

# **An Annealing Memetic Algorithm for Instructional Scheduling**

<https://doi.org/10.3991/ijet.v13i11.9601>

Shen-min Lv

Shandong Women's Univesity, Shandong, China  
shenminlv3889@21cn.com

**Abstract**—In order to solve the problems of large-scale instructional scheduling, many constraints, and complex interrelation, it is studied based on the changes in the scheduling problem and the adjustment of related algorithms. Firstly, the problem of instructional scheduling is described as a whole, and the relationship between each stage, in particular the impact of the combined plan on scheduling is also described, and an optimization plan is proposed. Then the optimal scheduling scheme is solved with Memetic algorithm. According to the constraints, the fitness value function in the algorithm is designed flexibly. Finally, the convergence of the algorithm is proved based on the Banach compression mapping theorem. Practice has proved that: Memetic algorithm can effectively solve the problem of inadequate teaching resources and difficult of instructional scheduling. Therefore, the Memntic algorithm can be used in instructional scheduling.

**Keywords**—instructional scheduling, Memetic algorithm, scheduling program

## **1 Introduction**

The traditional description of the teaching scheduling problem mainly refers to the class arrangement before the class, that is, the class, the teacher, and the course to be attended are arranged in the classroom that meets the requirements. The problem of curriculum arrangement is the link between the past and the future in teaching, and it is also a very important part; a more reasonable schedule arrangement will make more efficient use of campus resources, which will optimize the allocation of resources. In the problem of curriculum arrangement in universities, the number of courses, classes and teachers in universities is much larger than that of primary and secondary schools, and there are many types of courses, and the teaching mode is more complex. All of these have caused a lot of trouble to solve this problem, and it is difficult to find a satisfactory solution to the problem with the traditional analytical method. For the scheduling problem, in addition to the early adopting graph theory, integer programming, constraint-based methods, heuristic search, expert systems, etc, in recent years, more and more intelligent optimization algorithms have been introduced to solve this problem, such as genetic algorithm, simulated annealing algorithm, tabu search

algorithm, Memetic algorithm, ant colony algorithm, etc. The introduction of these intelligent algorithms has opened up important ideas for solving such problems.

However, the scheduling problem is not the whole problem of teaching scheduling, but an important part of it. The teaching scheduling process includes five parts: the teaching plan stage, the joint class stage, the teaching task stage, the class scheduling stage, and the final course selection stage. These five parts have a relatively close relationship, especially the two parts of the joint class and the class scheduling have a complementary relationship. The results of the joint class will directly affect the scheduling plan, and the constraints of the class scheduling will also place more demands on the joint class. On the other hand, although the joint class and the class scheduling are the two main stages to generate the class schedule, if there is no preparatory work in the early stage, such as the data preparation of the teaching plan and the teaching task, the joint work and the class scheduling work will not be carried out. Therefore, the teaching and scheduling work is regarded as an overall stage and is not only concerned with the course scheduling, which will give the problem a more comprehensive consideration, so that the whole problem will be better solved.

Memetic algorithm adopts a framework and process operation similar to genetic algorithm, and based on this, combined with local search strategy, it successfully inherits the advantages of global search strategy and local search strategy. At the same time, it avoids the disadvantages, so that all individuals in each iteration can achieve local optimum, ensuring that each iteration is very efficient and greatly reducing the number of iterations. Through the genetic algorithm, the population is globally searched, and the excellent areas in the space are explored, and the search breadth is widened. At the same time, the local search strategy is used to perform individual deep search, which avoids the phenomenon that the algorithm falls into local optimum and produces premature, and the search depth is improved. Therefore, the Memetic algorithm not only has strong global optimization ability, but also performs local search after each genetic operation, which greatly improves the solution speed, ensures the convergence performance, and finally obtains the high quality optimal solution.

## **2 Literature review**

Deng et al. (2016) pointed out in the research that the teaching scheduling problem is a typical constraint optimization problem. The goal is to solve the better scheduling scheme under certain constraints, so that the teaching arrangement executed according to this scheme has the characteristics of excellent teaching effect, good resource utilization rate and high teaching satisfaction [1]. Xu et al. (2014) pointed out in the research that the classification of optimization problems can be divided into single-objective optimization problems and multi-objective optimization problems according to the number of problem objective functions; according to whether the problem has constraints, the problem can be divided into unconstrained optimization problem and constrained optimization problem [2]. Gong et al. (2017) pointed out that the research on different types of problems has formed different optimization algorithms, such as the steepest descent method for solving unconstrained optimization problems, Newton

method, conjugate gradient method, etc. The direct difference between different methods is mainly to select the falling direction and the falling point by different methods; the solution of the constrained optimization problem can transform the constrained optimization problem into a series of unconstrained optimization problems, and then call the unconstrained optimization method to solve the problem; the commonly used methods are penalty function method, multiplier method, etc. [3]. Yannibelli et al. (2015) pointed out that the teaching scheduling process has five important stages: teaching plan, joint processing, teaching tasks, scheduling, and class selection. These five phases are a complete process. Except for the first phase and the last phase, the output of each phase is the input of the next phase. At the teaching plan stage, the correspondence between the major and the required course is determined, and the correspondence between the class and the course is obtained according to the class corresponding to the major; after determining the correspondence between the class and the course, the joint class relationship of the classes in the same course will be determined according to the teacher resources and other resources. Combine a number of classes in the same course with several logical classes to improve the efficiency of resource use [4]. Shen et al. (2017) pointed out in the study that class scheduling is the core of teaching scheduling, and the joint class is an important task before the class scheduling. The result of the joint class is the input of the class arrangement, which will have an important impact on the class arrangement; a good result of the joint class will reduce the constraint conflicts in the class arrangement and reduce the number of shared crossovers, which will reduce the complexity of the class scheduling algorithm and increase the utilization of resources such as classrooms [5]. Zhang et al. (2016) pointed out in the research that the Memetic algorithm is a new intelligent algorithm combining genetic algorithm and local search strategy, and it is also one of the "Cultural Evolutionary Computation". Therefore, many people call the Memetic algorithm a hybrid genetic algorithm, a genetic local optimization algorithm, etc. [6]. Azad et al. (2017) pointed out in the research that the Memetic algorithm is not a single algorithm, but an algorithm framework that combines global search strategy with local search strategy. The combination of different search strategies constitutes a different Memetic algorithm. Genetic algorithm is usually used as the global search strategy of Memetic algorithm [7]. Keshavarz (2015) pointed out in the research that the local search strategy can use hill climbing algorithm, simulated annealing algorithm, tabu search algorithm and so on. For different fields and problems, it is possible to flexibly construct a Memetic algorithm suitable for solving this problem, so as to solve some problems that are difficult or even impossible to be solved by traditional methods more efficiently [8].

### **3 Methodology**

#### **3.1 Description of teaching scheduling problems**

Teaching scheduling is the main work in the preparation stage of teaching. The main goal is to arrange all the courses, teachers, and classes to a certain time and classroom through a certain organization, so as to complete a series of more reasonable schedules,

so that the teaching work can be completed smoothly. In the process of arrangement, each element (refers to teachers, courses, classrooms, classes, etc.) has inherent constraints, and there are certain constraints between events and events. It is precisely because of the existence of these constraints and the large number of elements that lead to the larger scale of the problem, which makes the problem more complicated.

The teaching scheduling problem is a timetable problem with a five-tuple,  $F$  (timetable) =  $\{C, L, T, W, R\}$ .  $C, L, T, W,$  and  $R$  represent class collections, course collections, teacher collections, time collections, and classroom collections, respectively. The details are as follows:

Class collections:  $C = \{C_1, C_2, \dots, C_l\}$

Course collections:  $L = \{L_1, L_2, \dots, L_m\}$

Teacher collections:  $T = \{T_1, T_2, \dots, T_n\}$

Time collections:  $W = \{W_1, W_2, \dots, W_p\}$

Classroom collections:  $R = \{R_1, R_2, \dots, R_q\}$

In the above five-tuple group, classes, courses, and teachers first form a class arrangement unit through teaching programs, joint classes, and teaching tasks. Time and classroom are grouped into resources to be allocated. Therefore, the problem of scheduling is also converted into the scheduling unit to the resources to be allocated according to certain constraints, so that the normal teaching work can be successfully completed according to the schedule.

The general solution to the problem of teaching scheduling can be divided into two stages. The first stage aims to solve the feasible solution, and the second stage aims to solve the optimal solution. The main content of modeling this problem is as follows:

The first step is to quantify the constraints.  $V_i(X)$  is used to describe certain types of constraints for certain resources, and  $X$  represents one or more resources. Take the constraint of “more than one course in the weekly schedule is as even as possible” as an example. It assumes that six lessons can be arranged each day (two lessons in the morning, two lessons in the afternoon, two lessons in the evening, each lesson includes two 45-minute sections with a 5-minute break), seven days a week, the total number of courses available is 42. If the weekly schedules in more than one course are as even as possible, the time gap between adjacent classes will be as close as possible. The following steps are used to calculate the penalty function value of each meta course for this constraint condition.

The second step is to build the objective function. Since the infeasible solution will not enter the actual iterative process, the constraints contained in the rigid constraint will not be introduced into the objective function. Once any violation of the rigid constraint occurs, the scheme is directly removed from the iterative process. So the objective function is the sum of a series of flexible constraints after quantization:

$$f(tp_i) = \sum_{i=1}^k w_i \sum v_i(X) \quad (1)$$

In summary, the core of the teaching scheduling problem lies in scheduling classes. The main task of the class scheduling is to combine the courses, classes, teachers, and classrooms in a reasonable way, and arrange them into feasible time films to generate

a class schedule for all courses and all classes. In the process of arrangement, the constraints of the problem need to be considered and the optimal solution in the objective function formula (1) is obtained by the relevant optimization solution algorithm.

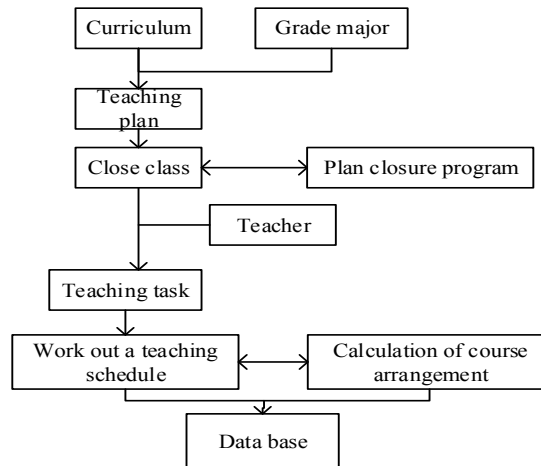


Fig. 1. The structure diagram of the teaching scheduling system

### 3.2 Solution of the teaching scheduling joint class problem based on annealing Memetic algorithm

Course scheduling is a constrained optimization problem, which is to solve the optimal scheduling plan under certain constraints. The goal is to find a reasonable schedule for each class that meets the following rigid constraints: first of all, the teaching plan for each class should be reflected in the class schedule; then, teachers, classrooms, and classes can't appear multiple times in the same time period at the same time. No resource conflicts such as teachers, classrooms, classes, etc. can occur; finally, the classroom capacity is greater than the number of students in the class, and the type of classroom is required to meet the type of course required.

The main goal of the joint class is to combine the classes with shared courses to generate logical classes according to the teaching plan, which provides a good foundation for the class arrangement. The joint class has the following two important data, the first is the teaching plan: the teaching plan is to prepare the courses for each major of the semester before the new semester, which is the beginning and main metadata of the teaching and scheduling work; the second is the number of teachers who can be provided by the starter of each course for the course. According to the number of teachers available, the class to be taught is combined into a corresponding number of logical classes.

It assumes that there are K schemes for the overall joint class, then the formula for calculating K is as follows:

$$K = C_{n1}^{m1} \cdot C_{n2}^{m2} \dots C_{nN}^{Mn} \quad (2)$$

Although the number of schemes for the overall joint class is determinable, the number is very large. At the same time, because the scale of the problem itself is relatively large, it is difficult to calculate the optimal joint plan with the enumeration method. With this in mind, using some intelligent optimization algorithms will be an effective solution.

The basic steps of the algorithm are as follows:

Chromosome coding method: coding is the primary problem to be solved by genetic algorithm. By coding, the solution of the solution space is transformed into a representation of evolutionary computation, that is, from phenotype to genotype. The points in the solution space are transformed into a manipulated form of the genetic algorithm to generate new individuals through subsequent crossover and mutation operations. As a basic element of evolutionary computation, the way chromosomes are encoded will directly affect the efficiency of the algorithm. The mapping of phenotypes to genotypes is called coding. Conversely, when performing fitness value calculations, the genotypes need to be converted to phenotypes. This mapping is called decoding.

Initialized population: after determining the chromosomal representation of the joint class scheme, N initial string structure data will be randomly generated. Each string structure data is called an individual, which results in an initial population.

```

Basic parameters:
Class teaching plan list
The number of all courses M, No course logo for Coursei
The number of classes in class is Ci Number of Teachers is mi
Initialization HashMap:H0 : Initialization t=0
For(i=0;i<m;i++)
  Int Ci it =c m
  Mod Mi ik =c m
  For(j=0;j<m;j++)

    If j<k then C =Selectrandom(t+1); j
    Else C =Selectrandom(t); j
    End if .insertitem( , ); 0HijLCjC

  Next

Next
    
```

**Fig. 2.** Pseudo code that generates a single individual

The above figure depicts the pseudo code that generates a single individual. In Figure 2, the individual is first initialized and then a holistic cycle is performed for all courses. When operating for each course, first divide the number of classes by the number of logical classes and calculate the number of classes to be included in each logical class generated by the course. Then the remaining classes are added to the previous logical classes, so there will be two classes in the logical class, one for t and the other for t+1.

Evaluation of individuals based on deep prioritization algorithms: since the joint class is the pre-operation of the class arrangement, the merits and demerits of evaluating a joint plan mainly depend on whether the joint class scheme is beneficial to simplify the scheduling algorithm and whether it can improve the efficiency and quality of the class arrangement. According to the following function formula, the merits and demerits of each joint class mode will be evaluated. From the definition of the function, the smaller the fitness value, the better the performance of the individual in the whole problem. The specific function is as follows:

$$f_i = \max(\text{Number}(\text{Cluster}_i)) - (\text{count}(\text{Cluster}_i))^2 / 10000 + \sum_{i=1}^{M_i} \text{Major}_i \quad (3)$$

Selected action: common choices include Roulette Selection Wheel, Tournament Selection, Best Retain Selection, etc. Among these methods, Roulette Wheel Selection is one of the earliest proposed methods in genetic algorithms. Because it is simple and practical, it is also a commonly used method. It is a Proportional Model, which can be called the Monte Carlo method of choice. The probability of its retention is determined by the proportion of individual's fitness. The selected probability function is:

$$P_i = f v_i / \sum_{j=1}^M f v_j, j = 1 \dots M \quad (4)$$

The Memetic algorithm is mainly composed of a selection operator, a crossover operator, a mutation operator and a local search operator. Among them, the selection operator, the crossover operator and the mutation operator belong to the genetic operation, which plays a key role in the evolution of the population. It is because of the role of these operators that genetic algorithms can have better global optimization ability than traditional algorithms. The local search operator is the most fundamental reason why the Memetic algorithm is superior to the genetic algorithm. The local search greatly improves the search depth of the solution space, which further improves the search quality. The four most important operators in the Memetic algorithm in Figure 3 are highlighted.

Figure 3 is the flow chart of Meemctic's algorithm. The specific steps are as follows:

The first step is to determine the coding scheme of the problem and set the relevant parameters;

The second step is to initialize the population;

The third step is to perform a crossover operator;

The fourth step is to perform a neighborhood search on the individual with a local search algorithm to update all individuals;

In the fifth step, the mutation operator is executed to generate a new individual;

In the sixth step, the local search algorithm is used again to perform neighborhood search on the individual, and all individuals are updated;

In the seventh step, the fitness value of all individuals in the population is calculated by the fitness function;

In the eighth step, the selection operator is executed, the population is screened according to the natural rule of “survival of the fittest”, the individuals with poor fitness are discarded, and the individuals with high fitness are retained;

In the ninth step, it is judged whether the termination condition is satisfied. It is judged whether the algorithm has satisfied the optimization criterion or the termination condition of the algorithm, and if it is satisfied, the operation is terminated, otherwise the third step is continued.

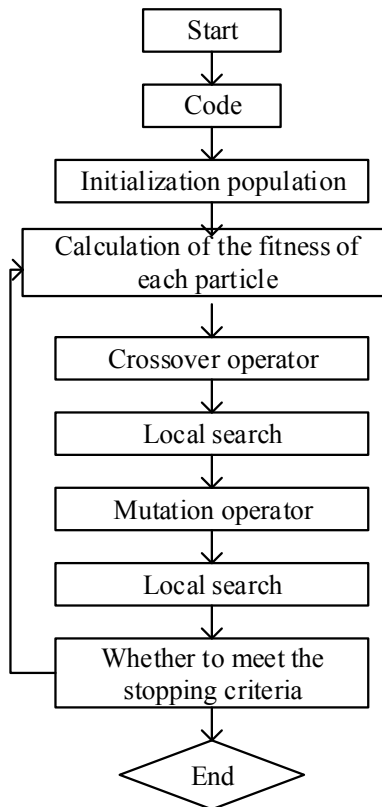


Fig. 3. Memetic algorithm flow chart

In order to better control the operation of the Memetic algorithm and ensure the effect of the algorithm, it is necessary to strictly control the parameters of the algorithm. The main parameters affecting the Memetic algorithm are group size, crossover probability, mutation probability, maximum evolution number, and maximum stagnation algebra.



### 3.3 Convergence analysis of annealed Memetic algorithm

Because the Memetic algorithm is mainly composed of genetic algorithm and local search algorithm, the research on Memetic algorithm is mainly for genetic algorithm. Especially in terms of the convergence of the algorithm, since the research of the Memetic algorithm started late, the past research mainly focused on the genetic algorithm. In the past, the research on the convergence of genetic algorithms mainly includes two contents. On the one hand, it studies the convergence of genetic algorithms, on the other hand, it studies the convergence speed of genetic algorithms. In the discussion of the convergence of genetic algorithms, the main methods include Markov chain-based research, functional analysis-based research and dynamic model-based research. The specific method is as follows:

Research based on Markov chain: the population  $X$  ( $t=x, x \in S$ ) produced by the genetic algorithm at the  $t$ th ( $t \geq 0$ ) generation is the population size. When the  $t$ -th generation group  $X(t)$  is operated by operators such as selection, intersection, mutation, etc., the algorithm will generate a new population at time  $t$ . According to the evolution process of GA, the new population depends on the state of the current populations  $t+1$ ,  $X(t+1)$ ,  $X(t+1)$ , and  $X(t)$  in a probabilistic manner. This indicates that the population sequence  $\{X(t)\}$  has a Markov property, and  $\{X(t)\}$  constitutes a discrete-time Markov chain. The Markov process is a suitable description model for GA probabilistic behavior, and it is also an important method in the previous research on the convergence of genetic algorithms. The study of GA convergence based on Markov chain begins by defining the probability convergence of the algorithm. Then, according to the genetic operation such as “select cross mutation” of the algorithm, the state transition matrix of the Markov chain is constructed separately. By studying the transition probability in the state transition matrix, the following two propositions are obtained for the simple genetic algorithm (SGA):

First, the SGA is a homogeneous, mutual Markov chain;

Second, the SGA is a traversed Markov chain.

After studying genetic algorithms and applying them widely, Holland proposed a model theory describing genetic algorithms. The model theory guarantees that the samples of the better model grow exponentially, which satisfies the necessary condition for finding the optimal solution. That is, the genetic algorithm has the possibility of searching for the global optimal solution. At the same time, the basic goal of genetic algorithms is to maintain the evolvability of the population. If the model space of the problem satisfies the building block hypothesis, the algorithm can not only describe the modes of low-order, short-range, high-adaptivity values, but also make these modes combine with each other under the action of genetic operators to generate patterns of high-order, long-distance, high-average fitness values, and finally generate global optimal solutions. The pattern theorem lays a good mathematical foundation for genetic algorithms through computational models.

The kinetic method is to study the nature of the system state over time. It mainly investigates the change mode and final characteristics of variables, such as whether there are fixed points, periodic points, mutual attraction or mutual exclusion, and stability of system structure.

By analyzing the operator of the genetic algorithm with the compression mapping theorem in functional analysis, Michalewicz proposed that Banach fixed point theorem can be used to explain the convergence of genetic algorithm. Firstly, a metric space is constructed and then the algorithm is improved. It is concluded that the genetic operator with the optimal solution reservation strategy is the compression mapping operator. Finally, the improved genetic algorithm with optimal reservation strategy is proved to be convergent.

According to the Banach compression mapping principle, for a finite state space  $S$ , if a suitable metric space can be found, so that the mapping formed by the SAMA process becomes a compression map, then it can be proved: for any initial population, the SAMA process converges to a unique fixed point  $X$  in  $NX(0)S$ . Let  $f(i)$  be the fitness function without loss of generality. It assumes that the maximum value of  $f(i)$  is solved (the solution process of the minimum is similar), and  $M$  is set to some upper bound, then, the mapping is defined as:

$$d(x_1, x_2) = \begin{cases} 0, & |M - \max_{x_i \in X_1} f(x_i)| + |M - \max_{y_i \in X_2} f(y_i)| \\ |M - \max_{x_i \in X_1} f(x_i)| + |M - \max_{y_i \in X_2} f(y_i)|, & X_1 \neq X_2 \end{cases} \quad (5)$$

## 4 Results

The teaching scheduling problem is discussed based on the whole, and the interaction of each stage of teaching scheduling and the data flow between each stage are analyzed. The pre-stage of the class arrangement is obtained, that is, the joint class has a more obvious influence on the class arrangement, and then it is proposed to find a better joint class plan to facilitate the smooth operation of the class scheduling algorithm.

The teaching scheduling problem is modeled and the evolutionary algorithm is used to solve the optimal joint class scheme. In the modeling of the joint class problem, firstly, the fitness value function of the problem is obtained by analyzing the characteristics of the joint class scheme that is beneficial to solving the class scheduling problem. Then the depth-first algorithm is used to solve the fitness value of individual. Finally, the annealing Memetic algorithm is used to solve the optimal joint class scheme.

The problem of teaching scheduling is studied in detail and the model of the problem is established. Through the analysis of the problem, the fitness value function is first defined according to the constraints of the problem. This definition has certain flexibility and scalability. When the constraints change, only the fitness value function needs to be modified. Then the problem is solved iteratively based on the annealed Memetic algorithm to obtain the optimal value. In the process of solving, according to the characteristics of the problem, the individual expression forms, genetic operators such as crossover and mutation, and local search methods are defined.

Based on the real data set of one semester of school, the experiments of solving the joint class algorithm and solving the class scheduling algorithm are carried out under different parameters. Compared with the previous class results, the algorithm has been improved in terms of the satisfaction of the joint class results and the efficiency of the

algorithm. Finally, the Banach compression mapping theorem proves that the annealing Memetic algorithm based on the optimal retention strategy is convergent.

Using the Memetic algorithm can quickly and effectively solve the teaching scheduling problem, reduce the probability of failure, and reduce the number of iterations of the calculation. The global search combined with the local search strategy not only increases the breadth of the search, but also increases the depth of the search. It shows a certain superiority in solving the problem of class scheduling and has a good application prospect. At the same time, more superior global search and local search strategies are also being explored to further improve the efficiency of the algorithm.

## 5 Conclusion

The teaching scheduling problem attracts a wide range of research interests in the field of optimization and artificial intelligence due to its large scale, many constraints and high complexity. The current population-based evolutionary algorithm is the most widely used and successful method for solving the optimal teaching scheduling problem. The main method gradually approaches the optimal solution of the problem through the iteration of the population. The key to solving such problems is to reasonably model the problem and translate the constraints into effective fitness value functions. The problem is discussed in detail, and the various stages of teaching scheduling are regarded as a whole and the impact of the joint class problem on the scheduling problem is considered. Firstly, the optimization algorithm is used to solve the most suitable scheme for class scheduling; on the basis of this, inspired by the self-pollination of some plants, an annealing Memetic algorithm based on self-intersection is designed. The constraint condition of the class scheduling problem is effectively transformed into the fitness value function. The Memetic algorithm and the simulated annealing hybrid algorithm are used to solve the problem, and the related genetic operators that solve the problem are defined to make it more efficient to solve the problem. By testing the real data of a semester of Tianjin University, the algorithm effectively improves the quality of class scheduling and improves the efficiency of calculation. Finally, the Banach compression mapping theorem is used to prove the convergence of the annealed Memetic algorithm. It is concluded that the annealed Memetic algorithm based on the optimal individual retention strategy is convergent. With the transformation of the IT era into the DT era, big data processing will become more and more common. Therefore, the research of intelligent algorithms will attract more and more scholars. As an efficient intelligent algorithm, Memetic algorithm has obvious advantages in solving large-scale, complex, high-dimensional and dynamic problems.

## 6 References

- [1] Deng, J., Wang, L., Wang, S., Zheng, X. (2016). A competitive memetic algorithm for the distributed two-stage assembly flow-shop scheduling problem. *International Journal of Production Research*, (12): 1-17. <https://doi.org/10.1080/00207543.2015.1084063>

- [2] Xu, J., Yin, Y., Cheng, T. C. E., Wu, C. C., Gu, S. (2014). An improved memetic algorithm based on a dynamic neighbourhood for the permutation flowshop scheduling problem. *International Journal of Production Research*, 52(4): 1188-1199. <https://doi.org/10.1080/00207543.2013.848042>
- [3] Gong, X., Deng, Q., Gong, G., Liu, W., Ren, Q. (2017). A memetic algorithm for multi-objective flexible job-shop problem with worker flexibility. *International Journal of Production Research*, (2): 1-17.
- [4] Yannibelli, V., Amandi, A. (2015). Project scheduling: a memetic algorithm with diversity-adaptive components that optimizes the effectiveness of human re-sources. *Forensic Science International*, 52(9): 93-103.
- [5] Shen, X. N., Minku, L. L., Marturi, N., Guo, Y. N., Han, Y. (2017). A q-learning-based memetic algorithm for multi-objective dynamic software project scheduling. *Information Sciences*: 428-511.
- [6] Zhang, G., Li, Y. (2016). A memetic algorithm for global optimization of multi-modal non-separable problems. *IEEE Transactions on Cybernetics*, 46(6): 1375-1387. <https://doi.org/10.1109/TCYB.2015.2447574>
- [7] Azad, A. S., Islam, M., Chakraborty, S. (2017). A heuristic initialized stochastic memetic algorithm for mdpvrp with interdependent depot operations. *IEEE Transactions on Cybernetics*, 47(12): 4302-4315. <https://doi.org/10.1109/TCYB.2016.2607220>
- [8] Keshavarz, T., Salmasi, N., Varmazyar, M. (2015). Minimizing total completion time in the flexible flowshop sequence-dependent group scheduling problem. *Annals of Operations Research*, 226(1): 351-377. <https://doi.org/10.1007/s10479-014-1667-6>

## 7 Author

**Shen-min Lv** is with Shandong Women's University, Shandong 250300, China (shenminlv3889@21cn.com)

Article submitted 22 September 2018. Resubmitted 19 October 2018. Final acceptance 26 October 2018. Final version published as submitted by the author.