Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2012, Article ID 985930, 12 pages
doi:10.1155/2012/985930

Research Article

Time Series Adaptive Online Prediction Method Combined with Modified LS-SVR and AGO

Guo Yangming,[1] Zhang Lu,[2] Cai Xiaobin,[3] Ran Congbao,[1] Zhai Zhengjun,[1] and Ma Jiezhong[1]

[1] *School of Computer Science and Technology, Northwestern Polytechnical University, Xi'an 710072, China*
[2] *School of Software and Microelectronics, Northwestern Polytechnical University, Xi'an 710072, China*
[3] *Science and Technology Commission, Aviation Industry Corporation of China, Beijing 100068, China*

Correspondence should be addressed to Guo Yangming, yangming_g@nwpu.edu.cn

Received 27 July 2012; Revised 19 November 2012; Accepted 29 November 2012

Academic Editor: Huaguang Zhang

Fault or health condition prediction of the complex systems has attracted more attention in recent years. The complex systems often show complex dynamic behavior and uncertainty, which makes it difficult to establish a precise physical model. Therefore, the time series of complex system is used to implement prediction in practice. Aiming at time series online prediction, we propose a new method to improve the prediction accuracy in this paper, which is based on the grey system theory and incremental learning algorithm. In this method, the accumulated generating operation (AGO) with the raw time series is taken to improve the data quality and regularity firstly; then the prediction is conducted by a modified LS-SVR model, which simplifies the calculation process with incremental learning; finally, the inverse accumulated generating operation (IAGO) is performed to get the prediction results. The results of the prediction experiments indicate preliminarily that the proposed scheme is an effective prediction approach for its good prediction precision and less computing time. The method will be useful in actual application.

1. Introduction

Prediction technique is essential to ensure the operational safety of complex systems. However, the complex systems are not easy to establish their precise physical models. In the practice, time series-based prediction methods have attracted increased attention [1–7].

Compared with other reported methods, Support Vector Regression (SVR) is based on statistical theory and structural risk minimization principle [8–10], and it has a global optimum and exhibits better accuracy in nonlinear and nonstationary time series data prediction via kernel function. However, aiming at the large sample data, the quadratic programming (QP) problem becomes more complex. Thus, Least Squares Support Vector Regression (LS-SVR) was proposed by Suykens et al. [11] In LS-SVR, the inequality constrains are replaced by equality constrains, which can reduce the calculation time effectively. Thus, LS-SVR has more attention in time series forecasting [12–16].

Because a great number of uncertain elements exist in practical application, online prediction scheme [17, 18] is applied to meet the actual training and prediction condition, and then achieve better prediction results. Moreover, the incompleteness, noisiness, and inconsistency existing in the raw time series sample data will reduce the prediction accuracy. In these cases, two problems should be considered simultaneously, one is how to improve prediction accuracy, the other one is how to reduce the training time and prediction time.

Some researchers combined several methods to improve prediction performance [19, 20], which can fuse their advantages and avoid their drawbacks. Using this idea, we propose a scheme to fix the above problems. In the new scheme, we integrate accumulated generating operation (AGO) [21] with a modified LS-SVR-based model. It includes the following two aspects.

(1) Based on the grey system theory [21], we conduct AGO with the raw time series to improve the data quality and regularity, and finally we can obtain the results using inverse accumulated generating operation (IAGO).

(2) We set up a new adaptive online prediction model based on LS-SVR, which utilizes incremental learning algorithm to enrich information and modifies the LS-SVR model with more simple form to reduce prediction calculation time.

The remainder of the paper is organized as follows. Section 2 gives a brief introduction of LS-SVR; Section 3 proposes the new scheme in detail, which includes the AGO-based prediction strategy and the simple modified prediction model based on incremental learning algorithm; Section 4 shows experiments and results analysis; conclusions are given in Section 5.

2. A Brief Introduction of LS-SVR

Consider a training sample data set $\{(x_k, y_k)\}_{k=1}^{N}$ with input data $x_k \in R^n$ and output $y_k \in R$, where N denotes the number of training samples. The goal of LS-SVR is to obtain a function as follows:

$$y(x) = w^T \phi(x) + b, \tag{2.1}$$

where the nonlinear mapping function $\phi(\cdot)$ maps the input data into a higher dimensional feature space. It means that the method makes the nonlinear fitting problem in input feature space to be replaced by a linear fitting problem in high-dimensional feature space. w is the weight vector and b is the bias constant.

According to the structure risk minimization principle, w and b can be found via a constrained convex optimization as follows:

$$\begin{aligned} \min \quad & J(w,e) = \frac{1}{2}w^T w + \frac{1}{2}c\sum_{k=1}^{N} e_k^2, \\ \text{s.t.} \quad & y_k = w^T \varphi(x_k) + b + e_k, \quad k = 1, 2, \ldots, N, \end{aligned} \tag{2.2}$$

where J is the loss function, $e_i \in R$ is the slack variables, and c is a regularization parameter. Equation (2.2) can be transformed into dual form with Lagrange function as follows:

$$L(w, b, e, \alpha) = J(w, e) - \sum_{k=1}^{N} \alpha_k \left\{ w^T \varphi(x_k) + b + e_k - y_k \right\}, \tag{2.3}$$

where α_k are the Lagrange multipliers.

It is obvious that the optimal solution of (2.2) satisfies Karush-Kuhn-Tucker (KKT) conditions, then the optimal conditions are shown as follows:

$$\begin{gathered} \frac{\partial L}{\partial w} = w - \sum_{i=1}^{n} \alpha_i \varphi(x_i) = 0 \Longrightarrow w = \sum_{i=1}^{n} \alpha_i \varphi(x_i), \\ \frac{\partial L}{\partial b} = -\sum_{i=1}^{n} \alpha_i = 0 \Longrightarrow \sum_{i=1}^{n} \alpha_i = 0, \\ \frac{\partial L}{\partial \alpha_i} = w^T \varphi(x_i) + b + e_i - y_i = 0 \Longrightarrow y_i = w^T \varphi(x_i) + b + e_i, \\ \frac{\partial L}{\partial e_i} = c e_i - \alpha_i = 0 \Longrightarrow e_i = \frac{1}{c} \alpha_i. \end{gathered} \tag{2.4}$$

After eliminating w and e_i from (2.4), we can obtain the solution by the following linear equations:

$$\begin{bmatrix} 0 & \mathbf{1}_n^T \\ \mathbf{1}_n & \mathbf{K} + \dfrac{\mathbf{I}}{c} \end{bmatrix} \begin{bmatrix} b \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{y} \end{bmatrix}, \tag{2.5}$$

where $\mathbf{K}(i, j) = k(x_i, x_j) = \varphi(x_i)^T \varphi(x_j)$, $\alpha = [\alpha_1, \alpha_2, \ldots, \alpha_n]^T$, $\mathbf{1}_n$ is an n-dimensional vector of all ones, $\mathbf{I}$ is a unite matrix, and $\mathbf{y} = [y_1, y_2, \ldots, y_n]^T$.

In LS-SVR, the optimization problem is simplified to a linear equations instead of more complex quadratic programming problem in SVR. Therefore, the computational complexity is decreased significantly. Obviously, (2.5) can be factorized into a positive definite system [22]. Thus, the solutions of α_k and b could be easily obtained. Then the LS-SVR model for function estimator can be expressed as follows:

$$y(x) = \sum_{k=1}^{N} \alpha_k K(x, x_k) + b. \tag{2.6}$$

3. Proposed Adaptive Online Prediction Scheme for Time Series

3.1. AGO-Based Prediction Method for Time Series

Time series prediction models employ the historical data values for extrapolation to obtain the future values. If a time series has a regular pattern, then a value of the series should be a function of previous values. If Y is the target value that we are trying to model and predict, and Y_t is the value of Y at time t, then the goal is to create a prediction model f as following form:

$$Y_t = f(Y_{t-1}, Y_{t-2}, Y_{t-3}, \ldots, Y_{t-n}) + e_t, \tag{3.1}$$

where Y_{t-1} is the value of Y for the previous observation, Y_{t-2} is the value two observations ago, and e_t represents noise that does not follow a predictable pattern. This way, the good prediction accuracy always depends on the high quality of time series data.

In 1982, a famous Chinese scholar Deng [21] proposed the grey system theory, which has already been used widely in all kinds of fields. As the core of the grey prediction theory, the accumulated generating operation (AGO) and inverse accumulated generating operation (IAGO) are the main methods which provide a manageable approach to treating disorganized evidence, that is, AGO has a main advantage that it can reduce the disturbance with the stochastic factors. AGO makes the rules hidden the raw time series to be presented fully and enhances the law of time series data.

Consider a raw time series $x^{(0)} = (x^{(0)}(1), x^{(0)}(2), ..., x^{(0)}(n))$, and take the AGO with $x^{(0)}$ as follows:

$$x^{(1)}(i) = \sum_{r=1}^{i} x(r), \quad i = 1, 2, \ldots, n. \tag{3.2}$$

Then $x^{(1)} = (x^{(1)}(1), x^{(1)}(2), \ldots, x^{(1)}(n))$ will be the new time series.

After the new time series $x^{(1)}$ is formed, it can be used to predict the future values based on the previous and current values. The previous and the current values of the time series are used as input for the following prediction model:

$$\left\{x^{(1)}(t+1), x^{(1)}(t+2), \ldots, x^{(1)}(t+h)\right\} = F\left(x^{(1)}(t), x^{(1)}(t-1), \ldots, x^{(1)}(t-m+1)\right), \tag{3.3}$$

where h represents the number of ahead predictions, F is the prediction model, and m is the size of repressor. According to (3.3), we can obtain the training sample.

In this paper, we use the following prediction strategy to predict the next data-point value. The strategy is that the prediction data will join into the sample data set as known data. This prediction process is called incremental algorithm. The model can be constructed with one-step prediction:

$$x^{(1)}(t+1) = F\left(x^{(1)}(t), x^{(1)}(t-1), \ldots, x^{(1)}(t-m+1)\right). \tag{3.4}$$

The regressor of the model is defined as the vector of inputs $(x(t), x(t-1), \ldots, x(t-m+1))$.

After getting the prediction values sequence, we can compute the real prediction results by IAGO:

$$\hat{x}^{(0)}(t+1) = \hat{x}^{(1)}(t+1) - \hat{x}^{(1)}(t). \tag{3.5}$$

3.2. Modified LS-SVR Model Based Incremental Learning

For adaptive online prediction, real-time is the basic requirement, that is, in the adaptive online prediction based on incremental learning algorithm, the new predicted data is constantly added, and we hope that the total computing time of training and prediction should be less than the sampling period. In this case, the most important issue is to compute Lagrange multipliers α and bias term b, which has effect on the computing time because they need recalculate when the new sample data is added in. After analyzing the research results of Yaakov et al. [23] we develop a simple prediction model based on LS-SVR to solve the problem.

Suppose the training sample data are described by the input $\{(x_i, y_i)\}_{i=1}^{i=t}$, $(x_i \in R^d,\ y_i \in R)$. In time t, give the sample data set $\{(\mathbf{x}(t), \mathbf{y}(t)\}$, $\mathbf{x}(t) = [x_1(t), x_2(t), \ldots, x_t(t)]$, $\mathbf{y}(t) = [y_1(t), y_2(t), \ldots, y_t(t)]$. As time t goes on, the new sample data joins. We give a new approach to eliminate the bias term b.

Let $\mathbf{w}(t) = \begin{bmatrix} w(t) \\ b/\lambda \end{bmatrix}$ and $\phi_i(t) = \begin{bmatrix} \varphi_i(t) \\ \lambda \end{bmatrix}$, where $\varphi_i(t)$ is the mapping function of the ith input data at time t and λ is a constant. Then rewritten optimization problem of (2.2) as follows:

$$\begin{aligned} \min \quad & J(w, e) = \frac{1}{2}\mathbf{w}(t)^T\mathbf{w}(t) + \frac{c}{2}\sum_{i=1}^{t} {e_i}^2(t), \\ \text{s.t.} \quad & y_i(t) = \mathbf{w}(t)^T\phi_i(t) + e_i(t), \quad i = 1, 2, \ldots, t. \end{aligned} \tag{3.6}$$

The solution of (3.6) is also obtained after constructing the Lagrange function:

$$L(\mathbf{w}, e, a) = \frac{1}{2}\mathbf{w}(t)^T\mathbf{w}(t) + \frac{1}{2}c\sum_{i=1}^{t} e_i^2(t) - \sum_{i=1}^{t} a_i(t)\Big(\mathbf{w}(t)^T\phi_i(t) + e_i(t) - y_i(t)\Big), \tag{3.7}$$

where $a_i(t)$ is the ith Lagrange multiplier. The new optimal conditions are shown as follows

$$\begin{aligned} \frac{\partial L}{\partial \mathbf{w}(t)} &= \mathbf{w}(t) - \sum_{i=1}^{t} a_i(t)\phi_i(t) = 0 \Longrightarrow \mathbf{w}(t) = \sum_{i=1}^{t} a_i(t)\phi_i(t), \\ \frac{\partial L}{\partial a_i(t)} &= \mathbf{w}(t)^T\phi_i(t) + e_i(t) - y_i(t) = 0 \Longrightarrow y_i(t) = \mathbf{w}(t)^T\phi_i(t) + e_i(t), \\ \frac{\partial L}{\partial e_i(t)} &= ce_i(t) - a_i(t) = 0 \Longrightarrow e_i(t) = \frac{1}{c}a_i(t). \end{aligned} \tag{3.8}$$

After eliminating $e_i(t)$ and $\mathbf{w}(t)$, the solution of (3.8) is given by the following set of linear equations:

$$\begin{bmatrix} \varphi_1(t)^T\varphi_1(t)+\lambda^2+\dfrac{1}{c} & \varphi_2(t)^T\varphi_1(t)+\lambda^2 & \cdots & \varphi_t(t)^T\varphi_1(t)+\lambda^2 \\ \varphi_1(t)^T\varphi_2(t)+\lambda^2 & \varphi_2(t)^T\varphi_2(t)+\lambda^2+\dfrac{1}{c} & \cdots & \varphi_t(t)^T\varphi_2(t)+\lambda^2 \\ \vdots & \vdots & \vdots & \vdots \\ \varphi_1(t)^T\varphi_t(t)+\lambda^2 & \varphi_2(t)^T\varphi_t(t)+\lambda^2 & \cdots & \varphi_t(t)^T\varphi_t(t)+\lambda^2+\dfrac{1}{c} \end{bmatrix} \begin{bmatrix} a_1(t) \\ a_2(t) \\ \vdots \\ a_t(t) \end{bmatrix} = \begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_t(t) \end{bmatrix}. \tag{3.9}$$

Let

$$\begin{aligned} \mathbf{a}(t) &= \begin{bmatrix} a_1(t) & a_2(t) & \cdots & a_t(t) \end{bmatrix}^T, \\ \mathbf{y}(t) &= \begin{bmatrix} y_1(t) & y_2(t) & \cdots & y_t(t) \end{bmatrix}^T, \\ k_{ij}(t) &= \varphi_i(t)^T\varphi_j(t), \\ H_{ij}(t) &= k_{ij}(t)+\lambda^2, \\ \mathbf{H}(t) &= \left\{H_{ij}(t)\right\}_{i,j=1}^{t}+\frac{\mathbf{I}}{c}, \end{aligned} \tag{3.10}$$

where $\mathbf{I}$ is an identity matrix with $t \times t$. Equation (3.9) can be rewritten as follows:

$$\mathbf{a}(t) = \mathbf{y}(t)\mathbf{H}(t)^{-1}. \tag{3.11}$$

The new regression function is presented as follows:

$$y_{t+1}(t) = \sum_{i=1}^{t} a_i(t)\Big(k(x_i(t), x_{t+1}(t)) + \lambda^2\Big). \tag{3.12}$$

According to (3.11) and (3.12), when the new sample data $y_{t+1}(t)$ is added, the calculation only needs to compute parameter $\mathbf{a}$.

At time $t+1$, $\mathbf{a}$ can be calculate via (3.11), that is, $\mathbf{a}(t+1) = \mathbf{y}(t+1)\mathbf{H}(t+1)^{-1}$. Here,

$$\begin{aligned}
\mathbf{y}(t+1) &= \left[\mathbf{y}(t)^T \quad y_{t+1}(t)\right]^T, \\
\mathbf{H}(t+1) &= \begin{bmatrix} \mathbf{H}(t) & \mathbf{V}(t+1) \\ \mathbf{V}(t+1)^T & v(t+1) \end{bmatrix}, \\
\mathbf{V}(t+1) &= \begin{bmatrix} \varphi_{t+1}(t)^T \varphi_1(t) + \lambda^2 \\ \varphi_{t+1}(t)^T \varphi_2(t) + \lambda^2 \\ \vdots \\ \varphi_{t+1}(t)^T \varphi_t(t) + \lambda^2 \end{bmatrix}, \\
v(t+1) &= \varphi_{t+1}(t)^T \varphi_{t+1}(t) + \lambda^2 + \frac{1}{c}, \\
x_{t+1}(t) &= x_{t+1}(t+1), \\
y_{t+1}(t) &= y_{t+1}(t+1), \\
\varphi_{t+1}(t) &= \varphi_{t+1}(t+1).
\end{aligned} \tag{3.13}$$

$\mathbf{H}(t+1)^{-1}$ can be fast computed with block matrix approach [24, 25].

From above description, it is obvious that the modified prediction model is a simple model, and it is maybe applied to reduce the computing time for its less number of parameters.

4. Experiments and Results Analysis

We perform two simulation experiments and an application experiment to evaluate the proposed scheme. All the experiments adopt MatlabR2011b with LS-SVMlab1.8 Toolbox (The software can be downloaded from http://www.esat.kuleuven.be/sista/lssvmlab) under Windows XP operating system.

In this paper, we use the prediction Root Mean Squared Error (RMSE) [26] as evaluation criteria, which is defined as follows:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{k=1}^{n}\left(x(k) - x'(k)\right)^2}, \tag{4.1}$$

where n is the number of training sample data, $x'(k)$ and $x(k)$ are the prediction and the actual value, respectively.

4.1. Simulation Experiments and Results Analysis

In order to validate the performance of the proposed AGO-based method described in Section 3.1 and the modified LS-SVR model-based incremental learning described in Section 3.2, we perform two simulation experiments. In the experiments, the sample time series data of variable x has 75 sample data values, which come from one complex avionics system (shown in Figure 1, omit dimension).

The data points from 1 to 50 in time series are taken as 45 initial training sample data. The first sample data set consists of points 1 to 6, with the first 5 as the input sample vector

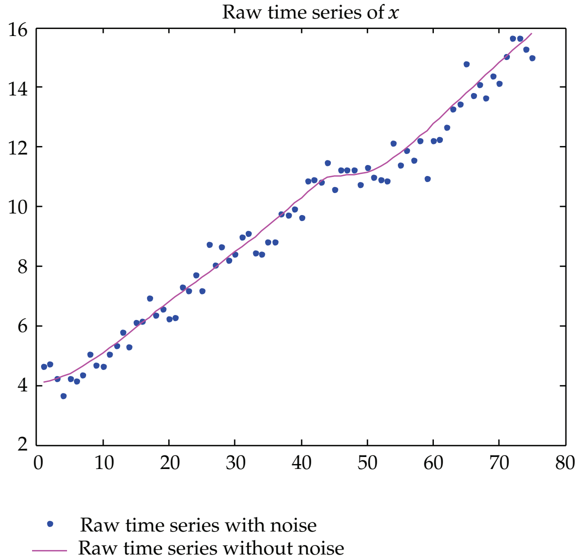

Figure 1: Sample time series of a certain avionics system.

Table 1: Prediction results of Experiment I.

Method	TrTime/(s)	PrTime/(s)	PrMR
Traditional LS-SVR in [4]	0.0632	0.0200	2.4411
Proposed AGO-based method in Section 3.1	0.0706	0.0209	2.1569

and the 6th point as the output. The second sample data set consists of points 2 to 7, with the point 2 through 6 as input sample vector and 7th point as the output. This way we have 45 training data out of the first 50 data points. And then we predict the no. 51 to no. 75 time series data using the trained model. The performance is measured by the prediction mean RMSE (PrMR), training time (TrTime), and prediction time (PrTime). The experiments are repeated 100 times and the average results are obtained.

In the simulation Experiment I, we compare the proposed AGO-based method with traditional LS-SVR presented in [4]. Here, Gaussian RBF kernel $k(x, y) = \exp(-\|x - y\|^2/2\sigma^2)$ is adopted as kernel function, and the parameters are jointly optimized with traditional gridding search method, where the search rang of c and σ^2 is $[10, 3000]$. The prediction results are shown in Figure 2 (ALS-SVR presents the proposed AGO-based method in the figure) and Table 1.

From Table 1 and Figure 2, we can see that the proposed AGO-based method has better prediction accuracy. This is may be due to that the proposed AGO-based method can avoid the random disturbances existing in the raw time series and improve the regularity of the time series.

In simulation Experiment II, we compare the two incremental learning-based methods: one is the proposed modified LS-SVR model, the other one is the traditional LS-SVR model. The other selections are the same as Experiment I. The prediction results are reported in Table 2.

The results in Table 2 show that the modified LS-SVR model can reduce the computing time more with the same prediction accuracy as the traditional LS-SVR model. Compared

Table 2: Prediction results of Experiment II.

Incremental learning-based method	TrTime/(s)	PrTime/(s)	PrMR
Traditional LS-SVR model	0.1872	0.0206	2.2142
Proposed modified LS-SVR model in Section 3.2	0.1186	0.0205	2.2142

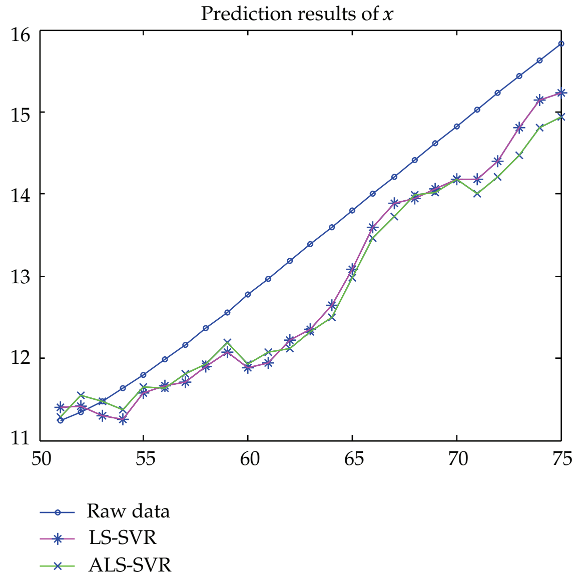

Figure 2: Prediction results of x.

with the results of traditional LS-SVR in [4] (shown in Table 1), the incremental learning-based methods have better prediction accuracy, which is increased by 10.25%. This may be due to that the prediction information is used more fully by incremental learning algorithm.

4.2. Application Experiment and Results Analysis

In the application experiment, the experimental sample time series with 2000 sample data values, which come from a certain complex electronic system, shown in Figure 3.

We set the first 1000 time series data as training samples, and any continuous 11 are taken as a sample, where the data of the first 10 data compose an input sample vector and last one as the output vector, that is, in the application experiment, we have 990 training data. And then we predict the no. 1001 to no. 2000 time series data using the trained prediction model. The test is also repeated 100 times.

The application experiment is executed using the proposed method presented in Section 3 (called Proposed Method) and traditional incremental learning based LS-SVR method (called Traditional Method). The prediction performance is measured by training time (TrTime), prediction time (PrTime), and prediction mean RMSE (PrMR). The results are shown in Figures 4 and 5 and Table 3. In addition, we also utilize traditional gridding search method to joint optimized all the parameters, and the search rang of c and σ^2 is $[0.1, 200]$.

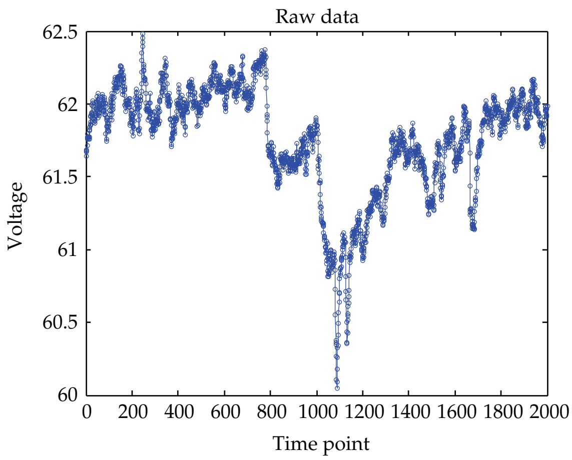

Figure 3: Sample time series of a certain complex electronic system.

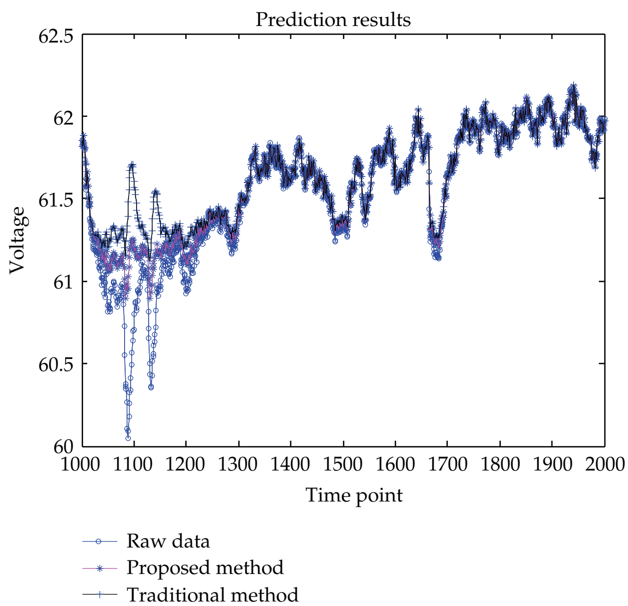

Figure 4: Prediction results of the application experiment.

Table 3: Prediction results of application experiment.

Method	TrTime/(s)	PrTime/(s)	PrMR
The traditional method	1.2921	0.2208	3.1941
The proposed method	0.8245	0.2289	2.0963

Figures 4 and 5 and Table 3 show that the proposed method has better prediction accuracy with lower computing time. This may be due to that we modify the LS-SVR model to reduce the parameters of the model, and the data quality and regularity are improved by AGO. Thus, the proposed method maybe has higher application value.

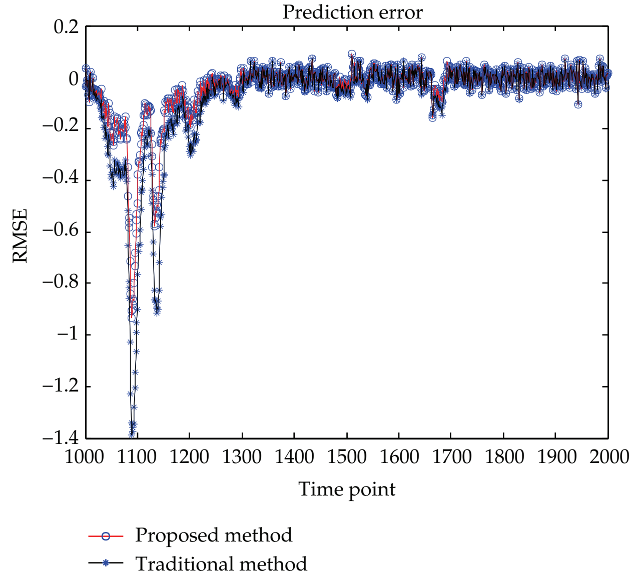

Figure 5: Prediction error of the application experiment.

5. Conclusions

In this paper, we address the issue of adaptive online prediction method based on LS-SVR model. We utilize two approaches to gain better prediction accuracy. Firstly, we make the accumulated generating operation (AGO) with raw time series to reform the quality of raw time series, which avoids the random disturbances and improves the regularity of the time series. Secondly, we modify the traditional LS-SVR model with simple form to simplify incremental learning, which helps to reduce the computing time in the process of adaptive online training and prediction.

We conduct three prediction experiments to demonstrate the effectiveness of the propose method. The results show that the prediction method has better prediction performance, and it is suitable for the adaptive online prediction in practice.

Acknowledgments

The authors would like to thank Professor Ming J. Zuo, the Director of Reliability Research Laboratory of University of Alberta, for his useful comments. This work is supported by the National Basic Research Program of China (973 Program), the National Natural Science Foundation of China (no. 61001023 and no. 61101004), Shaanxi Natural Science Foundation (2010JQ8005), and Aviation Science Foundation of China (2010ZD53039).

References

[1] W. Caesarendra, A. Widodo, P. H. Thom, B. S. Yang, and J. D. Setiawan, "Combined probability approach and indirect data-driven method for bearing degradation prognostics," *IEEE Transactions on Reliability*, vol. 60, no. 1, pp. 14–20, 2011.

[2] D. Liu, Y. Peng, and X. Y. Peng, "Online adaptive status prediction strategy for data-driven fault prognostics of complex systems," in *Proceedings of the Prognostics and System Health Management Conference (PHM '11)*, pp. 1–6, Shenzhen, China, May 2011.

[3] M. Pecht and R. Jaai, "A prognostics and health management roadmap for information and electronics-rich systems," *IEICE Fundamentals Review*, vol. 3, no. 4, pp. 25–32, 2010.

[4] J. Qu and M. J. Zuo, "An LSSVR-based algorithm for online system condition prognostics," *Expert Systems with Applications*, vol. 39, no. 5, pp. 6089–6102, 2012.

[5] M. Qi and G. P. Zhang, "Trend time-series modeling and forecasting with neural networks," *IEEE Transactions on Neural Networks*, vol. 19, no. 5, pp. 808–816, 2008.

[6] V. V. Gavrishchaka and S. Banerjee, "Support vector machine as an efficient framework for stock market volatility forecasting," *Computational Management Science*, vol. 3, no. 2, pp. 147–160, 2006.

[7] Y. M. Guo, C. B. Ran, X. L. Li, and J. Z. Ma, "Adaptive online prediction method based on LS-SVR and its application in an electronic system," *Journal of Zhejiang University Science C*, vol. 13, no. 12, pp. 881–890, 2012.

[8] V. Kecman, *Learning and Soft Computing: Support Vector Machines. Neural Networks, and Fuzzy Logic Models*, MIT Press, Cambridge, Mass, USA, 2001.

[9] J. V. Hansen and R. D. Nelson, "Neural networks and traditional time series methods: a synergistic combination in state economic forecasts," *IEEE Transactions on Neural Networks*, vol. 8, no. 4, pp. 863–873, 1997.

[10] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, NY, USA, 1995.

[11] J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewaller, *Least Squares Support Vector Machines*, World Scientific Publishing, 2002.

[12] G. Hebin and G. Xiaoqing, "Application of least squares support vector regression in network flow forecasting," in *Proceedings of the 2nd International Conference on Computer Engineering and Technology (ICCET '10)*, pp. V7-342–V7-345, April 2010.

[13] Y. H. Zhao, P. Zhong, and K. N. Wang, "Application of least squares support vector regression based on time series in prediction of gas," *Journal of Convergence Information Technology*, vol. 6, no. 1, pp. 243–250, 2011.

[14] Y. Guo, Z. Zhai, and H. Jiang, "Weighted prediction of multi-parameter chaotic time series using least squares support vector regression (LS-SVR)," *Journal of Northwestern Polytechnical University*, vol. 27, no. 1, pp. 83–87, 2009.

[15] A. Shilton, D. T. H. Lai, and M. Palaniswami, "A division algebraic framework for multidimensional support vector regression," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 40, no. 2, pp. 517–528, 2010.

[16] Y. M. Guo, X. L. Li, G. H. Bai, and J. Z. Ma, "Time series prediction method based on LS-SVR with modified Gaussian RBF," in *Neural Information Processing*, vol. 7664 of *Lecture Notes in Computer Science, Part 2*, pp. 9–17, 2012.

[17] H. Liu and Z. H. Sun, "An online algorithm for support vector machine based on subgradient projection," *Control Engineering and Applied Informatics*, vol. 13, no. 3, pp. 18–24, 2011.

[18] S. Shalev-Shwartz, *Online learning: theory, algorithms, and applications [Ph.D. thesis]*, The Senate of the Hebrew University, July 2007.

[19] W. M. Tang, "New forecasting model based on grey support vector machine," *Journal of Systems Engineering*, vol. 21, no. 4, pp. 410–413, 2006.

[20] D. H. Zhan, Y. X. Bi et al., "Power load forecasting method based on series grey neural network," *System Engineering-Theory & Practice*, no. 23, pp. 128–132, 2004.

[21] J. L. Deng, *The Primary Methods of Grey System Theory*, Huazhong University of Science and Technology Press, Wuhan, China, 2004.

[22] F. Ojeda, J. A. K. Suykens, and B. De Moor, "Low rank updated LS-SVM classifiers for fast variable selection," *Neural Networks*, vol. 21, no. 2-3, pp. 437–449, 2008.

[23] E. Yaakov, M. Shie, and M. Ron, "Sparse online greedy support vector regression," in *Proceedings of the 13th European Conference on Machine Learning*, Berlin, Germany, 2002.

[24] J. K. Zhang, *Linear Model's Parameters Estimation Method and Its Improvement*, National University of Defense Technology Press, Changsha, China, 1992.

[25] C. C. Cowen, *Linear Algebra for Engineering and Science*, Indiana West Pickle Press, West Lafayette, Ind, USA, 1996.

[26] M. Y. Ye, X. D. Wang, and H. R. Zhang, "Chaotic time series forecasting using online least squares support vector machine regression," *Acta Physica Sinica*, vol. 54, no. 6, pp. 2568–2573, 2005.