Journal of Computers Vol. 32 No. 5, 2021, pp. 15-30
doi:10.53106/199115992021103205002

A Hybrid SVR Model Based on Support Vector Regression and Differential Evolution for Milling Force Prediction of Titanium Alloy

Xiao-Xia Zhang*, Yin-Yin Hu, Jiao Yang

School of Computer Science and Software Engineering, University of Science and Technology LiaoNing, Anshan 114051, China
aszhangxx@163.com, 2799386740@qq.com, 2036588541@qq.com

Received 29 January 2021; Revised 30 May 2021; Accepted 2 July 2021

Abstract. Prediction of milling force plays an important role in milling process of titanium alloy. In this paper, the milling process of titanium alloy is studied, and the material quality is affected directly by the milling force. Support vector regression (SVR) has shown a prominent performance for many practical applications. Although there is some literature about parameter optimization techniques of SVR model, it still needs further research and improvement on the performance and accuracy of this model. We present a hybrid milling force prediction model, namely DE&SVR, which hybridizes the SVR with differential evolution (DE) to enhance the prediction accuracy for milling force of titanium alloy. The main advantage of hybrid model is that the DE is adopted to optimize the kernel parameters of the SVR. The main parameters affecting milling force, such as the milling depth, feeding speed, and cutting speed, are considered in this study. The results have shown that the hybrid model yields better prediction accuracy, and the percentage prediction milling force deviation is found to be less than 3.5% for all the cases tested, *NRMSE* is only 0.0200, and *MAPE* is only 1.4791%. Thus, this methodology can be widely applied to the fields of material processing optimization.

Keywords: differential evolution, BP neural network, support vector regression, milling force prediction

1 Introduction

In recent years, the titanium alloys are widely used in modern titanium industries because of many advantages, such as high specific strength, high temperature resistance characteristics, excellent corrosion resistance, good welding performance, and so on. These materials are generally used in chemical industry, medical treatment, aerospace, automobile and other fields [1-2]. With the wide application of titanium alloy in various industries, the high speed machining technology of titanium alloy has become a difficult problem in manufacturing industries. The milling force is an important physical parameter in titanium alloy milling process, and the changes of its value directly can affect the processing material quality. Hence, it is important to research the influence parameters of milling force in the milling process. In titanium industries, the major process parameters affecting milling force are summarized as milling depth, feeding speed and cutting speed. In this work, we mainly studies titanium alloy milling process, in which, a typical problem is how to build an optimized prediction model, which provides accurate prediction of the milling force according to process parameters. Because the titanium alloy material has the characteristics of low elastic modulus, low wear-resistant cutting tool, low thermal conductivity and excellent high temperature in milling process, it is difficult to be machined. So the researches of titanium alloy machining, especially milling performance and optimizing prediction model, have a practical significance of improving the processing efficiency, cutting down manufacturing cost and expending the applications of titanium alloys.

* Corresponding Author

At present, the optimization of milling process parameters of titanium alloy is an important study topic, and it has also been widely applied recently in many industries. Though the development of processing technology has made great progress in titanium industries, researchers are still seeking to improve performance. Baker [3] studied the influence of cutting speed on cutting force. Elmagrab et al. [4] explored experimentally the effect of cutting parameters of feed rate on the surface integrity by milling of titanium alloy. The results showed that it has a great influence on the surface roughness. Wang et al. [5] established the simulation model of cutting titanium alloy based on the principle of finite element approximation. The results showed that the deviation of the main cutting force between simulation results and measured results was less than 10%. Özel et al. [6] established the three-dimensional mathematical model of the titanium alloy by the finite element approach and discussed the effect of the different technological parameters on the cutting force, temperature, and the surface roughness. Taking the cutting force, low surface roughness, and processing benefit as optimization objectives, André et al. [7] proposed a genetic algorithm to seek optimal milling parameters. Tansel et al. [8] also designed a hybrid genetic algorithm with neural network model to discuss the influence of milling speed, the feed rate and the depth of cutting on surface roughness of titanium alloy. Liu et al. [9] adopted the optimization algorithm based on the kriging interpolation and genetic algorithm for the titanium alloys, and the results indicated that the optimum parameter levels for different variables have been suggested. Wu et al. [10] proposed a cutting force prediction model in milling process by increasing chip thickness. Experiment results showed that simulated results are consistent with measured results.

Support vector regression (SVR) model is based on the SVM, ant it was originally applied to solve problems of pattern recognition and classification. The SVR model is a novel machine learning method that is the basis of converting low-dimensional nonlinear functions to multidimensional space. With the discovery of ε-insensitive function, the SVM model is conveniently expanded to help solve estimation problems of nonlinear regression, namely SVR model. The SVR model has been developed to solve many practical forecasting problems and achieved remarkable results [11], such as engineering and software field forecasting [12], traffic flow forecasting [13-14], electric load forecasting [15-16], adhesion strength of coating performance [17], and so on.

As discussed in the published literature, the main research is focused on proposing different models of improved GA for milling force prediction of titanium alloy, but there are several aspects in the existing models to be further discussed as follows.

(1) The basic principle of genetic algorithm (GA) is survival of the fittest, so it is still easy to trap into local optimum.

(2) It has also been proven that SVR computational complexity increases, as the problem size and the number of samples increase. How to effectively apply it to massive datasets is still a serious challenge.

(3) The actual results have shown that the prediction accuracy is still not ideal because of the lack of knowledge of parameters selection in the SVR model. There is some literature about parameter optimization technology of SVR model, but the accuracy of this model needs further research and discussion.

Inspired by these aspects, a new milling force predicting model (DE&SVR) is proposed in this paper. The technical achievements of this paper are summarized as follows.

(1) The main innovation of the hybrid DE&SVR model is to combine the differential evolution (DE) with the support vector regression (SVR) model for milling force prediction of titanium alloy.

(2) Within the hybrid model, the DE algorithm is adopted to automatically optimize three parameters of SVR model for increasing the forecasting accuracy.

(3) The selection strategy originates from the natural survival principle of the fittest, and this selection rule makes the search easily trap into local optimum. To avoid being trapped in local minima, we design a novel selection strategy with probabilistic escaping mechanism to determine whether the trial vectors will be one of target vectors in the next iteration.

(4) Experimental results show that proposed model is very efficient in terms of forecasting accuracy.

The remainder of this paper is organized as follows. Section 2 introduces differential evolution (DE) algorithm. Section 3 describes support vector regression (SVR) model. The proposed DE&SVR forecasting is presented in detail in Section 4. Experimental simulation results are presented in Section 5. Finally, some conclusions and the future research suggestions are provided in Section 6.

2 Differential Evolution

Differential evolution (DE) is considered as a new meta-heuristic based on a parallel search algorithms. Storn and Price [18] addressed the first DE algorithm to solve real-world problems. The main idea of this algorithm principle is to make the population evolve through mutation, crossover, and selection operations at each iteration. Starting from target vectors, the evolution process adopts a mutation operation to produce mutant vectors. Then, to improve the diversity of current population, these mutant vectors are combined with target vectors to generate trail vectors by implementing the crossover process, in which the process can enable the trail vector to obtain some attributes inheriting from the mutant vector. Then, a selection strategy will be performed. The target vector or the trail vector will evolve to the next generation according to the evolutionary principle of survival of the fittest, and the better vector will be chosen to survive into the next iteration. The above operations are repeated until a predefined termination condition is the initial population size, the employed mutation strategy, crossover factors, selection strategy, and so on. Therefore, a large number of variants are proposed in literature to adjust these factors to enhance the performance of DE method. A comprehensive survey about the DE algorithm can be found at the work proposed in [18]. DE algorithm adopts a random search strategy, which can record the best solution and optimal information within the current population. The complexity of simple genetic operation is reduced and the global convergence ability is improved. As being applied to continuous optimization, DE algorithm directly manipulates the floating-point numbers and can be easily used for solving parameter optimization problems, so it reduces the computational cost for processing encoding and decoding. Since the DE algorithm has some advantages, i.e., simplicity, few parameters, fast convergence, and so on, it has been successfully used in scheduling [19], satellite image registration [20], biogeography [21].

3 Support Vector Regression (SVR) Model

Support vector machine (SVM) is typical supervision learning and also satisfies statistical learning theory. The SVM has originally been adopted for classification purposes, but, with the discovery of ε-insensitive loss function [22], the theory was expanded easily to regression prediction. The SVR model, a version of a SVM for regression, is able to solve nonlinear estimation problems effectively. In this paper, the SVR model will be used.

The SVR model is a robust approximation technique based on machine learning theory [23], and its idea of SVR model is to convert the nonlinear original samples into a high dimensional line space. In the high dimensional feature, the SVR model seeks to find such a linear function to describe the nonlinear relationship between input samples and output samples. Given a training set $T = \{(x_1, y_1), (x_2, y_2), (x_i, y_i), \ldots, (x_n, y_n)\}$, $y_i \in \{-1, 1\}$, where x_i is an input vector, and $y_i \in R$ is the actual output value for $i = 1, 2, \ldots, n$, in which n denotes the sample size in the training set. The input x_i is first converted into a high dimensional feature space adopting a feature function $\phi(x)$, and then a linear SVR function can be generally defined as formula (1).

$$f(x) = w^T \phi(x) + b\,, \tag{1}$$

where $\phi(x)$ denotes the nonlinear function converting the input examples into a higher dimensional feature space; $f(x)$ is forecasting values; $\boldsymbol{w}$ denotes the weight vector, and b denotes the bias coefficient. Nonlinear regression is transformed into linear regression. With referring to a penalty function, the coefficients ($\boldsymbol{w}$ and b) can be evaluated as follows:

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

In the SVR model, our purpose is to seek a function $f(x)$ with at most ε deviation, and in the meantime it is as flat as possible. That is to say, it is concerned that the error is less than ε, but doesn't allow any deviation greater than this. The most common SVR model adopts an ε-insensitive loss function as follows:

$$|y_i - f(x)|_\varepsilon = \begin{cases} 0 & if \quad |y_i - f(x)|_\varepsilon \le \varepsilon \\ |y_i - f(x)| - \varepsilon & \text{otherwise} \end{cases}, \tag{3}$$

where y_i refers to target data. Formula (3) is employed to find out the optimum hyper plane and minimize the deviation error. The slack variables ξ_i and ξ_i^* denote the distance between the actual values and boundary ε values, and they are introduced to deal with infeasible constraint problem as described in formula (2). Fig. 1 depicts the linear regression performed graphically by support vector regression. Only the points in the range of -ε and ε parameters can make a contribution to the prediction accuracy, while the points outside the region have a contribution to the error, as the deviations should be penalized in a linear pattern.

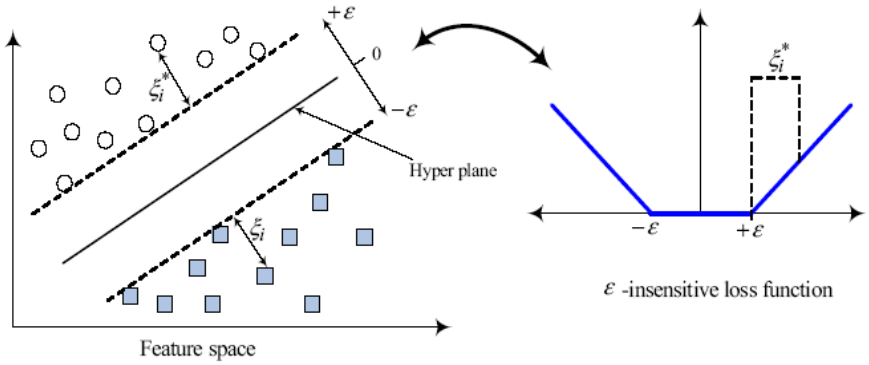

Fig. 1. Transformation process for a linear SVR model

By replacing formula (2) with formula (3), we may seek a function which suits the training set with a low deviation, which is less than or equal to ε. The introduction of slack variables, i.e., ξ_i and ξ_i^*, makes the infeasible constraint optimization problem formula (2) reformulate into a minimizing constrained optimization problem. The optimal SVR function is determined from the estimation of $\boldsymbol{w}$ and b as follows:

$$\text{Minimize } \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\xi_i^* + \xi_i),$$
$$s.t.\begin{cases} f(x_i) - y_i \le \varepsilon + \xi_i^* \\ y_i - f(x_i) \le \varepsilon + \xi_i \\ \xi_i \ge 0, \xi_i^* \ge 0, i = 1, 2, ..., n \end{cases}, \tag{4}$$

where w^2 represents the regularization term [17]. The penalize parameter C is regarded to make a coordination between the empirical risk and the model flatness. The parameter ε is the insensitive loss function, and the value of parameter ε determines the number of support vectors. The dual of the problem may be solved by adopting convex programming techniques.

In most cases, using the method of Lagrange multiplies; the constrained optimization problem shown in formula (4) may be solved more easily by transforming it into its dual formulation. The core principle is to build a Lagrange function based on the objective and its constraints by using a dual variable set. By introducing the Lagrange equation, the dual optimization problem is obtained as formula (5),

$$\max(\beta_i, \beta_i^x) = \sum_{i=1}^{n} y_i(\beta_i - \beta_i^*) - \varepsilon\sum_{i=1}^{n}(\beta_i + \beta_i^*) - \frac{1}{2}\sum_{i,j=1}^{n}(\beta_i - \beta_i^*)(\beta_j - \beta_j^*)k(x_i, x_j),$$
$$s.t.\begin{cases} \sum_{i=1}^{n}(\beta_i^* - \beta_i) = 0 \\ 0 \le \beta_i, \beta_i^* \le C \end{cases}, \tag{5}$$

where β_i, β_i^* are Lagrange multipliers, which meet the equality $\beta_i\beta_i^*$=0. The Lagrange multipliers can be obtained by solving quadratic optimization problems with inequality constraints. After β_i and β_i^* are calculated, the vector $\boldsymbol{w}$ is rewritten in formula (1) as follows,

$$w = \sum_{i=1}^{n} (\beta_i - \beta_i^*) x_i . \tag{6}$$

To get no-linearity, the training data pattern $\boldsymbol{x}_i$ may be mapped to a multidimensional feature space. This kernel is define as formula (7),

$$k(x_i, x_j) = \phi(x_i)^T \phi(x_j) . \tag{7}$$

After transforming the problem described above to a Lagrangian problem and simultaneously solving this dual problem, formula (1) becomes the explicit mathematical expression. For an input vector, the output, namely the nonlinear decision function of SVR regression can be determined as formula (8),

$$f(x) = \sum_{i=1}^{n} (\beta_i - \beta_i^*) k(x_i, x) + b, x_i \in R^n, b \in R, \tag{8}$$

where $k(x_i, x)$ represents the non-linear kernel function which converts the original non-linearly input data as multidimensional feature space, and the kernel value is defined as the scalar product of vectors, x_i and x. The introduction of kernel function can enable SVR model to easily find the solution to the non-linear regression problem. The SVR model provides several different types of kernel functions such as radial basis, lineal, polynomial and sigmoid functions, which can specify the various characteristics of models. In comparison with some other kernel functions, the Gaussian radial basis function (RBF) is the most widely used kernel function, and it can have a better performance than polynomial kernel function. Because of fewer parameters to be set, the Gaussian radial basis function is not only convenient to execute, but also it performs nonlinearly converting between the input space and a high-dimensional space, and thus it is appropriate to solve nonlinear problems. Because of its flexibility in treating more complex parameters and its better stability in the course of nonlinear model, in this study, the data set presents obvious nonlinearity, so we choose the Gaussian RBF as the kernel function $k(x_i, x)$ as formula (9),

$$k(x_i, x) = \exp(-\frac{1}{2\delta^2} \| x - x_i \|^2), \delta > 0 . \tag{9}$$

In formula (9), δ is the kernel parameter which denotes the structure of high dimensional feature space. Variables x_i and x are input vectors of training data and test data respectively. In summary, the SVR model is similar to a neural network in form. 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, each hidden node should correspond to one support vector, and weights, namely the Lagrange multipliers, control the relative influence of the training data for the final result, as illustrated in Fig. 2.

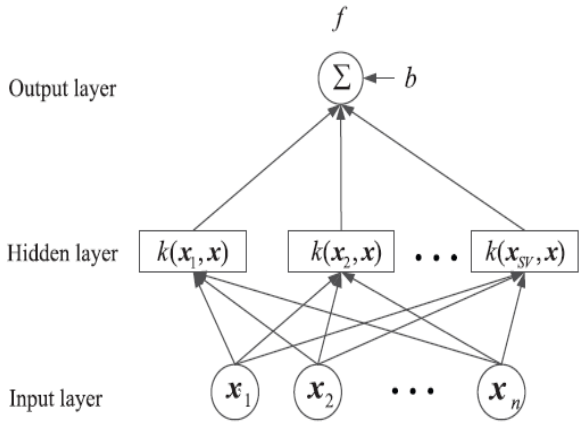

Fig. 2. SVR prediction based on phase space reconstruction

Generally, there exist three parameters (C, δ and ε) to be selected in the SVR. In practical application, the parameters also have a great effect on the accuracy of SVR models. Therefore, to increase forecasting

accuracy, we propose a hybrid model which combines the SVR with the DE algorithm to better estimate the hyper-optimized parameters (C, δ and ε). The next section will describe the hybrid model in detail.

4 Hybrid SVR Prediction Model Based on DE&SVR

The accuracy of the SVR performance mainly depends on three parameters, namely, penalty parameter C, kernel function parameter δ, and non-sensitivity coefficient ε. These parameters should be optimized, because they directly affect the feature space and the prediction performance of the model. However, there are few studies on the structural methods for effective parameter selection. Consequently, it is necessary to seek optimization methods to optimize C, δ and ε to enhance the learning ability of SVR. In this work, we proposed a hybrid DE&SVR model for optimizing parameters of the SVR to forecast milling forces. In the hybrid DE&SVR model, the DE algorithm is used to optimize the three parameters of the SVR, which are applied to build the SVR model of forecasting milling forces. Fig. 3 shows the process of hybrid DE&SVR model.

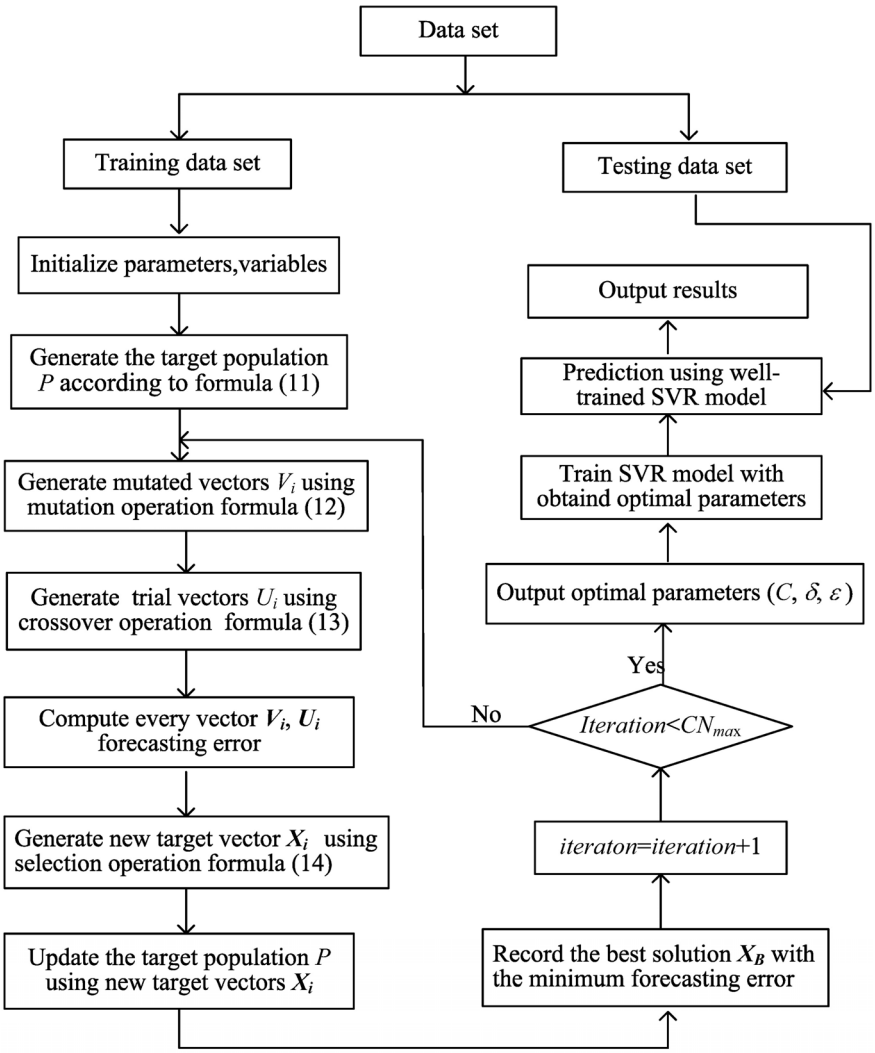

Fig. 3. The flowchart of DE&SVR model

The choice of parameters, C, δ and ε of the SVR model influences the accuracy of forecasting. 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 DE to seek the best parameter values for C, δ and ε in SVR. As discussed in the section 2, in the process of parameter optimization, starting from target vectors $\boldsymbol{P}$, crossover operation and mutation operation are performed on the current population to produce a trail vector population, $\boldsymbol{U} = \{\boldsymbol{U}_1, \boldsymbol{U}_2, \ldots, \boldsymbol{X}_{NP}\}$. Then, the greedy selection operation is used to select the two populations one-to-one

to produce a new population $\boldsymbol{P}$. These operations should be repeated until a given termination criterion is met, and then output optimization parameters, which will be brought into the training data set to construct forecasting SVR model. The general framework of the proposed DE&SVR consists of initialization, mutation, crossover, selection and evaluating function for parameter optimization.

4.1 Population Initialization

Firstly, let $\boldsymbol{P}$ denote the population $\boldsymbol{P} = \{\boldsymbol{X_1}, \boldsymbol{X_2}, \ldots, \boldsymbol{X_i}, \ldots, \boldsymbol{X_{NP}}\}$, in which each initial solution or individual $\boldsymbol{X_i} = (x_{i,1}, x_{i,2}, ..., x_{i,j}, ..., x_{i,D})$ is an D-dimension vector, $x_{i,j}$ denotes the jth parameter of the vector $\boldsymbol{X_i}$, $i = 1, 2, ..., NP$, $j = 1, 2, ..., D$, NP represents population number, and D represents the search space dimension, which is used to characterize the optimization problem solution. The initial population includes the entire solution search space by randomizing the parameters uniformly within the bounds of the constrained search space $X_{min} = \{x_{min,1}, ..., x_{min,D}\}$ and $X_{max} = \{x_{max,1} ..., x_{max,D}\}$. NP solutions are randomly produced in the feasible space. Each initial solution $\boldsymbol{X_i} = (x_{i,1}, x_{i,2}, ..., x_{i,j}, ..., x_{i,D})$ is also known as a target vector, and the values of the jth element of the ith individual is obtained by:

$$x_{i,j} = x_{min,j} + \text{rand}(0, 1) \cdot (x_{max,j} - x_{min,j}) \; j = 1, 2, \ldots, D, \tag{10}$$

in which rand(0, 1) denotes a uniformly distributed variable.

4.2 Mutation Operation

Following the population initialization, the algorithm evolves to mutation operator stage. In the population, each vector X_i, named as the target vector, will generate a mutant vector V_i. This mutation vector $V_i = \{v_{i,1}, v_{i,2}, ..., v_{i,j}, ..., v_{i,D}, i = 1, 2, ..., NP, j = 1, 2, ..., D\}$, is generated via mutation operation strategy by using three randomly chosen target vectors. The DE algorithm adopts the mutation procedure to generate a mutation vector V_i with regard to each given target vector X_i in the current population by calculating the weighted difference value between randomly chosen solutions from the current population. That is to say, three target vectors X_{p1}, X_{p2}, X_{p3} will be randomly chosen from the current population such that indices $p_1, p_2, p_3 \in \{1, 2,\ldots, NP\}$ and these indices must be diverse from each other. For the target vectors X_i, $i = 1, 2,\cdots, NP$, the mutant vector V_i is then computed as follows:

$$V_i = X_{p_1} + F \cdot (X_{p_2} - X_{p_3}) \; p_1 \neq p_2 \neq p_3 \neq i, \tag{11}$$

where F denotes a constant real value $F \in [0, 2]$, a mutation parameter, which determines the magnification of the differential variation between the second and the third vectors, i.e., $(X_{p2} - X_{p3})$. Moreover, this differential variation includes 4 vectors, so the population size NP is supposed to be at least 4 solutions. In the evolution process, we should guarantee that the parameters C, δ, ε should meet the boundary conditions. If an element of the mutant vector is out of the given range, it is set as a bound value.

4.3 Crossover Operation

After finishing the mutation operation, a crossover operation should be executed to enhance the global searching ability. The crossover operation is the most important operation of DE algorithms, and it is the key to the convergence of algorithms. In this phase, a trail vector, $\boldsymbol{U_i} = (u_{i,1}, u_{i,2}, ..., u_{i,D})$, $i = 1, 2, ..., NP$, is obtained as a recombination of the target and the mutant vectors in which this process can enable the trail vector to obtain several attributes inheriting from mutant vectors. The crossover operation can extend the diversity of the perturbed vectors by swapping the elements between a target vector and its mutant vector by a crossover probability parameter $p_{cr} \in [0, 1]$, which is a crossover probability constant determining the indices of inherited features. For a target vector $\boldsymbol{X_i}$, a trail vector $\boldsymbol{U_i} = (u_{i,1}, u_{i,2}, ..., u_{i,D})$ is produced by recombining $\boldsymbol{X_i}$ with corresponding mutated vector $\boldsymbol{V_i}$ using the following scheme:

$$u_{i,j} = \begin{cases} v_{i,j}, & \text{if } rand(j) \leq p_{cr} or \; j = r \\ x_{i,j}, & \text{otherwise} \end{cases}, \tag{12}$$

where $u_{i,j}$ represents the jth parameter of trial vector $\boldsymbol{U_i}$, $i = 1, 2,\ldots, NP$, $j = 1, 2, \ldots, D$, $rand(j) \in (0, 1)$ is a uniform distribution value for determining the jth dimension of the trial vector $\boldsymbol{U_i}$, and $r \in \{1, 2,\ldots, D\}$ is a random index. The purpose of adopting the parameter r is to ensure that $\boldsymbol{U_i}$ gets at least one attribute from $\boldsymbol{V_i}$ if the condition $j = r$ is satisfied. Otherwise, the remaining elements of the trail vector $\boldsymbol{U_i}$ are obtained from the vector $\boldsymbol{X_i}$. Then $\boldsymbol{U_i}$ is evaluated with the objective function $f(\boldsymbol{U_i})$ to prepare for the next selection operation.

4.4 Selection Operation

After crossover operation, the trail vector $\boldsymbol{U_i}$ and target vector $\boldsymbol{X_i}$ are calculated by the objective function values. The main purpose of selection phase is to define whether a target vector or a trail vector will evolve to the next iteration according to the survival of the fittest. In selection operation, the trial vectors are calculated which adopt the traditional greedy selection strategy to update the current population. If $\boldsymbol{U_i}$ has less function values than $\boldsymbol{X_i}$, not only $\boldsymbol{U_i}$ will replace $\boldsymbol{X_i}$, but also the corresponding parameters are regarded as the successful parameters. Otherwise, $\boldsymbol{X_i}$ will remain unchanged in the new generation. The vectors with the lower function values will be selected to survive to the next iteration. The greedy selection scheme is described as follows:

$$X_i = \begin{cases} U_i, & if\ f(U_i) \le f(X_i) \\ X_i, & \text{otherwise} \end{cases}, \tag{13}$$

in which $f(\boldsymbol{U_i})$, $f(\boldsymbol{X_i})$ denote the function values of $\boldsymbol{U_i}$ and $\boldsymbol{X_i}$, respectively. That is, these parameters with fewer test errors have more probability of being selected in future. The selection strategy is from the survival principle of the fittest between $\boldsymbol{U_i}$ and $\boldsymbol{X_i}$, and this selection rule makes the search easily trap into local optimum. Hence, to avoid falling into local minima, we have proposed a new selection strategy with probabilistic escaping mechanism to determine whether the trial vector will be an individual of target vectors in the new generation. The formula is defined as follows,

$$X_i = \begin{cases} U_i, & if\ rand() < \min\{1, e^{(f(U_i)-f(X_i))/kT}\} \\ X_i, & \text{otherwise} \end{cases}, \tag{14}$$

where k denotes constant value and T represents temperature. By adopting this new selection strategy, a certain inferior solutions as target vectors can be accepted with a probability base on formula (14) during the search process. Thus, the target vectors can be diversified, and the proposed algorithm has a chance to escape from the local optimum.

4.5 Evaluating Function

To our knowledge, the primary purpose of seeking optimum parameters is to promote predictive accuracy by the evaluation criteria. Hence, to improve the prediction accuracy of the milling force, we adopt the DE algorithm to determine proper parameters (C, δ, ε), by which, the main principle of DE algorithm is to seek the optimization fitness function. In the model of parameter optimization, the mean absolute percent error (*MAPE*) is adopted as the fitness function to estimate the accuracy of hybrid model. The mean error is more suitable because it is independent of the data range. The lower the *MAPE*, the more excellent are the forecast values to the actual measured values. The forecasting error can be defined according to formula (15),

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - f_i}{y_i}\right| \times 100\%, \tag{15}$$

where n denotes the number of testing dataset, y_i represents the actual value, and f_i denotes the forecasting value.

4.6 DE&SVR Model

In this paper, to forecast the milling force demands, we present a novel SVR model which hybridizes the SVR model with the DE method to enhance the prediction accuracy. Fig. 3 describes the overall process of hybrid SVR model, which mainly consists of four stages. The effect of SVR prediction model performance mainly depends on three parameters, C, δ, and ε. These parameters should be optimized, because they directly affect the feature of high dimensional space and the prediction performance of models. For the first stage, because the purpose of the SVR model is to obtain the minimum mean absolute percent error (*MAPE*) between the predictions of SVR and actual values of testing set, the range of the parameters should be determined. In the proposed SVR model, cutting parameters were used as inputs and milling force as the output. In the second stage, DE algorithm is applied to optimize the parameters of the SVR to obtain the excellent performance. And then, these obtained optimal parameters will be used to set up the final forecasting SVR model. Finally, the prediction model is adopted to predict the milling force of testing data set, and calculate the forecasting error values. The hybrid DE&SVR model includes the following steps:

Step 1: Data preparation. The samples are partitioned into two sets, namely, training set and test set. Initialize terminal condition, i.e., the maximum number of iterations CN_{max}. Initialize C, δ and ε parameters, and the non-sensitivity coefficient. NP is population size.

Step 2: According to practical optimization problems, we should determine the value range of three parameters $x_{i,j} \in [x_{min,j}, x_{max,j}]$, $j = 1, 2, \ldots, D$. Variable dimension D denotes the number of parameters for optimizing, $D = 3$, i.e., three parameters C, δ, ε, respectively. $x_{min,j}$ and $x_{max,j}$ represent the upper bound and lower bound for parameters. The initial population, namely target individuals, should include the entire solution search space by randomizing the parameters uniformly within the bounds of the constrained search space $X_{min} = \{x_{min,1}, \ldots, x_{min,3}\}$ and $X_{max} = \{x_{max,1} \ldots, x_{max,3}\}$. Generate the target population $\boldsymbol{P} = \{\boldsymbol{X_1}, \boldsymbol{X_2}, \ldots, \boldsymbol{X_{NP}}\}$, in which each individual $\boldsymbol{X_i} = (x_{i,1}, x_{i,2}, x_{i,3})$, $i = 1, 2, \ldots, NP$, is an 3-dimension vector, corresponding to three parameters (C, δ, ε) by using the formula (10), and then evaluate each initial target individual $f(\boldsymbol{X_i})$.

Step 3: Mutation operation. Three target vectors X_{p1}, X_{p2}, X_{p3} are randomly determined from the target population such that indices $p_1, p_2, p_3 \in \{1, 2, \ldots, NP\}$. A mutation vector $\boldsymbol{V_i} = (v_{i,1}, v_{i,2}, v_{i,3})$, $i = 1, 2, \ldots, NP$, is generated via certain mutation strategy by three randomly selected target vectors according to formula (11).

Step 4: Crossover operation. Produce a trial vector, $\boldsymbol{U_i} = (u_{i,1}, u_{i,2}, \ldots, u_{i,3})$, $i = 1, 2, \ldots, NP$, is produced by combining the target vector $\boldsymbol{X_i}$ with the corresponding mutated vector $\boldsymbol{V_i}$ according to the formula (12).

Step 5: Selection operation. Perform the selection on each pair of $\boldsymbol{X_i}$ and $\boldsymbol{U_i}$ to determine new target population for next generation by using formula (13) and (14). Compute the fitness function values of trial vector $\boldsymbol{U_i}$ and target vector $\boldsymbol{X_i}$, $i = 1, 2, \ldots, NP$. If $f(\boldsymbol{U_i}) < f(\boldsymbol{X_i})$ is satisfied, then $\boldsymbol{U_i}$ will replace the target vector $\boldsymbol{X_i}$, i.e. $\boldsymbol{X_i} = \boldsymbol{U_i}$; otherwise, a certain inferior solutions as target vectors can be accepted with a probability base on formula (14).

Step 6: Record the best solution $\boldsymbol{X_B} = (x_{B,1}, x_{B,2}, x_{B,3})$ with the minimum forecasting error.

Step 7: *iteration* = *iteration* +1.

Step 8: Check whether the condition $iteration < CN_{max}$ is met, if $iteration < CN_{max}$, repeat Step3-7 until the terminal criteria condition is satisfied.

Step 9: Stop the train process, and output the global optimal solution $\boldsymbol{X_B} = (x_{B,1}, x_{B,2}, x_{B,3})$, corresponding to three optimization parameters (C, δ, ε).

Step 10: Build the final forecasting SVR model according to these optimal parameters (C, δ, ε). These optimal parameters (C, δ and ε) will be brought into the training data set to construct forecasting SVR model

Step 11: Predict the mill force of testing data set using this model.

Step 12: Calculate the forecasting error values, stop executing the program, and then output the final result.

5 Computational Results

In the section, we evaluated the accuracy of proposed DE&SVR model for the milling force prediction problem. The proposed model was compared with various methods. Because different kernel functions can significantly affect the final prediction performance of SVR model, firstly, we demonstrated performance of SVR model under different kernel functions. Secondly, we compared DE&SVR with basic SVR. Moreover, DE&SVR and the BP Neural Networks were used for comparison. Consequently, total three prediction models were investigated in this milling force forecasting problem. In this section, data set and preprocessing, parameter setting, and forecasting results and analysis are described as follows.

5.1 Data Set and Preprocessing

In the section, we presented results of hybrid proposed model, which was implemented in MATLAB software environment. It is very necessary to provide a wide range of experimental datasets when constructing hybrid model for milling force prediction of titanium alloy. To testify the accuracy of proposed model, the performance of our model was tested on the experimental dataset from the published literature [1, 24-25]. According to above description, the major parameters affecting milling force are summarized as milling depth (A), feeding speed (f), and cutting speed (v). Table 1 gives the description of these parameters in detail. In this experimental dataset, these three parameters are adopted as input parameters and the milling force is used as the forecasting value of output parameters through optimized SVR model. There are 96 samples in the experimental data. To guarantee fair representation of data sets, these samples are randomly partitioned into two sets, namely, training set and test set, and they are arranged by the training set before the test set. Fig. 4 shows the actual values of milling force in the experimental dataset. The first 81, the training set, is applied to train models, and the remaining 15, the test set, is used to estimate predicting efficiency of models. Table 2 lists test data of processing parameters and milling force.

Table 1. The parameter description in experiments

No.	Terminology	Symbol	Range of parameters
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

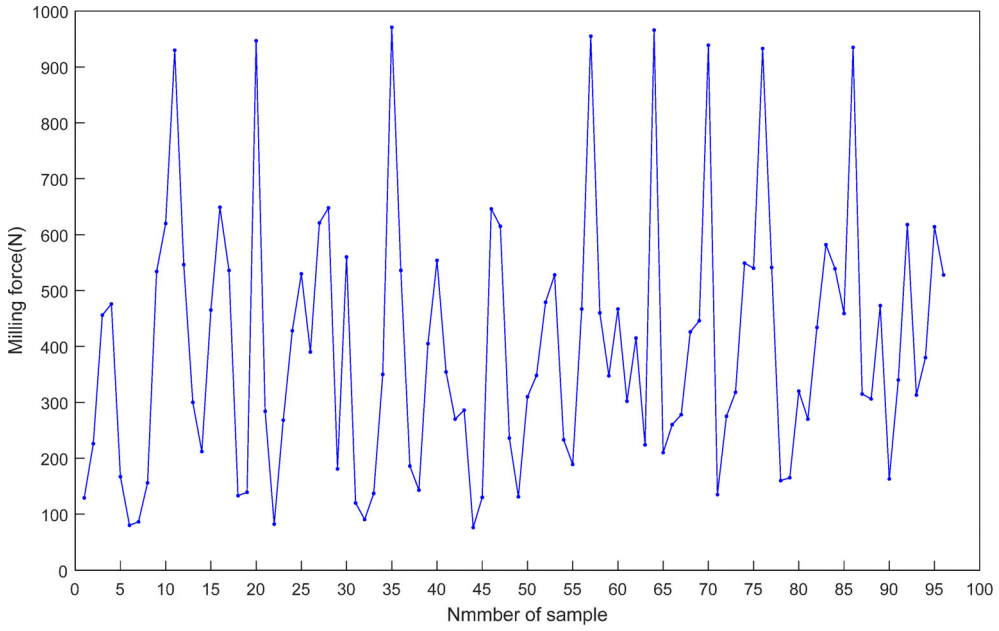

Fig. 4. The actual values of milling force

Table 2. Test data of processing parameters and milling force

No.	Input parameters			Output parameters
	A (mm)	*f* (mm/r)	*v* (m/min)	*F* (N)
1	1.4	0.1	100	434
2	0.5	0.6	70	582
3	0.3	0.8	90	539
4	0.9	0.2	90	459
5	0.9	0.4	90	935
6	0.3	0.4	90	315
7	0.3	0.4	70	306
8	0.9	0.2	70	473
9	0.3	0.2	70	163
10	0.5	0.3	50	340
11	0.5	0.6	90	618
12	0.3	0.4	60	313
13	1.4	0.06	90	380
14	0.5	0.6	80	614
15	0.3	0.8	70	528

5.2 Parameter Setting

To clearly verify the accuracy of SVR model, four statistical metrics were used as the 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). *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* denotes relative percentage deviation between the predicted and actual milling forces.

Because different kernel functions directly influence the performance of SVR model significantly, we validate performance of four kernel functions. As Table 3 lists, the SVR model with RBF kernel function enhances the accuracy as listed by the reduced *NRMSE* and *MAPE*. As each kernel function needs to optimize different parameters, the grid method was adopted to seek the best kernel function δ, penalty parameter C and the non-sensitivity coefficient ε for SVR mode. While the training errors are reduced, modify three parameters, C, δ, and ε to compute the deviation errors. After that, the parameters with minimum errors may be chosen as the most suitable parameters as shown in Table 4. In Table 3, a comparison of different kernel functions shows that the SVR with RBF kernel has the smaller *NRMSE* and *MAPE*, and it has the highest coefficient of determination in both training set and testing set. Therefore, the SVR model with RBF kernel can provide more accurate predictions. In DE algorithm, we set parameter values as follows: a crossover parameter $p_{cr} \in [0.45, 0.70]$, population size $NP = 20$, $CN_{max} = 1000$.

Table 3. Performance of SVR model with different kernel function

Data set	Statistical parameters	Kernel function			
		Linear	Polynomial	Sigmoid	RBF
Training set	*MAPE* (%)	26.4675	14.4576	34.5006	7.2383
	NRMSE	0.2842	0.1249	0.3033	0.0641
	R^2	0.7026	0.9425	0.6613	0.9849
Testing set	*MAPE* (%)	16.6678	8.9031	18.688	5.0893
	NRMSE	0.2578	0.0939	0.2538	0.0641
	R^2	0.4772	0.9306	0.4931	0.9676

Table 4. Parameters of SVR model with different Kernel function

No.	Kernel function	C	δ	ε
1	Linear kernel	0.2852	16	0.01
2	Polynomial kernel	110.000	4.450	0.01
3	Sigmoid kernel	0.9266	0.2832	0.01
4	Radial basis function kernel (RBF)	8.000	1.250	0.01
5	DE&SVR	3.6553	0.7792	0.01

5.3 Forecasting Results and Analysis

As we have discussed, the performance of SVR model mainly depends on three parameters, i.e., penalty parameter C, kernel function parameter δ, and non-sensitivity coefficient ε. The main purpose of optimization parameters for the SVR is to optimize the process sufficiently, which can search a finite parameter subset of all possible values in order to obtain optimal parameters with minimum generalization error. To enhance the performance of SVR model in milling force forecasting, we proposed a novel DE&SVR model, in which we adopted the DE algorithm to determine the most suitable parameter values of SVR model. After getting optimal parameters, using these parameters can generate the SVR model to implement the prediction task.

Fig. 5 and Fig. 6 give the comparison between the predicted milling force of DE&SVR model and the actual force of training set and testing set respectively. The red solid line with red asterisk indicates actual forces, while the blue dashed line with blue circles represents the forecasting force. The predicted milling forces are in good agreement with the actual milling forces as a whole. The forecasting results, the *NRMSE*, *MAPE* and R^2 in testing phases for the models of milling force are illustrated in Table 5 where Actual values refer to measured milling force, SVR to basic SVR, BP model to BPNN, DE&SVR to we proposed, and *Error* denotes relative percentage deviation between the predicted force and actual milling forces. Among the 15 samples, the *Error* value of 13 samples is less than 2.5, and the *Error* value of only 2 samples is between 2.5 and 3.5. When the values of *NRMSE*, and *MAPE* are smaller, the predicted values are closer to the actual values. From Table 5, the *NRMSE* is found to be only 0.0200 for all the cases tested, while the mean absolute percent error (*MAPE*) is 1.4791%. Therefore, we conclude that the DE&SVR model can possess strong self-learning ability and simultaneously obtain the excellent performance. Moreover, because the DE&SVR model is a good compromise scheme, it not only ensures stability but also improves the accuracy.

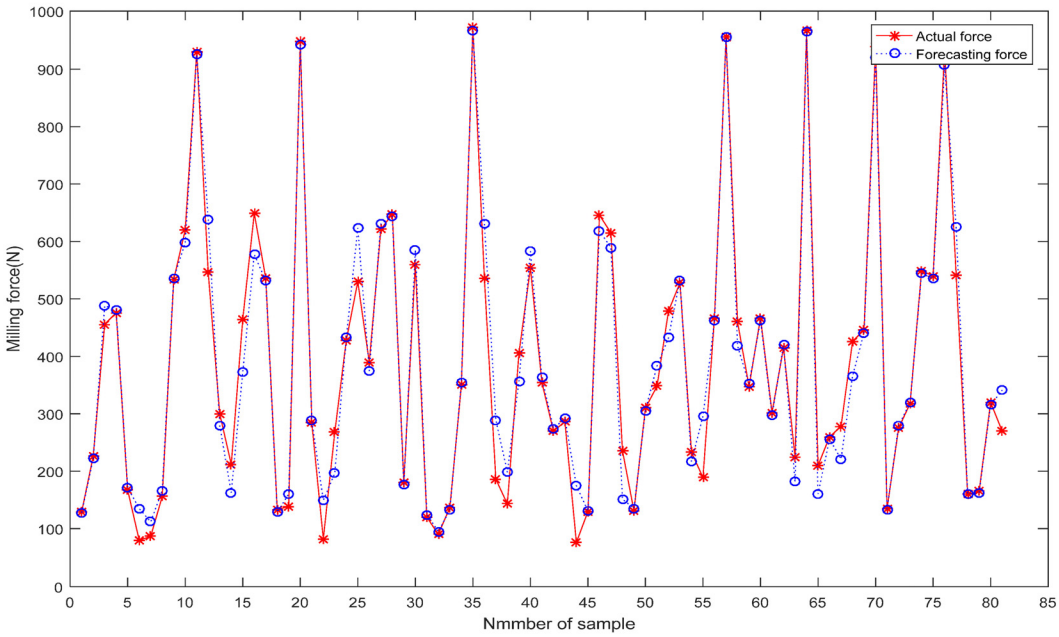

Fig. 5. The result comparison of DE&SVR models and actual values in training set

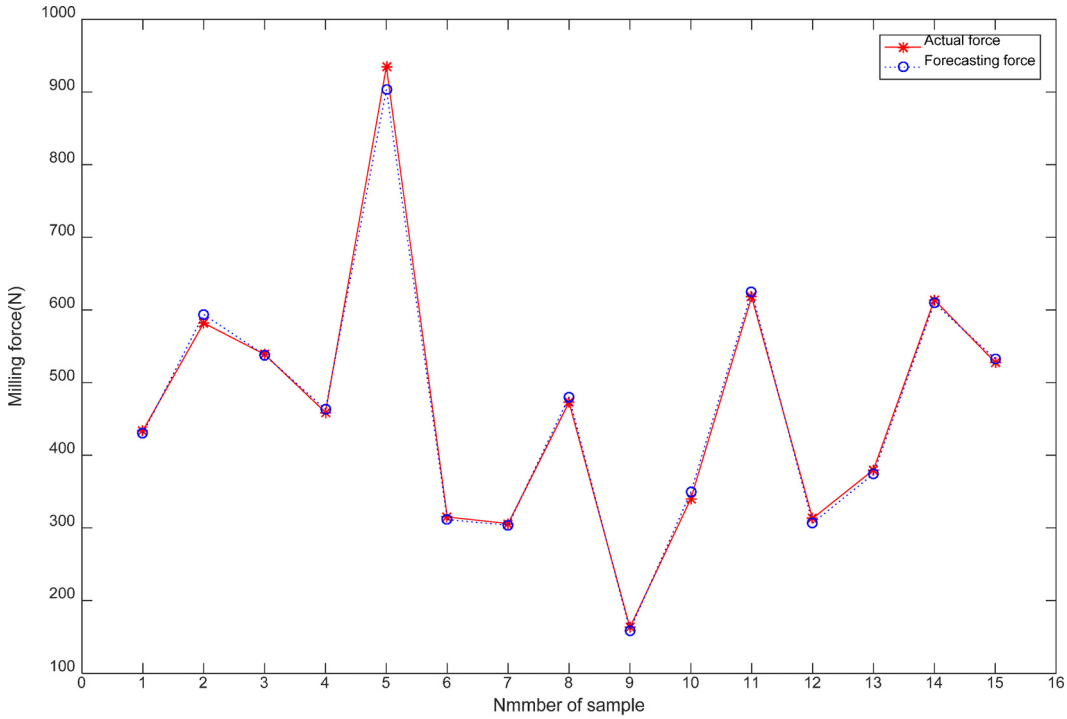

Fig. 6. The result comparison of DE&SVR models and actual values in testing set

Table 5. Comparisons among BP, SVR and DE&SVR model

No	Actual value (N)	BP		SVR		DE&SVR	
		Forecasting value (N)	Error (%)	Forecasting value (N)	Error (%)	Forecasting value (N)	Error (%)
1	434	462.59	6.59	466.38	7.46	429.94	0.94
2	582	698.59	20.03	612.15	5.18	593.93	2.05
3	539	531.83	1.33	543.55	0.84	538.10	0.17
4	459	476.29	3.77	444.86	3.08	463.17	0.91
5	935	903.12	3.41	862.97	7.70	903.58	3.36
6	315	312.41	0.82	291.65	7.41	311.42	1.14
7	306	304.68	0.43	293.04	4.24	303.81	0.72
8	473	467.81	1.10	487.00	2.96	480.32	1.55
9	163	148.15	9.11	159.56	2.11	159.14	2.37
10	340	348.50	2.50	366.61	7.83	349.15	2.69
11	618	688.65	11.43	672.29	8.78	625.17	1.16
12	313	292.56	6.53	284.19	9.20	306.58	2.05
13	380	346.09	8.92	378.00	0.53	374.26	1.51
14	614	695.08	13.21	664.99	8.30	609.70	0.70
15	528	540.81	2.43	531.72	0.70	532.66	0.88
MAPE(%)		6.1070		5.0893		1.4791	
NRMSE		0.08865		0.0641		0.0200	
R^2		0.9382		0.9676		0.9968	

To further verify better forecasting accuracy compared with other models, Table 5 shows the milling force forecasting results of DE&SVR model, basic SVR and BP model. Fig. 7 has shown the forecasting values of three models and actual values. For each model, the evaluation criteria *NRMSE*, *MAPE a*nd *Error* are adopted to represent the degree of deviation between the actual value and forecasting value, while coefficient of determination (R^2) is used to evaluate overall performance. Table 5 gives two comparison schemes. The first comparison scheme is between basic SVR versus DE&SVR model. The DE&SVR model has the ability to adjust automatically the parameters combination (C, δ, ε). But instead, basic SVR model needs more manual manipulation. Compared with basic SVR, the DE&SVR always is consistent with the actual value to obtain the minimum deviation error. But, some forecasting values

obtained from the basic SVR model are inaccurate, which makes *Error* more than 9%. The *MAPE* value of the DE&SVR model is less than that of basic SVR, and the *NRMSE* value of the DE&SVR model is also less than that of basic SVR. The DE&SVR model has a significant advantage over the SVR in terms of *NRMSE*, *MAPE* and *Error*. Thus, the results have shown that the accuracy of DE&SVR prediction model is higher than that of basic SVR model.

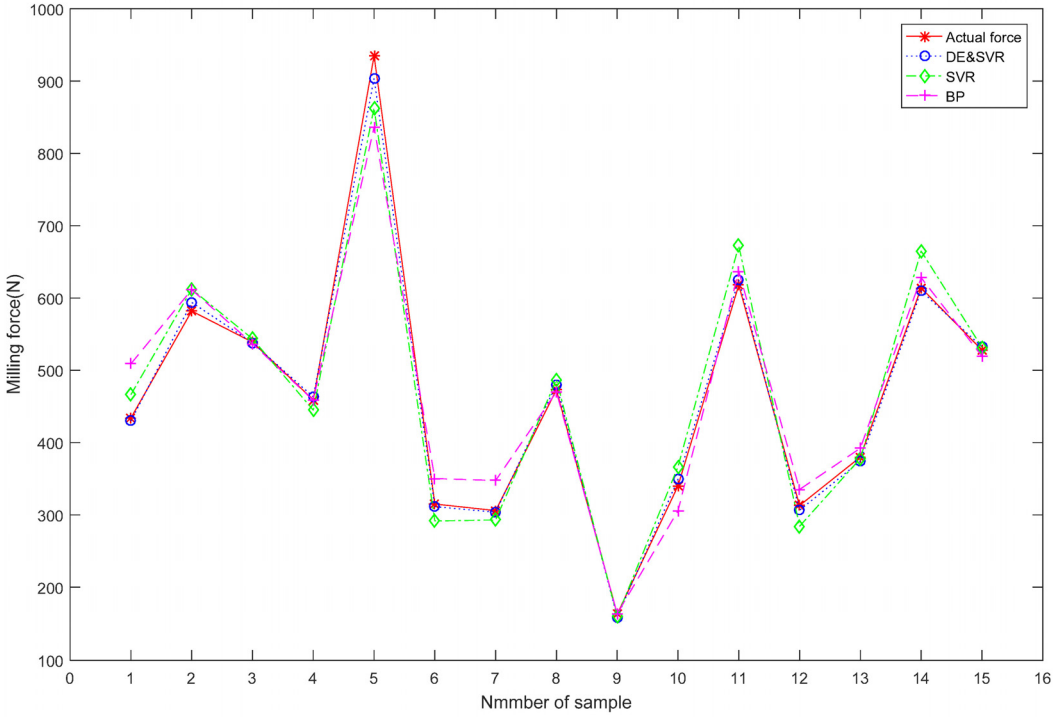

Fig. 7. The forecasting values of three models

The second comparison scheme is between the SVR&DE model and the BP model. The BP model is a common and effective verification method examine because it is the traditional approach for forecasting model in practical optimization problem. When the values of *NRMSE*, *MAPE* and *Error* are smaller, the predicted values are the closer to the actual values. The actual values of milling force and the forecasting values of three models are illustrated in Fig. 7. From Table 5 and Fig. 7, it is easy to see that the BP model has the largest forecasting errors. A few forecasting values of the BP are somewhat inaccurate, which makes *Error* more than 20%. The reason may be because that BP should need a lot of training data to build the training model. Compared with BP, the basic SVR model has a better prediction performance. Moreover, the prediction performance of DE&SVR model is much better than that of basic SVR. It is clear that DE&SVR model performs better than BP model.

Obviously, the hybrid proposed DE&SVR model has smaller *NRMSE*, *MAPE* and *Error* compared with basic SVR and BP model. The percentage prediction milling force deviation is found to be less than 3.5% for all the cases tested, normalized root mean squared error (*NRMSE*) is only 0.0200, and the mean absolute percent error (*MAPE*) is only 1.4791%. Moreover, the proposed DE&SVR model has larger R^2 compared with basic SVR and BP models. The coefficient of determination R^2 is as high as 0.9968, almost close to 1, and this indicates a perfect model. From the above analysis, we can conclude that proposed DE&SVR model has more accurate prediction than BP and basic SVR in forecasting milling force.

Aiming at the milling process of titanium alloy, we have put forward a milling force prediction model based on the DE and SVR model. In this study, we first adopted the DE algorithm to optimize three important parameters of SVR model, and then used these optimal parameters (C, δ and ε) to establish the final SVR model. The well-trained model is adopted to predict the test data and compute the forecasting error values. The optimized parameters of the DE algorithm can enhance the efficiency of SVR model for forecasting processing. The prediction of milling force based on the DE&SVR model can be analyzed conveniently on the physical phenomena in the process of tool wear and processing quality.

6 Conclusions

In this article, we have designed a prediction model of milling force based on support vector regression (SVR) and differential evolution (DE) according to process parameters in modern titanium alloy industries. Within the hybrid model, the DE algorithm is adopted to automatically optimize three parameters of SVR model for increasing the forecasting accuracy. The experiment results have shown that: firstly, because different kernels can have influence on final performances of the SVR, we demonstrate that the SVR model with RBF kernel can provide more accurate predictions. Secondly, the DE&SVR model always is consistent with actual values to obtain minimum deviation errors compared with basic SVR. Finally, the proposed DE&SVR model has lower error compared with BP model. Obviously, the hybrid proposed DE&SVR model has smaller *NRMSE*, *MAPE* and *Error* compared with basic SVR and BP model. The percentage prediction milling force deviation is found to be less than 3.5% for all the cases tested, *NRMSE* is only 0.0200, and *MAPE* is only 1.4791%. Moreover, the coefficient of determination R^2 is as high as 0.9968, almost close to 1, and this indicates a perfect model. Thus, this methodology can be expanded to the fields of material processing parameters optimization. In this research, the main limitation mainly lies in that the instances are not large enough. Therefore, we will make further efforts to enhance the prediction performance of the DE&SVR model, especially in cases of large-scale instances. For our future research, other advanced optimization methods about parameters optimization and other improved strategies will be integrated into the SVR forecasting model.

Acknowledgements

This work was supported by Project of Liaoning Xincheng Co., Ltd (Grant No. L20170989) and national college students' innovation and entrepreneurship training program project in 2021.

References

[1] G.Q. Xiang, T. Lu, Prediction analysis of titanium alloy milling force based on support vector machine, Machine Tool & Hydraulics 44(2016) 142-146.

[2] E.O. Ezugwu, J. Bonney, Y. Yamane, An overview of the machinability of aeroengine alloys, Journal of Materials Processing Technology 134(2)(2003) 233-253.

[3] M. BaKer, Finite element simulation of high-speed cutting forces, Journal of Materials Processing Technology 176(1-3) (2006) 117-126.

[4] N. Elmagrabi, C.H. Che Hassan, A.G. Jaharah, High speed milling of Ti-6Al-4V using coated carbide tools, European Journal of Operational Research 22(2)(2008) 153-162.

[5] L. Wang, Q.C. Zheng, Y.H. Hu, C.L. Yang, Simulation and experimental research of cutting force in titanium alloys, Journal of Chinese Agricultural mechanization 37(2016) 280-284.

[6] T. Özel, M. Sima, A.K. Srivastava, 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 59(1)(2010) 77-82.

[7] F.H.L. André, N.L. Coppini, E.A. Baptista, Genetic algorithm applied to investigate cutting process parameters influence on workpiece price formation, Material and Manufacturing Processes 26(2011) 550-557.

[8] I.N. Tansel, S. Gülmez, M. Demetgul, Taguchi method-GONNS integration, Expert Systems with Applications 38(2011) 4780-4789.

[9] C.J. Liu, D.B. Tang, H. He, 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 32(4)(2013) 469-474.

[10] B.H. Wu, X. Yan, M. Luo, G. Gao, Cutting force prediction for circular end milling process, Chinese Journal of Aeronautics 26(4)(2013) 1057-1063.

[11] M. Singla, D. Ghosh, 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 11(3)(2020) 1-27.

[12] P.F. Pai, W.C. Hong, Software reliability forecasting by support vector machines with simulated annealing algorithms, Journal of System and Software 79(6)(2006) 747-755.

[13] Y. Zhang, D. Xin, Dynamic optimization long short-term memory model based on data preprocessing for short-term traffic flow prediction, IEEE Access 8 (2020) 91510-91520.

[14] W.C. Hong, Traffic flow forecasting by seasonal SVR with chaotic simulated annealing algorithm, Neurocomputing 74(12-13)(2011) 2096-2107.

[15] P.F. Pai, W.C. Hong, Forecasting regional electric load based on recurrent support vector machines with genetic algorithms, Electric Power Systems Research 74(3)(2015) 417-425.

[16] A. Kavousi-Fard, H. Samet, F. Marzbani, A new hybrid modified firefly algorithm and support vector regression model for accurate short term load forecasting, Expert System Applications 41(13)(2014) 6047-6056.

[17] E. Hazir, T. Ozcan, K. Huseyin, Prediction of adhesion strength using extreme learning machine and support vector regression optimized with genetic algorithm, Arabian Journal For Science and Engineering 45(8)(2020) 1-20.

[18] R. Storn, K. Price, Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces, Journal of Global Optimization 11(4)(1997) 341-359.

[19] G. Onwubolu, D. Davendra, Scheduling flow shops using differential evolution algorithm, European Journal of Operational Research 171(2)(2006) 674-692.

[20] I. De Falco, A. Della Cioppa, D. Maisto, E. Tarantino, Differential evolution as a viable tool for satellite image registration, Applied Soft Computing 8(4)(2005) 1453-1462.

[21] I. Boussaied, A. Chatterjee, P. Siarry, M. Ahmed-Nacer, Two-stage update biogeography-based optimization using differential evolution algorithm, Computer & Operations Research 38(8)(2011) 1188-11198.

[22] V.N. Vapnik, An overview of statistical learning theory, IEEE transactions on neural networks 10(5)(1999) 988-999.

[23] V. Vapnik, S. Golowich, A. Smola, Support vector method for function approximation, regression estimation, and signal processing, Advances in Neural Information Processing Systems 9(1996) 281-287.

[24] C.J. Yin, Titanium cutting force prediction model and optimization of processing parameters, Master's thesis, Tianjin University of Technology, 2014.

[25] Y.L. Wang, Y.H. Hu, J.B. Sun, T.Q. Wang, Research on cutting force of high speed cutting titanium alloy based on orthogonal experiment, Tool Engineering 50(2)(2016) 17-19.