Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 3, No. 1, 2023

Research on the Cuckoo Algorithm for Flexible Workshop Scheduling Problems

Changlong Huang *, Ling Pei, Qi Ouyang

Hohai University, Nanjing, China

* **Corresponding author**: Changlong Huang (Email: 2061010512@hhu.edu.cn)

Abstract: As a typical combinatorial optimization problem, the essence of the solution to the scheduling problem is to formulate a reasonable scheduling scheme to arrange and allocate production resources, thereby obtaining the optimal scheduling result. The Cuckoo Search Algorithm (CS), a revolutionary meta-heuristic, is based on the cuckoo's breeding behavior and combines Lévy flight and random walk strategies to more efficiently attain the search goal. CS algorithm has been extensively employed in a variety of intricate combinatorial optimization issues, with its few parameters, precise resolution, random search path optimization, robust search aptitude, and uncomplicated implementation leading to satisfactory outcomes. After in-depth research, we found that the cuckoo search algorithm excels in dealing with flexible workshop scheduling problems. We conduct a series of comparative experiments to further confirm the efficacy of this algorithm. The results show that the cuckoo algorithm performs well in global search and can be used to solve complex flexible workshop scheduling models.

Keywords: Flexible workshop scheduling; Cuckoo algorithm; Search capability.

1. Introduction

One of the most extensively researched and utilized issues in production management, the scheduling problem has a critical part to play in the energy, petrochemical, pharmaceutical, and steel processing sectors. Optimizing production process while adhering to constraints, and creating a set of suitable scheduling strategies to achieve particular scheduling objectives -- this is what scheduling is all about. Through this form, the factory production management is optimized to achieve the refined management of factory production. Flexible Flow-shop Scheduling Problem (FFSP) is a scheduling problem under the generalization of assembly shop and parallel machine environments, by Salvador in 1973 [1] [2]. The scheduling problem of a workshop type, possessing a strong engineering background and broad application, is of great practical importance. The flexible job shop scheduling issue, an expansion of the customary job shop scheduling problem, is now a reality [3]. With the emergence of flexible job shop scheduling problems, the reduction of machine constraints has greatly increased uncertainty, which has been described as an NP-Hard problem [4]. Currently, the algorithms for tackling the scheduling issues of flexible work workshops are mainly particle swarm, tabu search, genetic, and cuckoo. The field of production scheduling has seen a surge in the importance of research into efficient algorithms due to the advancement of technology. Because of the complexity of flexible workshop scheduling, there are currently three different types of algorithms: precise algorithms, heuristic algorithms, and modern meta-heuristic algorithms, all of which can effectively solve this problem. Yang of the University of Cambridge and Deb of the Raman School of Engineering have proposed a new modern meta-heuristic algorithm, the cuckoo search algorithm, which parasitizes the nest of certain cuckoo species (brood parasitism). Breeding behavior and Lévy flight behavior of birds, fruit flies, etc., have the advantage of having less parameter control and being able to strike a balance between local and global searches, which greatly improves the accuracy and efficiency of searches. Studies have shown that this algorithm is far more efficient than particle swarm and genetic algorithms. After that Yang and Deb then put forth a novel multi-objective cuckoo search algorithm that could effectively resolve the engineering design optimization issue. In recent years, the CS algorithm has seen remarkable advancement in both domestic and international studies, particularly in resolving the continuous optimization issue, and has made considerable strides in recent times. However, in solving discrete problems, the application of CS algorithm is still quite rare, with only a few articles, such as Ouyang Xinxin's proposal for a discrete valley bird search algorithm to solve the spherical traveling salesman problem. However, there has been no research on the use of this algorithm to solve job scheduling problems. Therefore, this paper aims to explore the effective application of CS algorithm in flexible workshop scheduling.

2. Flexible Workshop Scheduling Problem

2.1. Problem Description

The flexible workshop scheduling problem is simplified to: n workpieces are processed on m machines, and each machine can adjust the processing process according to needs, but the processing time of each machine will be different [5]. In order to achieve the best production efficiency, the scheduling scheme should select the best processing machine according to the sequence of processing, so as to effectively optimize the processing process of each machine, so as to minimize the final completion time. In addition, the following constraints need to be met:

(1) The number of machines and workpieces is determined and cannot be changed halfway;

(2) At the same time, each machine can only process one workpiece;

(3) At the same time, a workpiece can only be processed by one machine;

(4) The processing of the same workpiece has a sequential

time constraint, that is, the processing of the next process can only be started after the processing of the previous process;

(5) Once the machine starts processing the workpiece, it cannot be stopped or interrupted in the middle;

(6) The machine starts working from 0 moment.

In actual production, factories often have some different demand goals for the production of workpieces, and these demand goals are expressed in the scheduling model as the objective function of scheduling. If there is only one scheduling target, it is called a single-target problem; If there are multiple scheduling targets, it is called a multi-target problem. In general, FJSP problems mainly have the following scheduling goals:

(1) Maximum completion time. The smaller the maximum completion time, the higher the machine production efficiency, and the maximum completion time is also the most common evaluation indicator in FJSP.

(2) Order delivery time. In actual production, if the workpiece is completed prematurely, it will generally be put into inventory before the delivery date, which will lead to increased inventory pressure, but if the production is not completed by the delivery date, it will affect customer satisfaction. Therefore, when carrying out actual production, it is necessary to reasonably allocate the processing time so that the completion time is close to the delivery time.

(3) Machine load situation. In the production process, the specific production distribution will have an impact on the load of the machine, and if a machine is in a high-load operation state for a long time, it will cause a certain degree of damage to it. Therefore, when processing, it is necessary to try to avoid a large number of workpieces piled up on a machine for processing, and try to evenly distribute the workpieces as much as possible without affecting other demand targets to balance the load of the machine.

(4) Green manufacturing. In addition to considering the economic indicators of production, green scheduling also needs to achieve the purpose of reducing production costs, reducing carbon emissions, reducing production energy consumption, avoiding energy waste, and preventing environmental pollution without reducing product quality.

2.2. Scheduling Model

First, a mathematical model of FJSP is given, where the symbol definition is as follows [6]:

n represents the number of workpieces;

m represents the number of operations;

Mj represents the set of usable machines for the jth process;

Pijk represents the time required for the process j of the workpiece i to be processed on machine k;

Sijk indicates the start time of the process j of the workpiece i on the machine k;

Cijk represents the end machining time of the workpiece i's process j on machine k;

Njk represents the number of workpieces processed by process j on machine k;

Pjk represents the Rth workpiece machined on machine k of process j;

If Xijk is 1, the process j of workpiece i is processed on machine k; If Xijk is 0, a mathematical model is built with the minimized maximum time to complete the optimization goal, and its objective function is: min (max (Cijk)).

The constraints are shown in Equations (1) through (6).

$$\sum_{k=1}^{M_j} x_{ijk} = 1 \quad (1)$$

$$\sum_{k=1}^{M_j} N_{jk} = n \quad (2)$$

$$C_{ijk} \leq S_{i(j+1)r}, r \in M_{j+1} \quad (3)$$

$$C_{ijk} = S_{ijk} + P_{ijk} \quad (4)$$

$$C_{R_{jk}jk} \leq S_{(R+1)_{jk}jk}, R = 1,2,\ldots,N_{jk} - 1 \quad (5)$$

$$x_{ijk} \in \{0,1\} \qquad \forall i,j,k \quad (6)$$

Where i=1,2,3...., n; j=1, 2, 3... , m; k∈Mj, equation (1) means that each operation can only be performed at the same time Done by one machine; Equation (2) means that for each process, the sum of the number of workpieces assigned to all machines in the process is n; Equation (3) indicates that the same workpiece must follow the processing sequence constraint when processing, that is, only after completing the processing of the previous process, can the processing of the next process begin; Equation (4) indicates that the completion time of the workpiece should be equal to the sum of the workpiece start processing time and the time required for the processing of the process; Equations (5) indicate that each machine can only complete one process at the same time; Equation (6) is the range of values of the variable.

3. Optimization Mechanism of Cuckoo Algorithm

3.1. Principles of the Cuckoo Search Algorithm

A meta-heuristic algorithm, simulating the parasitic reproductive behavior of cuckoos during breeding, is known as the cuckoo search algorithm. Compared with other birds, the biggest feature of the cuckoo is that it does not build its own nest and hatch, nor does it feed its own young, but uses a parasitic method to lay eggs in the nests of other birds and use other birds to reproduce offspring. In order to increase the probability of survival, some cuckoos will imitate the eggs of the host bird, so that they are as consistent as possible in color and size with the eggs of the host bird, so as to achieve the purpose of deceiving the host bird. Because the hatching period of cuckoo eggs is shorter than that of other birds, in order to ensure the survival rate of offspring, cuckoos usually parasitize only one of their own eggs in a nest, and the first cuckoo chicks to hatch will instinctively push other birds or eggs out of the nest, leaving the parent birds to raise themselves alone to ensure their own survival. If the host bird finds that it is not its own egg, it will throw the cuckoo egg out of the nest, or simply abandon its nest and build a new one.

3.2. Lévy Flight Mechanism

Lévy flight was first proposed by the famous French scientist Lévy in 1930, after studying the foraging behavior of some insects and animals in the wild, it was found that their foraging trajectory conformed to a random walking process of heavy tail distribution. In 2009, Yang and Deb introduced the Lévy flight strategy into the cuckoo algorithm with good results. Lévy Flight Strategy is a complex random migration pattern that is extremely jumpy, can search quickly over short distances, or can constantly explore over long distances, swimming in a random way. The long-range detection of Lévy flight can effectively expand the scope of the search to avoid falling into the local best state, while the short-range search can more effectively ensure the best results for the whole world. The advent of Lévy flight strategy has aroused the great interest of scholars and researchers in the field of swarm intelligence optimization algorithm, and they have applied it to the design of swarm intelligence optimization algorithm, which greatly improves the overall optimization ability of the algorithm and can better capture the characteristics of multiple groups, thereby greatly improving the search efficiency of the algorithm.

3.3. Specific Implementation of the Cuckoo Search Algorithm

(1) Each time, the cuckoo will lay a unique egg, and they will be randomly arranged into a host nest;

(2) the number of host nests has been fixed from the beginning;

(3) The probability of cuckoo eggs being found in the host nest was Pa, Pa∈[0,1];

(4) Should the host avian discover parasitic eggs in its nest, it will abandon the nest;

(5) If the host bird does not find a parasitic egg, the nest will be retained and carried to the next generation.

Based on the above hypotheses, it can be concluded that the parasitic egg in each nest represents a feasible solution, and if the parasitic egg is found and discarded, the cuckoo will look for a new host nest, that is, produce a new feasible solution. As shown in flowchart 1, the specific implementation of the cuckoo algorithm is mainly divided into the following four steps:

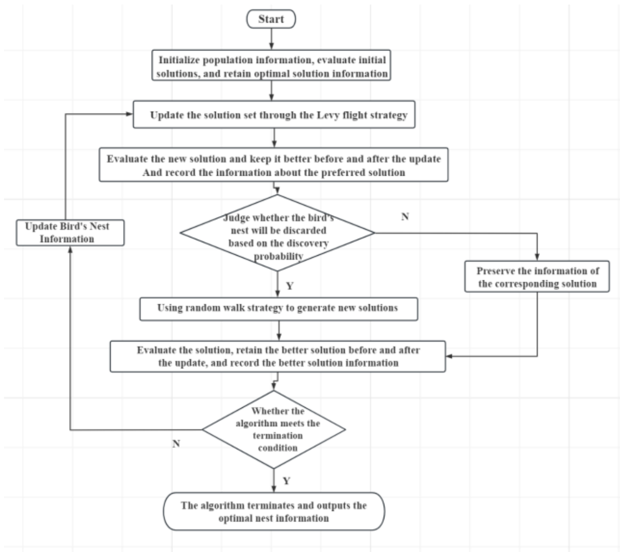

Figure 1. Flow Chart of Cuckoo Algorithm Implementation

Step1: Begin by establishing the population data, including the maximum number of iterations, the probability of discovery, and the upper and lower limits of the search space dimension; then, randomly create N nests in the D-dimensional space in accordance with the dimension's limitations, assess the initial nest information, and keep the best nest data.

Step2: Suppose that the position of the i-th nest is to evaluate the nest information $x_i = (x_{i1}, x_{i2}, ..., x_{iD}), 1 \leq i \leq N,$ of the tth generation and retain the optimal nest information, according to the formula (7) and the formula (8) Update to get the T +1 generation nest information:

$$x_i^{t+1} = x_i^t + \alpha \oplus Levy(\beta) \quad (7)$$

$$x_i^{t+1} = x_i^t + \alpha_0 \times \frac{\Phi\times\mu}{|v|}(x_i^t - x_{best}) \quad (8)$$

Where x_i^t it represents the ith (i=1,2,3...) of the t generation,⊕are point-to-point multiplication, α ⊕usually take a value of 1, which is a random search path, following $Levy(\beta)Levy$the distribution:

$Levy(\beta)$~μ=$t^{-\beta}$, 1<β≤3

For ease of operation, random numbers are generated using $Levy$equation (9).

$$Levy\beta = \frac{\Phi\times\mu}{|v|^{\frac{1}{\beta}}} \quad (9)$$

Where $\mu$and v follow the standard normal distribution, β=1.5

$$\Phi = \left(\frac{\Gamma(1+\beta)\times sin(\pi\times\beta/2)}{\Gamma(((1+\beta)/2)\times\beta\times 2^{\frac{\beta-1}{2}})}\right) \quad (10)$$

Step3: Generate a uniformly distributed random number r,r ∈(0,1) to determine whether the cuckoo egg has been found by the host bird, if r> Pa, the nest is discarded and the old nest is replaced by the random walk strategy shown in Equation (11). After the update, the nest information is evaluated and the current optimal nest information is retained;

$$x_i^{t+1} = x_i^t + \gamma(x_g^t - x_k^t) \quad (11)$$

Where γ is a uniformly distributed scaling factor thatx_g^t represents two random numbers $x_k^t$of generation t.

Step4: Determine whether the termination condition is met, if not, go to Step 2 to continue iteration, if the termination condition is met, the algorithm terminates and outputs the optimal nest information.

4. Problem Solving of Flexible Workshop Based on Cuckoo Search Algorithm

4.1. Encoding Method

In this paper, an operation-based coding rule is adopted to facilitate the solution of JSP problems. According to this rule, chromosomes are composed of n ×m genes, each of which represents an arrangement of processes in which each part number appears only m times [7]. For example, in the example of a 4 workpiece × 3 machine, the encoding of chromosomes is 121334412234.Therefore, its corresponding process sequence is: (J 1,1, J 2,1, J 1,2, J 3,1, J3,2, J4,1, J4,2, J1,3, J2,2, J2,3, J3,3, J4,3).

Among them, Ji,j represents the j-th process of the i-th workpiece, and j represents the number of occurrences of workpiece i. Therefore, the expression means that the first process of the first workpiece is processed firstly, then the first process of the second workpiece, and then the first process. The second operation of the first workpiece, the first process of machining the third workpiece, and so on, finally the 4th machine the third operation of the workpiece. Therefore, by decoding the order of the workpieces, it can be converted into an effective scheduling scheme.

4.2. Algorithm Process

To conclude, the job shop scheduling problem can be solved by employing the cuckoo search algorithm as follows.

(1) Effective training can be achieved by adjusting the number of nests n, the probability Pa of the host detecting foreign bird eggs, and adjusting the maximum number of iterations Max T and the search accuracy ε.

(2) According to the process coding rules provided in Section 4.1, the position of the bird's nest is converted into an ordered arrangement to calculate the objective function value of each nest position, so as to determine the current optimal nest position.

(3) Update all other nest positions through Levy flight while keeping the optimal nest position of the previous generation unchanged, the next generation of nests is randomly generated and the objective function value of each

nest after position update is evaluated. At this stage, the Mantegna algorithm is used to achieve the Levy flight [8].

(4) When conducting a local search, adjust the position of each nest according to some specific conditions: use a random number Ra as the probability that the nest owner finds exotic eggs, then compare it to Pa. If Ra is greater than Pa, we will randomly adjust the position of the nest, otherwise we will maintain the original position. And calculate the objective function value of each nest after adjustment, and record the current best nest position.

(5) By comparing the results of this iteration and the previous iteration, if the new optimal value is found to be lower than the original optimal value, it is used as the objective function value of the current optimal bird's nest position to ensure the best result.

(6) If the number of searches exceeds the expected threshold, or if the accuracy meets the requirements, transfer them to (7), and if they do not meet the requirements, transfer them to (3).

(7) Output the optimal scheduling value and the corresponding scheduling solution.

5. Flexible Workshop Scheduling Simulation Experiment

5.1. Study Extension

Based on the Benchmark study set [9], this paper expands 12 test studies applicable to the scheduling problem of distributed flexible assembly plant, and ends with "Original study name - number of workshops the parameters of the generated test study are shown in Table 1. The interpretation of the name of the study is such as "J10C5C-3" to indicate 10 workpieces and 5 processes, and the process can be selected as follows Class C (Class C: 2 machines are optional for the middle process, and 3 machines are available for the rest of the process; Class D: All processes have 3 machines optional), the number of workshops is 3, and the processing time is [3,20] obeys discrete uniformly distributed random numbers, and transit time [1,4] obeys discrete uniformly distributed random numbers.

Table 1. Generates Study Parameters

Study Name	Number of Workers	The Number of Operations	Process Processing Machine Layout	Number of Workshops	Processing Time	Transit Time
J10c5c-k	10	5	3/3/2/3/3	K {2,3,4}	[3,20]	[1,4]
J10c10d-k	10	10	3/3/3/3/3/3/3/3/3/3	K {2,3,4}	[3,20]	[1,4]
J15c5d-k	15	5	3/3/3/3/3	K {2,3,4}	[3,20]	[1,4]
J15c10c-k	15	10	3/3/3/3/2/3/3/3/3/3	K {2,3,4}	[3,20]	[1,4]

5.2. Problem Description

Many parameters influence the CS algorithm's performance, the primary ones being: the number of nests (Pop), the cross probability (Pc), the chance of variation (Pm), and the probability of discovering the original nest owner (PA). In this paper, the parameters of the CS algorithm are set by orthogonal test method, which are Pop = 80, Pc = 0. 8, and Pm = 0.3, Pa= 0.3.

5.3. Problem Description

This paper contrasts the cuckoo and genetic algorithms to demonstrate the CS algorithm's efficacy and superiority in tackling DFFSP issues. Among them, the parameter settings of the CS algorithm are set according to the above conclusions, and the population initialization strategy is adopted, while the other three are used the algorithm completely adopts the random initialization strategy. The algorithm running environment is: Intel Corei7, CPU2.8 GHZ, RAM8GB, Win1064bit Operating system and Matlab2016a programming software, the results of this experiment are shown in Table 2.

To guarantee the precision of the outcomes and diminish the production of mistakes, 30 independent calculations were performed on each algorithm, and its performance was measured by three indicators: optimal value (Cbset), average value (Avg) and relative error (RPD), among which RPD calculation method can be expressed by the equation (12).

$$RPD=\frac{C_0-C^*}{C^*}\times 100 \quad (12)$$

Where C0 represents the optimal value of the current algorithm after 30 iterations, and C* is obtained by all algorithms after 30 iterations Optimal value.

According to the experimental data in Table 2, the three performance indicators of CS (optimal value, average value and relative error value) are optimal, and this result is The CS algorithm provides strong support and proves its effectiveness and feasibility.

Figure 2 shows the iteration curve using the j15c10c-2 study as an example, and CS adopts a population initialization strategy based on the completion time of the optimized workpiece, which significantly improves the quality of the initial solution, and L The flight mechanism and population evolution strategy also help prevent falling into local optimum, which greatly improves the convergence speed of the algorithm.

Table 2. Experimental Results

Example	CS			DCS		
	Cbest	AGV	RPD	Cbest	AGV	RPD
j10c5c-2	54	59	0	58	62	7.4
j10c10d-2	100	103	0	102	106	2.0
j15c5d-2	63	66	0	64	69	1.59
j15c10c-2	120	127	0	127	135	5.83
j10c5c-3	56	58	0	57	59	1.79

Example	DPSO			GO		
	Cbest	AGV	RPD	Cbest	AGV	RPD
j10c5c-2	60	64	11.11	56	60.30	3.7
j10c10d-2	103	106	3.00	101	105.37	1.00
j15c5d-2	65	69	3.17	65	67.17	3.17
j15c10c-2	129	135	7.50	123	129.97	2.50
j10c5c-3	62	65	10.71	57	60.77	1.79

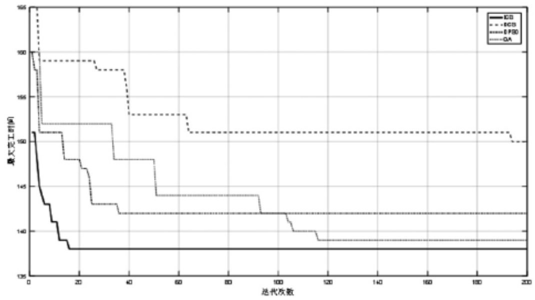

Figure 2. J15C10C-2 Study Iteration

6. Conclusion

Based on the actual production requirements, this paper constructs a distributed flexible assembly workshop scheduling model considering the transportation time. In order to better adapt to the characteristics of the model, the corresponding coding scheme and population initialization strategy are designed using the cuckoo algorithm. After the verification of comparative experiments, the cuckoo algorithm shows excellent global search ability, which can effectively solve complex model problems. Although the model proposed in this paper can solve the problem of flexible workshop scheduling, its limitations are too idealistic, because in the real production environment, there are often various unpredictable situations, such as sudden orders, equipment failures, etc. Therefore, future research can focus on flexible workshop scheduling problems with complex variables in the hope of obtaining better results.

References

[1] Haojia Luo. Application of Cuckoo Algorithm in Flexible Workshop Scheduling Problem. Master of China West Normal University 2022-04-01.

[2] Tuo Wu. Research on Scheduling Optimization of Flexible Flow Workshop Based on Carbon Emission Model. Master of Guizhou University 2021-06-01.

[3] Shikui Zhao. Research on Flexible Resource Scheduling Optimization Method Based on Genetic Algorithm. Zhejiang University. 2013-10-01. Ph.D.

[4] Jiayi Wang; Ruilin Pan; Fei Qin. Improved genetic algorithm to solve flexible job shop scheduling problem. Manufacturing Automation. 2022-12-25 Journal.

[5] Yun Zou. Research on multi-time factor flexible operation workshop scheduling based on mixed discrete particle swarm optimization. Master of Dalian Jiaotong University. 2019-06-19.

[6] Qi Cui; Xiuli Wu; Jianjun Yu. Variable Neighborhood Improved Genetic Algorithm to Solve the Scheduling Problem of Mixed Assembly Workshop Computer Integrated Manufacturing System. 2017-09-15 Journal.

[7] Shengwang Yin; Yuexia Zhang. Research on Adaptive Step Value Fruit Fly Algorithm for Job Workshop Scheduling Problem. Measurement and Control Technology. 2018-01-18 Journal.

[8] Na Chen. Parameter optimization in image registration and fusion based on improved cuckoo algorithm. Hebei University. 2016-06-01 Master.

[9] Hongtao Tang; Jiayi Liu. Improved Cuckoo Algorithm Solves Distributed Flexible Assembly Workshop Scheduling Problems Considering Transportation Time. Operations Research and Management. 2021-11-24 Journal.