Article citation info:
GUO Y-M, RAN C-B, LI X-L, MA J-Z, ZHANG L. Weighted prediction method with multiple time series using multi-kernel least squares support vector regression. Eksploatacja i Niezawodnosc – Maintenance and Reliability 2013; 15 (2): 188–194.

Yang-Ming GUO
Cong-Bao RAN
Xiao-Lei LI
Jie-Zhong MA
Lu ZHANG

WEIGHTED PREDICTION METHOD WITH MULTIPLE TIME SERIES USING MULTI-KERNEL LEAST SQUARES SUPPORT VECTOR REGRESSION

METODA WAŻONEJ PREDYKCJI WIELOKROTNYCH SZEREGÓW CZASOWYCH Z WYKORZYSTANIEM WIELOJĄDROWEJ REGRESJI WEKTORÓW WSPIERAJĄCYCH METODĄ NAJMNIEJSZYCH KWADRATÓW (LS-SVR)

Least squares support vector regression (LS-SVR) has been widely applied in time series prediction. Based on the case that one fault mode may be represented by multiple relevant time series, we utilize multiple time series to enrich the prediction information hiding in time series data, and use multi-kernel to fully map the information into high dimensional feature space, then a weighted time series prediction method with multi-kernel LS-SVR is proposed to attain better prediction performance in this paper. The main contributions of this method include three parts. Firstly, a simple approach is proposed to determine the combining weights of multiple basis kernels; Secondly, the internal correlative levels of multiple relevant time series are computed to present the different contributions of prediction results; Thirdly, we propose a new weight function to describe each data's different effect on the prediction accuracy. The experiment results indicate the effectiveness of the proposed method in both better prediction accuracy and less computation time. It maybe has more application value.

Keywords: *time series, weighted prediction, least squares support vector regression (LS-SVR), multiple kernel learning (MKL).*

Regresja wektorów wspierających metodą najmniejszych kwadratów (LS-SVR) jest szeroko stosowana w predykcji szeregów czasowych. Opierając się na fakcie, że jeden rodzaj niezdatności może być reprezentowany przez wiele relewantnych szeregów czasowych, w niniejszej pracy wykorzystano wielokrotne szeregi czasowe do wzbogacenia informacji predykcyjnych ukrytych w szeregach czasowych oraz posłużono się metodą uczenia wielojądrowego (multi-kernel) w celu mapowania informacji do wysoko wymiarowej przestrzeni cech, a następnie zaproponowano metodę ważonej predykcji wielokrotnych szeregów czasowych z wykorzystaniem wielojądrowej regresji LS-SVR służącą osiągnięciu lepszej wydajności prognozowania.Metoda składa się z trzech głównych części. Po pierwsze, zaproponowano prosty sposób określania łącznej wagi wielu jąder podstawowych. Po drugie, obliczono wewnętrzne poziomy korelacyjne wielokrotnych szeregów czasowych w celu przedstawienia różnego udziału wyników prognozowania. Po trzecie, zaproponowano nową funkcję wagi do opisu różnego wpływu poszczególnych danych na trafność predykcji. Wyniki doświadczenia wskazują na skuteczność proponowanej metody zarówno jeśli chodzi o lepszą trafność predykcji jak i krótszy czas obliczeniowy. Proponowane rozwiązanie ma potencjalnie dużą wartość aplikacyjną.

Słowa kluczowe*: szereg czasowy, predykcja ważona, regresja wektorów wspierających metodą najmniejszych kwadratów (LS-SVR), uczenie wielojądrowe (MKL).*

1. Introduction

Fault or health trend prediction technique has become one of the effective ways to protect the safe operation of high reliable systems. However complex systems often show complex dynamic behaviors and uncertainty, which lead to hardly establishing their precise physical models. In this case, in order to obtain the satisfactory prediction results, time series analysis methods are often used to perform the prediction in practice [2, 12, 15, 19, 26]. Among the known non-linear time series prediction methods, the effectiveness of statistics theory based methods have been demonstrated, such as Artificial Neural Networks (ANN), Support Vector Regression (SVR), etc.

ANN has been applied in many fields due to its universal approximation property. However ANN suffers from local minimum traps, difficulty in determining the hidden layer size and learning rate, poor capacity for generalization, etc. [8, 10, 32] On the contrary, SVR overcomes the problems existing in ANN. SVR aims at the global optimum and exhibits better accuracy in non-linear and non-stationary time series data prediction due to its implementation of the structural risk minimization principle [10, 27, 28]. But complexity of SVR depends not only on the input space dimension, but also on the number of sample data. For large sample data, the quadratic programming (QP) problem is more complex, it will cost a lot of computing time. For this reason, LS-SVR was proposed by Suykens et al. [16, 23] In

LS-SVR, the inequality constrains are replaced by equality constrains. This way, solving a QP is converted into solving linear equations, and the calculation time is reduced significantly. Thus, LS-SVR attracts more attention in time series prediction [5, 6, 19, 20, 26, 33].

In many applications of fault or health condition prediction, one certain condition may be represented by one major variable and several relevant variables. In order to achieve satisfactory prediction, these auxiliary time series relating to the major time series are utilized to enrich the information and improve the prediction accuracy. In this case, how to fully present the information hiding in the multiple time series data becomes a key issue. The kernel function is used to map the input data to high dimensional feature space, so it influences the learning performance of LS-SVR, that means a appropriate kernel function can more fully present the information in time series data. However, LS-SVR with a single kernel function is not a good choice to all the data sets, especially for multiple time series data, although the kernel parameters can be optimally chosen to enhance the generalization capability.

Some researchers applied Multiple Kernel Learning (MKL) to solve the above problems [13, 29]. MKL provides a more flexible framework than single kernel. Under the framework, the information in time series data can be mined more adaptively and effectively, i.e., MKL explicitly learns the weights of basis kernels from different time series data sources, and the relationships among them are learned meanwhile. Moreover, MKL can avoid the difficulty of appropriate kernel function selection. Thus, multi-kernel LS-SVR has better prediction accuracy in practice [13,14].

However in order to obtain better prediction results, some problems which accord to the requirements of applications, should still be considered, such as fast and accurate prediction.

(1) In MKL framework, the time series data samples are generally learned by a linear convex combination of basis kernels. The reported methods of determining the combining weights of basis kernels, such as software packages [1] and joint optimization selection algorithm [9, 34], are always complex. They are generally unapt for applications.

(2) Although some researchers also used multiple relevant time series to perform prediction [6, 17, 31, 35, 36], different interrelated levels between major time series and auxiliary time series have different influences on prediction accuracy. It is necessary to determine the interrelated levels between them, and they represent the weight values of each time series for the prediction.

(3) The original prediction methods always assume that all the training time series data have same contribution to the prediction. According to the new information principle [3], the data near the current prediction point will affect the prediction much more. Thus, in order to achieve more accurate results, each sample data should be weighted according to their distance far from the current prediction point.

Thus a weighted prediction method with multi-kernel LS-SVR using multiple relevant time series is proposed in this paper. According to the application requirements, we apply three ways to achieve better prediction results. One is to compute correlative levels of multiple relevant time series to represent their different contributions to prediction results; Secondly, we propose a weight function to present the different influence of each history data on prediction; Finally, we establish a new multi-kernel LS-SVR based on time-distance-weighted factor of each time series, and in order to improve the application value of the proposed method, a simple approach of determining the combining weights of the multiple basis kernels is proposed to reduce the calculation time.

The rest of the paper is organized as follows: Section 2 gives a brief review of LS-VR and multiple kernel learning (MKL) algorithm; Section 3 proposes the weighted prediction method which includes three computational approaches: (1) combination coefficients of multiple basis kernels, (2) correlative levels of the multiple time series, and (3) time-distance-weighted factors of each time series data; Section 4 shows simulation and application experiments; and the conclusions are drawn in Section 5.

2. A brief review of related work

2.1. Least squares support vector regression

LS-SVR has many advantages, such as simpler algorithm, faster operation speed, etc. It is widely applied in regression. The goal of LS-SVR is to estimate a function that is as "close" as possible to the target values for every data point, and at same time, is as "flat" as possible for good generalization. The regression principle of LS-SVR can be expressed as follows.

Consider a training data set of n data points $\{x_i, y_i\}_{i=1}^n$ with input data $x_i \in R^d$, and $y_i \in R$ is the corresponding output or target value. LS-SVR is to construct the regression function with the following form

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

where $\varphi(\cdot)$ is used to non-linearly map the input data to the high dimensional feature space , w is the weight vector and b is the bias term.

According to the structural risk minimization principle[27,28], the function regression problem can be represented as a constraint optimization problem as follows

$$\begin{aligned} \min \quad & J(w,b) = \frac{1}{2}\|w\|^2 + \frac{c}{2}\sum_{i=1}^{n} e_i^2 \\ s.t. \quad & y_i = w^T \varphi(x_i) + b + e_i \end{aligned} \tag{2}$$

where $i = 1,2,...,n$, $J(w,b)$ is the cost function, c is a positive real constant (regularization parameter) and $e_i \in R$ is an error variable.

In order to solve the above constraint optimization problem, the Lagrangian function is constructed by transforming constraint optimization problems into unconstraint ones

$$L(w,b,e,\alpha) = \frac{1}{2}w^T w + \frac{1}{2}c\sum_{i=1}^{n} e_i^2 - \sum_{i=1}^{n}\alpha_i (w^T \varphi(x) + b + e_i - y_i) \tag{3}$$

where α_i is the i-th Lagrange multiplier. It is obvious that the optimal solution of Eq.(2) satisfies the Karush-Kuhn-Tucker (KKT) conditions. The optimal conditions are expressed as follows

$$\begin{cases} \dfrac{\partial L}{\partial w} = w - \sum_{i=1}^{n}\alpha_i\varphi(x_i) = 0 \Rightarrow w = \sum_{i=1}^{n}\alpha_i\varphi(x_i) \\ \dfrac{\partial L}{\partial b} = -\sum_{i=1}^{n}\alpha_i = 0 \Rightarrow \sum_{i=1}^{n}\alpha_i = 0 \\ \dfrac{\partial L}{\partial \alpha_i} = w^T\varphi(x_i) + b + e_i - y_i = 0 \Rightarrow y_i = w^T\varphi(x_i) + b + e_i \\ \dfrac{\partial L}{\partial e_i} = ce_i - \alpha_i = 0 \Rightarrow e_i = \dfrac{1}{c}\alpha_i \end{cases} \tag{4}$$

After eliminating w and e_i from Eq.(4), we could obtain the solution by the following linear equations

$$\begin{bmatrix} 0 & \mathbf{1}_n^T \\ \mathbf{1}_n & \mathbf{K}+\mathbf{I}/\mathbf{c} \end{bmatrix} \begin{bmatrix} b \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{y} \end{bmatrix} \quad (5)$$

where $\mathbf{K}(i,j) = k(x_i, x_j) = \varphi(x_i)^T \varphi(x_j)$, $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, ..., \alpha_n]^T$, $\mathbf{1}_n$ is an n-dimensional vector of all ones, $\mathbf{I}$ is a unite matrix and $\mathbf{y} = [y_1, y_2, ..., y_n]^T$. Eq.(5) can be factorized into a positive definite system[18].

Let $\mathbf{H} = \mathbf{K} + \mathbf{I}/c$, we get the equations from Eq.(5)

$$\begin{cases} \mathbf{1}_n^T \boldsymbol{\alpha} = 0 \\ \mathbf{1}_n b + \mathbf{H}\boldsymbol{\alpha} = \mathbf{y} \end{cases} \quad (6)$$

Then Lagrange dual variables $\boldsymbol{\alpha}$ and bias term b are obtained solely by

$$\begin{cases} \boldsymbol{\alpha} = \mathbf{H}^{-1}(\mathbf{y} - \mathbf{1}_n b) \\ b = \mathbf{1}_n^T \mathbf{H}^{-1} \mathbf{y} (\mathbf{1}_n^T \mathbf{H}^{-1} \mathbf{1}_n)^{-1} \end{cases} \quad (7)$$

Any unlabeled input x can be subsequently regression estimation by the following function

$$\hat{y}(x) = \sum_{i=1}^{n} \alpha_i k(x_i, x) + b \quad (8)$$

2.2. Multiple kernel learning algorithm

The selection of kernel function and its corresponding parameters is the key issue for prediction accuracy. However no rules have been reported to guide the selection in theory. In this case, MKL was proposed by Lanckriet, et al. [13]

In MKL framework, a combined kernel function is defined as the weight sum of several individual basis kernels. Researchers proposed a variety of methods to integrate multiple basis kernels [30]. The linear convex combination of basis kernels is most frequently used. In this paper, using the equations described by Sonnenburg et al. [24] We consider the following form of combined kernel

$$\mathbf{K} = \sum_{j=1}^{m} \mu_j \mathbf{K}_j \quad (9)$$

where $\sum_{j=1}^{m} \mu_j = 1$, $\mu_j \geq 0$ ($j = 1,2,...,m$), m is the number of basis kernels, μ_j is the combining weight of the j-th basis kernel. Obviously $\mathbf{K}$ is Symmetric Positive Semidefinite Matrix [22], i.e., $\mathbf{K} \succeq 0$. Afterward, all kernel matrices $\mathbf{K}_j$ are normalized by replacing $\mathbf{K}_j(x_p, x_q) \Big/ \sqrt{\mathbf{K}_j(x_p, x_p)\mathbf{K}_j(x_q, x_q)}$ to get unit diagonal matrices.

The key of MKL is to obtain the optimal combining weights μ_j. This problem can be solved as a QCQP problem[34] efficiently by general-purpose optimization software packages[14]. Moreover, some researchers also applied joint optimization selection algorithms to obtain the combining weights μ_j and the parameters of LS-SVR simultaneously. But all the solution methods are complex in practice. Thus, we proposed a simple method to fix this problem in this paper.

3. Proposed weighted multiple time series prediction method

In this section, we propose a new scheme to obtain better prediction performance. Firstly, we use multiple kernel functions consisting of several basis kernels to show the information more effectively in the high dimensional mapping feature space. A simple approximate approach is presented to compute the combining weights with less calculation complexity. Then we propose the weighted prediction method. In the method, we calculate the correlation coefficient of each time series as weight factor, which present the influence factor on prediction accuracy with each time series, and based on the distance of each time series data far from the current prediction point, we weight the effects of the history data on prediction via a modified weight function.

3.1. Combination coefficients of multiple basis kernels

In this paper, we apply the new kernel with a linear combination of basis kernels, shown as Eq.(9). In order to reduce the computing complexity, we propose a simple method to determine combining weights, i.e., the combining weights of basis kernels are determined according to the root mean squared error (RMSE) of each LS-SVR with each single basis kernel. This way, smaller RMSE value will get bigger weight value. The RMSE of multiple time series prediction is defined as follows

$$\sigma_{RMSE} = \sqrt{\frac{1}{MN} \sum_{i=1}^{N} \sum_{k=1}^{M} (y_i(k) - \hat{y}_i(k))^2} \quad (10)$$

where M is the number of the relevant parameters, N is the number of original training sample data, and $y_i(k)$ and $\hat{y}_i(k)$ are the prediction value and actual value respectively. The linear combining weights μ_j can be computed as follows

$$\mu_j = \frac{\sum_{r=1}^{m} \sigma_r - \sigma_j}{(m-1)\sum_{r=1}^{m} \sigma_r} \quad (11)$$

where σ_j is the prediction RMSE of the j-th kernel, $\sum_{r=1}^{m} \sigma_r$ is the sum RMSE of all basis kernels, and $\sum_{r=1}^{m} \sigma_r - \sigma_j$ presents the contribution of the j-th kernel.

Obviously, the proposed method of calculating combination coefficients has less complexity comparing with the methods described in section 2.

3.2. Weight factors of multiple time series

3.2.1. Weight factors of major and auxiliary time series

Multiple relevant time series are used to enrich the information in data. However, each time series has different effects on prediction because they have different degrees of information which represent the system's fault or health condition. In this paper, we select the time series, which mainly represents the system fault or health state, as major time series, the others are auxiliary time series. Then the correlation coefficients between the major time series and the auxiliary time series will be computed, and they will be utilized to improve the prediction accuracy.

The purpose of correlation analysis is to measure and interpret the strength of linear or non-linear relationship between two continuous variables [11, 22]. We select the commonly used correlation coefficients, Pearson correlation coefficient [4, 7], to assess the strength of the relationships of multiple relevant time series. The Pearson correlation coefficient computing formula is shown as follows

$$R=\frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2\sum_{i=1}^{n}(y_i-\bar{y})^2}} \tag{12}$$

where R is the correlation coefficient between bivariate data x_i and y_i values ($i=1,2,\ldots,n$), $\bar{x}$ and $\bar{y}$ are the mean values of the x_i and y_i respectively. The Pearson correlation coefficient may be computed by means of a computer-based statistics program "Microsoft Excel" using the option "Correlation" under the option "Data Analysis Tools". Moreover it can also be calculated by Matlab.

3.2.2. Time-distance-weighted factors of each time series

The time series data closing to the current prediction point have greater relevance to current prediction, on the contrary, less relevance with data far from the current prediction point. Hence, we propose a modified weight function to present the different weight factor of each historical data.

According to Ref.[25] and Ref.[37], consider the generation sample set from the raw time series $\{x_k, y_k\}$ ($k=1,2,\ldots,n$), we define a new weight function of x_k as follows

$$d_k=e^{-(n-k)^2/2\lambda^2}, \quad k=1,2,\ldots,n \tag{13}$$

where λ is a given parameter, and a small d_i can reduce the storage of historical data and speed up the training. The objective function is expressed as follows

$$\begin{aligned} &\min \quad J(w,e)=\frac{1}{2}w^Tw+\frac{1}{2}cd_k\sum_{i=1}^{n}e_k^2 \\ &s.t. \quad y_k=w^T\varphi(x_k)+b+e_k, \quad k=1,2,\ldots,n \end{aligned} \tag{14}$$

Then the Lagrangian function is established below

$$L(w,b,e,\alpha)=\frac{1}{2}w^Tw+\frac{1}{2}cd_k\sum_{k=1}^{n}e_k^2-\sum_{k=1}^{n}\alpha_k(w^T\varphi(x)+b+e_k-y_k) \tag{15}$$

where $\alpha_k \ge 0$ ($k=1,2,\ldots,n$) are the Lagrangian multipliers.
According to KKT conditions, we can get the following equations

$$\begin{cases} \frac{\partial L}{\partial w}=w-\sum_{k=1}^{n}\alpha_k\varphi(x_k)=0 \Rightarrow w=\sum_{k=1}^{n}\alpha_k\varphi(x_k) \\ \frac{\partial L}{\partial b}=-\sum_{k=1}^{n}\alpha_k=0 \Rightarrow \sum_{k=1}^{n}\alpha_k=0 \\ \frac{\partial L}{\partial \alpha_k}=w^T\varphi(x_k)+b+e_k-y_k=0 \Rightarrow y_k=w^T\varphi(x_k)+b+e_k \\ \frac{\partial L}{\partial e_k}=cd_ke_k-\alpha_k=0 \Rightarrow e_k=\frac{1}{cd_k}\alpha_k \end{cases} \tag{16}$$

And then rewrite Eq.(5) with a new form as follows

$$\begin{bmatrix} 0 & 1 & \cdots & 1 \\ 1 & k(x_1,x_1)+1/cd_1 & \cdots & k(x_1,x_n) \\ \vdots & \vdots & \cdots & \vdots \\ 1 & k(x_n,x_1) & \cdots & k(x_n,x_n)+1/cd_n \end{bmatrix}\begin{bmatrix} b \\ \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix}=\begin{bmatrix} 0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix} \tag{17}$$

4. Experiments and results analysis

We conduct two simulation experiments and one application experiment to evaluate the performance of proposed method. The prediction experiments are run 100 times and the averages results are taken. All the experiments adopt MatlabR2011b with LS-SVMlab1.8 Toolbox (The software and guide book can be downloaded from http://www.esat.kuleuven.be/sista/lssvmlab) under Windows XP operating system.

4.1. Simulation experiments and results analysis

The simulation experiments include Experiment I and Experiment II. They are conducted to test the proposed method presented in section 3. All the simulation experiments are performed using Lorenz function, because Lorenz function is a typical time series and its variables depend on each other. Lorenz function's corresponding differential equations are shown as follows

$$\begin{cases} x'=-ax+yz \\ y'=-b(y-z) \\ z'=-xy+cy-z \end{cases}$$

Let $a=8/3$, $b=10$, $c=28$, range of initialization as [1,1,1], and simulation step as 0.1 with Fourth-oder Runge-Kutta method. We collect 800 data of the three time series of x (major time series), y and z (auxiliary time series) respectively. We select the first 400 data of x, y and z time series as training data and the last 400 time series data are testing data. In addition, we apply C-C method[21] to generate training sample sets because Lorenz time series is chaotic time series. The prediction efficiency depends on the RMSE, training time (TrTime) and prediction time (PrTime). One Gaussian RBF $K(x,y)=\exp(-\frac{\|x-y\|^2}{2\sigma^2})$ and one Linear kernel function $K(x,y)=x^Ty$ are adopted as basis kernel functions. All the parameters will be jointly optimized by traditional gridding search method with rang of [0.1, 1000]

In Experiment I, we use variable x time series alone to do prediction with tradition multi-kernel LS-SVR reported in Ref.[35] and Ref.[36]. This experiment compares the following two methods: one is obtains the combining weights via optimization software packages (called Method A); the other one is the proposed simple approximate approach in section 3.1(called Method B). The results are shown in Figure1 and Table 1.

From Figure1 and Table 1, we can see that although the prediction accuracy of the new simple computing method (Method B) is adequately reduced comparing with Method A, it also has a good results. Method B can greatly reduce the total computing time, especially

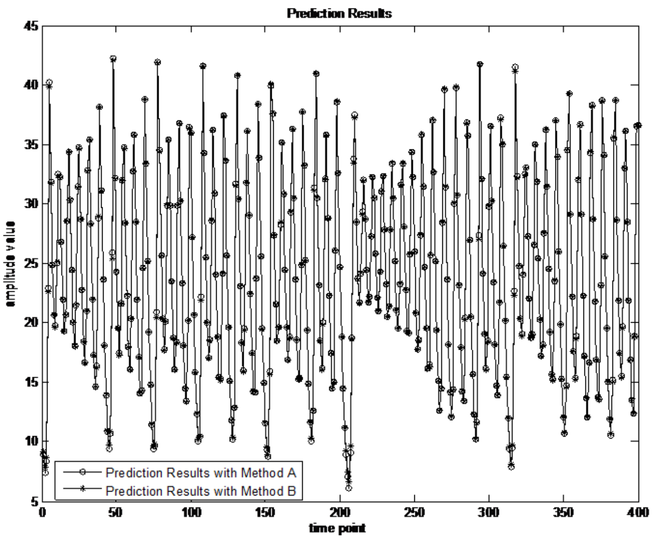

Fig. 1. Prediction Results with Method A and Method B

Table 1. Prediction Results of Method A and Method B

x	TrTime/s	PrTime/s	RMSE
Method A	4.1501	0.2406	2.2258
Method B	3.0480	0.2456	2.8927

training time. The results also indicate that the proposed approximate method is an effective method.

In Experiment II, the variables y and z time series are utilized to enrich the information of variable x time series. We use same multi-kernel LS-SVR model to compare the following methods: Method C doesn't consider the different contributions of each time series and their history data on prediction; Method D applies the proposed approach proposed in section 3.2. Here, we select same kernel functions and optimization method as Experiment I. The results are reported in Table 2, Figure 2 and Table 3.

Figure 2 and Table 3 show that the weighted time series prediction method can improve the prediction accuracy efficiently, and the computing time is not large increase. These are due to that the proposed method takes the different influence factors on prediction accuracy with each auxiliary time series and their history data into account. The other reason is that almost all the middle values at the calculation process of weight factors are already computed and stored in the process of setting up the prediction method.

Table 2. Correlation Coefficient of Time Series

	r_{xy}	r_{xz}
Correlation Coefficient	-0.0581	-0.0348

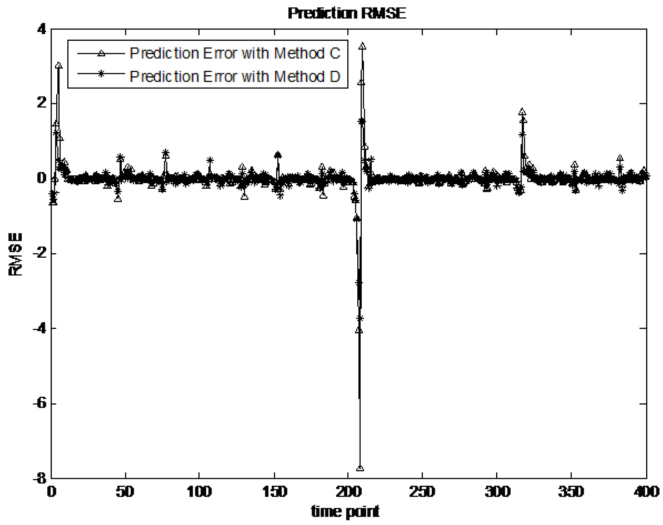

Fig. 2. Prediction Error of Method C and Method D

Table 3. Prediction Results of Method C and Method D

x	TrTime/s	PrTime/s	RMSE
Method C	8.0479	0.8016	2.1765
Method D	9.1241	0.8203	2.0252

4.2. Application experiment and results analysis

We apply the proposed method in a prediction application of one complex avionics system. Four relevant variables time series are collected. They are shown in Figure 3 after preprocessing (omit dimension).

We take the first 15 data of each time series as training samples and look at any continuous 6 as a sample, i.e., the data points from 1 to 15 in the time series are taken as the 10 initial training sample data. The first sample data set consists of points 1 through 6, with the first 5 as the input sample vector and the 6th point as the output. The second sample data set consists of points 2 through 7, with the points 2 through 6 as the input sample vector and the 7th point as the output. This way we have 10 training data out of the first 15 data points.

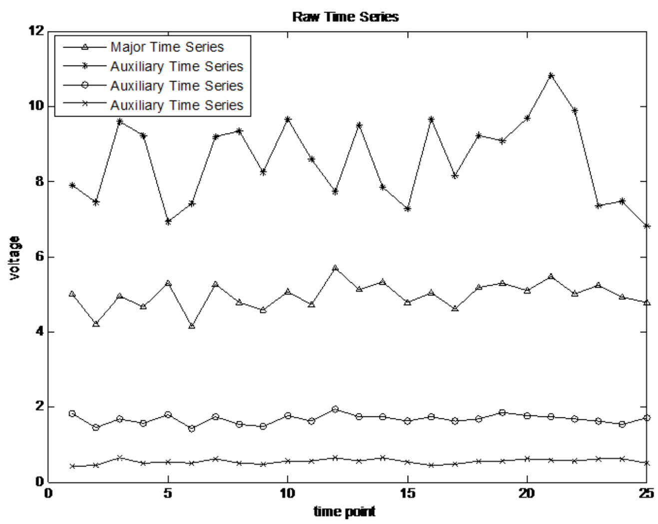

Fig. 3. Raw Time Series of Complex Avionics System

All the parameters are set same as simulation Experiments I and II. The contrast prediction experiment applies Method A and Method E (described in Section 3). The prediction results of the major time series (see Figure 3) are shown in Figure 4 and Figure 5.

In order to show the results clearly, we report them in Table 4 and Table 5.

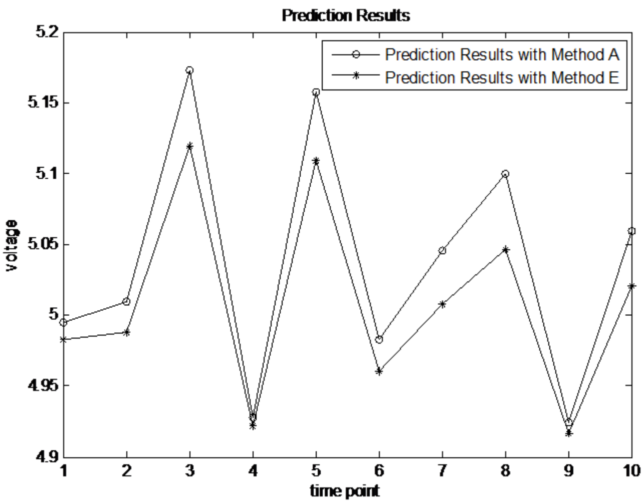

Fig. 4. Results with Method A and Method E

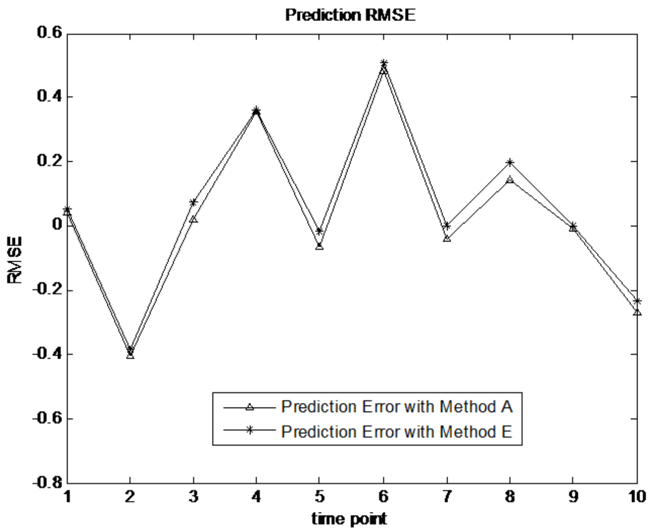

Fig. 5. Error with Method A and Method E

Table 4. Correlation Coefficient of the Time Series

	r_1	r_2	r_3
Correlation Coefficient	0.2791	0.8514	0.6065

Table 5. Prediction Results of Method A and Method E

	TrTime/s	PrTime/s	RMSE
Method A	1.4322	0.0808	0.6771
Method E	1.2927	0.0122	0.5789

From Figure 4, Figure 5 and Table 5, we can see that the proposed method has better prediction results in prediction accuracy and computing time. The results also indicate the proposed method is a good approach, and it can adapt the application better.

5. Conclusions

In this study, we aim at the requirements of applications and analyze the drawbacks of multiple time series prediction by LS-SVR, and then we propose a novel weighted multiple time series prediction method based on multi-kernel LS-SVR. In the new method, we determine the combining weights of each basis kernels by calculating the root mean squared error (RMSE) of prediction using each basis kernel, compute the different contributions to prediction results via correlation analysis between the major time series and auxiliary time series, and make the each historical data with different weight factor based on their distance far from the current prediction point via a modified weight function. The results of simulation and application experiments show that the proposed prediction scheme is an effective approach. It can satisfy the application requirements and may be more valuable in practice.

***Acknowledgments**:*

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), Aviation Science Foundation of China (2010ZD53039), and the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

1. Andersen ED, Andersen AD. The MOSEK interior point optimizer for linear programming: an implementation of the homogeneous algorithm. In High performance optimization Norewll, Frenk H, Roos C, Terlaky T, Zhang S (eds.), Kluwer Academic Publishers, 2000.
2. Caesarendra W, Widodo A, Pham Hong Thom, Bo-Suk Yang, Setiawan JD. Combined Probability Approach and Indirect Data-Driven Method for Bearing Degradation Prognostics. IEEE Transactions on Reliability 2011; 60(1):14–20.
3. Deng JL. The primary methods of grey system theory. Wuhan: Huazhong University of Science and Technology Press, 2004.
4. Freund JE. Mathematical statistics (5th ed.) Upper Saddle River, NJ: Prentice Hall,1992.
5. Guo HB, Guan XQ. Application of Least Squares Support Vector Regression in Network Flow Forecasting. The 2nd International Conference on Computer Engineering and Technology, April, 2010.
6. Guo YM, Zhai ZJ, Jiang HM. Weighted prediction of multi-parameter chaotic time series using least squares support vector regression. Journal of Northwestern Polytechnical University 2009; 27(1):83–86.
7. Goldman RN, Weinberg JS. Statistics: an introduction. Upper Saddle River, NJ: Prentice Hall, 1985.

8. Hansen JV, Nelson RD. Neural networks and traditional time series methods: A synergistic combination in state economic forecasts. IEEE transactions on Neural Networks 1997; 8(4):863–873.
9. Jian L, Xia Z H, Liang X J, Gao C H. Design of a multiple kernel learning algorithm for LS-SVM by convex programming. Neural Networks 2011; (24):476–483.
10. Kecman V. Learning and soft computing: support vector machines, neural networks, and fuzzy logic models. Cambridge, MA,USA: MIT Press, 2001.
11. Krzanowski WJ. Principles of multivariate analysis: a user's perspective. Oxford, England: Clarendon, 1988.
12. Liu DT, Wang SJ, Peng Y, Peng XY. Online adaptive status prediction strategy for data-driven fault prognostics of complex systems. 2011 IEEE conference on autotestcon, Sep. 2011.
13. Lanckriet GR, Cristianini N, Bartlett P L, Ghaoui L E, Jordan M I. Learning the kernel matrix with semidefinite programming. Journal of Machine Learning Research 2004; (5):27-72.
14. Li M, Xu JW, Yang JH, Yang DB. Prediction for chaotic time series based on phase reconstruction of multivariate time series. Journal of University of Science and Technology Beijing 2008; 30(2): 208–211,216.
15. Michael P, Rubyca J. A prognostics and health management roadmap for information and electronics-rich systems. IEICE Fundamentals Review 2010; 3(4):25–32.
16. Muller KR, Smola AJ, Ratsch G, et al. Predicting time series with support vector machines. Artificial Neural Networks 1997; 1327(4):999-1004.
17. Nazih AS, Fawwaz E, Osama M A. Medium-term electric load forecasting using multivariable linear and non-Linear regression. Smart Grid and Renewable Energy 2011; 2:126-135.
18. Ojeda F, Suykens J, De MB. Low rank update LS-SVM classifiers for fast variable selection. Neural Network 2008; 21:443-449.
19. Qu J, Zuo MJ. An LSSVR-based algorithm for online system condition prognostics. Expert Systems with Applications 2012; 39(5):6089-6102.
20. Qu J, Zuo MJ. An LSSVR-based machine condition prognostics algorithm for slurry pump systems. Proceedings of the Canadian Society for Mechanical Engineering Forum 2010, June, 2010.
21. Qin Y Q, Cai WD, Yang BR. Research on phase space reconstruction of non-linear time series. Journal of System Simulation 2008; 20(11): 2969–2973.
22. Rodriguez RN, Correlation. In: Kotz S, Johnson N.L., eds. Encyclopedia of statistical sciences. New York, NY: Wiley, 1982.
23. Suykens JAK, Van GT, De BJ, De MB, Vandewaller J. Least squares support vector machines. World Scientific Publishing, 2002.
24. Sonnenburg S, Räosch G, Schäer C, Schökopf B. Large scale multiple kernel learning. Journal of Machine Learning Research 2006; 7:1531–1565.
25. Suykens JAK, Vandewalle J. Least square support vector machines. IEEE Transactions on Circuits and Systems-I 2000; 47(7):1109–1114.
26. Tay FEH, Cao L. Application of support vector machines in financial time series forecasting. The International Journal of Management Science (Omega) 2001; 29: 309–317.
27. Vapnik V. The nature of statistical learning theory. New York, USA Springer Verlag,1995.
28. Vapnik V. Statistical learning theory. John Wiley and Sons, New York, 1998.
29. Wang Z, Chen S, Sun T. MultiK-MHKS: A novel multiple kernel learning algorithm. IEEE Transactions on Pattern Analysis and Machine Intelligence 2008;30(2):348–353.
30. Wang H Q, Sun F C, Cai Y N, Chen N, Ding L G. On multiple kernel learning methods. Acta Automatica Sinica 2010; 36(8):1037–1050.
31. Ye M Y, Wang X D, Zhang H R. Chaotic time series forecasting using online least squares support Vector machine regression. Acta Physica Sinica 2005; 54(6):2568-2573.
32. Zhao X H, Wang G, Zhao K K, Tan D J. On-line least squares support vector machine algorithm in gas prdecion. Mining Science and Technology 2009; 19:194-198.
33. Zhao Y H, Zhong P, Wang K N. Application of least squares support vector regression based on time series in prediction of gas. Journal of Convergence Information Technology 2011; 6(1):243-250.
34. Zhang W M, Li C X, Zhong B L. LSSVM parameters optimizing and non-linear system prediction based on cross validation. The fifth International Conference on Natural Computation, Aug, 2009.
35. Zhang X R, Hu L Y, Wang Z S. Multiple kernel support vector regression for economic forecasting. International Conference on Management Science & Engineering (17th), November, 2010.
36. Zhang J F, Hu S S. Chaotic time series prediction based on multi-kernel learning support vector regression. ACTA Physica Sinica (Chinese Physics) 2008; 57(5):2708–2713.
37. Zheng XX, Qian F. Based on the support vector machine online modeling and application. Information and Control 2005; (5):636–640.

Associate Prof. Yang-Ming GUO, Ph.D.
Cong-Bao RAN, Master candidate
Xiao-Lei LI, Master candidate
Associate Prof. Jie-Zhong MA
Lu ZHANG, Master candidate
School of Computer Science and Technology
Northwestern Polytechnical University
Youyi West Road 127, Xi'an Shaanxi, 710072
P. R. China
E-mails: yangming_g@nwpu.edu.cn (Y. M. Guo), fengmingr@gmail.com (C. B. Ran), 465628547@qq.com (X. L. Li), majz@nwpu.edu.cn (J. Z. Ma), 956046544@qq.com (L. Zhang)