Adaptive Prediction Algorithm Based on ARMA and RBFNN Models for Radar Performance

Jie Wu\(^1\)^2 and Yongle Lv\(^1\)

\(^1\) Nanjing Research Institute of Electronics Technology, Nanjing, PR China
\(^2\) Corresponding author
Email: chiying188@163.com

Abstract. To implement health management for large complex systems, it is of great importance to monitor health online and track the trend of working performance. Fault prediction is a relatively weak but critical part in Prognostics and Health Management. When selecting characteristic parameter time series to monitor and predict the health of the system, applicability of a single model is limited. To satisfy the requirement of real-time performance and accuracy, an algorithm based on Autoregressive Moving Average Model and Radial Basis Function Neural Network Model was promoted, updating the model to predict automatically when the current model is no longer applicable. Multi-threaded parallel modeling strategy can be used in engineering application. The simulation results show that compared with algorithm employing the single model, the adaptive algorithm can effectively improve the real-time performance and accuracy.

1. Introduction

Large complex systems represented by radar are playing increasingly important roles in national defense and military, and the demand for their working reliability is getting higher. It is necessary to implement health management [1] for radar. Fault prediction is a significant functional part in Prognostics and Health Management (PHM). To make it convenient for users to discover and solve the problems in time when working performance declines, carrying out preventive maintenance, tracking and predicting the trend of performance degradation count. So that function of radar can be guaranteed, and negative effects caused by shutdown can also be avoided. Characteristic parameter index which characterizes health of radar is usually selected for modeling and predicting.

A time series [2] is a series of observation sample data collected in chronological order. In the following, the time series of the parameters obtained by monitoring the state of radar is referred to as the observation sequence, and has the following characteristics: Trend-performance degradation of radar components caused by corrosion and aging leads to the slowly varying trend of the observation sequence; Randomicity-there are complex interactions between radar system components, and internal uncertainty also leads to some randomness of the observation sequence. In addition, the observation sequence is affected by electromagnetic interference, measurement noise, human control and other factors, often includes noise and outliers. In order to facilitate the discussion, the observation sequence referred to in this paper is the sequence after eliminating outliers and noise. Some methods can be used to pretreat such as singular value decomposition filtering algorithm [3], which are not the focus of this paper.

The basic method of fault prediction based on observation sequences is as follows: The modeling obtains the dependence between the monitored values of the radar in the normal working state and predicts the future value based on the built model. If the monitored values do not conform to the
statistical rules expressed by the model, the system is judged as failures. Traditional fault prediction methods such as polynomial fitting method can only fit the trend components of the observation sequence, and the modeling accuracy is low. In order to improve the prediction accuracy, the random components of the observation sequence should be modeled. Common modeling methods include linear methods such as autoregressive moving average model [4] and nonlinear methods such as neural network models [5].

At different working stages, the stress levels of radar systems are different, and the statistical rules of observation sequence are not fixed. The applicability and accuracy of prediction based on a single model are limited. The implementation of online health management of radar requires models with high modeling speed and accuracy, which can automatically and intelligently predict the performance parameters without human intervention. Aiming at this requirement, an algorithm which adaptively builds appropriate models in different working states was promoted. As online modeling and prediction are required to be real-time and accurate, simple and efficient Autoregressive Moving Average Model with high modeling speed and Radial Basis Function Neural Network Model with simple structure and high precision are selected. The prediction model automatically updates when it is no longer applicable.

2. ARMA and RBFNN Models

2.1. Autoregressive Moving and Average Model

Autoregressive Moving Average (ARMA) Model is widely used in time series prediction, which is expressed with a combined form of Autoregressive Model (AR) and Moving Average Model (MA), usually marked as $ARMA(p,q)$, $p$ and $q$ are orders of ARMA model, which can be expressed as:

$$Y_t = \phi_1 Y_{t-1} + \cdots + \phi_p Y_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \cdots - \theta_q \varepsilon_{t-q}, t \in N$$

(1)

The autoregressive coefficients $\phi_1, \phi_2, \ldots, \phi_p$ establishes the relation between the observation value at the current moment and the values at the previous $p$ points, while the moving average coefficients are $\theta_1, \theta_2, \ldots, \theta_q$. $\{\varepsilon_t\}$ represents independent identically distributed random series.

Single-step predicted value based on ARMA model is given by:

$$\hat{Y}_{t+1} = \hat{\phi}_1 Y_t + \cdots + \hat{\phi}_p Y_{t-p+1} - \hat{\theta}_1 \varepsilon_t - \cdots - \hat{\theta}_q \varepsilon_{t-q+1}, t \in N$$

(2)

Where $\hat{\phi}_1, \hat{\phi}_2, \ldots, \hat{\phi}_p, \hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_q$ are estimated values of the parameters obtained by modeling.

Traditional Box-Jenkins method employs autocorrelation and partial autocorrelation functions of data to determine orders of model, which is greatly affected by subjective factors and is not applicable for engineering. Commonly used order criteria are white noise test criteria, minimum error square sum criteria, information criteria such as AIC and BIC. The BIC values are as follows [6]:

$$BIC_c = \ln(\hat{\sigma}_e^2) + r \ln(n) / n + c$$

(3)

Where $r$ is the number of parameters to be determined, $\hat{\sigma}_e^2$ is the maximum likelihood estimate of the variance of the modeling errors of the sequence, $n$ is the length of the sample sequence for modeling, and $c$ is a constant. The order corresponding to the minimum criterion value is considered to be the optimal order.

Maximum likelihood method, least square method is commonly employed to estimate the model parameters. The multi-step prediction of the model can be iteratively obtained from the one-step prediction.

The steps for building ARMA model and predicting are as follows:

Step1: Pretreat the observation series;

Step2: Check stationarity, employ the difference operation repeatedly if non-stationary until the series is stationary; otherwise, go to Step3;
Step3: Zero mean normalize the series;  
Step4: Aiming at the zero-mean differential series, information criteria can be applied to determine orders of $p$ and $q$.  
Step5: Estimate the parameters of ARMA model: $\phi_1, \phi_2, \ldots, \phi_p, \theta_1, \theta_2, \ldots, \theta_q$;  
Step6: Carry out one-step or multi-step prediction based on the built ARMA model, and revert mean and difference.

2.2. Radial Basis Function Neural Network Model
Radial Basis Function Neural Network (RBFNN) is a simple and effective neural network consisting of input layer, hidden layer and output layer. It has rapid speed of convergence, and can approximate any nonlinear functions at any precision. The weight from input layer to hidden layer is 1. The radial basis function of hidden layer space is a series of centrosymmetric nonlinear functions. The input vectors are locally approximated and the output layer neurons outputs.

The numbers of input layer nodes and hidden layer nodes affect the prediction performance of the network. If the number of nodes is too small, the performance may be poor. With the increase of the number of nodes, the modeling accuracy is getting higher but the network is computationally expensive and the structure is complicated. In addition, the phenomenon of “overfitting” easily occurs. Common methods for determining the number of input layer nodes are: false nearest neighbor method [7], saturation correlation dimension method [8] and singular value decomposition. The traditional methods for determining the number of hidden layer nodes mainly include “mean square error threshold method” and “leave-one-out method” [9]. The selection of activation function also affects the prediction performance of the network. In this paper, Partial autocorrelation method [3] is employed to determine the number of input layer nodes, and gaussian function is employed as a hidden layer activation function:

$$\varphi_j(\|\vec{Y}_t - c_j\|) = \exp\left(\frac{-\|\vec{Y}_t - c_j\|^2}{2b_j^2}\right)$$

Where $\vec{Y}_t = \{Y_{t-k+1}, Y_{t-k+2}, \ldots, Y_t\}$ is the input vector, $c_j$ is the center point of the activation function, $b_j$ is the width parameter.

$M$ is the number of hidden layer nodes, then the output:

$$\hat{Y}_{t+1} = \sum_{j=1}^{M} \omega_j \varphi_j(\|\vec{Y}_t - c_j\|)$$

The diagram of RBFNN model is shown in figure 1.

![Diagram of RBFNN model](image-url)
3. Adaptive Prediction Algorithm

3.1. Algorithm Thinking

When modeling and analyzing the observation sequence, stationarity hypothesis is the most important hypothesis when making statistics, inference and prediction based on monitoring records. When all the statistical properties do not change over time, the series is strictly stationary. Actually the constraint conditions are usually relaxed, if the mean value and the variance stay the same, the process is called stationary. This paper hypothesizes that non-stationary sequence can be considered to consist of a series of stationary sequences.

The algorithm studied in this paper includes linear and nonlinear models. Linear and nonlinear modeling methods have their own advantages and disadvantages. Linear models are simpler and of which the modeling time are shorter. When modeling, linear methods need smaller sample size but the prediction accuracy is lower than nonlinear methods. While nonlinear methods are more complicated. The algorithm employs linear models to predict at first, when the model is no longer applicable, the sequence may contain nonlinear components, making it difficult for the linear model to accurately model the sequence. Then nonlinear models are employed to continue predicting instead. Otherwise, the prediction results may be inaccurate. In engineering, it is preferable to choose as few samples as possible so that the modeling does not rely too much on historical data, maintaining the real-time applicability of the model.

3.2. Algorithm Flow

An adaptive prediction algorithm based on ARMA and RBFNN models was put forward, rectifying models to predict adaptively according to model evaluation and characteristics of the sequence. The method evaluates the quality and prediction effect of newly built model timely, and automatically updates the current prediction model. The flow chart of the algorithm is shown in figure 2.

![Flow chart of adaptive prediction algorithm](image-url)

Figure 2. Flow chart of adaptive prediction algorithm.
ARMA and RBFNN models are chosen to predict, starting with relatively simpler ARMA model. Corresponding constraint conditions are set for each modeling time, so that the algorithm prefers selecting modeling and prediction method which is simpler and has less parameter within the acceptable range of error. When ARMA modeling error is unacceptable, the more complicated and accurate RBFNN modeling process starts. The modeling time is recorded by variable counter. The error of modeling and prediction is estimated by mean square error (MSE):

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (Y_i' - Y_i)^2$$  \hspace{1cm} (6)

Where $N$ is sample size, $Y_i'$ is predicted value, $Y_i$ is observation value.

**Step1:** Monitor the pretreated observation sequence online, initialize counter to 1, set values for modeling evaluation threshold $thr1$ and modeling rectifying threshold $thr2$;

**Step2:** The process starts with ARMA modeling, and evaluate the prediction effect of the built model. Check if $MSE < thr1$. If satisfied, the current model is adopted to predict. Otherwise, the next-time ARMA modeling process starts and update counter by adding 1. Repeat **Step2** until model is adopted or the modeling times reach the maximum $counter\_max$, initialize counter to 1;

**Step3:** When the modeling times reach the maximum and the modeling error is still greater than $thr1$, the RBFNN modeling process starts. The process lasts until model is adopted or the modeling times reach the maximum $counter\_max$, initialize counter to 1;

**Step4:** After adopting the model, evaluate $MSE$ of the prediction continuously. Checked if $MSE < thr2$. If satisfied, then continue predicting; otherwise, repeat **Step2** to **Step4**.

4. Engineering Application Strategy

In engineering applications, the working performance of radar can be reflected by the observation sequence. By continuously monitoring and predicting the monitored data, users know the declining trend of the performance of the radar in time and take measures as soon as possible to avoid negative impact on the normal operation of the radar.

The error of predicting accumulates to a certain extent over time, and new models need to be built. When the modeling time is short, a single thread can be used to predict and update the model. When the modeling time is greater than the sampling interval of the sequence, which means if the modeling time is long, a single thread will affect the timeliness of subsequent data point prediction. Two-threaded or multi-threaded parallel method can solve this problem.

Take two-thread as an example, the observation sequences are modeled in thread 1, the main thread. Alternative models are built based on historical data at intervals in thread 2, and when the current model in thread 1 is no longer applicable, the newest model built in thread 2 will be adopted to continue predicting in the main thread, modifying the prediction model of the main thread with minimum time cost. Considering multi-thread, different modeling cycles can be set in each thread.

![Figure 3. Diagram of multi-threaded strategy.](image-url)
5. Experiment and Result Analysis
The transmitting power directly influences radar range and is an important parameter related to radar performance. The experimental series of radar’s transmitting power was acquired with the period of 20 seconds. Singular value decomposition filtering method was employed to filter the series of which the outliers had been eliminated.

The model evaluation threshold $thr1$ is set as 2‰ of the variance of the observation sequence, which is 0.016. In the stage of model evaluation, the degree of fitting is relatively high and the error is small. The evaluation condition is appropriately relaxed in the predicting period. The model rectification threshold $thr2$ is set 1.5 times the $thr1$, which is 0.024. The number of input layer nodes is 3. The modeling process employs the latest historic observation series with the length of 250 to build the model, the former 200 for modeling and the latter 50 for evaluation. The prediction starts at time point 251. The constraint conditions during the modeling process are shown in table 1. The observation sequence, denoted by $\{S_i\}_{i=0}^{\infty}$, had been normalized.

The adaptive prediction process is shown in figure 4 and table 2.

![Figure 4. Adaptive piecewise modeling](image)

Table 1. Constraint conditions during modeling

| Modeling times | ARMA model order | Number of RBFNN hidden layer nodes(X) |
|----------------|------------------|-------------------------------------|
|                | $p_{\text{max}}$ | $q_{\text{max}}$ | min | max |
| 1st            | 3                | 0                     | 5   | 10  |
| 2nd            | 5                | 0                     | 5   | 30  |
| 3rd            | 5                | 5                     | 5   | 50  |

Table 2. Record of modeling and prediction process.

| Time point | Newly-built model | $MSE$          | Adopted |
|------------|-------------------|----------------|---------|
| 251        | ARMA(3,0)         | $0.01752 > thr1$ | No     |
| 251        | ARMA(3,0)         | $0.01752 > thr1$ | No     |
| 251        | ARMA(3,0)         | $0.01752 > thr1$ | No     |
| 251        | RBFNN(X=10)      | $0.01595 < thr1$ | Yes    |
| 463        | ARMA(3,0)         | $0.01649 > thr1$ | No     |
| 463        | ARMA(4,0)         | $0.01578 < thr1$ | Yes    |

In table 2, an ARMA (3,0) model of which the order $p=3,q=0$, was built at time point 251. The autoregressive coefficients are 0.6369,-0.0411, 0.2217. Another ARMA (3,0) model, of which the
order $p=3$, $q=0$, was built at time point 463. The autoregressive coefficients are 0.6369, -0.0411, 0.2217. The ARMA (4, 0) model with the autoregressive coefficients of 1.4026, -1.0904, 0.9855, -0.3892 was built later.

On the basis of the simulation results, the adaptive algorithm built ARMA models and RBFNN model adaptively according to the errors of evaluation and prediction. At first, all the ARMA models built in 3 times of ARMA modeling process did not meet the accuracy requirement, and then RBFNN modeling process started. The RBFNN model of which the number of hidden layer nodes is 10 met the prediction accuracy requirement, and the RBFNN model was adopted to start predicting from time point 251. At time point 462, the model failed because the prediction error was greater than modeling rectifying threshold $thr2$. The newly adopted ARMA model started to predict from time point 463.

The single ARMA prediction method employed the ARMA (3, 3) model built by the first 200 sampling data. Its autoregressive coefficient are -0.0576, 0.3853, 0.4698, moving average coefficients are 0.6866, -0.0071, -0.3552.

| Table 3. MSE of 3 different prediction methods. |
|-----------------------------------------------|
| Prediction methods   | Adaptive | Single ARMA model | Single RBFNN model |
| MSE of prediction process | 0.01632  | 0.04127           | 0.01836            |

The comparison in figure 5 and table 3 shows that adaptive prediction has the highest prediction accuracy, followed by RBFNN model and ARMA model with the lowest prediction accuracy.
Compared with the ARMA model, the prediction accuracy of RBFNN model is improved by 55.51% and the prediction accuracy of adaptive algorithm is improved by 60.46%.

### 6. Conclusion

Consider that a single model may no longer meet the prediction accuracy requirement over time, in order to ensure the real-time performance and accuracy of the health of systems when monitoring online, an adaptive modeling and prediction algorithm based on ARMA models and RBFNN models was put forward, expanding the range of applicable models and improving the prediction accuracy. When applied to practical engineering, the effects caused by the time of modeling and evaluating are reduced. Also, users do not have to care about the internal modeling process. Only the model evaluation threshold $thr_1$ and model rectifying threshold $thr_2$ need to be adjusted to meet the specific requirement of engineering. Simulation results have proved effectiveness and accuracy of the algorithm.

### 7. Acknowledgments

This work was supported by a grant from National Defense Basic Scientific Research during the 13th Five-Year Plan Period of China (No. JCKY2017210B001). Also, the work was supported by a grant from pre-research projects of Central Military Commission Equipment Development Department of China (No. 51317050202).

### 8. References

[1] Shengkui Zeng, Pecht M G and Wu Ji 2005 Status and Perspectives of Prognostics and Health Management Techonology *Acta Aeronaut. Astronaut. Sin.* **26** (5) 626 - 32.

[2] Jonathan D. Cryer and Kung-Sik Chan 2011 *Time Series Analysis with Application in R* (Beijing: China Machine Press).

[3] Yongle Lv. 2009 Methods of Prediction Model Buildings for Airborne Equipment’s Working Performance and Applications *Beijing University of Astronautics and Aeronautics*.

[4] Box G. P. E, Jenkins G. M. and Reinsel G. C. 2005 *Time Series Analysis: Forecasting and Control 3rd Edition. Pearson Education Asia Ltd*.

[5] Buhmann M D. 2003 Radial Basis Functions: Theory and Implementation (Cambridge Univ. Press) pp 56 - 104.

[6] Olivier C, Colot O and Courtellemont P 1994 Information Criteria for Modeling and Identification. IEEE 1813-18.

[7] Matthew B. K, Reggie B and Henry D. I. A. 1992 Determining Embedding Dimension for Phase-Space Reconstruction Using A Geometrical Construction *Phys. Rev. A* **45** (6).

[8] Grassberger P and Procaccia I. 1983 Measuring the Strangeness of Strange Attractors *Physica D* **9** 189-208.

[9] Ghosh J and Nag A 2001 An Overview of Radial Basis Function Networks *Physica-Verlag HD* **67** 1-36.