International Journal of Engineering Research and Development
eISSN : 2278-067X, pISSN : 2278-800X, www.ijerd.com
Volume 2, Issue 2 (July 2012), PP. 49-56

Computation of state space evolution of chaotic systems from time series of output, based on neural network models

Archana R[1], A Unnikrishnan[2] , R. Gopikakumari[3]
[1]*Federal Institute of Science &Technology Angamaly, Kerala, India*
[2]*Naval Physical & Oceanographic Laboratory, Cochin, Kerala, India*
[3]*Cochin University of Science & Technology, Cochin, Kerala, India*

Abstract*—A novel method for generating the state space evolution of a nonlinear system from the output time series using the Recurrent Neural Network (RNN) model, trained with improved nonlinear Kalman filter is proposed in this paper. The identification of nonlinear and chaotic systems from the output time series is an important and challenging problem. Nonlinear identification using neural network models, particularly Recurrent Neural Networks (RNN), trained with suitable algorithms, have received particular attention due to their potentialities to approximate nonlinear behaviour. It is also well known that the evolution of state space provides more information on the behaviour of the systems.. The neural network weights are estimated using the Extended Kalman Filter (EKF) algorithm. The performance of the EKF algorithm is further improved by the Expectation Maximization (EM) method, which is used to derive the initial states and covariance of the Kalman filter. The approach provides an accurate identification as well as results in a smaller Mean Squares Error (MSE). The minimum embedding dimension of the time series is calculated using the method of false nearest neighbours, which helps to decide the number of states required to model the system. By allowing the system to freewheel driven by white noise, after the modelling, the state space evolution is produced. A case study using the famous sunspot time series is carried out by the proposed algorithm and the Lyapunov exponents of the model are calculated, from the state space evolution. The results presented here confirms the efficacy of the Extended Kalman Filter algorithm combined with EM techniques in building a good RNN model for nonlinear identification of chaotic systems.

***Keywords:* Artificial Neural Network, Extended Kalman Filter, Expectation maximization, Recurrent Neural Networks, Lyapunov exponent, chaotic systems, Sunspot time series.**

I. INTRODUCTION

It is well known that the Rrecurrent neural networks can exactly approximate any nonlinear map, and has high convergence [2][15]. Accordingly, the Neural Networks have been applied extensively in the modelling and analysis of non-linear and chaotic system with great success.

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 [17]. The modelling of chaotic systems, based on the output time series is quite promising, since the output often represents the characteristics behaviour of the total system. Artificial neural networks have the required self-learning capability to tune the network parameters (i.e. weights) to identify highly nonlinear and chaotic systems. [2][12]

In the present work, the efficacy of modelling a chaotic system using dynamic neural networks has been demonstrated. The Sunspot time series is a collection of sunspot numbers tabulated each day along the years 1818 to 2000. The time series is inherently chaotic in nature [16].The system is modelled using the neural network system. 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:

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

$$y(k) = h(x_k, y(k-1), y(k-2) \ldots y(k-p), v_k, v_{k-1}, v_{k-2} \ldots, W) \tag{2}$$

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 neighbours. The Lyapunov exponents, which characterize the behaviour of the system, are also calculated from the state space evolution and verified. The state space evolution shows that the sunspot time series is similar to the famous Rossler chaotic system.

II. RECURRENT NEURAL NETWORK AND TIME SERIES MODELING

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 modelling 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 recurrences .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 [15].

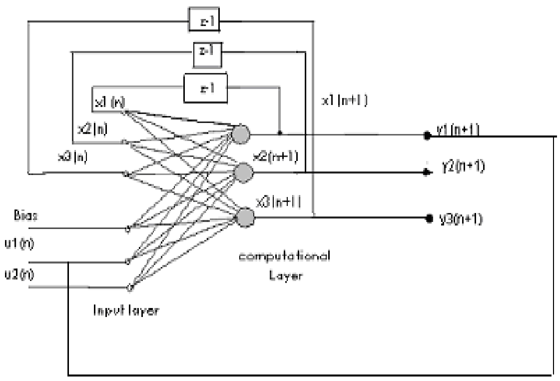

Fig.1. .Structure of RNN for modeling the state space evolution

The present paper proposes a ne w approach to model the time series, simultaneously generating the state space as given in Eqn. (1) and (2) . The time series is modelled using the RNN and after the error stabiles to an acceptable value, the system is allowed to freewheel driven by white noise, when the state space evolution is generated[11]. The obvious question here is: how many state variables are required to represent a system for which only the output time series is known? It is proposed to use the concept of self embedding dimensions [8], which can be computed readily from the output time series.

III. MINIMUM EMBEDDING DIMENSION

The minimum embedding dimension is calculated using the method of false nearest neighbours as explained below. A set has embedding dimension n if n is the smallest integer for which it can be embedded into without intersecting itself[8]. Whitney's embedding theorem states that if a manifold has topological dimension 2 its embedding dimension is at most 2n. Taken's theorem [9] [10] 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 neighbour, as this dimension increases, this distance should not change, if the points are really nearest neighbours.
2. Define the distance between a point and its nearest neighbour as $R_{d.}$ It is calculated using the following formula

$$(R_{d+1}(t))^2 = (R_d(t))^2 + [x(t+d\tau) - x^{NN}(t+d\tau)]^2 \qquad (3)$$

Now add one more dimension and calculate the change in distance

$$(R_{d+1}(t))^2 = (R_d(t))^2 + [x(t+d\tau) - x^{NN}(t+d\tau)]^2 \qquad (4)$$

where NN indicates the nearest neighbour. 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 state space .Define a threshold R_T as a criteria for false nearest neighbours

$$R_T \prec \frac{\left|[x(t+d\tau) - x^{NN}(t+d\tau)]\right|}{R_d(t)} \qquad (5)$$

Using this criterion one can test our sequence of points and, as d increases, find where the percentage of nearest neighbours goes to 0. After finding out the percentage of false nearest neighbours a graph is plotted between the percentage of false nearest neighbours and embedding dimension..The lowest point in the graph gives self the minimum embedding dimension.[2] For the sun spot time series , Fig. 2 confirms that the self embedding dimension is 3

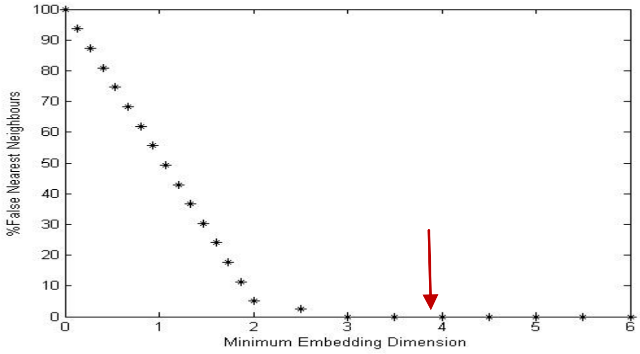

Fig. 2. Computation of the self embedding dimension from sun spot time series

IV. MODELING THE TIME SERIES 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 [5].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

$$x_k = f(x_{k-1}, u_k, w_{k-1}) \tag{6}$$

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

$$E(ww^T) = Q \tag{7}$$

The measurement equation is

$$z_k = h(x_k, v_k) \tag{8}$$

Where v_k is the measurement noise with covariance

$$E(vv^T) = R \tag{9}$$

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

B. Extended Kalman Filter[5]

The Extended Kalman filtering (EKF) 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].

$$x_k = \tilde{x}_k + A(x_{k-1} - \hat{x}_{k-1}) + Ww_{k-1} \tag{10}$$

$$z_k = \tilde{z}_k + H(x_k - \tilde{x}_k) + Vv_k \tag{11}$$

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 a posteriori 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 Jacobean matrix of partial derivatives of f with respect to x, H is the Jacobean matrix of partial derivatives of x with respect to h , W is the Jacobean matrix of partial derivatives of with respect to w , V is the Jacobean matrix of partial derivatives of with respect to v.

C.EKF time update equations:

Project the state ahead

$$\hat{x}_k^- = f(\hat{x}_{k-1}, u_k, 0) \tag{12}$$

Project the error covariance ahead

$$\bar{P}_k = A_k P_{k-1} A_k^T + W_k Q_{k-1} W_k^T \tag{13}$$

D.EKF measurement update equations:
Compute the Kalman gain

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + V_k R_k V_k^T)^{-1} \tag{14}$$

Update estimate with measurement

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - h(\hat{x}_k^-, 0)) \tag{15}$$

Update error covariance

$$P_k = (I - K_k H_k \bar{P}_k) \tag{16}$$

One of the basic problems in the implementation of the EKF 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 [7] to compute the initial values of state and the state co variance.

E. EM Algorithm

The EKF Algorithm [5] for training Multi Layer Perceptrons (MLPs) suffers from some shortcomings, namely choosing the initial states and covariance x(0), P(0/-1)., along with the process error covariance Q and measurement error covariance R. We propose to alleviate the problem by using the EM algorithm [6]. After computing the forward estimates in EKF, the ‘’Rauch-Tung- Striebel smoother’’[7] is executed on the same data series to do the following backward recursions.

$$J_{k-1} = P_{k-1} A^T P_{k-1}^{-1} \tag{17}$$

$$x_{k-1} = \hat{x}_{k-1} J_{k-1} \left(\hat{x}_{k-1} - A \hat{x}_{k-1} \right) \tag{18}$$

$$P_{k-1} = \hat{P}_{k-1} + J_{k-1} \left(\hat{P}_k - \hat{P}_{k-1} \right) J_{k-1}^T \tag{19}$$

$$P_k = P_{k-1} J_{k-1}^T + J_k \left(P_{k-1} - A P_{k-1} \right) J_{k-1}^T \tag{20}$$

Finally the values of x(0) and Q(0/-1) are given by

F. RNN Training Using EKF Algorithm for both the state space evolution and time series modelling
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. 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 Jacobean 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.

$$H = \left[\frac{\partial h(.)}{\partial w} \ldots\ldots \frac{\partial h(.)}{\partial w} \right] \tag{21}$$

$$A = \begin{bmatrix} I \ldots\ldots\ldots\ldots 0 \\ 0 \ldots\ldots \dfrac{\partial f(.)}{\partial x} \end{bmatrix} \tag{22}$$

The output of the neural network is computed using

$$z_k = g(\Sigma\, y_{k-n} w_n + \nu_l w_l + w_1 x_1 + w_2 x_2 + w_3 x_3),$$

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.

V. 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. [12]

Formally the Lyapunov Exponent can be defined by

$$\lambda_i = \lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} \ln\left[\frac{\partial f(x_i)}{\partial x_i}\right] \tag{23}$$

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

VI. SYSTEM SIMULATION

A. Sunspot Time series

Sunspots appear as dark spots on the surface of the Sun. They typically last for several days, although very large ones may live for several weeks. Sunspots are magnetic regions on the Sun with magnetic field strengths thousands of times strongerhan the Earth's magnetic field. Sunspots usually come in groups with two sets of spots. One set will have positive or north magnetic field while the other set will have negative or south magnetic field. Although sunspots themselves produce only minor effects on solar emissions, the magnetic activity that accompanies the sunspots can produce dramatic changes in the ultraviolet and soft x-ray emission levels. These changes over the solar cycle have important consequences for the Earth's upper atmosphere.

The sunspot number is calculated by first counting the number of sunspot groups and then the number of individual sunspots. The "sunspot number" is then given by the sum of the number of individual sunspots and ten times the number of groups. Since most sunspot groups have, on average, about ten spots, this formula for counting sunspots gives reliable numbers even when the observing conditions are less than ideal and small spots are hard to see. The given time series contains the sunspot number measured from year 1818 to 2011.The system is subjected to noise reduction. The raw and noise reduced time series are shown in Fig2 and Fig3

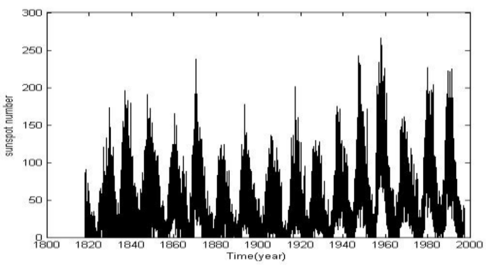

Fig.3.Noisy time series of Sunspot Number

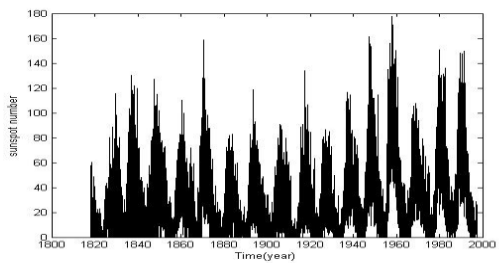

Fig.4.Noise reduced time series of Sunspot Number.

In our system the recurrent neural networks are trained with a single channel time series data of the sunspot. 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 modelling error comes to an appreciable level of 0.00254×10^{-5} as shown in Fig.4

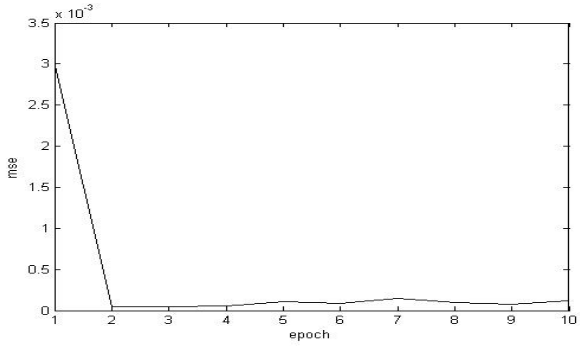

Fig.5.MSE versus data samples('Sunspot time series')

B. 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. The state space evolution shows that the system modelled from the sunspot time series using the method reported here is very much similar to the famous chaotic system developed by Otto Rossler.(Fig 6 to fig13). The Rossler system is described by the following nonlinear equations.

$$\frac{dx}{dt} = -y - z \tag{24}$$

$$\frac{dy}{dt} = x + ay \tag{25}$$

$$\frac{dz}{dt} = bx - cz + xz \tag{26}$$

The state space evolution of Rossler system generated from the above dynamical system is also given below for comparison. It can be seen that Fig. 6 and 7, 8 and 9, 10 and 11 and 12 and 13, taken pair wise, underscores the similarity between the systems . As suggested by Min Han [15] we could prove that the sunspot time series shows exact similarity to Rossler system .Further, the minimum embedding dimension calculated as three from the method of false nearest neighbours is also validated from the state space evolution.

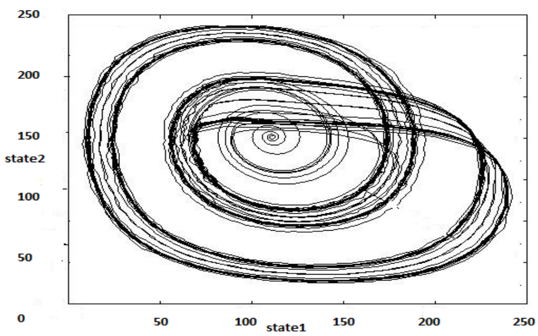

Fig.6.Phase plot (states1 and 2) ('Sunspot time series')

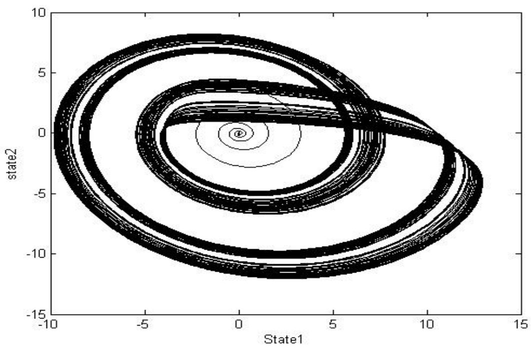

Fig.7.Phase plot (states1 and 2) ('Rossler system')

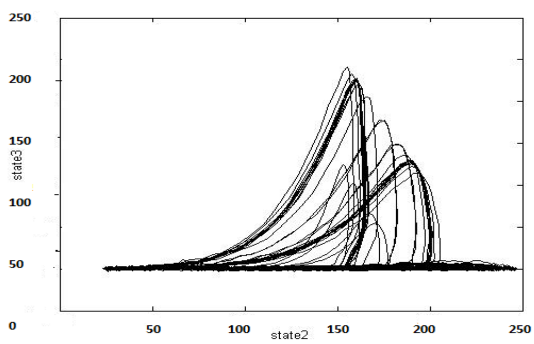

Fig.8.Phase plot (states2 and 3) ('Sunspot time series')

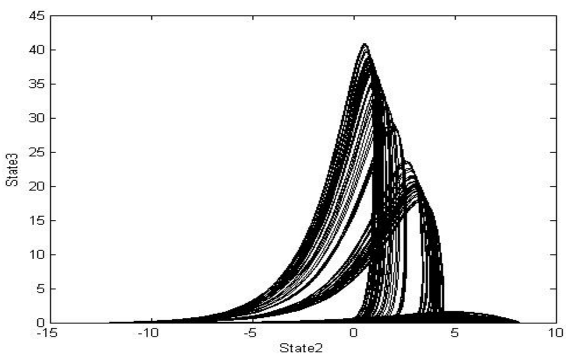

Fig.9.Phase plot (states2 and 3) ('Rossler system')

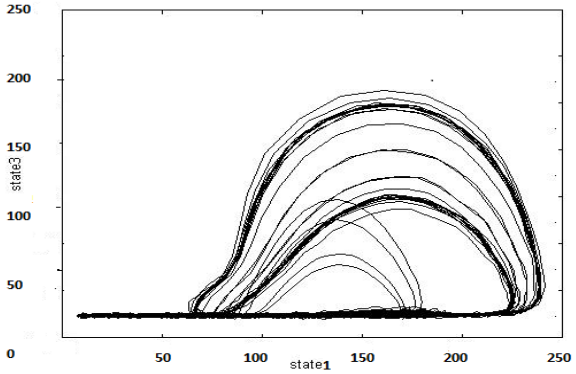

Fig.10.Phase plot (states1 and 3) ('Sunspot time series')

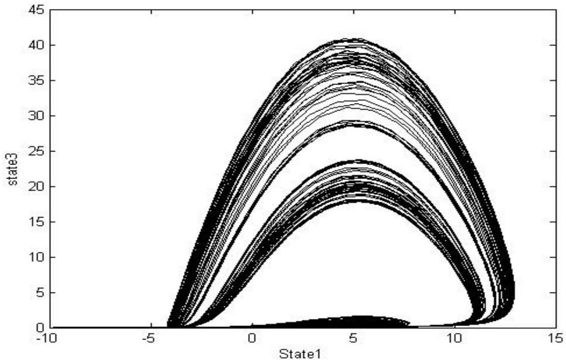

Fig.11.Phase plot (states1 and 3) ('Rossler system')

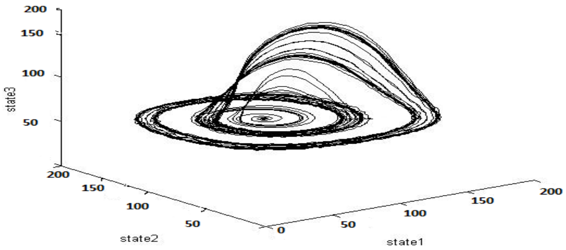

Fig.12.Phase plot (states1,2 and 3) ('Sunspot time series')

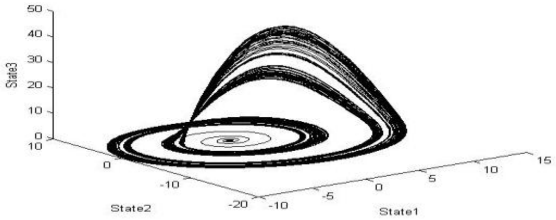

Fig.13.Phase plot (states1, 2 and 3) ('Rossler system')

The Lyapunov Exponents of the Sunspot time series is calculated using the method described in section IV and is given in table 1.

Table.1.Lyapunov exponents

-5.2446	0	0.0694

It can be seen that one of Lyapunov exponent is negative, one is zero and the other one is positive verifying the chaotic behaviour.

VII. CONCLUSION

It is demonstrated that the recurrent networks trained with EKF-EM algorithm can be efficiently used to identify chaotic systems, from the time series. The results are demonstrated on the Sunspot time series from a noisy measurement. The number of states required for modelling is evaluated using the method of minimum embedding dimension and thereafter, the given time series is modelled with an embedding dimension of three. 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 time series y_k . The results are shown to have an excellent matching with the state space evolution generated from the mathematical equations describing the Rossler system. The positive Lyapunov exponents show the hidden chaotic nature of the time series. The proposed method gives very low modelling error and considerably low computational time. This method can be further extended for the prediction of highly nonlinear and chaotic systems.

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-Shalom 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]. Chuong B Do and Serafim Batzoglou "*What is the expectation maximization algorithm*?"Nature Biotechnology Vol 26 No 8 pp 897-899,August 2008.
[7]. Simo Sarkka "*Unscented Rauch Tung striebel Smoother*"IEEE transactions on automatic control Vol 53 No 3 pp 845-849
[8]. Liangyue Cao "*Practical method for determining the minimum embedding dimension from a scalar time series"*Physica D: Nonlinear Phenomena Vol 110 pp 43-50 December 1997.
[9]. Takens.F "*Detecting Strange attractors in turbulence"* Lecture notes in Mathematics 898 Springer Verlag New York pp 360-381. 1980.
[10]. Takens F *"On the numerical determination of dimension of an attractor* "Lecture notes on MathematicsVol 898 Springer Verlag pp230-241, 1981.
[11]. 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.
[12]. 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
[13]. Geist K U Parlitz and W Lauterborn"*Comparison of different methods for computing Lyapunov exponents* "Progress of theoretical Physics 83-5 pp 875-893 1990.
[14]. 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 2004
[15]. Min Han,Jianhui Xi,Shiguo Xu "*Chaotic system identification based on Kalman filter* "Proceedings of the international joint conference on neural networks Vol 1 pp 675-680 .2002.
[16]. Farmer J D, J J Sidorovich "Predicting chaotic time series" Physical Review Letters Vol 59 pp843-848 , 1987.