Modeling of Venice Lagoon Time series with Improved Kalman Filter based neural networks

Archana R
Federal Institute of Science &Technology Angamaly, Kerala, India

A Unnikrishnan
Naval Physical & Oceanographic Laboratory, Cochin, Kerala, India

R. Gopikakumari
Cochin University of Science & Technology, Cochin, Kerala, India
,

ABSTRACT
The identification of nonlinear and chaotic systems is an important and challenging problem. Neural network models, particularly Recurrent Neural Networks (RNN) trained with suitable algorithms, have received particular attention in the area of nonlinear identification due to their potentialities to approximate any nonlinear behavior. A method of nonlinear identification based on the RNN model trained with improved nonlinear Kalman filter is proposed in this paper. The neural network weights are estimated using the Extended Kalman Filter(EKF) algorithm, augmented by the Expectation Maximization(EM) algorithm is used to derive the initial states and covariance of the Kalman filter. It was shown that not only could this chaotic approach provide an accurate identification, but it was also more effective in the sense that the approach had a smaller mean squares error (MSE). An experimental case study using the famous Venice lagoon time series is analyzed by the proposed algorithm. The minimum embedding dimension of the time series is calculated using the method of false nearest neighbors. The Lyapunov exponents of the model are calculated, from the state space evolution, The numerical results presented here indicate that the traditional Extended Kalman Filter algorithm combined with EM techniques are effective in building a good NN model for nonlinear identification.

Key Words
Artificial Neural Network, Extended Kalman Filter, Expectation maximization, Recurrent Neural Networks, Lyapunov exponent, chaotic systems, Venice Lagoon time series**.**

I. INTRODUCTION
Neural network is composed of highly interconnected artificial neurons and they can be used for system modeling as well as identification. Recurrent neural networks can exactly approximate any nonlinear map, and has high convergence. The Neural Network system has been applied in the modeling and analysis of non-linear and chaotic system with great success [2][8].
The Chaotic systems have been of interest to many researchers over years. Chaos is a complex and unpredictable phenomena, which occur in non linear systems that are sensitive to their initial conditions. The modeling of chaotic systems, based on the output time series is quite challenging. Fortunately, artificial neural networks have the required self-learning capability to tune the network parameters (i.e. weights) to identify highly nonlinear and chaotic systems. [10][11]

In the present work, the efficacy of modeling a chaotic system using dynamic neural networks has been demonstrated. The Venice Lagoon time series is a measure of the level of water in the Lagoon in centimeters each hour along the years 1940-1990.Non linear stochastic models are used for modeling the system. Unusually high tides and other climatic conditions sometimes drive the time series to show chaotic nature. The recurrent architecture also generates the state space evolution, while trying to arrive at the model of the output time series. The parameters of the neural network are estimated using the Extended Kalman Filter (EKF) algorithm, by choosing the weights of the neural network as the states of the Extended Kalman Filter. Further, the Expectation Maximization algorithm is used to effectively arrive at the initial states and the state covariance, required in the EKF algorithm. The recurrent network, shown in Fig. 1. models the following system:

$$x_{k+1} = f(x_k , W) \qquad (1)$$

$$y (k) = h(x_k, y(k-1), y(k-2)...y(k-p), \nu_k, \nu_{k-1}, \nu_{k-2}...,W \quad (2)$$

The proposed neural network system models the chaotic time series effectively. The state variables continue to generate the state space evolution of the system, responsible for generating the time series. The minimum embedding dimension of the time series is calculated using the method of false nearest neighbors. The Lyapunov exponents, which characterize the behavior of the system are also calculated from the state space evolution and verified.

2. RECURRENT NEURAL NETWORK
The recurrent networks have the potential to be used in unison in systems with dynamic elements and feedback [2]. In effect recurrent neural networks used for modeling or model based predictive control are multi-layer neural networks with a delay element in their feedback loop. Recurrent neural networks could be built with multi-layer networks in their feedback loop, creating a system where the structures compute in tandem. In recurrent network nodes are connected back to other nodes or themselves. The Information flow is multidirectional. Such networks inherently possess sense of time and memory of previous state(s).Biological nervous systems show high levels of recurrency .Hence the networks could be used in unison creating systems with both dynamic elements and feedback. The presence of feedback loops has a profound impact on the learning capability of the network and its performance. Moreover the feedback loops involve unit delay elements denoted by z^{-1}, which results in a nonlinear dynamical behavior [2].

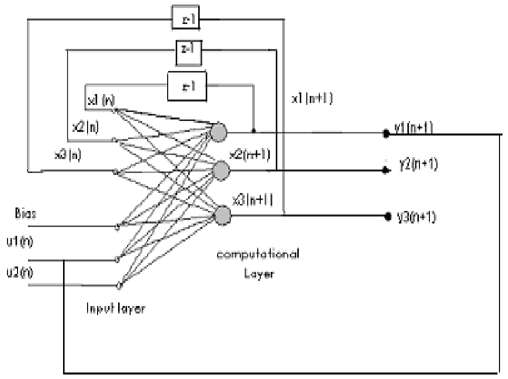

Fig.1.Structure of RNN

3. MODELING WITH EXTENDED KALMAN FILTER

A. System representation

Consider a discrete time non linear dynamic system, described by a vector difference equation with additive white Gaussian noise that models "unpredictable" disturbances. The Kalman filter deals with linear systems .We can see that Kalman filter needs modifications for adapting the nonlinear behavior of the system. The dynamic plant equation is given by the following nonlinear equations

$$x_k = f(x_{k-1}, u_k, w_{k-1})$$
(3)

where x_k is an n dimensional state vector u_k is an m dimensional known input vector, and w_k is a sequence of independent and identically distributed zero mean white Gaussian process noise with covariance

$$E(ww^T) = Q$$
(4)

The measurement equation is

$$z_k = h(x_k, v_k)$$
(5)

Where v_k is the measurement noise with covariance

$$E(vv^T) = R$$
(6)

The functions f and h and the matrices Q and R are assumed to be known.

B. Extended Kalman Filter

The Extended Kalman filtering process has been designed to estimate the state vector in a non linear stochastic difference model.[5]To estimate a process with non-linear difference and measurement relationships, we begin by writing new governing equations that linearize equation (3) and equation (5)

Expanding the functions f and h along the Taylor series, one gets the following equations for the Extended Kalman filter. [5].

$$x_k = \tilde{x}_k + A(x_{k-1} - \hat{x}_{k-1}) + Ww_{k-1}$$
(7)

$$z_k = \tilde{z}_k + H(x_k - \tilde{x}_k) + Vv_k$$
(8)

Where x_k and z_k are the actual state and measurement vectors, $\tilde{x}_k$ and $\tilde{z}_k$ are the approximate state and measurement vectors. $\hat{x}_k$ is an aposteriori estimate of the state at step k , w_k and v_k are the random variables and represent the process and measurement noise. A is the Jacobian matrix of partial derivatives of f with respect to x, H is the Jacobian matrix of partial derivatives of x with respect to h , W is the Jacobian matrix of partial derivatives of with respect to w , V is the Jacobian matrix of partial derivatives of with respect to v.

C.EKF time update equations:

Project the state ahead

$$\hat{x}_k^- = f(\hat{x}_{k-1}, u_k, 0)$$
(9)

Project the error covariance ahead

$$P_k^- = A_k P_{k-1} A_k^T + W_k Q_{k-1} W_k^T$$
(10)

D.EKF measurement update equations:

Compute the Kalman gain

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + V_k R_k V_k^T)^{-1}$$
(11)

Update estimate with measurement

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - h(\hat{x}_k^-, 0))$$
(12)

Update error covariance

$$P_k = (I - K_k H_k P_k^-)$$
(13)

One of the basic problems in the implementation of the Kalman filter is the choice of the initial values of the state x and the state co-variance P. Since arbitrary choices can lead to the divergence of the filter, the present paper has used the EM algorithm [3] to compute the initial values of state and the state co variance.

E. EM Algorithm

The EKF Algorithm for training Multi Layer Perceptrons (MLPs) suffers from serious shortcomings, namely choosing the initial states and covariance $(x, Q and R)$. After computing the forward estimates in EKF, employ ‘’Rauch-Tung-Striebel smoother’’[] to do the following backward recursions.

$$J_{k-1} = P_{k-1} A^T P_{k-1}^{-1}$$

(14)

$$x_{k-1} = \hat{x}_{k-1} J_{k-1} \left(\hat{x}_{k-1} - A \hat{x}_{k-1} \right)$$

(15)

$$P_{k-1} = \hat{P}_{k-1} + J_{k-1} \left(\hat{P}_k - \hat{P}_{k-1} \right) J_{k-1}^T$$

(16)

$$P_k = P_{k-1} J_{k-1}^T + J_k \left(P_{k-1} - AP_{k-1} \right) J_{k-1}^T$$

(17)

F. RNN Training Using EKF Algorithm

In the modified Kalman algorithm the state and measurement equations are modified as follows:

Considering the parameter optimization as a state estimation as described above allows us to use the extended Kalman filter to update the weight estimates as well as the optimal hidden states[6], [7]. The augmented state vector is thus

$[w_1 , w_2 \ldots w_n , x_1, x_2, \ldots x_p]^T$

The algorithm is explained below:

1. All the weights and states are initialized to small random values. The state covariant matrix P(0/-1) is initialized to a diagonal matrix, with relatively small values. Let the covariant matrix for measurement noise is R and that of process noise is Q.
2. As before compute the out put at each node of the recurrent network.
3. Find the Jacobian matrix with respect to the state of the process and output at the current estimate of internal state and weights of the RNN. These matrices are given by H and A defined as follows.[3]

$$H = \left[\frac{\partial h(.)}{\partial w} .... \frac{\partial h(.)}{\partial w} \right]$$

(18)

$$A = \begin{bmatrix} I ....0 \\ 0....... \dfrac{\partial f(.)}{\partial x} \end{bmatrix}$$

(19)

The output of the neural network is computed using

$z_k = g(\Sigma\, y_{k-n} w_n + v_l w_l + w_1 x_{1} + w_2 x_{2} + w_3 x_3)$, where g is

any chosen non linear function. The network works with EKF algorithm as per the equations described in section .II.B

with the state vector x replaced by the weights of the RNN.

4. MINIMUM EMBEDDING DIMENSION

A set has embedding dimension n if n is the smallest integer for which it can be embedded into without intersecting itself. Whitney’s embedding theorem states that if a manifold has topological dimension 2 its embedding dimension is at most 2n. Taken’s theorem states that the original dynamic properties of the attractor can be retained as long as the embedding dimension de $\geq$ 2d+1 where d is the correlation dimension of the attractor’s. Choosing an Embedding dimension can be done by the method of False Nearest Neighbours as explained by the following algorithm:

1. Measure the distances between a point and its nearest neighbor, as this dimension increases, this distance should not change,if the points are really nearest neighbors .

2. Define the distance between a point and its nearest neighbor as R_d. It is calculated using the following formula

$$(R_{d+1}(t))^2 = (R_d(t))^2 + [x(t + d\tau) - x^{NN}(t + d\tau)]^2 \quad (20)$$

Now add one more dimension and calculate the change in distance

$$(R_{d+1}(t))^2 = (R_d(t))^2 + [x(t + d\tau) - x^{NN}(t + d\tau)]^2 \quad (21)$$

we can now look at the relative change in the distance as a way to see if our points were not really close together but a projection form a higher phase space .Define a threshold R_T as a criteria for false nearest neighbours

$$R_T \prec \frac{|[x(t + d\tau) - x^{NN}(t + d\tau)]|}{R_d(t)} \quad (22)$$

Using this criterion we can then test our sequence of points and, as d increases, find where the percentage of nearest neighbors goes to 0.After finding out the percentage of false nearest neighbors a graph is plotted between the percentage of false nearest neighbors and embedding dimension..From the figure we will get the minimum embedding dimension.[2]

5. LYAPUNOV EXPONENTS

The Lyapunov Exponents of a system are a set of invariant geometric measures that describe the dynamical content of the system. In particular, they serve as a measure of how easy it is to perform prediction on the system under consideration. Lyapunov Exponents quantify the average rate of convergence or divergence of nearby trajectories in a global sense. A positive exponent implies divergence and a negative one implies convergence. Consequently, a system with positive exponents has positive entropy in that trajectories that are initially close together move apart over time. The more positive the exponent, the faster they move apart. Similarly, for negative exponents, the trajectories move together. A system with both positive and negative Lyapunov Exponent is said to be chaotic. [9]

Mathematically Lyapunov Exponent can be defined by

$$\lambda_i = \lim_{n \to \infty} \sum_{t=0}^{n-1} \ln \left[\frac{\partial f(x_i)}{\partial x_i} \right]$$

(23)

where x_i , is the i^{th} state variable of the system and $f(x_i)$ is the output of the system.

6. SYSTEM SIMULATION

A. Venice Lagoon Time series

Unusually high tides, or sea surges, result from a combination of chaotic climatic elements in conjunction with the more normal, periodic, tidal systems associated with a particular area. The prediction of such events has always been subject of intense interest to mankind, not only from human point of view, but also from an economic one. The most famous example of flooding in the Venice Lagoon occurred in November 1966 when, driven by strong winds, the Venice Lagoon rose by nearly 2 m above the normal water level. The damage to the city's homes, churches and museums ran into hundreds of millions of Euros. Tide's behavior is difficult to be predicted, because it depends on too much factors, like the astronomic and atmospheric agents. The problem has been approached by numerical models and statistical methods. Numerical models require the computation of the meteorological forcing functions on each point of the finite difference grid and, hence, they are computationally expensive. Non Linear stochastic models are suitable for online forecasting since they are simple and their computation burden is low. The given time series contains the level of water in the Venice Lagoon measured in centimetres each hour along the years 1940-1990.The system is subjected to noise reduction.The raw and noise reduced time series are shown in Fig and Fig

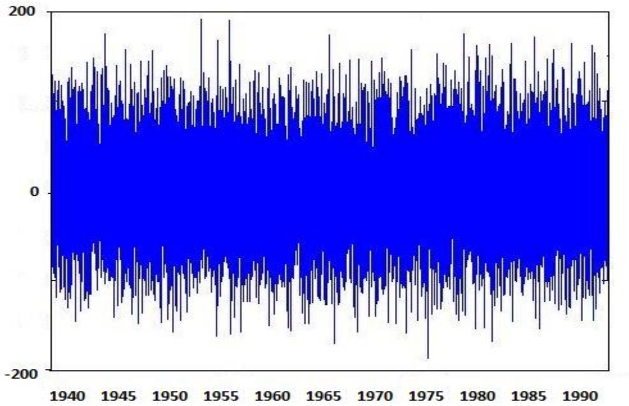

Fig.2.Noisy time series of Venice Lagoon

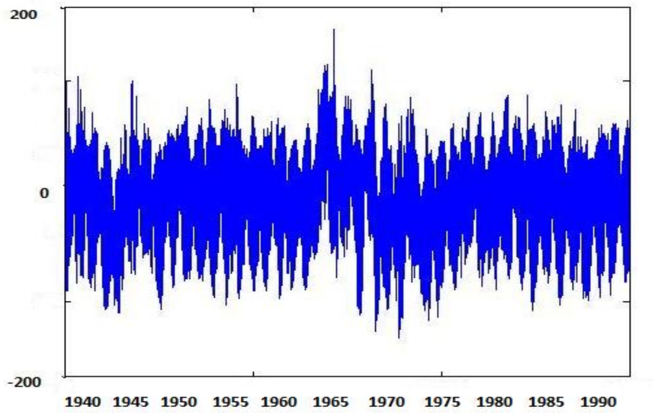

Fig.3.Noise reduced time series of Venice Lagoon.

Recurrent neural networks are trained with a single channel time series data.

All the three sets of weights w^1 … are updated using the EKF equations. The initial values of P(0/-1) and x(0) are obtained using the equations(14) to (17) executed forwards and backwards , over 10000 data samples on the time series. The training is continued until the modeling error comes to an appreciable level of 0.00254×10^{-5} as shown in Fig.4

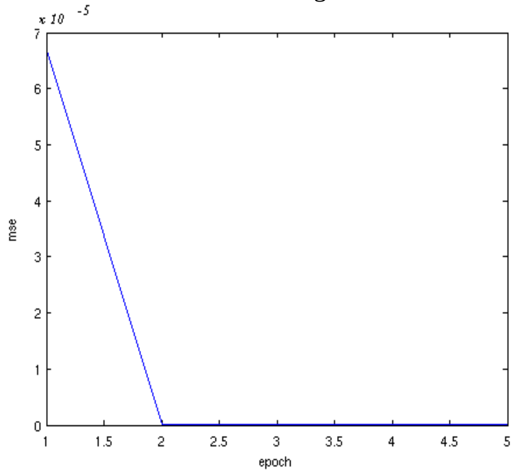

Fig.4.MSE versus data samples('Venice Lagoon time series')

B.Embedding dimension

The minimum embedding dimension is calculated using the method of false nearest neighbors as explained in section IV. After finding out the percentage of false nearest neighbors a graph is plotted between the percentage of false nearest neighbors and embedding dimension as shown in Fig .From the figure we found the value of embedding dimension equal to two. The Lyapunov Exponents of the time series are calculated.

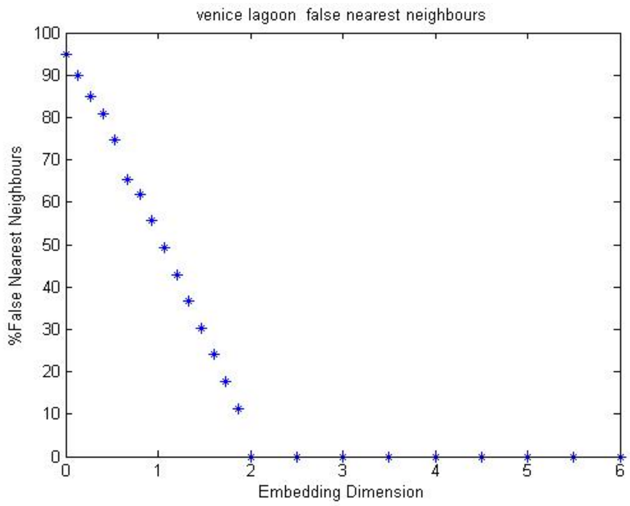

Fig.5.Percentage of false nearest neighbours

C.State space analysis

The dynamics of the states of the systems are evaluated using recurrent network. The state space analysis is done and change in dynamics of the system is described in the different time intervals. The two states of the system are exactly reproduced by the NNEKF model.

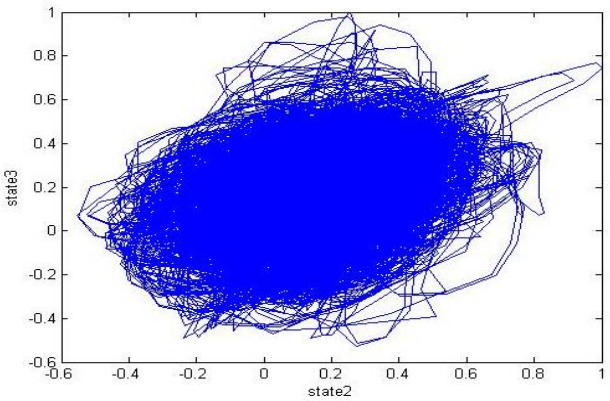

Fig.6.Phase plot(states1 and 2)('Venice Lagoon time series')

D.Calculation of Lyapunov Exponents

The Lyapunov exponents of the the Venice Lagoon time series is calculated. and the ANN – EKF model are calculated

TABLE 1.

Lyapunov exponents	
0	0.05

We can see that one of Lyapunov exponent is zero and the other one is positive verifying the chaotic behavior.

7. CONCLUSION

It is demonstrated that the recurrent networks trained with EKF-EM algorithm can be efficiently used to identify chaotic systems. We extracted the Venice Lagoon time series from a noisy measurement. We also evaluated the embedding dimension and the given time series is modelled with an embedding dimension of two. The proposed method results in vey low modelling error. The proposed method has high ability of extracting the structure of the original chaotic systems. The system states $\boldsymbol{x_k}$ and the set of model parameters $\boldsymbol{w}$ for the dynamic system are simultaneously estimated from only the observed the time series yk **.**The Lyapunov exponents of the system are calculated. A positive Lyapunov exponent shows the hidden chaotic nature of the time series. The State space evolution is plotted which also predicts the chaotic nature. The proposed method gives very small modeling error and considerably low computation time. This method can be further extended for the prediction of highly nonlinear and chaotic systems.

8. REFERENCES

[1] N.K Sinha and B Kuszta, *Modeling and Identification of Dynamic systems*,Van Nostrand Reinhold Company, New York,1983.

[2] Simon Haykin, *Neural Networks a comprehensive Foundation*, Prentice Hall International Editions, 1999.

[3] Fa-Long Luo and Rolf Unbehauen, *Applied Neural Networks for Signal Processing*, Cambridge University Press, 1997.

[4] Yaakov Bar-Shalaom and Xiao-Rong Li, *Estimation and Tracking: Principles, Techniques and Software*. Artech House, Boston, London, 1993.

[5] Greg Welch and Garry *Bishop"An introduction to KalmanFilter"*cc.cs.unc.edu/~tracker/s2001/kalman

[6] A.S.Poznyak,Wen Yu and E.N Sanchez, *"Identification and control of unknown chaotic systems via dynamic neural*

networks", IEEE Trans. on Circuits and Systems I: FundamentalTheoy and Applications, Vol. 46, No. 12 pp.1491 -1495,1999.

[7] G.V. Puskorius and L.A Feldkamp, " *Neuro Control of nonlinear dynamical systems with Kalman Filter trained recurrent networks*" IEEE Transactions on neural networks, Vol 5, No.2, pp 279-297, 1994.

[8] K.S. Narendra and K Parthasarathy, "*Identification and control of Dynamical systems using neural networks*", IEEE Transactions on Neural Networks, Vol 1, No.2, pp 4-27, March 1990

[9] Geist K U Parlitz and W Lauterborn"*Comparison of different methods for computing Lyapunov exoonents*"Progress of theoretical Physics 83-5 pp 875-893 1990.

[10] Zhiven Zhu and Henry Leung *"Identification of Linear Systems Driven by Chaotic Signals"*

IEEE transactions on circuits and systems—i: fundamental theory and applications, vol. 49, no. 2, February 2002

[11] Ramon A. Felix, Edgar N. Sanchez, *Senior Member, IEEE*, and Guanrong Chen, *Fellow, IEEE "Reproducing Chaos by Variable Structure Recurrent Neural Networks*" IEEE transactions on neural networks, vol. 15, no. 6, November 200