The Research of Flexible Scheduling of Workshop Based on Artificial Fish Swarm Algorithm and Knowledge Mining

Jieyang Peng[1], Jiahai Wang[1(✉)], Dongkun Wang[2], Andreas Kimmig[3], and Jivka Ovtcharova[3]

[1] College of Mechanical Engineering, Tongji University, Siping Road, Shanghai 200083, China
jhwang@tongji.edu.cn
[2] Universität Stuttgart, Keplerstr. 7, 70174 Stuttgart, Germany
[3] Karlsruhe Institute of Technology, 76133 Karlsruhe, Germany

Abstract. The Job Shop Scheduling problem is critical in the manufacturing industry. At present, the decision tree reasoning technique and data mining are often used in multi-objective optimization research to solve flexible job shop scheduling issues. Unfortunately, when job shop scheduling problems involve complex logic, it becomes difficult to implement data-driven automatic scheduling without human intervention. Based on the analysis of mass data and specialized knowledge in the scheduling domain, an ontology-based scheduling knowledge model and a method of knowledge representation can be established. Considering the relationship between data mining and knowledge, this paper illustrates the acquisition process of scheduling rules. These scheduling rules were applied to improve the initialization process of the artificial fish algorithm. Then, a scheduling experiment was designed, the results of which show that the efficiency and accuracy of the algorithm has been improved. The desired uncertain information analysis, decision-making support for production planning and scheduling on the shop floor are provided and an adaptive scheduling algorithm for complex manufacturing systems is established by building a knowledge-based system.

Keywords: Data-driven · Ontology · Knowledge mining · Production scheduling · Artificial fish swarm algorithm

1 Introduction

Massive production data contains abundant knowledge of production scheduling. Through the data mining technology, valuable rules can be obtained, which contribute to decision-making in the field of production scheduling. Under such urgent demand, how to analyze scheduling data has become a key issue.

For the optimal dispatching of the production workshop, the establishment of a multi-objective optimization model is currently an effective solution. There are many achievements in the study of job shop scheduling problems [1–3]. However, the following issues remain to be resolved: (1) The calculation accuracy is relatively low,

which brings certain difficulties to the implementation; (2) The solution procedure is inefficient. (3) Lack of knowledge-based guidance leads to limitations in the selection of scheduling optimization goals and rules. Therefore, it is necessary to study how to discover scheduling knowledge from the results of the scheduling algorithm [4], which aims at forming new scheduling rules, and apply them to the further optimization of the scheduling solution.

In recent years, the knowledge-based production scheduling method [5–9] has gradually become a research hotspot. However, due to the bottlenecks of knowledge acquisition, the knowledge-based scheduling method cannot be directly applied to the production scheduling field. To solve the above problems, an ontological knowledge representation method and a multi-objective optimization method for job shop scheduling based on knowledge mining is proposed, which improves on traditional knowledge representation and has become widely applied in the field of knowledge engineering.

2 Ontology-Based Scheduling Knowledge Representation

The representation of scheduling knowledge is key to realize knowledge mining of production scheduling. The purpose of establishing a scheduling ontology is to abstract the knowledge and describe the relationships between the entities.

2.1 Ontology Modeling of Scheduling Knowledge

To analyze the information in the field of shop scheduling, semantically label the knowledge on the basis of ontology, we sort out the basic relationship between the knowledge in the scheduling field, so that the scheduling knowledge can be understood and used by the controller.

Triplet C = <D, W, K> can be adopted to formalize the concept ontology, where C represents the set of elements in the domain, W represents the state set of related transactions in the domain, and K represents the conceptual relationship set in the domain space <D, W>. According to the meta-language proposed by Perez et al., including classes or concepts, relationships, functions, axioms, and examples, the core concepts selected from the field include: tasks, indicators, goals, constraints, rules, resources, and solutions. The attributes of the concept ontology include decomposition, satisfaction, guidance, implementation, distribution, and execution. Other relationships include subclass-of, has, and attribute-of.

Based on the above theory, the domain ontology of the JSSP (Job Shop Scheduling Problem) scheduling problem can be defined with a six-tuple:

$$\text{JSSP\_Ontology} = <C, A^C, R, A^R, H, X>$$

In the formula, C represents a concept set, A^C denotes sets of attributes, R represents set of conceptual relationships, A^R represents relational attribute set, H represents set of Conceptual dependency, and X denotes axiom set.

The Knowledge Ontology in the field of real-time scheduling described by core concepts and well-defined attributes is shown in Fig. 1. Based on the production tasks and

the Job Shop production scheduling theory, the task is often divided into several sub-tasks and assigned on different assembly-line in parallel to realize production allocation optimization. The scheduling rules are applied to guide the scheduling process under the constraint's conditions, and then the production resources are allocated according to the optimal scheduling scheme so that the task running cycle can be reduced.
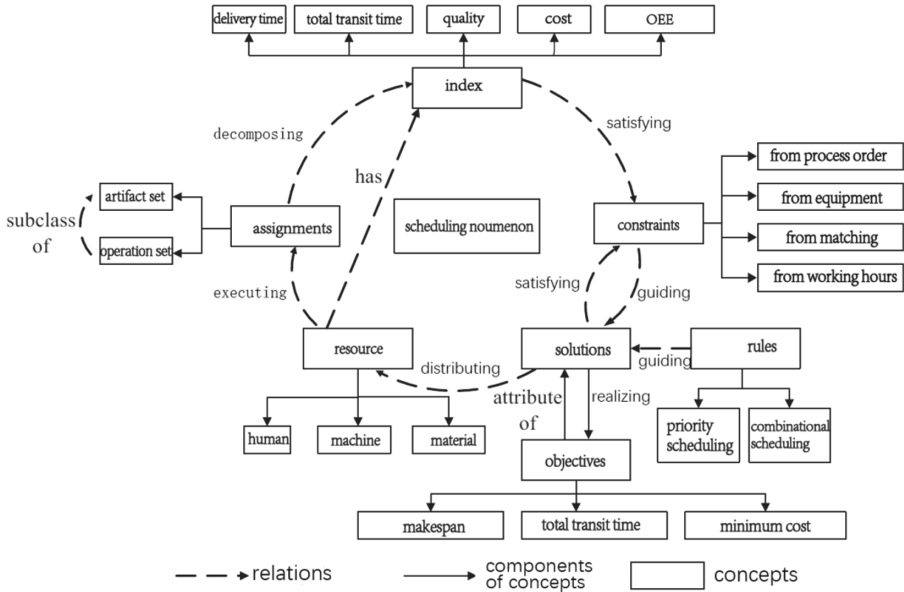

Fig. 1. Relational model of scheduling noumenon

2.2   Conception-Ontology and Attribute Representation in the Field of Scheduling

The ontology relationship diagram can directly represent the scheduling ontology model. However, in order to further investigate the constructing method of knowledge ontology, it is necessary to define its core concepts, essential attributes, and relational attributes. The ontology can be described by the following formula:

$$\text{JSSP\_ConceptOntology} = <N, P, \rho^n>$$

In the formula, N denotes the name of the concept; P denotes the attribute of the concept; ρ^n denotes the concept relation. The production scheduling task includes all machining tasks and its subtasks. The subtasks set should contain the corresponding process flow. The concept ontology of the task can be expressed as:

$$\text{JSSP\_TaskConcept} = <\text{Task\_Details, Task\_Attribute, Task\_Relation}>$$

In the formula, Task_Details includes the task's name, type (workpiece set/working procedure set), target number, and other specific information; Task_Attribute represents

attribute decomposition, which means the work piece set can be decomposed into process sets, and the process sets can be decomposed into work steps; Task_Relation represents the subordinate relationship between tasks, such as, working steps subset to the process.

In terms of production scheduling problems, there are many performance indicators, such as time metrics, production cost index, quality index, and OEE (Overall Equipment Effectiveness). When solving the multi-objective optimization problem, the weight balance between different indicators needs to be considered. Based on principles above, the target ontology can be defined as:

$$JSSP\_TargetConcept = <Target\_Sett, Target\_Attribute, Target\_Functions >$$

In the formula, Target_Sett represents the set of indicators; Target_Attribute represents the attribute such as coefficient and attribution function; Target_Functions denotes the relationship between different goals, that is, weight information of different indicators.

Based on the above indicators, production tasks are allocated to production resources according to the constraints and different scheduling rules, for example, rules based on the shortest machining cycle, rules based on the earliest delivery date, rules based on optimal shop condition etc. These scheduling rules can be defined as:

$$JSSP\_RuleConcept = <Rule\_Name, Rule\_Attribute, Rule\_Relation>$$

In the formula, Rule_Name represents the name and unique identifier of the scheduling rule; Rule_Attribute represents the content of the scheduling rule, that is, by comparing processing time, number of tasks, delivery time, and other factors to determine the product task priority list; Rule_Relation represents that the relationships between various scheduling rules, such as inclusion, combination, and confliction.

In the field of scheduling, the core concepts of resource ontology include workers, machines, materials, etc. The formal definition of the core concept is determined by the specific resource type. The concept of scheduling resource ontology can be defined with five tuples:

$$JSSPResourceOntology = <C(man, machines, materials),$$
$$A^C(Person\_Attribute, Machine\_Attribute, Material\_Attribute),$$
$$R(attribute, hierarchy, instance relations, etc.), H(parallel scheduling),$$
$$X(matching constraint, mutual exclusion constraint)>.$$

3 Association Rules Mining Based on Decision Tree

In data mining field, traditional approaches, for instance, machine learning techniques can be applied to mine implicit and explicit knowledge behind scheduling data [12, 13]. This paper applies the multivariate decision tree to extract the knowledge of the scheduling rules from the scheduling data sets, and assign a value to its attribute based on information theory. Different attribute values form decision tree branches, and the decision trees can be constructed recursively. Then, a certain kind of knowledge and rules can be obtained from the decision tree.

3.1 The Framework of Scheduling Knowledge Acquisition

Data-driven production scheduling optimization focuses on how to let the integration of data and knowledge play an important role. As is shown in Fig. 2, the scheduling knowledge mining system performs data conversion and pre-processing on the collected sample data. The generated rules are stored in the fuzzy rule base and updated on-line by a self-organizing procedure, which improves the knowledge base, and implementing knowledge-driven automation decisions at decision points, so that better results can be achieved.
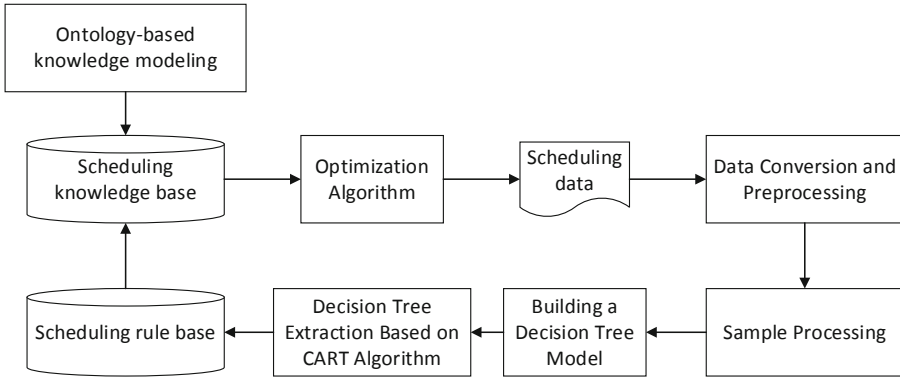

Fig. 2. Scheduling knowledge mining framework based on data mining

3.2 Rules Extraction of Decision Tree Based on CART Algorithm

Attribute is the branch point of the decision tree, and its selection strategy becomes the core of the decision tree construction. The most common attribute selection rules are the entropy-based information gain and the Gini Impurity Index. The CART algorithm determines the attributes of each node based on the Gini purity minimization classification criteria. The definition of the Gini Impurity Index is as follows:

If a schedule data set D has k classes, its Gini index is:

$$G(D) = 1 - \sum_{k=1}^{k} [f(k|D)]^2 \tag{1}$$

Where: D denotes the scheduling data set and f(k|D) denotes the probability of the k-class in the data set D. When the probability of the category of the node attribute is the same, the maximum value of $1 - 1/n$ is taken, and the minimum value of 0 means that there is only one category at the node.

The parent node of the decision tree selects the sample t. If the CART selects the attribute x into two sub-nodes, the corresponding sets are t_{v_1} and t_{v_2}. The split Gini Impurity Index is:

$$G(t, x) = \frac{|t_{v_1}|}{|t|} Gt_{v_1} + \frac{|t_{v_2}|}{|t|} Gt_{v_2} \tag{2}$$

$|\cdot|$ represents the number of records in the corresponding collection, and v_i represents different values of the attribute x.

At each node, calculate the Gini index of the attribute and select the attribute with the largest amount of reduction in impurity. The definition of impurity reduction is as follows:

$$\Delta = G(\text{parent}) - \sum\nolimits_{v \in V} f_{\text{parent},v} G(\text{child}|v) \tag{3}$$

In the above formula, V is a set of all attribute values corresponding to the attribute. $f_{\text{parent},v}$ represents the probability that the instance satisfied by the parent node attribute value v accounts for the entire data set, and G(child|v) indicates that the child node attribute value satisfies Gini Impurity Index of v.

The specific steps of the algorithm are as follows:

Step 1: Convert and pre-process the scheduling data set to determine the attributes, attribute values, and category values.

Step 2: Select the training set and test set that contain all classes from the sample set;

Step 3: The parent node corresponds to the sample set t, and the attribute with the least Gini impurity is taken as the root node of the branch.

Step 4: In each subset, select the attribute with the greatest amount of reduction in impurity as the root node of the branch. Build the decision tree recursively. If the subset contains only one category, the split ends.

Step 5: The test set can be used to determine the error in the decision tree in step 4. If there is an over-fitting, then use the post-prune to optimize the decision tree.

Taking the JSSP scheduling problem of 6×6 scale as an example, 300 groups of scheduling data are selected obtained by the optimization algorithm to constitute the scheduling data set, among which 250 groups of data are used to train the classifier, and the remaining 50 are used to test the decision tree. The premise of the CART algorithm is the discreteness of the data, and it is necessary to discretize the continuous data in the samples. For 6×6 scheduling problems, the goal is to meet the earliest delivery data and shortest processing time. Select the indicator related to the target as the input attribute of the decision tree, take the data of each sample's corresponding indicator as the attribute value, use the sorting status of the work piece waiting for processing on the same machine tool as a category, and the numerical data can be used to represent the sorting and serve as the category value. The structure of the processed scheduling data set is shown in Table 1.

The CART algorithm uses post-pruning method to prevent over-fitting of the decision tree. After removing the leaf node of the same parent node, the impurity error is smaller, so starting from the leaf node, remove the leaf nodes and re-starting the cycle calculation from the parent node, as shown in Fig. 3. The decision tree as shown in Fig. 3 can be represented as the "if-then" rule combination as in Table 2.

Table 1. Data structure of MOO data set (CR = (Time of Delivery − current time)/processing time)

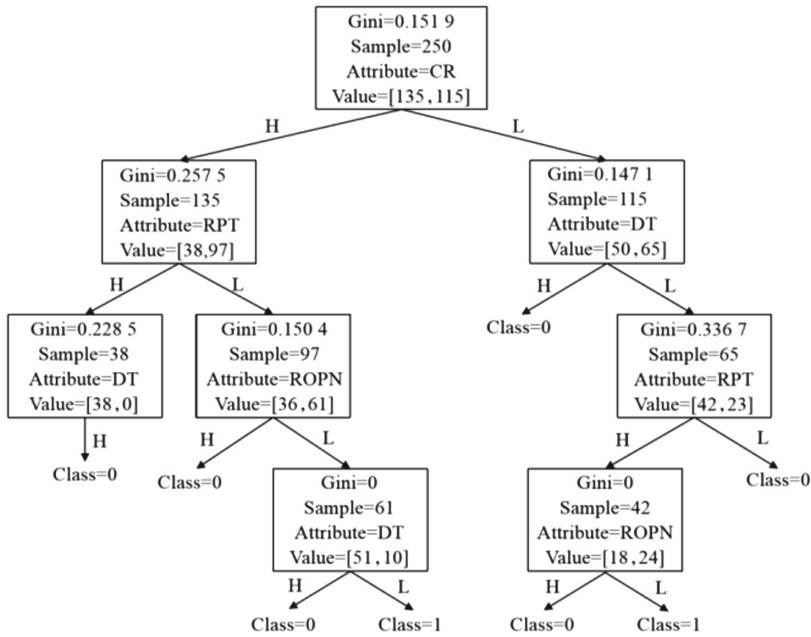| Attributes	Attributes value	Type value
RPT (remaining process time)	H (high)	0 (executed first)
ROPN (remaining operation process number)	L (low)	1 (executed later)
DT (delivery time)	H (high)	0 (executed first)
CR (critical ratio)	H (high)	0 (executed first)

Fig. 3. CART tree diagram with the lowest impurity classification

4 Knowledge-Driven Production Scheduling Optimization

In this chapter, a method of improving initialization process of artificial fish swarm algorithm is proposed based on the data-based scheduling rules mining and scheduling knowledge representation.

4.1 Target Model

The target model of knowledge-driven Production Scheduling Optimization problem can be summarized as follows. If a job shop has n work pieces to be processed, the work piece set is N = {1, 2, . . . , n}; there are m sets of machining equipment, and the equipment set is M = {1, 2, . . . , m}; each work piece contains a variety of processes

Table 2. If-Then decision tree rules

No.	If	Then
1	CR=L, DT=H	result=0
2	CR=L, DT=L, RPT=L	result=0
3	CR=L, DT=L, RPT=H, ROPN=L	result=1
4	CR=H, RPT=L, ROPN=H	result=0
5	CR=H, RPT=H, DT=H	result=0
6	CR=H, RPT=L, ROPN=L, DT=L	result=1

j, $1 \leq j \leq m$. $O_{ijk} = 1$ represents that the jth process of the work piece i should be processed on the kth machine, the process starting time is S_{ijk}, the processing time is T_{ijk}, time of completion is E_{ijk}. The machine start time of the last process of the workpiece is LS_{ijk}, and the processing time is LT_{ijk}. The set of processes waiting for the machine at the same time on device k is combined as $[O_{ijk}, O_{nmk}]$, and the priority of the process is expressed as a binary value. $[O_{ijk}, O_{nmk}] = [1, 0]$ means that O_{ijk} is processed on device k first.

Based on the above features, the scheduling objective function can be defined as:

$$F = min\left[\max_{1 \leq i \leq n} \left(LS_{ijk} + LT_{ijk}\right) \right] \quad (4)$$

During the processing, some process constraints must be satisfied:

(1) One device can only operate one process at a time:

$$\sum_{i=1}^{n} O_{ijk} = 1 \ (k = 1, 2, 3, \ldots, m) \quad (5)$$

(2) One work piece can only be processed on one device at the same time;

$$\sum_{k=1}^{m} O_{ijk} = 1 \ (i = 1, 2, 3, \ldots, n) \quad (6)$$

(3) The completion time is equal to the start time plus the processing time:

$$E_{ijk} = S_{ijk} + T_{ijk} \quad (7)$$

(4) Only after the previous procedure of the same workpiece is completed can the next procedure be started:

$$S_{i(j+1)} > E_{ij} \quad (8)$$

Besides, the processes between different work pieces are independent of each other, the process of a work piece is determined and cannot be changed;

4.2   Artificial Fish Swarm Algorithm with Improved Initial Population

In the traditional artificial fish swarm algorithm, artificial fish populations are initialized randomly [14], resulting in a slow initial search speed and a slow convergence in the later period of the algorithm, which reduces the convergence speed of the algorithm. For the JSSP scheduling problem, combined with the application of decision tree scheduling rule, the scheduling knowledge can be used to initialize the process sequencing at each decision point. Then, the IF-THEN rules generated by the decision tree are used to compare the attribute value of different processes so as to determine the execution sequence of the processes.

The knowledge-driven initial population optimization method for artificial fish swarm algorithm is used to minimize the total processing time. The algorithm flowchart is shown in Fig. 4.
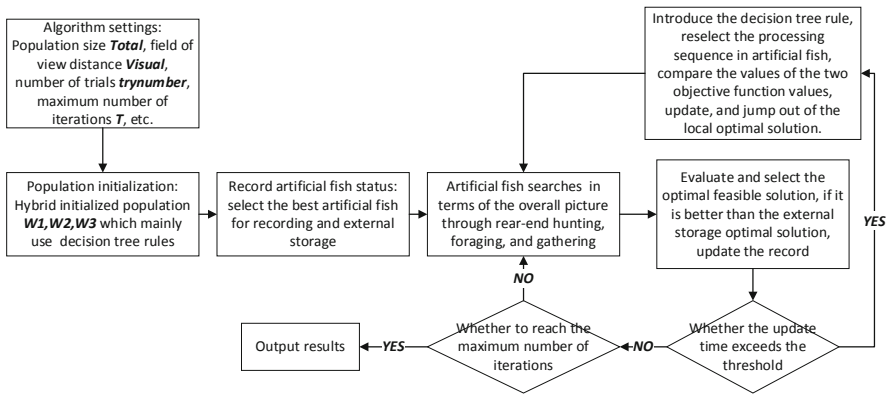

Fig. 4. Improved artificial fish swarm algorithm based on scheduling knowledge

5   Experiment Analysis

In order to verify the validity of the decision tree rules and the improvement of the decision tree rules on the artificial fish swarm algorithm in solution to the scheduling problem, the 6 × 6 and 8 × 8 JSSP scheduling problem is taken as an example.

The object model of 6 × 6 JSSP and constraint conditions are described in Sect. 4.1, while the model of 8 × 8 JSSP is derived from real workshop data in reference [15]. The scheduling data of 6 × 6 JSSP is shown in Table 3, in which 5 (130) indicates that the fifth process of the workpiece 1 is processed on the equipment M1, and the processing time is 130.

The algorithm parameters are set as follows: population size Total = 40, visual distance Visual = 6×6 = 36, crowd factor δ = 9, maximum number of iteration T = 20, tentative number try number = 30.

Table 3. 6 × 6 Scheduling data

Work piece	Machines					
	M1	M2	M3	M4	M5	M6
1	5(130)	3(240)	6(180)	4(270)	1(90)	2(180)
2	1(200)	4(160)	3(300)	5(240)	2(190)	6(100)
3	3(250)	5(80)	6(120)	2(110)	1(90)	4(280)
4	1(130)	3(240)	5(120)	4(270)	6(150)	2(90)
5	2(90)	4(140)	3(190)	5(140)	1(230)	6(100)
6	1(150)	6(190)	5(180)	2(190)	3(190)	4(90)

We perform 10 repeated simulations of the above two cases, and compare them with other scheduling rules, as showed in Table 4 and Table 5. It turned out that compared with other algorithms, the improved artificial fish swarm algorithm has a smaller variance, a better stability and robustness, and the optimal objective function value can be obtained in a relatively short time, which means the new algorithm has a higher efficiency.

Table 4. Comparison of results of algorithms (6 × 6 Scheduling data)

Optimization method	Total processing time/s	Variance	Average running time/s	Frequency of simulation
Traditional Artificial Fish Swarm-Algorithm	1660	1.53	5.42	10
Improved Artificial Fish-Swarm Algorithm	1560	0.38	4.10	10
Genetic algorithm	1690	1.46	4.76	10
Priority scheduling rules	1710	0	7.22	10

Table 5. Comparison of results of algorithms (8 × 8 Scheduling data)

Objective function	Genetic algorithm	Ant colony algorithm	Traditional Artificial Fish Swarm-Algorithm	Improved Artificial Fish-Swarm Algorithm
Minimum completion time	16	15	16	14

Figure 5 and Fig. 6 show the optimal scheduling sequence of the two algorithms through the Gantt chart. The Gantt chart reveals that due to the reduction of waiting time, the operation sequence obtained by knowledge-driven artificial fish swarm algorithm has higher equipment utilization and the make span is minimizing. Thus, it's a better scheduling solution.
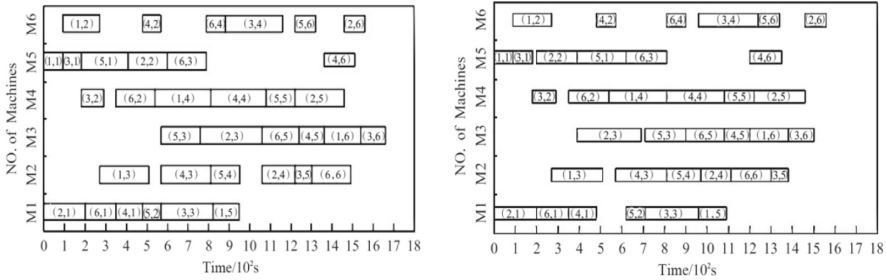

Fig. 5. Gantt-chart of traditional and improved Artificial Fish Swarm Algorithm (6 × 6 Scheduling data)
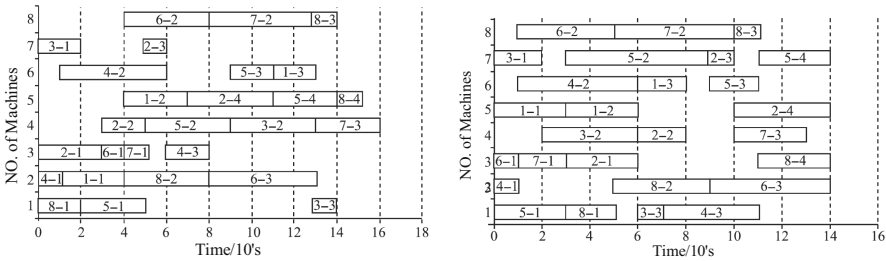

Fig. 6. Gantt-chart of traditional and improved Artificial Fish Swarm Algorithm (8 × 8 Scheduling data)

The above experiment re-processes the optimization problem by using the decision tree rule as algorithm constraint conditions. The optimized results are taken as new samples to rebuild the initial decision tree classifier so that the improved decision tree classification with higher accuracy rate is better performing. The rules of the decision tree after training are shown in Table 6.

Table 6. Decision tree rules after training

No.	If	Then
1	CR=L, DT=H	result=0
2	CR=L, DT=L, RPT=L	result=0
3	CR=H, RPT=L, ROPN=H	result=1
4	CR=H, RPT=H	result=0
6	CR=H, RPT=L, ROPN=L, DT=L	result=1

6   Conclusion

Aiming at the limitation of knowledge acquisition and application of a manufacturing system, this paper summarizes the professional experience and knowledge in the field of scheduling. From the aspect of knowledge representation, we adopt an ontology-based method. This is to represent the scheduling knowledge in the form of conception and formalization, to advance the cooperation of heterogeneous systems and prompt the communion of the scheduling knowledge. Secondly, it integrates CART-based learning algorithms to extract a decision tree classification rule set and a class association rule set respectively so that the dynamic scheduling rules Library can be built. Based on this, we study the effective combination of decision tree rules and artificial fish swarm algorithm to realize the acquisition of scheduling knowledge, and then use the knowledge to realize the optimization of production scheduling. Finally, a computer simulation experiment is designed to test the effectiveness and practicability of the solution. The simulation results show that knowledge-driven artificial fish-swarm algorithm can significantly improve the solution of the Job-Shop Scheduling Problem. The results also have some reference value to develop the scheduling system and improve the productivity and resource utilization of workshop in practical application.

References

1. Xu, B., Fei, X., Zhang, X.: Batch division and parallel scheduling optimization of flexible job shop. Comput. Integr. Manuf. Syst. **22**(8), 1953–1964 (2016)
2. Meeran, S., Morshed, M.S.: A hybrid genetic tabu search algorithm for solving job shop scheduling problems: a case study. J. Intell. Manuf. **23**(4), 1063–1078 (2012)
3. Cupek, R., Ziebinski, A., Huczala, L., et al.: Agent-based manufacturing execution systems for short-series production scheduling. Comput. Ind. **82**, 245–258 (2016)
4. Yan, H., Liu, F.: Knowledgeable manufacturing system-a new kind of advanced manufacturing system. Comput. Integr. Manuf. Syst. **7**(8), 7–11 (2001)
5. Yao, L., Wang, Z., Mu, H.: Knowledge-based optimization of manufacturing system and implementation under ASP.net. Comput. Technol. Dev. **21**(11), 1–3 (2011)
6. Wang, H., Yan, H.: Interoperable dynamic adaptive scheduling strategy in knowledgeable manufacturing based on multi-agent. Control Decis. **28**(2), 161–168 (2013)
7. Vegetti, M., Roldán, L., Gonnet, S., et al.: A framework to represent, capture, and trace ontology development processes. Eng. Appl. Artif. Intell. **56**, 230–249 (2016)

8. Zhang, X., Qiu, J.: Research on knowledge fusion framework in big data environment. Libr. Sci. Res. (8), 66–70 (2016)
9. Wu, Q.: Ontology-based domain knowledge representation and acquirement. Comput. Eng. Appl. **41**(31), 23–25 (2005)
10. Bandaru, S., Ng, A.H.C., Deb, K.: Data mining methods for knowledge discovery in multi-objective optimization: Part A - Survey. Expert Syst. Appl. **70**, 139–159 (2017)
11. Bandaru, S., Ng, A.H.C., Deb, K.: Data mining methods for knowledge discovery in multi-objective optimization: Part B - New developments and applications. Expert Syst. Appl. **70**, 119–138 (2017)
12. Zhao, M., Yin, H., Sun, D., et al.: Flexible job shop scheduling problem based on modified artificial fish swarm alogorithm. China Mech. Eng. **27**(8), 1059–1065 (2016)
13. Chen, X.: Research of Flexible Job Shop Scheduling Problem based on Artificial Fish Swarm Algorithm. Dalian University of Technology (2015)
14. Sun, F., Zhang, J.: Research on modified artificial fish swarm algorithm under framework of cultural algorithm. Comput. Simul. **31**(4), 407–411 (2014)
15. Kacem, I., Hammadi, S., Borne, P.: Approach by localization and multi-objective evolutionary optimization for flexible job-shop scheduling problems. IEEE Trans. Syst. Cybern. Part C **32**(1), 408–419 (2002)