IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received January 19, 2019, accepted January 22, 2019, date of publication January 25, 2019, date of current version March 20, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2895348

An Imperialist Competitive Algorithm With the Diversified Operators for Many-Objective Scheduling in Flexible Job Shop

MING LI, DEMING LEI, AND HEJIN XIONG
School of Automation, Wuhan University of Technology, Wuhan 430070, China

Corresponding author: Hejin Xiong (999wuhan@sohu.com)

This work was supported by the National Natural Science Foundation of China 61573264.

ABSTRACT A number of related works have been done on multi-objective flexible job shop scheduling problem, however, many-objective flexible job shop scheduling problems (MaOFJSP) with at least four objectives are seldom considered. In this paper, an imperialist competitive algorithm with the diversified operators (DOICA) is proposed for MaOFJSP with the minimization of makespan, total tardiness, total workload, and total energy consumption. In DOICA, the diversified assimilation and revolution are used according to the features of empires, a novel imperialist competition is implemented by excluding the strongest empire and doing multiple neighborhood search of a solution in the strongest empire. The extensive experiments are conducted by using a number of benchmark instances to test the impact of strategies of DOICA on its performance and compare DOICA with other algorithms from literature finally. The computational results validate that the new strategies of DOICA are effective and DOICA can provide promising results for the considered MaOFJSP.

INDEX TERMS Many-objective optimization, flexible job shop scheduling problem, imperialist competitive algorithm, diversified operators.

I. INTRODUCTION

Scheduling is the main part of manufacturing systems and has been extensively considered in the past decades [1]–[16]. Generally, scheduling problem can be divided into two types: single objective scheduling and multi-objective scheduling. Because manufacturers are often concerned about more than one objective, multi-objective scheduling can meet practical needs better than single objective one and has been investigated widely by using a number of meta-heuristics [17]–[54].

Flexible job shop scheduling problem (FJSP) is a well-known scheduling one and has been extensively applied in many industries such as automobile assembly, textile, chemical material process and semiconductor manufacturing etc [19], [20]. In the past decade, MOFJSP has been extensively considered by various meta-heuristics. These algorithms include genetic algorithm (GA) [21]–[28], particle swarm optimization (PSO) [20], [29]–[31], harmony search algorithm [32], artificial bee colony [33], [34], discrete virus optimization algorithm [35], tabu search [36], [37], variable neighborhood search (VNS [38]), shuffled frog leaping algorithm (SFLA [39]) and estimation of distribution algorithm [40].

In recent years, as a special MOFJSP, energy-efficient FJSP also has attracted some attention. Jiang *et al.* [41] proposed a blood-variation-based non-dominated sorting genetic algorithm-II (NSGA-II [42]). He *et al.* [43] introduced a nested partitions algorithm. Yin *et al.* [44] provided a multi-objective genetic algorithm (MOGA) for FJSP with objectives of productivity, energy efficiency and noise reduction. Lei *et al.* [45] applied a novel SFLA for FJSP with total energy consumption and workload balance. Piroozfard *et al.* [46] presented a MOGA for FJSP with the objectives of minimizing total carbon footprint and total late work criterion. A hybrid algorithm based on evolutionary algorithm and simulated annealing is studied by Mokhtari and Hasani [47] to minimize total completion time and total energy cost and maximize the total availability of the system. Liu *et al.* [48] proposed a hybrid fruit fly optimization algorithm to minimize carbon footprint and makespan.

The associate editor coordinating the review of this manuscript and approving it for publication was Zhiwu Li.

Lei *et al.* [49] presented a two-phase meta-heuristic based on imperialist competitive algorithm (ICA) and VNS to solve MOFJSP with total energy consumption threshold. Wu and Sun [50] applied a non-dominated GA for FJSP with energy-saving measures. Nouiri *et al.* [51] considered energy-efficient FJSP with machine breakdown and gave a predictive reactive method based on PSO. Wang *et al.* [52] developed a hybrid GA and Jiang and Deng [53] designed a discrete cat swarm optimization algorithm. Gong *et al.* [54] considered many-objective flexible job shop scheduling problem (MaOFJSP) with five objectives under dynamic electricity pricing and applied non-dominated genetic algorithm-III to solve it.

As described above, two or three objectives such as makespan, total tardiness and total energy consumption are often optimized in the previous works. However, MaOFJSP is seldom investigated [54], in which at least four objectives are considered. MaOFJSP extensively exists in the real-life manufacturing systems. Makespan, total tardiness and total workload are often dealt with simultaneously in the literature on MOFJSP [19], [22], [23], [33] and the problem is converted into MaOFJSP with the inclusion of total energy consumption; moreover, the relations among objectives become more complicated and time complexity of algorithm for MaOFJSP will greatly increase with the consideration of four or more than four objectives. These two reasons motivate us to handle MaOFJSP and find an effective path to solve it.

ICA [55] is a new meta-heuristic inspired by the sociopolitical behaviors. The main steps of ICA include initial empires, assimilation, revolution and imperialist competition. Unlike other optimization algorithms such as GA and PSO, ICA has good neighborhood search ability, effective global search property and good convergence rate [56]. It also has minor chance of trapping into local optima and flexible structure. In recent years, ICA is used in many multi-objective production scheduling [49], [57]–[59]. It exhibited excellent characteristics in terms of convergence rate and global search property in these applications, thus, ICA has great potential and advantages to solve MaOFJSP.

The contributions of this study can be summarized as follows: (1) MaOFJSP is considered and a new algorithm named DOICA is proposed to minimize makespan, total tardiness, total workload and total energy consumption simultaneously. (2) To obtain high quality solutions, the diversified assimilation and revolution are executed in different empires. (3) A novel imperialist competition is implemented by the elimination of the strongest empire and multiple neighborhood search is done in the strongest empire.

The remainder of the paper is organized as follows. MaOFJSP is formulated in Section II. The detailed steps of DOICA for MaOFJSP are shown in Section III. The computational experiments are depicted in Section IV and a real-life example is considered in Section V. The conclusions are concluded in the last Section. We also discuss the future research topics in the final Section.

II. PROBLEM DESCRIPTION

MaOFJSP is composed of a set of jobs $J = \{J_1, J_2, \cdots, J_n\}$ and a set of machines $M = \{M_1, M_2, \cdots, M_m\}$. Job J_i consists of h_i operations. Operation o_{ij} is the jth operation of job J_i and can be processed on a set of S_{ij} of compatible machines, $S_{ij} \subset M$. Each machine M_k exists two modes: processing mode and stand-by mode. E_k is the energy consumption per unit time in processing mode and SE_k is the energy consumption per unit idle time.

There are several constraints on jobs and machines, such as,

Each machine can process at most one operation at a time,

No jobs may be processed on more than one machine at a time,

Operations cannot be interrupted,

Setup times and remove times are included in the processing times etc.

The problem has two sub-problems: machine assignment sub-problem for assigning an appropriate machine for each operation and scheduling one for sequencing the operations on machines.

The goal of the problem is to minimize simultaneously the following four objectives under the condition that constraints are all met.

$$f_1 = \sum\nolimits_{k=1}^{m} \int_{0}^{C_{\max}} \left(\sum\nolimits_{i=1}^{n} \sum\nolimits_{j=1}^{h_i} E_k y_{ijk}(t) + SE_k z_k(t) \right) dt \tag{1}$$

$$f_2 = \sum\nolimits_{i=1}^{n} \max\{C_i - D_i, 0\} \tag{2}$$

$$f_3 = \sum\nolimits_{k=1}^{m} W_k \tag{3}$$

$$f_4 = C_{\max} \tag{4}$$

where $y_{ijk}(t)$ is a binary variable. If machine $M_k \in S_{ij}$ is in processing mode at time t, then $y_{ijk}(t)$ is equal to 1; otherwise $y_{ijk}(t)$ is 0. $z_k(t)$ is 1 if M_k is in stand-by mode at time t and 0 otherwise.C_i,$D_i$indicate the completion time and duedate of job J_i. $C_{\max}$ is the maximum completion time of all jobs. W_k is the workload of machine M_k. f_1 is total energy consumption, which is composed of the energy consumption in processing mode and stand-by mode of machines, f_2 indicates total tardiness, f_3 denotes total workload and f_4 is makespan.

Objectives f_2, f_3, f_4 are often used in many papers on MOFJSP [19], [33].

Several concepts are listed below for the considered MaOFJSP. $x \succ y$ if $f_i(x) \leq f_i(y)$ for $\forall i \in \{1, 2, 3, 4\}$ and $f_i(x) < f_i(y) \exists i \in \{1, 2, 3, 4\}$, where $x \succ y$ means that x dominates y or y is dominated by x. For a set Φ and a solution $x \in \Phi$, if x is not dominated by any other solutions in Φ, then x is a non-dominated one regarding the set.

III. DOICA FOR MaOFJSP

ICA is a population-based meta-heuristic. Each individual of population represents a country and some best countries are selected as imperialists in the initialization. In general, ICA consists of initial empires construction, assimilation, revolution and imperialist competition. In the existing ICA [49], [60]–[62], assimilation and revolution are often done in the same way for each empire and seldom executed according to the features of empires; moreover, all empires are the participant of imperialist competition and some empires such as the strongest empire are seldom excluded from competition. Based on the above analyses, DOICA is proposed to solve MaOFJSP.

A. INITIAL POPULATION AND INITIAL EMPIRES

MaOFJSP is composed of scheduling sub-problem and machine assignment sub-problem. The solution of the problem is represented as a scheduling string and a machine assignment string [45], [49].

Initial population P with N solutions is randomly produced.

To construct initial empires, the cost c_i of each solution i is often defined. The smaller the cost is, the better the solution is. In this study, we directly calculate the normalized cost $\bar{c}_i$ by

$$\bar{c}_i = \max_{l=1,2,\cdots,N}\{rank_l\} - rank_i + dist_i \Big/ \sum\nolimits_{j\in\Theta_{rank_i}} dist_j \quad (5)$$

where Θ_{rank_i} is the set of solutions with rank value of $rank_i$, $rank_i$ is obtained by non-dominated sorting [42], $dist_i$ is defined by Lei *et al.* [49].

Then N_{im} solutions with biggest normalized cost are chosen as imperialists and the remained ones are colonies. For each imperialist k, the power $F_k = \bar{c}_k \big/ \sum_{l\in Z_{im}} \bar{c}_l$ and the number $NC_k = round\,(F_k \times N_{col})$ of colonies are computed sequentially and NC_k colonies are finally randomly allocated to imperialist $k = 1, 2, \cdots, N_{im}$. where Z_{im} is the set of all imperialists, $N_{col} = N - N_{im}$ indicates the number of colonies and $round(x)$ is a function that gives the nearest integer of x.

Unlike the existing ICA [49], [58]–[60], DOICA directly uses the normalized cost and the normalized total cost.

$$\overline{TC}_k = \bar{c}_k + \xi \sum\nolimits_{\lambda\in Q_k} \bar{c}_\lambda \big/ NC_k \quad k = 1, 2, \cdots, N_{im} \quad (6)$$

where $\overline{TC}_k$ represents the normalized total cost of empire k, Q_k is the set of colonies possessed by imperialist k and ξ is real number.

With respect to ξ, it is generally set to be 0.1 in many works [49], [55], [60], [62], we set $\xi = 0.1$ based on many experiments.

After initial empires are built, the normalized total cost $\overline{TC}_k$ is determined using Equation (6), the strongest empire with the biggest $\overline{TC}_k$ and the weakest empire with smallest $\overline{TC}_k$ are determined, respectively. Suppose that empire 1 is the strongest one, empire N_{im} is the weakest one and other empires are ones $2, \cdots, N_{im} - 1$.

B. THE DIVERSIFIED OPERATORS

Assimilation and revolution are the main operators for producing new solutions. Generally, these two operators are implemented in the same way for each empire. In this study, the diversified operators are applied, that is, the assimilation and revolution are executed differently in different empires.

Algorithm 1 The Diversified Operators

```
for k = 1 to N_im do
    if k = 1 then
        if t < T then
            all colonies in empire 1 are chosen for assimilation
        else
            randomly choose ⌊ρ × NC_1⌋ colonies
        end if
        for each chosen colony in empire 1 do
            assimilation is done between it and its imperialist
        end for
        Randomly choose colonies from empire 1 in terms of revolution probability U_R and then revolution is executed for each chosen colony.
    else
        if k ≤ N_im − 1 then
            each colony of empire k moves toward its imperialist
        else
            for each colony λ in empire N_im do
                if a random number s < α_2 then
                    it learns from a randomly selected imperialist from empires 2, ··· , N_im;
                else
                    colony moves toward the imperialist of empire 1.
                end if
            end for
        end if
        Revolution is implemented for each chosen colony in empire k according to revolution probability U_0.
    end if
end for
```

The two diversified operators of DOICA are described in Algorithm 1. where t indicates the number of loops, revolution probability U_R is defined by

$$U_R = \begin{cases} U_0 \times \cos(1 - \sqrt{t/T}) & t \le T \\ U_0 & t > T \end{cases} \quad (7)$$

where T is integer and $\lfloor x \rfloor$ gives the floor of x.

With respect to U_0, it is set to be 0.35 and 0.3 by Afruzi *et al.* [63] and Bayareh and Mohammadi [64], respectively. In this study, U_0 of 0.3 is chosen based on experiments. Since most colonies should perform assimilation, ρ of 0.7 is determined based on experiments.

Assimilation between colony and its learning object such as its imperialist is conducted as shown in [49]. α_1 is a probability of the first crossover. Generally, scheduling subproblem is more complicated than machine assignment one, so we set $\alpha_1 > 0.5$ to allocate more computing resources to scheduling subproblem. In this study, α_1 is determined to be 0.7 based on experiments.

Revolution for each colony is implemented by multiple neighborhood search [49] with parameter R, where R is the total number of neighborhood search.

Once a new solution z is generated by assimilation or revolution on colony λ, if z is not dominated by λ, then adjust archive Ω using z and replace λ with solution z or update the related parameters of multiple neighborhood search [49].

The initial archive Ω consists of the non-dominated solutions of initial population P. When a new solution z is applied to update Ω, it is first included into Ω, all solutions in Ω are compared according to Pareto dominance and the dominated ones are removed.

For assimilation and revolution in empire $k \geq 2$, when a new solution z is obtained and compared with colony λ, the condition in empire 1 is used to decide if the colony λ is replaced, archive Ω is updated by using z etc.

In this study, assimilation and revolution are implemented in diversified methods. For the strongest empire 1, not all colonies are related to assimilation after T loops and an adaptive revolution is applied. In empire N_{im}, each colony may learn from many other imperialists because the imperialist of empire N_{im} is often worse than other imperialists. The diversified operators make DOICA have many paths to generate new solutions, as a result, high diversity of population is kept and exploration ability is intensified greatly.

C. IMPERIALIST COMPETITION

Imperialist competition is an important step based on the total power of an empire. In general, when the winning empire is decided, the weakest colony of the weakest empire is directly added into the winning empire. In this study, a new imperialist competition is conducted among all empires except empire 1; moreover, the weakest colony of empire N_{im} is not always added into the winning empire.

Algorithm 2 Imperialist Competition for $N_{im} > 2$

Calculate $\overline{TC}_k$ and POW_k for $k = 2, 3, \cdots, N_{im}$.
Produce a vector $[POW_2 - r_2, \cdots, POW_{N_{im}} - r_{N_{im}}]$.
Choose an empire g with biggest $POW_g - r_g$ and decide the weakest colony λ_{wor} of the weakest empire.
Randomly select a solution x of empire 1.
Apply multiple neighborhood search on solution x.
if The best solution of multiple neighborhood search dominates λ_{wor} **then**
  Delete λ_{wor} and add the best solution into empire g
else
  Include λ_{wor} into empire g.
end if

Imperialist competition is described in Algorithm 2. where POW_k represents the power of empire k,

$$POW_k = \left| \overline{TC}_k \Big/ \sum_{l \in Z_{im}} \overline{TC}_l \right| \tag{8}$$

where r_k is a random number following uniform distribution on [0, 1].

When only empires 1 and 2 exist, no competition is done between them.

The elimination of empire from imperialist competition can avoid effectively the premature and make empires compete fully.

Algorithm 3 Search Procedure of DOICA

Randomly produce an initial population P, construct initial archive, $t = 1$.
while Termination condition is not met **do**
  for $k = 1$ to N_{im} **do**
    if $k = 1$ **then**
      Execute the diversified operators in Algorithm 1.
    else
      Perform the diversified operators in Algorithm 1.
    end if
    Exchange colony and its imperialist in empire k if possible.
  end for
  Sort all empires in the descending order of $\overline{TC}_k$.
  Conduct imperialist competition
  Eliminate the empire without any countries.
  $t = t + 1$.
end while

D. ALGORITHM DESCRIPTION

The detailed steps of DOICA are shown in Algorithm 3. where t is the number of loops and used in the search of empire 1, the stopping condition is max_it, which indicates the maximum number of objective function evaluations.

Figure 1 gives the flow chart of DOICA.

Exchange is as follows. For each colony in empire, if it dominates or is non-dominated with its imperialist, it becomes the new imperialist and the old imperialist turns into colony.

After all empires are sorted according to $\overline{TC}_k$, we still make empires 1 and N_{im} be the strongest one and weakest one, respectively and the remained empires are labelled as $2, \cdots, N_{im} - 1$.

In DOICA, empires compete each other in a new method, the strongest empire is excluded from imperialist competition to prevent premature; moreover, assimilation and revolution have different ways to produce new solutions in different empires, as a result, high diversity of population is kept and it is effective to avoid falling into local optima, thus, DOICA has some different features from the existing ICA [49], [60], [61].

IV. COMPUTATIONAL EXPERIMENTS

Extensive experiments are conducted on a set of problems to test the performance of DOICA for MaOFJSP. All experiments are implemented by using Microsoft Visual C++ 2015 and run on 4.0G RAM 2.00GHz CPU PC.

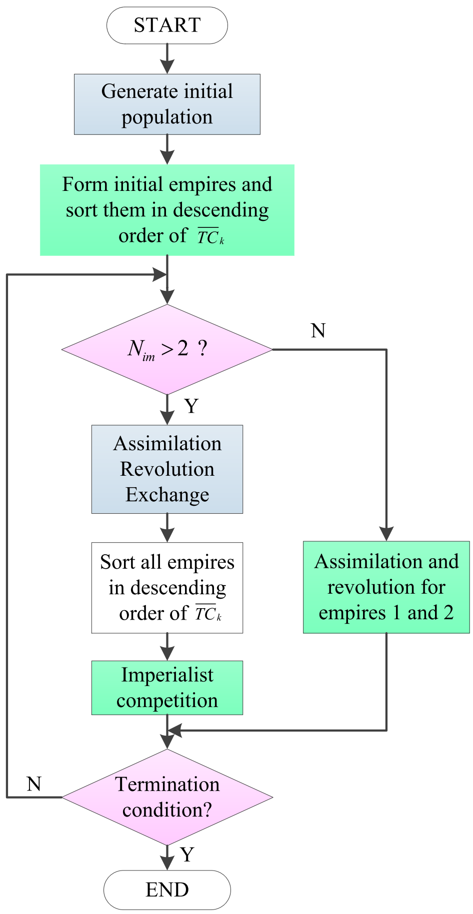

FIGURE 1. Flow chart of DOICA.

A. TEST INSTANCES, METRICS AND COMPARATIVE ALGORITHMS

33 extended instances MK01-15 [65] and DP1-18 [66] are used and described in [49]. For MK01-15, $n \in [10, 30]$, $m \in [4, 15]$; $n \in [10, 20]$ and $m \in [5, 10]$ for DP1-18. Two metrics are applied to measure the results of algorithms.

Metric DI_R [67] is often used to measure the convergence performance by computing the distance of the non-dominated set Ω_l relative to a reference set Ω^*.

$$DI_R(\Omega_l) = \frac{1}{|\Omega^*|} \sum_{y \in \Omega^*} \min \left\{ \sigma_{xy} \,|x \in \Omega_l \right\} \tag{9}$$

where σ_{xy} is the distance between a solution x and a reference solution y in the normalized objective space. The reference set Ω^* is composed of the non-dominated solutions in $\bigcup_l \Omega_l$.

Metric $\mathcal{C}$ [68] is applied to compare the approximate Pareto optimal set respectively obtained by algorithms. $\mathcal{C}(L, B)$ measures the fraction of members of B that are dominated by members of L.

$$\mathcal{C}(L, B) = \frac{|\{b \in B : \exists h \in L, h \succ b\}|}{|B|} \tag{10}$$

Very few papers are about MaOFJSP. Gong *et al*. [54] proposed a GA; however, this GA cannot be directly applied to solve our MaOFJSP. In this study, VNS [38] and MOGA [46] are chosen as comparative algorithm. Piroozfard *et al*. [46] proposed a MOGA for bi-objective energy-efficient FJSP. MOGA is composed of crossover and mutation for two sub-problems of FJSP, tournament selection, local search and population updating. The computational results show that MOGA performs better than NSGA-II; moreover, MOGA can be applied to solve MaOFJSP without any revisions. These two reasons make us choose MOGA as comparative algorithm.

TABLE 1. Computational results of DOICA and its variants on metric DI_R.

Instance	DOICA	DOICA1	DOICA2	Instance	DOICA	DOICA1	DOICA2
MK01	**1.832**	4.831	2.613	DP3	**1.368**	53.39	14.16
MK02	4.118	7.836	3.342	DP4	8.212	11.24	6.735
MK03	**1.282**	11.25	7.790	DP5	**2.034**	16.36	6.964
MK04	**2.750**	11.99	3.689	DP6	5.931	14.51	4.448
MK05	**3.456**	9.376	7.102	DP7	**0.644**	14.57	11.46
MK06	**3.063**	9.057	7.131	DP8	**2.555**	47.12	13.92
MK07	7.850	6.817	6.262	DP9	16.17	30.11	1.144
MK08	**4.378**	9.048	5.425	DP10	**4.362**	11.18	6.494
MK09	**0.135**	17.77	14.34	DP11	**2.122**	11.48	12.35
MK10	**4.243**	12.21	9.620	DP12	**0.650**	15.23	8.652
MK11	**1.965**	7.813	7.253	DP13	7.642	54.05	2.769
MK12	**2.674**	8.047	4.686	DP14	7.632	69.89	1.978
MK13	**0.000**	18.16	17.22	DP15	**0.753**	33.69	9.076
MK14	**3.544**	6.775	4.926	DP16	**2.599**	7.788	5.331
MK15	**0.667**	15.93	16.32	DP17	**0.454**	14.68	8.795
DP1	**2.614**	14.88	12.14	DP18	**9.055**	41.09	25.38
DP2	33.38	36.85	0.000				

Bagheri and Zandieh [38] introduced a VNS for MOFJSP. This algorithm can be applied to solve MaOFJSP after external archive and its updating strategies and the condition deciding if the current solution can be replaced with the new one in DOICA are adopted. Because VNS is the effective method for FJSP and the above revision of VNS is simple, we choose VNS as another comparative algorithm.

We conducted experiments on the main parameters of DOICA and found that DOICA has the best performance when the following settings $N = 80$, $N_{im} = 9$, $max\_it = 1.1 \times 10^5$, $T = 150$ and $R = 10$ are used.

The parameters of MOGA are decided based on experiments: population size of 100, crossover probability of 0.7, mutation probability of 0.4 and maximum generation of 1100.

For VNS, $max\_it = 110000$, other parameter is directly adopted from [38].

B. DISCUSSIONS ON NEW STRATEGIES OF DOICA

Two variants of DOICA, which are called DOICA1 and DOICA2, are built to test the two strategies of DOICA. In DOICA1, imperialist competition is implemented as done in the general ICA, that is, all empires compete each other and the weakest colony of the weakest empire is directly included into the winning empire. In another variant, assimilation and revolution are executed in a unique way, that is, colony just moves toward its imperialist and revolution is done in a fixed probability U_0. The usage of DOICA1 is to show the impact of the diversified operators on performance of DOICA.

Each algorithm randomly runs 20 times for each instance. Tables 1 and 2 show the comparative results of DOICA and

TABLE 2. Computational results of DOICA and its variants on metric $\mathcal{C}$.

Instance	$\mathcal{C}(D, D1)$	$\mathcal{C}(D1, D)$	$\mathcal{C}(D, D2)$	$\mathcal{C}(D2, D)$	$\mathcal{C}(D2, D1)$	$\mathcal{C}(D1, D2)$
MK01	0.441	0.126	0.312	0.221	0.456	0.151
MK02	0.563	0.125	0.500	0.125	0.313	0.400
MK03	0.789	0.194	0.815	0.113	0.658	0.352
MK04	0.784	0.246	0.490	0.295	0.804	0.020
MK05	0.908	0.052	0.830	0.143	0.486	0.274
MK06	0.714	0.344	0.564	0.590	0.778	0.154
MK07	0.312	0.612	0.289	0.777	0.527	0.444
MK08	0.791	0.134	0.524	0.493	0.806	0.063
MK09	1.000	0.000	0.978	0.039	0.821	0.089
MK10	0.750	0.276	0.714	0.172	0.075	0.514
MK11	0.802	0.040	0.807	0.082	0.609	0.173
MK12	0.831	0.062	0.421	0.410	0.759	0.021
MK13	1.000	0.000	1.000	0.000	0.763	0.124
MK14	0.647	0.268	0.343	0.480	0.569	0.167
MK15	1.000	0.000	0.953	0.000	0.352	0.400
DP1	0.857	0.000	0.364	0.059	0.286	0.364
DP2	0.667	0.000	0.000	1.000	1.000	0.000
DP3	1.000	0.000	0.957	0.000	1.000	0.000
DP4	0.511	0.152	0.055	0.758	0.807	0.018
DP5	1.000	0.000	0.839	0.128	0.667	0.339
DP6	1.000	0.000	0.029	0.818	1.000	0.000
DP7	1.000	0.000	0.625	0.059	0.600	0.250
DP8	1.000	0.000	0.500	0.105	1.000	0.000
DP9	1.000	0.000	0.238	0.778	1.000	0.000
DP10	0.873	0.000	0.779	0.000	0.667	0.116
DP11	0.784	0.064	0.830	0.064	0.135	0.528
DP12	1.000	0.000	0.911	0.071	1.000	0.000
DP13	1.000	0.000	0.667	0.429	1.000	0.000
DP14	1.000	0.000	0.000	0.333	1.000	0.000
DP15	1.000	0.000	0.842	0.000	1.000	0.000
DP16	0.692	0.182	0.642	0.076	0.442	0.189
DP17	1.000	0.000	0.929	0.000	0.897	0.000
DP18	1.000	0.000	0.786	0.000	1.000	0.000

its two variants, where the reference set Ω^* is composed of the non-dominated solutions of $\Omega_1 \cup \Omega_2 \cup \Omega_3$ and Ω_1, Ω_2 and Ω_3 are the archive of DOICA, DOICA1 and DOICA2, respectively, symbol "D" indicates DOICA, "D1" denotes DOICA1 and "D2" represents DOICA2.

As stated in Tables 1 and 2, DOICA is superior to DOICA1 and DOICA2 on two metrics. DI_R of DOICA is less than that of DOICA1 on 32 of 33 instances and smaller than that of DOICA1 by at least 10 on 15 instances; moreover, all non-dominated solutions of DOICA1 are dominated by those of DOICA on 15 instances. DI_R of DOICA is better than that of DOICA2 on 25 of 33 instances, $\mathcal{C}(D2, D)$ is less than or equal to $\mathcal{C}(D, D2)$ on 25 instances. The notable performance differences among DOICA and its two variants show that the diversified operators and the novel imperialist competition really have positive impact on the performances of DOICA.

C. RESULTS AND ANALYSES

Table 3 gives the ratio of the number of non-dominated solution to N for 10 instances, in which MIN and MAX indicate the smallest ratio and biggest ratio and AVG is the average ratio in the population of DOICA. It can be found that MAX doesn't exceed 0.2 and MIN is often less than 0.1. Deb *et al.* [69] described computation results for DTLZ2.

TABLE 3. Ratio the number of non-dominated solution to N.

Instance	MIN	AVG	MAX	Instance	MIN	AVG	MAX
MK02	0.063	0.105	0.175	DP1	0.038	0.110	0.200
MK06	0.075	0.114	0.175	DP5	0.050	0.107	0.188
MK10	0.088	0.116	0.150	DP10	0.075	0.128	0.200
MK12	0.050	0.133	0.200	DP15	0.038	0.086	0.138
MK14	0.075	0.178	0.250	DP18	0.063	0.131	0.213

The ratio is close to 0.4 for four objectives and close to 0.1 for two objectives, that is, MaOFJSP with four objectives doesn't reveal the notable feature of many-objective optimization problem, so no special strategies for many-objective optimization are adopted in DOICA.

TABLE 4. Computational results of DOICA, MOGA and VNS on metric DI_R.

Instance	DOICA	MOGA	VNS	Instance	DOICA	MOGA	VNS
MK01	**0.000**	15.02	26.28	DP3	**0.000**	52.62	22.07
MK02	**0.000**	6.250	45.73	DP4	21.27	12.17	15.61
MK03	**1.787**	4.130	73.41	DP5	**3.349**	14.68	34.55
MK04	**1.768**	9.548	22.31	DP6	**8.318**	16.03	35.93
MK05	21.37	22.23	17.39	DP7	**0.000**	53.43	9.457
MK06	**0.000**	10.46	58.73	DP8	12.35	65.38	11.20
MK07	3.718	2.342	51.27	DP9	**0.000**	15.61	29.63
MK08	**3.275**	4.731	35.54	DP10	**3.181**	13.97	28.21
MK09	**4.287**	7.111	105.6	DP11	**6.016**	14.00	60.12
MK10	**3.621**	15.89	98.90	DP12	12.72	9.591	69.88
MK11	**15.18**	20.43	33.96	DP13	**0.000**	21.45	52.81
MK12	**4.902**	12.97	49.78	DP14	**7.691**	58.06	22.03
MK13	5.953	2.905	101.0	DP15	**0.000**	45.23	24.04
MK14	**4.691**	15.12	32.33	DP16	**1.023**	14.78	44.08
MK15	**6.651**	8.765	92.73	DP17	**11.04**	18.74	61.54
DP1	**0.000**	38.02	21.00	DP18	**10.17**	15.14	64.45
DP2	11.77	8.425	53.69				

Tables 4, 5 and 6 report the computational results and times of DOICA and its two comparative algorithms, in which "V" stands for VNS and "M" denotes MOGA. Each algorithm runs 20 times for each instance, where the reference set Ω^* is composed of the non-dominated solutions of $\Omega_1 \cup \Omega_4 \cup \Omega_5$ and Ω_4 and Ω_5 indicate the non-dominated set of MOGA and VNS, respectively. Figure 2 gives descriptions on non-dominated solutions obtained by three algorithms, in which 1,2,3,4 correspond to f_1, f_2, f_3, f_4 respectively and the curve of each solution consists of three segments between 1,2; 2,3 and 3,4. It can be observed from Figure 2 that four objectives really conflict each other.

It can be found that DOICA performs better than other two algorithms on most of instances. DI_R of DOICA is superior to that of MOGA on 28 instances and less than that of VNS on 30 instances; moreover, all non-dominated solutions of MOGA are dominated by those of MOGA on 12 instances and all solutions of VNS are dominated by those of DOICA on 23 instances. DOICA is inferior to MOGA and VNS on very limited number of instances on two metrics.

DOICA has two different features: the diversified operators and new imperialist competition. The first makes DOICA have various strategies for producing new solutions

TABLE 5. Computational results of DOICA, MOGA and VNS on metric $\mathcal{C}$.

Instance	$\mathcal{C}(D,M)$	$\mathcal{C}(M,D)$	$\mathcal{C}(D,V)$	$\mathcal{C}(V,D)$	$\mathcal{C}(V,M)$	$\mathcal{C}(M,V)$
MK01	1.000	0.000	1.000	0.000	0.000	0.980
MK02	1.000	0.000	1.000	0.000	0.000	1.000
MK03	0.317	0.194	1.000	0.000	0.000	1.000
MK04	0.700	0.262	1.000	0.000	0.000	0.839
MK05	0.571	0.104	0.078	0.000	0.000	0.078
MK06	1.000	0.000	1.000	0.000	0.000	1.000
MK07	1.000	0.000	1.000	0.000	0.000	1.000
MK08	0.089	0.373	0.884	0.000	0.000	0.907
MK09	0.000	0.649	1.000	0.000	0.000	1.000
MK10	0.045	0.000	1.000	0.000	0.000	1.000
MK11	0.107	0.147	0.021	0.000	0.000	0.099
MK12	0.133	0.081	0.947	0.000	0.000	0.880
MK13	0.000	0.589	1.000	0.000	0.000	1.000
MK14	0.048	0.117	1.000	0.000	0.000	0.431
MK15	0.175	0.028	1.000	0.000	0.000	1.000
DP1	1.000	0.000	1.000	0.000	1.000	0.000
DP2	0.756	0.000	1.000	0.000	1.000	0.000
DP3	1.000	0.000	1.000	0.000	1.000	0.000
DP4	0.000	1.000	0.038	0.465	0.000	0.886
DP5	1.000	0.000	0.885	0.000	0.000	0.314
DP6	0.000	0.104	0.823	0.000	0.000	0.435
DP7	1.000	0.000	1.000	0.000	1.000	0.000
DP8	0.000	0.000	0.000	1.000	1.000	0.000
DP9	1.000	0.000	1.000	0.000	0.667	0.500
DP10	0.400	0.195	0.986	0.000	0.000	0.634
DP11	0.000	0.192	1.000	0.000	0.000	0.992
DP12	0.000	0.476	1.000	0.000	0.000	1.000
DP13	1.000	0.000	1.000	0.000	0.000	1.000
DP14	1.000	0.000	0.920	0.000	1.000	0.000
DP15	1.000	0.000	1.000	0.000	1.000	0.000
DP16	0.167	0.182	1.000	0.000	0.000	0.928
DP17	0.000	0.000	1.000	0.000	0.000	0.980
DP18	0.767	0.000	1.000	0.000	0.000	1.000

TABLE 6. Computational times of DOICA, MOGA and VNS.

Instance	Running time (s)			Instance	Running time (s)		
	DOICA	MOGA	VNS		DOICA	MOGA	VNS
MK01	6.645	6.662	7.097	DP3	13.74	20.97	15.33
MK02	3.910	6.580	5.804	DP4	27.82	21.93	32.26
MK03	13.72	14.22	18.70	DP5	22.57	21.27	33.95
MK04	8.016	9.273	9.595	DP6	25.85	22.21	41.55
MK05	16.22	12.15	25.65	DP7	23.55	33.12	24.76
MK06	12.14	12.12	14.88	DP8	23.03	33.78	30.53
MK07	18.43	9.977	14.29	DP9	21.32	33.92	29.02
MK08	34.72	30.24	46.27	DP10	43.76	33.47	46.58
MK09	21.91	31.70	44.33	DP11	44.96	33.02	69.41
MK10	23.34	22.49	45.14	DP12	44.63	32.70	78.54
MK11	53.81	20.03	45.39	DP13	37.36	51.05	41.28
MK12	38.24	22.15	46.54	DP14	34.44	50.76	41.38
MK13	28.40	22.30	47.05	DP15	35.08	52.52	44.36
MK14	51.36	21.22	50.74	DP16	69.52	50.82	76.11
MK15	41.87	21.46	49.51	DP17	62.37	51.76	96.31
DP1	14.16	20.79	14.40	DP18	63.80	51.39	104.5
DP2	17.10	20.88	16.30				

and good exploration ability. The second can effectively avoid premature. The combination of these two features can keep high diversity of population and avoid falling into local optima, thus, it can be concluded that DOICA is an efficient method for MaOFJSP.

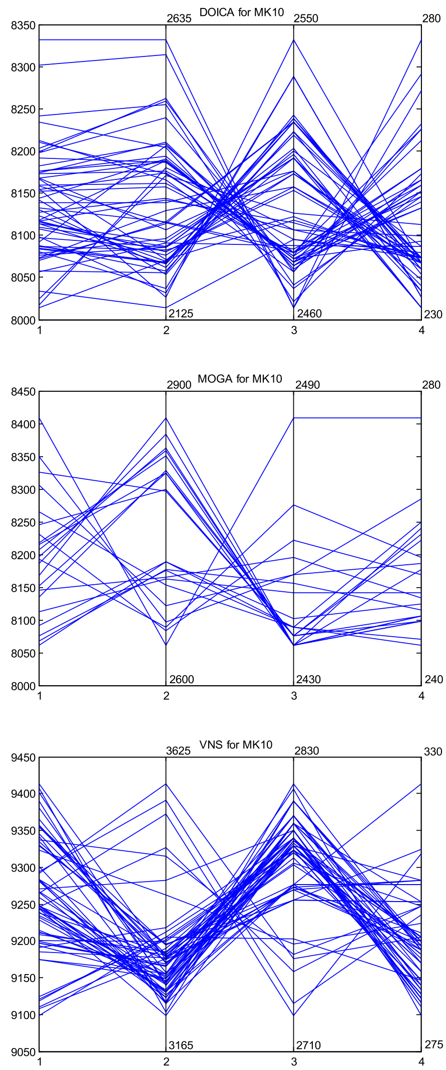

FIGURE 2. Non-dominated solutions of three algorithms.

V. A REAL-LIFE EXAMPLE

The cooling system of automotive engine, as an important system to maintain the normal operation of automobile, can effectively regulate the temperature generated during the operation of engine and ensure the performance of engine. In this study, we choose engine cooling fans produced in a workshop to test efficiency of the proposed algorithm. Engine cooling fan has four jobs which are adapter plate bracket, fan bracket, fan hub and adapter flange. Each job has more than 10 operations. There are two millers, two drilling machines, one machining center, one manual lathe and two CNCs in a workshop. Its production process is a typical flexible job shop problem. The corresponding machining process parameters are given in [44]. The calculation method of energy consumption in [44] is different from our method, we adopt the method of this paper.

Each algorithm also runs 20 times randomly for the real-life example. The non-dominated solutions produced

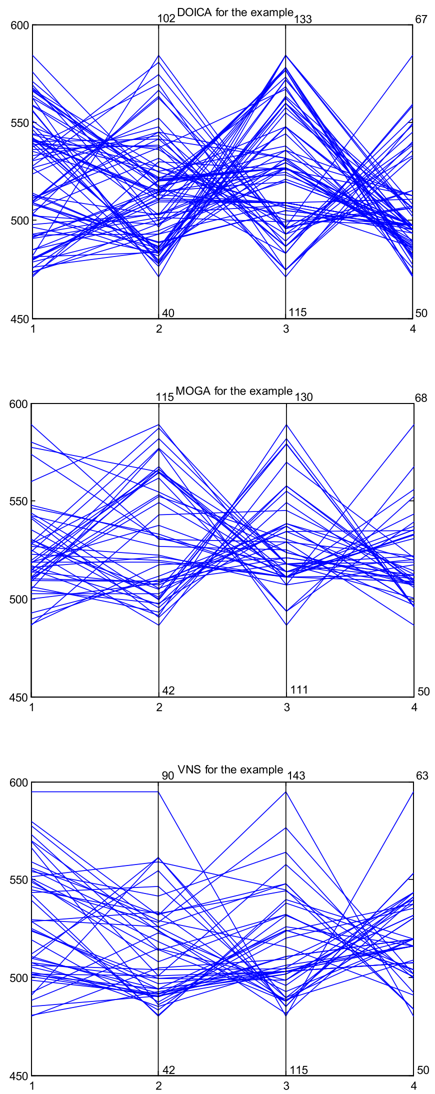

FIGURE 3. Non-dominated solutions of three algorithms of the example.

by DOICA and two comparative algorithms are shown in Figure 3. DI_R of DOICA, MOGA and VNS are 7.474, 13.89 and 16.85, respectively. $\mathcal{C}(D, M)$, $\mathcal{C}(M, D)$ are 0.213 and 0.113; $\mathcal{C}(D, V)$, $\mathcal{C}(V, D)$ are 0.404 and 0.088. Thus, it can be concluded from these results that DOICA performs better than other two algorithms on the real-life example.

VI. CONCLUSIONS

MaOFJSP is addressed and a new algorithm DOICA is proposed to simultaneously minimize makespan, total workload, total tardiness and total energy consumption in this study. DOICA has two new strategies including the diversified operators and a novel imperialist competition. New solutions are generated by using the diversified assimilation and revolution. The strongest empire is excluded from imperialist competition. A number of experiments are conducted and a real-life example is considered. The computational results show that the new strategies of DOICA are effective and DOICA is a very competitive method for MaOFJSP.

The energy-efficient scheduling problem is an important one and exists in the actual manufacturing systems. In the near future, we will continue to focus on this kind of problem and apply some new meta-heuristics such as teaching-learning-based optimization to solve the problem. On the other hand, we have paid attention to distributed scheduling with energy related objective and tried to design a powerful scheduling algorithm, so energy-efficient distributed scheduling is our future topic. We also try to solve other many-objective production scheduling problems, such as many-objective hybrid flow shop scheduling, in the future.

REFERENCES

[1] G. Tian, M. Zhou, and P. Li, "Disassembly sequence planning considering fuzzy component quality and varying operational cost," *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 2, pp. 748–760, Apr. 2018.

[2] Y. Hou, N. Wu, M. Zhou, and Z. Li, "Pareto-optimization for scheduling of crude oil operations in refinery via genetic algorithm," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 3, pp. 517–530, Mar. 2017.

[3] G. D. Tian *et al.*, "Operation patterns analysis of automotive components remanufacturing industry development in China," *J. Cleaner Prod.*, vol. 164, pp. 1363–1375, Oct. 2017.

[4] N. Wu, M. Zhou, and Z. Li, "Short-term scheduling of crude-oil operations: Enhancement of crude-oil operations scheduling using a Petri net-based control-theoretic approach," *IEEE Robot. Autom. Mag.*, vol. 22, no. 2, pp. 64–76, Jun. 2015.

[5] N. Wu, L. Bai, and M. C. Zhou, "An efficient scheduling method for crude oil operations in refinery with crude oil type mixing requirements," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 3, pp. 413–426, Mar. 2016.

[6] H. Zhang, X. Wang, P. Memarmoshrefi, and D. Hogrefe, "A survey of ant colony optimization based routing protocols for mobile Ad hoc Networks," *IEEE Access*, vol. 5, pp. 24139–24161, 2017.

[7] N. Q. Wu and M. C. Zhou, "Schedulability analysis and optimal scheduling of dual-arm cluster tools with residency time constraint and activity time variation," *IEEE Trans. Autom. Sci. Eng.*, vol. 9, no. 1, pp. 203–209, Jan. 2012.

[8] X. Liu, "Routing protocols based on ant colony optimization in wireless sensor networks: A survey," *IEEE Access*, vol. 5, pp. 26303–26317, 2017.

[9] N. Q. Wu and M. C. Zhou, "Modeling, analysis and control of dual-arm cluster tools with residency time constraint and activity time variation based on Petri nets," *IEEE Trans. Autom. Sci. Eng.*, vol. 9, no. 2, pp. 446–454, Apr. 2012.

[10] J. A. P. Pérez, F. Riera-Palou, and G. Femenias, "Multi-objective optimization of coordinated multipoint-aided MIMO-OFDMA systems with frequency reuse," *IEEE Access*, vol. 5, pp. 15448–15467, 2017.

[11] N. Wu, M. Zhou, L. Bai, and Z. Li, "Short-term scheduling of crude oil operations in refinery with high-fusion-point oil and two transportation pipelines," *Enterprise Inf. Syst.*, vol. 10, no. 6, pp. 581–610, May 2016.

[12] L. Bai, N. Wu, Z. Li, and M. Zhou, "Optimal one-wafer cyclic scheduling and buffer space configuration for single-arm multicluster tools with linear topology," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 10, pp. 1456–1467, Oct. 2016.

[13] F. Yang, N. Wu, Y. Qiao, M. Zhou, and Z. Li, "Scheduling of single-arm cluster tools for an atomic layer deposition process with residency time constraints," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 3, pp. 502–516, Mar. 2017.

[14] S. Zhang, N. Wu, Z. Li, T. Qu, and C. Li, "Petri net-based approach to short-term scheduling of crude oil operations with less tank requirement," *Inf. Sci.*, vol. 417, pp. 247–261, Nov. 2017.

[15] N. Wu, Z. Li, and T. Qu, "Energy efficiency optimization in scheduling crude oil operations of refinery based on linear programming," *J. Cleaner Prod.*, vol. 166, pp. 49–57, Nov. 2017.

[16] Q. Zhu, M. Zhou, Y. Qiao, and N. Wu, "Petri net modeling and scheduling of a close-down process for time-constrained single-arm cluster tools," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 48, no. 3, pp. 389–400, Mar. 2018.

[17] J.-Q. Li, H.-Y. Sang, Y.-Y. Han, C.-G. Wang, and K.-Z. Gao, "Efficient multi-objective optimization algorithm for hybrid flow shop scheduling problems with setup energy consumptions," *J. Cleaner Prod.*, vol. 181, no. 20, pp. 584–598, Apr. 2018.

[18] J.-Q. Li and Q.-K. Pan, "Solving the large-scale hybrid flow shop scheduling problem with limited buffers by a hybrid artificial bee colony algorithm," *Inf. Sci.*, vol. 316, no. 20, pp. 487–502, Sep. 2015.

[19] Y. Yuan and H. Xu, "Multiobjective flexible job shop scheduling using memetic algorithms," *IEEE Trans. Autom. Sci. Eng.*, vol. 12, no. 1, pp. 336–353, Jan. 2015.

[20] G. Moslehi and M. Mahnam, "A Pareto approach to multi-objective flexible job-shop scheduling problem using particle swarm optimization and local search," *Int. J. Prod. Econ.*, vol. 129, no. 1, pp. 14–22, Jan. 2011.

[21] F. T. S. Chan, T. C. Wong, and L. Y. Chan, "Flexible job-shop scheduling problem under resource constraints," *Int. J. Prod. Res.*, vol. 44, no. 11, pp. 2071–2089, Feb. 2006.

[22] J. Gao, M. Gen, L. Sun, and X. Zhao, "A hybrid of genetic algorithm and bottleneck shifting for multi-objective flexible job shop scheduling problems," *Comput. Ind. Eng.*, vol. 53, no. 1, pp. 149–162, Aug. 2007.

[23] T.-C. Chiang and H.-J. Lin, "A simple and effective evolutionary algorithm for multiobjective flexible job shop scheduling," *Int. J. Prod. Econ.*, vol. 141, no. 1, pp. 87–98, Jan. 2013.

[24] M. Rohaninejad, A. Kheirkhah, P. Fattahi, and B. Vahedi-Nouri, "A hybrid multi-objective genetic algorithm based on the ELECTRE method for a capacitated flexible job shop scheduling problem," *Int. J. Adv. Manuf. Technol.*, vol. 77, nos. 1–4, pp. 51–66, Mar. 2015.

[25] M. Rohaninejad, R. Sahraeian, and B. V. Nouri, "Multi-objective optimization of integrated lot-sizing and scheduling problem in flexible job shop," *RAIRO-Oper. Res.*, vol. 50, no. 3, pp. 587–609, Oct. 2015.

[26] J. Li, Y. Huang, and X. Niu, "A branch population genetic algorithm for dual-resource constrained job shop scheduling problem," *Comput. Ind. Eng.*, vol. 102, no. 1, pp. 113–131, Dec. 2016.

[27] E. Ahmadi, M. Zandieh, M. Farrokh, and S. M. Emami, "A multi objective optimization approach for flexible job shop scheduling problem under random machine breakdown by evolutionary algorithm," *Comput. Oper. Res.*, vol. 73, no. 1, pp. 56–66, Sep. 2016.

[28] X.-N. Shen, Y. Han, and J.-Z. Fu, "Robustness measures and robust scheduling for multi-objective stochastic flexible job shop scheduling problems," *Soft Comput.*, vol. 21, no. 21, pp. 6531–6554, Nov. 2017.

[29] W. Xia and Z. Wu, "An effective hybrid optimization approach for multi-objective flexible job-shop scheduling problems," *Comput. Ind. Eng.*, vol. 48, no. 2, pp. 409–425, Mar. 2005.

[30] M. R. Singh, M. Singh, S. S. Mahapatra, and N. Jagadev, "Particle swarm optimization algorithm embedded with maximum deviation theory for solving multi-objective flexible job shop scheduling problem," *Int. J. Adv. Manuf. Technol.*, vol. 85, no. 9, pp. 2353–2366, Aug. 2016.

[31] G. Zhang, X. Shao, P. Li, and L. Gao, "An effective hybrid particle swarm optimization algorithm for multi-objective flexible job-shop scheduling problem," *Comput. Ind. Eng.*, vol. 56, no. 4, pp. 1309–1318, May 2009.

[32] K.-Z. Gao, P. N. Suganthan, Q.-K. Pan, T. J. Chua, T. X. Cai, and C.-S. Chong, "Pareto-based grouping discrete harmony search algorithm for multi-objective flexible job shop scheduling," *Inf. Sci.*, vol. 289, pp. 76–90, Dec. 2014.

[33] J.-Q. Li, Q.-K. Pan, and M. F. Tasgetiren, "A discrete artificial bee colony algorithm for the multi-objective flexible job-shop scheduling problem with maintenance activities," *Appl. Math. Model.*, vol. 38, no. 3, pp. 1111–1132, Feb. 2014.

[34] L. Wang, G. Zhou, Y. Xu, and M. Liu, "An enhanced Pareto-based artificial bee colony algorithm for the multi-objective flexible job-shop scheduling," *Int. J. Adv. Manuf. Technol.*, vol. 60, pp. 1111–1123, Jun. 2012.

[35] C. Lu, X. Li, L. Gao, W. Gao, and Y. Jin, "An effective multi-objective discrete virus optimization algorithm for flexible job-shop scheduling problem with controllable processing times," *Comput. Ind. Eng.*, vol. 104, no. 1, pp. 156–174, Feb. 2017.

[36] J.-Q. Li, Q.-K. Pan, P. N. Suganthan, and T. J. Chua, "A hybrid tabu search algorithm with an efficient neighborhood structure for the flexible job shop scheduling problem," *Int. J. Adv. Manuf. Technol.*, vol. 52, nos. 5–8, pp. 683–697, Feb. 2011.

[37] S. Jia and Z. H. Hu, "Path-relinking tabu search for the multi-objective flexible job shop scheduling problem," *Comput. Oper. Res.*, vol. 47, no. 1, pp. 11–26, Jul. 2014.

[38] A. Bagheri and M. Zandieh, "Bi-criteria flexible job-shop scheduling with sequence-dependent setup times—Variable neighborhood search approach," *J. Manuf. Syst.*, vol. 30, no. 1, pp. 8–15, Jan. 2011.

[39] J. Li, Q. Pan, and S. Xie, "An effective shuffled frog-leaping algorithm for multi-objective flexible job shop scheduling problems," *Appl. Math. Comput.*, vol. 218, no. 18, pp. 9371–9395, May 2012.

[40] L. Wang, S. Wang, and M. Liu, "A Pareto-based estimation of distribution algorithm for the multi-objective flexible job-shop scheduling problem," *Int. J. Prod. Res.*, vol. 51, no. 12, pp. 3574–3592, Mar. 2013.

[41] Z. Q. Jiang, L. Zuo, and E. Mingcheng, "Study on multi-objective flexible job-shop scheduling problem considering energy consumption," *J. Ind. Eng. Manage.*, vol. 7, no. 3, pp. 589–604, May 2014.

[42] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[43] Y. He, Y. Li, T. Wu, and J. W. Sutherland, "An energy-responsive optimization method for machine tool selection and operation sequence in flexible machining job shops," *J. Cleaner Prod.*, vol. 87, no. 1, pp. 245–254, Jan. 2015.

[44] L. Yin, X. Li, L. Gao, C. Lu, and Z. Zhang, "A novel mathematical model and multi-objective method for the low-carbon flexible job shop scheduling problem," *Sustain. Comput., Inform. Syst.*, vol. 13, no. 1, pp. 15–30, Mar. 2017.

[45] D. Lei, Y. Zheng, and X. Guo, "A shuffled frog-leaping algorithm for flexible job shop scheduling with the consideration of energy consumption," *Int. J. Prod. Res.*, vol. 55, no. 11, pp. 3126–3140, 2017.

[46] H. Piroozfard, K. Y. Wong, and W. P. Wong, "Minimizing total carbon footprint and total late work criterion in flexible job shop scheduling by using an improved multi-objective genetic algorithm," *Resour., Conservation Recycling*, vol. 128, pp. 267–283, Jan. 2018.

[47] H. Mokhtari and A. Hasani, "An energy-efficient multi-objective optimization for flexible job-shop scheduling problem," *Comput. Chem. Eng.*, vol. 104, pp. 339–352, Sep. 2017.

[48] Q. Liu, M. Zhan, F. O. Chekem, X. Shao, B. Ying, and J. W. Sutherland, "A hybrid fruit fly algorithm for solving flexible job-shop scheduling to reduce manufacturing carbon footprint," *J. Clean. Prod.*, vol. 68, pp. 668–678, Dec. 2017.

[49] D. M. Lei, M. Li, and L. Wang, "A two-phase meta-heuristic for multiobjective flexible job shop scheduling problem with total energy consumption threshold," *IEEE. Trans. Cybern.*, vol. 49, no. 3, pp. 1097–1109, Feb. 2019.

[50] X. Wu and Y. Sun, "A green scheduling algorithm for flexible job shop with energy-saving measures," *J. Clean. Prod.*, vol. 172, pp. 3249–3264, Jan. 2018.

[51] M. Nouiri, A. Bekrar, and D. Trentesaux, "Towards energy efficient scheduling and rescheduling for dynamic flexible job shop problem," *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 1275–1280, Jan. 2018.

[52] H. Wang, Z. Jiang, Y. Wang, H. Zhang, and Y. H. Wang, "A two-stage optimization method for energy-saving flexible job-shop scheduling based on energy dynamic characterization," *J. Clean. Prod.*, vol. 188, pp. 575–588, Jul. 2018.

[53] T. Jiang and G. Deng, "Optimizing the low-carbon flexible job shop scheduling problem considering energy consumption," *IEEE Access*, vol. 6, pp. 46346–46355, 2018.

[54] X. Gong, T. De Pessemier, L. Martens, and W. Joseph, "Energy- and labor-aware flexible job shop scheduling under dynamic electricity pricing: A many-objective optimization investigation," *J. Clean. Prod.*, vol. 209, pp. 1078–1094, Feb. 2019.

[55] E. Atashpaz-Gagari and C. Lucas, "Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition," in *Proc. IEEE Congr. Evol. Comput.*, Singapore, Sep. 2007, pp. 4661–4667.

[56] S. Hosseini and A. A. Khaled, "A survey on the imperialist competitive algorithm metaheuristic: Implementation in engineering domain and directions for future research," *Appl. Soft Comput.*, vol. 24, pp. 1078–1094, Nov. 2014.

[57] P. Zhang, X. Zhao, X. Sheng, and J. Zhang, "An imperialist competitive algorithm incorporating remaining cycle time prediction for photolithography machines scheduling," *IEEE Access*, vol. 6, pp. 66787–66797, 2018.

[58] Z. X. Pan, D. M. Lei, and Q. Y. Zhang, "A new imperialist competitive algorithm for multiobjective low carbon parallel machines scheduling," *Math. Problems Eng.*, vol. 2018, pp. 1–13, Apr. 2018.

[59] H. Piroozfard, K. Y. Wong, and M. K. Tiwari, "Reduction of carbon emission and total late work criterion in job shop scheduling by applying a multi-objective imperialist competitive algorithm," *Int. J. Comput. Intel. Syst.*, vol. 11, no. 1, pp. 805–829, Jan. 2018.

[60] S. M. Goldansaz, F. Jolai, and A. H. Z. Anaraki, "A hybrid imperialist competitive algorithm for minimizing makespan in a multi-processor open shop," *Appl. Math. Model.*, vol. 37, no. 23, pp. 9603–9616, Dec. 2013.

[61] B. Naderi and M. Yazdani, "A model and imperialist competitive algorithm for hybrid flow shops with sublots and setup times," *J. Manuf. Syst.*, vol. 33, no. 4, pp. 647–653, Oct. 2014.

[62] B. Ghasemishabankareh, N. Shahsavari-Pour, M. A. Basiri, and X. D. Li, "A hybrid imperialist competitive algorithm for the flexible job shop problem," in *Proc. Australas. Conf. Artif. Life Comput. Intell.*, vol. 9592. Springer, Jan. 2016, pp. 221–233.

[63] E. N. Afruzi, A. A. Najafi, E. Roghanian, and M. Mazinan, "A multiobjective imperialist competitive algorithm for solving discrete time, cost and quality trade-off problems with mode-identity and resource-constrained situation," *Comput. Oper. Res.*, vol. 50, no. 1, pp. 80–96, Oct. 2014.

[64] M. Bayareh and M. Mohammadi, "Multi-objective optimization of a triple shaft gas compressor station using imperialist competitive algorithm," *Appl. Thermal Eng.*, vol. 109, pp. 384–400, Oct. 2016.

[65] P. Brandimarte, "Routing and scheduling in a flexible job shop by tabu search," *Ann. Oper. Res.*, vol. 41, no. 1, pp. 157–183, Sep. 1993.

[66] S. Dauzère-Pérès and Jan Paulli, "An integrated approach for modeling and solving the general multiprocessor job-shop scheduling problem using tabu search," *Ann. Oper. Res.*, vol. 70, no. 2, pp. 281–306, Apr. 1997.

[67] J. Knowles and D. Corne, "On metrics for comparing nondominated sets," in *Proc. Congr. Evol. Comput.*, Honolulu, HI, USA, May 2002, pp. 711–716.

[68] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach," *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257–271, Nov. 1999.

[69] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable test problems for evolutionary multiobjective optimization," in *Evolutionary Multiobjective Optimization*. Berlin, Germany: Springer-Verlag, 2005.

MING LI received the bachelor's degree in mechanical engineering and automation from the Hefei University of Technology, Hefei, China, in 2014, and the M.S. degree in mechanical engineering and automation from Anhui Polytechnic University, Wuhu, China, in 2017. He is currently pursuing the Ph.D. degree with the School of Automation, Wuhan University of Technology, Wuhan, China. His current research interest includes manufacturing systems intelligent optimization and scheduling.

DEMING LEI received the M.S. degree in applied mathematics from Xi'an Jiaotong University, Xi'an, China, in 1996, and the Ph.D. degree in automation science and engineering from Shanghai Jiao Tong University, Shanghai, China, in 2005. He is currently a Professor with the School of Automation, Wuhan University of Technology, Wuhan, China. He has published over 50 journal papers. His current research interests include intelligent system optimization and control, and production scheduling.

HEJIN XIONG received the Ph.D. degree in control theory and control engineering from the Huazhong University of Science and Technology, in 1999, where he obtained an Excellent Doctoral Dissertation Award. In 2002, he held a Postdoctoral Station with the Wuhan University of Technology. He has served as the Secretary General of international and domestic academic conferences on many occasions. He has been a Reviewer for more than ten core journals. He has participated in the completion of eight national and provincial level projects, two of which were evaluated as excellent by the fund committee; hosting and participating in the completion of enterprise technology development issues. He has published more than 70 papers at home and abroad, and more than 40 articles have been included in SCI and EI. He holds more than 30 authorized patents. As the first author, he edited *Intelligent Information Processing* (first edition, second edition) and *Gray Control*. The research direction is complex system modeling and control, and electrical intelligence systems.