Robust Neural Predictor for Noisy Chaotic Time Series Prediction

Min Han and Xinying Wang

Abstract—A robust neural predictor is designed for noisy chaotic time series prediction in this paper. The main idea is based on the consideration of the bounded uncertainty in predictor input, and it is a typical Errors-in-Variables problem. The robust design is based on the linear-in-parameters ESN (Echo State Network) model. By minimizing the worst-case residual induced by the bounded perturbations in the echo state variables, the robust predictor is obtained in coping with the uncertainty in the noisy time series. In the experiment, the classical Mackey-Glass 84-step benchmark prediction task is investigated. The prediction performance is studied for the nominal and robust design of ESN predictors.

I. INTRODUCTION

TIME series analysis and prediction is an important problem encountered in many fields of practical applications, such as meteorology, hydrology, engineering and biology. Up to now, many time series proved to be chaotic and inherently deterministic. Deterministic laws have been studied for chaotic time series. According to these rules, many efforts have been focused on the chaotic time series prediction. One of the questions is how to improve the prediction accuracy for long term prediction within the permission of chaotic rules [1], [2] and influence of noise [3], [4].

The chaotic time series prediction problem draws many attentions in neural network and machine learning communities. There are many neural networks or kernel-based methods introduced to the problem, such as the standard MLP (Multi-layer Perceptron) [5], RBF (Radial Basis Function) neural network [6], [7], SVR (Support Vector Regression) [8] and GP (Gaussian Process) [9], RNN (Recurrent Neural Network) including NAR (Nonlinear AutoRegressive) [10], RPNN (Recurrent Predictor Neural Networks) [11] and ESN (Echo State Networks) [12], [13] are also studied for nonlinear time series prediction. However, almost every realistic measurement is to some extent contaminated by noise, which limits the performance of many techniques of modeling and prediction. In the above listed methods, few of them could deal with the noisy time series prediction problem. Reference [12] reports a prediction method trained on noiseless time series recently, and in [11] and [14], a noise reduction method is used as a beforehand step, and the prediction model is built on the noise-reduced real life time series data. In [7] and [8], the noisy time series are considered by simulation studies, however, the influences of noise and uncertainty on the prediction model are not well studied. Reference [9] analyzes the uncertainty propagation through an iterative predictor, but no techniques are recommended to cope with the uncertainties in time series.

Noisy time series prediction can be seen as typical EIV (Errors-in-Variables) problem, in which the inputs to predictor model are the noisy history time series. TLS (Total Least Squares) also known as orthogonal regression is a common method to deal with EIV problem [15], which allows for the errors in the data matrix. Recently, robust extension to TLS or "robust counterpart" is studied, for example, in [16], [17], [18]. The common idea of them is to minimize the worst-case residual for a fixed uncertainty bound in sample data. TLS and its "robust counterpart" can not only be used in linear system but also in nonlinear system. In dealing with nonlinear system, some nonlinear basis functions are introduced so that the modeling becomes a linear-in-parameter problem [16].

In this paper, we use ESN which is a special kind of recurrent neural networks to model the noisy chaotic time series. With the help of a large, fixed and sparsely connected reservoir, the training of ENS becomes a simple linear regression problem, in other words, a linear in parameter problem. It is analyzed that because reservoir is fixed before the determination of readout weights, the perturbation in the echo state is induced by the perturbation in the input vector. For a target time series, once the perturbation bound in the echo state is well estimated, the robust design can be carried out. So, in this paper, the robust design is studied under the assumption that the perturbation in the echo state is known and bounded, and we will show how the perturbation bound is introduced into the robust design process. This paper is organized as follows: In Section II, the noisy chaotic time series prediction problem is firstly highlighted, and robust design method based on ESN and minimization of worst-case residual for a fixed uncertainty bound is introduced. Section III gives a demonstration to show the proposed robust design method, and shows that an accurate predictor does not mean a good prediction when the target time series contains noise and uncertainties. Conclusions and discussions are given in Section IV.

II. ROBUST DESIGN OF CHAOTIC TIME SERIES PREDICTOR

A. Noisy Chaotic Time Series Prediction Problem

Consider an unknown dynamical system whose evolution in discrete time is described by the nonlinear difference equation:

\[ x(k + 1) = F(x(k)) \]  

(1)
where \( \mathbf{x}(k) \) is the \( n \)-dimension state vector of the system at time \( k \), and \( \mathbf{F}(\cdot) \) is a vector-valued function. Time series \( \{y_a(k)\} \) is observed precisely from the state vector by:

\[
y_a(k) = h(\mathbf{x}(k))
\]

where \( h(\cdot) \) is a scalar-value function. The delay embedding vector is defined as:

\[
\mathbf{d}(k) = [y_a(k), y_a(k-\tau), y_a(k-2\tau), \ldots, y_a(k-(m-1)\tau)]^T
\]

where \( m \) and \( \tau \) are the embedding parameters of the time series. According to the Takens’ theorem [19], if the embedding dimension \( m \) is large enough, the evolution of delay embedding vector \( \mathbf{d}(k+1) = \mathbf{F}(\mathbf{d}(k)) \) can recover the original dynamic system without ambiguity. It is assumed that the evolution of the delay embedding vector is described by:

\[
\mathbf{d}(k+1) = \mathbf{F}(\mathbf{d}(k))
\]

and we assume \( y_a(k) \) is available through a measurement function \( g(\cdot) \):

\[
y_a(k) = g(\mathbf{d}(k))
\]

Generally speaking, an \( h \)-step-ahead predictor can be obtained by fitting a nonlinear regression model to the following mapping:

\[
\mathbf{d}(k) \rightarrow y_a(k+h)
\]

for any \( k \), where \( \mathbf{d}(k) \) is defined as the prediction origin, and \( y_a(k+h) \) is defined as the prediction horizon. The nonlinear regression model can be MLPs, RBF or SVMs, etc.

In practice, the time series often contain considerable noise and uncertainty. Assume that the practical measured time series is:

\[
y_a(k) = y_a(k) + v(k)
\]

and \( v(k) \) is the measurement noise or uncertainty at time \( k \), and the new prediction becomes:

\[
\mathbf{d}(k) = [y_a(k), y_a(k-\tau), y_a(k-2\tau), \ldots, y_a(k-(m-1)\tau)]^T = [y_a(k) + v(k), y_a(k-\tau) + v(k-\tau), y_a(k-2\tau) + v(k-2\tau), \ldots, y_a(k-(m-1)\tau) + v(k-(m-1)\tau)]^T,
\]

and the prediction horizon is \( y_a(k+h) = y_a(k+h) + v(k+h) \). The nonlinear mapping needed for modeling is actually:

\[
\mathbf{d}(k) \rightarrow y_a(k+h)
\]

It is difficult to model the mapping above, because the true mapping is polluted by the underlying noise and uncertainty. So a very practical problem is how to deal with the noise and uncertainty containing in the time series. Nonlinear noise reduction method is a commonly used tool for chaotic time series analysis. The main idea of the usual noise reduction is to replace each measurement \( y_a(k) \) by the average value of this coordinate over points in a suitably chosen neighborhood. In some chaotic time modeling methods, the noise reduction is firstly used to recover the true chaotic signal, and then nonlinear regression model is used to fit the noise reduced data. However, in some situations, the time series may contain considerable noise, and the noise reduction method may have limited performance. Sometimes, the noise reduced time series may still contain much noise, and it is especially true when the time series is short. So it is necessary to design a technique coping with this problem when the noise and uncertainty in the time series can not be ignored.

### B. Echo State Networks Prediction Model

The equations of the ESNs can be written as [12]:

\[
\mathbf{x}(k+1) = \text{tansig}(W_x \cdot \mathbf{x}(k) + W_{in} \cdot \mathbf{u}(k) + \mathbf{v}(k+1)),
\]

\[
y(k) = \mathbf{w}^T \mathbf{x}(k).
\]

where \( \text{tansig} \) denotes hyperbolic sigmoid function which is applied elementwise, \( \mathbf{x}(k) \) denotes the state variables in the "reservoir", \( \mathbf{u}(k) \) and \( y(k) \) are the input and output of ESNs, respectively, \( \mathbf{v}(k+1) \) is an optional noise vector. \( W_x, W_{in} \) and \( \mathbf{w} \) are the internal connection weights of the reservoir, the input weights to the reservoir and the readout (output) weights from the reservoir, respectively. Basic idea of ESN learning is the large and fixed "reservoir", from which the desired output is obtained by training suitable output weights. The "reservoir" has a large number of neurons which are randomly and sparsely connected. Determination of optimal output weights is a linear task of MSE minimization:

\[
\min_{\mathbf{w}} ||X \cdot \hat{\mathbf{w}} - \mathbf{y}_t||
\]

where \( X = [\mathbf{x}^T(\text{Init}), \mathbf{x}^T(\text{Init} + 1), \ldots, \mathbf{x}^T(\text{Trn})]^T \) is state matrix, and \( \mathbf{y}_t = [y_t(\text{Init}), y_t(\text{Init} + 1), \ldots, y_t(\text{Trn})]^T \) is the target vector, \( \text{Init} \) and \( \text{Trn} \) are the beginning and ending index of the training examples, respectively. The size of the training set is \( N_t = \text{Trn} - \text{Init} + 1 \). \( \text{Init} \) is usually set to certain value to discard the influence of reservoir initial transient.

Previous direct prediction methods such as MLPs treat the relationship between the prediction origin \( \mathbf{d}(k) \) and the prediction horizon \( y_a(k+h) \) as a static mapping. Because of the feedback and memory, ESNs are inherently dynamic systems. So the chaotic time series modeling using ESN becomes a dynamic system identification task. Fortunately, the multi-step chaotic time series prediction problem posed by the previous subsection can be reformulated as a dynamic system modeling task.

We assume function \( \mathbf{F}_0(\cdot) = 0 \), and the mapping between \( \mathbf{d}(k) \) and \( y_a(k+h) \) can be realized by the state space representation:

\[
\begin{bmatrix}
\mathbf{d}(k) \\
\mathbf{d}(k+1) \\
\mathbf{d}(k+2) \\
\vdots \\
\mathbf{d}(k+h)
\end{bmatrix} =
\begin{bmatrix}
\mathbf{F}_0(\mathbf{d}(k-1)) \\
\mathbf{F}(\mathbf{d}(k)) \\
\mathbf{F}(\mathbf{d}(k+1)) \\
\vdots \\
\mathbf{F}(\mathbf{d}(k+h-1))
\end{bmatrix} +
\begin{bmatrix}
1 \\
0 \\
0 \\
\vdots \\
0
\end{bmatrix} \cdot \mathbf{d}(k)
\]
where the vector \[ \mathbf{d}(k), \mathbf{d}^T(k+1), \ldots, \mathbf{d}^T(k+h) \] is the augmented state variable, and this state space realization for the input-output sequences \{ \mathbf{d}(k), y_{da}(k+h) \} \] is not unique. The training of ESN becomes the approximation of dynamic system described by Eq. (13). We will show in the simulations that this ESN prediction method can obtain high prediction accuracy.

If we feed the noisy embedding vector \( \mathbf{d}(k) \) to ESN \((u(k) = \mathbf{d}(k))\), the echo state \( \mathbf{x} \) will deviate from its true value because of the input noise. The perturbation \( \Delta \mathbf{y} \) in the target \( \mathbf{y}_t(k) \) is bounded \((\| \Delta \mathbf{y}_t \|_2 < \delta_y)\). Because \( W_x \) and \( W_{in} \) is fixed before the network training, the perturbation \( \Delta \mathbf{x} \) in state matrix \( \mathbf{X} \) is bounded \((\| \Delta \mathbf{x} \|_2 < \delta_x)\).

C. Minimize the Worst-case residual for a Fixed Uncertainty Bound

In this study, we would like to design a robust ESN predictor coping with the underlying uncertainty in time series. The bounded uncertainty in the input vector \( \mathbf{d}(k) \) induces a bounded uncertainty in echo state. The uncertainty bound of state matrix \( \mathbf{X} \) and target vector \( \mathbf{y}_t \) are assumed to a fixed value \( \delta_x \) and \( \delta_y \). Instead of Eq. (12), the determination of optimal output weights becomes the minimization of the worst residual:

\[
\min \limits_{\hat{\mathbf{w}}} \max \limits_{\| \Delta \mathbf{x} \| < \delta_x, \| \Delta \mathbf{y} \| < \delta_y} \{ \| (X + \Delta \mathbf{X})\hat{\mathbf{w}} - (\mathbf{y} + \Delta \mathbf{y}_t) \|_2 \}
\]

The solution to Eq. (14) is connected with the robust least square, which can be converted into a problem of regularization (ridge-regression). The optimal value of \( \hat{\mathbf{w}} \) can be obtained by singular value decomposition and solving a secular equation \([18]\), second-order cone programming \([16]\) or semi-definite programming \([16]\) if the bound of the uncertainty is known.

Some traditional design methods sometimes can also lead to the form of regularization when the noise is considered, for example, in \([7]\). In this design method, when the regularization is very weak, the problem is unconstrained and the solution is completely determined from the training data. Meanwhile, when the regularization is strong, it will result in a very smooth mapping, and it "is another way of saying that the training data are unreliable" \([7]\). In other words, strong regularization means strong uncertainness. Another feature of the traditional method is that the design has to rely on the noisy time series itself. The regularization level has to be determined by the noisy time series itself, for example using cross validation technique on the noisy time series.

The features of the robust method from the traditional one can be addressed as follows:

1) The robust design method clearly highlights the idea of uncertainness. The uncertainty in the target time series is quantified, as a result, the regularization level has a clear relationship with the uncertainness in the data, and there is no need of the cross validation for regularization level determination.

2) The robust design process does not necessarily rely on the noisy time series itself. In fact, the design can be carried out based on the nominal model (noiseless data) and the fixed uncertain bound.

Because there is no quantization of uncertainness, the traditional method can not create a predictor with the robustness against noise and uncertainness (this will be shown in the simulation). If the predictor is designed based on only the noiseless time series, and the cross validation has no access to the uncertainness in the time series. However, the robust design method, if provided a proper uncertainness bound, even based on the noiseless time series, can obtain a robust predictor. In other words, in traditional method, the uncertainness in the testing examples should accord with the uncertainness in the training examples, and the predictor have to be created from the noisy time series. However, the robust design method has no such disadvantages.

III. SIMULATION RESULTS

In this section, we will give a demonstration to show the main results of this paper. The simulation is based on the Mackey-Glass 84-step benchmark prediction problem. First of all, we will show how the nominal model deteriorates in presence of different level of noise. And then, we fix the uncertainness bound of the state matrix, and design robust predictors by solving the problem described by Eq. (14).

The Mackey-Glass system is a time-delay system with the form:

\[
\frac{dx}{dt} = \beta x(t) + \frac{\alpha x(t - \delta)}{1 + x(t - \delta)^10}
\]

where \( x(t) \) is the value of time series at time \( t \). The system is chaotic for \( \delta > 16.8 \). The parameter values are chosen as \( \alpha = 0.2, \beta = -0.1 \) and \( \delta = 17 \). The data set is constructed using second-order Runge-Kutta method with the step length of 0.1.

In this simulation, the prediction performance is measured by the root mean squared error on the test sequence pairs normalized by the variance of the original time series (N-RMSE).

\[
NRMSE = \sqrt{\frac{\sum_{i=1}^{T_n} (y_d(i) - y(i))^2}{T_n \cdot \sigma^2}}
\]

where \( y_d(i) \) denotes the 84-step target value, \( y(i) \) is the corresponding prediction output, \( T_n \) is the number of the test examples, and \( \sigma^2 \) is the variance of the original time series. The size of the reservoir is \( 700 \times 700 \), and the parameter settings of the reservoir are listed in TABLE I. Length of training sequence pairs is 1200. The first 200 steps are discarded to wash out initial transient, and echo states \( \mathbf{x}(k) \) are sampled from the remaining 1000 steps.
A. Noiseless Time Series Prediction (Nominal Model Design)

In this simulation, both the prediction origin $d(k)$ and the prediction horizon $y_{d1}(k+h)$ are noiseless. The predictor is trained on the ideal data without considering any uncertainty in the design process. The regularization parameter is determined by cross validation. A noiseless validation data set (data set length is 100) is created. In this paper, a nominal model is a predictor which is trained on noiseless data without considering the uncertainty in the target time series.

Once the nominal model is obtained, the prediction performance is firstly tested on a noiseless time series (700 samples). The NRMSE is measured by the prediction of the new noiseless testing data set. The NRMSE of the noiseless time series prediction is 0.016.

The prediction performance of the nominal model on the noiseless is excellent. It is interesting to investigate whether the predictor still works well when the nominal model is applied to noisy time series, when the input to the predictor contains uncertainty.

Noisy time series with different noise level (ratio between standard deviation of noise and signal standard deviation) are used to test the nominal model, and the NRMSE is calculated and listed in TABLE II. The test examples use the same 700 data points as before, added by Gaussian noise with different variances. The noise levels are 0.02%, 0.2%, 1% and 2%, respectively.

B. Noisy Time Series Prediction (Robust Design)

Because the bad results of nominal design for noisy time series, it is urgent to design a robust predictor for noisy time series.

The uncertainty bound of the state matrix is fixed in the experiments, and the studies are carried out to disclose the relationship between the uncertainty bound and the prediction performance for time series with different noise levels. The perturbation bound of state matrix $\delta_x$ is set to 0.001 and 10, respectively. After solving the problem Eq. (14), two robust models are obtained.

The same as the nominal model, the robust models are also tested by noiseless and different noisy time series. TABLE III lists the NRMSE of robust model ($\delta_x = 0.001$) for noiseless time series and the time series with noise level: 0.05%, 0.5%, 1% and 10%, respectively.

The robust design with $\delta_x = 0.001$ means that the optimal weights can minimize the worst residual caused by the perturbation of state matrix within $\|\Delta X\|_2 < 0.001$. The prediction error for noiseless time series is 0.129, and it is far worse than 0.016, which is the result of nominal design model. However, for 0.5%, 1% and higher noise level, the robust predictor performance remains relative better. For example, the NRMSE of robust predictor for 0.5% noise level time series is 0.179, and it is lower than 0.434, which is the NRMSE of nominal model predicting at 0.2% noise level.

In the second robust design case, we allow a larger perturbation in the state matrix $\|\Delta X\|_2 < \delta_x = 10$. Both the noiseless and noisy time series are tested on the robust model. The noise levels are 10%, 30%, 60% and 80%, respectively. The results are listed in TABLE IV. NRMSE on the noiseless time series goes up to 0.339, which is almost three times that of robust design with $\delta_x = 10$ and 20 times that of the nominal design. For the noiseless time series to the 30% noise level time series, the prediction performance changes slightly from 0.339 to 0.344. For a larger noise level, the performance deteriorates in some degree. For example, NRMSE for 60% and 80% noise level are 0.393 and 0.428, respectively. Though the NRMSE is large for the 10%, 30%, 60% or 80% noise level time series, the results are still better than those of nominal model and robust model with $\delta_x = 0.001$.

In the three designs, it can be observed that the nominal design is preferred if the target time series is noiseless, and for a small noise level (for example, 0.5%) the robust design with $\delta_x = 0.001$ can be chosen. The robust design with $\delta_x = 10$ functions well when the noise level is larger than 30%. So the prediction performance is dependent on both the noise level in the target time series and the perturbation bound $\delta_x$ in the process, and the perturbation bound should fit to the uncertainty in the target time series.

In Fig. 1, the relationships of $\log_{10}(\text{NRMSE})$ and
\[ \log_{10}(\text{noise level}) \] of nominal design, robust design with \( \delta_x = 0.001 \) and robust design with \( \delta_x = 10 \) are given. For different levels of target noisy time series, it is clear that the perturbation bound in the robust design should be different. Large perturbation bound is used when the noise level in the target time series is high, and for a low level of noisy time series, the \( \delta_x \) should be set small.

<table>
<thead>
<tr>
<th>Noise Level</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>NRMSE</td>
<td>0.339 \quad 0.340 \quad 0.344 \quad 0.393 \quad 0.428</td>
</tr>
</tbody>
</table>

Table IV: The NRMSE of robust model \((\delta_x = 10)\) for different noise level

![Graph showing log_{10}(NRMSE) of different designs versus time series with different noise levels.]

**IV. CONCLUSIONS**

For noisy time series, a robust predictor design method is studied. The design idea is minimizing the worst-case residual when the perturbation of the state matrix in ESN is bounded. In the experiment, the classical Mackey-Glass 84-step benchmark prediction task is investigated. The prediction performance is studied for the nominal and robust design of ESN predictor. The following conclusions can be obtained:

- The uncertainty bound of problem data is introduced to the design process, and one can have a better understanding of the prediction task and access of the predictability of the time series in presence of noise.
- Large uncertainty in the problem data will lead to bad prediction result, and the prediction performance is largely dependent on the noise level in the time series. Properly robust design can avoid unnecessary performance deterioration.
- The ESN solution can be improved using the idea of minimizing the worst-case residual. With the help of the robust design method, the ESN solution can be well understood, especially in dealing with EIV problem.

There are also other problems to be solved for making the robust design well carried out. For example, how to estimate the noise level for a given real life time series. Because in the robust design process, the perturbation bound \( x \) should match the perturbation in the inputs which are fed to the reservoir.

**REFERENCES**