Milling Force Prediction of Titanium Alloy Based on Support Vector Machine and Ant Colony Optimization

Xiaoxia Zhang, Ziqiao Yu, Yinyin Hu, and Jiao Yang

***Abstract*—Milling force prediction of titanium alloy plays an important role in titanium alloy milling process. In the article, the milling process of titanium alloy materials is analyzed, and this material quality is affected directly by the milling force. Support vector machine (SVM) has shown prominent performance for many real-world problems. However, SVM computational complexity increases, as the number of samples increases. How to effectively apply it to massive datasets is still a serious challenge. Although there are some literatures on parameter optimization techniques of SVM regression, the performance of this model still needs to be further studied and improved. We present a new milling force predicting model, namely ACO&SVM, which integrate SVM regression with ACO algorithm to enhance the prediction accuracy of milling force for mill process of titanium alloy. The primary innovation feature of hybrid model is to apply the ACO to automatically determine parameters of SVM regression model. In addition, as being applied to discrete optimization, to solve continuous optimization problems, ACO algorithm needs to transform continuous variables into discrete ones by discretizing of continuous variables. The results have shown that proposed ACO&SVM model yields better prediction accuracy, and thus it can be widely applied to the fields of material processing optimization.**

***Index Terms*—Ant colony optimization, BP neural network, Support vector machine, Titanium alloy.**

I. Introduction

In Titanium industries, optimizing process parameters is one of the most important production processes. The modern titanium enterprises begin to pay more attention to the innovation of theory and technology methods that reinforce to increase the titanium alloy factory efficiency and fully utilize their existing resources to ensure their strong position in the international competition. In this paper, we concentrate our attention on titanium alloy milling process. In the titanium alloy milling process, a typical problem is how to optimize process parameters, which plays a significance role in the titanium alloy production.

Titanium alloys have many advantages, such as very high strength features, elevated temperature resistance characteristics, and excellent abrasion resistance. They are used in chemical industry, medical treatment, aerospace, automobile and other fields [1]. Because, the titanium alloy material has the characteristics of low elastic modulus, low wear-resistant cutting tool, and high temperature in the cutting process, it is difficult to be machined. So the research of titanium alloy machining, especially milling performance and optimizing process parameters, has a practical significance of improving the processing efficiency, cutting production costs and promoting the application of the materials.

At present, the optimization of milling process parameters in titanium alloy is a significance research topic, which has been widely studied in recent times. Despite the application of parameters optimization has made great progress in titanium industries, researchers are still seeking to improve the performance and accuracy of models. André et al. [2] focused their research on taking cutting force and surface roughness and processing benefit as optimization objective, and they adopted the genetic algorithm (GA) to search milling parameters for optimization. Özel et al. [3] constructed a multi-dimensional model by the finite element approach. This model could be used to explore the influence of the different technological parameters on the surface roughness. Tansel [4] discussed the effects of feed rate, milling speed and depth on surface roughness of titanium alloy by combining genetic algorithm (GA) with neural network. Liu et al. [5] adopted an optimization model, which combined the kriging interpolation with genetic algorithm for the titanium alloys turning, and the results indicate that the optimum parameter levels for different variables have been suggested. Wu et al. [6] designed a cutting force prediction model in milling process by increasing chip thickness. Experiment results showed that simulated results are consistent with measured results. Elmagrab et al. [7] explored confirmatory the influence of milling parameters on milling surface characteristics of titanium alloy materials. The results showed that feed rates had a greater effect on surface roughness. Although there are some literatures on optimizing process parameters in titanium industries, the optimization of process parameters remains to

Manuscript received August 25, 2020; revised February 18, 2021. This work was supported by Project of Liaoning Xincheng Co., Ltd (Grant No. L20170989).

X. X. Zhang is a Professor of School of Computer Science and Software Engineering, University of Science and Technology LiaoNing, Anshan,11041,China (corresponding author, phone:86-0412-5929812; e_mail: aszhangxx@163.com).

Z. Q. Yu is a Master Student of School of Computer Science and Software Engineering, University of Science and Technology LiaoNing, Anshan, 114051, China (e_mail: 1152899934@qq.com).

Y. Y. Hu is a Master Student of School of Computer Science and Software Engineering, University of Science and Technology LiaoNing, Anshan, 114051, China (e_mail: 2799386740@qq.com).

J. Yang is a Master Student of School of Computer Science and Software Engineering, University of Science and Technology LiaoNing, Anshan, 114051, China (e_mail: 2036588541@qq.com).

be further studied to increase the titanium alloy production efficiency.

Support vector machine (SVM) model was originally employed to solve pattern recognition and classification problems, and it is a widely used and effective machine learning algorithm. The SVM models have been successfully used to practical problems. With the discovery of a *e*-insensitive function, the SVM model is conveniently expanded to help settle estimation problems of nonlinear regression, namely SVM regression. The SVM regression has been developed to solve many practical forecasting problems [8], such as engineering and software field forecasting [9], public traffic forecasting [10-11], electric load forecasting [12-13], adhesion strength of coating performance [14], and so on. However, it has also been observed that as the number of samples increases, SVM regression computational complexity increases. How to effectively apply it to massive datasets is still a serious challenge. Moreover, the practical results show that prediction accuracy is poor due to the lack of knowledge of parameter choice in a SVM regression model. Nevertheless, the structured methods in selecting several parameters for a SVM regression model may not perform well. Although there are some literatures on parameter optimization techniques of SVM regression model, the accuracy of this model still needs to be further studied and improved. In our study, we proposed a new milling force prediction model. The main features of this paper are different from the above papers in the following points: First, the important idea of our paper is to integrate the construction solution characteristics of the ACO with the SVM regression, which leads to a novel model that we name ACO&SVM. Second, within the ACO&SVM model, we apply the ACO to automatically determine three parameters of SVM regression model. Moreover, as being applied to discrete optimization, to solve continuous optimization problems, ACO algorithm needs to transform continuous variables into discrete ones by discretizing of continuous variables. Finally, the results of experiment have proved that ACO&SVM model yields better prediction in the terms of the accuracy.

The paper is arranged as follows. Ant colony optimization is presented in Section 2. Support vector machine regression are introduced in Section 3. The proposed ACO&SVM forecasting is presented in Section 4. Section5 describes experimental results that reveal the forecast performance. Section 6 presents the conclusions.

II. Ant colony optimization

Ant colony optimization (ACO) is an evolution method based on population evolution to get the best solution. The main principle of the ACO is that artificial ants imitate ant foraging behavior as they search for food position and seek the shortest path from the nests to food locations. The ACO method [15] was first proposed in 1997, and it has been successfully adopted to solve traveling salesman problems [16]. Many researchers have tried to put forward new strategies to promote performance of basic ACO, which has been successfully applied to different practical problems [17-18]. Each metaheuristc has its own advantages and weakness. Therefore, many researchers are trying to improve the search performance to achieve the efficient search.

There are various versions of basic ACO algorithm. Applications of this algorithm have successfully covered many practical optimization fields. ACO algorithms mainly consist of two basic steps: constructing solutions and updating pheromone trail. Ant k travels from node i to next adjacent node j by adopting the formula (1) as follows:

$$j=\begin{cases}\underset{u\notin M_k}{\operatorname{argmax}}\{t_{iu}\cdot[h_{iu}]^b\}, & \text{if } q\le q_0\\ \Gamma, & \text{otherwise}\end{cases} \tag{1}$$

in which t_{iu} represents the trail strength of edge (i,u), and β denotes a parameter. q denotes a variable, $q\in[0,1]$, and q_0 represents a parameter value. The visibility h_{iu} denotes the inverse of edge length between nodes (i, u). M_k represents the set of cities chosen by ant k, and the cities in the set will not be allowed to select again. If the condition is met $q\le q_0$, the ant moves the best route according to formula (1). If $q>q_0$, an edge can be selected according to random variable Γ. The value of variable Γ depends on the probabilistic distribution as formula (2):

$$P_{ij}^k=\begin{cases}\dfrac{\tau_{ij}\cdot[\eta_{ij}]^\beta}{\sum_{u\notin M_k}\tau_{iu}\cdot[\eta_{iu}]^\beta}, & j\notin M_k\\ 0, & \text{otherwise}\end{cases} \tag{2}$$

where p_{ij}^k denotes a probability and ant k uses this to determine to move routes.

There are two types of the pheromone trail updating, including local and global updating. Local updating is implemented in constructing solutions, and global updating will be performed after constructing solution phase. Through the analysis of the ACO algorithm, it mainly depends on the positive feedback theory and random selection probability strategy. However, the positive feedback mechanism can make the pheromone intensity of the links. This way of pheromone updating enables solution search process prone to premature convergence. To overcome the weakness of the ACO algorithm, we adopt a deterministic selection and random selection probability in the search process by adjusting the parameters to change deterministic selection and random selection chances.

III. Support vector machine regression

Support vector machine (SVM) model is a very efficient supervised learning technique proposed by Vapnic et al. [19], and it has already been developed in many industries. The major application fields of SVM model should be parted into two categories: SVM classification and SVM regression. Support vector machines have first been adopted for classification purposes, but the theory was expanded easily to regression prediction. In this article, we adopted the SVM regression model to predict the milling force.

The SVM regression is an approximation method based on machine learning theory [20]. There is a linear function, which can describe nonlinear relations between input samples

and output samples. Let T be a training set, i.e., $T=\{(x_1,y_1), (x_2,y_2), \ldots, (x_n,y_n)\}$, in which x_i denotes an input variable, y_i denotes actual variable value, and n represents the sample size in the training set. Fig. 1 depicts the linear regression performed graphically by support vector machine. With the help of using a nonlinear function $f(x)$, the input variable x_i is first converted into a feature space with high-dimension, and then a linear function can be generally defined as formula (3).

$$f(x) = w^T f(x) + b \tag{3}$$

In the formula, $f(x)$ is a forecast value, $f(x)$ denotes the nonlinear function, and w, b denote coefficients. By adopting a penalty function in order to evaluate w and b values, the SVM regression minimizes the errors in total as follows:

$$\text{Minimize } \frac{1}{2}\|w\|^2 + C\cdot\frac{1}{n}\sum_{i=1}^{n}|y_i - f(x)|_e \tag{4}$$

In the SVM, our objective is to search for $f(x)$ function with at most ε deviation between y_i and $f(x)$ for all the training data. That is to say, it is concerned that the error is less than ε, but doesn't allow any deviation greater than this. In a most common SVM, which is called e–SVM regression (Fig. 1), the model adopts a loss function to evaluate quality of forecast model as follows:

$$|y_i - f(x)|_e = \begin{cases} 0 & if \quad |y_i - f(x)|_e \le e \\ |y_i - f(x)| - e & Otherwise \end{cases} \tag{5}$$

in which y_i refers to the target data. Eq. (5) is adopted to search for an optimal hyperplane on the feature space with high-dimension separating the data set to two subsets. The key point of SVM regression is to seek the optimal hyperplane to minimize the training error. Two variables x_i and x_i^* indicate the distance between the actual values and the relevant boundary e values, and they are applied to settle infeasible constraints problem. Training errors above ε are defined as x_i^*, while training errors below $-\varepsilon$ are defined as x_i (Fig. 1). Only the points inside the threshold region will contribute to the prediction accuracy, whereas the points outside the threshold region contribute to the training error. The training vectors with non-zero multipliers should be named as support vectors.

By replacing Eq. (4) with Eq. (5), we can obtain a function, which is suitable for training set with a prediction error deviation less than e. The SVM regression mode is determined from form the estimation of $\boldsymbol{w}$ and b as follows:

$$\text{Minimize } \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(x_i^* + x_i)$$
$$s.t.\begin{cases} f(x_i) - y_i \le e + x_i \\ y_i - f(x_i) \le e + x_i \\ x_i \ge 0, x_i^* \ge 0, i = 1,2,\ldots,n \end{cases} \tag{6}$$

Where w^2 denotes a regularization rule term, which decides differences between the computational complexity and estimation accuracy of the SVM. A parameter C is regarded to control the amount of allowable deviation greater than parameter e is tolerated. e denotes the insensitive loss function, which associates with the approximation precision of training set, and the value of parameter e identifies the set of the support vectors. The dual of this problem may be solved by adopting convex programming techniques.

For formula (6) denotes the constrained optimization problem, in most cases, this problem should be solved more easily by converting it into its dual formulation adopting Lagrange multiplies method. The key principle is to build a Lagrange function based on the objective function and its constraints by using a dual variable set. By using Lagrange equation, thus, the dual problem can be obtained as formula (7),

$$\max(a_i, a_i^x) = \sum_{i=1}^{n} y_i(a_i - a_i^*) - e\sum_{i=1}^{n}(a_i + a_i^*) - \frac{1}{2}\sum_{i,j=1}^{n}(a_i - a_i^*)(a_j - a_j^*)k(x_i, x_j)$$
$$s.t.\begin{cases} \sum_{i=1}^{n}(a_i^* - a_i) = 0 \\ 0 \le a_i, a_i^* \le C \end{cases} \tag{7}$$

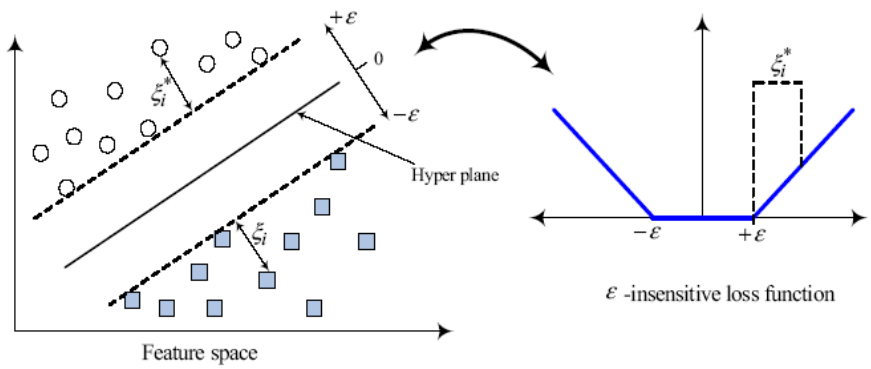

Fig. 1. Transformation process for a SVM regression model.

where a_i, a_i^* refer to Lagrange multipliers. The multipliers should meet the equality condition $\alpha_i\alpha_i^*=0$. After obtaining α_i and α_i^*, quadratic optimal problems are settled. The vector w can be obtained in formula (3) as follows,

$$w=\sum_{i=1}^{n}(a_i-a_i^*)x_i \tag{8}$$

The Lagrange multipliers are gained by settling a quadratic program. After transforming the above formula (3) into its dual problem as well as settling this dual problem, formula (3) turns into the explicit expression. Finally, for an input vector x, the output function of SVM regression mode is defined as formula (9),

$$f(x)=\sum_{i=1}^{n}(a_i-a_i^*)k(x_i,x)+b,\qquad x_i\in R^n, b\in R \tag{9}$$

In the formula, $k(x_i, x)$ is a kernel function, and this function can convert the input data into multidimensional feature space, and its value is defined as the scalar product of vectors x_i and x. The introduction of kernel function can enables SVM regression to easily find the solution for the non-linear regression problems. The SVM regression provides several different kernel functions, i.e., radial basis, polynomial, lineal and sigmoid functions. In comparison with some other kernel functions, Gaussian radial basis function (RBF) has been shown to obtain better performance results than the polynomial function. Because of fewer parameters to be set, the Gaussian radial basis function is not only handier to execute, but also it performs well nonlinearly converting between the input and high-dimension spaces, and thus, it is appropriate to solve nonlinear problems. Because of its flexibility in treating more complex parameters and its

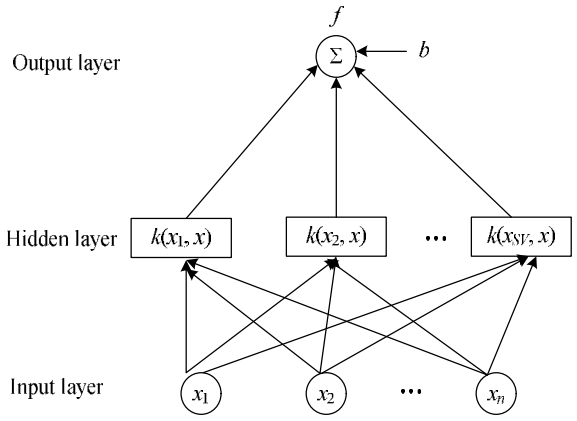

Fig. 2. SVM regression prediction based on phase space reconstruction.

computational reliability, efficiency, in this study, the data set presents obvious nonlinearity, so we choose the RBF kernel function,

$$k(x_i,x)=\exp\left(-\frac{1}{2d^2}\|x-x_i\|^2\right), d>0 \tag{10}$$

In formula (10), d denotes Gaussian RBF width, and variables x_i, x are input vectors of the training data and test data respectively. In summary, the SVM regression is similar in form to a neural network. To approximate the given observations in a multiple dimensional space, the output represents the linear combination function of the intermediate nodes in another feature space, and each hidden node should correspond to one support vector. As shown in Fig. 2, the entire SVM regression prediction structure includes input vector.

IV. Hybrid Model for Parameter Optimization of SVM

The accuracy of SVM model relies mainly on selection of the parameters, *C, d*, and *e*. These parameters must be chosen accurately, since they directly affect the feature space and algorithm complexity. However, there are few studies on structural methods for effective parameter selection. This study presents a new ACO&SVM model for optimizing parameters of SVM regression. In this ACO&SVM model, parameter values can be automatically adjusted by implementing the ACO algorithm. The ACO algorithm is well suited for seeking the global optimum in an intricate multidimensional search space. To get a better performance of ACO&SVM model, several parameters should be tuned. In the ACO&SVM model, we apply the ACO to determine three parameters of SVM regression model. After getting optimal parameters, the SVM model is built to implement the prediction task. The overall framework design of the ACO&SVM process is described in Fig. 3.

For the main idea of implementing the high prediction accuracy with the least test errors, in this paper, we have utilized the advantage of the ACO to seek the best parameter values for *C, d* and *e* in SVM model. Three parameters for SVM model may influence the prediction accuracy. Nevertheless, the structural techniques for determining efficient choice of parameters are very limited in literature. Three parameters, *C, d* and *e*, are continuous problems. The ACO algorithm was originally designed for discrete combinatorial optimization problem, and it should require some specified transforming strategy in continuous problems. In order to optimize these three parameters by ACO algorithm, *C, d* and *e* must be discretized. In this work, to transform continuous parameters into discrete ones, we first mesh these parameters into *N* grids, and then compute the grid distance of each parameter. Each grid node represents a combination of three parameters, and it is equivalent to a city point of TSP. In this research, the ACO algorithm is adjusted to choose three parameters for SVM model in discrete problems. A grid node *X* here is defined as $X = \{C, d, e\}$. Therefore, each grid node is defined to represent a combination of three parameters, *C, d* and *e*. The more grid nodes mean more ants participate in computation, which leads to computational complexity. But if the number of grids is quite larger, the convergent speed of algorithms is relatively slower. As above introduced in section 2, in ACO algorithm, artificial ants may construct solutions by considering pheromone trails. The general framework of the proposed ACO&SVM consists of state transition strategy, state updating rules and evaluating function.

A. State transition strategy

State transition strategy can make it possible for ants to explore grid nodes with three parameters combination by

according to the pheromone trails. In an ACO algorithm, each ant is responsible for constructing a subset of solutions by adopting a probabilistic rule to travel from grid node *i* to adjacent grid node *j* in its neighborhood list. The transition strategy is defined as follows:

$$P_{ij} = \frac{\tau_j \cdot (\eta_j)^{\beta}}{\sum_{j \in \Omega_i} \tau_j \cdot (\eta_j)^{\beta}} \tag{11}$$

In the above formula, t_j denotes the pheromone value of grid node *j*, W_i refers to the neighborhood list of grid node *i*, i.e., the set of possible nodes ants can select on grid node *i*. h_j stands for the inverse proportion to the forecasting error at the node *j*. *b* denotes a parameter determining the relative weight of pheromone trail level.

B. The updating rule

When the ants are foraging for food, they deposit a pheromone trail on the grid nodes they have already moved. The grid point with higher pheromone trail has higher selection probability, and the pheromone trail strengthens relating to the grid nodes, which is more attractive to the next generation of ants. This updating mechanism of pheromone based on biological nature is easy to stagnate. The pheromone updating rules mainly include local updating and global updating, which are used to solve the trouble problems. The updating strategy is applicable to the subset of parameters that generate the minimum prediction errors in this iteration. According to this updating strategy, the pheromone quantity of the better parameters subset should increase. Therefore, the ant that can find the optimal solution can place pheromone trail on its selected parameter nodes. This selection, together with using state transition strategy, aims to make a clearer search direction, so that ants can search in neighborhoods of the optimal solution produced by the current generation of the algorithm. After ants select grid nodes and move to the grid nodes, local pheromone updating is executed according to formula (12). The global updating can not be executed until all ants have finished their search processes. The global updating rule is increased by using the updating rule according to formula (13),

$$\tau_j = (1-r) \cdot \tau_j + r \cdot \tau_0 \tag{12}$$

$$\tau_j = (1-x) \cdot \tau_j + x \cdot \Delta\tau_j \tag{13}$$

In formula (12), r is the local evaporation coefficient, $r \in (0,1)$, and t_0 denotes the initial pheromone. In formula (13), x denotes a parameter of global trail decay degree, $x \in (0,1)$. $\Delta\tau_j$ is adopted to enhance the pheromone trail on the route of the current solution. $\Delta\tau_j = 1/L$ where L is a fitness function value in formula (14). The updating strategy helps to provide greater pheromone trail of the solutions that cause fewer test errors. That is, these parameter combinations with fewer test errors have more possibility to be traveled by the other ants in the future.

C. Evaluating Function for parameter optimization

As we all know, the prime objective of seeking the optimal combination of three parameters (C, d , e) is to reach the better prediction accuracy according to the evaluation criteria. Thus, the core idea of the ACO is to choose appropriate combination of parameters (C, d , e) to optimize the evaluation criteria as the fitness function. Here, in the optimization of model parameters, the mean absolute percent error (*MAPE*) can be defined as a fitness function to evaluate the predictive accuracy. The mean error is more suitable because it is independent from the data range. The smaller the *MAPE* value is, the more excellent the prediction result is to the measured values. The forecasting error can be defined according to formula (14),

$$MAPE = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{y_i - f_i}{y_i} \right| \times 100\% \tag{14}$$

in which n represents the number of testing dataset. y_i and f_i denote the actual and forecasting values, respectively.

D. ACO&SVM model

This paper designs a novel ACO&SVM model, which hybridizes SVM regression with ACO algorithm to enhance the prediction accuracy. In this ACO&SVM, for the purpose of parameter determination of the SVM is to minimize the forecast deviation, first, the range of the parameters should be determined. In the proposed SVM model, cutting parameters were used as inputs and milling force as the output. Three parameters of SVM model are automatically changed by executing ACO algorithm. Next, the obtained best parameters are used to build prediction model, and this model performs the forecasting task after the training phase. Fig. 3 gives the framework of ACO&SVM model and also shows the process of parameter choice of SVM model according to the ACO. In the article, we have developed a SVM regression model which mainly includes three stages. At first stage, we apply the ACO algorithm to search three optimal parameters for SVM model simultaneously to reach a better generalization performance. In the second stage, these optimal parameters can be applied to build a prediction model. Finally, the prediction model is used to forecast milling forces. The detailed process of the ACO&SVM model is illustrated as follows:

Step 1: Data preparation. The samples are parted into training data set and test data set. Initialize the number of ants num_{ant}, the maximum number of iterations CN_{max} and terminal condition, i.e., predefined precision J. Initialize the trails t and three parameters: C, r, d, and the non-sensitivity coefficient e.

Step 2: A grid node X here is defined as $X = \{C, d, e\}$. After that, compute the grid distance of each parameter, and divide the parameters into N grids. According to the actual problem, specify the boundary of m parameters $x_j \in [x_j^{low}, x_j^{upper}]$, j=1, 2, …, m, in which m denotes the number of parameters for optimizing, m=3, i.e., three parameters C, d, e, respectively. x_j^{low} and x_j^{upper} denote the upper and lower bound for parameters. h_j denotes the grid interval of each parameter, which is computed according to formula

(15). In the same grid interval h_j, mesh parameters into N small grids. Initialize pheromone trail on the grid points, and place randomly ants on the grids as the starting nodes of the ants.

$$h_j = (x_j^{low} - x_j^{upper}) / N \quad (15)$$

Step 3: Each ant travels from grid node i to adjacent grid node j in its neighborhood list W_i according to the probabilistic distribution as formula (11). W_i is the set of possible nodes which ants can select on node i. The node j denotes a combination of three parameters, C, d and e. Record three parameters C, d and e values and calculate the forecasting error in the meantime.

Step 4: Evaluate forecasting errors. Input three parameter values, C, d and e , into the SVM model to train data and build predictive model. After training SVM model using C, d and e parameters, the algorithm estimates parameter combination nodes by computing fitness function. In this article, the mean absolute percent error (*MAPE*) can be adopted as a fitness function, which is formulated as formula (14).

Step 5: Perform local pheromone updating. After the ants select grid nodes and move to the grid nodes, the pheromone on the grid nodes is updated immediately according to formula (12).

Step 6: Repeat Steps 3-5 until all ants finish their choice of nodes, that is, the ants reach the predetermined number of movements.

Step 7: Record the grid node with the best mean squared error and update global pheromone trail according to formula (13).

Step 8: Check whether the condition $iteration<CN_{max}$ is met, if $iteration<CN_{max}$, then go to step 3; or else, continue to the next step to diminish the grid interval.

Step 9: Seek and record the node subscript with the most pheromone trail quality, and then reduce grid interval distance of the parameter h_j.

Step 10: Repeat Steps 2-9 until grid distance h_j is smaller than given value J that denotes a termination condition.

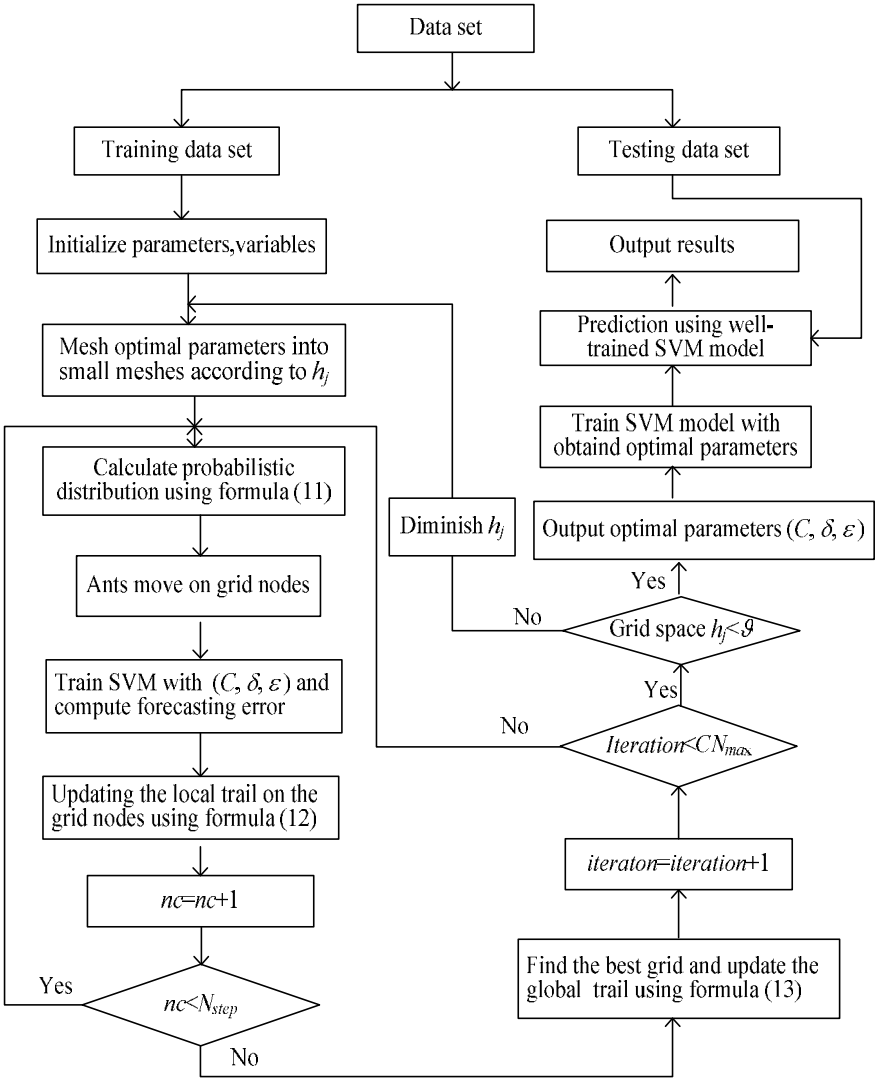

Fig. 3. The flowchart of ACO&SVM model.

Step 11: if the terminal criteria condition is met $h_j<J$, we can get the global optimum values of (C, d, ε). These optimum parameters will be brought into the training set to build a prediction model. The model should be adopted to predict milling forces in the test data, and then the algorithm goes to stop.

V. Computational results

In the section, we demonstrated the accuracy of proposed ACO&SVM model for the milling force prediction problem. The proposed method was compared with various methods. Firstly, we validate performance of SVM model with different kernel function because distinct kernel functions can affect significantly the final prediction performance of SVM. Secondly, we compared ACO&SVM with basic SVM. Thirdly, ACO&SVM and the BP Neural Networks were used for comparison. Moreover, ACO&SVM can be compared with GA&SVM. Consequently, total four prediction approaches were investigated in this milling force forecasting problem. The details of comparison results are described as follows.

A. Data set and preprocessing

In this section, our proposed ACO&SVM model has been implemented in MATLAB environment. It is very important to provide a wide range of experimental data sets in the establishment of intelligent prediction modeling for milling force. The accuracy of predictive models largely relies on the generality of the experimental data, so it is significant to collect enough data to develop the prediction model. To assess the prediction performance of our model, the experimental dataset is obtained from the published literature [1, 21-22]. According to above description, the major parameters affecting milling force are summarized as milling depth, cutting speed and feeding speed. Table I gives the description of these parameters in detail. In this experimental dataset, three parameters are adopted as input parameters and the milling force is used as the forecasting value through optimized SVM model. Table II lists experimental data of processing parameters and milling force. Fig.4 shows the actual values of milling force. There are 88 samples in the experimental data. To guarantee fair representation of data samples, these samples are randomly parted into training test sets, and they are arranged according to training set before the test set. The first 75 are adopted as training data to train models and the remaining 13 as test set to estimate the predicting efficiency of models.

TABLE I
THE PARAMETERS RANGE IN EXPERIMENTS.

No.	Factor	Symbol	Range
1	Milling depth (mm)	A	0.2–1.4
2	Feeding speed (mm/r)	f	0.1–0.9
3	Cutting speed (m/min)	v	50–120
4	Milling force (N)	F	0–1000

mm = millimeter, m/min=meter per minute, N = Newton.

TABLE II
Experimental data of processing parameters and milling force.

No	A	f	v	F	No	A	f	v	F	No	A	f	v	F	No	A	f	v	F
1	0.6	0.05	65	86	23	1	0.06	60	300	45	0.6	0.02	120	76	67	0.6	0.2	80	302
2	1.4	0.06	80	390	24	1	0.14	80	465	46	0.9	0.4	90	935	68	0.6	0.1	90	210
3	0.5	0.3	50	340	25	0.6	0.4	50	649	47	0.5	0.6	50	554	69	0.5	0.6	70	582
4	0.3	0.8	100	536	26	0.3	0.8	90	539	48	0.9	0.2	70	473	70	0.8	0.6	70	966
5	1	0.02	70	143	27	0.6	0.4	90	646	49	1.4	0.02	120	189	71	0.3	0.2	80	160
6	0.2	0.14	100	133	28	0.8	0.6	100	947	50	0.3	0.4	90	315	72	1	0.1	120	270
7	0.2	0.1	120	129	29	0.5	0.6	60	560	51	0.8	0.6	80	955	73	1	0.1	100	275
8	0.3	0.2	50	167	30	1.4	0.02	100	186	52	0.9	0.2	80	467	74	0.8	0.3	80	541
9	0.8	0.3	70	536	31	0.2	0.14	80	137	53	0.2	0.14	120	131	75	1.4	0.1	120	426
10	0.8	0.3	90	546	32	0.6	0.02	100	82	54	0.6	0.06	120	233	76	0.3	0.8	50	549
11	0.8	0.6	50	971	33	1	0.02	100	120	55	0.3	0.8	70	528	77	0.5	0.6	100	621
12	0.9	0.4	50	933	34	0.5	0.3	90	348	56	1.4	0.1	80	428	78	0.3	0.2	60	165
13	1	0.06	100	268	35	0.6	0.4	70	615	57	0.9	0.4	60	930	79	0.2	0.1	90	135
14	0.9	0.2	100	456	36	0.3	0.8	80	534	58	0.9	0.4	100	939	80	0.6	0.2	50	270
15	0.3	0.8	70	528	37	0.3	0.2	70	163	59	1.4	0.1	100	434	81	0.6	0.2	70	284
16	0.8	0.3	50	530	38	1	0.02	80	139	60	0.9	0.2	90	459	82	0.5	0.3	60	348
17	1	0.14	120	479	39	0.5	0.3	80	354	61	0.6	0.06	100	224	83	0.9	0.2	80	467
18	0.6	0.05	80	80	40	0.5	0.6	90	618	62	0.3	0.8	60	540	84	0.9	0.2	50	476
19	0.3	0.2	90	156	41	0.5	0.6	80	614	63	1	0.06	120	260	85	0.5	0.3	70	350
20	1.4	0.06	100	405	42	0.3	0.4	50	320	64	0.6	0.4	80	620	86	0.6	0.4	100	648
21	0.6	0.2	90	318	43	0.3	0.4	70	306	65	1.4	0.1	60	415	87	0.3	0.4	80	310
22	0.6	0.05	51	90	44	0.2	0.1	100	130	66	1	0.06	80	278	88	0.3	0.4	60	313

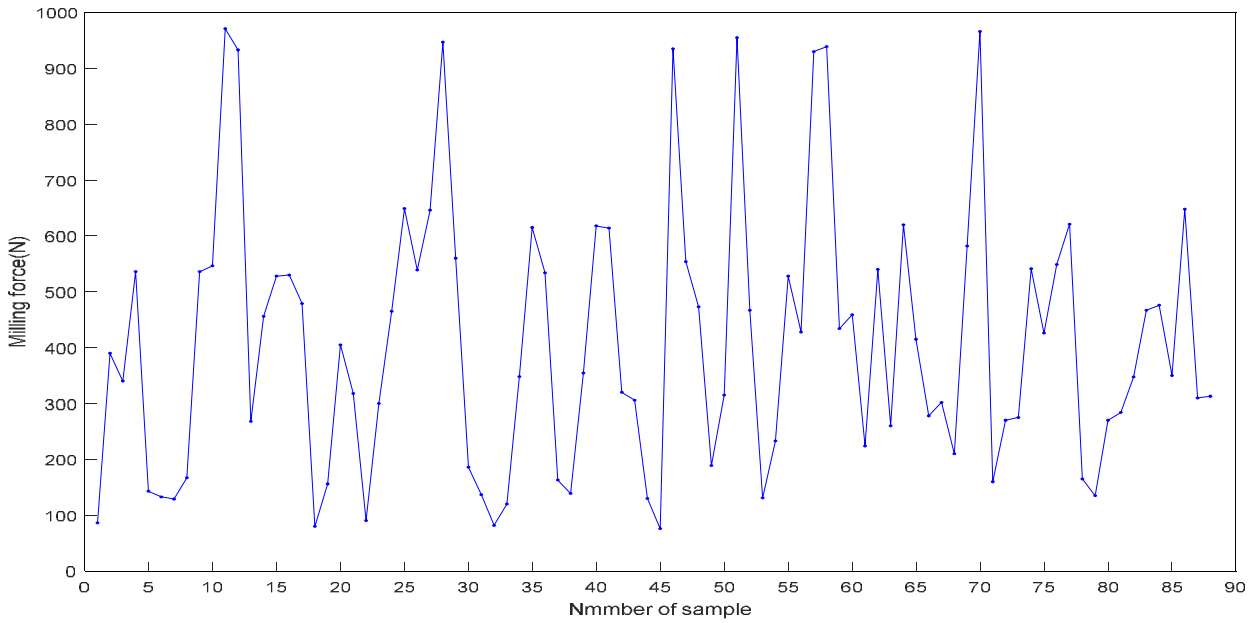

Fig. 4. The actual values of milling force.

B. *Parameter setting*

Because different kernels directly affect the predict performance of SVM model significantly, we validate performance of this model with four different Kernel functions. To clearly verify the accuracy of the SVM model, four statistical parameters were used as evaluation criteria. These parameters consist of normalized root mean squared error (*NRMSE*), mean absolute percentage error *(MAPE*), relative error (*Error*), and coefficient of determination (R^2), as defined in Table III. *NRMSE*, *MAPE* and R^2 are adopted to estimate overall performance. The values of *NRMSE* and *MAPE* are smaller, and the predicted values are the closer to the actual values. The larger the parameter R^2, the more stable the model is, that is, R^2=1 indicates a perfect model, while R^2=0 means an inaccurate mode. Moreover, *Error* represents percentage deviation of the predicted value and actual milling forces.

The performance of the SVM model with different kernel functions is presented in Table IV. From this table, the performance of the SVM model with RBF kernel function enhances the accuracy as listed by the lowed *NRMSE a*nd *MAPE*. As each kernel function needs to optimize different parameters separately, a grid searching method was adopted to seek the best kernel function parameter *d*, penalty parameter *C* and the non-sensitivity coefficient *e* for the SVM. When training errors get smaller, modify three parameters, *C*, *d* , and ε to compute the validation error. After that, three parameters can be determined as the most suitable parameters as shown in table V. From table IV, a comparison of different kernel functions shows that the SVM model with RBF kernel has the smaller *NRMSE* and *MAPE*, and it has the smaller *NRMSE* and *MAPE*, and it has the highest coefficient of determination in both training set and testing set. Therefore, the SVM model with RBF kernel can provide more accurate

TABLE IV
PERFORMANCE OF SVM MODEL WITH DIFFERENT KERNEL FUNCTION.

Data set	Statistical parameters	Kernel function			
		Linear	Polynomial	Sigmoid	RBF
Training set	*MAPE* (%)	27.56	13.756	27.459	10.045
	NRMSE	0.288	0.1348	0.2932	0.0604
	R^2	0.686	0.9310	0.6737	0.9861
Testing set	*MAPE* (%)	15.65	7.0645	15.264	7.2120
	NRMSE	0.202	0.1146	0.2014	0.0758
	R^2	0.713	0.9084	0.7176	0.9600

TABLE III
ERROR MEASURES FOR ACCURACY ASSESSMENT.

Error measure	Mathematical expression
Normalized root mean squared error, *NRMSE*	$NRMSE=\sqrt{\sum_{i=1}^{n}(y_i-f_i)^2}\Big/\sum_{i}^{n}y_i^2$
Mean absolute percentage error, *MAPE*	$MAPE=\frac{1}{n}\sum_{i=1}^{n}\left\|\frac{y_i-f_i}{y_i}\right\|\times 100\%$
Relative error , *Error*	$Error=\frac{\|y_i-f_i\|}{y_i}\times 100\%$
Coefficient of determination, R^2	$R^2=1-\frac{\sum_{i=1}^{n}(y_i-f_i)^2}{\sum_{i=1}^{n}(y_i-\sum_{i=1}^{n}y_i/n)^2}$

TABLE V
PARAMETERS OF SVM MODEL WITH DIFFERENT KERNEL FUNCTION.

Kernel function	*C*	*d*	*e*
Linear kernel	4.000	$9.765e^{-4}$	0.001
Polynomial kernel	128.000	5.650	0.01
Sigmoid kernel	64.000	0.0313	0.001
Radial basis function kernel (RBF)	12.000	1.000	0.05
GA&SVM	4.000	0.500	0.01
ACO&SVM	11.314	0.7071	0.01

predictions. In ACO algorithm, we set parameter values as follows: $q_0 \in [0.60, 0.85]$, $\beta \in [3, 4, 5]$, $\rho \in [0.35, 0.60]$, $num_{ant} = 10$. $CN_{max} = 500$.

C. Forecasting results and analysis

As we have discussed, the prediction accuracy of SVM model mainly depends on the three parameters, i.e., kernel function parameter d, penalty parameter C, and the non-sensitivity coefficient e. The main purpose of optimization parameters for the SVM model is to optimize the process sufficiently, which can search for a finite subset to obtain the best parameters with minimum generalization error. To enhance the performance of SVM model in milling force forecasting, we present a hybrid ACO&SVM model, in which we adopt ACO algorithm to determine the most suitable parameters of SVM model. We can find the best C, d, and$e$ by updating the pheromone values continuously. After getting these parameters, the SVM model is built to implement the prediction task.

Fig. 5 showed evolutionary process of the optimal parameters. Training samples were feeding into this proposed model, and the ACO algorithm was applied to search for three best parameters, i.e., C, d , and e in the SVM regression model. The process searching for optimal parameters should be executed for 200 generations. In Fig. 5, the fitness (*MAPE*) value decreased as the generation number increased. In the whole searching process, the function value gradually converged until the optimal parameters with the fitness value 9.1% were obtained at generation 150. Therefore, with the ACO algorithm, the best parameters in the SVM regression were obtained to be C=11.314, d =0.707 and e =0.01. These optimal parameters were used to build SVM regression prediction model.

Table VI showed the comparison between the grid search and ACO&SVM model. In the grid search, the penalty factor C was explored from 0 to 15 at an interval of 0.5, the kernel parameter d was from 0.5 to 10 at an interval of 0.05, and the loss function e was explored from 0.001 to 0.05 at an interval of 0.005. As shown in the table VI, the ACO&SVM has a better performance than the grid search in terms of *MAPE*, *NRMSE* and R^2. Moreover, the ACO&SVM required a small amount of training time while the grid search took much time. Fig. 6 and Fig. 7 show the comparison between the predicted milling force of hybrid ACO&SVM model and the actual force of training data set and testing set separately. The predicted milling forces are in good consistent with actual milling forces on the whole. The forecasting results of models are illustrated in Table VII in which Actual value refers to gauged milling force, the SVM model to basic SVM, BP model to BPNN, GA&SVM model to combination of genetic algorithm and SVM, ACO&SVM to we proposed, and *Error* is percentage deviation of predicted value and actual milling forces. Among the 13 samples, the *Error* value of 11 samples between 3 and 6. The *NRMSE* value is found to be only 0.0237 for all the cases tested, while The *MAPE* value is 1.7175%. Therefore, we conclude that the ACO&SVM can possess strong self-learning ability and simultaneously obtains the excellent performance.

To verify the forecasting accuracy compared with other algorithms, the milling force results of ACO&SVM model basic SVM and BP model are reported in Table VII. Fig. 8

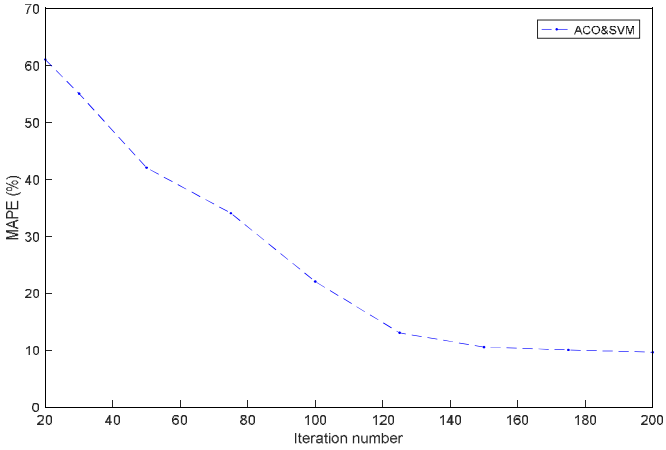

Fig. 5. The convergence performance of ACO&SVM models.

TABLE VI
COMPARISONS OF PERFORMANCE BETWEEN GRID SEARCH AND ACO&SVM MODEL.

No	Data set and parameter	Grid search			ACO&SVM		
		MAPE (%)	*NRMSE*	R^2	*MAPE* (%)	*NRMSE*	R^2
1	Training set	9.35	0.079	0.9702	3.4224	0.0257	0.9975
2	Testing set	1.85	0.0255	0.9905	1.7175	0.0237	0.9961
3	Time(s)		629.62			18.87	
3	C		10.55			11.314	
4	d		0.6830			0.7071	
5	e		0.01			0.01	

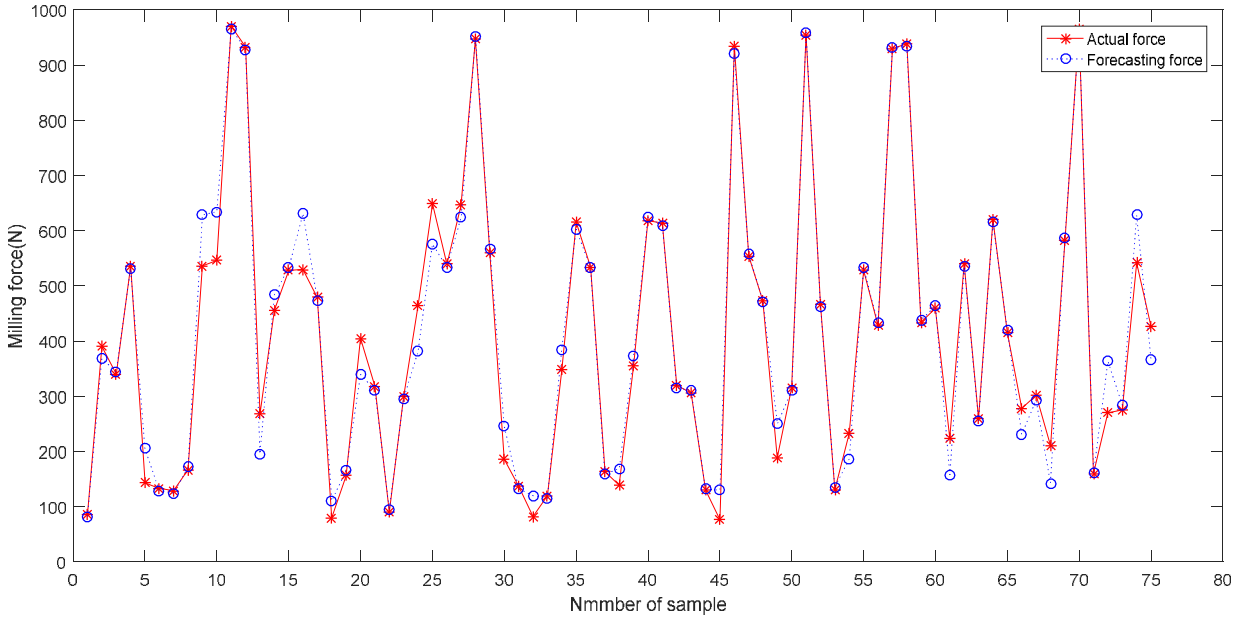

Fig. 6. The result comparison of ACO&SVM models and actual values in training set.

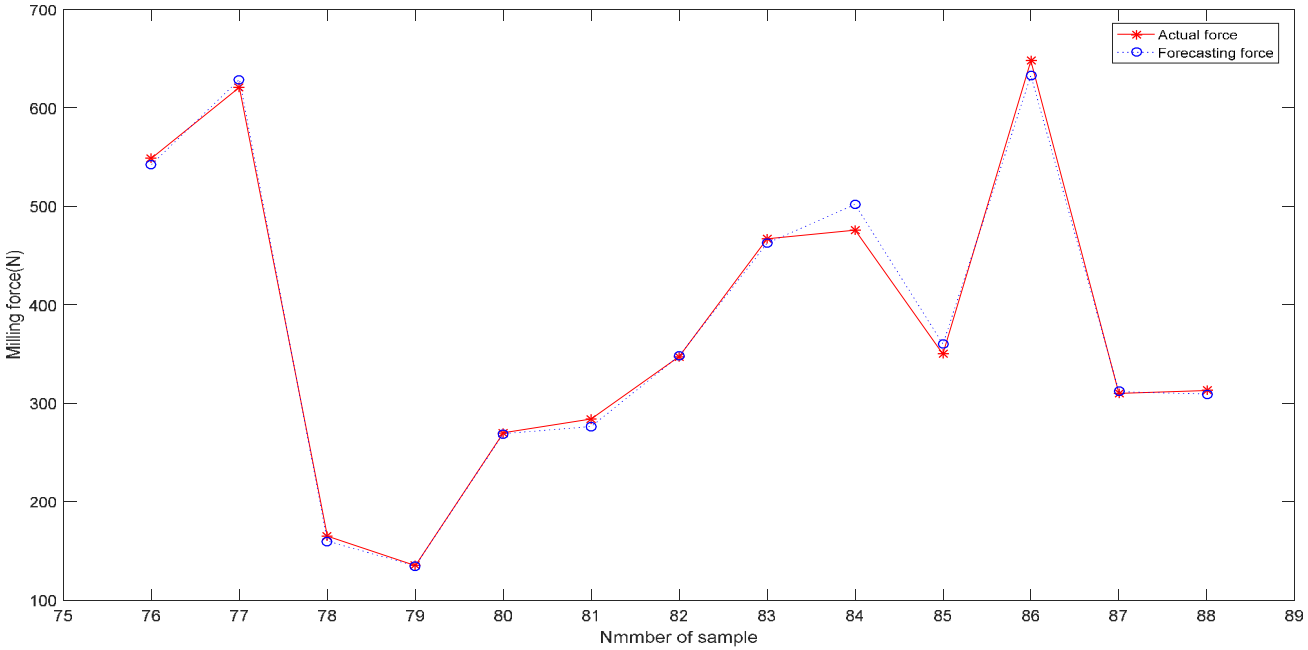

Fig. 7. The result comparison of ACO&SVM models and actual values in testing set.

gives the forecasting results of four models. For each model, the evaluation criteria *NRMSE*, *MAPE a*nd *Error* are adopted to compute the deviation between the actual and forecasting values, while coefficient of determination (R^2) is used to evaluate overall performance. Table VII and Table VIII show three types of comparisons. The first comparison is between basic SVM versus ACO&SVM model. The ACO&SVM is able to regulate automatically three parameters *C*, *d*, and *e*. But instead, basic SVM requires more manual manipulation. Compared with basic SVM, the ACO&SVM is consistent with actual values to obtain the minimum error. But, some forecasting values obtained from the basic SVM are inaccurate, which makes *Error* more than 10% for the test set. The *MAPE* value of ACO&SVM model is less than that of basic SVM, and the *NRMSE* value of the ACO&SVM is also less than basic SVM. The ACO&SVM has significant advantage over the SVM in terms of *NRMSE*, *MAPE* and *Error*. Therefore, the results have proven that the performance of ACO&SVM mode is more accurate than that of basic SVM.

The second comparison is between the ACO&SVM and the BP. The BP is a common and effective verification method because it is the traditional approach for forecasting model in practical optimization problem. The actual values and predicted values of different algorithms are listed in Table VII. From Table VII and Fig. 8, it is easy to see that the BP can obtain the relatively large forecasting errors. A few forecasting values make *Error* more than 20%. This problem

TABLE VII
COMPARISONS OF FORECASTING RESULTS AMONG BP,SVM,GA&SVM AND PROPOSED MODEL.

No	Actual value (N)	BP		SVM		GA&SVM		ACO&SVM	
		Forecasting value (N)	Error(%)	Forecasting value (N)	Error(%)	Forecasting value (N)	Error(%)	Forecasting value (N)	Error(%)
76	549	526.40	4.12	471.27	14.16	538.50	1.91	542.51	1.18
77	621	575.85	7.27	633.42	2.00	636.82	2.55	628.97	1.28
78	165	196.82	19.28	184.55	11.85	157.17	4.74	159.91	3.08
79	135	95.47	29.28	119.14	11.75	127.49	5.57	134.72	0.21
80	270	255.90	5.22	231.44	14.28	286.32	6.04	269.06	0.35
81	284	307.40	8.24	249.60	12.11	279.28	1.66	276.28	2.72
82	347.5	377.95	8.76	358.26	3.10	347.61	0.03	348.03	0.15
83	467	465.35	0.35	458.14	1.90	469.69	0.58	462.50	0.96
84	476	491.29	3.21	434.45	8.73	497.79	4.58	502.01	5.46
85	350	398.14	13.76	358.47	2.42	349.88	0.03	360.03	2.87
86	648	550.49	15.05	625.96	3.40	604.59	6.70	632.91	2.33
87	310	334.87	8.02	311.16	0.37	304.36	1.82	311.71	0.55
88	313	325.10	3.86	337.06	7.69	305.19	2.49	309.31	1.18
88	313	325.10	3.86	337.06	7.69	305.19	2.49	309.31	1.18

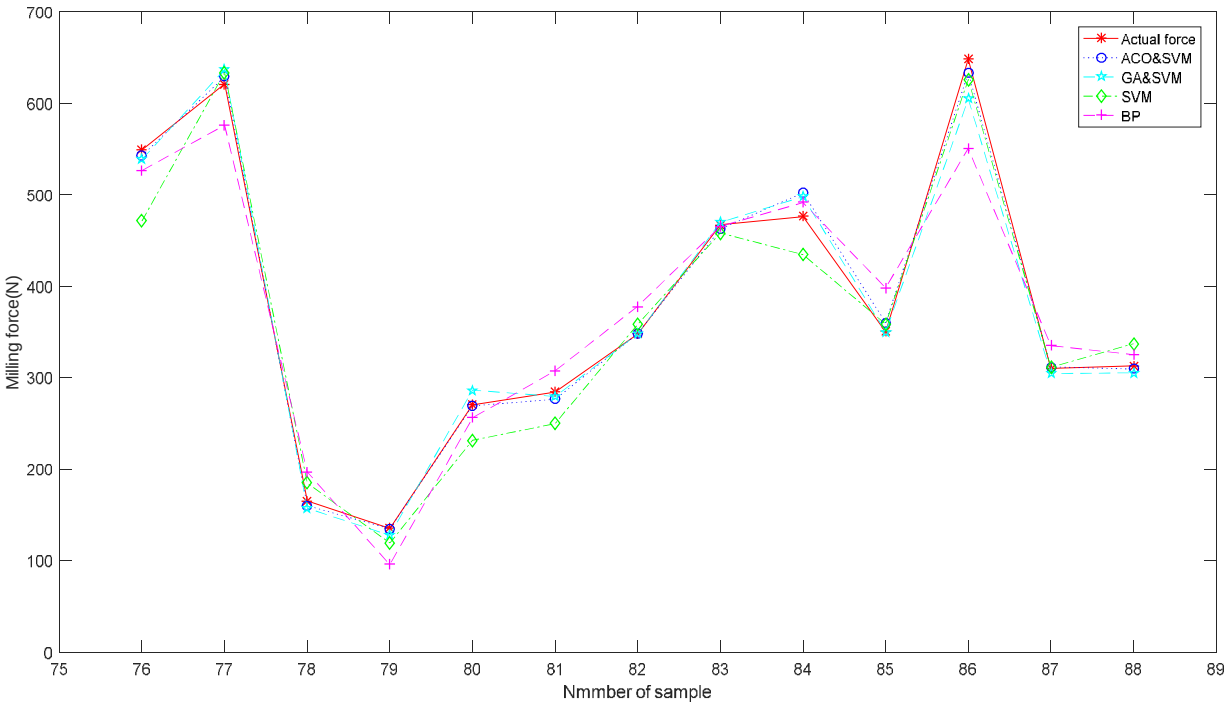

Fig. 8. The actual values of milling force and the forecasting values of four models.

may be because that BP is subjected to the disadvantages of demand for a big training data. Compared with BP, the basic SVM has a better prediction performance. Moreover, the accuracy of ACO&SVM is much higher than basic SVM. It is obvious that the ACO&SVM model performs better than BP model.

The third comparison is between the ACO&SVM and GA&SVM model. GA&SVM model applies the genetic algorithm (GA) to determine three parameters of SVM regression model [23-24]. Because the selection strategy of GA is that only the best solution can evolve to the next generation, GA may be easy to falling to local optimum. The percentage deviation from milling forces is obtained to be less than 7% for all samples tested, *NRMSE* is 0.0384, and *MAPE* is 2.9774%. And for the training set, *NRMSE* is 0.040, and *MAPE* is 4.6%. Compared with GA&SVM model, ACO&SVM model has an excellent performance. Clearly, ACO&SVM model has lower *NRMSE*, *MAPE* and *Error* compared with basic SVM, BP, and GA&SVM models. The percentage deviation is found to be less than 6% for all the instances tested, *NRMSE* is only 0.0237, and *MAPE* is 1.7175%. And for the training set, *NRMSE* is 0.257, and

TABLE VIII
SUMMARY RESULTS OF VARIOUS MODELS.

No	Models	MAPE(%)		NRMSE		R^2	
		Training set	Testing set	Training set	Testing set	Training set	Testing set
1	BP	11.6633	9.7259	0.1008	0.0947	0.9614	0.9375
2	SVM	10.0445	7.2120	0.0604	0.0758	0.9861	0.9600
3	GA&SVM	4.60	2.9774	0.04	0.0384	0.9939	0.9803
4	ACO&SVM	3.4224	1.7175	0.0257	0.0237	0.9975	0.9961

MAPE is 3.4224%. Moreover, the ACO&SVM has larger R^2 compared with basic SVM and BP model. R^2 is as high as 0.9961 for the test set, and this means a better model. From the above analysis, it has been proven that proposed ACO&SVM model has much higher predicting accuracy than basic SVM, BP and GA&SVM model in forecasting of milling force.

Aiming at the milling force prediction ACO&SVM model, we first adopted the ACO algorithm to optimize three parameters of SVM model. After then, use these optimal parameters (C, d and ε) to construct the final SVM model. Finally, the well-trained model is adopted in order to predict the test data and calculate the prediction error values. The ACO&SVM model can increase the accuracy of forecasting milling force, and it can be analyzed conveniently on the physical phenomena in the process of tool wear and processing quality.

VI. CONCLUSION

In the paper, we have proposed an ACO&SVM model to predict milling force in titanium alloy milling process. The ACO&SVM model hybridizing the solution construction characteristics of the ACO with the SVM based on supervised learning algorithm is applied to predict milling force so as to improve the efficiency of milling. The experimental results have shown that this methodology is very efficient, and thus can be used in material processing fields. This study can offer a novel method for the material processing technology optimization, and it has good research and popularization values. In this research, the main limitation is that the instances are not large enough. Therefore, we will make further efforts to increase the performance of the ACO&SVM model in relatively large instances. For our future research, other optimization algorithms and strategies still have potentiality to be integrated into a SVM forecasting model.

REFERENCES

[1] G. Q. Xiang and T. Lu, "Prediction Analysis of Titanium Alloy Milling Force Based on Support Vector Machine," *Machine Tool & Hydraulics*, vol. 44, no.3, p.142-146, 2016.

[2] F. H. L. André, N. L. Coppini, E. A. Baptista, et al. "Genetic Algorithm Applied to Investigate Cutting Process Parameters Influence on Workpiece Price Formation," *Materials and Manufacturing Processes*, vol.26, no.3,pp.550-557, 2011.

[3] T. Özel, M. Sima, A. K. Srivastava and B. Katanoglu," Investigations on the effects of multi-layered coated inserts in machining Ti–6Al–4V alloy with experiments and finite element simulations," CIRP Annals - Manufacturing Technology, vol. 59, no. 1, pp.77-82, 2010.

[4] I. N. Tansel, S. Gülmez and M. Demetgul," Taguchi Method-GONNS integration," *Expert Systems with Applications An International Journal*, vol. 38, no.5, pp.4780-4789, 2011.

[5] C. J. Liu, D. B. Tang, H. He and X.Q. Cheng," The Optimization for the Titanium Alloys Turning Based on the Kriging Interpolation and Genetic Algorithm," *Mechanical Science and Technology for Aerospace Engineering*, vol. 32, no.4, pp.469-474, 2013.

[6] B. H. Wu, X. Yan, M. Luo and G. Gao, "Cutting force prediction for circular end milling process," *Chinese Journal of Aeronautics*, vol. 26, no.4, pp.1057-1063, 2013 .

[7] N. Elmagrabi, C. H. Che Hassan and A. G. Jaharah, "High speed milling of Ti-6Al-4V using coated carbide tools ," *European Journal of Scientific Research*, vol.22, no.2,pp. 153 – 162, 2008.

[8] M. Singla, D. Ghosh and K. K. Shukla. A survey of robust optimization based machine learning with special reference to support vector machines, *International Journal of Machine Learning and Cybernetics*, no.23, pp.1-27, 2019.

[9] P.F. Pai and W.C. Hong, " Software reliability forecasting by support vector machines with simulated annealing algorithms," J*ournal of Systems and Software*, vol.79, no.6, pp.747-755, 2006.

[10] Y. Zhang and D. Xin," Dynamic Optimization Long Short-Term Memory Model Based on Data Preprocessing for Short-Term Traffic Flow Prediction," *IEEE Access*, vol.8, pp.91510-91520, 2020.

[11] W. C. Hong, "Traffic flow forecasting by seasonal SVR with chaotic simulated annealing algorithm," *Neurocomputing*, vol.74, no.12, pp. 2096-2107, 2011.

[12] P.F. Pai, and W.C. Hong, "Forecasting regional electric load based on recurrent support vector machines with genetic algorithms," *Electric Power Systems Research*, no.74 , pp. 417-425, 2015.

[13] A. Kavousi-Fard, H. Samet and F. Marzbani, " A new hybrid Modified Firefly Algorithm and Support Vector Regression model for accurate Short Term Load Forecasting," *Expert Systems with Applications*, no. 41, pp. 6047-6056, 2014.

[14] E. Hazir, T. Ozcan and K. Huseyin, " Prediction of Adhesion Strength Using Extreme Learning Machine and Support Vector Regression Optimized with Genetic Algorithm, " *Arabian Journal for Science and Engineering*, no.4, pp.1-20, 2020.

[15] M. Dorigo and L. M. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem," *IEEE Transactions on Evolutionary Computation*, vol.1, no.1, pp.53-66, 1997.

[16] M. Dorigo, V. Maniezzo and A. Colorni, "Ant system: optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 26, no. 1, pp. 29-41, 1996.

[17] W. Deng, J. Xu and H. Zhao, "An Improved Ant Colony Optimization Algorithm Based on Hybrid Strategies for Scheduling problem, " *IEEE Access*, vol.7,no.1, pp. 20281-20292, 2019.

[18] J. E. Bell and P. R. McMullen, "Ant colony optimization techniques for the vehicle routing problem," *Advanced Engineering Information*, vol. 18, no.1, pp.41-48, 2004.

[19] V. N. Vapnik, " The nature of statistical learning theory," *New York: Springer*, p.138, 2000.

[20] V. Vapnik, S. Golowich and A. Smola, "Support vector machine for function approximation, regression estimation, and signal processing," *Advances in Neural Information Processing Systems*, vol. 9, pp. 281-287, 1996.

[21] C.J. Yin, "Titanium Cutting Force Prediction Model and Optimization of Processing Parameters," *Master's thesis, Tianjin University of Technology*, 2014.

[22] Y.L Wang，Y. H. ,Hu，J. B. Sun and T.Q. Wang, "Research on Cutting Force of High Speed Cutting Titanium Alloy Based on

Orthogonal Experiment," *Tool Engineering*, vol.50, no.2, pp. 17-19, 2016.

[23] X. Q. Bian, B. Han, Z. M. Du, J. Jaubert, and M. J. Li, "Integrating support vector regression with genetic algorithm for CO2-oil minimum miscibility pressure (MMP) in pure and impure CO2 streams," *Tool Fuel*, vol.182, no.15, pp. 550-557, 2016.

[24] Q. H. Zheng, X. Y. Tian, M. Q. Yang, H. Su, "The Email Author Identification System Based on Support Vector Machine (SVM) and Analytic Hierarchy Process (AHP), " *IAENG International Journal of Computer Science*, vol.46, no.2, pp. 178-191,2019.