TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 01/24 pp285–304
DOI: 10.26599/TST.2023.9010015
Volume 29, Number 2, April 2024

Mathematical Modeling and a Multiswarm Collaborative Optimization Algorithm for Fuzzy Integrated Process Planning and Scheduling Problem

Qihao Liu, Cuiyu Wang, Xinyu Li*, and Liang Gao

Abstract: Considering both process planning and shop scheduling in manufacturing can fully utilize their complementarities, resulting in improved rationality of process routes and high-quality and efficient production. Hence, the study of Integrated Process Planning and Scheduling (IPPS) has become a hot topic in the current production field. However, when performing this integrated optimization, the uncertainty of processing time is a realistic key point that cannot be neglected. Thus, this paper investigates a Fuzzy IPPS (FIPPS) problem to minimize the maximum fuzzy completion time. Compared with the conventional IPPS problem, FIPPS considers the fuzzy process time in the uncertain production environment, which is more practical and realistic. However, it is difficult to solve the FIPPS problem due to the complicated fuzzy calculating rules. To solve this problem, this paper formulates a novel fuzzy mathematical model based on the process network graph and proposes a MultiSwarm Collaborative Optimization Algorithm (MSCOA) with an integrated encoding method to improve the optimization. Different swarms evolve in various directions and collaborate in a certain number of iterations. Moreover, the critical path searching method is introduced according to the triangular fuzzy number, allowing for the calculation of rules to enhance the local searching ability of MSCOA. The numerical experiments extended from the well-known Kim benchmark are conducted to test the performance of the proposed MSCOA. Compared with other competitive algorithms, the results obtained by MSCOA show significant advantages, thus proving its effectiveness in solving the FIPPS problem.

Key words: Integrated Process Planning and Scheduling (IPPS); fuzzy processing time; fuzzy completion time; MultiSwarm Collaborative Optimization Algorithm (MSCOA)

1 Introduction

Process planning and shop scheduling are two independent subsystems that are carried out sequentially in a conventional manufacturing system[1–3]. However, minimal coordination between the two subsystems may result in process and production fragmentation, which in turn, may lead to the generation of bottleneck resources, conflicting optimization objectives, and unbalanced machine loads[4]. Integrated Process Planning and Scheduling (IPPS) could achieve better process routes than process planning alone; it can also provide more efficient scheduling solutions than shop scheduling alone[5]. Therefore, due to the urgent need to investigate the IPPS problem, this has become a hot research topic in the field of industrial engineering.

Khoshnevis and Chen[6] are the first to propose

• Qihao Liu, Cuiyu Wang, Xinyu Li, and Liang Gao are with State Key Laboratory of Intelligent Manufacturing Equipment and Technology, School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China. E-mail: lllqh@hust.edu.cn; wcyhust@foxmail.com; lixinyu@mail.hust.edu.cn; gaoliang@mail.hust.edu.cn.
∗ To whom correspondence should be addressed.
Manuscript received: 2023-01-29; revised: 2023-02-16; accepted: 2023-03-07

the concept of integrating process planning and shop scheduling by demonstrating a methodology and the potential benefits of the integration. Jain et al.[7] tried to solve the Job-shop Scheduling Problem (JSP) with multiple process routes in a Flexible Manufacturing System (FMS). Chan et al.[8] proposed a genetic algorithm with dominant genes to solve the scheduling problem in FMS with optional process routes. Li et al.[9] proposed a hybrid algorithm for solving the Kim dataset, which refreshes most of the results. Zhang and Wong[10] proposed an enhanced ant colony optimization algorithm to solve the Kim benchmark and refreshed 17 out of 24 problems, including four new solutions reaching the lower bound. Li et al.[11] subsequently developed a GA and variable neighborhood search algorithm and, once again, updated nine solutions of this dataset, of which three reaches the lower bound. Liu et al.[12] first built a new MILP model based on the network graph's OR-nodes, and established four submodels with less flexibility through a flexibility decomposition strategy. They then combined this with the model collaboration framework to achieve the rapid search over a large-scale IPPS problem, thus creating one of the most effective methods for solving the IPPS problem.

In the above studies, the production parameters are treated as deterministic values[13–17]. However, a variety of uncertain factors may lead to the uncertainty of production parameters[18–20]. Specifically, one of the characteristics that cannot be neglected in actual production is the uncertainty of the processing time of the workpiece[21, 22]. Therefore, investigating the shop scheduling problem considering fuzzy or uncertain processing time is of great practical significance[23, 24].

Related to the abovementioned problem, Behnamian[25] presented an extensive review of the fuzzy shop scheduling problem according to shop types, including single machines, parallel machines, flowshop, job shop, and open shop. Among them, fuzzy job shop scheduling and flow shop scheduling problems were studied the most, while parallel machines and open shop fuzzy scheduling were studied the least. Abdullah and Abdolrazzagh-Nezhad[26] conducted a review of the fuzzy JSP, focusing on the meta-heuristics of the solution methods and found that 63% of the approaches for solving fuzzy JSP are GA related. Wang et al.[27] proposed a hybrid artificial bee colony algorithm for solving the fuzzy flexible job-shop scheduling problem. Wang et al.[28] further designed a hybrid adaptive differential evolution algorithm to solve a multiobjective fuzzy JSP.

Gao et al.[29] enhanced the classic differential evolution algorithm through a selection mechanism to solve JSP with fuzzy execution time and fuzzy completion time. Cai and Lei[30] proposed a solution to the distributed fuzzy hybrid flow shop scheduling problems with fuzzy processing time by designing a cooperated shuffled frog-leaping algorithm to optimize the objective of fuzzy makespan, total agreement index, and fuzzy total energy consumption simultaneously. For the FJSP in the type-2 fuzzy logic system, Li et al.[31] proposed an improved artificial immune system algorithm to solve a special case of FJSP with the processing time, which is the nonsymmetric triangular Interval T2FS (IT2FS) value.

Although there are relatively more studies on fuzzy scheduling problems for flow shop and job shop types[32], very few studies have investigated the FIPPS problem[33]. In fact, Wen et al.[34] used the Triangular Fuzzy Number (TFN) to represent uncertain processing time and proposed a multiobjective GA to solve the multiobjective IPPS problem by minimizing makespan, maximal machine workload, total machine workload, and total flow time. Zhang et al.[35] also used the TFN to represent the processing time and the transportation time in the distributed manufacturing environment, after which they proposed an extended GA with a three-class encoding method, improved crossover, and mutation to optimize the FIPPS problems. Wen et al.[36] further improved the encoding based on previous research and proposed a multilayer collaborative optimization method to solve the fuzzy multiobjective IPPS problem while including customer satisfaction as one of the optimizing objectives.

In accordance with the research status on the fuzzy shop scheduling problem presented above, we can conclude that, first, the current mainstream solving approach is still the intelligent optimization algorithm[37–41]. Among them, the swarm intelligence algorithm is one of the most common methods[42, 43]. Second, fewer studies have been conducted on the FIPPS problem, of which modeling research is urgently required. Third, the encoding methods based on the problem characteristics have not yet been significantly improved. Furthermore, most studies fail to effectively increase the integration of process planning and shop scheduling.

Therefore, the current paper establishes a new mathematical model for the FIPPS problem based on the process network graph, and designs a MultiSwarm Collaborative Optimization Algorithm (MSCOA) with a novel integrated encoding approach. The integrated encoding approach can improve the integration degree of process planning and shop scheduling, extend the original solution space, and provide the possibility of finding more candidate solutions with better quality. The collaboration and interaction among multiple swarms enable the algorithm to achieve a global and local search balance.

The remainder of this paper is organized as follows: Section 2 presents the problem description and the mathematical model; Section 3 introduces the proposed MSCOA; Section 4 shows the comparative experiments and discussions; and finally, Section 5 discusses the conclusions and future research directions.

2 Problem Formulation

2.1 Problem description

The FIPPS problem can be defined as follows: N jobs with pending process routes are assigned to M machines for processing with fuzzy processing time. The FIPPS aims to plan proper process routes for each job and to assign reasonable machines to meet specific constraints and optimize the objective, such as maximum fuzzy completion time.

The process information of the jobs is usually represented by network graphs, as shown in Fig. 1. The corresponding values 9 of processing time of the operations are in the form of TFNs, as shown in Table 1. The three numbers in the bracket are the minimum, most probable, and maximum values of the processing time, respectively. Every job of the IPPS has its process network graph from which the process routes are planned. The network graph comprises five types of nodes: (1) the start node, which is virtual and represents the start of a job's production process;

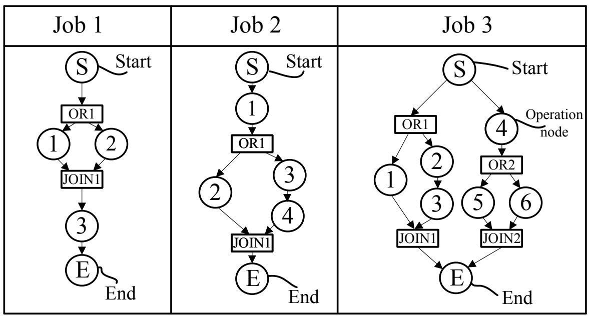

Fig. 1 Network graphs of three jobs.

Table 1 Processing time of the three jobs in Fig. 1.

Job No.	Operation	Alternative machine	Processing time
1	1	1, 2	(1, 2, 4), (2, 3, 5)
	2	2, 4	(1, 1, 2), (1, 2, 2)
	3	2, 3	(2, 4, 5), (2, 3, 5)
2	1	3, 5	(2, 3, 5), (2, 4, 6)
	2	3, 4	(1, 4, 5), (1, 2, 4)
	3	1, 5	(1, 2, 3), (2, 3, 5)
	4	1, 4, 5	(1, 2, 3), (1, 2, 4), (2, 3, 5)
3	1	2, 3	(2, 3, 5), (1, 2, 4)
	2	2, 4	(1, 3, 5), (1, 2, 4)
	3	1, 5	(2, 4, 6), (5, 6, 9)
	4	1, 3	(1, 2, 4), (2, 3, 6)
	5	1, 3, 4	(2, 3, 4), (1, 2, 4), (1, 2, 3)
	6	2, 4	(3, 4, 6), (2, 3, 5)

(2) the end node, which is also virtual and indicates the end of the production process; (3) the intermediate nodes, which imply operations; and (4) the OR-nodes, which are combined with (5) the JOIN-node to represent the process flexibilities. The arrows connecting the operation nodes represent the precedence relationships between operations. For example, the arrow connecting operation nodes 3 and 4 in Job 2's network indicates that Operation 3 of Job 2 should be processed before Operation 4. Furthermore, only one of the two links between a pair of OR-node and JOIN-node would be selected, and the operations inside the selected link will be chosen as well. The time unit in the paper is omitted.

The operating rules on the TFN values of fuzzy processing time, $\tilde{P} = (\tilde{p}_1, \tilde{p}_2, \tilde{p}_3)$ and $\tilde{P}' = (\tilde{p}'_1, \tilde{p}'_2, \tilde{p}'_3)$, are introduced below.

(1) Addition rule:

$\tilde{P} + \tilde{P}' = (\tilde{p}_1 + \tilde{p}'_1, \tilde{p}_2 + \tilde{p}'_2, \tilde{p}_3 + \tilde{p}'_3)$.

(2) Subtraction rule:

$\tilde{P} - \tilde{P}' = (\tilde{p}_1 - \tilde{p}'_1, \tilde{p}_2 - \tilde{p}'_2, \tilde{p}_3 - \tilde{p}'_3)$.

(3) Comparing rule:

Situation 1:

If $\dfrac{\tilde{p}_1 + 2\tilde{p}_2 + \tilde{p}_3}{4} > (<) \dfrac{\tilde{p}'_1 + 2\tilde{p}'_2 + \tilde{p}'_3}{4}$, then $\tilde{P} > (<) \tilde{P}'$.

Situation 2:

If $\dfrac{\tilde{p}_1 + 2\tilde{p}_2 + \tilde{p}_3}{4} = \dfrac{\tilde{p}'_1 + 2\tilde{p}'_2 + \tilde{p}'_3}{4}$, then compare $\tilde{p}_2$ and $\tilde{p}'_2$. If $\tilde{p}_2 > (<) \tilde{p}'_2$, then $\tilde{P} > (<) \tilde{P}'$.

Situation 3:

If $\tilde{p}_2 = \tilde{p}'_2$, then compare $(\tilde{p}_3 - \tilde{p}_1)$ and $(\tilde{p}'_3 - \tilde{p}'_1)$.

If $(\tilde{p}_3 - \tilde{p}_1) > (<)(\tilde{p}'_3 - \tilde{p}'_1)$, then $\tilde{P} > (<)\tilde{P}'$.

(4) Max rule:

If $\tilde{P} > \tilde{P}'$, then $\tilde{P} \vee \tilde{P}' = \tilde{P}$; otherwise, $\tilde{P} \vee \tilde{P}' = \tilde{P}'$.

2.2 Mathematical model of FIPPS

The FIPPS mathematical model is introduced below[12, 44, 45], and the notations used in this model are shown in Table 2.

Objective:

$$\min C_{\max} \tag{1}$$

Constraints:

$$\sum_l R_{irl} = 1, \forall i, r \tag{2}$$

$$X_{ij} \leqslant M \times (1 - W_{ijrl}) + M \times R_{irl}, \forall i, j, r, l \tag{3}$$

$$X_{ij} \geqslant 1 - M \times \sum_r \sum_l W_{ijrl} \times (1 - R_{irl}), \forall i, j \tag{4}$$

$$\sum_k Z_{ijk} = X_{ij}, \forall i, j \tag{5}$$

$$M \times (2 - X_{ij} - X_{ij'}) + Y_{ijij'} \geqslant U_{ijj'}, \forall i, j, j', j \neq j' \tag{6}$$

Table 2 Definition of notations in the FIPPS model.

Notation	Definition
N	Set of jobs
J_i	Operation set of Job i
R_i	OR-node set of Job i
M	Set of machines
i, i'	Job, $1 \leqslant i, i' \leqslant \|N\|$
j, j'	Operation, $1 \leqslant j, j' \leqslant \|J_i\|$
O_{ij}	Operation j of Job i
k, k'	Machine, $1 \leqslant k \leqslant \|M\|$
r, r'	OR-node, $1 \leqslant r \leqslant \|R_i\|$
l, l'	Link
Φ_i^{Job}	Operation set of Job i
Φ_i^{Last}	Possible last operation set of Job i
$U_{ijj'}$	1, O_{ij} is processed before $O_{ij'}$ according to the precedence relationship represented by the network of Job i; 0, otherwise
$\tilde{P}_{ijk}$	Fuzzy processing time of O_{ij} on Machine k
W_{ijrl}	1, O_{ij} is located in the l-th link of the r-th OR-node; 0, otherwise
R_{irl}	1, if the l-th link of the r-th OR-node of Job i is selected; 0, otherwise
X_{ij}	1, if O_{ij} is selected; 0, otherwise
Z_{ijk}	1, if O_{ij} is processed on Machine k; 0, otherwise
$Y_{iji'j'}$	1, if O_{ij} is processed before $O_{i'j'}$; 0, otherwise
$\tilde{S}_{ij}$	Fuzzy start time of O_{ij}
$\tilde{C}_{ij}$	Fuzzy completion time of O_{ij}
$\tilde{C}_{\max}$	Maximum fuzzy completion time

$$\tilde{S}_{ij'} \geqslant \tilde{S}_{ij} + \sum_k \tilde{P}_{ijk} \times Z_{ijk} - M \times (3 - X_{ij} - X_{ij'} - Y_{ijij'}), \quad \forall i, j, j', j \neq j' \tag{7}$$

$$\tilde{S}_{ij} \geqslant \sum_k \tilde{P}_{ijk} \times Z_{ij'k} - M \times (2 - X_{ij} - X_{ij'} + Y_{ijij'}), \quad \forall i, j, j', j \neq j' \tag{8}$$

$$\tilde{S}_{i'j'} \geqslant \tilde{S}_{ij} + \tilde{P}_{ijk} - M \times (3 - Z_{ijk} - Z_{i'j'k} - Y_{iji'j'}), \quad \forall i, i', j, j', i \neq i' \text{ or } j \neq j', k \tag{9}$$

$$\tilde{S}_{ij} \geqslant \tilde{S}_{i'j'} + \tilde{P}_{i'j'k} - M \times (2 - Z_{ijk} - Z_{i'j'k} + Y_{iji'j'}), \quad \forall i, i', j, j', i \neq i' \text{ or } j \neq j', k \tag{10}$$

$$C_{\max} \geqslant S_{ij} + \sum_k \tilde{P}_{i'j'k} \times Z_{ijk} - M \times (1 - X_{ij}), \quad \forall i, j \tag{11}$$

Constraint in Eq. (2) means that only one link of an OR node can be selected. Constraints in Formulas (3) and (4) are the OR-node's link-selecting conditions, while constraint in Eq. (5) indicates that one operation would only be assigned to one machine for processing. Constraint in Formula (6) corresponds to the precedence relationships in the process network. Meanwhile, constraints in Formulas (7) and (8) denote that the two operations of the same job are supposed to be processed sequentially according to the precedence relationships of the corresponding networks. Similarly, the operations assigned to the same machines should also be arranged according to the precedence constraints in the process networks that are formulating constraints in Formulas (9) and (10). Finally, constraint in Formula (11) is the maximum fuzzy completion time constraint. The TFN calculation rules make the model nonlinear, which means it cannot be solved and verified by a general mathematical solver. Therefore, the above model is only used for the problem description function.

3 Multiswarm Collaborative Optimization Algorithm for FIPPS

The proposed MSCOA framework is shown in Fig. 2. The main steps of MSCOA are described below.

Step 1: Randomly generate three swarms named swarm_1, swarm_2, and swarm_3, and set the generation number Gen=1.

Step 2: Perform crossover and mutation operator on swarm_2 and update its individuals.

Step 3: Perform crossover and mutation operator on swarm_3 along with the modified N8 searching on it to update its individuals.

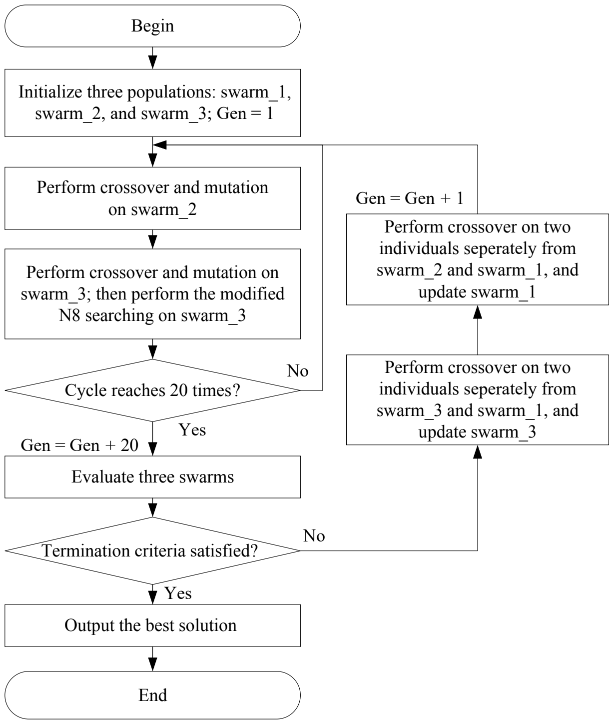

Fig. 2 MSCOA framework.

Step 4: Repeat Steps 2 and 3 20 times and set Gen = Gen + 20.

Step 5: Evaluate the three swarms and determine whether the termination criteria are satisfied. If "Yes", go to Step 7; if "No", go to Step 6.

Step 6: Execute crossover on individuals between swarm_3 and swarm_1, and update the individuals of swarm_3. Execute crossover on individuals between swarm_2 and swarm_1 and update the individuals of swarm_1. Gen = Gen + 1. Go to Step 2.

Step 7: Output the best solution.

Once the three swarms are generated randomly, swarm_1 does not undergo any evolution during the 20 iterations of Steps 2 and 3 to preserve the original status and to provide a basis for swarm backtracking. Instead, swarm_2 and swarm_3 are evolved differentially: crossover and mutation are executed for swarm_2, and a modified critical path-based N8 searching is additionally performed for swarm_3[46], which can enhance the local search capability of the MSCOA. Next, inter-swarm interactions are executed once sufficient differentiation between the various swarms has been generated (20 generations of evolution in each). These interactions are as follows: individuals from swarm_1 and swarm_3 are selected separately to perform crossover operators to update the two swarms; then, the same is done to individuals from swarm_1 and swarm_2 to update the two swarms.

The purpose of differential evolution is to increase the diversity of individuals and improve the overall global search ability of the algorithm. At present, the N8 neighborhood searching in swarm_3 is the most effective critical path structure for solving JSP[46], as it can effectively enhance the local searching ability. Finally, we can achieve a balance between the global and local searching of the MSCOA after following the above two mechanisms.

3.1 Encoding and decoding method

The paper adopts the integrated encoding approach proposed by Liu et al.[47] There are three coding strings for the corresponding three subproblems of the FIPPS problem. The first string is the OR-node string, which represents the link-selecting state of each OR-node. This is arranged according to the job number order. The inside numbers "1" and "2" represent the options of selecting either the left or the right link. As for the operation string, it is arranged according to the processing sequence. A pair of numbers "i-j" in the block represents the operation O_{ij}, as shown in Fig. 3. The third string is the machine string, which represents the processing machine assignment plan. This string is also arranged according to the job numbers from left to right. For example, the number "2" in the second block means that operation 2 of Job 1 selects Machine 2 for processing. We adopt the semiactive schedule decoding method to obtain the production scheme. Here, the operations are arranged one by one on the corresponding machines according to the processing sequence and the machine assignment plans.

3.2 Initialization and selection

In accordance with the abovementioned encoding rules, we adopt the random initialization method to initialize different strings of individuals. Thus, the OR-node strings are initialized by randomly selecting the corresponding links. Machine strings are initialized

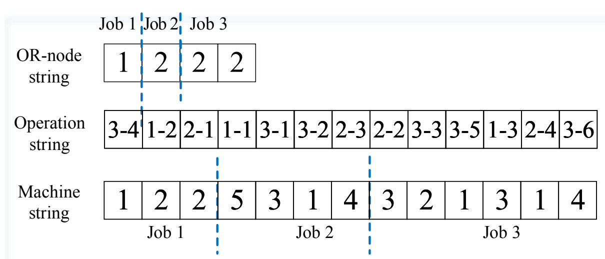

Fig. 3 Example of an encoding individual.

by randomly selecting the processing machine for each operation. As for the operation string, this is initialized by randomly sequencing the operations under the constraint of precedence relationships in the networks.

Next, we adopt the tournament selection method to select individuals that can carry evolutionary operators. Two individuals in the swarm are chosen randomly, and the one with better fitness will be selected if the randomly generated value (between 0 and 1) is greater than the predefined reproduction probability value P_r. Otherwise, the individual with the worse fitness will be selected. In the paper, the reproduction probability P_r is set to 0.8.

3.3 Crossover and mutation

The crossover for the OR-node string is shown in Fig. 4. One crossover point a is randomly selected. Parents 1 and 2 pass the left part of the blocks to the corresponding Offspring 1 and 2, after which they pass the right half of the blocks to the other one. The mutation operator for the OR-node string is a single-point shifting action, in which a mutation point is randomly selected, and the state of that point is shifted, as shown in Fig. 5.

The crossover operator for the operation string is shown in Fig. 6. Two crossover points, a and b, are randomly selected. The blocks outside the two points of Parent 1 are transferred to offspring individual 1, and the blocks outside the two points of Parent 2 are transferred to offspring individual 2. Then, Parent 2 passes the different blocks to Offspring 1, and Parent 1 passes the different blocks to Offspring 2. There are three steps

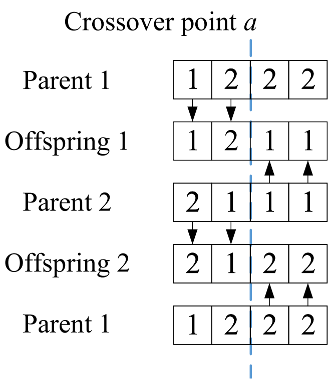

Fig. 4 Crossover for OR-node string.

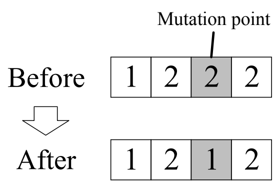

Fig. 5 Mutation for OR-node string.

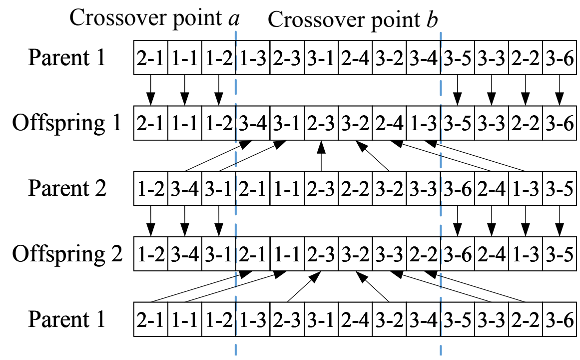

Fig. 6 Crossover for the operation string.

involved in conducting mutation on an operation string, as shown in Fig. 7. First, a mutation point is randomly chosen, after which the mutation range of the chosen point is determined according to the corresponding network. Finally, one of the available positions inside the mutation range is randomly selected, and the chosen operation is inserted.

Given that the position orders of their blocks in the machine string contain the corresponding job and operation information, the crossover operator transfers the blocks at the same positions from the parent to offspring individuals, as shown in Fig. 8. The mutation operator of the machine string is similar to the action of the OR-node string. Then, one mutation point is randomly selected, and the current state is shifted to another one, as shown in Fig. 9.

3.4 TFN-based critical path searching

The critical path-related approach was first applied in a study on JSP in Ref. [48]. Previous studies have proven

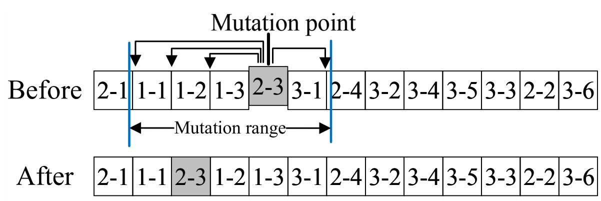

Fig. 7 Mutation for the operation string.

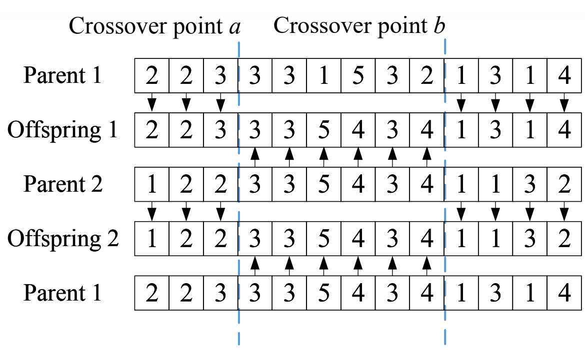

Fig. 8 Crossover for the machine string.

Mutation point

Before	2	2	3	3	3	1	5	3	2	1	3	1	4
After	2	2	3	3	3	5	5	3	2	1	3	1	4

Fig. 9 Mutation for the machine string.

that movements on noncritical paths do not change the makespan. In common shop scheduling problems, the critical path is defined as the longest path from the first operation of the first job to the last operation of the final job[49]. The maximum and comparing calculations are achieved according to the normal number rather than the TFN. Therefore, the calculation method of the critical path in the TFN way must be introduced.

A scheduling scheme of the FIPPS can be represented in the form of a disjunctive graph, as shown in Fig. 10. One disjunctive graph consists of a set of nodes and a set of arcs. The nodes correspond to the operations, while the arcs connect pairs of nodes, each representing one precedence relationship. The two nodes, O_{start} and O_{end}, represent the start and the end of the processing, respectively. The two consecutive operation nodes of the same job are connected by the conjunctive arcs indicated by the solid arrows, while the operation nodes assigned successively on the same machines are connected by the disjunctive arcs indicated by the dashed arrows[50]. The length of the arc between O_{ij} and $O_{i_0 j_0}$ is denoted as $D(O_{ij}, O_{i_0 j_0})$, and the maximum fuzzy completion time of the scheme is equal to $D(O_{\text{start}}, O_{\text{end}})$. In explaining the critical path calculation, we introduce the head length $\tilde{H}_{ij}$ and the tail length $\tilde{T}_{ij}$ of O_{ij} in the form of TFN. The critical path is the longest path from node O_{start} to node O_{end}, and the nodes within this range are considered critical operations[50]. According to this definition, the critical operation in fuzzy scheduling has the following attribution: $\tilde{H}_{ij} + \tilde{P}_{ij} + \tilde{T}_{ij} = L(O_{\text{start}}, O_{\text{end}})$, where P_{ij} is the fuzzy processing time

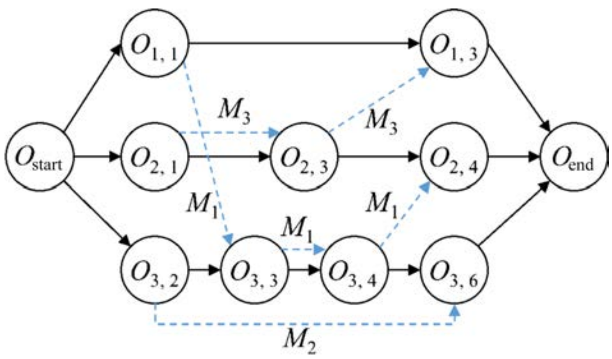

Fig. 10 Disjunctive graph of an FIPPS example.

of O_{ij} on its assigned machine, and the head and tail lengths can be obtained,

$$\tilde{H}_{\text{start}} = \tilde{T}_{\text{end}} = (0, 0.0) \tag{12}$$

$$\tilde{H}_{ij} = \max\{\tilde{H}_{\text{JP}_{ij}} + \tilde{P}_{\text{JP}_{ij}}, \tilde{R}_{\text{MP}_{ij}} + \tilde{P}_{\text{MP}_{ij}}\} \tag{13}$$

$$\tilde{T}_{ij} = \max\{\tilde{T}_{\text{JS}_{ij}} + \tilde{P}_{\text{JS}_{ij}}, \tilde{T}_{\text{MS}_{ij}} + \tilde{P}_{\text{MS}_{ij}}\} \tag{14}$$

In Eqs. (13) and (14), JP_{ij} is the predecessor operation of O_{ij} of the same job, and JS_{ij} is the successor operation of O_{ij} of the same job. In addition, MP_{ij} is the predecessor operation of O_{ij} on the same machine, and MS_{ij} is the successor operation of O_{ij} on the same machine. The critical path of one FIPPS scheme can be determined based on the abovementioned definition. Similarly, the effective movements on the operation nodes can also be applied to this fuzzy critical path. The most advanced neighborhood structure is N8, proposed by Xie et al.[46], which mainly has three types of movement: (1) moving operation in the critical block out to the noncritical path parts, (2) inserting the first or the last operation of the critical block into the inner segment, and (3) moving operation inside the critical block to the head or the tail of the block (see Fig. 11).

To increase the disturbance extent on the critical path, this paper modifies the three movements of the N8 neighborhood by (1) swapping the critical operation in the critical block with the operation on the noncritical path and by (2) swapping the head or tail operation with the inner operation of the critical path, as shown in Fig. 12.

4 Experimental Study and Discussion

In testing the effectiveness of the MSCOA, we performed comparative experiments with the classical GA and Particle Swarm Optimization (PSO) algorithm. The testing instances are extended from the most well-known Kim benchmark, which has been widely used in IPPS-related research since its introduction in 2003[51]. The benchmark contains 24 instances comprising 18 jobs and 15 machines. Each instance consists of a different combination of jobs. For follow-up researchers using a new benchmark, the fuzzy processing time data of the 18 jobs are presented in Tables A1–A6 in Appendix A. The minimum and maximum values are obtained from the processing time in Kim's original data with a random deviation of $\pm$(10–40)%. The original processing time is treated as the most probable value of the fuzzy processing time, and the iteration number is set to 500. The parameters are the Swarm_Size = 200, the reproduction probability $P_r = 0.8$, the crossover

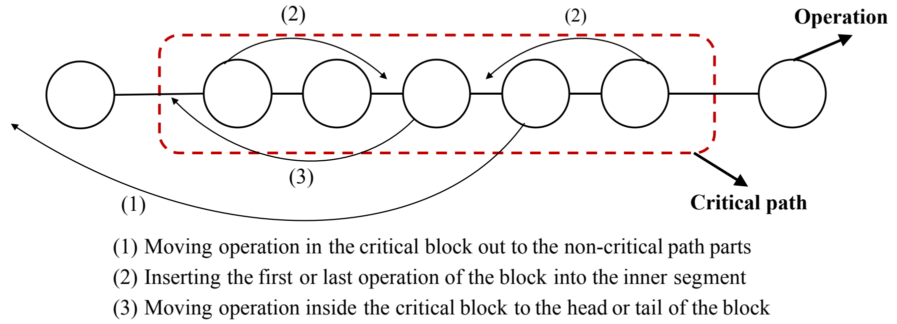

Fig. 11 N8 neighborhood structure schematic.

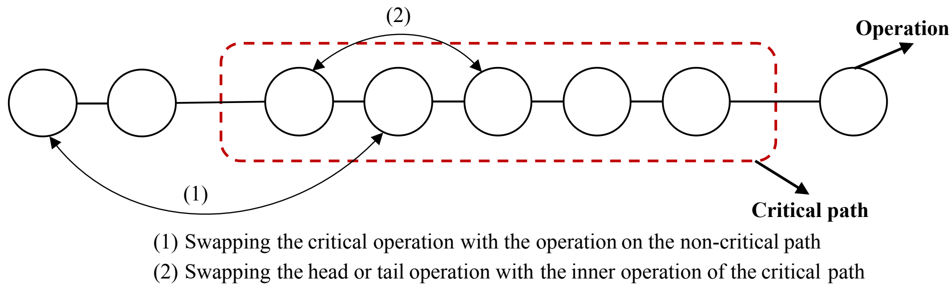

Fig. 12 New neighborhood structure adopted in MSCOA.

probability $P_c = 0.8$, and the mutation probability P_m=0.2. The experiment platform is a PC with an i7-8700 CPU and 16 GB RAM. The Gantt chart of the best solution (391, 522, 663) of Kim 24 is shown in Fig. 13. The results are obtained by running the algorithms independently 20 times in Tables 3–6.

Table 3 shows the best results that can be obtained upon using the three algorithms. The proposed MSCOA can find better results on all 24 problems than the best results of GA and PSO. From a statistical perspective, we analyze the results of 20 independent runs. As shown by the results, the proposed MSCOA has a greater advantage in the minimum, most probable, and maximum values of $\tilde{C}_{\max}$ with better average and standard deviation values, thus indicating the superiority and stability of the results. In Table 5, all 24 results of the most probable value are found to be superior to those of GA and PSO, thereby proving that the MOSCA outperforms the other two algorithms.

Observing the results in Tables 3–6, we can see

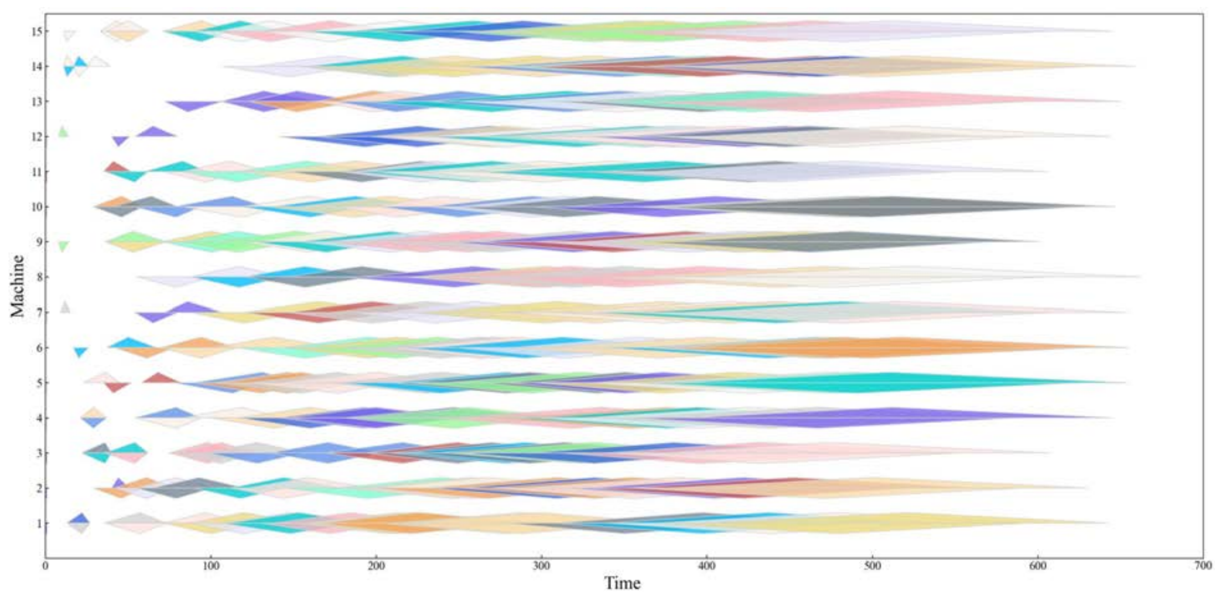

Fig. 13 Gantt chart of the best solution pertaining to Kim_24 (time unit is omitted according to the original data).

Table 3 Best results (maximum fuzzy completion time) of 20 independent runs of the three algorithms.

Number	MSCOA			GA			PSO		
	Minimum	Most probable	Maximum	Minimum	Most probable	Maximam	Minimum	Most probable	Maximum
1	**306**	**428**	**529**	319	429	533	313	428	526
2	**250**	**343**	**411**	257	348	425	255	345	414
3	**249**	**346**	**430**	270	352	428	265	349	426
4	**226**	**306**	**373**	234	306	368	217	305	388
5	**229**	**314**	**385**	226	318	378	227	314	397
6	**313**	**427**	**527**	331	451	559	323	437	545
7	**262**	**372**	**468**	266	372	467	265	375	460
8	**250**	**339**	**423**	264	339	429	257	345	415
9	**315**	**428**	**520**	318	430	527	313	427	527
10	**317**	**424**	**534**	358	464	577	340	437	541
11	**273**	**350**	**424**	285	371	454	275	354	424
12	**234**	**322**	**388**	247	337	408	236	336	404
13	**317**	**431**	**530**	355	471	575	351	467	585
14	**273**	**378**	**474**	298	403	510	295	392	473
15	**313**	**427**	**527**	341	445	555	322	436	537
16	**328**	**432**	**537**	363	494	613	363	483	576
17	**291**	**384**	**449**	348	466	569	316	432	538
18	**246**	**349**	**419**	284	390	496	283	386	484
19	**331**	**445**	**546**	399	505	624	376	490	601
20	**301**	**394**	**501**	342	464	570	320	448	572
21	**317**	**436**	**539**	369	482	602	363	462	583
22	**358**	**481**	**616**	423	568	702	388	543	674
23	**326**	**440**	**558**	378	528	635	348	490	626
24	**391**	**522**	**663**	445	619	776	445	590	726

Note: Highlighted values are the best ones of the same types of results.

that, on the one hand, the PSO can obtain better fuzzy values than the conventional GA algorithm, mainly due to the adoption of a more advanced evolutionary framework. The proposed MSCOA, on the other hand, can achieve better results than the two compared algorithms, benefiting from the enhanced swarm collaborative framework and the critical path-based neighborhood search. The experimental results also demonstrate the effectiveness of the improvements at the level of the algorithmic framework and the operators.

5 Conclusions and Future Research Directions

The current study focuses on the FIPPS problem considering fuzzy processing time and builds a process network based mathematical model. Then, we propose the MOSCA based on an integrated encoding method with critical path neighborhood searching. In this work, we sufficiently consider the advantages of the critical path-based N8 neighborhood searching and apply it to the FIPPS problem by modifying the calculation procedure according to the TFN rules. The experimental results on the extended famous benchmark indicate the outstanding performance of the proposed MSCOA in terms of its searching capability and stability.

Furthermore, by guiding the searching and evolving of each swarm in different directions, we ensure the diversity of all individuals and prevent the premature maturity of the algorithm. The proposed integrated encoding can also effectively promote the integration of the two systems of process and production, thus providing more candidate solutions with better quality. Combined with the modified critical path N8 searching, the proposed approach effectively reduces the invalid searching actions, enabling the proposed MSCOA to demonstrate significant superiority on famous open problems.

In the comparison results on the minimum values of $\tilde{C}_{\max}$, MSCOA had three small-scale instances that did not perform best. Therefore, exploring more adaptive swarm collaboration strategies in future works might improve the above situation. Furthermore, developing new critical path structures according to the TFN calculation rules is another important research direction

Table 4 Minimum values in $\tilde{C}_{\max}$ by T-test ($p\leqslant 0.05$).

Number	MSCOA		GA		PSO		Significant?	
	Average	Standard deviation	Average	Standard deviation	Average	Standard deviation	MSCOA better than GA	MSCOA better than PSO
1	**314.30**	**5.17**	330.70	10.08	327.95	9.36	**Yes**	**Yes**
2	**253.45**	**4.41**	264.40	7.38	262.60	8.42	**Yes**	**Yes**
3	**263.25**	**6.96**	278.70	12.24	277.90	9.69	**Yes**	**Yes**
4	**231.75**	5.64	235.40	**5.61**	234.85	7.11	No	No
5	**230.85**	**5.23**	238.10	7.22	235.95	7.19	**Yes**	No
6	**326.70**	**8.35**	352.65	13.76	337.45	8.70	**Yes**	**Yes**
7	**269.50**	**5.69**	274.55	6.35	271.50	6.80	No	No
8	**253.60**	**4.08**	265.00	8.01	265.15	11.22	**Yes**	**Yes**
9	**320.05**	7.69	330.10	11.62	324.10	**7.35**	**Yes**	No
10	**325.80**	**7.04**	364.20	14.45	348.50	12.44	**Yes**	**Yes**
11	**275.60**	9.52	304.15	**7.00**	296.60	14.35	**Yes**	**Yes**
12	**239.40**	**8.69**	272.60	13.73	262.15	13.22	**Yes**	**Yes**
13	**335.90**	**10.94**	376.35	16.05	363.65	12.66	**Yes**	**Yes**
14	**280.90**	**9.69**	316.80	14.15	306.90	14.48	**Yes**	**Yes**
15	**324.50**	9.43	354.05	**8.86**	344.40	16.07	**Yes**	**Yes**
16	**343.85**	15.38	395.20	21.52	374.80	**15.02**	**Yes**	**Yes**
17	**303.60**	**12.92**	363.05	17.56	352.70	15.31	**Yes**	**Yes**
18	**264.35**	12.29	322.50	20.37	300.60	**8.15**	**Yes**	**Yes**
19	**354.90**	**15.18**	413.30	19.21	395.25	22.42	**Yes**	**Yes**
20	**311.30**	**15.66**	367.30	17.09	344.80	17.58	**Yes**	**Yes**
21	**342.65**	**9.65**	395.40	15.76	374.35	15.34	**Yes**	**Yes**
22	**382.30**	**13.00**	444.60	16.80	428.30	19.11	**Yes**	**Yes**
23	**342.65**	**11.63**	421.05	17.19	395.30	20.29	**Yes**	**Yes**
24	**411.15**	**11.94**	494.05	25.55	467.85	22.22	**Yes**	**Yes**

Note: Highlighted values are the best ones of the same types of results.

that should be explored.

Appendix A

Table A1 Fuzzy processing time of Jobs 1–3.
Table A2 Fuzzy processing time of Jobs 4–6.
Table A3 Fuzzy processing time of Jobs 7–9.
Table A4 Fuzzy processing time of Jobs 10–12.
Table A5 Fuzzy processing time of Jobs 13–15.
Table A6 Fuzzy processing time of jobs 16–18.

Acknowledgment

This work was supported by the National Key R&D Program of China (No. 2022YFB3302700), the National Natural Science Foundation of China (Nos. U21B2029 and 52188102), and the Key R&D Program of Hubei Province (No. 2021AAB001).

References

[1] Y. Fu, Y. Hou, Z. Wang, X. Wu, K. Gao, and L. Wang, Distributed scheduling problems in intelligent manufacturing systems, *Tsinghua Science and Technology*, vol. 26, no. 5, pp. 625–645, 2021.

[2] L. Wang and D. Z. Zheng, An effective hybrid optimization strategy for job-shop scheduling problems, *Comput. Oper. Res.*, vol. 28, no. 6, pp. 585–596, 2001.

[3] X. Wu, Z. Cao, and S. Wu, Real-time hybrid flow shop scheduling approach in smart manufacturing environment, *Complex Syst. Model. Simul.*, vol. 1, no. 4, pp. 335–350, 2021.

[4] Y. Pan, K. Gao, Z. Li, and N. Wu, Solving biobjective distributed flow-shop scheduling problems with lot-streaming using an improved Jaya algorithm, *IEEE Trans. Cybern.*, vol. 53, no. 6, pp. 3818–3828, 2023.

[5] L. Wang, Z. Pan, and J. Wang, A review of reinforcement learning based intelligent optimization for manufacturing scheduling, *Complex Syst. Model. Simul.*, vol. 1, no. 4, pp. 257–270, 2021.

[6] B. Khoshnevis and Q. M. Chen, Integration of process planning and scheduling functions, *J. Intell. Manuf.*, vol. 2, no. 3, pp. 165–175, 1991.

[7] A. Jain, P. K. Jain, and I. P. Singh, An integrated scheme for process planning and scheduling in FMS, *Int. J. Adv. Manuf. Technol.*, vol. 30, nos. 11&12, pp. 1111–1118, 2006.

[8] F. T. S. Chan, S. H. Chung, and L. Y. Chan, An introduction of dominant genes in genetic algorithm for FMS, *Int. J. Prod. Res.*, vol. 46, no. 16, pp. 4369–4389, 2008.

[9] X. Li, X. Shao, L. Gao, and W. Qian, An effective hybrid

Table 5 Most probable values in $\tilde{C}_{\max}$ by T-test ($p\leqslant 0.05$).

Number	MSCOA		GA		PSO		Significant ?	
	Average	Standard deviation	Average	Standard deviation	Average	Standard deviation	MSCOA better than GA	MSCOA better than PSO
1	**427.20**	**0.52**	443.75	0.73	438.50	0.70	**Yes**	**Yes**
2	**344.80**	**0.48**	358.50	0.62	357.90	0.67	**Yes**	**Yes**
3	**346.60**	**0.61**	366.45	0.80	358.95	0.71	**Yes**	**Yes**
4	**307.25**	0.54	315.05	**0.54**	314.90	0.61	Yes	Yes
5	**321.55**	**0.52**	328.55	0.62	327.25	0.61	**Yes**	Yes
6	**433.80**	**0.66**	467.45	0.85	447.60	0.68	**Yes**	**Yes**
7	**371.35**	**0.55**	379.75	0.58	377.30	0.60	Yes	Yes
8	**345.00**	**0.46**	355.65	0.65	359.10	0.77	**Yes**	**Yes**
9	**424.40**	0.64	441.30	0.78	435.85	**0.62**	**Yes**	Yes
10	**438.85**	**0.61**	482.75	0.87	463.65	0.81	**Yes**	**Yes**
11	**359.75**	0.71	401.45	**0.61**	386.95	0.87	**Yes**	**Yes**
12	**331.05**	**0.68**	370.10	0.85	358.75	0.83	**Yes**	**Yes**
13	**447.90**	**0.76**	504.80	0.92	484.25	0.82	**Yes**	**Yes**
14	**385.85**	**0.71**	435.35	0.86	418.60	0.87	**Yes**	**Yes**
15	**433.45**	0.70	466.40	**0.68**	454.20	0.92	**Yes**	**Yes**
16	**456.65**	0.90	529.10	1.06	500.10	**0.89**	**Yes**	**Yes**
17	**409.35**	**0.82**	493.15	0.96	465.30	0.90	**Yes**	**Yes**
18	**367.25**	0.80	439.05	1.04	407.55	**0.65**	**Yes**	**Yes**
19	**475.95**	**0.89**	550.30	1.01	530.50	1.09	**Yes**	**Yes**
20	**426.20**	**0.91**	499.90	0.95	472.10	0.96	**Yes**	**Yes**
21	**453.80**	**0.71**	520.45	0.91	489.75	0.90	**Yes**	**Yes**
22	**505.45**	**0.83**	593.95	0.94	571.35	1.00	**Yes**	**Yes**
23	**468.85**	**0.78**	572.25	0.95	531.10	1.03	**Yes**	**Yes**
24	**550.40**	**0.79**	663.95	1.16	624.25	1.08	**Yes**	**Yes**

Note: Highlighted values are the best ones of the same types of results.

algorithm for integrated process planning and scheduling, *Int. J. Prod. Econ.*, vol. 126, no. 2, pp. 289–298, 2010.

[10] S. Zhang and T. N. Wong, Integrated process planning and scheduling: An enhanced ant colony optimization heuristic with parameter tuning, *J. Intell. Manuf.*, vol. 29, no. 3, pp. 585–601, 2018.

[11] X. Li, L. Gao, Q. Pan, L. Wan, and K. M. Chao, An effective hybrid genetic algorithm and variable neighborhood search for integrated process planning and scheduling in a packaging machine workshop, *IEEE Trans. Syst. Man Cybern.*: *Syst.*, vol. 49, no. 10, pp. 1933–1945, 2019.

[12] Q. Liu, X. Li, L. Gao, and J. Fan, A multi-MILP model collaborative optimization method for integrated process planning and scheduling problem, *IEEE Trans. Eng. Manage.*, doi: 10.1109/TEM.2022.3208431.

[13] E. Jiang, L. Wang, and J. Wang, Decomposition-based multi-objective optimization for energy-aware distributed hybrid flow shop scheduling with multiprocessor tasks, *Tsinghua Science and Technology*, vol. 26, no. 5, pp. 646–663, 2021.

[14] L. Wang, Q. K. Pan, P. N. Suganthan, W. H. Wang, and Y. M. Wang, A novel hybrid discrete differential evolution algorithm for blocking flow shop scheduling problems, *Comput. Oper. Res.*, vol. 37, no. 3, pp. 509–520, 2010.

[15] L. Wang, S. Wang, Y. Xu, G. Zhou, and M. Liu, A bi-population based estimation of distribution algorithm for the flexible job-shop scheduling problem, *Comput. Ind. Eng.*, vol. 62, no. 4, pp. 917–926, 2012.

[16] H. Li, K. Gao, P. Y. Duan, J. Q. Li, and L. Zhang, An improved artificial bee colony algorithm with Q-learning for solving permutation flow-shop scheduling problems, *IEEE Trans. Syst. Man Cybern.*: *Syst.*, vol. 53, no. 5, pp. 2684–2693, 2023.

[17] X. Han, Y. Han, Q. Chen, J. Li, H. Sang, Y. Liu, Q. Pan, and Y. Nojima, Distributed flow shop scheduling with sequence-dependent setup times using an improved iterated greedy algorithm, *Complex System Modeling and Simulation*, vol. 1, no. 3, pp. 198–217, 2021.

[18] Z. X. Pan, L. Wang, J. F. Chen, and Y. T. Wu, A novel evolutionary algorithm with adaptation mechanism for fuzzy permutation flow-shop scheduling, in *Proc. IEEE Congress on Evolutionary Computation*, Kraków, Poland, 2021, pp. 367–374.

[19] S. Wang, L. Wang, Y. Xu, and M. Liu, An effective estimation of distribution algorithm for the flexible job-shop scheduling problem with fuzzy processing time, *Int. J. Prod. Res.*, vol. 51, no. 12, pp. 3778–3793, 2013.

[20] W. Zhang, W. Hou, C. Li, W. Yang, and M. Gen, Multidirection update-based multiobjective particle swarm optimization for mixed no-idle flow-shop scheduling problem, *Complex System Modeling and Simulation*, vol. 1, no. 3, pp. 176–197, 2021.

Table 6 Maximum values in $\tilde{C}_{\max}$ by T-test ($p \leqslant 0.05$).

Number	MSCOA		GA		PSO		Significant ?	
	Average	Standard deviation	Average	Standard deviation	Average	Standard deviation	MSCOA better than GA	MSCOA better than PSO
1	**530.15**	**6.24**	551.35	14.00	546.15	14.21	**Yes**	**Yes**
2	**420.40**	**8.65**	435.80	14.37	435.95	17.47	**Yes**	**Yes**
3	**428.65**	**7.80**	448.15	15.30	441.95	12.09	**Yes**	**Yes**
4	**372.70**	7.14	383.40	**10.32**	386.90	11.90	Yes	Yes
5	**392.35**	**8.27**	404.70	10.25	401.60	9.07	**Yes**	Yes
6	**538.80**	**9.45**	582.25	18.77	558.30	13.27	**Yes**	**Yes**
7	**466.50**	**6.52**	478.35	9.39	472.60	11.54	Yes	No
8	**420.95**	**8.29**	434.75	12.35	440.55	24.74	**Yes**	**Yes**
9	**527.55**	6.07	546.95	13.36	543.10	**12.11**	**Yes**	Yes
10	**545.70**	**13.52**	608.35	21.48	577.35	21.37	**Yes**	**Yes**
11	**446.00**	10.46	496.55	**15.02**	482.15	19.63	**Yes**	**Yes**
12	**404.25**	**14.27**	455.75	21.10	441.45	22.01	**Yes**	**Yes**
13	**557.45**	**16.16**	628.65	24.53	603.25	13.75	**Yes**	**Yes**
14	**485.20**	**10.10**	548.10	18.71	519.75	26.34	**Yes**	**Yes**
15	**539.40**	11.49	582.05	**14.90**	563.40	19.54	**Yes**	**Yes**
16	**571.20**	21.90	663.70	34.52	625.80	**22.86**	**Yes**	**Yes**
17	**503.25**	**19.66**	605.60	25.95	575.80	28.72	**Yes**	**Yes**
18	**456.10**	16.32	543.20	22.90	506.60	**19.27**	**Yes**	**Yes**
19	**595.55**	**18.76**	685.35	25.83	666.45	33.07	**Yes**	**Yes**
20	**530.35**	**22.02**	623.70	21.69	591.25	23.09	**Yes**	**Yes**
21	**565.50**	**15.78**	649.90	26.27	610.20	20.75	**Yes**	**Yes**
22	**634.75**	**14.89**	743.70	22.79	711.25	25.03	**Yes**	**Yes**
23	**582.60**	**20.87**	710.45	33.51	666.75	24.58	**Yes**	**Yes**
24	**684.60**	**18.59**	826.40	28.09	773.05	32.74	**Yes**	**Yes**

Note: Highlighted values are the best ones of the same types of results.

[21] K. Z. Gao, P. N. Suganthan, Q. K. Pan, and M. F. Tasgetiren, An effective discrete harmony search algorithm for flexible job shop scheduling problem with fuzzy processing time, *Int. J. Prod. Res.*, vol. 53, no. 19, pp. 5896–5911, 2015.

[22] Y. Xu, L. Wang, S. Y. Wang, and M. Liu, An effective teaching-learning-based optimization algorithm for the flexible job-shop scheduling problem with fuzzy processing time, *Neurocomputing*, vol. 148, pp. 260–268, 2015.

[23] R. Li, W. Gong, L. Wang, C. Lu, and S. Jiang, Two-stage knowledge-driven evolutionary algorithm for distributed green flexible job shop scheduling with type-2 fuzzy processing time, *Swarm Evol. Comput.*, vol. 74, p. 101139, 2022.

[24] R. Li and W. Y. Gong, An improved multi-objective evolutionary algorithm based on decomposition for bi-objective fuzzy flexible job-shop scheduling problem, (in Chinese), *Control Theory Appl.*, vol. 39, no. 1, pp. 31–40, 2022.

[25] J. Behnamian, Survey on fuzzy shop scheduling, *Fuzzy Optim. Decis. Making*, vol. 15, no. 3, pp. 331–366, 2016.

[26] S. Abdullah and M. Abdolrazzagh-Nezhad, Fuzzy job-shop scheduling problems: A review, *Inf. Sci.*, vol. 278, pp. 380–407, 2014.

[27] L. Wang, G. Zhou, Y. Xu, and M. Liu, A hybrid artificial bee colony algorithm for the fuzzy flexible job-shop scheduling problem, *Int. J. Prod. Res.*, vol. 51, no. 12, pp. 3593–3608, 2013.

[28] G. G. Wang, D. Gao, and W. Pedrycz, Solving multiobjective fuzzy job-shop scheduling problem by a hybrid adaptive differential evolution algorithm, *IEEE Trans. Ind. Inform.*, vol. 18, no. 12, pp. 8519–8528, 2022.

[29] D. Gao, G. G. Wang, and W. Pedrycz, Solving fuzzy job-shop scheduling problem using de algorithm improved by a selection mechanism, *IEEE Trans. Fuzzy Syst.*, vol. 28, no. 12, pp. 3265–3275, 2020.

[30] J. Cai and D. Lei, A cooperated shuffled frog-leaping algorithm for distributed energy-efficient hybrid flow shop scheduling with fuzzy processing time, *Complex Intell. Syst.*, vol. 7, no. 5, pp. 2235–2253, 2021.

[31] J. Q. Li, Z. M. Liu, C. Li, and Z. X. Zheng, Improved artificial immune system algorithm for type-2 fuzzy flexible job shop scheduling problem, *IEEE Trans. Fuzzy Syst.*, vol. 29, no. 11, pp. 3234–3248, 2021.

[32] R. Li, W. Gong, and C. Lu, Self-adaptive multi-objective evolutionary algorithm for flexible job shop scheduling with fuzzy processing time, *Comput. Ind. Eng.*, vol. 168, p. 108099, 2022.

[33] W. Gong, Z. Liao, X. Mi, L. Wang, and Y. Guo, Nonlinear equations solving with intelligent optimization algorithms: A survey, *Complex System Modeling and Simulation*, vol. 1, no. 1, pp. 15–32, 2021.

Table A1 Fuzzy processing time of Jobs 1–3.

Job No.	Operation	Alternative machine	Processing time
1	1	9, 14	(12, 13, 16), (8, 10, 13)
	2	11, 15	(20, 24, 32), (15, 18, 25)
	3	15	(27, 43, 59)
	4	12	(29, 43, 47)
	5	13	(22, 30, 38)
	6	4, 12	(28, 32, 38), (16, 25, 31)
	7	1, 5, 11	(26, 40, 48), (32, 49, 68), (34, 39, 45)
	8	8	(35, 47, 57)
2	1	5, 8, 14	(8, 10, 14), (13, 16, 22), (11, 13, 18)
	2	8, 9, 15	(5, 6, 7), (6, 8, 10), (4, 7, 9)
	3	4	(29, 40, 49)
	4	6, 7, 9, 12	(12, 14, 17), (12, 20, 28), (8, 10, 12), (10, 13, 15)
	5	1, 7, 11	(20, 33, 41), (31, 40, 48), (29, 43, 49)
	6	1, 5	(30, 42, 48), (25, 38, 42)
	7	6, 11, 15	(19, 25, 31), (23, 33, 42), (22, 30, 35)
	8	10, 15	(36, 41, 52), (37, 44, 53)
	9	2, 13	(9, 10, 12), (8, 12, 14)
	10	11, 14, 15	(25, 34, 38), (15, 24, 30), (22, 30, 38)
	11	6, 11	(26, 38, 52), (31, 42, 56)
	12	4, 8, 12	(20, 25, 33), (17, 26, 34), (18, 30, 35)
	13	7	(30, 39, 50)
	14	10, 12	(23, 37, 50), (27, 40, 51)
3	1	4, 7, 11	(21, 29, 37), (31, 36, 50), (27, 34, 47)
	2	2, 8, 11, 12, 15	(28, 35, 43), (21, 29, 37), (17, 27, 37), (26, 30, 35), (26, 33, 40)
	3	1, 6, 7, 10, 14	(9, 11, 14), (7, 9, 11), (7, 8, 9), (15, 19, 22), (9, 12, 15)
	4	2, 3, 6, 7	(14, 18, 23), (13, 20, 23), (20, 27, 36), (11, 13, 16)
	5	2, 3, 4, 5, 7	(13, 19, 22), (16, 24, 30), (17, 22, 26), (23, 31, 41), (27, 37, 41)
	6	1, 7, 10, 12, 13	(9, 13, 16), (5, 8, 11), (10, 12, 16), (5, 9, 12), (4, 5, 6)
	7	1, 5, 6, 13	(36, 50, 56), (28, 39, 46), (30, 44, 61), (36, 48, 54)
	8	1, 12	(5, 6, 7), (7, 9, 12)
	9	3, 4, 5, 13, 14	(29, 44, 57), (32, 36, 50), (19, 30, 34), (26, 39, 45), (28, 33, 41)
	10	3, 4, 7, 12, 13	(28, 39, 50), (29, 45, 59), (33, 41, 56), (42, 50, 67), (34, 40, 54)
	11	2, 4, 9	(23, 29, 40), (22, 36, 40), (25, 33, 38)
	12	1, 4, 9, 12, 14	(16, 19, 22), (13, 20, 25), (11, 17, 21), (11, 16, 22), (14, 21, 29)
	13	9, 14, 15	(33, 40, 45), (23, 33, 45), (25, 35, 48)
	14	2, 4, 6, 7	(9, 11, 14), (9, 12, 16), (12, 14, 18), (11, 15, 19)
	15	1, 8, 12, 13, 14	(7, 10, 13), (16, 19, 26), (16, 20, 23), (14, 17, 21), (14, 16, 21)
	16	10, 15	(44, 49, 65), (34, 44, 51)
	17	10, 12, 13	(18, 20, 23), (28, 33, 40), (28, 39, 44)
	18	6, 7, 9, 10, 13	(24, 30, 38), (21, 29, 33), (31, 40, 46), (24, 39, 48), (26, 33, 45)
	19	3, 8, 10, 14, 15	(14, 20, 22), (23, 29, 35), (27, 40, 50), (27, 34, 47), (28, 31, 40)

[34] X. Wen, X. Li, L. Gao, L. Wan, and W. Wang, Multi-objective genetic algorithm for integrated process planning and scheduling with fuzzy processing time, in *Proc. 6th Int. Conf. Advanced Computational Intelligence*, Hangzhou, China, 2013, pp. 293–298.

[35] S. Zhang, Z. Yu, W. Zhang, D. Yu, and Y. Xu, An extended genetic algorithm for distributed integration of fuzzy process planning and scheduling, *Math. Probl. Eng.*, vol. 2016, p. 3763512, 2016.

[36] X. Wen, X. Lian, K. Wang, H. Li, and G. Luo, Multi-layer collaborative optimization method for solving fuzzy multi-objective integrated process planning and scheduling, *Meas. Control*, vol. 53, nos. 9&10, pp. 1883–1901, 2020.

[37] H. Zhang, J. Xie, J. Ge, J. Shi, and Z. Zhang, Hybrid particle swarm optimization algorithm based on entropy theory for solving DAR scheduling problem, *Tsinghua Science and Technology*, vol. 24, no. 3, pp. 282–290, 2019.

[38] L. Wang, G. Zhou, Y. Xu, S. Wang, and M. Liu, An effective artificial bee colony algorithm for the flexible job-shop scheduling problem, *Int. J. Adv. Manuf. Technol.*, vol. 60,

Table A2 Fuzzy processing time of Jobs 4–6.

Job No.	Operation	Alternative machine	Processing time
4	1	2	(13, 18, 23)
	2	8, 12	(33, 38, 44), (27, 30, 35)
	3	9	(16, 20, 26)
	4	5, 11	(6, 8, 9), (7, 9, 13)
	5	3	(21, 29, 40)
	6	2, 4, 5	(23, 36, 44), (27, 33, 41), (33, 39, 50)
	7	1, 10	(15, 23, 31), (17, 20, 25)
	8	6	(32, 45, 50)
	9	3, 12	(4, 5, 7), (5, 9, 12)
	10	5, 9	(30, 39, 47), (27, 33, 39)
	11	7, 11	(24, 36, 40), (30, 41, 57)
	12	14, 15	(19, 31, 35), (18, 29, 39)
	13	1, 2, 6	(19, 28, 33), (18, 22, 27), (16, 21, 24)
	14	10, 14	(16, 18, 24), (20, 28, 39)
	15	12	(20, 24, 27)
	16	7, 9	(18, 23, 27), (17, 25, 29)
5	1	3, 4, 5, 11	(29, 35, 48), (21, 29, 33), (23, 36, 45), (27, 31, 38)
	2	1, 3, 5, 10, 15	(35, 40, 45), (30, 34, 39), (36, 44, 51), (36, 41, 48), (34, 39, 45)
	3	11, 15	(12, 15, 17), (9, 13, 15)
	4	2, 6, 7, 10, 14	(27, 31, 42), (26, 33, 43), (21, 29, 40), (22, 27, 37), (19, 25, 34)
	5	6, 12, 13, 14	(9, 13, 16), (6, 9, 12), (6, 8, 11), (11, 14, 17)
	6	11, 12	(24, 28, 34), (21, 29, 37)
	7	2, 4, 10, 12, 14	(24, 31, 37), (20, 24, 27), (25, 28, 34), (21, 26, 36), (28, 32, 38)
	8	6, 10, 11, 13	(27, 34, 38), (25, 33, 40), (23, 30, 38), (24, 29, 40)
	9	3, 4, 13	(34, 41, 55), (24, 37, 45), (25, 40, 45)
	10	1, 4, 5, 8, 9	(33, 38, 51), (21, 29, 38), (21, 35, 45), (26, 30, 35), (23, 31, 39)
	11	6, 10, 11, 14, 15	(41, 48, 65), (41, 50, 68), (32, 44, 59), (29, 41, 57), (35, 47, 53)
	12	6, 7, 9, 12, 13	(18, 26, 33), (26, 32, 39), (28, 38, 42), (19, 29, 36), (23, 30, 39)
	13	6, 10, 13, 14, 15	(18, 23, 25), (13, 20, 25), (19, 25, 34), (12, 18, 24), (17, 22, 25)
	14	2, 8, 10, 12	(11, 14, 17), (9, 11, 14), (14, 17, 20), (10, 13, 17)
	15	3, 5, 13	(20, 27, 38), (17, 24, 31), (17, 26, 30)
	16	4, 8, 13	(14, 20, 23), (13, 19, 23), (12, 14, 16)
	17	2, 3, 4, 7, 14	(17, 27, 34), (14, 21, 28), (24, 28, 37), (20, 30, 36), (23, 29, 39)
	18	3, 7, 10, 14	(34, 39, 53), (29, 34, 46), (27, 40, 50), (26, 35, 41)
6	1	3, 4, 11	(31, 38, 49), (27, 33, 40), (32, 36, 44)
	2	2, 3, 6	(14, 22, 26), (14, 21, 26), (12, 19, 23)
	3	5	(11, 14, 16)
	4	5, 8	(15, 17, 22), (15, 20, 22)
	5	3, 6, 11	(22, 36, 40), (27, 33, 38), (31, 39, 54)
	6	2, 3, 10	(16, 24, 31), (16, 20, 25), (12, 18, 21)
	7	3, 9, 15	(16, 21, 23), (15, 17, 19), (16, 24, 32)
	8	8	(33, 38, 44)
	9	4, 11	(15, 19, 22), (9, 15, 20)
	10	8, 10, 15	(11, 14, 18), (15, 19, 23), (11, 17, 23)
	11	1, 4, 9, 13	(22, 25, 30), (14, 21, 27), (15, 19, 23), (21, 28, 36)
	12	7, 12	(29, 42, 52), (30, 43, 58)
	13	1, 2, 6	(41, 48, 67), (27, 42, 58), (29, 46, 54)
	14	7, 10	(7, 10, 13), (11, 14, 18)
	15	2, 13, 14	(10, 14, 19), (14, 16, 20), (11, 13, 15)
	16	1, 4, 5, 8	(22, 36, 48), (24, 33, 44), (20, 31, 37), (26, 34, 45)
	17	1, 12	(30, 47, 56), (27, 44, 48)
	18	9, 10, 13	(25, 30, 36), (21, 26, 34), (18, 29, 33)
	19	1, 5, 12	(11, 18, 24), (14, 19, 25), (11, 15, 17)
	20	9, 11	(17, 24, 30), (22, 25, 28)

Table A3 Fuzzy processing time of Jobs 7–9.

Job No.	Operation	Alternative machine	Processing time
7	1	7, 8	(9, 12, 15), (12, 17, 20)
	2	2, 5, 8, 12	(5, 7, 9), (5, 6, 8), (6, 8, 11), (8, 11, 13)
	3	11, 15	(21, 30, 34), (24, 27, 30)
	4	7	(24, 27, 35)
	5	2, 3, 9	(7, 10, 12), (8, 11, 14), (13, 16, 19)
	6	4, 11, 13	(32, 46, 58), (45, 50, 67), (44, 49, 67)
	7	3	(18, 22, 28)
	8	5, 10	(7, 9, 12), (7, 10, 11)
	9	1, 2, 6	(20, 27, 37), (17, 28, 36), (19, 24, 31)
	10	6, 12, 13	(7, 8, 10), (4, 5, 7), (8, 11, 14)
	11	2, 9	(31, 47, 65), (40, 48, 56)
	12	1, 8	(22, 27, 37), (24, 30, 40)
	13	9, 13, 14	(12, 18, 24), (15, 19, 22), (16, 20, 28)
	14	7, 11	(14, 22, 29), (15, 20, 25)
	15	7, 8, 13	(10, 13, 16), (8, 11, 13), (9, 14, 19)
	16	6, 14	(12, 15, 19), (7, 10, 12)
	17	4, 5, 7	(15, 21, 26), (20, 26, 35), (17, 20, 25)
	18	4, 5, 13	(19, 29, 34), (26, 30, 35), (19, 26, 30)
	19	1, 2	(23, 35, 46), (24, 31, 41)
	20	5, 6, 11	(15, 22, 25), (15, 18, 22), (14, 23, 32)
	21	1, 7, 10	(22, 32, 40), (25, 33, 43), (24, 28, 39)
8	1	4	(33, 50, 57)
	2	8, 15	(15, 23, 28), (17, 21, 24)
	3	12	(22, 35, 42)
	4	4, 7, 8	(7, 11, 13), (11, 13, 16), (13, 16, 22)
	5	2, 11	(14, 18, 22), (15, 20, 25)
	6	4, 13	(32, 36, 49), (20, 33, 44)
	7	1, 3	(24, 38, 49), (25, 35, 44)
	8	6, 15	(12, 16, 20), (15, 17, 21)
	9	9	(20, 24, 32)
	10	5, 14	(17, 23, 27), (20, 26, 29)
	11	1, 5, 8	(11, 15, 19), (12, 16, 18), (10, 17, 20)
	12	3, 10	(38, 43, 47), (32, 49, 58)
	13	10	(29, 44, 59)
	14	13, 14	(20, 32, 44), (27, 31, 41)
	15	10, 13	(28, 36, 45), (27, 38, 42)
	16	14	(21, 28, 38)
	17	2, 10	(26, 39, 54), (24, 34, 39)
	18	5, 7	(12, 18, 22), (11, 15, 18)
	19	11	(14, 16, 20)
	20	3, 12	(33, 45, 53), (36, 48, 53)
9	1	5, 9, 12, 13, 14	(28, 31, 35), (24, 27, 37), (14, 21, 29), (23, 28, 34), (17, 23, 28)
	2	7	(18, 21, 27)
	3	7, 9, 10, 13	(19, 21, 26), (16, 22, 27), (17, 25, 33), (17, 20, 24)
	4	3, 15	(12, 13, 16), (12, 15, 18)
	5	3, 6, 9, 12, 13	(4, 6, 7), (4, 5, 6), (6, 7, 8), (7, 10, 12), (7, 9, 12)
	6	1, 2, 5, 6, 10	(23, 37, 50), (20, 33, 43), (32, 39, 53), (22, 29, 38), (20, 32, 36)
	7	4, 10, 11, 13, 15	(6, 7, 8), (6, 8, 9), (6, 9, 12), (4, 5, 7), (4, 6, 7)

(To be continued)

Table A3 Fuzzy processing time of Jobs 7–9.

(Continued)

Job No.	Operation	Alternative machine	Processing time
9	8	1, 4, 8, 9, 12	(35, 42, 49), (34, 41, 49), (35, 39, 47), (31, 45, 61), (34, 44, 57)
	9	4, 7, 14	(8, 10, 14), (8, 9, 12), (10, 14, 19)
	10	4, 5, 7, 8, 14	(13, 19, 22), (12, 14, 17), (11, 15, 18), (11, 12, 16), (8, 10, 14)
	11	3, 9, 10, 11, 15	(23, 28, 39), (15, 24, 32), (18, 27, 32), (14, 22, 31), (17, 20, 25)
	12	4, 6	(34, 45, 58), (29, 41, 51)
	13	2	(40, 44, 51)
	14	4, 5, 14, 15	(34, 47, 52), (33, 43, 59), (29, 44, 57), (33, 42, 48)
	15	2, 5, 6, 9, 14	(12, 17, 19), (11, 14, 15), (15, 20, 25), (18, 21, 26), (14, 18, 24)
	16	2, 4, 13	(39, 45, 55), (28, 42, 48), (34, 46, 58)
	17	1, 3, 5, 6, 15	(20, 27, 30), (20, 25, 28), (17, 23, 30), (24, 28, 35), (17, 20, 24)
	18	2, 7, 9, 10, 11	(8, 10, 11), (8, 12, 16), (14, 17, 21), (13, 16, 21), (7, 11, 13)
	19	2, 13	(6, 9, 12), (8, 10, 13)
	20	4, 6, 8, 12	(12, 18, 21), (15, 23, 31), (18, 21, 28), (22, 25, 30)

Table A4 Fuzzy processing time of Jobs 10–12.

Jobs No.	Operations	Alternative machines	Processing time
10	1	1, 2, 3, 4	(22, 34, 47), (27, 39, 45), (34, 40, 54), (24, 33, 42)
	2	6, 15	(19, 27, 33), (18, 20, 24)
	3	1, 13	(13, 22, 27), (16, 24, 27)
	4	10, 13	(15, 22, 29), (16, 20, 23)
	5	4, 7	(33, 37, 41), (31, 35, 48)
	6	5, 9	(8, 10, 12), (8, 12, 15)
	7	8, 12, 14	(32, 39, 44), (24, 32, 40), (23, 36, 50)
	8	12	(32, 44, 53)
	9	2, 3, 6, 9	(19, 23, 27), (15, 24, 30), (15, 21, 29), (15, 19, 26)
	10	3, 12	(42, 48, 66), (30, 45, 53)
	11	14	(11, 17, 22)
11	1	1, 6	(30, 38, 47), (20, 30, 41)
	2	5, 8, 14, 15	(33, 39, 48), (33, 40, 51), (22, 36, 45), (27, 44, 59)
	3	3, 5, 11, 12, 13	(8, 11, 13), (9, 13, 15), (8, 9, 10), (10, 12, 13), (6, 8, 11)
	4	5, 6, 8, 13, 14	(16, 21, 26), (16, 23, 29), (18, 29, 37), (20, 27, 35), (18, 25, 28)
	5	3, 4, 6	(22, 33, 39), (28, 31, 37), (23, 29, 36)
	6	2, 10	(21, 28, 33), (19, 27, 38)
	7	1, 14, 15	(30, 40, 49), (28, 42, 53), (35, 46, 63)
	8	2, 7, 9, 11, 14	(4, 6, 7), (5, 8, 10), (7, 10, 13), (9, 11, 12), (6, 7, 8)
	9	5, 9, 13	(30, 40, 45), (27, 39, 50), (31, 36, 40)
12	1	1, 11	(22, 31, 36), (24, 29, 36)
	2	8, 15	(33, 46, 63), (27, 44, 57)
	3	5, 11	(4, 5, 6), (7, 11, 13)
	4	12	(29, 41, 53)
	5	15	(19, 24, 32)
	6	2, 13	(27, 42, 52), (39, 45, 62)
	7	8, 11	(14, 19, 22), (11, 15, 21)
	8	3, 12	(14, 18, 22), (17, 20, 27)
	9	6, 14	(4, 5, 6), (5, 7, 9)
	10	4	(15, 18, 22)
	11	7	(25, 39, 53)
	12	6, 10	(12, 13, 16), (5, 7, 9)
	13	2, 3	(22, 26, 35), (16, 22, 25)
	14	1, 8, 13	(4, 5, 6), (5, 8, 10), (7, 9, 11)
	15	9	(27, 39, 44)
	16	7, 10	(6, 10, 13), (11, 13, 14)
	17	4, 12	(25, 41, 46), (25, 38, 48)
	18	5, 9, 13	(17, 21, 29), (16, 22, 29), (15, 19, 23)

Table A5 Fuzzy processing time of Jobs 13–15.

Job No.	Operation	Alternative machine	Processing time
13	1	3, 4, 10, 11, 13	(35, 46, 61), (31, 47, 57), (31, 44, 57), (34, 41, 54), (35, 50, 59)
	2	5, 7, 10, 12, 15	(5, 8, 9), (3, 5, 7), (8, 10, 14), (10, 11, 13), (6, 9, 11)
	3	2, 7	(13, 16, 19), (8, 12, 15)
	4	1, 3, 4, 6, 15	(5, 7, 8), (4, 5, 7), (8, 13, 16), (11, 12, 15), (7, 8, 10)
	5	6, 8, 10	(21, 26, 36), (17, 24, 29), (24, 28, 37)
	6	1, 9, 12, 15	(4, 5, 6), (3, 4, 5), (5, 7, 8), (8, 9, 11)
	7	2, 8, 10, 12	(23, 27, 34), (21, 30, 36), (28, 33, 42), (23, 29, 35)
	8	2	(32, 40, 47)
	9	3, 6, 7, 9	(19, 23, 26), (16, 24, 33), (18, 29, 33), (16, 21, 29)
	10	1, 2, 5, 9, 10	(9, 12, 15), (12, 14, 17), (14, 19, 26), (11, 18, 21), (13, 17, 20)
	11	10, 11, 12	(30, 47, 53), (43, 49, 61), (42, 50, 60)
	12	4, 6, 13	(39, 44, 57), (29, 38, 47), (27, 41, 48)
	13	1, 9, 10, 12	(16, 22, 26), (15, 21, 29), (14, 16, 21), (14, 18, 22)
	14	1, 4, 6, 13, 14	(11, 15, 18), (15, 18, 22), (10, 13, 15), (12, 14, 17), (13, 19, 21)
	15	3, 4, 6, 13	(5, 6, 7), (3, 4, 5), (4, 5, 6), (8, 9, 12)
	16	5, 11, 14	(11, 15, 19), (13, 18, 21), (9, 13, 15)
	17	8, 9, 11	(13, 15, 21), (11, 16, 22), (12, 19, 22)
	18	2, 15	(32, 44, 59), (41, 50, 64)
14	1	3, 9	(38, 46, 63), (28, 43, 59)
	2	1, 2, 7, 12	(7, 10, 12), (13, 17, 23), (9, 11, 14), (9, 13, 18)
	3	4, 7, 8	(6, 8, 9), (6, 9, 11), (8, 10, 11)
	4	3, 6	(11, 18, 20), (18, 25, 30)
	5	4	(6, 9, 12)
	6	3, 4, 13	(25, 29, 35), (22, 27, 37), (20, 33, 37)
	7	5, 9	(22, 30, 33), (23, 29, 32)
	8	2, 3	(7, 9, 10), (5, 8, 11)
	9	5, 12, 15	(12, 18, 23), (8, 10, 14), (14, 19, 26)
	10	9, 15	(22, 28, 32), (16, 25, 32)
	11	4, 14, 15	(26, 42, 54), (31, 43, 49), (37, 47, 55)
	12	9, 10, 14	(22, 35, 47), (26, 31, 38), (26, 29, 38)
	13	6, 10	(8, 9, 10), (5, 7, 10)
15	1	1, 11	(17, 20, 25), (14, 18, 23)
	2	12, 13	(27, 41, 50), (27, 43, 58)
	3	6	(12, 17, 23)
	4	7	(5, 8, 11)
	5	2, 15	(8, 12, 16), (13, 15, 18)
	6	4, 5	(29, 48, 63), (35, 43, 52)
	7	9	(35, 47, 58)
	8	8, 12	(17, 28, 34), (22, 30, 36)
	9	2, 8	(16, 18, 23), (19, 22, 30)
	10	14	(37, 50, 55)
	11	8, 13	(5, 6, 8), (4, 7, 9)
	12	5, 7	(35, 48, 67), (33, 45, 56)
	13	1, 5, 6	(7, 9, 12), (8, 10, 14), (10, 11, 14)
	14	3, 6, 9	(17, 22, 26), (17, 24, 28), (19, 21, 28)
	15	1, 12	(31, 42, 50), (31, 47, 65)

Table A6 Fuzzy processing times of Jobs 16–18.

Job No.	Operation	Alternative machine	Processing time
16	1	1, 2, 11	(35, 43, 52), (31, 45, 53), (35, 41, 52)
	2	7	(26, 32, 43)
	3	3, 6, 13	(25, 33, 46), (27, 39, 50), (26, 35, 48)
	4	4, 5, 7, 9, 15	(32, 40, 51), (28, 43, 49), (30, 41, 48), (37, 44, 51), (42, 49, 61)
	5	1, 2, 6	(16, 25, 30), (24, 30, 39), (19, 31, 35)
	6	6, 12	(4, 5, 6), (7, 10, 12)
	7	3, 5, 6, 11, 14	(6, 7, 9), (4, 5, 6), (8, 9, 11), (6, 8, 9), (7, 10, 13)
	8	5, 7, 9	(10, 16, 22), (14, 19, 24), (14, 18, 20)
	9	1, 4, 9, 10, 13	(10, 12, 15), (11, 18, 25), (12, 14, 16), (6, 9, 12), (9, 15, 18)
	10	6, 11, 12, 13, 14	(17, 19, 25), (8, 11, 13), (14, 16, 20), (14, 20, 27), (19, 21, 25)
	11	4, 7, 8, 10, 15	(24, 31, 35), (32, 39, 47), (19, 30, 39), (26, 33, 39), (32, 40, 48)
	12	5, 8, 14	(22, 28, 35), (20, 27, 36), (17, 24, 27)
	13	1, 4, 7, 8, 15	(42, 50, 60), (37, 44, 50), (30, 47, 64), (40, 49, 67), (35, 48, 62)
	14	1, 2, 8, 9	(11, 17, 19), (13, 19, 21), (12, 20, 25), (16, 21, 28)
	15	1, 2, 4, 5, 7	(6, 8, 10), (6, 7, 8), (4, 6, 7), (8, 9, 12), (8, 10, 12)
	16	9, 12	(3, 5, 6), (4, 6, 7)
	17	1, 3, 8, 13, 15	(18, 21, 26), (15, 20, 26), (16, 24, 30), (17, 19, 21), (18, 23, 26)
	18	4, 5, 10, 11	(21, 26, 30), (23, 27, 30), (15, 24, 28), (15, 18, 21)
	19	3, 11	(16, 19, 21), (16, 20, 23)
	20	7, 8, 14, 15	(16, 19, 26), (13, 20, 28), (11, 15, 20), (20, 23, 27)
	21	2, 12, 14	(23, 27, 33), (23, 29, 35), (12, 14, 16)
17	1	10, 11	(29, 46, 59), (30, 44, 53)
	2	5, 10	(12, 16, 20), (10, 13, 17)
	3	4	(9, 11, 13)
	4	12, 13	(11, 13, 16), (10, 14, 18)
	5	7, 12	(10, 11, 13), (15, 17, 21)
	6	13	(41, 46, 64)
	7	2, 11	(17, 23, 31), (16, 19, 22)
	8	1, 5, 9	(14, 20, 22), (16, 18, 24), (13, 17, 23)
	9	2, 12	(20, 29, 40), (19, 30, 39)
	10	13, 15	(12, 16, 20), (9, 13, 17)
	11	1	(17, 24, 33)
	12	4, 8	(13, 21, 29), (13, 17, 20)
	13	6	(28, 33, 38)
	14	1, 3, 6	(13, 17, 21), (9, 12, 15), (12, 14, 15)
	15	5, 8	(5, 8, 11), (4, 7, 9)
	16	1, 8	(3, 5, 6), (6, 8, 9)
	17	2	(34, 42, 50)
	18	2, 10	(13, 15, 19), (15, 18, 21)
	19	1, 6	(4, 6, 7), (5, 7, 8)
	20	9, 10, 11	(3, 5, 6), (8, 10, 12), (6, 9, 10)
	21	3, 12	(10, 15, 18), (13, 18, 23)
	22	6, 11, 14	(13, 19, 26), (11, 17, 22), (14, 20, 25)
18	1	3, 8	(5, 8, 11), (11, 13, 17)
	2	5, 6, 8	(14, 16, 20), (8, 12, 16), (10, 13, 15)
	3	2	(19, 21, 28)
	4	1, 5, 10	(8, 13, 16), (12, 16, 21), (14, 18, 22)
	5	9	(13, 17, 24)
	6	5, 8	(40, 46, 64), (42, 47, 65)

(To be continued)

Table A6 Fuzzy processing times of Jobs 16–18.

(Continued)

Job No.	Operation	Alternative machine	Processing time
	7	3, 7, 10	(39, 44, 55), (36, 48, 64), (38, 49, 64)
	8	5, 6, 13	(14, 17, 22), (10, 14, 16), (7, 10, 12)
	9	8, 15	(11, 16, 19), (9, 13, 15)
	10	3, 11, 15	(20, 28, 34), (17, 27, 34), (20, 30, 42)
	11	10, 13	(40, 48, 54), (43, 50, 56)
18	12	5, 13, 15	(25, 31, 40), (25, 32, 43), (25, 36, 41)
	13	3, 6, 9	(27, 30, 38), (23, 28, 36), (20, 26, 35)
	14	2	(8, 11, 12)
	15	1, 14	(10, 16, 18), (12, 18, 22)
	16	4, 15	(14, 18, 24), (16, 19, 24)
	17	3, 10, 14	(28, 36, 40), (22, 32, 41), (23, 35, 48)

nos. 1–4, pp. 303–315, 2012.

[39] R. Wang, W. Ma, M. Tan, G. Wu, L. Wang, D. Gong, and J. Xiong, Preference-inspired coevolutionary algorithm with active diversity strategy for multi-objective multi-modal optimization, *Inf. Sci.*, vol. 546, pp. 1148–1165, 2021.

[40] R. Wang, Q. Zhang, and T. Zhang, Decomposition-based algorithms using Pareto adaptive scalarizing methods, *IEEE Trans. Evol. Comput.*, vol. 20, no. 6, pp. 821–837, 2016.

[41] F. Zhao, S. Di, J. Cao, J. Tang, and Jonrinaldi, A novel cooperative multi-stage hyper-heuristic for combination optimization problems, *Complex Syst. Model. Simul.*, vol. 1, no. 2, pp. 91–108, 2021.

[42] E. D. Jiang, L. Wang, and Z. P. Peng, Solving energy-efficient distributed job shop scheduling via multi-objective evolutionary algorithm with decomposition, *Swarm Evol. Comput.*, vol. 58, p. 100745, 2020.

[43] Z. Pan, D. Lei, and L. Wang, A knowledge-based two-population optimization algorithm for distributed energy-efficient parallel machines scheduling, *IEEE Trans. Cybern.*, vol. 52, no. 6, pp. 5051–5063, 2022.

[44] Q. Liu, X. Li, and L. Gao, A novel MILP model based on the topology of a network graph for process planning in an intelligent manufacturing system, *Engineering*, vol. 7, no. 6, pp. 807–817, 2021.

[45] Q. Liu, X. Li, and L. Gao, Mathematical modeling and a hybrid evolutionary algorithm for process planning, *J. Intell. Manuf.*, vol. 32, no. 3, pp. 781–797, 2021.

[46] J. Xie, X. Li, L. Gao, and L. Gui, A new neighbourhood structure for job shop scheduling problems, *Int. J. Prod. Res.*, vol. 61, no. 7, pp. 2147–2161, 2023.

[47] Q. Liu, X. Li, L. Gao, and Y. Li, A modified genetic algorithm with new encoding and decoding methods for integrated process planning and scheduling problem, *IEEE Trans. Cybern.*, vol. 51, no. 9, pp. 4429–4438, 2021.

[48] C. Zhang, P. Li, Z. Guan, and Y. Rao, A tabu search algorithm with a new neighborhood structure for the job shop scheduling problem, *Comput. Oper. Res.*, vol. 34, no. 11, pp. 3229–3242, 2007.

[49] Y. Li, X. Li, and L. Gao, An effective solution space clipping-based algorithm for large-scale permutation flow shop scheduling problem, *IEEE Trans. Syst. Man Cybern.: Syst.*, vol. 53, no. 1, pp. 635–646, 2023.

[50] J. Fan, Y. Li, J. Xie, C. Zhang, W. Shen, and L. Gao, A hybrid evolutionary algorithm using two solution representations for hybrid flow-shop scheduling problem, *IEEE Trans. Cybern.*, vol. 53, no. 3, pp. 1752–1764, 2023.

[51] Y. K. Kim, K. Park, and J. Ko, A symbiotic evolutionary algorithm for the integration of process planning and job shop scheduling, *Comput. Oper. Res.*, vol. 30, no. 8, pp. 1151–1171, 2003.

Xinyu Li received the PhD degree in industrial engineering from HUST, China in 2009. He is a professor at State Key Laboratory of Intelligent Manufacturing Equipment and Technology, School of Mechanical Science and Engineering, HUST. He has published more than 90 peer-reviewed papers. His research interests include intelligent algorithms, big data, and machine learning, among others.

Cuiyu Wang received the MEng degree in industrial engineering from HUST, China in 2005. She is currently a PhD candidate in industrial engineering at HUST. Her current research interests include optimization algorithms and production scheduling.

Liang Gao received the BEng degree from Xidian University, China in 1996, and the PhD degree from HUST, China in 2002. He is currently a full professor at HUST. He serves as the editor-in-chief of *IET Collaborative Intelligent Manufacturing*. He is also an associate editor of *Swarm and Evolutionary Computation* and *Journal of Industrial and Production Engineering*.

Qihao Liu received the PhD degree in industrial engineering from Huazhong University of Science and Technology (HUST), China in 2022. He is a postdoctoral research associate in industrial engineering at School of Mechanical Science and Engineering, HUST. His research interests include process planning and shop scheduling.