine the energy consumption, and record the completion time of the machine and operations of two jobs.

Step 4: Repeat Steps 1–3 until all numbers in the string have been extracted.

Generally, the objective function is regarded as a basis for the fitness function. However, this method has a scaling problem and causes premature convergence in low selection pressure (SP) [32]. This study uses rank-based fitness assignment to overcome this drawback and demonstrates that this assignment has good robustness [32]. Equation (4) is used to transform the TFNs into crisp values for the ranking because the objective function values are TFNs. The fitness value of individuals (Fit) can be determined by Eq. (19) in a linear ranking form:

$$Fit(Pos) = 2 - SP + \frac{2(SP-1)(Pos-1)}{N-1},$$

$$SP \in [1.0,\ 2.0], \tag{19}$$

where N is the size of the population, Pos denotes the individual ranking position in the population, and SP is the selection pressure. SP in the present study is fixed at 2, and the fitness values of populations range from 0 to 2.

4.3 Selection and elitism

Notable selection techniques include roulette wheel, tournament, and local selection. In this study, for the population of one generation, 80% of individuals are selected for reproduction by using the stochastic universal sampling method. The optimal solution in these selected individuals is stored as an elite and directly added to the next generation without gene recombination by crossover or mutation, which could enable enhanced propagation of the individual with the highest fitness value. The selected chromosomes experience crossover and mutation in the evolution of populations toward the global optimal solution.

4.4 Crossover

As a basic operator for producing new offspring, crossover creates new chromosomes that inherit features or genetic materials from both parents. The power of GA stems from the crossover process, which induces a randomized and structured exchange of genes between two chromosomes with the possibility of producing "better" solutions [33]. The commonly used crossover approaches are single-point, multi-point, and uniform types. This study adopts a five-point crossover method.

The crossover rate or crossover possibility (p_c) and mutation possibility (p_m) are unchanged in traditional GA, which might limit the search capability and cause premature convergence and a low convergence rate. The balance between the two characteristics, namely, capacity to converge to the optimum and capacity to explore new areas, is controlled by p_c and p_m. No uniform guidance is available to determine the values of p_c and p_m in specific cases. In most GA practices, p_c is moderately large (0.5–1.0), and p_m is relatively small (0.001–0.05) [33]. The trade-off between exploitation and exploration is realized by adaptively varying p_c and p_m. The fundamental idea is that the values of p_c and p_m are closely related to the fitness values of populations. These values increase when solutions are stuck in a local optimum, and they decrease when solutions are scattered in the solution space. The standard deviation σ of the fitness values is a yardstick for adaptive modification. σ is likely to be small when individuals converge to an optimum. In this regard, p_c and p_m should be increased to avoid the local optimum. When the population tends to converge to the global optimum, high p_c and p_m may disrupt the near-optimal solutions. Therefore, different individuals in one population with high fitness values should be subject to low crossover and mutation rates in order to preserve "good" individuals.

An adaptive technique of the crossover and mutation possibility is derived from the hormone modulation mechanism. Farhy et al. [34] proposed a general law for hormone secretion ($F(G)$) by hormone glands. $F(G)$ exhibits obvious non-negativity and monotonicity. The rising function $F_{\text{up}}(G)$ and declining function $F_{\text{down}}(G)$ for hormone adjustment are consistent with the Hill function, as shown in Eqs. (20) and (21):

$$F_{\text{up}}(G) = \frac{G^n}{T^n + G^n}, \tag{20}$$

$$F_{\text{down}}(G) = \frac{T^n}{T^n + G^n}, \tag{21}$$

where T ($T > 0$) is the threshold, G is an independent variable, and n ($n \geqslant 1$) refers to the Hill parameter. T and n jointly determine the slope of the curve. Figure 6 displays the rising and declining functions with $T = 1$. Notably, F rises or declines exponentially with G, which implies swift hormone modulation with external variable G. Supposing that the excretion of hormone $h1$ is affected by hormone $h2$, the relation between the excreting speed (F_{h1}) of $h1$ and the concentration (G_{h2}) of $h2$ can be expressed by Eq. (22):

$$F_{h1} = aF(G_{h2}) + F'_{h1}, \tag{22}$$

where F'_{h1} denotes the initial excreting speed of $h1$ and a is a constant.

According to Eqs. (21) and (22) and the analysis of the crossover rate variation, adaptive p_c is designed as follows to accelerate the convergence and promote individual diversity.

$$p_c = p_c^0 + \left(\frac{1}{gen}\right)^{\alpha_1} - \alpha_2 \frac{\sigma^{n_c}}{\sigma^{n_c} + \left(\max(Fit) - Fit(i)\right)^{n_c}}, \tag{23}$$

where gen denotes the number of generations, σ is the standard deviation, $Fit(i)$ is the fitness value of the ith individual, Fit denotes the fitness values of the population, α_1, α_2, and n_c are all constants, and p_c^0 is the initial crossover possibility.

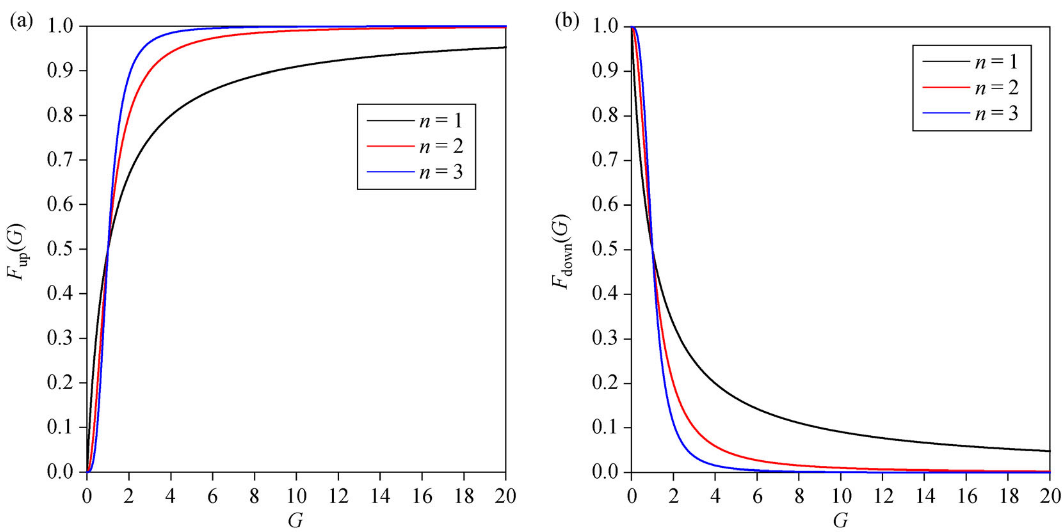

Fig. 6 Curves of the Hill function under varying parameters: (a) Upward curves and (b) downward curves.

Subsequently, the multipoint crossover procedure is implemented as follows:

Step 1: Randomly generate five points to segregate the job sequencing strings of two chromosomes, but these points should not separate the successive machine (#5) in the corresponding machine allocation strings. As shown in Fig. 7, the last segregation point is illegal, and another point should be regenerated.

Step 2: Identify the missing or abundant numbers and locations in offspring 1 and 2. The missing numbers in offspring 1 are the abundant ones in offspring 2 and vice versa.

Step 3: Exchange the abundant numbers in the two offspring.

Step 4: Check and modify the job sequencing and machine allocation strings by using the procedures (Steps 1.1–1.4, 2.1 and 2.2) in Section 4.1.

4.5 Mutation

Mutation in GA involves bringing unexpected genetic materials into the chromosome with certain probability p_{m}. This behavior avoids premature convergence to the local optimum solution. The role of mutation in GA is usually regarded as a background operator and ensures the possibility of searching any areas and recovering "good" genetic materials. Similar to the crossover possibility, mutation possibility p_{m} is also associated with generation number *gen*, standard deviation σ, and maximum fitness value of the population; it can be determined with Eq. (24):

$$p_{\mathrm{m}} = p_{\mathrm{m}}^{0} + \left(\frac{1}{gen}\right)^{\beta_1} - \beta_2 \frac{\sigma^{n_{\mathrm{m}}}}{\sigma^{n_{\mathrm{m}}} + \left(\max(Fit) - Fit(i)\right)^{n_{\mathrm{m}}}}, \tag{24}$$

where β_1, β_2, and n_{m} are constants and p_{m}^{0} is the initial mutation possibility. In the former generations, the mutation rate is higher for an extensive search in the solution space. Individuals with a highly scattered population or a high fitness value require a low mutation rate for the diversity of the population and maintenance of good chromosomes. The mutation procedures are implemented as follows:

Step 1: Randomly select two numbers from the job sequencing string of a chromosome. If their corresponding machine numbers contain "5" (similar to the crossover procedure), then repeat the number selection. Otherwise, exchange these two numbers in the job sequencing string.

Step 2: Legalize the mutated chromosome through the checking and modification of job sequencing and machine allocation strings by using the procedures (Steps 1.1–1.4, 2.1 and 2.2) in Section 4.1.

4.6 Reinsertion and termination

The crossover and mutation processes above are performed on 80% of the selected individuals in a population. Afterward, the generated offspring is inserted into the current population, namely, partly replacing parents with offspring to form a new generation. The insertion is fitness-based selection and replacement. All of the generated offspring after crossover and mutation replace the least-fit parents. The termination condition of this IAGA rests on the maximum generation. If the repetition of the principal steps reaches the predefined maximum generation, then the search work will be terminated. Then the best solution, relevant schedule, convergent generation, and run time will be provided as systematic outputs.

5 Computational results

IAGA is implemented on the MATLAB platform and run on a personal computer with 4.0 GB of RAM and 2.4G CPU under Windows 10. The parameters of this algorithm are set as follows: The population size is 100; the maximum generation number is 90; initial crossover possibility p_{c}^{0} and initial mutation possibility p_{m}^{0} are 0.8 and 0.6, respectively; $n_{\mathrm{c}} = n_{\mathrm{m}} = 1$; $\alpha_1 = \alpha_2 = 0.5$; $\beta_1 = 0.5$; and $\beta_2 = 0.7$. Through the computation of the proposed IAGA, the minimal energy consumption for 12 returned crankshafts is $E = (23.66, 30.54, 37.52)$ kW·h. The relevant operation sequences are presented in Fig. 8. Considering the explicitness of the conventional Gantt chart, we display the operation of jobs in the traditional form instead of the fuzzy Gantt chart (Fig. 5) to avoid inconvenient observation of the figure. A large amount of the resulting

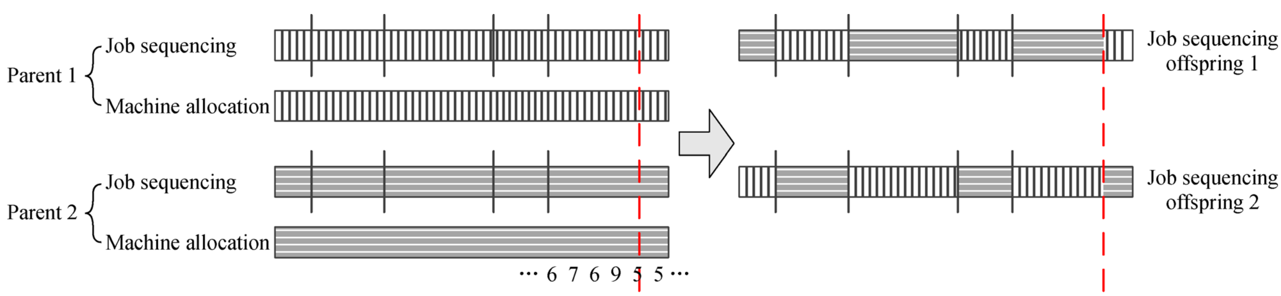

Fig. 7 Multipoint crossover for the job sequencing string.

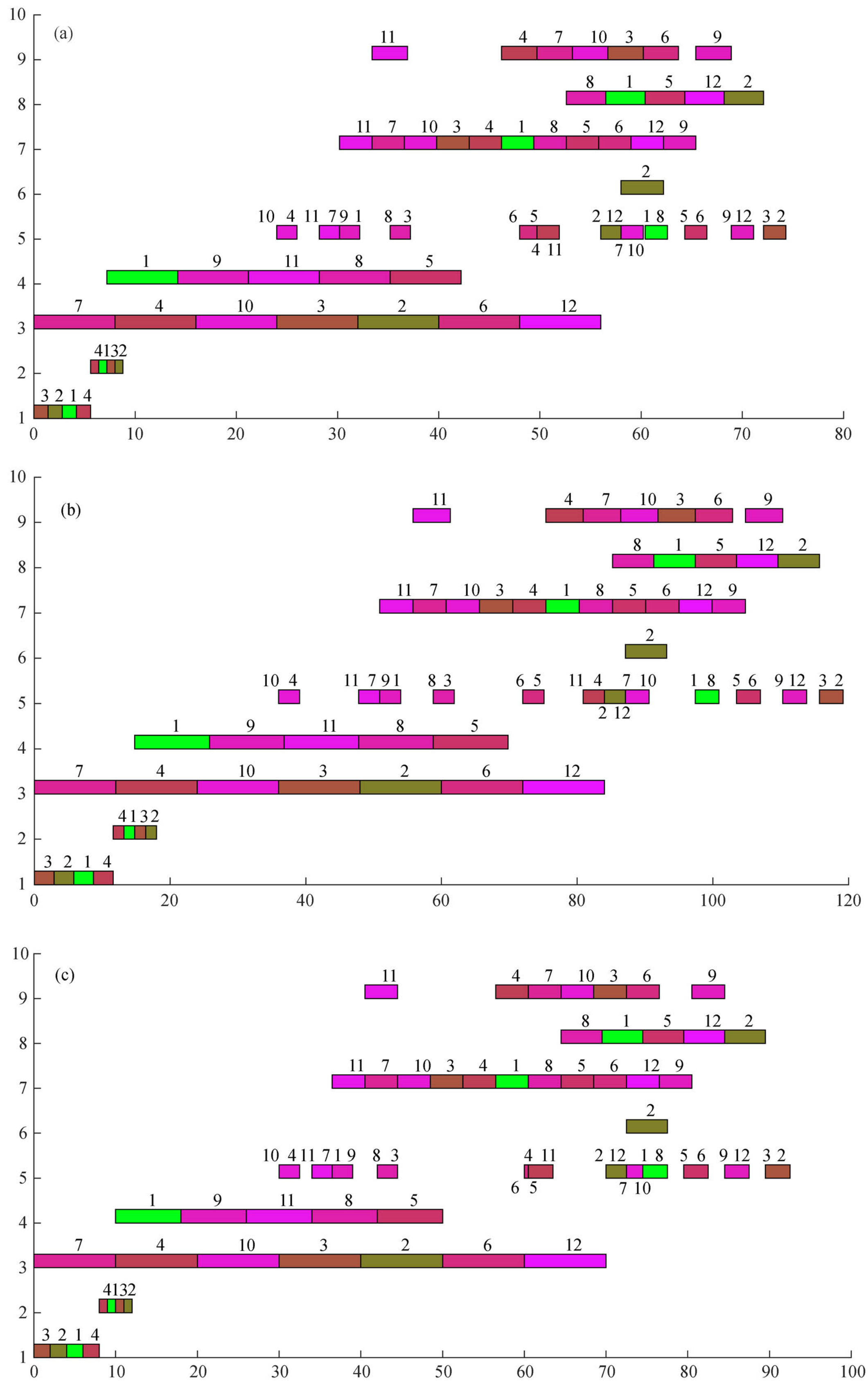

Fig. 8 Gantt chart of an optimal solution in traditional form: (a) Optimistic situation, (b) pessimistic situation, and (c) most plausible situation.

information is decomposed into three situations, namely, optimistic, pessimistic, and most plausible, which is consistent with the fuzzy Gantt chart.

Figure 8 indicates that the remanufacturing cleaning equipment deals with two components simultaneously. Given that each job involves two cleaning operations, the equipment is required to complete a total of 12 operations for all jobs. Another noteworthy point in this figure is that the #6 machine only processes the #2 job during the entire scheduling. Although the rated power of the #6 machine is smaller than that of its counterpart #7 machine, its processing efficiency is much lower. Therefore, the fifth operation of most jobs is arranged on the #6 machine for energy conservation. However, this task may lead to imbalanced utilization of the machine. If the due date variable is fully relaxed or neglected, one machine tool for the polishing operation is presumably enough for crankshaft remanufacturing, which helps reduce the investment budget. To decrease the energy consumption of idleness, machines usually perform processing successively, as reflected in Fig. 8. Given that the #9 machine (manual operation) is free of energy use and the #5 machine shuts down directly after completing an operation, the minimum energy consumption obtained by this algorithm can be regarded as the global optimal value.

Crankshaft remanufacturing belongs to small-lot production. Slightly and severely damaged components share several identical processes. Thus, returned components are not classified but randomly selected to reprocess during the remanufacturing process. According to the 100 legal chromosomes or schedules randomly created using the method of initial population generation, the average energy consumption for the batch of crankshafts is E=(24.94, 32.17, 39.66) kW·h, which implies approximately 1.7 kW·h of energy saving. Therefore, the energy efficiency improvement under the optimal schedule is around 5%.

To demonstrate the superiority of IAGA, we compare this algorithm with traditional GA, an adaptive GA (AGA) proposed by Wei et al. [35], and random key GA (RKGA) [36]. These algorithms share identical parameters, such as generation number, population size, and generation gap. Additionally, apart from the determination of crossover and mutation rates, they experience the same procedures described in Section 4 and are executed under an identical evolutionary environment for a fair comparison. Figure 9 depicts the evolution of the objective variable under these algorithms. The Y-axis means the total energy consumption defuzzied by Eq. (4). As can be observed from Fig. 9, IAGA shows good convergent speed, but it is slightly slower than RKGA. It can obtain better results compared with the other algorithms.

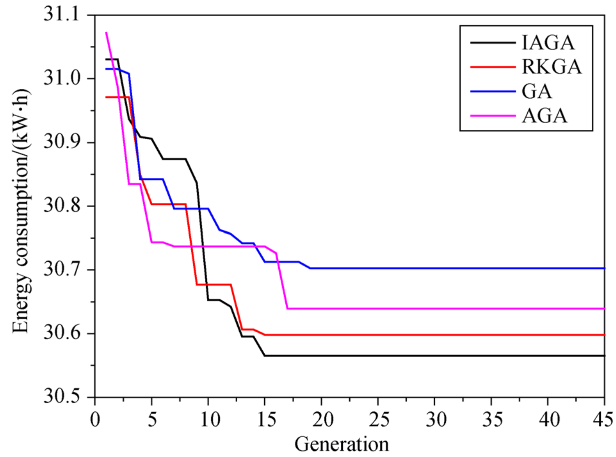

Fig. 9 Comparison of the convergent curves of four algorithms.

To mitigate the effects of random factors, we perform 20 computation trials to examine the performance of the algorithms. A comparison is made in terms of average optimal values, run time, and convergent generation. Table 2 summarizes the computational results of the four algorithms. The disparities in energy consumption in vertical or horizontal comparison are remarkably small. This phenomenon is arguably attributed to the small size of this FJSSP. The average convergent generation number of IAGA is slightly smaller than those of GA and AGA but larger than that of RKGA. This value in the specific algorithm is unstable and presents a large standard deviation in the 20 trials. In addition, the average run time of IAGA is approximately 2.2, 3.3, and 2.7 s less than those of GA, AGA, and RKGA on the average, respectively. Adaptive modification of crossover and mutation can avoid unnecessary crossover and mutation processes, which could result in reduced run time. RKGA presents good performance in convergent speed, and the numerical results are close to those of IAGA. However, the run time is greater than those of GA and IAGA. Although the average convergent generation number and run time of AGA are slightly greater than those of GA, its optimization result is better in terms of minimum, average, and maximum energy consumption. Adaptive crossover and

Table 2 Computational results of four algorithms

Algorithm	Minimum energy consumption /(kW·h)	Average energy consumption /(kW·h)	Maximum energy consumption /(kW·h)	Convergent generation	Run time/s
GA	(23.70, 30.55, 37.59)	(23.79, 30.62, 37.67)	(23.87, 30.65, 37.74)	31.25	13.96
AGA	(23.71, 30.56, 37.55)	(23.76, 30.59, 37.62)	(23.83, 30.63, 37.67)	33.50	15.09
RKGA	(23.68, 30.55, 37.53)	(23.75, 30.58, 37.61)	(23.81, 30.62, 37.66)	19.80	14.42
IAGA	(23.66, 30.54, 37.52)	(23.75, 30.58, 37.61)	(23.80, 30.61, 37.65)	30.25	11.75

mutation possibility rationally enable extensive searches in the solution space, maintenance of diversity, and protection of “good” individuals during the evolution stage. Thus, incorporating the adaptive mechanism into the operator of crossover and mutation provides good results with high possibility.

6 Experimental study

To validate the effectiveness of the IAGA proposed in this study, we use three additional instances with a size (number of jobs $\times$ number of operations) of 10×6, 9×9, and 12×11, hereby denoted as Problems 1, 2, and 3, respectively. Problems 1 and 2 are derived from Li [37] and Peng et al. [38], and Problem 3 is designed as shown in Table A1 in the Appendix for the minimization of energy consumption. Similar to the energy conservation problem, Problem 1 pertains to the minimization of the processing cost. The objective of Problem 2 adopts the case in Ref. [38] to minimize the energy use in the remanufacturing process. The computational results of the four algorithms under these instances are shown in Table 3.

Given that the size of Problem 1 is small, all of the algorithms could find the optimal solution in 20 trials, but the traditional GA is inferior in terms of the average value. In Problem 2, AGA, RKGA, and IAGA have identical optimal values. The average values of RKGA and IAGA overlap, i.e., the pessimistic value of IAGA is less than that of RKGA, whereas the optimistic value of IAGA is greater; most of their plausible values are equal. In Problem 3, IAGA has an edge over the other algorithms in terms of optimal and average values.

7 Discussion and implications

Operation scheduling to facilitate manufacturing sustainability has elicited increasing interest from the industrial community. The reduction of energy consumption in manufacturing is one of the most critical strategies to promote the sustainability of manufacturing. For the three pillars of sustainability (environmental, economic, and social), energy conservation directly facilitates environmental and economic performance. Implementation of energy-aware scheduling of the remanufacturing process would not only improve energy efficiency but also cut the energy cost. Energy-cost-effective scheduling that considers the peak power load was investigated by Ref. [39]. The primary strategy in this study was to shift the energy consumption from on-peak hours to off-peak or mid-peak hours. However, sometimes, doing so in consideration of labor price is unrealistic in practice because at night when the labor cost is high, the electricity price is usually low. Therefore, energy efficiency improvement continues to be the core of energy cost reduction. Given that the regular production period usually focuses on on-peak hours, the electricity reduction (around 1.7 kW·h) in processing 12 returned crankshafts means increased cost saving.

The environmental performance of the manufacturing process has received extensive attention during decision making and is considered an additional objective in multi-objective optimization for production planning and scheduling [40]. Aside from the cost-saving benefit, energy conservation through scheduling lowers the environmental burdens of the remanufacturing system. Electricity generation in China primarily depends on hard coal and releases tremendous amounts of greenhouse emissions [41]. As analyzed above, scheduling for crankshaft remanufacturing improves energy efficiency by approximately 5%. Therefore, the environmental performance proportionally increases.

Crankshaft, as a single research object in this study, is one of the “seven pieces” in a diesel engine; the others are the cylinder head, connection rod, cylinder block, fly wheel, gear box, and fly wheel housing. Comprehensive scheduling that considers all of the components at the factor level or even at the level of the entire supply chain is expected to provide tremendous energy conservation. Thus, follow-up work on this inclusive scheduling is desirable. Energy savings result in considerable economic cost reduction and positive environmental effects, which might further motivate the research on energy-aware scheduling for remanufacturing systems.

Table 3 Computational results on three problems

Instance	Type	Problem 1	Problem 2	Problem 3
GA	Average	(0.403, 0.470, 0.567)	(379.82, 409.86, 439.86)	(54.96, 74.46, 90.59)
	Optimal	(0.358, 0.446, 0.542)	(379.22, 408.94, 438.75)	(53.72, 72.67, 88.18)
AGA	Average	(0.370, 0.456, 0.552)	(379.58, 409.64, 439.62)	(54.60, 73.92, 89.89)
	Optimal	(0.358, 0.446, 0.542)	(378.97, 408.70, 438.46)	(53.60, 72.54, 88.06)
RKGA	Average	(0.358, 0.446, 0.542)	(379.54, 409.53, 439.48)	(54.56, 73.85, 89.82)
	Optimal	(0.358, 0.446, 0.542)	(378.97, 408.70, 438.46)	(53.48, 72.50, 88.02)
IAGA	Average	(0.358, 0.446, 0.542)	(379.68, 409.53, 439.42)	(54.54, 73.70, 89.48)
	Optimal	(0.358, 0.446, 0.542)	(378.97, 408.70, 438.46)	(53.37, 72.33, 88.00)

8 Conclusions

Under legislative pressure and market competition, a successful remanufacturing system requires careful consideration of energy efficiency. This study considers the differences between remanufacturing and conventional manufacturing processes, particularly the uncertain processing time and routes. Uncertain parameters are presented in the form of TFNs. In accordance with the investigation at the engine remanufacturer, the returned crankshafts are roughly classified into two types (slightly and severely damaged) and subjected to two different reprocessing routes. A basic assumption in traditional manufacturing scheduling is that one machine can process only one job at a time. By contrast, the remanufacturing cleaning equipment can handle multiple components simultaneously, which complicates its modeling, algorithm, and programming. Through the introduction of virtual operations and machines, we formulate the crankshaft remanufacturing process into a 12×7 fuzzy scheduling problem. To solve this FJSSP, an IAGA is developed by adopting the hormone modulation mechanism, which can swiftly modulate the crossover and mutation possibility. Comparison with traditional GA and AGA indicates that IAGA is superior in terms of average run-time and convergent generation and minimum, average, or maximum energy consumption. Simulation results indicate that compared with random operation sequences in practice, the optimal scheduling scheme with energy consumption E=(23.66, 30.54, 37.52) kW·h saves approximately 1.7 kW·h of electricity. The energy efficiency of the crankshaft remanufacturing process is increased by around 5%. Energy conservation brings additional benefits, such as cost saving and environmental burden reduction. Therefore, energy-aware job-shop scheduling should be integrated into the sustainability-related decision making of enterprises.

Uncertainties in the remanufacturing system are multivariant and originate from multiple sources. We merely considered the uncertainty of processing time and routes. Other uncertainties, such as returned quantity and time of engines, always exist in the remanufacturing system. Inclusive integration of these uncertainties into scheduling would significantly increase complexity. Future efforts should be exerted to model and consider additional uncertainties. The present study only investigated crankshaft remanufacturing irrespective of remanufacturing other engine components, which also have a large potential for energy conservation. Comprehensive capture of engine parts in scheduling remains a topic for future research. Moreover, the problem size in this study is small. When the scheduling involves a large number of components, such as entire engine parts, the computation process becomes time-consuming. Therefore, enhancement of computational efficiency and robustness should be explored in the future.

Appendix

Table A1 Processing data in Problem 3

Operation	Machine	Operation power/kW	Idle power/kW	Time duration/min
1	#1	3.0	0.9	(2.5, 2.9, 3.4)
	#2	2.5	0.7	(3, 3.5, 4)
2	#3	4.0	1.2	(6, 7.5, 8.5)
3	#4	4.5	1.2	(5, 6, 7)
	#5	4.0	1.0	(5.5, 6.6, 7.5)
4	#6	7.0	2.1	(7, 8, 9)
5	#7	1.1	0.3	(5, 7, 9)
6	#8	6.5	2.3	(4.2, 5.8, 6.9)
7	#9	3.5	0.8	(1.5, 3.0, 4.5)
8	#10	10.0	2.5	(5, 8, 10)
9	#11	5.5	1.8	(2.5, 4.0, 5.5)
10	#12	7.5	2.5	(6, 8, 10)
11*	#13	16.0	4.8	(4, 5, 6)

Note: *: Processing three jobs simultaneously.

Acknowledgements The authors highly appreciate the investigation opportunities provided by SINOTRUK, Jinan Fuqiang Power Co., Ltd. We are also grateful for the financial support from the National Natural Science Foundation of China (Grant Nos. 51775086 and 51605169), and Natural Science Foundation of Guangdong Province China (Grant No. 2014A030310345).

References

1. Du Y, Li C. Implementing energy-saving and environmental-benign paradigm: Machine tool remanufacturing by OEMs in China. Journal of Cleaner Production, 2014, 66: 272–279
2. Xiang W, Ming C. Implementing extended producer responsibility: Vehicle remanufacturing in China. Journal of Cleaner Production, 2011, 19(6–7): 680–686
3. International Energy Agency. World Energy Outlook 2016. Paris: Organization for Economic Co-operation and Development, 2016, 284–285
4. Central People’s Government of China. Notification of energy conservation and emission reduction strategy for “13th Five-Year Plan” delivered by the State Council. Available at the State Council of the People’s Republic of China website, 2018-11-20
5. Chu H, Cao Q, Fei R. MAS-based production scheduling system for manufacturing cell-based workshop. Frontiers of Mechanical Engineering in China, 2006, 1(4): 375–380
6. Jiang Z, Jiang Y, Wang Y, et al. A hybrid approach of rough set and case-based reasoning to remanufacturing process planning. Journal of Intelligent Manufacturing, 2019, 30(1): 19–32
7. Gutowski T G, Sahni S, Boustani A, et al. Remanufacturing and energy savings. Environmental Science & Technology, 2011, 45(10): 4540–4547
8. Sutherland J W, Adler D P, Haapala K R, et al. A comparison of

manufacturing and remanufacturing energy intensities with application to diesel engine production. CIRP Annals-Manufacturing Technology, 2008, 57(1): 5–8
9. Yang S S, Ngiam H Y, Ong S K, et al. The impact of automotive product remanufacturing on environmental performance. Procedia CIRP, 2015, 29: 774–779
10. Lage Junior M, Godinho Filho M. Master disassembly scheduling in a remanufacturing system with stochastic routings. Central European Journal of Operations Research, 2017, 25(1): 123–138
11. Sun H, Chen W, Liu B, et al. Economic lot scheduling problem in a remanufacturing system with returns at different quality grades. Journal of Cleaner Production, 2018, 170: 559–569
12. Guide V D R Jr, Srivastava R, Kraus R E. Product structure complexity and scheduling of operations in recoverable manufacturing. International Journal of Production Research, 1997, 35(11): 3179–3200
13. Zhang R, Ong S K, Nee A Y C. A simulation-based genetic algorithm approach for remanufacturing process planning and scheduling. Applied Soft Computing, 2015, 37: 521–532
14. Wen H, Liu M, Liu C, et al. Remanufacturing production planning with compensation function approximation method. Applied Mathematics and Computation, 2015, 256: 742–753
15. Singh P, Khan B, Vidyarthi A, et al. Energy-aware online non-clairvoyant scheduling using speed scaling with arbitrary power function. Applied Sciences, 2019, 9(7): 1467
16. Liu G S, Zhou Y, Yang H D. Minimizing energy consumption and tardiness penalty for fuzzy flow shop scheduling with state-dependent setup time. Journal of Cleaner Production, 2017, 147: 470–484
17. Shrouf F, Ordieres-Meré J, García-Sánchez A, et al. Optimizing the production scheduling of a single machine to minimize total energy consumption costs. Journal of Cleaner Production, 2014, 67: 197–207
18. Tang D, Dai M, Salido M A, et al. Energy-efficient dynamic scheduling for a flexible flow shop using an improved particle swarm optimization. Computers in Industry, 2016, 81: 82–95
19. Abdullah S, Abdolrazzagh-Nezhad M. Fuzzy job-shop scheduling problems: A review. Information Sciences, 2014, 278: 380–407
20. Lei D. Fuzzy job shop scheduling problem with availability constraints. Computers & Industrial Engineering, 2010, 58(4): 610–617
21. Sakawa M, Kubota R. Fuzzy programming for multiobjective job shop scheduling with fuzzy processing time and fuzzy duedate through genetic algorithms. European Journal of Operational Research, 2000, 120(2): 393–407
22. Wang S, Wang L, Xu Y, et al. An effective estimation of distribution algorithm for the flexible job-shop scheduling problem with fuzzy processing time. International Journal of Production Research, 2013, 51(12): 3778–3793
23. Lei D. A genetic algorithm for flexible job shop scheduling with fuzzy processing time. International Journal of Production Research, 2010, 48(10): 2995–3013
24. Gao K Z, Suganthan P N, Pan Q K, et al. An improved artificial bee colony algorithm for flexible job-shop scheduling problem with fuzzy processing time. Expert Systems with Applications, 2016, 65: 52–67
25. Geng Z, Zou Y. Study on job shop fuzzy scheduling problem based on genetic algorithm. Computer Integrated Manufacturing Systems, 2002, 8: 616–620 (in Chinese)
26. Sakawa M, Mori T. Efficient genetic algorithm for job-shop scheduling problems with fuzzy processing time and fuzzy duedate. Computers & Industrial Engineering, 1999, 36(2): 325–341
27. Lei D. Co-evolutionary genetic algorithm for fuzzy flexible job shop scheduling. Applied Soft Computing, 2012, 12(8): 2237–2245
28. Wang X, Chan H K. An integrated fuzzy approach for evaluating remanufacturing alternatives of a product design. Journal of Remanufacturing, 2013, 3(1): 10
29. Lin F T. Fuzzy job-shop scheduling based on ranking level (λ, 1) interval-valued fuzzy numbers. IEEE Transactions on Fuzzy Systems, 2002, 10(4): 510–522
30. Balin S. Parallel machine scheduling with fuzzy processing times using a robust genetic algorithm and simulation. Information Sciences, 2011, 181(17): 3551–3569
31. Zhu Q, Sarkis J, Lai K. Supply chain-based barriers for truck-engine remanufacturing in China. Transportation Research Part E: Logistics and Transportation Review, 2015, 74: 94–108
32. Wang X, Cao L. Genetic Algorithm and Implementation of Software Applications. Xi'an: Xi'an Jiaotong University Press, 2002, 25–39 (in Chinese)
33. Srinivas M, Patnaik L M. Adaptive probabilities of crossover and mutation in genetic algorithms. IEEE Transactions on Systems, Man, and Cybernetics, 1994, 24(4): 656–667
34. Farhy L S, Straume M, Johnson M L, et al. A construct of interactive feedback control of the GH axis in the male. American Journal of Physiology: Regulatory, Integrative and Comparative Physiology, 2001, 281(1): R38–R51
35. Wei L, Li X, Li Y, et al. Optimization of tandem cold rolling schedule based on improved adaptive genetic algorithm. Journal of Mechanical Engineering, 2010, 46(16): 136–141 (in Chinese)
36. Gonçalves J F, Resende M G C. Random-key genetic algorithms. In: Martí R, Panos P, Resende M, eds. Handbook of Heuristics. Cham: Springer, 2016, 1–13
37. Li M. Research on remanufactured flexible job shop scheduling based on damaged parts. Thesis for the Master's Degree. Dalian: Dalian University of Technology, 2019, 29–30 (in Chinese)
38. Peng S, Li T, Zhao J, et al. Petri net-based scheduling strategy and energy modeling for the cylinder block remanufacturing under uncertainty. Robotics and Computer-integrated Manufacturing, 2019, 58: 208–219
39. Gong X, De Pessemier T, Joseph W, et al. A generic method for energy-efficient and energy-cost-effective production at the unit process level. Journal of Cleaner Production, 2016, 113: 508–522
40. Liu C, Dang F, Li W, et al. Production planning of multi-stage multi-option seru production systems with sustainable measures. Journal of Cleaner Production, 2015, 105: 285–299
41. Peng S, Li T, Tang Z, et al. Comparative life cycle assessment of remanufacturing cleaning technologies. Journal of Cleaner Production, 2016, 137: 475–489