

## A New Approach to Cutting Temperature Prediction Using Support Vector Regression and Ant Colony Optimization

Zeng Dehuai<sup>1,3</sup>, Liu Yuan<sup>1</sup>, Jiang Lianbo<sup>1</sup>, Li Li<sup>1</sup>, Xu Gang<sup>1,2,\*</sup>

<sup>1</sup> College of Mechatronics and Control Engineering, Shenzhen University, Shenzhen, 518060

<sup>2</sup> Key Laboratory of Advanced Optical Precision Manufacturing Technology of Guangdong Higher Education Institutes, Shenzhen University, Shenzhen, 518060, China

<sup>3</sup> School of mechanical and automotive engineering, South China University of Technology, Guangzhou, 510640

\*Corresponding author email: xgrobot@126.com

**Keywords:** Temperature prediction, Support vector regression, Ant colony algorithm, Machining

**Abstract:** In this paper, support vector regression with ant colony optimization is presented for the prediction of tool-chip interface temperature depends on cutting parameters in machining. Ant colony (ACO) optimization was developed to optimize three parameters of SVR, including penalty parameter  $C$ , insensitive loss function  $\varepsilon$  and kernel function  $\sigma$ . SVR constructs hyperplane in high dimension space and fits the data in non-linear form. Normalized mean square error (NMSE) of fitting result is used as target of ant colony optimization. ACO finds the best parameters which correspond to the NMSE. The results showed that the proposed approach, by comparing with back-propagation neural network model, was an efficient way to model tool-chip interface temperature with good predictive accuracy.

### Introduction

Temperature rise in machining, especially along the tool-chip interface, is one of the major concerns and the main limitation in the selection of process parameters. It not only induces tool wear and limited the productivity, but also affects the cutting force and residual stress [1-2] on the machined surface. Hence it is crucial to predict the cutting temperature accurately.

The chip-tool interface temperature occurred in metal cutting as a result of heat directly influencing the tool wear behaviour of the cutting tool, therefore, temperature is of fundamental importance in metal cutting operations. Several experimental and analytical techniques have been developed for the measurement of temperatures generated in cutting processes. Due to the nature of metal cutting, it is not possible to measure temperature precisely in the cutting zone and thus it is difficult to verify the theoretical results in a precise manner. Because of nature of the metal cutting, determination of internal temperatures on the cutting tool are very difficult.

Many different analytical temperature models for confirming and reducing on experimental studies of machining are available. As the machining process is nonlinear and time dependent, it is difficult for the traditional identification methods to provide an accurate model. Compared with traditional computing methods, the artificial neural networks have been applied as an effective and an alternative method for the experimental studies that the mathematical model cannot be formed in recently. In the field of cutting, the model of the chip break [3], surface roughness [4] and tool wear[5] based on the neural network have been successfully established. ANNs-based models seem to obtain improved and acceptable performance in cutting operation forecasting issue, however, the conventional ANNs still suffer from several weaknesses such as the need for a large number of controlling parameters, the difficulty in obtaining stable solutions, the danger of over-fitting and thus the lack of generalization capability.

In order to overcome the shortcoming, support vector machines (SVMs) proposed by Vapnik [6] has been receiving increased attention with remarkable results. Established on the unique theory of the structural risk minimization principle, SVM is resistant to over-fitting problem and shown to avoid the above limits of ANN. Originally, SVM has been developed to solve pattern recognition and classification problems. However, with the introduction of  $\varepsilon$ -insensitive loss function, SVM has

been extended to solve nonlinear regression estimation problems, such as new techniques known as support vector regression (SVR), which have been shown to exhibit excellent performance. Thus, SVR had been successfully employed to solve forecasting problems in many fields. Such as traffic flow time series forecasting, engineering and software field forecasting [7], and so on. The practical results indicated that poor forecasting accuracy is suffered from the lack of knowledge of the selection of the three parameters ( $\sigma$ ,  $C$ , and  $\varepsilon$ ) in a SVR model.

Due to lacking of structured ways in determining three free parameters in SVR, in this investigation, SVR model, whose control parameters were optimized by the ant colony optimization (ACO), was developed to predict the building cooling load. Subsequently, SVR model was built using building cooling load historic data as the training samples.

The remainder of this paper is organized as follows. In Sections 2 and 3, we explain the methodology. Section 4 gives experimental results. Finally, Section 5 concludes the paper.

## Theoretical background

### A. Regression arithmetic of support vector machine

Using SVR is to map the input data  $x$  into a higher dimensional feature space through a nonlinear mapping  $F$  and then a linear regression problem is obtained and solved in this feature space. Here, given a set of data  $T = \{(x_1, y_1), \dots, (x_i, y_i), \dots, (x_l, y_l)\}$ , where  $x_i \in R^n$  is an arbitrary data point and  $y_i \in R$  denotes the corresponding output value, the SVM regression function can be formulated as

$$y = \omega \cdot \varphi(x) + b \quad (1)$$

where  $\varphi(\cdot)$  denotes the non-linear mapping function,  $\omega$  denotes the weight vector and  $b$  denotes the bias term.  $\varepsilon$ -SVR is to find a function  $y$  that has at most  $\varepsilon$  deviation from the actually obtained targets  $d_i$  for all the training data and at the same time is as flat as possible. We do not care about errors as long as they are less than  $\varepsilon$ , but will not accept any deviation larger than this. Flatness in this case means to reduce the model complexity, so that we can write this problem as a convex optimization problem:

$$\text{Min } R(y) = C \frac{1}{N} \sum_{i=1}^n L_\varepsilon(d_i, y_i) + \frac{1}{2} \|\omega\|^2 \quad (2)$$

It is known that the regression estimation function is the one that minimizes (2) with the following  $\varepsilon$ -insensitive loss function, defined as:

$$L_\varepsilon(d_i, y_i) = \begin{cases} |d_i - y_i| - \varepsilon, & |d_i - y_i| > \varepsilon \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

where both  $C$  and  $\varepsilon$  are user-determined parameters;  $d_i$  denotes the actual value at period  $i$ ;  $y_i$  represents the forecasting value at period  $i$ . Additionally, the first term in (2) denotes the empirical error. The second term in (2) represents the function flatness. The  $C$  is used as the trade-off between the empirical risk and the model flatness. Sometimes, this may not be the case, or we also may want to allow for some errors. Two positive slack variables  $\xi_i$  and  $\xi_i^*$  are introduced to represent the distance from actual values to the corresponding boundary values of the  $\varepsilon$ -tube. Then, Eq.(2) is transformed into the following constrained form:

$$\begin{aligned} \text{Min } S &= \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*) \\ \text{s.t. } &\begin{cases} d_i - \omega \cdot \varphi(x_i) - b \leq \varepsilon + \xi_i, \\ \omega \cdot \varphi(x_i) + b - d_i \leq \varepsilon + \xi_i^*, \\ \xi_i, \xi_i^* \geq 0, \end{cases} \end{aligned} \quad (4)$$

Introduction Lagrange multipliers  $\alpha_i, \alpha_i^*, \eta_i$  and  $\eta_i^*$ , we can write the corresponding primal Lagrangian as:

$$\begin{aligned} L(\omega, \xi_i, \xi_i^*) = & \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*) \\ & - \sum_{i=1}^n \alpha_i (\varepsilon + \xi_i + d_i - \omega \cdot \varphi(x_i) - b) \\ & - \sum_{i=1}^n \alpha_i^* (\varepsilon + \xi_i + \omega \cdot \varphi(x_i) + b - d_i) \\ & - \sum_{i=1}^n (\eta_i \xi_i + \eta_i^* \xi_i^*) \end{aligned} \quad (5)$$

Finally, by applying Karush-Kuhn-Tucker (KKT) conditions for regression, the optimization formulation can be transformed into a dual problem, get:

$$\begin{aligned} \text{Max } Q(\alpha_i, \alpha_i^*) = & \sum_{i=1}^n y_i (\alpha_i^* - \alpha_i) - \varepsilon \sum_{i=1}^n (\alpha_i^* + \alpha_i) - \\ & \frac{1}{2} \sum_{i,j=1}^n (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) K(x_i, x_j) \\ \text{s.t. } & \begin{cases} \sum_{i=1}^n (\alpha_i^* - \alpha_i) = 0, \\ 0 \leq \alpha_i, \alpha_i^* \leq C, i = 1, 2, \dots, n \end{cases} \end{aligned} \quad (6)$$

where  $\alpha_i, \alpha_i^*$  are the so-called Lagrangian multipliers. They satisfy the equalities  $\alpha_i \cdot \alpha_i^* = 0$ .

The Lagrange multipliers  $\alpha_i$  and  $\alpha_i^*$ , are calculated and an optimal weight vector of the regression hyperplane is expressed as:

$$w^* = \sum_{i=1}^{nsv} (\alpha_i - \alpha_i^*) K(x_i, x) \quad (7)$$

Hence, the regression function is expressed as

$$f(x) = \sum_{i=1}^{nsv} (\alpha_i - \alpha_i^*) K(x_i, x) + b \quad (8)$$

Where the constant b is written as

$$b = \frac{1}{2} \sum_{i=1}^{nsv} (\alpha_i - \alpha_i^*) (K(x_r, x_i) + K(x_s, x_i)) \quad (9)$$

Notably, a number of coefficients  $(\alpha_i - \alpha_i^*)$  are non-zero values and the corresponding training data points have approximation error equal to or larger than zero. They are called support vectors.  $x_r$  and  $x_s$  in Eq.(9) are support vectors.  $nsv$  denotes the number of the total support vectors.

The term  $K(x_i, x_j)$  in Eq.(8) is defined as kernel function, where the value of kernel function equals the inner product of two vectors  $x_i$  and  $x_j$  in the feature space  $\varphi(x_i)$  and  $\varphi(x_j)$ , meaning that  $K(x_i, x_j) = \varphi(x_i) \varphi(x_j)$ . The kernel function is intended to handle any dimension feature space without the need to calculate  $\varphi(x)$  accurately. If any function can satisfy the Mercer's conditions and performs the non-linear mapping. In SVM, radial basis function (RBF)  $K(x_i, x_j) = \exp(-\|x_i - x_j\|/2\sigma^2)$ , polynomial basis function  $K(x_i, x_j) = ((x_i \cdot x_j) + b)^d$  ( $b \geq 0$ ,  $d$  is the natural number), and sigmoid function  $K(x_i, x_j) = \tanh(k(x_i \cdot x_j) + v)$  are typical examples of kernel function. Only one variable needs to be determined in the radial basis function, In addition, SVM constructed by radial basis function has excellent nonlinear classification ability. Thus, in this work, RBF is used in the SVM.

The selection of the three parameters ( $\sigma$ ,  $C$ , and  $\varepsilon$ ) of a SVR model influence the accuracy of forecasting. However, structural methods for confirming efficient selection of parameters efficiently are lacking. Recently, several researchers have committed to parameters selection. Li et al adopted the genetic algorithm [8] and particle swarm optimization algorithm [9] to optimize the feature subset and model parameter for the SVM. Grid search method was used by Coussement et al.

Derivative-free optimization was employed to parameters optimization. More recently, artificial immunization algorithm was used to optimize the model parameters. However, as mentioned above that GA and SA are lack of knowledge memory functions, which leads to time consuming in the searching the suitable parameters of a SVR model. Therefore, ant colony optimization (ACO) is used in the proposed SVR model to optimize parameter selection instead of using the abovementioned evolutionary algorithm.

## B. Ant Colony Optimization Algorithm

Ant Colony Optimization (ACO) algorithm is a kind of algorithm inspired by real ants. The principle of the method is based on the way ants search for food and find their way back to the nest. During trips of ants a chemical trail called pheromone is left on the ground. The role of pheromone is to guide the other ants towards the target point. For one ant, the path is chosen according to the quantity of pheromone. Ants are capable of exploring and exploiting pheromone information, which have been left on the routes when they traversed. They choose routes according to the amount of pheromone. The larger amount of pheromone is left on a route, the greater is the probability of selecting the route by artificial ants. In ACO, artificial ants find solutions starting from a start node and moving to feasible neighbor nodes in the process of building the solutions. Pheromone evaporation is a process of decreasing the intensities of pheromone trails over time. This process is used to avoid locally convergence and to explore more search space.

Each ant builds a tour by repeatedly applying a stochastic greedy rule, which is called the state transition rule.

$$P(r, u) = \begin{cases} \arg \max_{u \in J(r)} \{ [\tau(r, u)]^\alpha \cdot [\eta(r, u)]^\beta \}, & \text{if } q \leq q_0 \\ S, & \text{otherwise} \end{cases} \quad (10)$$

$(r, u)$  represents an edge between point  $r$  and  $u$ , and  $\tau(r, u)$  stands for the pheromone on edge  $(r, u)$ .  $\eta(r, u)$  is the desirability of edge  $(r, u)$ , which is usually defined as the inverse of the length of edge  $(r, u)$ .  $q$  is a random number uniformly distributed in  $[0, 1]$ ,  $q_0$  is a user-defined parameter with  $0 \leq q_0 \leq 1$ ,  $b$  is the parameter controlling the relative importance of the desirability.  $J(r)$  is the set of edges available at decision point  $r$ .  $S$  is a random variable selected according to the probability distribution given below.

$$P(r, s) = \begin{cases} \frac{[\tau(r, u)]^\alpha \cdot [\eta(r, u)]^\beta}{\sum_{u \in J(r)} [\tau(r, u)]^\alpha \cdot [\eta(r, u)]^\beta}, & \text{if } s \in J(r) \\ 0, & \text{otherwise} \end{cases} \quad (11)$$

The selection strategy used above is also called ‘roulette wheel’ selection since its mechanism is a simulation of the operation of a roulette wheel. Every city has its percentage in the roulette wheel and the bigger this percentage is, the larger the width of slot in the wheel so that the probability of choosing that city becomes larger. After a random spinning of the wheel, which is performed by generating a random number, a slot is chosen and the next route the ant will go on is determined.

Once ants have completed their tours, the most pheromone deposited by ants on the visited paths is considered as the information regarding the best paths from the nest to the food sources. Therefore, the pheromone dynamic updating plays the main role in real ant colonies searching behaviors. Here we introduce two kinds of pheromone update strategies, called local updating rule and the global updating rule. While constructing its tour, an ant will modify the amount of the pheromone on the passed edges by applying the local updating rule

$$\tau(r, s) \leftarrow (1 - \rho) \cdot \tau(r, s) + \rho \tau_0 \quad (12)$$

Where  $\rho$  is the coefficient representing pheromone evaporation,  $0 < \rho < 1$ .

Once all ants have arrived at their destination, the amount of pheromone on the edge is modified again by applying the global updating rule

$$\tau(r, s) \leftarrow (1 - \delta) \cdot \tau(r, s) + \delta \cdot \Delta \tau(r, s) \quad (13)$$

Where

$$\Delta\tau(r,s)=\begin{cases} 1/L_{gb}, & \text{if } (r,s) \in \text{global best tour} \\ 0, & \text{otherwise} \end{cases}$$

The  $\delta$  is the global pheromone decay parameter,  $0 < \delta < 1$ , the  $\Delta\tau(r,s)$  expressed as Eq., is used to increase the pheromone on the path of solution, and  $L_{gb}$  is the length of the globally best tour from the beginning of the trial. We can see that only the ant that finds the global best tour can achieve the pheromone increase.

## Cutting temperature prediction model based on SVR-ACO

### A. Problem formulation

According to the changes of time series of the cutting temperature, the current temperature is certainly linked with that of several minutes ago. Thus the previous tool-chip interface temperature sequence data can be used to predict the future cutting temperature. Assuming  $T_i(t)$  is the temperature value of sintered temperature in  $t$  moment,  $T_i(t-1)$  is the temperature value in  $t-1$  moment. The temperature value of sintered temperature of current and previous  $m$  time period can be used to forecast the tool-chip interface temperature value in the future time period. Let  $T_i(t)$ ,  $T_i(t-1)$ , ...,  $T_i(t-m)$  be the samples input vector in  $t$  moment as  $x_i$ ,  $T_i(t+1)$  be the samples output value  $y_i$ .

Chosen sintered temperature value samples as initial training set:  $S = \{s_i | s_i = (x_i, y_i), i = 1, 2, \dots, n\}$ . According to SVR algorithm, the initial prediction regression function is obtained. Then the prediction value for sintered furnace temperature  $\hat{y}_i$  is:

$$\hat{y}_i = \sum_{i=1}^l (\alpha_i - \alpha_i^*) K(x_i, x) + b \quad (14)$$

Our process of using ACO SVR hyperparameters optimization and SVM for cutting temperature forecasting is demonstrated in Fig. 1. To tailor this mechanism, it is necessary to use the dependency measure as the stopping criterion. This means that an ant will stop building its feature subset when the dependency of the subset reaches the maximum for the dataset. If the data of the first attribute subset cannot meet the required value, then it returns the best subsets to choose the data of another attribute subset, it does not stop until finding optimization function value. Based on the optimal value, SVR constructs fitting curve, and normal mean square error (NMSE) gotten by 5-fold cross-validation method is used as ants optimization objective, the NMSE for final optimized results is the smallest.

In this study, to discretize those continuous parameters, each digit of the parameters is represented by ten cities. Thus, each digit contains 10 possible values from 0 to 9. In this algorithm, the limits of kernel function  $\sigma$  with range (0~9.999); penalty parameters  $C$  with range (0~9999), loss function with range (0.000~9.999). The bit number in total is 12bits when these three parameters arranged in order, by which the ant passes in each cycle, there are 10 paths to be selected (0~9) from one node to another. Every optimization process is recorded in the path table, which is expressed in a one-dimension array.

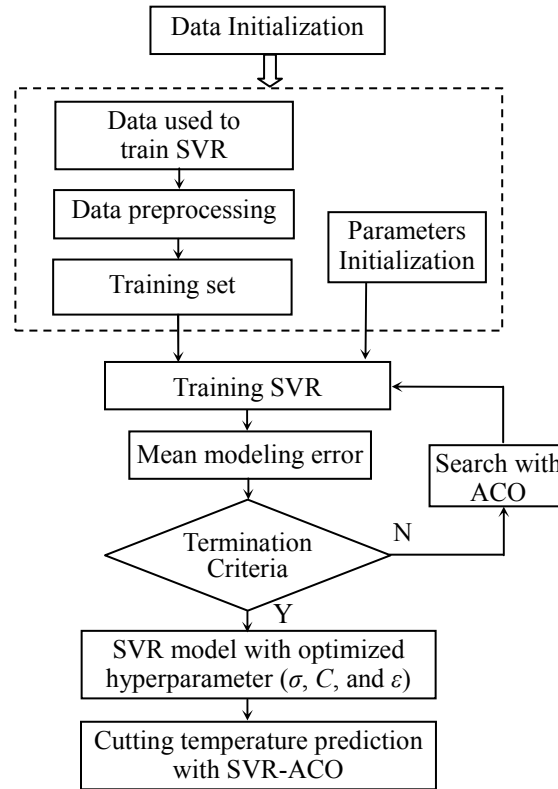


Fig.1 Flowchart of SVR-ACO algorithm

The basic steps to optimize SVR parameter with colony algorithm is described as following:

Step1: Design optimization objective, take normalized mean square error gotten by 5-fold cross-validation with each group parameters of SVR as performance index.

Step2: Set ant number  $n$ , and define a one-dimension array  $A_k$  including 12 elements for each ant  $k$  ( $k = 1, 2, \dots, n$ ), in which the ordinate value of 12 notes the ant  $k$  passed by is stored and represents the ant  $k$  moving path.

Step3: Set time counter  $t=0$ , cycle number  $N=0$ , the maximum cycling times  $\max N$ ,

Step4: Set circulation volume  $i = 1$

Step5: Calculate probability that ant moves to each path note with transition probability formula; and the note which ant moves forward to is calculated with roulette selection method. Move the ant to this note, and store the coordinate value to the element  $i$  of array  $A_k$ .

Step6: Set  $i = i + 1$ , if  $i \leq 10$ , jump to step 5, else step 7.

Step7: According to the path ant  $k$  ( $k = 1 \sim m$ ) passed by, i.e. array  $A_k$ , SVR parameter is calculated, which is used in support vector regression and calculate the normalized mean square error of fitting data.

Step8: Update pheromone, reset parameter  $t$  and  $N$ , clear array  $A_k$ .

Step9: If  $N_c < N_{\max}$  and the whole ant colony doesn't converge to the same path, reset the ant colony to the initial point O, jump to step 4; if  $N_c < N_{\max}$ , but the whole ant colony converge to the same path, then the optimized parameter is achieved.

## Experiments and results

### A. Experiment setup

The test system of drilling temperature is made up of PC computer, data acquisition cards, amplifier and thermocouple closed-loop and so on. A mineral isolated K type thermocouple on the 1.5 mm and length 50 mm was used for temperature measurement. The system temperature signal is measured through the semi-artificial thermocouple. Firstly, the temperature signal is converted into

electrical signal. Then the weak electrical is amplified by signal amplification, is connected to an Advantech PCI 1710 HG data acquisition and is inputted into the computer. The measurements data is obtained through software analyze finally.

### B. Data set and preprocessing

To evaluate the prediction applicability of cutting temperature with SVR-ACO, some common baseline cutting temperature prediction methods are exploited for performance comparison. Data from the tool-chip interface temperature are divided into two parts and these parts constitute training and testing data. The experimental data set consists of 50 values. The 44 values are used for training of SVR with ACO whereas, the 6 values are performed for testing of the SVR-ACO prediction model performance.

Subject to random factors (e.g. transmission errors, etc.), it can not be avoided to lose data accuracy such as data errors and data loss, so the primary data preprocessing need to be implemented to correction errors. The training data is smoothly processed to eliminate singular value, and the experimental data (including training data and testing data) is normalized, which can improve generalization capability of SVR. In order to avoid the influence of difference between factors, the parameters of input and output are normalized as Eq.(15), the learning data are scaled into [0, 1]:

$$\bar{y}_i = \frac{(y_i - y_{\min})}{y_{\max} - y_{\min}} \quad (15)$$

For the SVR-ACO model, a rolling-based forecasting procedure was conducted and a one-minute-ahead forecasting policy adopted. Then, several types of data-rolling are considered to forecast cutting temperature in the next minute. In this investigation, the ACO is employed to determine suitable combination of the three parameters in a SVR model. Parameters of the SVR-ACO models with the minimum testing NMSE values were selected as the most suitable model for this investigation. The SVR forecasting model was programmed by using mySVM software kit. The operating environment is: CPU Pentium 2.4 GHz, Memory 2G, MS WinXP operation system.

### C. Results

Table 1 indicates that SVR-ACO models perform the best when 20 input data are used for cutting temperature prediction respectively.

Tab.1 SVR-ACO optimal forecasting parameters

Num.of input data	Hyper-parameters			NMSE of testing
	$\sigma$	C	$\epsilon$	
5	0.6126	1196	0.4228	0.3617
10	0.6742	1703	0.4614	0.3353
15	0.6258	2317	0.9014	0.2565
<b>20</b>	<b>0.6541</b>	<b>2315</b>	<b>0.9321</b>	<b>0.2387</b>
25	0.6215	4125	0.5475	0.2839

Here, the ANN compared with SVR-ACO is back propagation neural network (BPNN) model. Comparison of the forecasting results among SVR-ACO, SVR and BPNN are shown in Table 2. It indicates that SVR-ACO has more excellent performance than SVR, BPNN in forecasting hourly cooling load.

Tab.2 Performance comparison of different prediction model

No.	MAPE(%)		
	SVR-ACO	SVR	BPNN
1	4.6	5.0	5.4
2	3.9	4.3	4.5
3	5.5	6.0	6.3
4	7.3	8.9	9.3
5	6.4	7.0	7.2
6	5.4	5.7	6.0
7	6.1	7.5	7.8
8	4.9	5.2	5.6
Mean	5.2	6.2	6.5

## Conclusions

This study applied SVR to the forecasting tool-chip interface temperature. To build stable and reliable forecasting models, the parameters of SVR must be specified carefully. Generally, ACO can be used to select suitable parameters to cutting temperature, which avoids over-fitting or under-fitting of the SVM model occurring because of the improper determining of these parameters. In this study, ACO is used to optimize the three parameters of the support vector regression. Experimental results demonstrate the feasibility of successfully applying this novel hybrid SVR-ACO model to the complex forecasting problem. Moreover, the experimental results also suggest that within the forecasting fields of tool-chip interface temperature, the SVR-ACO is typically a reliable forecasting tool, with the forecasting capacity more precise than that of SVR and BPNN model.

## Acknowledgments

This work is partly supported by the National Natural Science Foundation of China (50905119); the Natural Science Foundation of Guangdong Province (941806001001985); China Postdoctoral Science Foundation special fund (201104354); Shenzhen Science & Technology Program (SYB200907090130A / JSA200903190981A) and the SZU R/D Fund (2010004).

## References

- [1] Lazoglu I, Altintas Y. Prediction of tool and chip temperature in continuous and interrupted machining. *Int Jnl of Mach Tools Manu.* 2002, 42(9):1011-1022
- [2] Henriksen EK. Residual stresses in machined surfaces. *Trans of ASME.* 1951. 73(1): 69-76.
- [3] H.G.Kim, J.H.Sim, and H.J.Kweon, Performance evaluation of chip breaker utilizing neural network, *J Mater Process Technol*, vol. 209, pp.647-656, Fed 2009.
- [4] D. Karayel, Prediction and control of surface roughness in CNC lathe using artificial neural network, *J Mater Process Technol*, vol. 209, pp. 3125-3137, Fed 2009
- [5] S.S.Panda, D.Chakraborty, and S.K.Pal, Flank wear prediction in drilling using back propagation neural network and radial basis function network, *Applied soft computing*, vol. 8, Jul. 2008, pp. 858-871
- [6] Corinna Cortes, Vladimir Vapnik, Support-Vector Networks, *Machine Learning*, 20, pp.273-297 (1995)
- [7] Pai, P.F., Lin, C.S., Hong, W.C., Chen, C.T.: A hybrid support vector machine regression for exchange rate prediction. *International Journal of Information and Management Sciences* 17, 19–32 (2006)
- [8] Li Xuemei, Ding Lixing, Hybrid Genetic Algorithm and Support Vector Regression in Cooling Load Prediction, *Proc. of 3rd Int. Conf.on Knowledge Discovery and Data Mining*, 2010, pp527-531
- [9] Li Xuemei, Shao Ming, et al. Particle Swarm Optimization-based LS-SVM for Building Cooling Load Prediction, *Journal of Computers*, Vol 5, No 4 (2010), 614-621