Peer-reviewed version available at Sensors 2018, 18, 233; doi:10.3390/s18010233

Article

Dynamic Measurement Errors Prediction Model of Sensors Based on NAPSO-SVM

Minlan Jiang [1,*], Lan Jiang [1], Dingde Jiang [2,*], Fei Li [1] and Houbing Song [3]

[1] College of Mathematics, Physics and Information Engineering, Zhejiang Normal University, Jinhua 321004, China;
sa1105587@163.com(L.J.); m18329019792@163.com(F.L.)
[2] School of Astronautics and Aeronautic, University of Electronic Science and Technology of China, Chengdu 611731, China
[3] Department of Electrical, Computer, Software, and Systems Engineering, Embry-Riddle Aeronautical University, Daytona Beach, FL 32114 USA; h.song@ieee.org
* Correspondence: xx99@zjnu.cn(M. Jiang); jiangdd@uestc.edu.cn(D. Jiang); Tel.: +86-138-6797-9259

Abstract: Dynamic measurement error correction is an effective method to improve the sensor precision. Dynamic measurement error prediction is an important part of error correction, support vector machine (SVM) is often used to predicting the dynamic measurement error of sensors. Traditionally, the parameters of SVM were always set by manual, which can not ensure the model's performance. In this paper, a method of SVM based on an improved particle swarm optimization (NAPSO) is proposed to predict the dynamic measurement error of sensors. Natural selection and Simulated annealing are added in PSO to raise the ability to avoid local optimum. To verify the performance of NAPSO-SVM, three types of algorithms are selected to optimize the SVM's parameters, they are the particle swarm optimization algorithm (PSO), the improved PSO optimization algorithm (NAPSO), and the glowworm swarm optimization (GSO). The dynamic measurement error data of two sensors are applied as the test data. The root mean squared error and mean absoluter percentage error are employed to evaluate the prediction models' performances. The experiment results show that the NAPSO-SVM has a better prediction precision and a less prediction errors among the three algorithms, and it is an effective method in predicting dynamic measurement errors of sensors.

Keywords: Sensors; Dynamic measurement errors; Prediction; Improved PSO; Support Vector Machine

1. Introduction

Today, sensors are widely used in the real world, sensor error is on of the key to evaluate the measurement quality of the sensor results. With the development of modern measurement technology, dynamic measurement has gradually become the mainstream of modern measurement.

As an effective theory to improve the measurement accuracy and reduce the measurement error, real-time error correction of sensors have been widely used in the dynamic measurement. Predicting the dynamic measurement error is useful to correct the errors of sensor. Dynamic measurement errors of sensors are difficult to modeling with traditional mathematics cause they has four features[1]: time-varying, randomness, correlation and dynamic. Because its complexity, predicting the dynamic error has been a popular research fields[2-3].

In recent years, several modeling methods are used to predict dynamic error like the gray theory, Bayesian networks and neural network. Every method has its own advantages and drawbacks. Harmonic analysis method is suitable to model the periodic sequences, but it is not suitable for the

random sequence[4]. Bayesian networks is useful for prediction modeling, however, it requires the prior distribution and independent samples, which is difficult to achieve in the real systems[5]. Grey theory model can be constructed by a few samples, but it only depicts a monotonically increasing or decreasing process[6]. Artificial neural network has a good performance of non-linear mapping, however, it has disadvantages, such as over-fitting and easy to falls into a local minimum[7].

Support vector machine (SVM) adopts structural risk minimization to improve generalization ability[8]. It can better solve the problems of nonlinear data and small samples. SVM has been widely applied to solve the problem of function fitting[9]. However, the generalization ability of SVM depends heavily on the appropriate parameters, the model's parameters has huge influence on the precision of the model predictions[10-11]. Thus, many optimization algorithms have been adopted to optimize the SVM parameters, like the particle swarm optimization algorithm, genetic algorithm and glowworm swarm optimization algorithm. There are limitations in these methods, the particle swarm optimization and genetic algorithm fall into the local extremes easily[12-14], the glowworm swarm optimization algorithm has low convergence precision and slow convergence speed[15]. NAPSO algorithm is an improved particle swarm optimization algorithm based on the natural selection strategy and simulated annealing mechanism. these two methods are used to improve the global search ability and convergence speed. In this study, a method of dynamic measurement error prediction for sensors based on NAPSO optimize support vector machine is proposed.

The rest of the paper is organized as follows, in section 2, the overview of SVM algorithm is provided in detail. Then, in section 3, PSO, NAPSO algorithm and the process of Optimization are described briefly. Section 4 reports on a simulation of the dynamic measurement error prediction model. The results of experiments are discussed in section 5. Conclusions are drawn in the last section.

2. SVM Algorithm

2.1. SVM

SVM is a machine learning method based on the statistical learning theory developed in mid-1990s. The basic idea of SVM is that the data of input space R^n are mapped to a high dimensional feature space F by a nonlinear mapping, then finish the linear regression operations in the high dimensional feature space.

For a given training dataset $\{(x_i, y_i), i = 1,2, \quad n\}$, x_i is a n-dimensional input vector and y_i is the corresponding output value, $\phi(x)$ is the nonlinear mapping from input space R^n to high dimensional feature space F.

$$R^n \to F : x \to \phi(x) \tag{1}$$

The regression function of SVM is formulated as follows:

$$f(x) = [\omega \cdot \varphi(x)] + b \qquad \omega \in R^m, b \in R \tag{2}$$

Where ω is the weight vector and b is the threshold, the main goal of the SVM is to find the optimal ω, the optimization equation can be expressed as follows:

$$\begin{aligned} &\min_{\omega,\xi} && \varphi(\omega) = \frac{1}{2}\|\omega\|^2 \\ &\text{s.t.} && |y_i - f(x_i)| \le \varepsilon, \quad \mathrm{i} = 1,2, \quad ,\mathrm{n} \end{aligned} \tag{3}$$

Where ε is a parameter of the insensitive loss function. In practice, two slack variables ξ_i, ξ_i^* and a punishment coefficient C are introduced in the equation (3). According to the risk minimization, equation (3) can be rewritten as the equation (4). The first item of equation (4) is the regularization part, which is used to smooth the function, improves generalization ability. And the second item is an empirical error term. C is the punishment coefficient, which can regulate the balance of the two items.

$$\begin{aligned} \min_{\omega,\xi} \quad & \varphi(\omega,\xi)=\frac{1}{2}\|\omega\|^2+C\sum_{i=1}^{n}(\xi_i+\xi_i^*) \\ \text{s.t.} \quad & y_i-f(x_i)\le\xi_i+\varepsilon \\ & f(x_i)-y_i\le\xi_i^*+\varepsilon \\ & \xi_i,\xi_i^*\ge 0 \\ & (i=1,2,\quad ,n) \end{aligned} \tag{4}$$

Introduce the Lagrange multipliers α_i and α_i^*, then the regression problem can be solved by solving a dual problem as equation(5).

$$\begin{aligned} \max \ & W(\alpha_i,\alpha_i^*)=-\frac{1}{2}\sum_{i,j=1}^{n}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*)K(x_i,x_j) \\ & -\varepsilon\sum_{i=1}^{n}(\alpha_i+\alpha_i^*)+\sum_{i=1}^{n}y_i(\alpha_i-\alpha_i^*) \\ \text{s.t.} \quad & \sum_{i=1}^{n}y_i(\alpha_i-\alpha_i^*)=0 \\ & 0\le\alpha_i,\alpha_i^*\le C,\quad i=1,2,\quad ,n \end{aligned} \tag{5}$$

Where $K(x_i,x_j)$ is the Kernel function. In the last, the SVM regression function is formulated as:

$$f(x)=\sum_{i=1}^{n}(\alpha_i-\alpha_i^*)K(x_i,x_j)+\mathrm{b} \tag{6}$$

2.2. Kernel Function

Kernel function is a key concept of SVM, the performance of SVM mainly depends on the kernel function. As shown in the equation (1), the kernel function establishes a relation between the input space R^n and the high dimensional feature space F. Different selection of kernel functions will construct different regression models.

$$K(x_i,x_j)=\phi(x_i)^T\cdot\phi(x_j) \tag{7}$$

The common kernel functions include the polynomial kernel function, linear kernel function, fourier kernel function and radial basis function (RBF) kernel function. The kernel function parameters has a directly influence on the complexity of the function, RBF kernel function has the advantages of fewer parameters and good performance. Thus, RBF kernel function is used in this paper.

The RBF kernel function is expressed as follows:

$$K(x_i,x_j)=\exp\left\{-\frac{\left|x_i-x_j\right|^2}{2\sigma^2}\right\} \tag{8}$$

Where σ is the width coefficient of the kernel function.

The SVM parameters determine both its generalization ability and learning ability, the punishment coefficient C and RBF kernel function width σ have a directly impact on the accuracy and efficiency of the SVM prediction model. C adjusts the balance between generalization and empirical error. When C is greater, the model's complexity will be increased and it will fall into the "over-fitting" phenomenon easily, if C is too small, the model's complexity will be reduced and it will fall into the "under-fitting" phenomenon easily. The value of σ affects the complexity of the sample data distribution in feature space. In this paper, NAPSO algorithm is used to optimize the two parameters to achieve a better prediction results.

3. SVM Parameters Optimization Based On NAPSO

3.1. PSO

Particle swarm optimization was proposed by Eberhart and Dr. Kennedy in 1995[12], PSO was derived from research on bird flocks' preying behavior. When a flock of birds is looking for food in an area randomly, if there is only one piece of food in the area being searched, the most effective and simple method to find the food is to follow the bird that is closest to the food.

In PSO algorithm, every single solution is a particle in the search space. Each particle has a fitness value, which is determined by an optimization function, each particle has its own velocity and position. The velocity and position of each particle will be changed by the particle best position and global best position. The update equations of the velocity and position are shown by the following expression:

$$v_{i.d}(t+1) = \omega v_{i.d}(t) + c_1 r_1[p_{best} - x_{i.d}(t))] + c_2 r_2[g_{best} - x_{i.d}(t))] \quad (9)$$

$$x_{i.d}(t+1) = x_{i.d}(t) + v_{i.d}(t+1) \quad (10)$$

In the D-dimensional space, t is the iteration number, $v_{i.d}(t)$ is the velocity of particle i at iteration t, $v_{i.d}(t+1)$ is the velocity of particle i at iteration $t+1$, $x_{i.d}(t)$ is the position of particle i at iteration t, $x_{i.d}(t+1)$ is the position of particle i at iteration $t+1$, ω is the inertia weight. c_1 is the cognition learning factor, c_2 is the social learning factor, r_1 and r_2 are random numbers that are uniformly distributed in [0,1], p_{best} is the particle best position for the individual variable of particle i, g_{best} is the global best position variable of the particle swarm.

The initial position and velocity of each particle are randomly generated and will be updated based on the formula (9) and formula (10) until a satisfactory solution is found. In the PSO algorithm, a single particle moves to its p_{best} and g_{best}, each particle's movement generates fast convergence, thus PSO algorithm converges rapidly. However, the fast convergence also makes the update of each particle depend too much on its p_{best} and g_{best}, which makes the algorithm fall into local optimum and premature convergence easily. Therefore, in this paper, an improved PSO algorithm (NAPSO) is used to optimize the parameters of SVM.

3.2. NAPSO

NAPSO algorithm is an improved PSO algorithm based on the methods of natural selection and simulated annealing. In the NAPSO algorithm, the simulated annealing mechanism is used to improve the ability of the algorithm to jump out of a local optimum trap, the natural selection method is employed to accelerate the rate of convergence.

NAPSO algorithm starts with a set of random velocities and positions. Before the iteration, each particle's personal best position and global best position are calculated by the fitness function. Each particles update its velocity and position by the formula (9) and formula (10) at each iteration.

After updating a particle's speed, position l and fitness value $f^{'}$, the particle moves to a random position $l_1^{'}$ in its neighborhood and computes its new fitness value $f_1^{'}$. The movement formula is expressed as follows:

$$l_1^{'} = l + r_3 * [v_{max} - v_{min}] * r_l \quad (11)$$

Where r_3 is the normally distribution random numbers of D-dimension that are distributed in [0,1], r_4 is a random number that is uniformly distributed in [0,1], v_{max} is the maximum value of the velocity, and v_{min} is the minimum value of the velocity.

When $f_1^{'} > g_{best}$, keep the position l. When $f_1^{'} < g_{best}$, if $f_1^{'} < f^{'}$, use the new position $l_1^{'}$ to replace the position l; if $f_1^{'} > f^{'}$, use the new position $l_1^{'}$ to replace the position l by the simulated annealing operation, the operation of simulated annealing is expressed as follows:

$$l = \begin{cases} l & if\ \exp((-1)*(f_l^{'} - f^{'})/T > r4) \\ l_1^{'} & if\ \exp((-1)*(f_l^{'} - f^{'})/T <= r4) \end{cases} \quad (12)$$

Where r_4 is a random number that is uniformly distributed in [0,1], T is the simulated annealing temperature.

Each particle uses the simulated annealing operation to determine whether to accept the new position, and then updates the particle's p_{best} and g_{best} by its position. The simulated annealing operation can significantly enhance the ability of the algorithm to jump out of the local optimum trap. At the end of each iteration, all particles have been ranked by their fitness values, from best to worst, and using the better half to replace the other half. In this way, the stronger adaptability particles are saved. Finally, the NAPSO algorithm is terminated by the satisfaction of a termination criterion.

The pseudo code of the NAPSO algorithm is presented as follows:

Algorithm NAPSO

Input ω, c_1, c_2, T

Output g_{best}

Initialization: x, p_{best}, g_{best}

while t<maximum number of iterations and g_{best} >minimum fitness **do**

for each particle **do**

update the velocity v, position l, and fitness $f^{'}$

find a new position $l_1^{'}$ in the neighborhood and Calculate its fitness value $f_1^{'}$

if1 ($f_1^{'} < g_{best}$) **then**

if2 ($f_1^{'} - f^{'} < 0$) **then**

accept the new position $l_1^{'}$

else if2

accept the new position $l_1^{'}$ by the simulated annealing operation

end if2

else if1

accept the old position l

end if1

update the p_{best}, g_{best} and Simulated temperature T

end for

rank all particles by their fitness value, use the better half to replace the other half.

t=t+1

end while

return the g_{best}

The simulated annealing operation will slow the rate of convergence, thus increasing the convergence time. The natural selection operation will reduce the sample diversity of samples. However, these two operations can compensate for each other, the simulated annealing operation can increase sample diversity, and the natural selection operation can speed up the convergence rate. These two operations are used to both ensure the convergence rate of the algorithm and guarantee that the ability of the algorithm to jump out of the local optimal trap can be enhanced.

3.3. Otimization Process

The NAPSO algorithm is applied to optimize the SVM parameters C and σ as follows:

Step 1: Initialize the NAPSO algorithm, set the number of particles velocity, particles positions and the other parameters. Because the search space is 2 dimensional, the position of each particle contains two variables. Set T to be the simulated temperature; the initial T is 5000°C, and the

lower limit of T is 1°C. Calculate the fitness value of each particle. The fitness evaluation function is defined as follows:

$$J = \sum_{i=1}^{n} (Y_i - Y_i^{'})^2 / n \tag{13}$$

Where Y_i is the actual value, $Y_i^{'}$ is the predicted value and n is the number of the training samples.

Step 2: According to the fitness value of each particle to set the personal best position p_{best} and global best position g_{best}.

Step 3: Update the position l and velocity of each particle. Evaluate the fitness value $f^{'}$.Then, randomly find a new position $l_1^{'}$ in the neighborhood of the particle, calculate the new fitness value ($f_1^{'}$) of the new position.

Step 4: Calculate the difference between the fitness value $f^{'}$ and the new fitness value $f_1^{'}$, $\Delta f = f_1^{'} - f^{'}$.

Step 5: When $f_1^{'} >= g_{best}$, keep the original position l. When $\Delta f > 0$ and $f_1^{'} < g_{best}$, according the formula (12) to accept the new position $l_1^{'}$, if $\Delta f < 0$ and $f_1^{'} < g_{best}$, replace the original position with the new position. Then, update the p_{best} and g_{best}.

Step 6: When the updates of each particle has completed, then rank all of the particles according to the each particle's fitness value, employ the better half particles' information to replace the other half particles' information and update the temperature $T = T * 0.9$.

Step 7: If the termination conditions are satisfied, output the two variables of the g_{best}; Otherwise, return to Step 2.

4. Experiments

4.1. Data Description

In this paper, two cases have been considered to illustrate the effectiveness of the proposed method. The data of case 1 is the dynamic error sequence, which is derived from the measuring error of the angular instrument with anticlockwise rotation (speed 2r/min) based on standard value interpolation under room temperature, the error sequence contains a total of 240 samples. In case 2, the measuring error sequence of the length grating contains a total 141 samples. The process of collecting data is expressed as follows: the measurement range is 500mm and the sample interval is 25mm, the computer receive the actual displacement from the laser interferometer and the measuring displacement from the length grating. The difference of the two data is the dynamic measurement error of the length grating.

4.2. Preprocessing

The two datasets both are one-dimensional data, in order to achieve the better predict results and get more information from the data, these two one-dimensional data must be converted to multi-dimensional data[16]. Assuming p is the dimension of the input vector, the reconstructed samples are listed in Table 1.

According to the reconstructed method listed in the Table 1, in case 1,the dimension number p is 16, the number of restructured sample is 224, selecting the first 124 samples for training and the final 100 samples for testing. The proportion of training samples to testing samples is 1.24:1, in case 2, the dimension p is 12, the number of restructured sample is 129, the first 100 samples are selected as training data and the rest are used as testing data. The proportion of training samples to testing samples is 3.44:1.

Table 1. Reconstructed samples.

Input	Output
$X(1), X(2), \quad , X(p)$	$X(p+1)$
$X(2), X(3), \quad , X(p+1)$	$X(p+2)$
...	...
$X(n-p), X(n-p+1), \quad , X(n-1)$	$X(n)$

Preprocess the data by the normalized method, then perform parameter optimization and train the model.

4.3. Valuation Index

To further evaluate the prediction of the NAPSO-SVM model, the root mean square error (RMSE) and mean absolute percent error(MAPE) are used as evaluation indices. The definition of MAPE and RMSE are expressed as follows:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - Y_i^{'})^2} \tag{14}$$

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{Y_i - Y_i^{'}}{Y_i}\right| \tag{15}$$

Where Y_i is the actual value, $Y_i^{'}$ is the prediction value and n is the number of the training sample.

Using the NAPSO algorithm to determine the punishment coefficient C and RBF kernel function width σ. The SVM model is built based on the training samples and optimal parameters. To show the performance of the proposed method, the particles swarm optimization and glowworm swarm optimization are also implemented.

5. Results

In case 1, the prediction results of three models are shown in Figures. 1, 2 and 3, respectively.

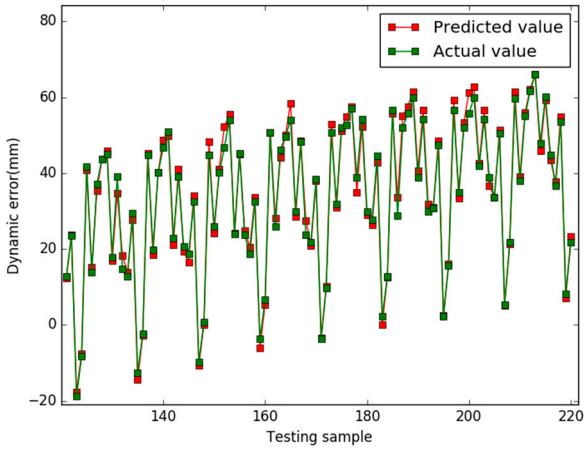

Figure 1. Predicted results of the NAPSO-SVM (case1)

To make a fair comparison, the maximum generation, population size, minimum fitness value, range of gains, dimension of search space and initial positions are identical for all the algorithms. The maximum number of generations is 100, the minimum fitness value is 0.1, the size of the population is 100, and the dimension of the search space is 2. The parameters for NAPSO were set as follows: the inertia weight $w = 0.9$, the acceleration constant c_1 and c_2 are 2, the initial temperature is 10000℃, the lower limit of temperature is 1℃, the maximum value of velocity is 1, the minimum value of velocity is -1. In GSO algorithm, the light absorption coefficient is 50, the minimum value of

attractiveness is 0.8, the maximum value of attractiveness is 1.0, the value of initial step size factor is 0.5. The PSO algorithm has the same inertia weight and acceleration constant as the NAPSO algorithm.

Figure.4 presents the comparison results of predicted residuals by the three models. The MAPE value and RMSE value of the three models are listed in Table 2.

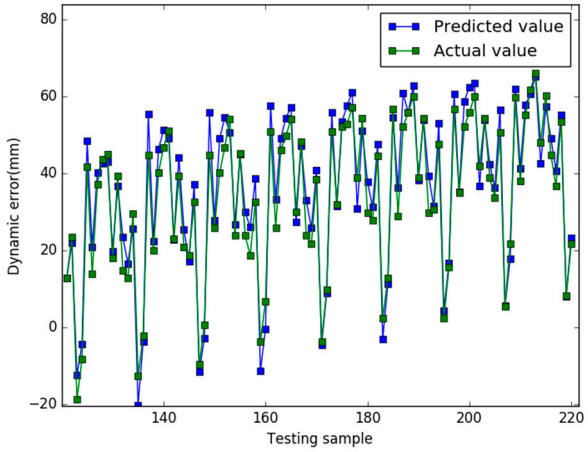

Figure 2. Predicted results of the PSO-SVM (case1)

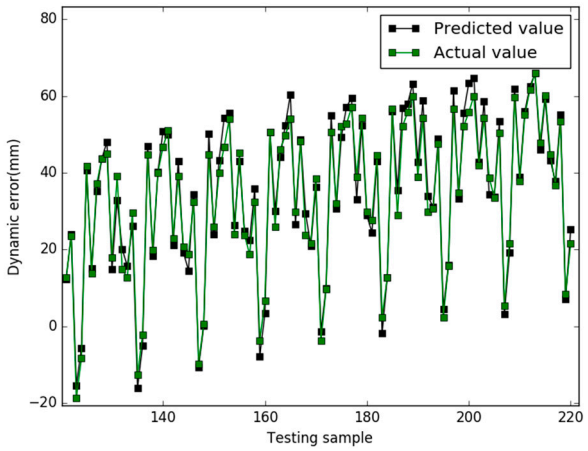

Figure 3. Predicted results of the GSO-SVM (case1)

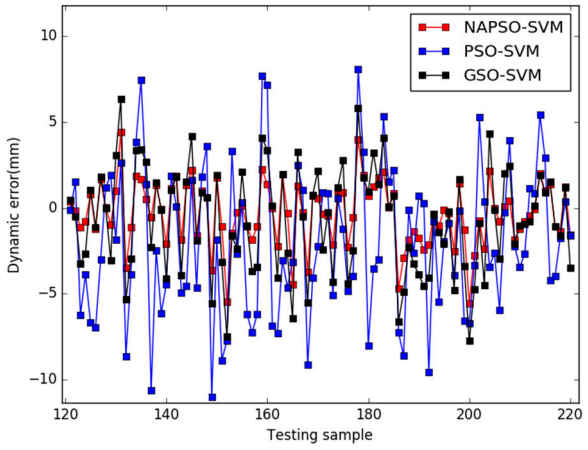

Figure 4. Comparison of three models for predicted residuals (case 1)

By comparing Figures. 1-3, we find that the NAPSO-SVM model outperforms the PSO-SVM and GSO-SVM model. The prediction performance of NAPSO-SVM is better than GSO-SVM model and accuracy much better than PSO-SVM.

The residual curves of the three models are shown in the Figure. 4, The prediction residual curve of the PSO-SVM model is large, ranging from -11 to $8^{''}$, and the prediction residual of the GSO-SVM model is smaller than the PSO-SVM model. But it is still relatively large, ranging from -8 to $6^{''}$. The predicted residual of the NAPSO-SVM is smaller than the others and tends to more gentle, ranging from -5 to $4^{''}$.The results prove that dynamic measurement error prediction ability of NAPSO-SVM model is better than PSO-SVM and GSO-SVM model, and the NAPSO algorithm is an effective method for parameters optimization.

To further verify the ability of the three models. Table 2 lists the comparison results between the three models for prediction accuracy indexes.

Table 2. Comparison of the index value among the three models (case 1).

MODEL	MAPE	RMSE
NAPSO-SVM	0.0744	0.1879
PSO-SVM	0.2423	0.4710
GSO-SVM	0.1493	0.3128

In Table 2, the MAPE value and RMSE value of the NAPSO-SVM model are smaller than the PSO-SVM and GSO-SVM model. The MAPE value is approximately 0.0744 for NAPSO-SVM model compared with approximately 0.2423 and 0.1493 for the PSO-SVM and GSO-SVM model, respectively. Furthermore, the RMSE value is 0.1876 in the case of NAPSO-SVM model. Compared with the NAPSO-SVM model, the RMSE value of the GSO-SVM model and PSO-SVM model are 0.4710 and 0.3128 respectively. In summary, the results of the Table 2 are accorded with the Figure. 4, the NAPSO-SVM model has the best dynamic measurement error prediction ability among the three methods.

In case 2, the parameters of each algorithm are essentially the same as the previous case, the prediction results of three models are shown in Figures.5, 6 and 7. Figure.8 shows the comparison results of predicted residuals by three models. The MAPE value and RMSE value of the three models are listed in the Table 3.

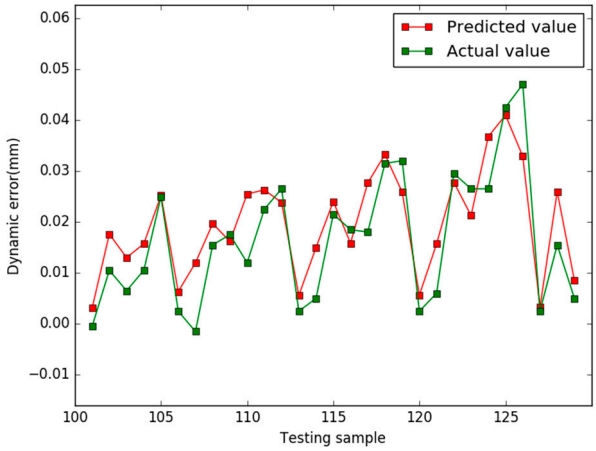

Figure 5. Predicted results of the NAPSO-SVM (case 2)

In Figures.5-7, when the ratio of training samples and testing samples is approximately 3.5, the prediction curve of the NAPSO-SVM model is closest to the actual value curve, and the prediction curve of the NAPSO-SVM model is approximately the same as the actual value curve. However, unlike the case 1, The prediction results of the PSO-SVM model is better than the PSO-SVM model, but the prediction curves of these two models still lag behind the actual value curve.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 20 November 2017 doi:10.20944/preprints201711.0132.v1

Peer-reviewed version available at Sensors 2018, 18, 233; doi:10.3390/s18010233

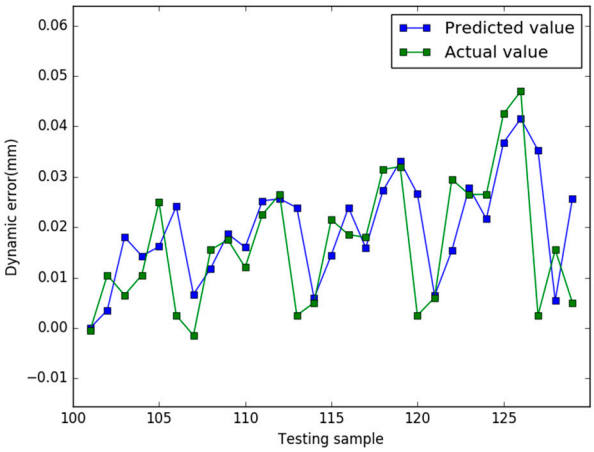

Figure 6. Predicted results of the PSO-SVM (case 2)

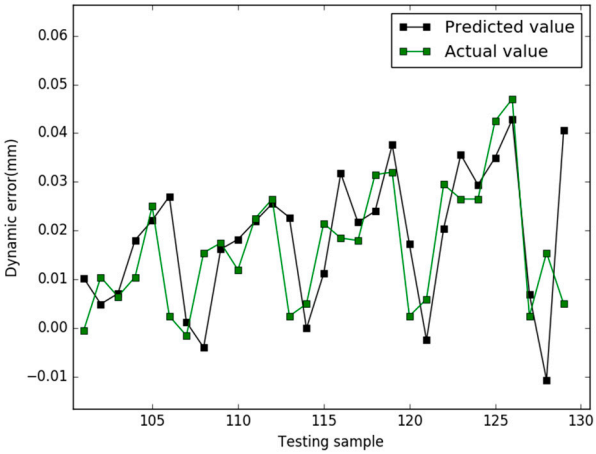

Figure 7. Predicted results of the GSO-SVM (case2)

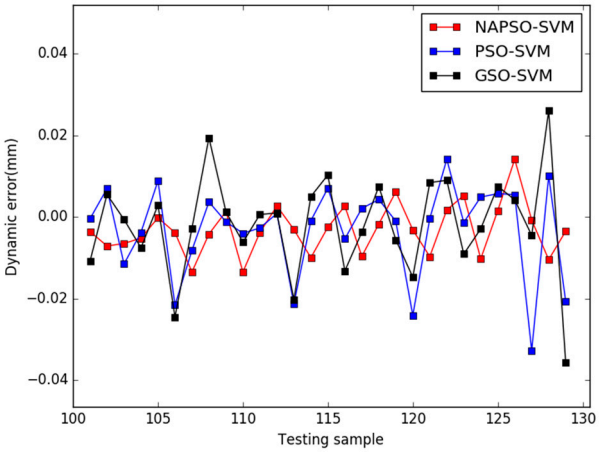

Figure 8. Comparison of the predicted residuals of the three models(case2)

In Figure.8, the prediction residual of the NAPSO-SVM model is smallest among the three models, ranging from -0.013 to 0.014 mm. The prediction residual of the GSO-SVM model is ranging from -0.035 to 0.026 mm, and the prediction residual of the PSO-SVM model is ranging from -0.032 to 0.013 mm.

Table 3. Comparison of the index value among the three models (case 2).

MODEL	MAPE	RMSE
NAPSO-SVM	1.0833	0.0013
PSO-SVM	1.9714	0.0021
GSO-SVM	2.2948	0.0023

As Table 3 shows, the MAPE value is approximately 1.0833 for the NAPSO-SVM model compared with approximately 1.9714 and 2.2948 for the PSO-SVM model and GSO-SVM model. NAPSO-SVM model has the smallest RMSE value of the three algorithms, acquiring RMSE value of 0.0013 and the RMSE values of the PSO-SVM model and GSO-SVM model are 0.0021 and 0.0023, respectively. The prediction ability of the NAPSO-SVM model is clearly better than the other models, and GSO-SVM model has the worst performance.

The results of the two cases show that the NAPSO-SVM model has the best prediction accuracy among the three methods. This indicate that the NAPSO algorithm has the better capability of global search than the other two algorithm, the reason is that the updating of the position and velocity of the particles in the PSO algorithm are dependent too much on current best particle. Compared with the PSO algorithm, the NAPSO algorithm uses the simulated annealing and natural selection mechanism, it is easier jump out of the local trap and search the global optimal solution in the global space.

6. Conclusions

Dynamic measurement has been a hot area of research for several years, and dynamic measurement error prediction is an useful method to improve the sensor measurement accuracy. In this study, a method of dynamic measurement error prediction based on NAPSO-optimized SVM parameters is proposed. To improve the prediction accuracy, the NAPSO algorithm is used to optimize the SVM parameters to avoid the problems of "over-fitting" and "under-fitting" of SVM. The results of the two cases show that compared with the PSO-SVM and GSO-SVM model, the NAPSO-SVM model has the better prediction accuracy. The proposed method provides a new way for predicting the sensor's dynamic measurement error and has definite value for application in dynamic measurement. However, like the standard PSO, NAPSO has the intrinsic property randomness. In the future, we plan to study which is the more effective method for improving the prediction results.

Acknowledgments: The work was supported in part by the National Natural Science Foundation of China (Nos. 51305407, 61571104).

Author Contributions:. M.J. and L. J. were the main designers of the proposed scheme and wrote the paper. F.L. focused on the research, and part of the paper writing, D.J. and H.S. revised the paper and analyzed the results.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Yakovlev, V.T. Method of Determining dynamic Measurement Error Due to Oscillographs. *Measurement Techniques* **1987**, *30*, 331-334.
2. Chen, Y. Area-Efficient Fixed-Width Squarer with Dynamic Error-Compensation Circuit. *IEEE Transactions on Circuits and Systems II: Express Briefs* **2015**, *62*, 851-855.
3. Yakovlev, V.T.; Collins, E.; Flores, K.; Pershad, P.; Stemkovski, M.; Stephenson, L. Statistical Error Model Comparison for Logistic Growth of Green Algae (Raphidocelis subcapitata). *Applied Mathematics Letters* **2017**, *64*, 213-222.
4. Trapero, J.; Kourentzes, N.; Martin, A. Short-term solar irradiation forecasting based on dynamic harmonic regression. *Energy* **2015**, *84*, 289-295.
5. Yang, H.; Zhang, L.; Zhou, J.; Fei, Y.; Peng, D. Modelling of dynamic measurement error for parasitic time grating sensor based on Bayesian principle. *Guangxue Jingmi Gongcheng/Optics and Precision Engineering* **2015**, *24*, 2523-2531.

6. Ge, L.; Zhao W.; Zhao, S; Zhou, J. Novel error prediction method of dynamic measurement lacking information. *Journal of Testing and Evaluation* **2012**, *40*, n1.
7. Barbounis, T.G.; Theocharis, J.B. A locally recurrent fuzzy neural network with application to the wind speed prediction using spatial correlation. *Neurocomputing* **2007**, *70*, 1525-1542.
8. Sasikala, S.; Balamurugan, S.; Geetha, S. A Novel Memetic Algorithm for Discovering Knowledge in Binary and Multi Class Predictions Based on Support Vector. *Machine.Applied Soft Computing Journal* **2016**, *49*, 407-422.
9. Malvoni M.; De G.; M, G.; Photovoltaic Forecast Based on Hybrid PCA–LSSVM Using Dimensionality Reduced Data. *Neurocomputing* **2016**, *211*, 72-83.
10. Wang X.; Kang M.; Fu X.; Li C. Predictive Modeling of Surface Roughness in Lenses Precision Turning Using Regression and Support Vector Machines. *International Journal of Advanced Manufacturing Technology* **2016**, *87*, 1273-1281.
11. Suganyadevi, M.; Babulal, C. Support Vector Regression Model for the Prediction of Loadability Margin of a Power System. *Applied Soft Computing* **2014**, *24*, 304-315.
12. Kennedy J.; Eberhart R. Particle Swarm Optimization. In Proceedings of the First IEEE International Conference on Neural Networks, Perth, Australia,1995; pp, 1942-1948.
13. Liu Y. Study on an Improved PSO Algorithm and its Application for Solving Function Problem. *International Journal of Smart Home* **2016**, *10*, 51-62.
14. Zhong Y.; Ning J.; Zhang H. Multi-agent Simulated Annealing Algorithm based on Particle Swarm Optimisation Algorithm. *International Journal of Computer Applications in Technology* **2012**, *43*, 335-342.
15. Zainal N.; Zain A.; Radzi N.; Glowworm Swarm Optimization (GSO) for optimization of machining parameters. *Journal of Intelligent Manufacturing* **2016**, *27*, 998-1006.
16. Jiang M.; Luo J.; Jiang D.; Xiong J.; Song H.; Shen J. A Cuckoo Search-support Vector Machine Model for Predicting Dynamic Measurement Errors of Sensors. *IEEE Access* **2016**, *PP*, 99.