

A novel hybrid estimation of distribution algorithm for solving hybrid flowshop scheduling problem with unrelated parallel machine

SUN Ze-wen(孙泽文), GU Xing-sheng(顾幸生)

Key Laboratory of Advanced Control and Optimization for Chemical Process, Ministry of Education
(East China University of Science and Technology), Shanghai 200237, China

© Central South University Press and Springer-Verlag GmbH Germany 2017

Abstract: The hybrid flow shop scheduling problem with unrelated parallel machine is a typical NP-hard combinatorial optimization problem, and it exists widely in chemical, manufacturing and pharmaceutical industry. In this work, a novel mathematic model for the hybrid flow shop scheduling problem with unrelated parallel machine (HFSPUPM) was proposed. Additionally, an effective hybrid estimation of distribution algorithm was proposed to solve the HFSPUPM, taking advantage of the features in the mathematic model. In the optimization algorithm, a new individual representation method was adopted. The (EDA) structure was used for global search while the teaching learning based optimization (TLBO) strategy was used for local search. Based on the structure of the HFSPUPM, this work presents a series of discrete operations. Simulation results show the effectiveness of the proposed hybrid algorithm compared with other algorithms.

Key words: hybrid estimation of distribution algorithm; teaching learning based optimization strategy; hybrid flow shop; unrelated parallel machine; scheduling

1 Introduction

The hybrid flow shop (HFS) problem is the side effect of the actual production process, existing widely in chemical, manufacturing, pharmaceutical industries. As a typical scheduling problem, even the two-stage HFS problem is strongly NP-hard [1] on minimizing the maximum completion time (makespan). So the multi-stage HFS problem is even more complicated than the two-stage one. In this work, we focus on the hybrid flowshop scheduling problem with unrelated parallel machine (HFSPUPM). Unrelated parallel machine problem means that there are non-identical machines available to each job. To solve the two-stage HFS, MARQUEZ et al [2] studied an HFS problem for which there is only one machine at the second stage and explored the opportunity to use Monte Carlo continuous time simulation modelling to improve preventive maintenance scheduling in semiconductor fabs. In 2003, LIN et al [3] studied the two-stage HFS problem with setup time and dedicated machines. A recent short review, provided by FANJUL-PEYRO et al [4] focused on the unrelated parallel machines scheduling problem in 2010. CHAUDHRY et al [5] proposed a spreadsheet based genetic algorithm (GA) for flexible manufacturing system(FMS). Many heuristics and meta-heuristics were

developed to search for optimal or near-optimal solutions to the NP-hard problem. WANG et al [6] proposed an effective bi-population based estimation of distribution algorithm for solving the flexible job-shop scheduling problem with the criterion to minimize the makespan. LI et al [7] proposed a hybrid variable neighborhood search(VNS) algorithm combining the chemical-reaction optimization (CRO) and estimation of distribution algorithm (EDA) for solving the hybrid flow shop scheduling problem. A detailed encoding and decoding mechanism was developed for the problem. In order to propose an effective artificial bee colony algorithm, PAN et al [8] presented a total of 24 heuristics, which were utilized to generate an initial population with a high level of quality and diversity, for solving HFS problem with the makespan criterion. SATAPATHY et al [9] presented a TLBO based on orthogonal design, combining TLBO algorithm with orthogonal design to make TLBO algorithm converge faster. RAO et al [10] made a comparative study on teaching learning based optimization algorithm on multi-objective unconstrained and constrained functions. XU et al [11] firstly started research work on a teaching–learning-based optimization algorithm for solving the flexible job-shop scheduling problem with fuzzy processing time. They designed a series of special search operators for the teaching, learning and studying phases.

Foundation item: Projects(61573144, 61773165, 61673175, 61174040) supported by the National Natural Science Foundation of China; Project(222201717006) supported by the Fundamental Research Funds for the Central Universities, China

Received date: 2015–12–30; **Accepted date:** 2016–02–01

Corresponding author: GU Xing-sheng, Professor, PhD; Tel: +86–21–64253463; E-mail: xsgu@ecust.edu.cn

Inspired by the success of the intelligent optimization algorithm in solving the hybrid flowshop scheduling problem, we propose a novel mathematic model of hybrid flowshop scheduling problem with unrelated parallel machine (HFSPUPM). In order to improve the search ability, we design a discrete representation for HFSPUPM in the proposed algorithm. Taking advantage of the global search ability in EDA and the local search ability in TLBO, we propose a hybrid EDA which combines EDA and TLBO for solving HFSPUPM. The simulation results demonstrate the effectiveness and efficiency of the proposed H-EDA in solving this problem.

2 Problem definition and assumption

The hybrid flowshop scheduling problem with unrelated parallel machine can be described as follows. There are n jobs $J=\{1, 2, \dots, i, \dots, n-1, n\}$ that have to be performed on s stages $S=\{1, 2, \dots, j, \dots, s-1, s\}$, and each stage S has $MS[k]$ ($k=1, 2, \dots, s$) unrelated parallel machines. The unrelated parallel machines are continuously available from time zero and have the same effect. And there is at least one stage which has more than one unrelated parallel machines. Each job must visit all of the stage and be processed on one machine. A machine can process at most one job at a time and a job can be processed by at most one machine at a time. The optimization for this scheduling problem is to minimize the makespan by choosing a machine at each stage for each job and determine the sequence of jobs on each machine.

From a structural point of view, the hybrid flowshop scheduling problem with unrelated parallel machine is a combination of hybrid flow shop problem and unrelated parallel machine problem. This complex scheduling problem widely exists in real manufacturing environments, such as chemical, oil, food, tobacco, textile, paper, and pharmaceutical industries. In this work, we give a new model for hybrid flowshop scheduling problem with unrelated parallel machine to minimize makespan.

2.1 Assumption

In order to reflect the characteristics of the problem better and solve it accordingly, we give some assumptions for the hybrid flow shop scheduling problem with unrelated parallel machine as follows:

There are more than one stage for the hybrid flow shop scheduling problem.

At least one stage of the hybrid flow shop scheduling problem has more than one machine.

The machines in each stage are unrelated parallel machines.

Each job is processed through every stage in the same order.

At different stages, machines are continuously available from time zero.

Setup time for jobs is calculated as the processing time.

Each job must be processed in each stage on any unrelated parallel machine.

Jobs' processing will not be interrupted and jobs start the next stage of the processing only after that the previous stage of the processing is carried out.

A machine can process at most one job at a time and a job can be processed by at most one machine at a time. Each unrelated parallel machine processes jobs in order. Preemption, lot streaming, and backlogging are not allowed.

Moreover, the problem formulation uses the notations and parameters, which are shown in the appendix, to describe the whole procedure briefly.

2.2 Mathematical model

The hybrid flow shop scheduling problem is a hotspot among the complex scheduling problems. After scholars paying a lot of time and effort, some preliminary understandings of this issue have been established. But it is still little comment on hybrid flow shop scheduling problem with unrelated parallel machine. Because each stage contains unrelated parallel machines, this problem demonstrates more complexity. Therefore, it is able to better reflect real processing problems. A relatively simple HFS (e.g., a two-stage HFS with limited waiting time) is NP hard in the strong sense [12, 13]. There is no doubt that hybrid flow shop scheduling problem with unrelated parallel machine has at least the same difficulty. In this work, we focus on hybrid flow shop scheduling problem with unrelated parallel machine and present a new model for solving this problem with makespan minimization.

The mathematic model of the hybrid flow shop scheduling problem with unrelated parallel machine can be formulated as follows:

$$\begin{aligned} & \text{Minimize} \quad C_{\max}(\pi) \max_{i=1,2,\dots,n} \{C_{i,k}\} \\ & \quad k = s \text{ (the last stage)} \end{aligned} \quad (1)$$

$$\begin{cases} T_{s\pi(1,1),1} = 0 \\ T_{s\pi(i,1),1} = 0, i = PS_{\partial} \text{ (special job in first stage)} \\ \partial = 2, 3, \dots, MS(k), k = 1 \text{ (the first stage)} \end{cases} \quad (2)$$

$$\begin{cases} C_{i,k} = T_{ei,k} \\ T_{si,k} \geq T_{ei,k-1}, k = 1, 2, \dots, n \\ T_{s\pi(i,k),k} \geq T_{e\pi(i-1,k)k}, j = 1, 2, \dots, MS(k) \\ T_{ei,k} = T_{si,k} + T_{i,j,k}, k = 1, 2, \dots, S \end{cases} \quad (3)$$

where π is the job permutation i.e. the solution to the hybrid flow shop scheduling problem with unrelated parallel machine. $MS(k)$ represents the number of unrelated parallel machines at the stage k ; $T_{si,j}$ represents the start time of the job i at the stage j ; $T_{ei,j}$ represents the end time of the job i at the stage j ; $T_{i,j,k}$ represents the process time of job i processed on machine j at the stage k ; the goal is to get the max completion time.

Equation (1) describes the objective function which minimizes the makespan $C_{\max}(\pi)$. Equation (2) describes the special processing information in the first stage. Equation (3) gives the formulation to calculate makespan $C_{\max}(\pi)$. Time constraints of the processing jobs are also detail described.

Equation (1) describes the objective function which minimizes the makespan $C_{\max}(\pi)$. Following the above recursive Eqs. (2) and (3), the authors firstly calculate the completion time of jobs at the stage one, then that of the stage two, until the last stage.

To illustrate the model and the issues described above, here is an example of processing matrix for understanding its structure. The following example contains 5 jobs and 3 stages. There are three unrelated parallel machines at stage 1, two unrelated parallel machines at stage 2 and three unrelated parallel machines at stage 3.

$$T_{i,j,1} = \begin{bmatrix} 2 & 9 & 5 \\ 10 & 1 & 10 \\ 1 & 4 & 2 \\ 8 & 3 & 3 \\ 9 & 9 & 2 \end{bmatrix}; T_{i,j,2} = \begin{bmatrix} 2 & 9 \\ 9 & 7 \\ 6 & 4 \\ 6 & 6 \\ 2 & 5 \end{bmatrix}; T_{i,j,3} = \begin{bmatrix} 1 & 5 & 5 \\ 3 & 1 & 4 \\ 2 & 10 & 10 \\ 2 & 10 & 4 \\ 3 & 5 & 2 \end{bmatrix}$$

3 Hybrid estimation of distribution algorithm

The focus of this work is the hybrid flowshop scheduling problem with unrelated parallel machine (HFSPUPM). To our best knowledge, this is the first work of hybrid estimation of distribution algorithm for the HFSPUPM. The EDA has features in using probability matrix to demonstrate the solution space. The TLBO has features of simple computation and rapid convergence. Considering these features, we propose an algorithm which combines the structure of EDA with the strategies of TLBO for solving this problem. Teaching–learning-based optimization (TLBO) is a kind of newly introduced meta-heuristics [10, 14–16]. It is a population based optimization algorithm inspired by passing on knowledge within a classroom environment, where learners first acquire knowledge from a teacher (i.e., teacher phase) and then from classmates (i.e., learner phase). There are lots of research demonstrating that TLBO outperforming ES, PSO, ABC and DE on a

number of constrained benchmark functions and constrained mechanical design problems, as well as on continuous non-linear numerical optimization problems [9, 10, 14–16]. We introduce the teaching learning based optimization strategies into EDA to improve the efficiency of the individual optimization. Our algorithm combines estimation of distribution algorithm's global search ability to keep diversity with the teaching learning based optimization algorithm's local search ability to explore better solution. By this way, proposed algorithm greatly improves the search efficiency and effectiveness. In the following, we will discuss details of the proposed estimation of distribution algorithm for hybrid flowshop scheduling problem with unrelated parallel machine.

3.1 Estimation of distribution algorithm

Estimation of distribution algorithms is proposed firstly by MUHLENBEIN et al [17] in 1996. As a novel randomly optimization algorithm, the estimation of distribution algorithms has a lot of advantages. Firstly, the estimation of distribution algorithms has much difference with traditional evolutionary algorithms when taking the mathematical model into consideration. Therefore, it is based on a mathematical model of the entire population and describes the evolutionary trends of the entire group. So it is just a way of modeling on the macro-level. Secondly, the estimation of distribution algorithms provides a new tool for solving complex optimization problems.

EDA is an optimization algorithm which through probability model acquires the excellent solution of the problem based on a probabilistic model. The probabilistic model demonstrates the distribution of the solution to the problem. Depending on different methods of sampling the probability model, EDA develops many measures relying on the characteristic of the problem. But it can mainly be summarized as the following two important steps. The first step is to build a probability model to describe the solution space. Through the assessment of the population, we can choose a collection of outstanding individuals. Then, the statistical learning methods are used to construct a probability model to describe the current solution set. Secondly, the model is sampled to generate the new population. Mostly, using the Monte Carlo method to sample the model for getting new population, the algorithm structure is shown in Fig. 1.

3.2 Teaching–learning-based optimization algorithm

The teaching–learning-based optimization (TLBO) algorithm is a population-based algorithm, proposed by RAO et al in 2011 [14]. The main idea behind TLBO is the simulation of the influence of a teacher on output of

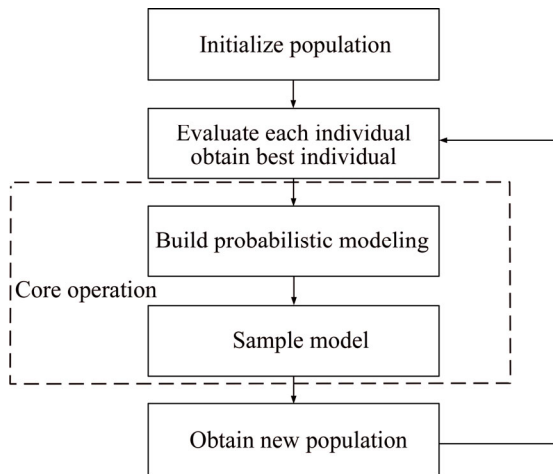


Fig. 1 Structure of estimation of distribution algorithms

learners in a class. So the core operation of the algorithm consists of two stages i.e. teaching phase and learning phase.

From the introduction in the former paragraph, we can clearly see that the teacher phase and learning phase are the main parts of the algorithm. During the teacher phase, the teaching role is assigned to the best individual (X_{teacher}). The algorithm attempts to improve other individuals (X_i) by moving their positions towards the position of the X_{teacher} by taking into account the current mean value of the individuals (X_{mean}). This is constructed using the mean-values for each parameter within the problem space (dimension), and represents the qualities of all students from the current generation. Equation (4) simulates how student improvement may be influenced by the difference between the teacher's knowledge and the qualities of all students. For keeping the algorithm with a certain randomness, two randomly-generated parameters are applied within the equation: r ranges between 0 and 1; and TF is a teaching factor, which can be either 1 or 2, thus emphasizing the importance of student quality.

$$x_{\text{new}} = x_{\text{old}} + \text{rand}() \cdot (x_{\text{teacher}} - T_F \cdot M_k) \quad (4)$$

During the learner phase, student (X_i) starts to acquire knowledge by peer learning from another random selected student (X_j), where j is unequal to i . If X_j is better than X_i , X_i is moved towards X_j (Eq. (5)). Otherwise, it is moved away from X_j (Eq. (6)). If student X_{new} performs better by following Eq. (5) or (6), it will be accepted into the population. The algorithm will continue its iterations until reaching the termination conditions or the maximum iterations.

$$x_{\text{new}} = x_i + r_i(x_j - x_i) \quad (5)$$

$$x_{\text{new}} = x_i + r_i(x_i - x_j) \quad (6)$$

The TLBO technique is a new, simple

meta-heuristic algorithm that has been successfully used in solving combinatorial optimization problems. It has been proven to be superior to other advanced optimization techniques. The evolutionary algorithms are applied to many engineering optimization problems and proven effective in solving specific types of problems. The structure for the implementation of TLBO algorithm is summarized in Fig. 2.

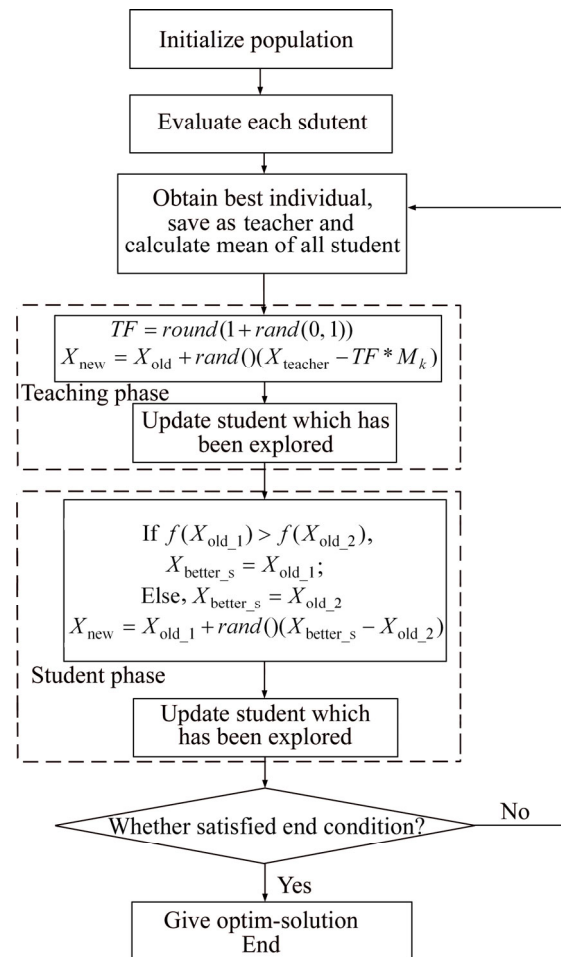


Fig. 2 Structure of teaching learning based optimization

3.3 Hybrid estimation of distribution algorithm for hybrid flowshop scheduling problem with unrelated parallel machine

The main characteristic of the EDA is its explicit probability model, which is employed to perform optimization procedure [18]. It is an optimized algorithm which through probability model approaches the problem of excellent solution based on a probabilistic model to describe the problem. Depending on the complexity and the different methods of sampling probability models, the estimation of distribution algorithms develops many implementations relying on the characteristic of the problem. To trace a more promising searching area, the probability model is adjusted by information of some superior solutions in the population. In such an iterative way, it evolves and obtains satisfactory solutions. By this

means, the EDA is easy to search more breadth suitable place in the global scope. Teaching learning based optimization seems to have some potential for solving non-linear programming and engineering design optimization problems. So we decide to propose a hybrid efficient algorithm which mixes the TLBO technique with EDA. To better solve the hybrid flowshop scheduling problem with unrelated parallel machine (HFSPUPM), we design a number of operations for the problem in the hybrid estimation of distribution algorithm. The details of each step in the proposed hybrid algorithm are given.

3.3.1 Individual representation

As the original estimation of distribution algorithm and teaching and learning based algorithm are used for solving continuous problem, it is not ideal to be used for discrete scheduling problem. In order to solve the problem better, we design a discrete representation for HFSPUPM in the proposed algorithm.

For decoding a given solution into a feasible schedule, jobs are arranged to their machine at each stage in the order of the job priority vector. In order to distinguish each machine and each stage, we use a series of 0, -1 as an identifier. 0 is the unique identifier of the processing machine end; -1 is the unique identifier of the processing stage end. There is an example showed in Eq. (7), where π demonstrates a schedule solution which contains five jobs and three stages. There are three machines in the first stage, one machine in the second stage and two machines in the third stage. As the 0 and -1 are identifiers, it represents that job J_1 is processed on the first machine; jobs J_5 and J_2 are processed on the second machine in turn while jobs J_4 and J_3 are processed on the third machine in turn at the same first stage. In the same principle, job numbers J_3, J_5, J_2, J_1, J_4 are processed in turn on the only one machine at the second stage. Jobs J_2 and J_5 are processed on the first machine in turn while job number J_1, J_3 and J_4 are processed on the second machine in turn at third stage. It is easy to find that this representation method of a feasible solution meets the demand of mantling the solution space with uniqueness and comprehensive.

$$\pi = [J_1 \ 0 \ J_5 \ J_2 \ 0 \ J_4 \ J_3 \ -1 \ J_3 \ J_5 \ J_2 \ J_1 \ J_4 \ -1 \ -1 \ -1 \ J_2 \ J_5 \ 0 \ J_1 \ J_3 \ J_4 \ -1 \ -1] \quad (7)$$

3.3.2 Population initialization and probabilistic model

In order to make the initial population of individuals uniformly scattered in the solution space, the initialization naturally uses a simple random initialization method. To guarantee the initial population with a certain quality, a few individuals are established according to the little process time on each machine at each stage. We use a parameter N_{sp} to record the number of special individuals. The parameter N_{sp} is

from 1 to 3. As a special individual containing some knowledge of the process matrix, it will be well-performing for the problem.

As a crucial part of the hybrid EDA, probability model describes the distribution of the searching space. Generally, the probability model is built based on characteristics of superior solutions. Then, the EDA generates new solutions by sampling according to the probability model. Therefore, a proper probability model and sampling strategy is critical to the performance of hybrid EDA. Considering the structure of hybrid flow shop scheduling problem with unrelated parallel machine, the probability model is designed as P_s ($s=1, 2, \dots, S$) for the s -th stage. In this work, the optimization objective is to minimize makespan for hybrid flow shop scheduling problem with unrelated parallel machine. P_s is related to the job priority in the s -th stage.

$$P_s = \begin{bmatrix} p_{11}(l) & p_{12}(l) & \cdots & p_{1n}(l) \\ p_{21}(l) & p_{22}(l) & \cdots & p_{2n}(l) \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1}(l) & p_{n2}(l) & \cdots & p_{nn}(l) \end{bmatrix} \quad (8)$$

Element $p_{ij}(l)$ in the probability matrix P_s represents the probability that job J_j appears before or in the i -th position at s -th stage of the job priority vector at the l -th iteration. The value of $p_{ij}(l)$ refers to the importance of a job when decoding a solution into a scheme. At the initialization stage of the hybrid algorithm, the elements in matrix are initialized as $p_{ij}(l)=1/n$ (for all i and j), which implies a uniform distribution in the searching space [6]. Choose a number at random $[0, 1]$ and compare it with the element in the probability matrix to decide the job permutation. Insert the identifier 0 randomly into the permutation. Replenish identifier -1 at the end of each stage.

3.3.3 Teaching learning based optimization strategy for population updating

The teaching learning based algorithm basically mimics the teaching-learning ability of teacher and learners in a classroom. Teacher phase and learner phase are core operation during the whole algorithm. We use this strategy to optimize the individuals by iteration.

During the teacher phase, the teaching role is assigned to the best individual ($X_{teacher}$). $X_{teacher}$ represents a best solution. We choose at random a stage of the $X_{teacher}$ to replace the same stage of student (X_i). Then student X_{new} is generated. If student X_{new} performs better, we replace the individual student (X_i) with X_{new} . Figure 3 shows the details to generate X_{new} . Therefore, the new generated individual will achieve randomly a piece of excellent information from the teacher.

During the learner phase, each individual acquires knowledge from others as a student $X_{student}$. In order to improve the effectiveness of the hybrid EDA, we

optimize knowledge acquisition situation from other individuals and add the self-study section at this stage. We choose at random a stage of another student X_i to replace the same stage of student (X_{student}). In this way, we will acquire the X_{new} . Figure 4 shows the details to generate X_{new} . If student X_{new} performs better, there is no doubt that we replace X_{student} with X_{new} . Otherwise, we replace randomly X_{student} with X_{new} . If not a constant number, the acceptance probability can be also adaptive according to the iteration and population diversity.

There is a lot of information gap between the individual permutation and process matrixes. It is worth designing more operations. We also can change the individuals with the different stage or a part of jobs in another one. But it needs another operation for individual legalization.

3.3.4 Local search method and updating mechanism

While a new neighborhood solution is being generated, a local search approach, job-based method and machine-based method are implemented to improve it by iteratively exchanging the positions of jobs in the solution with those of jobs positioned after them. Notably, the local search strategy is searching for the better solution from the origin one's neighborhoods. Machine-based method also gives more opportunities to redistribution process jobs.

After a series of operations, we update the individuals in the population. Then we choose a few better individuals among the individuals in the population to update the probability model P_s . Eqs. (9) and (10) describe the details to update the probability model P_s . Following the knowledge of the better individuals, the probability model is toward to the space which contains better solutions.

$$p_{ij}(l+1) = (1-\alpha)p_{ij}(l) + \frac{\alpha}{i \times N_{\text{better}}} \sum_{a=1}^{N_{\text{better}}} I_{ij}^a \quad (9)$$

$$I_{ij} = \begin{cases} 1, & \text{if job } j \text{ appears before or in position } i \\ 0, & \text{else} \end{cases} \quad (10)$$

where $\alpha \in (0, 1)$ is the learning rate from the new better individuals; I_{ij} describes whether the job j is located before the position i . The probability model is adjusted to represent more effective searching region. As a consequence, an updating mechanism is employed to adjust the model at each generation. This is also a updating mechanism for getting new individuals during the iteration.

3.3.5 Diversity and rebuild probabilistic model

A majority of intelligent optimization algorithms will be faced with the challenge to escape from local minima after a large number of computations. Diversity of the evolutionary population is also an essential role in the algorithm. In the proposed hybrid estimation of distribution algorithm, we bring diversity under control to acquire a better solution for hybrid flow shop scheduling problem with unrelated parallel machine. This method is effective to abandon the local optimize position. During each 20 times iteration of the proposed algorithm, it continuously calculates the difference value between the best individual and worst individual for three iterations. If the difference value is small enough and stays unchanged, the proposed hybrid algorithm assumes that the diversity of the population is damaged to a certain extent. In order to expand the search breadth, we generate some new individuals with partly reserving the best individuals. The probability matrixes are initialed at the same time. According to the information

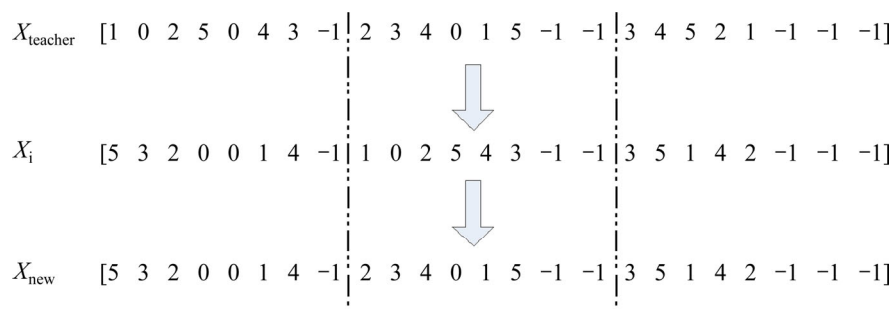


Fig. 3 Steps to generate X_{new} during teacher phase

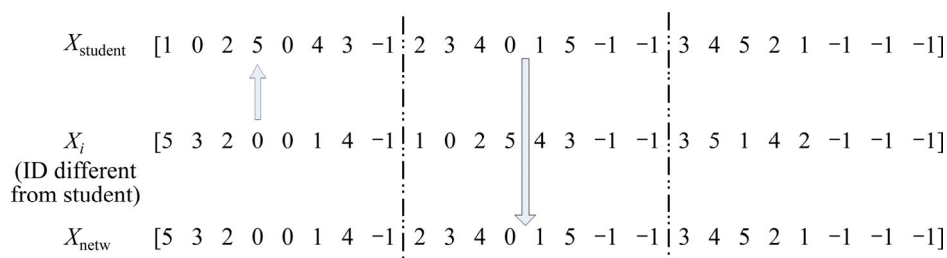


Fig. 4 Steps to generate X_{new} during learner phase

in the newly merged individual, we update the population. At the same time, we do some operations to improve the probability of generating a new matrix. Under the circumstance of reserving the better iteration information, this method keeps the population well diversified.

3.3.6 Procedures of H-EDA

Figure 5 presents the flow chart algorithm of the proposed hybrid estimation of distribution algorithm. First of all, the initial population is randomly generated as the majority of meta-heuristic algorithms. Then, generate N_{sp} special individuals containing some knowledge of the process matrix. After initializing each individual in the population, we set the iteration symbol iter to 1. The proposed H-EDA carries out each subsection which is described computational procedure circularly in turn. When the termination criterion is satisfied, the algorithm is ended. Following the termination of the proposed H-EDA procedures, the best global optimal solution is obtained to the hybrid flow shop scheduling problem with unrelated parallel machine. To the whole updating process, it has a roughly analyzed computational complexity of $O(G_{\max} * S_{\text{pop}})$.

4 Simulations

In this section, we conduct extensive computational experiments to compare the hybrid estimation of distribution algorithm with several other algorithms for hybrid flowshop scheduling problem with unrelated parallel machine by using the generated instances. Based on LIAO's [19] benchmark problems, this work generates a series of processing time matrix relying on flowshop scheduling with unrelated parallel machine problem. The generated processing time matrix largely preserves the original features of Liao's benchmark problems. Through comparisons, the results demonstrate the quality of the proposed hybrid algorithm based on EDA framework.

The proposed hybrid estimation of distribution algorithm is coded in C++(Visual Studio 2012) and run all the algorithms on an Inter(R) Core(TM) i7-2600 CPU 3.40 GHz with 2.85 GB available main RAM. The following parts describe the exhaustive concept of the parameter settings and computational results.

4.1 Date generation

Although there is a lot of research for hybrid flow shop scheduling problem (identical machines are continuously available from time zero and have the same effect at each stage), it is still hard to find the benchmarks for the hybrid flowshop scheduling problem with unrelated parallel machine. In order to better test

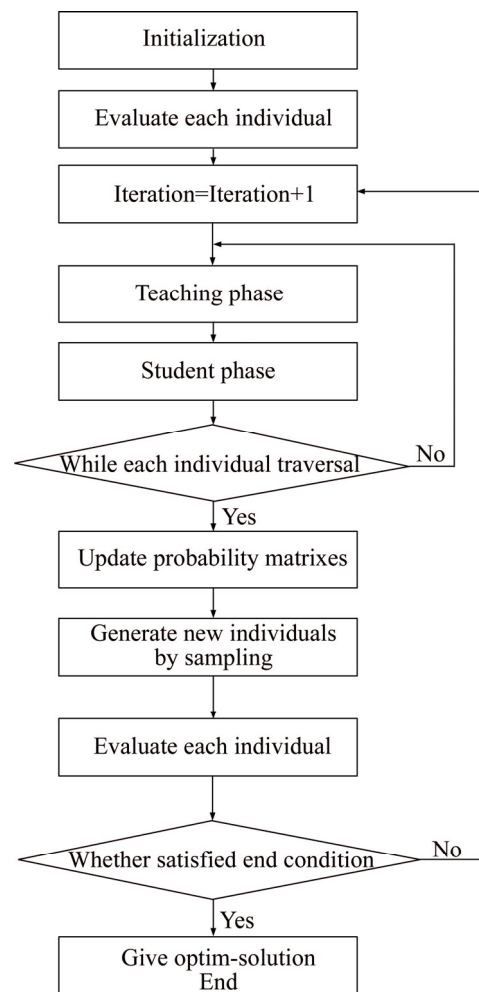


Fig. 5 Flow chart of algorithm of proposed hybrid estimation of distribution algorithm

whether the proposed algorithm can solve the hybrid flowshop scheduling problem with unrelated parallel machine, we carry out some expansion in the existing benchmarks. Combining the existing benchmarks with random generation, we generate a series of benchmarks for further study and analysis. The generated benchmarks can well meet the demand of a series of factors well.

In the late years, LIN et al [3] and LIAO et al [19] changed the job number from 15 to 30 to generate harder problems. During their research, machines are arranged into 5 stages cascade in each problem. At each stage, the machine number increases from 2 or 3 to a random number determined by $U(3,5)$. The range of processing time distribution is expanded from $U(3,20)$ to $U(1,100)$. For LIAO's benchmark problems, many researchers design meta-heuristic algorithms to solve hybrid flowshop scheduling problem. Therefore, there is much recognition to the LIAO's benchmark problems. This work generates the date based on the LIAO's benchmark problems. In the case of maintaining the machine number in each stage with the Liao's benchmark problems, we generate a series of process time matrixes for hybrid

flowshop scheduling problem with unrelated parallel machine. The generated benchmarks' factors and their levels are shown in Table 1. The following simulation results would demonstrate the efficiency and effectiveness of the proposed hybrid EDA.

Table 1 Factors and corresponding levels for generated benchmarks

Factor	Level
Number of jobs	30
Number of stages	5
Number of machines in each stage	$U(3,5)$
Processing time on each machine	$U(1,100)$

4.2 Parameter settings

In our simulation, the proposed hybrid estimation of distribution algorithm has a series of parameters which will affect the performance for solving the hybrid flowshop scheduling problem with unrelated parallel machines. In order to achieve better parameters, we select a benchmark problem as a test to guide the options of the parameters. To design the five parameters in the proposed estimation of distribution algorithm, we conduct a detailed experiment based on the Taguchi method of design of experiment (DOE)[6]. The factors and levels of the five significant parameters are given in Table 2. The S_{pop} represents the size of population. The N_{sp} represents the number of special generated individuals. The $P_{student}$ represents the probability to accept a worse X_{new} . The α represents the learning rates of achieving knowledge from a few better individuals. The G_{max} represents the number of max iterations to end the proposed hybrid algorithm. In Table 2, three levels for each parameter are given. That means that there are $3^5=243$ ways for parameter permutation. We independently calculate the benchmark “Thj30c5e1” 20 times for each permutation and employ the relative percentage deviation (RPD, D_{rp}) to describe the factor

Table 2 Factors and corresponding levels for guiding parameter settings

Factor	Level		
	1	2	3
S_{pop}	10	20	30
N_{sp}	1	2	3
$P_{student}$	0.02	0.06	0.10
α	0.2	0.3	0.4
G_{max}	50	100	150

level trend for the 5 parameters. As an effective parametric analysis method, RPD is calculated as follows:

$$D_{rp} = \frac{C_{avg} - C_{max}^*}{C_{max}^*} \times 100\% \quad (11)$$

where C_{avg} is the average value of makespan during the whole computation; C_{max}^* represents the best makespan acquired. The less of the RPD value, the better the computational result.

Figure 6 shows the computational results RPD as a map of makespan value. It shows the trend of parameter changing which helps to analyze the significance rank of each parameter. If there is no significant impact on the algorithm where the parameters change in a finitude range, we would consider some parameters more about its computation cost, such as Pop_size and max_generation. Therefore, authors finally set the parameters in proposed hybrid estimation of distribution as: $S_{pop}=10$, $N_{sp}=2$, $P_{student}=0.1$, $\alpha=0.3$, and $G_{max}=100$.

4.3 Performance assessment

We present the results of our computational experiments. In order to evaluate the performance of the proposed hybrid estimation of distribution algorithm for solving the hybrid flowshop scheduling problem with unrelated parallel machine, the proposed hybrid algorithm is compared with GA [20], ACO [21], TLBO

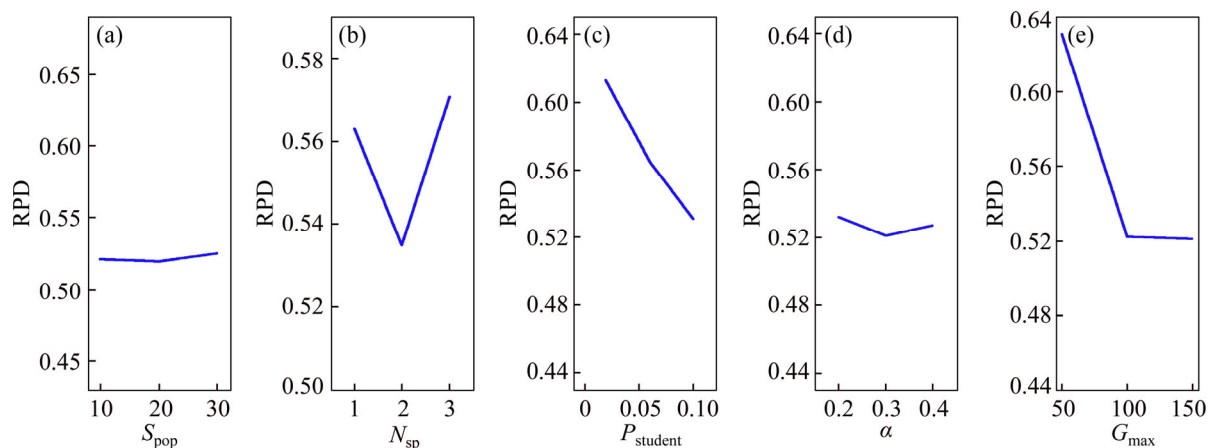


Fig. 6 Different factor level trends in RPD

[22]. Table 3 shows the computational results. Each of the algorithm is run 10 times independently for each problem. In Table 4, the RPD described in the previous section, is also used to count relative deviation between the algorithms' solutions and obtained best solution. There is another parameter, i.e. STD which indicates the value of standard deviation.

From Table 4, it is noted that the proposed hybrid estimation of distribution algorithm (H-EDA) acquires the best makespan in the compared algorithms. At the same time, most of H-EDA's ARPD values which are equal to the makespan in some factor, are less than other compared algorithm. It means that H-EDA can obtain a better solution than other algorithms in solving hybrid flow shop scheduling problem with unrelated parallel machines.

In addition, Fig. 7 shows the convergence curves of all the algorithms for Thj30c5e1. It is clear that the proposed H-EDA algorithm outperforms other three algorithms for the hybrid flowshop scheduling problem with unrelated parallel machines to minimize the

makespan. In a word, the simulation results display the effectiveness and efficiency of the proposed H-EDA in solving hybrid flowshop scheduling problem with unrelated parallel machine.

5 Conclusions

A novel mathematic model of hybrid flowshop scheduling problem is proposed with unrelated parallel machine (HFSPUPM). The H-EDA is developed by combining the global search ability in EDA and the local search ability in TLBO for solving this problem. A new type of vector encoding method is designed by catching the characteristics of solutions for HFSPUPM. In order to solve the scheduling problem, the discrete method is used to improve the efficiency in the H-EDA. In addition, an effect parameter setting method is employed based on the Taguchi method of DOE. The simulation results show that the proposed H-EDA is more effective and efficient in solving HFSPUPM with makespan minimization by comparing with other algorithms.

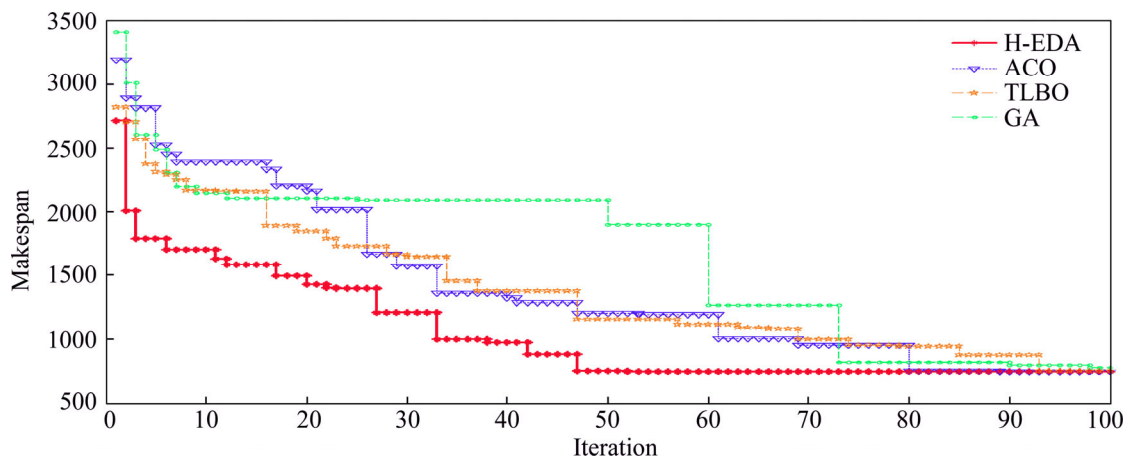


Fig. 7 Convergence curves of instance Thj30c5e1

Table 3 Comparison results of ten benchmark problems

Problem	GA			ACO			TLBO			H-EDA		
	Ave	Min	Std	Ave	Min	Std	Ave	Min	Std	Ave	Min	Std
Thj30c5e1	1.78	1.07	0.50	0.71	0.40	0.25	0.60	0.27	0.14	0.24	0.00	0.21
Thj30c5e2	1.21	0.63	0.29	0.74	0.52	0.14	0.61	0.31	0.10	0.23	0.00	0.12
Thj30c5e3	2.45	1.66	0.32	0.83	0.59	0.17	0.61	0.36	0.19	0.19	0.00	0.17
Thj30c5e4	1.37	0.73	0.31	0.63	0.36	0.22	0.58	0.36	0.18	0.21	0.00	0.13
Thj30c5e5	1.19	0.56	0.45	0.16	0.00	0.15	0.42	0.22	0.12	0.26	0.00	0.13
Thj30c5e6	2.27	1.50	0.37	0.70	0.50	0.17	0.54	0.25	0.16	0.17	0.00	0.16
Thj30c5e7	1.73	0.77	0.31	0.53	0.26	0.16	0.35	0.13	0.16	0.35	0.00	0.17
Thj30c5e8	1.29	0.48	0.28	0.93	0.48	0.11	0.58	0.36	0.14	0.23	0.00	0.17
Thj30c5e9	1.42	0.72	0.25	0.42	0.10	0.16	0.21	0.00	0.13	0.10	0.00	0.11
Thj30c5e10	0.91	0.25	0.34	0.34	0.00	0.21	0.68	0.38	0.14	0.16	0.00	0.19
Average	1.562	0.837	0.342	0.599	0.321	0.174	0.518	0.264	0.146	0.214	0.00	0.156

Note: Ave means average. Min means minimum and Std means standard.

Appendix:

i, j, k	Normally used as loop variables
n	Number of products
N	Whole number of products
m	Number of unrelated parallel machines
S	Number of max stage in hybrid flow shop scheduling problem
$MS(k)$	Number of unrelated parallel machines at the stage k
π	A designed structure i.e. the solution to the hybrid flow shop scheduling problem with unrelated parallel machine.
$T_{s_{ij}}$	Start time of the job i at the stage j
$T_{e_{ij}}$	End time of the job i at the stage j
$T_{i,j,k}$	Process time of job i processed on machine j at the stage k
P_h, T_h	Process time matrix, i.e. the set of $T_{i,j,k}$
C_{\max}	Optimization target, the makespan for hybrid flow shop scheduling problem with unrelated parallel machine
$C_{i,S}$	End time of job i at the last stage S
PS_{δ}, δ	First processed job on each unrelated parallel machine

References

- [1] GUPTA J N D. Two-stage hybrid flowshop scheduling problem [J]. Journal of the Operational Research Society, 1998, 39: 359–364.
- [2] MARQUEZ A C, GUPTA J N D, IGNIZIO J P. Improving preventive maintenance scheduling in semiconductor fabrication facilities [J]. Production Planning & Control: The Management of Operations, 2006, 17(7): 742–754.
- [3] LIN H T, LIAO C J. A case study in a two-stage hybrid flow shop with setup time and dedicated machines [J]. International Journal of Production Economics, 2003, 86(2): 133–143.
- [4] FANJUL-PEYRO L, RUIZ R. Iterated greedy local search methods for unrelated parallel machine scheduling [J]. European Journal of Operational Research, 2010, 207(1): 55–69.
- [5] CHAUDHRY I A, MAHMOOD S, SHAMI M. Simultaneous scheduling of machines and automated guided vehicles in flexible manufacturing systems using genetic algorithms [J]. Journal of Central South University, 2011, 18(5): 1473–1486.
- [6] WANG L, WANG S Y, XU Y, ZHOU G, LIU M. A bi-population based estimation of distribution algorithm for the flexible job-shop scheduling problem [J]. Computers & Industrial Engineering, 2012, 62(4): 917–926.
- [7] LI J Q, PAN Q K, WANG F T. A hybrid variable neighborhood search for solving the hybrid flow shop scheduling problem [J]. Applied Soft Computing, 2014, 24(11): 63–77.
- [8] PAN Q K, WANG L, LI J Q. A novel discrete artificial bee colony algorithm for the hybrid flowshop scheduling problem with makespan minimization [J]. Omega, 2014, 45(6): 42–56.
- [9] SATAPATHY S C, NAIK A. Cooperative teaching–learning based optimisation for global function optimization [J]. International Journal of Applied Research on Information Technology and Computing, 2013, 4(1): 1–17.
- [10] RAO R V, WAGHMARE G G. A comparative study of a teaching–learning-based optimization algorithm on multi-objective unconstrained and constrained functions [J]. Journal of King Saud University-Computer and Information Sciences, 2014, 26(3): 332–346.
- [11] XU Y, WANG L, WANG S Y, LIU M. An effective teaching-learning-based optimization algorithm for the flexible job-shop scheduling problem with fuzzy processing time [J]. Neurocomputing, 2015, 148(1): 260–268.
- [12] JUN S B, PARK J W. A hybrid genetic algorithm for the hybrid flow shop scheduling problem with nighttime work and simultaneous work constraints: A case study from the transformer industry [J]. Expert Systems with Applications, 2015, 42(15, 16): 6196–6204.
- [13] PAN Q K, WANG L, MAO K, ZHAO J H, ZHANG M. An effective artificial bee colony algorithm for a real-world hybrid flowshop problem in steelmaking process [J]. IEEE Transactions on Automation Science and Engineering, 2013, 10(2): 307–322.
- [14] RAO R V, SAVSANI V J, VAKHARIA D P. Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems [J]. Computer Aided Design, 2011, 43(3): 303–315.
- [15] BAYKASOĞLU A, HAMZADAYI A, KÖSE S Y. Testing the performance of teaching–learning based optimization (TLBO) algorithm on combinatorial problems: Flow shop and job shop scheduling cases [J]. Information Sciences, 2014, 276(8): 204–218.
- [16] BANERJEE S, MAITY D, CHANDA C K. Teaching learning based optimization for economic load dispatch problem considering valve point loading effect [J]. International Journal of Electrical Power and Energy Systems, 2015, 73(12): 456–464.
- [17] MUHLENBEIN H, PAASS G. From recombination of genes to the estimation of distributions I: Binary parameters [J]. Lecture Notes in Computer Science, 1996, 1141(1): 178–187.
- [18] WAN Z P, MAO L J, WANG G M. Estimation of distribution algorithm for a class of non-linear bilevel programming problems [J]. Information Sciences, 2014, 256(20): 184–196.
- [19] LIAO C J, TJANDRADAJA E, CHUNG T P. An approach using particle swarm optimization and bottleneck heuristic to solve hybrid flow shop scheduling problem [J]. Applied Soft Computing, 2012, 12(6): 1755–1764.
- [20] KAHRAMAN C, ENGIN O, KAYA I, YILMAZ M K. An application of effective genetic algorithms for solving hybrid flow shop scheduling problems [J]. International Journal of Computational Intelligence Systems, 2008, 1(2): 134–147.
- [21] ALAYKÝRAN K, ENGIN O, DÖYEN A. Using ant colony optimization to solve hybrid flow shop scheduling problems [J]. International Journal of Advanced Manufacturing Technology, 2007, 35(5, 6): 541–550.
- [22] NIKNAM T, AZIZIPANAH-ABARGHOOEE R, RASOUL-NARIMANI M. A new multi objective optimization approach based on TLBO for location of automatic voltage regulators in distribution systems [J]. Engineering Applications of Artificial Intelligence, 2012, 25(8): 1577–1588.

(Edited by FANG Jing-hua)

Cite this article as: SUN Ze-wen, GU Xing-sheng. A novel hybrid estimation of distribution algorithm for solving hybrid flowshop scheduling problem with unrelated parallel machine [J]. Journal of Central South University, 2017, 24(8): 1779–1788. DOI: <https://doi.org/10.1007/s11771-017-3586-6>.