ISBN: 978-981-18-5852-9
WCSE 2022 Spring Event: 2022 9th International Conference on Industrial Engineering and Applications
doi: 10.18178/wcse.2022.04.099

Research Survey on Open Shop Scheduling Strategies Based on Intelligent Algorithms

Chenjing Zhou [1 +] , Zhixin Ma [1]

[1] Zhejiang Sci-tech University

Abstract. Open shop scheduling problem (OSSP) is one of the fundamental scheduling problems. Many scenarios such as vehicle maintenance, the medical system, and food processing can be modeled as OSSP in production and life. Therefore, this problem has essential application value and research significance. With the rapid development of artificial intelligence, the research of intelligent algorithms has increased explosively. Genetic algorithms, swarm intelligence algorithms, and artificial neural networks are several important intelligent algorithms that can solve OSSP. This paper summarizes the OSSP based on intelligent algorithms. Firstly, it briefly introduces the intelligent algorithm and OSSP. Then, the research on evolutionary algorithms and swarm intelligence algorithms to solve the OSSP is classified and summarized. Finally, the current algorithms for solving the problem are analyzed and discussed, the problems not solved by the existing algorithms are proposed. Furthermore, the development direction of intelligent algorithms for OSSP has been prospected.

Keywords: intelligent manufacturing, open shop scheduling, intelligent algorithm, genetic algorithm, swarm intelligence algorithm

1. Introduction

Open shop scheduling problem (OSSP) is of importance in scheduling problems. It has a wide range of applications and research background. In practical industrial production and manufacturing, many scenarios such as intelligent physical examination systems and automobile repair shops can be modeled as OSSP. The problem is a well-known combinatorial optimization problem, which belongs to the NP-hard problem. The research on OSSP dates back to the 1970s [1]. The problem can be described as: there are M machines and a set of jobs, each of which contains N steps of operations that must be processed on M machines, but the jobs are processed in an arbitrary order, i.e., the jobs can start or end by choosing any process. On the premise that each machine can only operate independently one step at a time, to find optimal scheduling to minimize the makespan. With the increase of production scale, the traditional two-machine problem does not apply to the complex environment of modern enterprise intelligent workshops. Therefore, researchers have proposed the flexible open shop floor scheduling problem with multiple parallel machines. In the case of multiple types of parallel machines, the increase in the number of machines makes the problem more extensive and complex. Currently, researchers are more concerned with OSSP with special application scenarios that require consideration of various constraints.

The research methods of OSSP are mainly divided into two categories: deterministic algorithm and approximate algorithm. The commonly used deterministic algorithms include branch and bound method, dynamic programming, etc. Brucker et al. [2] first proposed the branch-and-bound method to solve the OSSP. Ozolins et al. [3] proposed a dynamic programming method and a new ruling rule to solve the problem. But the calculation cost of this way is high, especially for large-scale problems. The deterministic algorithm usually cannot get results in polynomial time. Therefore, this kind of algorithm is only suitable for solving small-scale issues. Standard approximation algorithms include heuristic algorithms and meta-heuristic algorithms, which can solve the problem in a reasonable time. Brasel et al. [4] proposed a heuristic search to optimize the branch-and-bound method to improve algorithm efficiency. Naderi et al. [5] proposed four heuristic rules to remove redundant solutions generated in the algorithm process. Mohammad et al. [6]

[+] Corresponding author. Tel.: +15868838030.
E-mail address: 15868838030@163.com.

proposed an efficient relaxation solution heuristic algorithm to solve OSSP for minimizing jobs' total weighted advance/delay. However, heuristic algorithms cannot guarantee the optimal solution or approximate the optimal solution every time. In contrast, meta-heuristic algorithms based on artificial intelligence can obtain satisfactory results in a limited time. Meta-heuristic algorithms are also known as intelligent optimization algorithms. And intelligent algorithms are better able to meet the complex environment. Many research teams have applied such algorithms to solve complex development shop scheduling problems

At present, intelligent algorithms are mainly divided into evolutionary algorithms and swarm intelligence algorithms. Compared with traditional algorithms, intelligent algorithms can solve complex optimization problems faster and with higher solution quality. Intelligent algorithms are indispensable for the rapid development of artificial intelligence and other fields. It is commonly used in intelligent manufacturing applications such as intelligent shop scheduling and machine vision recognition, and has achieved certain results. Since the solution space of OSSP is ample and complex to search, the intelligent algorithm can quickly lock the feasible solution range and search for the approximate optimal solution in a short time. Therefore, many researchers have applied intelligent algorithms to solve this problem. At the same time, in order to adapt to the special needs of complex production environments, they tried to improve the existing intelligent algorithms and proposed hybrid intelligent algorithms with higher performance.

This paper reviews and summarizes the research status of OSSP using intelligent algorithms in recent years. The specific contents are arranged as follows: Chapter 1 briefly introduces the OSSP; Chapter 2-3 summarizes the research progress of the evolutionary algorithm and swarm intelligence algorithm on this problem. In chapter 4, the shortcomings of this research field and the research prospect of intelligent algorithms to solve this problem have been discussed and prospected. Finally, it summarizes the whole paper.

2. Evolution Algorithm For Solving OSSP

Evolutionary algorithm (EA) is an optimization algorithm that draws on the biological evolutionary process in nature to obtain near-optimal solutions. And it is mature and has been widely used in practical problems. As optimization problems in the engineering field become more and more complicated, researchers put forward some new EAs with a simple structure and strong robustness. And they can obtain better solutions faster in specific problems. The diversification of optimization objectives also makes researchers turn their attention to the optimization and application of the multi-objective EAs, and those algorithms have provided reasonable solutions for solving multi-objective problems.

This chapter summarizes the research results of using EAs to solve OSSP in recent years and classifies them according to algorithm categories. Table I is a summary of the research results of applying EAs to solve OSSP. As the table shows, the research on the multi-objective OSSP is still in its infancy.

Table 1: EAs to solve the OSSP

References	Algorithms	Optimization objectives
[7]	GA	Minimize makespan
[8]	GA	Minimize makespan
[9]	GA	Minimize makespan
[10]	GA	Minimize makespan
[11]	GA	Minimize makespan
[12]	GA	Minimize makespan
[14]	DE	Minimize makespan
[15]	CEA	Minimize makespan
[17]	MA	Minimize makespan
[18]	MA	Minimize makespan, total delay
[19]	MOEA	Minimize makespan, , total machine workload
[20]	MOEA	Minimize total flow time, minimize workload of people and machines
[21]	MOEA	Minimize makespan and human error, maximize machine availability

2.1. Genetic Algorithm

Genetic algorithm (GA) is an ancient optimization algorithm and the earliest EA used to solve OSSP. To improve the efficiency of the algorithm to solve the more complex OSSP, some researchers proposed an improved scheme based on the existing GA. Ghosn et al. [7] proposed a parallel GA based on Beowulf swarm to solve this problem, which generates a set of compact scheduling schemes through a series of random and deterministic operations to improve the algorithm's exploration efficiency. Based on the single-program multi-data model, the algorithm utilizes message passing on the Beowulf swarm to ensure that each processor can operate independently on the subpopulation of individuals and periodically share its "optimal" individuals among the processors. Benziani et al. [8] used GA to solve the problem in this paper, encoding the chromosome as the processing order of the job, setting the fitness function as the completion time, initializing the population with greedy search and domain search algorithm, and improving the obtained solution. Hosseinabadi et al. [9] proposed a new extended GA to solve the OSSP problem and studied the influence of various effective crossover and mutation operators on the algorithm. The results show that using a single point crossover operator and displacement mutation operator in the algorithm can effectively improve the efficiency and quality of the solution.

As the scale of factory manufacturing increases and the complexity of product production increases, many researchers begin to pay attention to the specific OSSP. Compared with the general OSSP, the particular OSSP is more complex and closer to the actual situation. For machine maintenance in the process of OSSP, Shamshirband et al. [10] proposed an improved GA to minimize the maximum completion time. In this algorithm, each gene is represented by a one-dimensional array of length 3. The first two elements represent the machine on which a certain job is processed, and the third element indicates whether the machine needs to be maintained after performing the current operation. Barjouei et al. [11] considered the situation of machine unavailability, adopted GA and differential evolution algorithm to solve this problem, designed and introduced a semi-guided population (SGP), prioritized the important work when initializing the population. Experiments show that compared with random initialization population, SGP improves the robustness and effectiveness of the algorithm. For OSSP with distribution path, Abreu et al. [12] transformed the vehicle routing problem into OSSP, designed a biased random key GA with iterative greedy local search process, and proposed a new coding scheme.

2.2. Emerging Evolutionary Algorithm

On the basis of traditional EAs, researchers have proposed new EAs based on group competition and group cooperation. These algorithms can achieve better results when dealing with complex or specific optimization problems. For example, compared with GA, differential evolution algorithm (DE) has fewer parameters, faster convergence speed and higher solution quality [13]. Bai et al. [14] Studied the static and dynamic flexible OSSP and solved it using DE. Experiments show that DE can obtain high-quality solutions in medium-scale problems, but in the face of large-scale problems, the convergence speed of the algorithm is too slow to obtain better solutions in a short time. In the actual production scenario, customers often return goods. In view of this situation, Tanimizu [15] studied the OSSP including disassembly and post-processing operations, and proposed a co-evolutionary algorithm. However, the algorithm also takes too long in the face of large-scale problems.

2.3. Memetic Algorithm

Memetic algorithm (MA) combines global search and local search, and different search strategies can form different MAs [16]. This hybrid mechanism combines the advantages of global search and local search strategy, has higher stability and faster running speed, and is more suitable for large-scale OSSP.

Abdelmaguid et al. [17] considered the multi-processor open shop scheduling problem (MOSP). In this problem, the machines are classified into multiple workstations according to their functions. It is necessary to determine the processing order of each job and the order in which the job accesses the workstation. To solve this problem, the authors propose a decentralized search algorithm with path relinking, use the distance function to measure the similarity between the two solutions to ensure the diversity of solutions. They also design two domain search functions to improve the performance of the algorithm. For the flexible OSSP, Behnamian et al. [18] proposed a decentralized search algorithm to minimize the maximum completion time and total delay. Experimental results show that the performance of this algorithm is better than nondominated sorting genetic algorithm Ⅱ (NSGA-Ⅱ).

2.4. Multi-objective Evolutionary Algorithm

In the actual production process, a single optimization goal often cannot meet the needs of enterprises. Azadeh et al. [19] considered two optimization objectives, namely, minimizing the maximum completion time and minimizing the total workload of the machine. To solve this problem, the author establishes a new bi-objective mixed integer linear programming model and improves NSGA-Ⅱ. Ciro et al. [20] proposed two multi-objective genetic algorithms: NSGA-Ⅱ and NSGA-Ⅲ for a mechanical workshop composed of multiple machines and multiple workpieces, aiming at minimizing the total flow time of operations and the workload balance between people and machines. Experiments show that compared with NSGA-Ⅱ, NSGA-Ⅲ has better performance and is more suitable for solving problems with two or more optimization objectives. Taking completion time, human error and machine availability as optimization objectives, Sheikhalishahi et al. [21] proposed NSGA-Ⅱ, multi-objective particle swarm optimization algorithm and strength Pareto evolutionary algorithm (SPEA-Ⅱ) to solve this problem. At the same time, Taguchi method was used to adjust the parameters of meta heuristic algorithm to improve the effectiveness of the algorithm.

3. Swarm Intelligence Algorithm For Solving OSSP

Swarm intelligence optimization algorithm (SIOA) is a new evolutionary computing technology that mainly simulates the behavior of insects, birds and fish. SIOA has the advantages of simple structure and few parameters and has been widely used in practical problems[22]. This chapter will introduce the application of SIOAs in OSSP. Table II summarizes the research results of applying a SIOA to solve OSSP.

Table 2: SIOAs for solving the OSSP

References	Algorithms	Optimization objectives
[24]	ACO	Minimize makespan
[32]	ABC	Minimize makespan
[33]	ABC	Minimize makespan
[34]	BA	Minimize makespan
[37]	CSA	Minimize makespan
[28]	Two-stage PSO	Minimize makespan
[40]	Firefly-cuckoo hybrid algorithm	Minimize makespan
[42]	ACO	Minimize total flow time
[26]	ACO	Minimize total flow time
[29]	PSO	Minimize makespan, total flow time, machine idle time
[30]	Multi-objective PSO	Minimize makespan, maximize completion satisfaction
[39]	WOA	Minimize total lead time/delay time
[41]	Multi-objective simulated annealing-ant colony hybrid algorithm	Minimize total completion time and total delay time

3.1. Ant Colony Algorithm

In 1992, inspired by the ant colony's foraging behavior from nest to outside, Marco Dorigo [23] proposed ant colony optimization algorithm (ACO). Marrouche et al. [24] used cuckoo algorithm and improved ACO to solve the non-preemptive OSSP. The experimental results show that the computational results of the ACO algorithm are slightly better than those of the cuckoo algorithm. Still, the convergence speed of both of them is slower. Ciro et al. [25] considered the OSSP based on mechanical production workshop under multi resource constraints, took minimizing the total flow time as the optimization goal, and adopted GA and ACO method to solve large-scale problems. Experiments show that the performance of ACO is better than that of GA in real industrial cases. Aiming at the problem that it is challenging to optimize the parameters of ACO, Ciro et al. [26] proposed a fuzzy ACO to improve the quality of the solution, and the results demonstrate the program's feasibility

3.2. Particle Swarm Optimization

Particle swarm optimization algorithm (PSO) was proposed by Eberhart and Kennedy [27] in 1995. It simulates the foraging behavior of birds and is a random search algorithm based on group cooperation. PSO has the advantages of few parameters and fast convergence and is widely used in many fields.

In actual production, problems such as construction period delay and machine maintenance will cause significant losses to enterprises. Therefore, multi-objective optimization has become the focus of research in recent years. Pongchairerks et al. [28] proposed a two-level PSO and established an upper and lower PSO architecture. The upper PSO algorithm adjusts the parameter values for the lower PSO algorithm, and the lower PSO algorithm uses the given parameter values to solve the OSSP. Lin et al. [29] first proposed using PSO to solve multi-objective OSSP. Because the problem is a combinatorial optimization problem, the author modifies the particle position representation, particle velocity, and particle motion mode and tests the improved PSO on various benchmark problems. For the multi-objective OSSP with uncertain processing time and flexible deadline, Palacios et al. [30] used fuzzy set modeling and PSO to deal with the optimization objectives of minimizing completion time and maximizing customer satisfaction.

3.3. Artificial Bee Colony Algorithm

To optimize algebraic problems, Karaboga et al. [31] proposed an artificial bee colony algorithm (ABC) in 2005. The algorithm simulates bee colonies' cooperative honey collection behavior and has fast convergence speed. For the large-scale OSSP, Zhu et al. [32] proposed an improved ABC algorithm. The experimental results show that the performance of ABC is better than ACO, GA, PSO and cuckoo search algorithm. Huang et al. [33] proposed an ABC based on Idle Time Filtering (ITBF). The algorithm puts forward the concept of profitability and defines profitability as the reciprocal of partial scheduling idle time. Experiment shows that this method can improve search efficiency.

3.4. Other Swarm Intelligence Algorithms

Bat algorithm (BA) is an algorithm proposed by Yang et al. [34] in 2010. The algorithm is developed according to the echolocation and predator-prey behavior of bats and performs well in optimization problems. Shareh et al. [35] studied the task scheduling problem in the OSSP using the improved bat algorithm, and carried out experiments to verify the feasibility of the algorithm in this kind of problem.

Cat swarm algorithm (CSO) is a global optimization algorithm proposed by Chu et al. [36]. Bouzidi et al. [37] used the cat swarm algorithm to solve the OSSP. Experiments show that CSO can solve some common problems in benchmark examples, improve search efficiency and find better solutions, and effectively solve the problem of falling into local optimal solutions.

Whale optimization algorithm (WOA) is a new heuristic optimization algorithm that imitates the hunting behavior of humpback whales proposed by Mirjalili et al. [38]. The algorithm has the advantages of simple operation, few adjusted parameters and strong ability to jump out of local optimization. GUI et al. [39] proposed a hybrid whale optimization algorithm (HWOA) to minimize the total lead/delay time. They use a mutation operation based on exchanging neighborhoods to avoid premature convergence and improve the global search ability. A variable neighborhood search mechanism is designed to improve the search depth of the solution in the feasible region, so as to improve the ability of local search.

3.5. Hybrid Algorithm

Because a single SIOA has some limitations, some scholars have proposed a hybrid SIOA, which combines the advantages of some SIOAs to improve the efficiency and effectiveness of the algorithm.

The cuckoo search algorithm is a widely used and efficient heuristic algorithm. Compared with other algorithms, it has fewer parameters, simple operation, easy implementation, random search path superiority, and merit-seeking solid ability characteristics. N. Kamatchi et al. [40] proposed a hybrid algorithm of firefly-cuckoo search algorithm to solve the flexible OSSP. The algorithm can avoid the optimal local problem and get a better solution. They compared the proposed hybrid algorithm with ACO and GA, and the results proved the superiority of the algorithm.

There is usually the problem of slow convergence and inability to obtain the optimal solution in a reasonable amount of time in the ACO. Panahi et al. [41] consider the OSSP with the two objectives of minimizing the total completion time and total delay time, and propose a hybrid algorithm based on multi-objective simulated annealing and ACO. The algorithm uses simulated annealing to initialize the parameters, and then uses the improved ACO to optimize, which reduces the sensitivity of the final solution to the initial parameters. They compared the calculation results with NSGA-II, and the results proved the superiority of this method.

4. Problems and Prospects

4.1. Problems of Existing Algorithms

- Insufficient robustness: When the scale of the problem increases, the solution space also increases, so the timeliness of the optimal solution search can not be guaranteed. In addition, it is easy to fall into local optimization when the complexity of the problem increases, because the intelligent algorithm is generally solved from local expansion. Therefore, the existing intelligent algorithms can not meet the large-scale OSSP.
- Single optimization objective: The existing research results mostly focus on minimizing the completion time as the optimization objective. Still, in the actual production process, the influence of other factors needs to be considered. So, the single objective optimization scheme can not meet the needs of a complex environment in OSSP.
- Incomplete constraints: The existing research models do not fully consider the limitations and do not fully involve the complex needs such as job priority and worker uncertainty in intelligent workshop scheduling. Therefore, when establishing the scheduling model, we need to comprehensively consider smart workshops' more flexible, parallel, and acute scheduling requirements.

4.2. Research Prospect of Intelligent Algorithm for Ossp

In response to the problems of existing intelligent algorithms, this paper proposes possible future research directions from three aspects.

- Hybrid intelligent algorithm: Due to the defects of a single algorithm, such as easy to fall into local optimization and slow solution speed, many researchers propose MA and hybrid SIOA to solve the OSSP. These hybrid algorithms combine their respective advantages. For example, in the firefly-cuckoo search algorithm [39], because the firefly algorithm has global attributes, its introduction can avoid the problem that cuckoo search is easy to fall into local optimization. However, this approach is still being explored. Therefore, the hybrid intelligent algorithm provides a worthy research direction for improving the performance of intelligent optimization algorithms.
- Dynamic parameter adjustment: In the existing research of GAs and SIOAs, many researchers optimize the algorithm's performance by adjusting parameters, but the process is not universal. The tedious parameter adjustment process needs to be carried out for the new application scenario. Therefore, how to dynamically adjust the parameters so that the algorithm can adapt to different application scenarios to improve the algorithm's robustness remains to be further studied.
- Neural network: As an essential branch of artificial intelligence, the neural network has developed rapidly in recent years and has made exemplary achievements in many fields. Li et al. [42] tried to use a graph neural network to solve OSSP and train the model with reinforcement learning. Compared with traditional methods, the neural network has better performance and universality, and the computational cost will not increase significantly with the increase of the problem scale. At present, the research on using neural network to solve this problem is still in its infancy, and there are few achievements. Therefore, using a neural network to solve this problem will be very meaningful work in a complex environment.
- Multi-objective OSSP: Multi-objective OSSP is a very practical problem. In the actual production process, while seeking to minimize makespan, optimizing the total flow time of jobs or the whole

delay time of orders is also necessary. At present, the effectiveness of this research direction is general, and its efficiency in solving large-scale problems is not high. Therefore, more researchers need to pay attention to this research direction in the future.

- New OSSP model: This paper also puts forward the research direction of the new OSSP model for incomplete constraints. Due to the increasing demand for intelligent manufacturing, the OSSP needs to adapt to a more complex production environment. The existing OSSP model has few constraints, and the factors such as job priority, machine availability and uncertainty are rarely concerned by researchers. These factors will directly impact the scheduling scheme in the actual production scheduling process. Improving or designing a new OSSP model to meet the existing intelligent requirements is also essential for future research.

5. Summary

Industrial intelligent manufacturing has been the development trend of manufacturing enterprises, which has given rise to new application scenarios such as large-scale personalized customization, intelligent production and distribution, and smart workshop. All those can be modeled as OSSP. This paper explains the definition of this problem and its common research methods. Then, it analyzes the research status of intelligent algorithms in solving this problem from two aspects of EAs and SIOAs respectively. Finally, this paper summarizes that there are still three problems of intelligent algorithms in this research field: insufficient robustness, single optimization objective, and incomprehensive constraints. And combined with the new requirements of the smart workshop, this paper looks forward to five future research directions in this field.

6. References

[1] Ahmadian M M, Khatami M, Salehipour A, et al. Four decades of research on the open-shop scheduling problem to minimize the makespan[J]. European Journal of Operational Research, 2021.

[2] Brucker P, Hurink J, Jurisch B, et al. A branch & bound algorithm for the open-shop problem[J]. Discrete Applied Mathematics, 1997, 76(1-3): 43-59.

[3] Ozolins A. Dynamic programming approach for solving the open shop problem[J]. Central European Journal of Operations Research, 2019: 1-16.

[4] H, Bräsel, T, et al. Constructive heuristic algorithms for the open shop problem[J]. Computing, 1993, 51(2):95-110.

[5] Naderi B, Ghomis S , Aminnayeri M , et al. A contribution and new heuristics for open shop scheduling[J]. Computers & Operations Research, 2010, 37(1):213-221.

[6] Ahmadian M M, Salehipour A, Kovalyov M. An efficient relax-and-solve heuristic for open-shop scheduling problem to minimize total weighted earliness-tardiness[J]. Available at SSRN 3601396, 2020.

[7] Ghosn S B, Drouby F, Harmanani H M. A parallel genetic algorithm for the open-shop scheduling problem using deterministic and random moves[J]. Int. J. Artif. Intell, 2016, 14(1): 130-144.

[8] Benziani Y, Kacem I, Laroche P. Genetic algorithm for open shop scheduling problem[C]//2018 5th International Conference on Control, Decision and Information Technologies (CoDIT). IEEE, 2018: 935-939.

[9] Hosseinabadi A A R, Vahidi J, Saemi B, et al. Extended genetic algorithm for solving open-shop scheduling problem[J]. Soft computing, 2019, 23(13): 5099-5116.

[10] Shamshirband S, Shojafar M, Hosseinabadi A A R, et al. RETRACTED ARTICLE: OSGA: genetic-based open-shop scheduling with consideration of machine maintenance in small and medium enterprises[J]. Annals of Operations Research, 2015, 229(1): 743-758.

[11] Barjouei A S, Barabadi A, Tavakkoli-moghaddam R. Non-Preemptive Open Shop Scheduling Considering Machine Availability[C]//2019 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM). IEEE, 2019: 1280-1284.

[12] Abreu L R, Tavares-neto R F, Nagano M S. A new efficient biased random key genetic algorithm for open shop scheduling with routing by capacitated single vehicle and makespan minimization[J]. Engineering Applications of Artificial Intelligence, 2021, 104: 104373.

[13] Storn, Rainer, Kenneth price. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. Journal of global optimization 11.4 (1997): 341-359.

[14] Bai D, Zhang Z H, Zhang Q. Flexible open shop scheduling problem to minimize makespan[J]. Computers & Operations Research, 2016, 67: 207-215.

[15] Tanimizu Y, Sakamoto M, Nonomiya H. A co-evolutionary algorithm for open-shop scheduling with disassembly operations[J]. Procedia CIRP, 2017, 63: 289-294.

[16] Moscato, Pablo. On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms. Caltech concurrent computation program, C3P Report 826 (1989): 1989.

[17] Abdelmaguid T F. Scatter search with path relinking for multiprocessor open shop scheduling[J]. Computers & Industrial Engineering, 2020, 141: 106292.

[18] Behnamian J, Dezfooli S M, Asgari H. A scatter search algorithm with a novel solution representation for flexible open shop scheduling: a multi-objective optimization[J]. The Journal of Supercomputing, 2021: 1-24.

[19] Azadeh A, Goldansaz S M, Zahedi-anaraki A H. Solving and optimizing a bi-objective open shop scheduling problem by a modified genetic algorithm[J]. The International Journal of Advanced Manufacturing Technology, 2016, 85(5): 1603-1613.

[20] Ciro G C, Dugardin F, Yalaoui F, et al. A NSGA-II and NSGA-III comparison for solving an open shop scheduling problem with resource constraints[J]. IFAC-PapersOnLine, 2016, 49(12): 1272-1277.

[21] Sheikhalishahi M, Eskandari N, Mashayekhi A, et al. Multi-objective open shop scheduling by considering human error and preventive maintenance[J]. Applied Mathematical Modelling, 2019, 67: 573-587.

[22] Qin Xiaolin, Luo Gang, Li Wenbo, ZHANG Guohua. A Review of Swarm Intelligence Algorithms [J]. Unmanned Systems Technology,2021,4(03):1-10.

[23] Dorigo M. Optimization, learning and natural algorithms[J]. Ph. D. Thesis, Politecnico di Milano, 1992.

[24] Marrouche W, Harmanani H M. Heuristic approaches for the open-shop scheduling problem[M]//Information Technology-New Generations. Springer, Cham, 2018: 691-699.

[25] Campos Ciro G, Dugardin F, Yalaoui F, et al. Open shop scheduling problem with a multi-skills resource constraint: a genetic algorithm and an ant colony optimisation approach[J]. International Journal of Production Research, 2016, 54(16): 4854-4881.

[26] Ciro G C, Dugardin F, Yalaoui F, et al. A fuzzy ant colony optimization to solve an open shop scheduling problem with multi-skills resource constraints[J]. IFAC-PapersOnLine, 2015, 48(3): 715-720.

[27] Eberhart R, Kennedy J. A new optimizer using particle swarm theory[C]. MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science. IEEE, 1995: 39-43.

[28] Pongchairerks P, Kachitvichyanukul V. A two-level particle swarm optimisation algorithm for open-shop scheduling problem[J]. International Journal of Computing Science and Mathematics, 2016, 7(6): 575-585.

[29] Lin H H, Sha D Y. A new particle swarm optimization for multi-objective open shop scheduling[J]. Journal of Information and Optimization Sciences, 2011, 32(6): 1269-1287.

[30] Palacios J J, González-rodríguez I, Vela C R, et al. Swarm lexicographic goal programming for fuzzy open shop scheduling[J]. Journal of Intelligent Manufacturing, 2015, 26(6): 1201-1215.

[31] Karaboga D. An idea based on honey bee swarm for numerical optimization[R]. Technical report-tr06, Erciyes university, engineering faculty, computer engineering department, 2005.

[32] Zhuang Z, Huang Z, Lu Z, et al. An improved artificial bee colony algorithm for solving open shop scheduling problem with two sequence-dependent setup times[J]. Procedia CIRP, 2019, 83: 563-568.

[33] Huang Y M, Lin J C. A new bee colony optimization algorithm with idle-time-based filtering scheme for open shop-scheduling problems[J]. Expert Systems with Applications, 2011, 38(5): 5438-5447.

[34] Yang X S. Nature-inspired metaheuristic algorithms (Frome: Luniver Press)[J]. 2010.

[35] Shareh M B, Bargh S H, Hosseinabadi A A R, et al. An improved bat optimization algorithm to solve the tasks scheduling problem in open shop[J]. Neural Computing and Applications, 2021, 33(5): 1559-1573.

[36] Chu S C, Tsai P W, PAN J S. Cat swarm optimization[C]//Pacific Rim international conference on artificial intelligence. Springer, Berlin, Heidelberg, 2006: 854-858.

[37] Bouzidi A, Riffi M E, Barkatou M. Cat swarm optimization for solving the open shop scheduling problem[J]. Journal of Industrial Engineering International, 2019, 15(2): 367-378.

[38] Mirjalili S, Lewis A. The whale optimization algorithm[J]. Advances in engineering software, 2016, 95: 51-67.

[39] Gui Huimin, Qian Bin, HU Rong, et al. Hybrid Whale Optimization Algorithm for Open Shop Scheduling Problem[C].The 30th Chinese Process Control Conference(CPCC 2019).

[40] Kamatchi N, Saravanan D. A hybrid algorithm using firefly and Cuckoo search algorithm for flexible open shop scheduling problem[J]. International Journal of Mechanical and Production Engineering Research and Development (IJMPERD), 2017, 7(2): 155-166.

[41] Panahi H, Tavakkoli-moghaddam R. Solving a multi-objective open shop scheduling problem by a novel hybrid ant colony optimization[J]. Expert Systems with Applications, 2011, 38(3): 2817-2822.

[42] Li J, Dong X, Zhang K, et al. Solving open shop scheduling problem via graph attention neural net-work[C]//2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI). IEEE, 2020: 277-284.