2019 International Conference on High Performance Big Data and Intelligent Systems (HPBD&IS)

Scheduling Feature Selection for Data-driven Job Shop Scheduling System Using Improved Firefly Algorithm Optimization

Rong Luo[1]*, Lei Liu[1], Dong Tan[2], Sheng Yin[1]
[1]*The School of Advanced Manufacturing Engineering, Chongqing University of Posts and Telecommunications, Chongqing, China*
[2]*The Department of Technical, Chongqing Radio and Television Digital Media Co, Ltd., Chongqing, China*
luorong@cqupt.edu.cn, ORCID 0000-0002-5283-2090, 852041173@qq.com, yinsheng@cqupt.edu.cn

Abstract—**This study proposes an improved version of the Firefly Algorithm (FA) for scheduling feature selection optimization in data-driven job shop scheduling system. The proposed FA variant employs a dynamic step strategy with the elite individual dipartition in firefly population to improve the optimization ability of FA. Selecting essential schedule feature data from production system attributes data based on various production requirements to construct data-driven job shop scheduling system is a critical issue because of the existence of irrelevant and redundant attributes in job shop production system, by selecting the important attributes as schedule features, robust performance can be expected in data-driven job shop scheduling system. A wrapped scheduling feature selection approach based on the proposed FA variant and extreme learning machine (ELM) is presented for ELM-based job shop scheduling system. The feasibility and effectiveness of the proposed scheduling feature selection approach have been verified via a practical job shop scheduling case.**

Keywords—job shop scheduling, feature selection, firefly algorithm, extreme learning machine

I. INTRODUCTION

The job shop scheduling problem (JSP) has been widely concerned because of its strong theoretical foundation and important practical value since it was introduced firstly in 1950s [1]. The job shop scheduling in real world often has a problem that the established production scheduling scheme cannot keep up with production changes (such as equipment failure, temporary customer order insertion, and change of delivery time, etc.), because the scheduling scheme is done in the state of ideal constraints. When the complexity of manufacturing process, production environment, and production scale increases with much uncertainty, JSP becomes extremely difficult to solve by the traditional operation optimization methods. Although a variety of heuristics or precise algorithms theoretically are used to solve various benchmark job shop scheduling problems successfully, they are not applicable to the requirements of dynamic job shop scheduling in practical job shop production environment. Dispatching rules for JSP have low time complexity and the ability to respond to dynamic production changes in real time, but one widely accepted view is that no single rule is better than other rules under all performance indicators [2]. The efficiency of dispatching rules depends on production system characteristics, processing condition and scheduling objective. The successful application of data mining and machine learning technology in many fields has enabled the academic community to pay attention to the role of production data in solving JSP [3-5]. The production information systems in factory such as ERP (Enterprise Resource Planning), MES (Manufacturing Execution System), and SCADA (Supervisory Control And Data Acquisition) can provide data source assurance for data-driven job shop scheduling. It is feasible for data-driven job shop scheduling system to adopt a dispatching rule adaptive conversion mechanism that is implemented with machine learning classification models, and the scheduling system based on machine learning models can output an optimal dispatching rule in real time for current scheduling task according to the production status data of job shop, whereas describing the status of the job shop production with feature data(from the point of view of information representation in data science) selected from production information system attributes data(from the standpoint of system operation status) for the schedule system based on machine learning is challenging because of the existence of irrelevant and redundant attributes for job shop scheduling[6]. In machine learning task, using too many features to learn a concept causes over-fitting of training data and degrades the generalization ability of machine learning model, which leads to inaccurate classification and long learning time, therefore feature (attribute) selection is necessary for the "feature dimension disaster" problem. Similarly, omitting one important scheduling feature will harm the ability of the data-driven scheduling system. Feature selection is actually a combination optimization problem [7-8], and some meta-heuristic or bio-heuristic optimization algorithms have shown good performance in solving combination optimization problems including feature selection problem [9-12]. This work proposes an improved version of the Firefly Algorithm (FA) for scheduling feature selection optimization in data-driven job shop scheduling system.

In Section II, a data-driven job shop scheduling mechanism is introduced. After that, the essential stage of this mechanism, the scheduling system construction based on scheduling feature selection is studied in Section III. The validity of the method we propose is illustrated by the test of JSP case simulation and the test result shows in Section IV. Finally, Section V briefly explains the conclusions of this paper and future work.

This work was supported by the National Natural Science Foundation of China (NO.51805066).
*Corresponding author

978-1-7281-0467-6/19/$31.00 ©2019 IEEE

II. DATA-DRIVEN JOB SHOP SCHEDULING MECHANISM

A basic feasible mechanism of data-driven job shop scheduling is to use the historical optimally scheduling scheme data stored in production information system as benchmarking reference and real-time scheduling feature data derived from production information system as scheduling constrains and objectives, which can give better dispatching rule patterns to production managers for current job shop scheduling optimization. The scheduling mechanism adopted mainly includes three stages as shown in Fig. 1: black flow line for scheduling system construction stage, blue flow line for optimal dispatching rule decision stage and orange flow line for optimal scheduling scheme generation stage.

A. Scheduling System Construction

By mining historical scheduling scheme data, the knowledge of job shop scheduling can be obtained. The scheduling knowledge mainly includes job shop scheduling feature set (a subset of production system attributes), job shop scheduling performance criterion set, and optimal dispatching rule set. The scheduling knowledge is learned through machine learning method, which fits the programming relationship between scheduling features and optimal dispatching rules under certain scheduling performance criterion for building scheduling system of optimal dispatching rule decision making. Scheduling feature selection from production system attributes of job shop is the focus of this paper that proposes a dynamic step firefly algorithm with the elite individual dipartition for the selection of optimal scheduling features under certain scheduling performance criterion. Attention that even for the same production line in job shop, the scheduling feature set may be different due to different scheduling goals.

B. Optimal Dispatching Rule Decision-making

After the scheduling system is successfully established, it can be used for online (real-time) dispatching rule decision making. When it is at decision point (the occurrence time of work-piece processing conflicts, temporary orders insertion, equipment failures, etc.), the original job scheduling strategy is not suitable for the new situation, thus job shop production needs a new decision making for the most suitable dispatching rule. Production information system obtains current scheduling features data specified by the scheduling knowledge under certain scheduling performance criterion, and these real-time scheduling feature data is inputted into the scheduling system based on machine learning model for dispatching rule decision making that outputs the optimal dispatching rules that match current job shop production system status well.

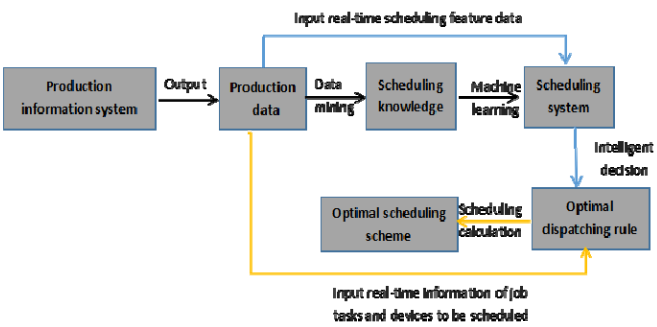

Fig. 1. Data-driven job shop scheduling mechanism

C. Optimal Scheduling Scheme Generation

The current job shop optimal dispatching rule obtained by the scheduling system is used to schedule work-piece and equipment. Then a set of optimized processing priority sequences of the combination of work-piece and equipment is generated, which is used as the optimal scheduling scheme that guides current job shop scheduling.

III. SCHEDULING SYSTEM CONSTRUCTION BASED ON SCHEDULING FEATURE SELECTION

A. The Improved FA for Scheduling Feature Selection

FA, a swarm intelligence algorithm inspired by natural behaviours of fireflies, is first proposed by Yang [13]. The basic idea of FA is as follows. Each firefly individual represents a solution to the problem to be solved, and the firefly brightness is related to the objective function value of the problem to be solved, that is, the brightness of firefly is stronger, then the objective function value of the solution is better. When the algorithm is running, a firefly with less brightness is attracted towards other fireflies with more brightness. As the algorithm iteration continues, the fireflies with weaker brightness in population move toward the brighter fireflies continuously. Finally, most of fireflies can gather around the brightest firefly, and the position of the brightest firefly represents optimal solution.

1) The elite individual dipartition-based dynamic step strategy

In original FA, the movement step of all fireflies takes a fixed value and cannot be adjusted according to the fireflies' actual search situation, which has some limitations for the adaptive optimization and avoiding the local optimal trap. If a larger step is taken for all fireflies, it will improve the global exploration ability of FA in solution space, but it is easy to skip the global optimization solution. If a smaller step is taken, it can improve the local mining capacity of FA in solution space, but it may reduce the convergence speed of FA. This study proposes the elite individual dipartition dynamic step FA(EDSFA) that adopts a larger step for better performing fireflies viewed as the elite fireflies to maintain their global exploration ability in larger solution space and uses a linearly reduced step for the remaining fireflies viewed as non-elite ones to enhance their performance on local search. This dynamic step strategy takes the performance difference of firefly individual into account, making the fireflies with overall better performance move faster for further global exploration and the ones with poor performance move more slowly for fine local search. The specific equations of this dynamic step strategy are designed as in (1), (2).

$$\alpha_i(t+1) = \begin{cases} \alpha_i(t)+\alpha_0 * \lambda/\mathrm{MaxGen} & \text{if } I_i > \theta * I_{best} \\ \alpha_i(t)-\alpha_0/\mathrm{MaxGen} & \text{others} \end{cases} \tag{1}$$

$$\alpha_i(t+1) = \alpha_0, \text{if } \alpha_i(t) > 1 \tag{2}$$

Where $\alpha_i(t+1)$ indicates the current step of the firefly i at the t+1th iteration, MaxGen represents the max iteration number of the algorithm, α_0 indicates the initial uniform step for all fireflies, λ represents the stochastic acceleration for increasing step and λ takes a random integer in[1,MaxGen/2]. I_i is the brightness of the firefly i, while I_{best} is the brightness of the best individual in current population. θ is the threshold for dividing the elite fireflies and non-elite ones, and θ takes a constant value in [0.85,0.95]. In (1),when the brightness of the firefly i is θ times greater than the brightness of the best individual in current population, it would be regarded as an elite firefly, then its step value is increased, otherwise the step value is linearly reduced. If the step value of firefly i is greater than 1 after increasing step, its step value would be reset as the initial step value, as in (2). In (1), the stochastic acceleration λ for increasing the step of fireflies has a significant effect on promoting the diversity of the population and avoiding the premature convergence of the algorithm search. In addition, λ can take a value with the sliding window value according to the specific dimensions of solution space, which makes the step adjustment more flexible.

2) Fireflies representation and initialization

Each firefly individual, a N-dimensional binary vector as its position vector, is a kind of production system attribute combination as illustrated in Fig. 2. Each bit of firefly vector is corresponding to a candidate production system attribute. The value "1" represents that the attribute has been chosen as a scheduling feature, while value "0" indicates that the attribute is not selected. For instance, in the firefly individual shown in Fig.2, the second attribute is selected to be a scheduling feature, yet the first one is not. The initial population of fireflies is generated randomly, and the bit positions for each firefly are randomly assigned as 1 or 0. The length of each firefly's bit positions is equal to the number of all candidate production system attributes.

3) Objective function

The goal of scheduling feature selection is to achieve the same or better classification effect for dispatching rule classifier using a small number of scheduling features, therefore the evaluation of feature sets consists of two parts: (a)The classification accuracy of dispatching rule classifier. The classifier is trained using the features data determined in the feature sets. (b)The number of feature selected. Considering the impact of the above two aspects, the objective function (the fitness) is designed as in (3).

$$f(accu, num \mid X_i) = \frac{10^3}{10^4 \times (1-accu) + \mathrm{k} \times num} \tag{3}$$

Where $accu$ is the dispatching rule classification accuracy obtained through each firefly individual (candidate feature set), and the 10 folds cross-validation result of the dispatching rule classifier is used as the accuracy value, then num is the number of '1' in each firefly binary vector(Candidate feature set). The accuracy weight is set as 10000 to pay more attention to the accuracy. In (3), K is a compromise parameter giving balance between the classification accuracy of dispatching rule classifier and the number of scheduling features selected. The larger value of K, the more attention is paid to the number of scheduling features selected. The value of K is 0.5 in the following experiment. In addition, the value of objective function is used as the brightness value of the firefly individual.

4) The attractiveness of fireflies

The distance r_{ij} between two fireflies X_i and X_j is defined based on the similarity ratio of the two fireflies' binary position vectors. The similarity ratio is calculated by the normalized Hamming distance of the two position vectors as in (4).

$$r_{i,j} = 1 - \frac{\sum_{\mathrm{k}=1}^{\mathrm{d}} \left| x_i^k \otimes x_j^k \right|}{\mathrm{d}} \tag{4}$$

Where x_{i}^k and x_{j}^k represent the k-th dimension component of the binary position vectors of two fireflies X_i and X_j respectively, $\otimes$ denotes the XOR operation and d is the position vector dimension.

The attractiveness β between a pair of fireflies is calculated as in (5).

$$\beta(r) = \beta_0 \mathrm{e}^{-\gamma r^2} \tag{5}$$

Where β_0 is the attractiveness at r = 0 and γ is the light absorption coefficient. The distance rij between two fireflies is determined as in (4).

5) The binary movement of fireflies

The original FA is designed for optimization problems with continuous variables. Given the fact that each firefly is a binary vector for feature selection problem, thus the movement of fireflies should be implemented by the binary movement operator rather than the continuous one. In this study, the new definition of the movement of fireflies is proposed, as in (6) and (7), similar to the approach used in [14].

$$\mathrm{v}_i^k = \begin{cases} x_j^k, \text{if } x_i^k \neq x_j^k \text{and } \beta > \mathrm{rand}(0,1) \\ x_i^k, \text{others} \end{cases} \tag{6}$$

$$x_i^k = \begin{cases} 1 - v_i^k, \text{if } \alpha_i > \mathrm{rand}(0,1) \\ v_i^k, \text{others} \end{cases} \tag{7}$$

When a firefly X_i moves to another firefly X_j more attractive, every bit in its position vector will make a decision to change its value or not. Changing a bit x_{i}^k in X_i is done in two steps: the β-step (attraction movement) as in (6), which is regulated by the attractiveness β, and the α-step (mutation movement) as in (7), which is controlled by the current step parameter α_i of firefly X_i.

B. Scheduling Feature Selection-based Scheduling System Construction

The proposed EDSFA is used for scheduling feature selection to optimize the performance of the job shop dispatching rule classifier based on extreme learning machine (ELM). The ELM-based dispatching rule classifier under some production performance criterion is to output current optimal dispatching rule for job shop scheduling optimization and the data-driven job shop scheduling system is built with a series of ELM-based dispatching rule classifiers that meet different production performance criterion.

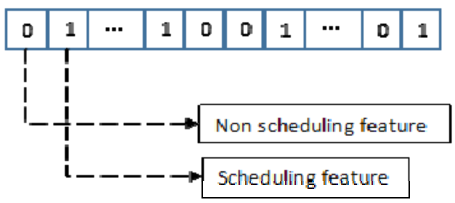

Fig. 2. Firefly binary vector representation

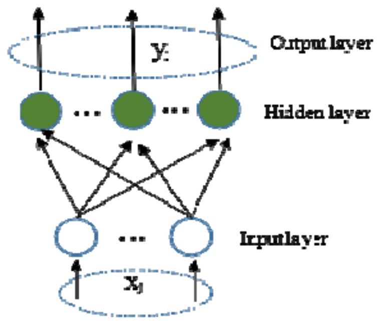

Fig. 3. The structure of ELM-based classifier

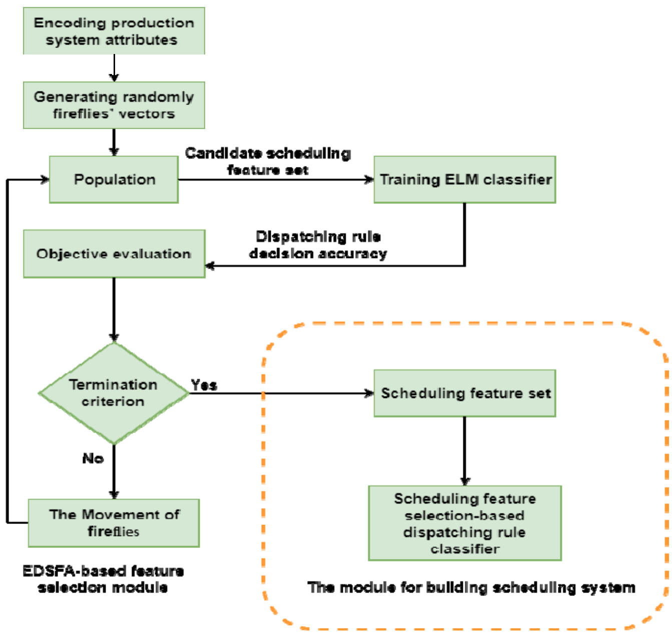

Fig. 4. Scheduling feature selection for scheduling system

1) ELM-based classifier for dispatching rule decision

ELM is a kind of machine learning algorithm based on feed forward neuron network and its main characteristics is that the hidden layer node parameters can be random or artificially given without the adjustment in train or learning process [15]. The learning process only needs to calculate the output layer weight values. ELM has the advantages of high learning efficiency and strong generalization ability, therefore it is widely used in the machine learning tasks [16-18]. The ELM-based classifier for dispatching rule decision making is designed as shown in Fig. 3.

The X vector of the input layer is the scheduling feature vector obtained by the feature selection and each dimension component of the y vector of the output layer is a real number between (0,1), which indicates the probability of selecting corresponding dispatching rule (strategy) in each dimension. Then the dispatching rule with the largest probability in some dimension of y vector is output as the current optimal scheduling strategy for job shop scheduling task.

2) Scheduling system construction based on scheduling feature selection

A wrapped feature selection process based on the proposed EDSFA and ELM is presented as shown in Fig.4. The features set selected from the original attributes set by EDSFA, then the features set as candidate scheduling feature set is taken to the ELM classifier for evaluation, that is, the ELM classifier is trained with the sample data specified by the feature set. After several iterations of scheduling feature selection optimization with EDSFA, the ELM-based dispatching rule classifier with robust generalization ability and better precision will be found when it has its training done with the optimal scheduling feature set under some production performance criterion. The data-driven job shop scheduling system is composed of a set of the ELM-based classifiers that has finished training with the optimal scheduling feature sets under all production performance criteria.

IV. EXPERIMENTS AND RESULTS

A. Sample Data of Job Shop Scheduling Problem

The job shop scheduling problem comes from the benchmark job shop scheduling case ft10 [19] that has a scale of 10 kind of jobs and 10 processing equipment. The specific introduction of the scheduling task and job shop condition is available in [19]. A JSP simulation model JSP-ft10simsys has been developed based on the Siemens Tecnomatix Plant Simulation14.1 platform. With the sufficient ft10 simulation sample data obtained by JSP-ft10simsys, 300 optimized scheduling samples are randomly selected to form the experiment data set that has 3 scheduling performance criteria of TP(Throughput), MCT(Mean cycle time), and NT (Number of tardy parts) and respectively contains 100 optimized scheduling samples under every scheduling performance criterion. An optimized scheduling sample is a 31-dimensional numerical vector consisted of 30 production system attribute values and a corresponding optimal dispatching rule id as classification label. The scheduling performance criteria, 30 production system attributes and 6 dispatching rules have been defined in [19].

B. Experiment Settings

The experiment sample data fall into two sets in disorder: seventy percent being the training set and thirty percent being the test set. For experiment comparison, GA (genetic algorithm), PSO (particle swarm optimization), and FA (firefly algorithm) are considered, and their scheduling feature selection methods are implemented in the same wrapped feature selection process as the proposed EDSFA. The proper parameters settings of PSO, GA, and FA are respectively as in [12], [19], and [14]. The parameters setting of EDSFA is consistent with the FA except for its step parameter setting according to section III in this paper. The main parameters of these algorithms are set as in TABLE I, in which the population size and the max iteration of these algorithms are the empirically optimal settings given by the literature. The experiment goes as follows: The training set data is input into the procedures of GA-ELM, PSO-ELM, FA-ELM, and EDSFA-ELM for scheduling feature selection, then the decision making of dispatching rules based on the selected scheduling features is executed on test set. For a further comparison, the ELM classifier is used for the dispatching rules decision making with all candidate production system attributes data on test set. In experiment, the number of neurons in ELM hidden layer is 50, and the activation function of ELM is the Sigmoid function. In order to obtain objective calculation results, all algorithms procedures are run for 20 times.

C. Results and Analysis

In Fig. 5, it shows the average fitness (optimization objective value) of the scheduling features sets obtained by the 20 runs of GA-ELM, PSO-ELM, FA-ELM and EDSFA-ELM on training set. The scheduling features sets are the subsets of candidate production system attributes and represented in the form of the global optimal individual in the population of GA, PSO, FA and EDSFA. The fitness of scheduling features sets can be calculated as in (3). It is obvious that the proposed EDSFA has a better performance than FA, PSO and GA in the optimization of scheduling features selection by the wrapped method, that is, EDSFA, FA, PSO and GA are combined with ELM respectively for feature selection. It should be noted that the FA-ELM does not perform as well as GA-ELM in terms of optimal scheduling feature selection under TP, MCT and NT, however, the EDSFA-ELM finds the scheduling features sets with higher fitness compared to GA, PSO and FA under all performance criteria as shown in Fig. 5.

TABLE II gives the results such as the mean(Ave), best value(Best) and standard deviation(SD) of dispatching rules classification accuracy obtained from the 20 runs of the dispatching rules decision procedures that use ELM classifier on test set respectively in combination with the different scheduling features sets given by EDSFA-ELM, FA-ELM, PSO-ELM and GA-ELM on training set. The accuracy of ELM classifier with the scheduling features selection based on EDSFA, FA, PSO and GA performs much better compared to all attributes-based ELM without feature selection. And EDSFA-ELM achieves the most excellent classification accuracy among the five algorithms on the test sample data set under all performance criteria. The classification performance of EDSFA-ELM is similar to FA-ELM and GA-ELM under NT, whereas it performs better than FA-ELM and GA-ELM under TP, MCT, moreover the performance of PSO-ELM is worse than EDSFA-ELM under TP, MCT and NT, which proves that the EDSFA has the advantage of searching optimal solution in solution space adaptively that is more excellent in comparison with FA, PSO and GA. The standard deviation (SD) is minimum for EDSFA-ELM under TP and MCT than the other algorithms. The EDSFA-ELM also gets a small standard deviation of the classification accuracy under NT, which proves that EDSFA outperforms the other comparison algorithms in the stability and ability of seeking optimal solution.

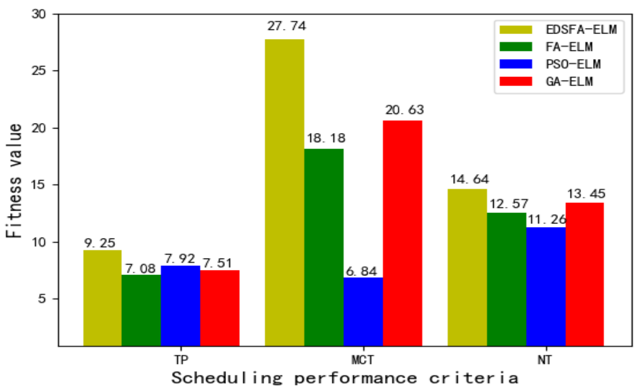

Fig. 5. The average fitness of the selected scheduling features sets after 20 runs

TABLE I. ALGORITHM PARAMETERS

Algorithm	Population Size	Max Iteration	Value of Other Parameters
GA[19]	500	500	The mutation probability,p1=0.005 The crossover probability,p2=0.6
PSO[12]	150	500	The acceleration coefficient,c1=c2=2 The inertia weight, w=1
FA[14]	30	100	γ=1.0,β_0=1.0,α=0.5
EDSFA	30	100	γ=1.0,β_0=1.0,α is a dynamic adjustment value,α_0=0.5

TABLE II. THE CLASSIFICATION ACCURACY OF DISPATCHING RULES*

Criterion	TP		MCT		NT	
Accuracy	*Ave (Best)%*	*SD*	*Ave (Best)%*	*SD*	*Ave (Best)%*	*SD*
EDSFA-ELM	**90.97** (92.43)	**1.08**	**97.10** **(100)**	**1.65**	**93.98** **(94.97)**	0.85
FA-ELM	88.06 (92.38)	2.15	95.26 (98.77)	2.46	93.05 (94.74)	2.03
PSO-ELM	89.32 (90.16)	1.18	87.60 (90.15)	2.58	92.26 (93.34)	1.16
GA-ELM	88.86 **(93.33)**	2.04	95.95 **(100)**	2.20	93.64 (94.02)	**0.52**
All attributes-ELM	68.06 (69.98)	1.84	72.95 (75.08)	1.89	70.16 (72.04)	1.74

*Note: The number in bold which is on behalf of the optimal value.

V. CONCLUSION

In order to dynamically select the dispatching rules according to the current status of job shop production system for data-driven adaptive scheduling, the first problem to be solved is how to effectively describe the status of job shop production system. The work presents a scheduling feature selection approach based on the improve FA. To determine the optimal dispatching rule based on the current values of scheduling features, the ELM-based classifier is applied in this work. Experiments show that the optimal dispatching rule decision accuracy of the ELM-based classifier based on scheduling feature selection is much better than that based on complete production system attributes, besides the proposed algorithm EDSFA-ELM can obtain better classification performance than other methods on scheduling sample set. With the optimal scheduling features selected by the EDSFA, the status of the production system can be properly described, which lays a solid foundation for the dynamic job shop scheduling based on production data. In future work, we will continue to study the problem of data-driven job shop scheduling with machine learning methods, and further investigation to optimize the dispatching rule classifier is also required.

ACKNOWLEDGMENT

The authors would like to thank the National Natural Science Foundation of China for supporting this project.

REFERENCES

[1] Johnson S M. “Optimal two - and three - stage production schedules with setup times included”. *Naval research logistics quarterly*, vol. 1, no. 1, pp. 61-68, 1954.

[2] Holthaus O, Rajendran C. “Efficient dispatching rules for scheduling in a job shop”. *International Journal of Production Economics*, vol. 48, no. 1, pp. 87-105. 1997.

[3] WANG J, CHEN Y. “Data-driven Job Shop production scheduling knowledge mining and optimization”. *Computer Engineering and Applications*, no. 1, pp. 42, 2018.

[4] Frazzon E M, Kück M, Freitag M. "Data-driven production control for complex and dynamic manufacturing systems". *CIRP Annals*, 2018.

[5] Ji W, Wang L. "Big data analytics based fault prediction for shop floor scheduling". *Journal of Manufacturing Systems*, vol. 43, pp. 187-194, 2017.

[6] Qiao F, Ma Y, Gu X. "Attribute selection algorithm of data-based scheduling strategy for semiconductor manufacturing". *Automation Science and Engineering (CASE)*, 2013 IEEE International Conference on, IEEE, 2013: 410-415.

[7] Xu Z, Jin R, Ye J, et al. "Non-monotonic feature selection". *Proceedings of the 26th Annual International Conference on Machine Learning*, ACM, 2009: 1145-1152.

[8] Xue B, Zhang M, Browne W N, et al. "A survey on evolutionary computation approaches to feature selection". *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 4, pp. 606-626, 2016.

[9] Tabakhi S, Moradi P. "Relevance‑redundancy feature selection based on ant colony optimization". *Pattern recognition*, vol. 48, no. 9, pp. 2798-2811, 2015.

[10] Xue Y, Jiang J, Zhao B, et al. "A self-adaptive artificial bee colony algorithm based on global best for global optimization". *Soft Computing*, 2017: 1-18.

[11] Diao R, Shen Q. "Nature inspired feature selection meta-heuristics". *Artificial Intelligence Review*, vol. 44, no. 3, pp. 311-340, 2015.

[12] Tran B, Xue B, Zhang M, et al. "Investigation on particle swarm optimization for feature selection on high-dimensional data: Local search and selection bias". *Connection Science*, vol.28, no. 3, pp. 270-294, 2016.

[13] Yang X S. "Nature-inspired meta-heuristic algorithms". *Luniver press*, 2010.

[14] Zhang J, Gao B, Chai H, et al. "Identification of DNA-binding proteins using multi-features fusion and binary firefly optimization algorithm". *BMC bioinformatics*, vol. 17, no. 1, pp. 323, 2016.

[15] Huang G B, Zhu Q Y, Siew C K. "Extreme learning machine: theory and applications". *Neurocomputing*, vol. 70, no.1-3, pp. 489-501, 2006.

[16] An L, Bhanu B. "Image super-resolution by extreme learning machine". *Image processing (ICIP)*, 2012 19th IEEE International Conference on. IEEE, 2012: 2209-2212.

[17] Xin J, Wang Z, Qu L, et al. "Elastic extreme learning machine for big data classification". *Neurocomputing*, vol. 149, pp. 464-471, 2015

[18] Weng Q, Mao Z, Lin J, et al. "Land-use scene classification based on a CNN using a constrained extreme learning machine". *International Journal of Remote Sensing*, 2018: 1-19.

[19] Shiue Y R. "Data-mining-based dynamic dispatching rule selection mechanism for shop floor control systems using a support vector machine approach". *International Journal of Production Research*, vol. 47, no. 13, pp. 3669-3690, 2009.