Commun. Theor. Phys. (Beijing, China) **45** (2006) pp. 641–646

Vol. 45, No. 4, April 15, 2006

Nonlinear Time Series Prediction Using LS-SVM with Chaotic Mutation Evolutionary Programming for Parameter Optimization*

XU Rui-Rui, CHEN Tian-Lun, and GAO Cheng-Feng

Department of Physics, Nankai University, Tianjin 300071, China

(Received June 28, 2005)

Abstract *Nonlinear time series prediction is studied by using an improved least squares support vector machine (LS-SVM) regression based on chaotic mutation evolutionary programming (CMEP) approach for parameter optimization. We analyze how the prediction error varies with different parameters (σ, γ) in LS-SVM. In order to select appropriate parameters for the prediction model, we employ CMEP algorithm. Finally, Nasdaq stock data are predicted by using this LS-SVM regression based on CMEP, and satisfactory results are obtained.*

PACS numbers: 05.45.-a
Key words: nonlinear time series prediction, least squares support vector machine, chaotic mutation evolutionary programming

1 Introduction

Nonlinear time series almost appear in all the fields of natural science and social science such as physics, meteorology, economics, national defense, and so on. Previously, various neural network prediction models, which are based on phase space reconstruction theory, have been used to predict these significant series.[1,2] Support vector machine (SVM) on the basis of statistical learning theory and the principle of structural risk minimization, was first proposed by Vapnik *et al.*, which has been successfully applied to classification and regression tasks for its excellent learning ability with fewer training data.[3,4] Later, many modified SVM methods by optimization techniques have been brought forward. LS-SVM is an efficient version of these improved ones. Instead of a quadratic programming (QP) problem in standard SVM, a set of linear equations based on KKT optimization condition[5] are solved in LS-SVM, which can reduce the computational complexity and time for training to a certain extent. LS-SVM with empirical parameters ($\sigma = 1.7$, $\gamma = 10^7$) has successfully been used to predict non-noisy nonlinear time series produced by typical nonlinear dynamic equation.[6] However, its capability of filtering noise needs to be advanced further. To pursue smaller errors with less predicting time, the effects of parameters in LS-SVM were studied detailedly.

In the present work, we first investigate the prediction errors with different parameters of LS-SVM. Then the methods to select appropriate parameters of network are discussed, and CMEP algorithm is introduced in LS-SVM to select parameters of network. At last the improved LS-SVM regression, namely LS-SVM based on CMEP approach, is utilized to predict Nasdaq stock data to test its prediction capability in actual work.

2 Prediction Model

Suppose a training set $\{x_i, y_i\}_{i=1}^N$ with input pattern $x_i \in R^n$ for the i-th example and $y_i \in R$ for the corresponding desired output pattern, which is produced by an unknown nonlinear dynamic equation, and in the feature space the assumption of LS-SVM can be described as

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

where the nonlinear mapping $\phi(x)$ can map the input data into a higher-dimensional feature space. Moreover, the coefficients (w and b) should satisfy the minimization of objection function (2) and an equality constraint (3):

$$\min_{w,e} J(w,e) = \frac{1}{2}w^{\mathrm{T}}w + \frac{1}{2}\gamma\sum_{i=1}^{N} e_i^2\,, \tag{2}$$

S.T.

$$y_i = w^{\mathrm{T}}\phi(x_i) + b + e_i \quad \text{for} \quad i = 1, \ldots, N\,, \tag{3}$$

where e_i is the error. Formulas (2) and (3) can be transformed to a set of linear equations (4) based on KKT optimization condition:

$$\begin{pmatrix} 1 & 1_\nu^{\mathrm{T}} \\ 1_\nu & \Omega + \dfrac{1}{\gamma} \end{pmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix}, \tag{4}$$

where $y = [y_1; \ldots; y_N]$, $1_\nu = [1; \ldots; 1]$, $\alpha = [\alpha_1; \ldots; \alpha_N]$, $\Omega_{ij} = \phi(x_i)^{\mathrm{T}}\phi(x_j) = K(x_i, x_j)$ for $i, j = 1, 2, \ldots, N$, and $K(x_i, x_j)$ is the kernel function that satisfies the case of Mercer's condition. Thus we can obtain LS-SVM prediction model:

$$y(x) = \sum_{i=1}^{N} \alpha_i K(x, x_i) + b\,, \tag{5}$$

*The project supported by National Natural Science Foundation of China under Grant No. 90203008 and the Doctoral Foundation of the Ministry of Education of China

where x, x_i denote training point and support vector respectively, y is the output of network, α_i and b are the solutions to Eq. (4). In all the simulations, the kernel is taken as the radial-basis function:

$$K(x_i, x_j) = \exp\Big(-\frac{|x_i - x_j|^2}{2\sigma^2}\Big) . \tag{6}$$

There exist two parameters (γ and σ) selected by the user in advance. γ is viewed as a regularization parameter, which controls the tradeoff between complexity of the machine and the number of non-separable points.[7] The kernel parameter σ denotes the width of Gauss function.

The prediction with different parameters can be analyzed via predicting noisy nonlinear time series produced by Mackey–Glass equation, which is a time-delayed differential equation first proposed as a model of white blood cell production,[8]

$$x_t = \frac{\alpha x_{t-s}}{1 + (x_{t-s})^{10}} + (1 - \eta) x_{t-s} , \tag{7}$$

where $\alpha = 0.2$, $\eta = 0.1$, and s is an adjustable parameter. When $s \geq 17$, equation (7) exhibits chaotic behavior with a fractal dimension. The parameter s is taken as 17 in present calculation. The prediction based on nonlinear time series produced by Mackey–Glass equation is regarded as a criterion for comparing the ability of different predicting methods. In addition, the noisy time series can be expressed as[9]

$$z'_t = z_t + \upsilon_t , \tag{8}$$

where $\upsilon_t = \beta k \mu$, β denotes an adjustable parameter that controls the degree of noise, μ is uniformly distributed in region $[-1, 1]$, k stands for the signal-to-noise ratio, namely the standard deviation of the Mackey–Glass time series divided by that of noise component.

3 Analysis of Parameters and Prediction Capability of LS-SVM

We first study how the parameters (γ and σ) affect the results of predicting noisy data with $\beta = 0.4$ produced by Mackey–Glass equation. The computation is performed in the two following ranges of parameter values:

(i) $\sigma \in (0.1, 10)$, $\gamma \in (10^5, 10^8)$;

(ii) $\sigma \in (0.1, 10)$, $\gamma \in (10^3, 10^5)$.

It should be noted that the empirical parameters ($\sigma = 1.7$, $\gamma = 10^7$) are included in range (i). The root-mean-square error (RMSE) and coincidence probability p are taken to evaluate the performance of the LS-SVM regression. The first criterion (RMSE) is calculated by

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x'_i - x_i)^2} , \tag{9}$$

where x'_i and x_i stand for the predicted and desired values respectively, and N denotes the total number for prediction. The second one (p), which shows the coincidence degree between the predicted and desired values, is evaluated as

$$p = \frac{N_c}{N} , \tag{10}$$

where N_c represents the number of points whose errors are less than certain value E.

In this case, we take the initial 20 data for training to construct the model, and then the following 30 (N) data are used to test the prediction capability of the model, and the embedded dimension is set to 7. The simulation results are shown in Figs. 1 $\sim$ 3.

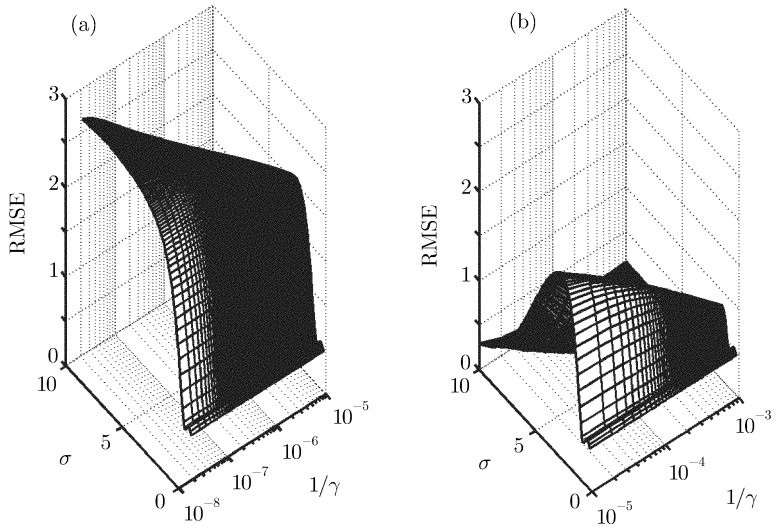

Fig. 1 RMSE in different parameter ranges. (a) For range (i); (b) For range (ii).

It can clearly be seen in Fig. 1 that RMSE significantly varies with γ and σ using LS-SVM. In addition, RMSE in

Fig. 1(b) is less than that in Fig. 1 (a) generally. Therefore, in the two parameter ranges we prefer choosing the proper parameters for LS-SVM in range (ii) when these noisy data are considered.

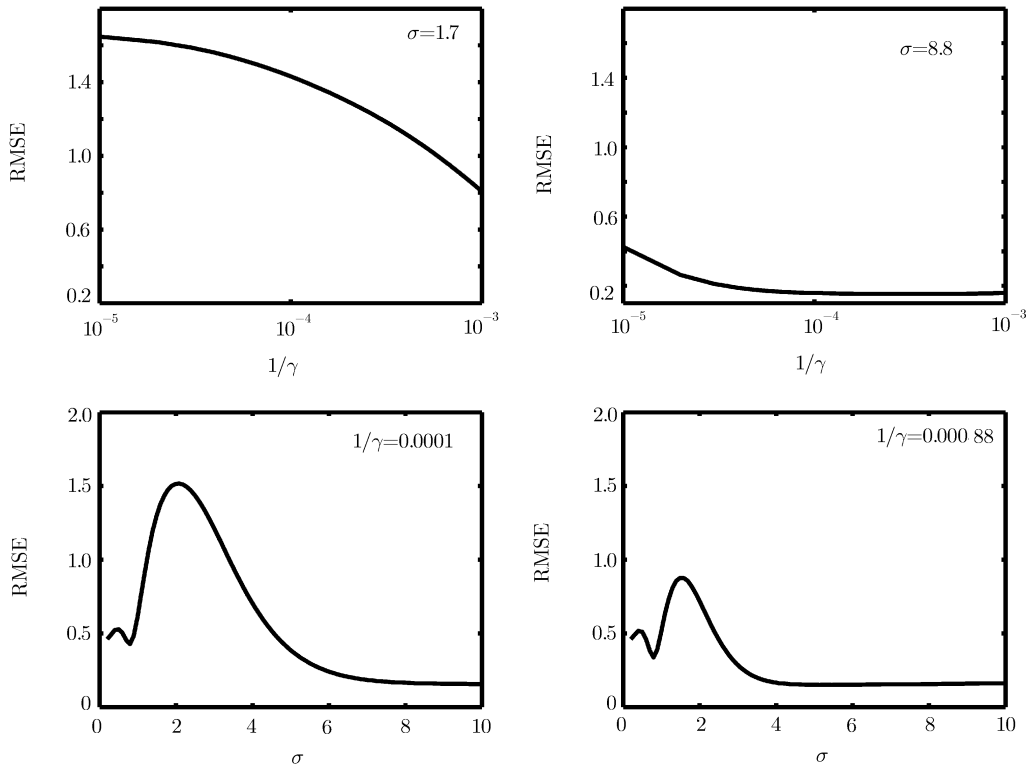

Fig. 2 Sections of Fig. 1(b).

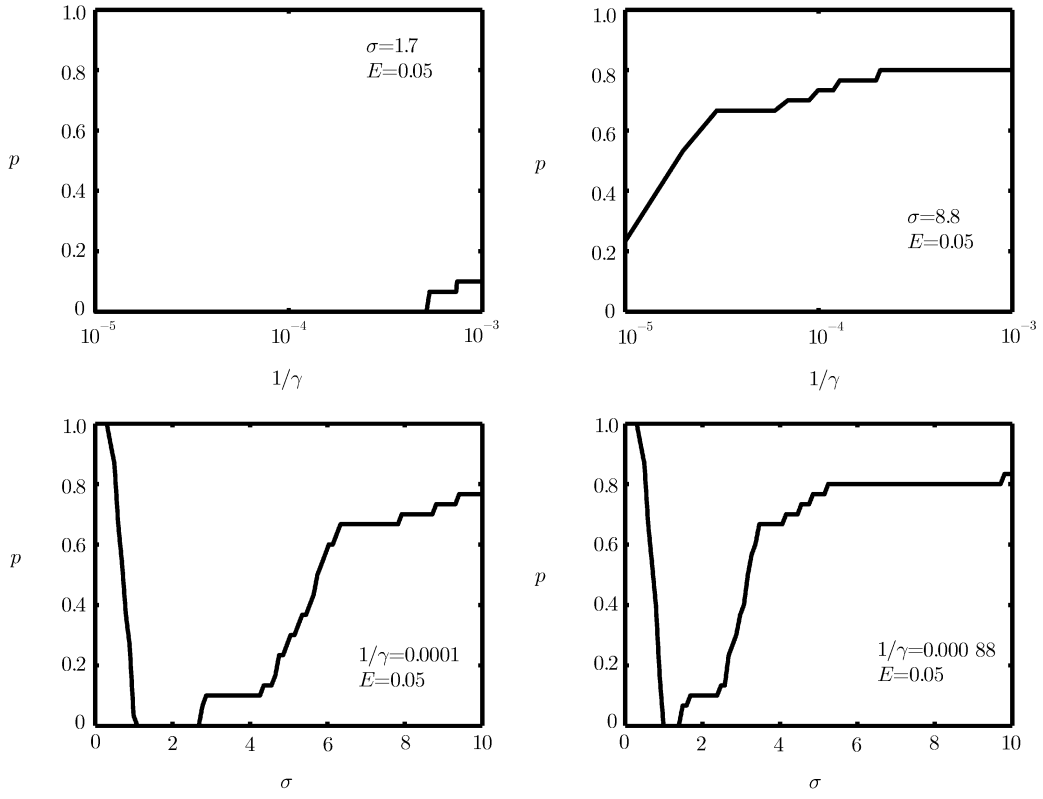

Fig. 3 The variation of p in the range (ii).

Then, we take four sections in Fig. 1(b) to analyze the effects of γ and σ. It is indicated in Fig. 2 that RMSE monotonously decreases with increasing $1/\gamma$, while varies with σ more complicatedly. The difference mainly attributes to the different effects of parameters to LS-SVM regression, namely γ only influences the training step in the whole process, while σ affects both the training step and the predicting one.

In the next simulation, we explore the coincidence probability of single point in range (ii) with $E = 0.05$. By contrast to the variations of RMSE in Fig. 2, the larger coincidence probability is obtained where the less RMSE exists, as shown in Fig. 3. It further illustrates that the better prediction results can be obtained by selecting parameters for LS-SVM regression in range (ii).

4 Chaotic Mutation Evolutionary Programming Approach for Parameter Optimization

Obviously, it plays an important role to select appropriate parameters in the whole process of prediction using LS-SVM. Motivated by pursuing smaller error, we have discussed the methods of selecting parameters. There have existed several effective optimization approaches for LS-SVM, including the method based on genetic algorithm (GA)[10] and multi-layer adaptive parameters optimization (MAPO) approach,[11] etc. However, they have some disadvantages, such as costing longer training time, easily destroying the searched results for crossover operator existing in GA, and easily trapping into those parameters with local minimization using MAPO. Therefore, we employ chaotic mutation evolutionary programming (CMEP) optimization algorithm[12,13] to select the LS-SVM's parameters, which mainly contains 4 steps:

(i) Initialization: take m sets of $(\sigma_i, \lambda_\sigma(j), \gamma_i, \lambda_\gamma(j))$, $i = 1, 2, \ldots, m$, as the initial parents, where m is the scale of the population. The initial evolution generation j is set to 1. σ_i and γ_i are selected randomly in range (ii), and $\lambda_\sigma(j)$, $\lambda_\gamma(j)$ represent the mutation step-sizes of σ_i and γ_i in the j-th generation, respectively. Then compute the fitness value of each parent: $f = 1/\mathrm{RMSE}$.

(ii) Mutation: m individuals of parents can be transformed into m individuals of children by using chaotic mutation operator, which is defined as

$$\begin{aligned} &\sigma_i'(j) = \sigma_i(j) + \lambda_{\sigma i}(j) Q_{\sigma j}\,, \\ &\gamma_i'(j) = \gamma_i(j) + \lambda_{\gamma i}(j) Q_{\gamma j}\,, \\ &i = 1, 2, \ldots, m; \quad j = 1, 2, \ldots, N_{\max} - 1\,. \end{aligned} \tag{11}$$

$N_{\max}$ is the maximum of evolution generation. The random number Q_j can be computed by $Q_j = a + bg(q_j)$, where g serves as Logistic equation:

$$u_{n+1} = g(u_n) = \mu u_n (1 - u_n) \quad u_n \in (0, 1) \tag{12}$$

with $\mu = 4.0$. For different parameters, Q_j is restricted within a given range because of the coefficients a and b. The evolution steps λ_σ and λ_γ are very crucial factors to dominate the scope of evolution efficiently, which are calculated by

$$\begin{aligned} &\lambda_\sigma(j) = \lambda_\sigma(1)\Big[\nu \exp\Big(-\frac{j}{N_{\max}}\Big) + \theta\Big]\,, \\ &\lambda_\gamma(j) = \lambda_\gamma(1)\Big[\nu \exp\Big(-\frac{j}{N_{\max}}\Big) + \theta\Big]\,, \\ &j = 2, 3, \ldots, N_{\max}\,, \end{aligned} \tag{13}$$

where ν and θ control the variety degrees of λ_σ and λ_γ with increasing j. For example, in all the simulations we take $\nu = 1.54$ and $\theta = 0.52$ for both σ and γ, which can keep $\lambda(j)$ decreasing within the range from 95% to 5% of $\lambda(1)$ with increasing j. In addition, we empirically choose $\lambda_\sigma(1) = 3$ and $\lambda_\gamma(1) = 180$. Finally, recalculate the fitness values of the children.

(iii) Selection: generate a new group G by integrating these children into the parents, and produce a q-dimensional group via randomly taking q elements from group G. Then compute the success number S_k, namely the number of some q-dimensional group elements whose fitness values are smaller than the fitness value of the k-th element in group G, $k = 1, 2 \ldots, 2m$. Sort the 2 m elements of group G from large to small by the success number S_k, and m individuals with larger success numbers are taken as the new parents of the next generation.

(iv) Estimate the evolution object. If satisfied, quit. Otherwise, set $j = j + 1$, then skip to step (ii).

We compare the evolution outputs of CMEP with those of MAPO by predicting the noisy data of Mackey–Glass equation with $\beta = 0.4$. The basic conditions of the simulation are similar to those in Fig. 1. Moreover, in CMEP the initial parents are selected in range (ii), and the random numbers are restricted within $|Q_{\sigma j}| \leq 1.0 \times 10^{-1}$ and $|Q_{\gamma j}| \leq 1.0 \times 10^{-5}$, and q is set to 60. In MAPO, we respectively choose 100 and 2500 as the net grid number in two simulations, and $N_{\max}$ is taken as 20. The results are shown in Table 1 and Fig. 4, and in Fig. 4, the RMSE of CMPE approach is the average result of 10 times simulations.

Table 1 Optimization time of two approaches for 20 generation steps.

Model	100 individuals in CMEP	100 net grids in MAPO	2500 net grids in MAPO
Time (s)	1200	700	10667

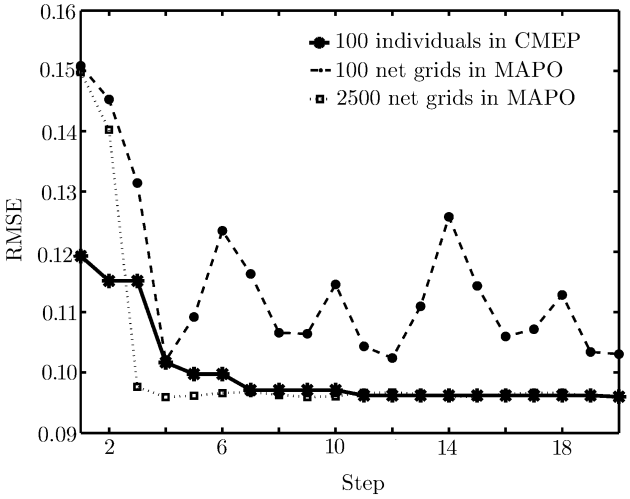

Fig. 4 Parameter optimization results with different approaches.

The results indicate that CMEP selects the appropriate parameters of LS-SVM more efficiently than MAPO does. Firstly, as shown in Table 1 and Fig. 4, MAPO method with 2500 net grids searches smaller RMSE than the same method with 100 net grids does, however, the time with 2500 net grids is longer than that with 100 ones by a factor of 15, which is not suit for actual prediction task. By adjusting the parameters λ_σ and λ_γ, CMEP method can obtain such good results as MAPO approach with 2500 net grids does using the time similar to what MAPO approach with 100 net grids needs, thus saves training time significantly. Furthermore, the RMSE of MAPO fluctuates distinctly over the whole step range, especially to the method with 100 net grids, while the one of CMEP monotonously decreases with the increasing step. It is mainly because MAPO always focuses on one of the parameter pairs whose RMSE is not near the smallest one, which easily induces algorithm to local minimization. In addition, MAPO cannot preserve the best result of each generation. As for CMEP algorithm, better parameters can be searched accurately and rapidly for the excellent performance of EP in global optimization and the ergodicity and randomicity in chaos respectively. Moreover, CMEP can preserve the best results of each generation, so its RMSE varies monotonously and more steadily. Therefore, CMEP approach is preferred to optimize the parameters in LS-SVM for prediction.

5 Stock Data Prediction

Many studies have shown that stock time series, which contain much noise, behave chaotically in a reconstructed phase space,[14] and it is a trend to analyze the fluctuation of stock data in nonlinear way. What is the most difficulty to predict in stock market are the data that fluctuate greatly. In this paper we investigate the higher-noisy part of Nasdaq stock data (1984.10.11 — 2003.3.18 http://finance.yahoo.com), in a chaotic way by using the improved LS-SVM regression based on CMEP.

We take the data from the 3000^{th} to the 4000^{th} time steps for training to construct the model, and choose initial 50 and 100 ones as the support vectors respectively, and select the proper parameters using CMEP approach with the remaining data. The initial parents are selected in $\sigma \in (0.1, 10)$, $\gamma \in (10^3, 10^5)$, and the embedded dimension is set to 7. Then we test the prediction capability of this model by predicting the data from the 4000^{th} to the 4500^{th} time steps. E is taken as 0.02. The results are shown in Table 2 and Fig. 5.

Table 2 Stock data prediction with different numbers of support vectors.

N	RMSE	p
50	0.0243	67%
100	0.0170	82.6%

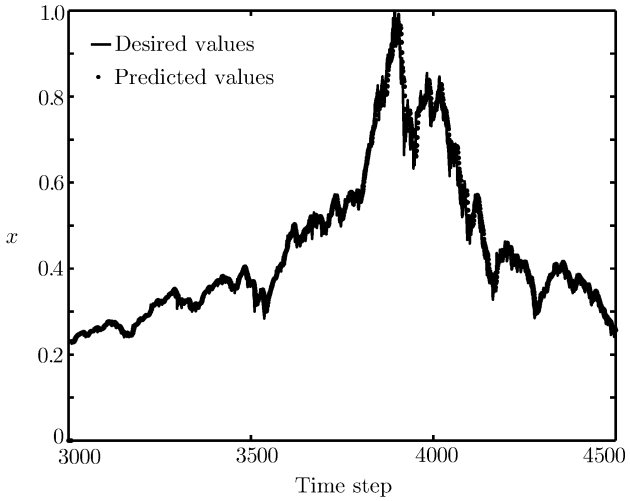

Fig. 5 Stock prediction with 50 support vectors.

It can clearly be seen that the predicted values with 50 support vectors are in good agreement with the desired ones in the whole range of time step covering at several sharp peaks, although the predicted results with 100 support vectors are better than those with 50 support vectors. Therefore, we can conclude that the improved LS-SVM regression exhibits excellent learning ability with fewer training data, and filters noise more efficiently. The generalization capability of LS-SVM is greatly improved.

6 Summary and Conclusion

We explore the influences of different parameters in LS-SVM used to predict the nonlinear time series. Considering the excellent performance of CMEP in searching global optimum, we apply CMEP approach to optimizing the parameters of LS-SVM regression. The simulation results indicate that CMEP can optimize the parameters

rapidly and accurately, and reduce the training time, and advance the capability of filtering noise simultaneously. Thus the improved LS-SVM regression based on CMEP can be applied generally. As an example, a set of Nasdaq data are successfully predicted by using this regression model.

In order to search the global optimum more efficiently, planned further work is to improve the mutation operator in CMEP and investigate how to choose the evolutions steps λ_σ and λ_γ to deal with different issues. CMEP can be applied to not only LS-SVM but also other modified versions of SVM. Its development will stimulate the application of SVM to more general fields.

References

[1] M. Small and C.K. Tse, Phys. Rev. E **66** (2002) 066701.

[2] K.P. Li and T.L. Chen, Chin. Phys. Lett. **19** (2002) 904.

[3] C. Cortes and V. Vapnik, Mach. Learn. **20** (1995) 273.

[4] A. Somla and B. Schölkopf, Stat. Comput. **14** (2004) 199.

[5] J.A.K. Suykens and J. Vandewalle, Neural Process. Lett. **9** (1999) 293.

[6] R.R. Xu, G.X. Bian, C.F. Gao, *et al.*, Commun. Theor. Phys. (Beijing, China) **43** (2005) 1056.

[7] H. Simon, *Neural Networks- A Comprehensive Foundation*, Tsinghua University Press, Beijing (2001).

[8] M. Mackey and L. Glass, Science **197** (1977) 287.

[9] K.P. Li, T.L. Chen, and Z.Y. Gao, Commun. Theor. Phys. (Beijing, China) **40** (2003) 311.

[10] S.K Dong, N.N. Ha, and S.P. Jong, "Genetic Algorithm to Improve SVM Based Network Intrusion Detection System", aina **2** (2005) 155 [C].

[11] J.Y. Zhu and Y. Yang, *et al.*, Control and Decision **19** (2004) 223.

[12] L.J. Yang and T.L. Chen, Commun. Theor. Phys. (Beijing, China) **38** (2002) 22.

[13] C.W. Jiang, W.M. Wang, *et al.*, Hydropower Automation and Dam Monitoring **28** (2004) 15.

[14] X.F. Zhu and S.Q. Gu, J. Management Sciences in China **3** (2000) 53.