International Journal of Electrical, Electronic and Communication Sciences
ISSN: 2517-9438
Vol:12, No:12, 2018

Investigation of Improved Chaotic Signal Tracking by Echo State Neural Networks and Multilayer Perceptron via Training of Extended Kalman Filter Approach

Farhad Asadi, S. Hossein Sadati

***Abstract*—**This paper presents a prediction performance of feedforward Multilayer Perceptron (MLP) and Echo State Networks (ESN) trained with extended Kalman filter. Feedforward neural networks and ESN are powerful neural networks which can track and predict nonlinear signals. However, their tracking performance depends on the specific signals or data sets, having the risk of instability accompanied by large error. In this study we explore this process by applying different network size and leaking rate for prediction of nonlinear or chaotic signals in MLP neural networks. Major problems of ESN training such as the problem of initialization of the network and improvement in the prediction performance are tackled. The influence of coefficient of activation function in the hidden layer and other key parameters are investigated by simulation results. Extended Kalman filter is employed in order to improve the sequential and regulation learning rate of the feedforward neural networks. This training approach has vital features in the training of the network when signals have chaotic or non-stationary sequential pattern. Minimization of the variance in each step of the computation and hence smoothing of tracking were obtained by examining the results, indicating satisfactory tracking characteristics for certain conditions. In addition, simulation results confirmed satisfactory performance of both of the two neural networks with modified parameterization in tracking of the nonlinear signals.

***Keywords*—**Feedforward neural networks, nonlinear signal prediction, echo state neural networks approach, leaking rates, capacity of neural networks.

I. Introduction

NEURAL networks methodology is an artificial computational approach developed based on the biological neuronal function and structure. These networks have nodes or neurons that are expressed by differential equations in discrete form. There are different kinds of neural networks some of which are used and developed in the machine learning field, while others are employed in the neuroscience and bioengineering fields.

The notion of Recurrent Neural Networks (RNNs) is a specific and important type of neural network owing its abilities to those of the biological neural networks [1], [2] and is used in many practical applications like detection of human diseases [3], [4], characterizing human control of dynamical systems [5], [6], agriculture [7], [8], and many other fields. ESN is known as a certain type of RNN. An interesting feature of ESN is its high feedback structure, leading to its rich inherent memory. In signal processing, there are different potential applications for ESN algorithm. In many physiological signal processing applications, different simultaneous sources of noises or interferences are mixed and contaminated the original signals like fetal electrocardiogram (ECG) [9], [10] or adult ECG [11].

Features of the original signal are classified or extracted by the ESN or by a combination of ESN and other computational algorithms [12]-[14]. Since this algorithm has the ability for tracking the predefined signals like gait recognition, in different mode and context of tasks for hip, ankle and knee joints, it can be used for saving and then tracking the various patterns of CPG networks by considering the different frequency shapes and synchronization inside the network [15], [16]. In recent years, various ESN and feedforward neural networks have been applied for time series prediction and optimization of systems [17].

In some applications involving investigation of temporal characteristics of dynamic systems, echo estate network and MLP networks are proposed. For the analysis of temporal dynamics of signals or systems, RNN or time delayed neural networks have practical features, but training of this type of networks is developed in recent years [18], [19].

Some references are used in the backpropagation through time for training of networks, but this learning approach suffers from slow learning process and may make a network causing instability which depends on topology of networks [20]. Another approach, known as reservoir computing method, for learning of RNN is developed in [21]. In this approach one recurrent topology is first created. Then by defining predetermined initial weights or functions for the hidden layer, its dynamics are driven by the input layer. The training phase in this approach consists of linear methods like regression and pseudo inverse approach, so a training phase has very low computational cost.

While in the learning phase of the neural network, if the input and output of the network data are available, the training approach is supervised. Yet, the choice of the free parameters as well as the network structure used in the learning algorithm must be chosen carefully. Reference [22] used the fixed point algorithm for the prediction of nonlinear and chaotic signal, leading to a fast rate of learning. Also [23] compared the performance of feedforward and ESN with Mackey-Glass nonlinear chaotic time series. Regressive moving average

Farhad Asadi has Master degree in Mechanical engineering in dynamic and control, K. N. Toosi, Tehran, Iran (phone: +989357430949, e-mail: f.asadi@mail.kntu.ac.ir).
S. Hossein Sadati is Assistant professor in Mechanical Engineering at K. N. Toosi University of Technology, Tehran, Iran (e-mail: sadati@kntu.ac.ir).

filter is applied to the hidden layer of the network and it enhances the prediction performance of both networks.

Generally, for tracking and prediction of nonlinear signals, estimator of algorithm and weights in networks should have features like minimum variance criterion and EKF approach is usable and practical for this index. For applying EKF, a priori distribution and statistical characteristics of measurement and process noise is needed. In other conditions when the process noise of system is large subsequently KALMAN gain is increased accordingly. This subject makes algorithm more sensible to noise of measurement and leads to abrupt changes in weight of networks. For improving EKF approach for estimating the statistical noise characteristics, different methods like adaptive updating covariance of process and measurement noise are proposed in literature [24]. These references used different manual control parameters that can control amount of the weights of variance and also is suitable for basis and input function selection for layers of network during the computation and can enhance the tracking performance as well. In this paper, noise statistical characteristic does not vary by time; and EKF approach is used to train feed forward and ESN neural networks in order to predict or track nonlinear signal and minimize error between model and prediction.

This paper is structured as follows: Section II gives an overall structure of algorithm in both ESN and feed forward network with training by EKF approach. Next, in Section III simulation of ESN neural network at different conditions are discussed and finally, in Section V simulation results for MLP neural networks with training of EKF approach for tracking chaotic signal are investigated.

II. Formulization and Description of Dynamics of ESN and Training with EKF Approach

In this section, overall structure of training with EKF approach is explained and then dynamics of the ESN network and major parameters for controlling performance of algorithm are explained. In this section the overall structure of algorithm for training of neural networks is organized to the state space dynamical form. This representation can be defined as (1):

$$h_{k+1} = h_k + \eta_k \text{ , } y_k = g_k(h_k, x_k) + z_k \tag{1}$$

In (1), k denotes discrete sampling time and z_k is measurement noise that corrupted measurement signal and η_k is process noise. In this algorithm z_k is modeled as an uncorrelated Gaussian distribution with associated covariance of N_k and η_k is a stochastic component of unknown input signal in our modeling, and it is related to covariance which is represented byQ_k. In this algorithm, g_k is an approximation mapping during computation that is obtained and tracked by neural networks and then appropriate weights of network are calculated. This calculation can be formulized by hypothesis space that probability of each term in formulization can be obtained and computed by Bayesian approach. Generally, this computation is formulated as in (2):

$$\text{P } (N_{k+1}, Q_{k+1} | y_{k+1}) = \frac{p(y_{k+1}|y_k, N_{k+1}, Q_k)}{p(y_{k+1}|y_k)} p(N_{k+1}, Q_k | y_k \tag{2}$$

For having smooth tracking and also benefit to choosing estimation of neural weights, EKF approach is usable and beneficial. This algorithm uses Taylor series expansion around previous estimation and consequently terms of bays rule are defined as in (3), (4):

$$\text{Prior=p } (h_{k+1}|y_k, Q_k, N_K) \tag{3}$$

$$\text{likehood} = \mathrm{p(y_{k+1}|h_{k+1}, N_{k+1}}, Q_k) \tag{4}$$

High order terms of series are ignored in this approach so EKF is an approximate method but it has better and faster performance like second order statistical or gradient descent approaches. Also, posterior density function is computed as in (5):

$$\text{Posterior=p } (h_{k+1}|\mathrm{h_{k+1}, N_{k+1}}, Q_k) \tag{5}$$

where weights, KALMAN gains parameter and covariance parameter are formulated as (6)-(8):

$$h^{\wedge}_{k+1} = h^{\wedge}_k + k_{k+1}(y_{k+1} - g(h_k, x_{k+1}) \tag{6}$$

$$p_{k+1} = p_k + Q_k - k_{k+1} g_{k+1}(p_k + Q_k) \tag{7}$$

$$k_{k+1} = (p_k + Q_k) g^T_{k+1} [N_{k+1} + g_{k+1}(p_k + Q_k) \ g^T_{k+1}]^{\wedge} - 1 \tag{8}$$

In addition to consider the above formulation, the gain of the filters is tuned in respect to different noises and the effects of the measurement on the performance of the filter, which are investigated before, are applied in the modelling as well [25]. Finally, by back propagating output error to equations in each sampling time, a weight vector for neural networks is computed.

In the ESN the layers consist of input, hidden layer and output layer that are represented as in (9)-(13):

$$\text{U (n)} = (u_1\text{(n)}\ldots u_k(n))' \tag{9}$$

$$\text{X (n)} = (x_1\text{(n)}\ldots x_k(n))' \tag{10}$$

$$\text{Y (n)} = (y_1\text{(n)}\ldots y_k(n))' \tag{11}$$

$$W^{in} = (W^{input}_{ij}), \ \ \text{w} = (w_{ij}) \tag{12}$$

$$W^{output} = (W^{output}_{ij}), \ W^{back} = (W^{back}_{ij}) \tag{13}$$

W^{output} and W^{in} are its associated connection weights for output and input layer. The activation function for neurons in hidden layer or updating cycle for these nodes is defined as in (14):

$$\text{X (N+1) =H } (W^{IN} U(N+1) + WX(N) + W^{BACK} D(N)) \tag{14}$$

where N=1, …, T is a neuron number in hidden layer and h=

(h_1… h_n) are basis activation function for each neuron. In these simulations we considered hyperbolic tangent function computed as in (15):

$$\text{Hyperbolic tangent function}=\frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \tag{15}$$

Then output of network is calculated as:

$$\text{Y (n+1)} = n^{out}(w^{out}(\text{u (n+1), x (n+1), y (n)})) \tag{16}$$

where $n^{out} = (n_1^{out}\ldots n_n^{out})$ are basis output function.

Generally, the summary of dynamics of neurons in hidden layer can be written as in (17)-(19):

$$x_{input} = Wx_t + W^{INPUT}U_t \tag{17}$$

$$x_{t+1} = \tanh(\,x_{t+1}) \tag{18}$$

$$y_t = W^{output}x_t \tag{19}$$

Finally, outweights are calculated by applying pseudo inverse matrix of M as in (20) and then these systems of equation during the teaching period are solved simultaneously.

$$\mathrm{W}^{\mathrm{OUT}} = (\mathrm{M}^{+}\mathrm{T})^{\mathrm{T}} \tag{20}$$

III. Simulation Results for ESN Neural Network Approach

Dynamical computational models need the feature vector for access and storing the time history for their input signals and related outputs. In ESN network, input signal dynamics is stored in hidden layer and this historic of input is processed by recurrent weight matrix in the hidden layer. Then by combination of different responses of neurons in hidden layer, an output response is obtained. ESN does not necessarily apply gradient learning approach and most of the time training approach in hidden layer of network does not take place and only assigned weights to neuron in this layer are added during computation time for reaching to the desired output signal. In this and next sections, some signals are used which are highly chaotic or are added by noise disturbance.

First structure of ESN network is as follows: Number of neuron in hidden layer is 1000 and internal weights of network are considered randomly in range of one to negative one value and spectral radius of network is defined as 0.79 and finally output layer has one node. Input nonlinear and chaotic signal for inputting to network is plotted in Fig. 1. Then, ESN is trained by 1000 time steps that 100 of it are regarded to damping the initial conditions.

In the first simulation the coefficient of activation function in hidden layer is considered 0.99 and tracking performance by network is plotted in Fig. 2. Then value of this susceptible and important parameter is changed to 0.01 and 0.3 respectively and corresponding simulation results are plotted in Figs. 3 and 4. This parameter has features like the control ability on dynamic activity of neurons in hidden layer and has role such as influencing of input and history of input during the computation. So as evident parameter t in Fig. 3, this parameter makes network lose the history of input and as a result, network cannot track and follow the input. Normalized square error (NSE) and different value of activation function is explained in Table I.

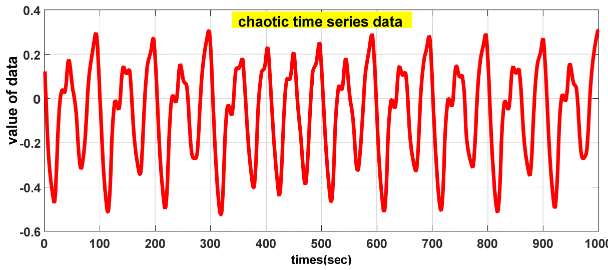

Fig. 1 Input nonlinear and chaotic signal to ESN network

TABLE I
COEFFICIENT OF BASIS FUNCTION IN HIDDEN LAYER AND NSE ERROR

Coefficient of activation function in hidden layer	Error of tracking
A=0.01	**NSE = 0.047689**
A=0.3	**NSE = 0.0044617**
A=0.99	**NSE = 1.7066e-06**

Finally, the associated variation of weights for ESN network for each simulated case is plotted in Fig. 5 and amplitude and variation of weights in second simulation is more than others.

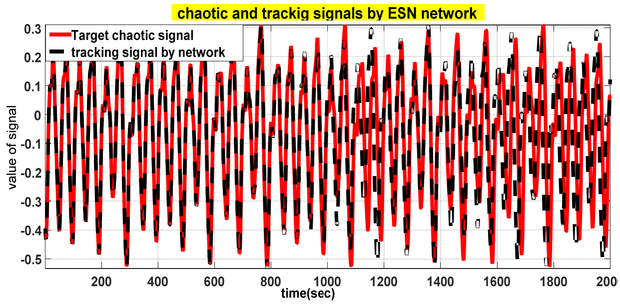

Fig. 2 Target and tracking signal by ESN networks with applying A=0.99 at first simulation

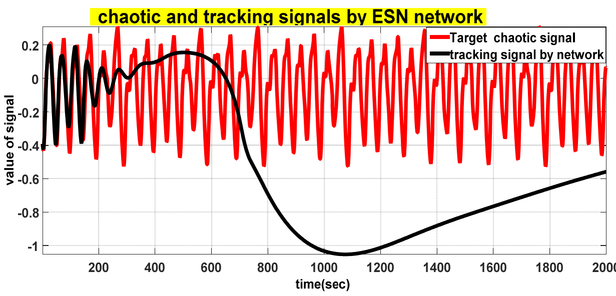

Fig. 3 Target and tracking signal by ESN networks with applying A=0.01 in second simulation

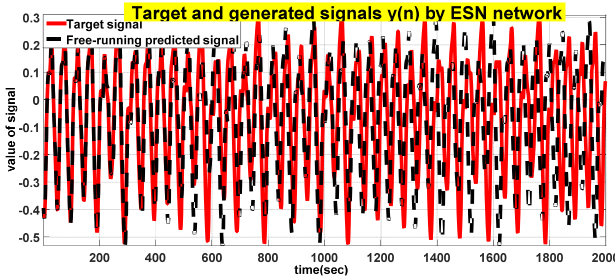

Fig. 4 Target and tracking signal by ESN networks with applying A=0.3 in third simulation

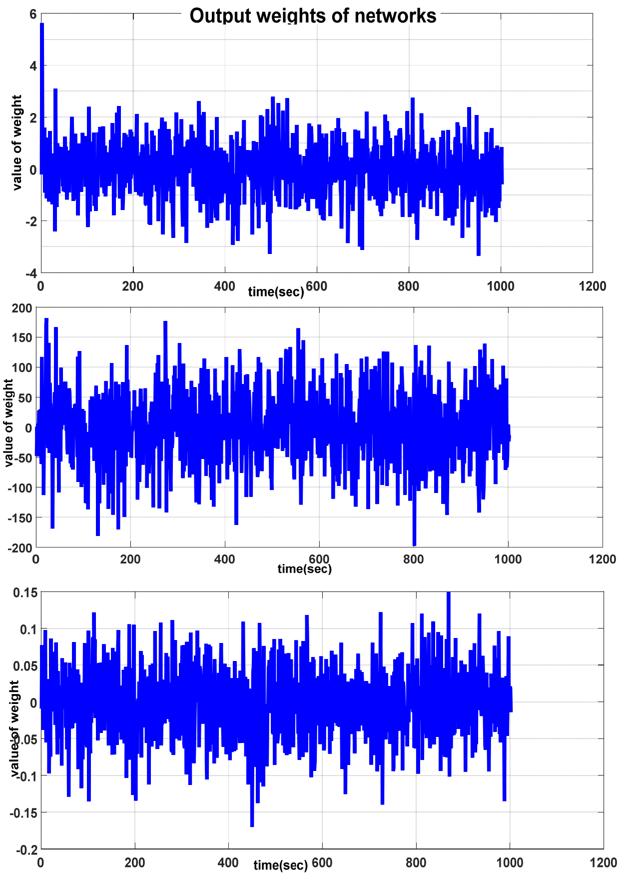

Fig. 5 Output weight variation during the computation according to simulation cases

For manifesting the role of activation function in former simulation in this section, we chose value of 0.01 for activation function while adding to noise and nonlinearity of input signal. In this case, tracking cannot be done at 450 time step by networks despite of enough neurons and initialization of network and this result is plotted in Fig. 6. Also neuronal responses of some nodes in hidden layer of network are plotted in Fig. 7 and associated weight variation during computation is plotted in Fig. 8.

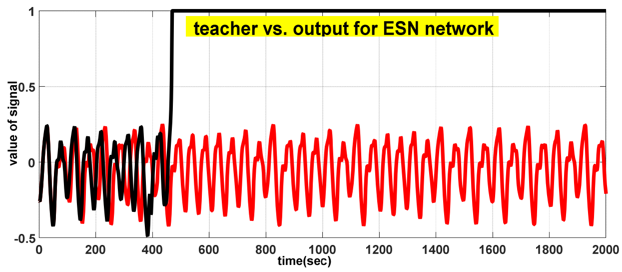

Fig. 6 Teacher and tracking signal for Lorentz signal and applying large noise

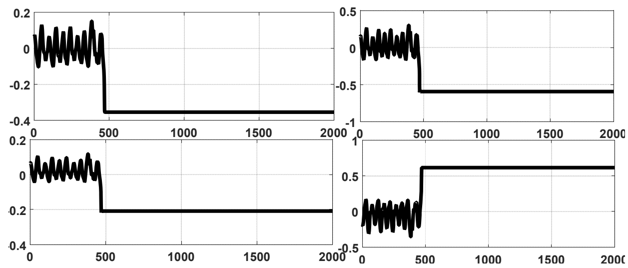

Fig. 7 Neuronal responses variation for some nodes in hidden layer of network

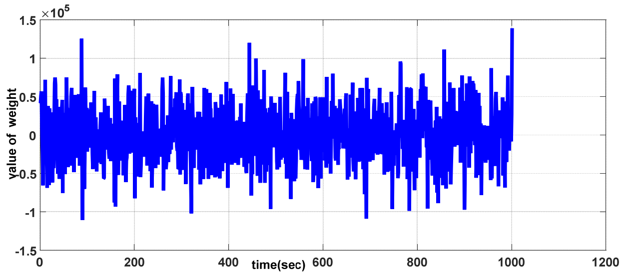

Fig. 8 Output weight variation during the computation

In this section, input signal is divided to 700 time steps that 350 of it are considered for training and others for testing. Value of input signal are normalized and scaled to 0.1. Firstly, we chose spectral radius to value of 0.5 and other defined characteristics is explained in Table II. The training results for this simulation are plotted in Fig. 9 and then testing process of network is plotted in Fig. 10.

TABLE II
DEFINED CHARACTERISTIC FOR FIRST AND SECOND SIMULATION

Number of neurons in input layer	**2**	spectral radius of network for first and second simulation	(first) **0.5** (second) 0.1
Number of neurons in hidden layer	(first)**20** 30 (second)	Activation function for hidden layer	tanh=$\frac{e^x-e^{-x}}{e^x+e^{-x}}$
Number of neurons in output layer	**1**	Method of calculating of error in network	Error mean squared
Value of normalizing input signal	**0.1**	Time step For training	350
		Time step for testing	350

International Journal of Electrical, Electronic and Communication Sciences
ISSN: 2517-9438
Vol:12, No:12, 2018

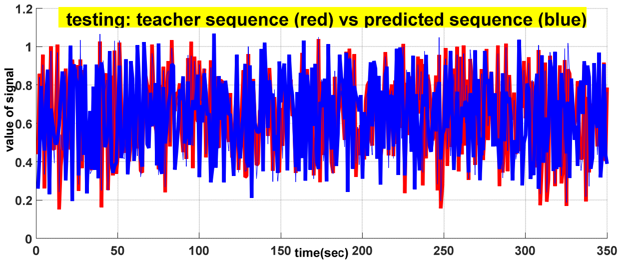

Fig. 9 Teacher and tracking signal for network during training process at first simulation

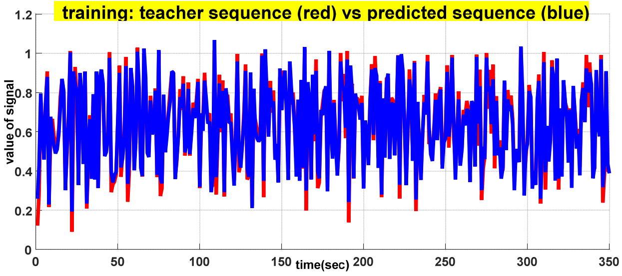

Fig. 10 Teacher and tracking signal for network during testing process at first simulation

TABLE III
DIFFERENT TRACKING OF NETWORK IN DIFFERENT SIMULATION RESULTS

Input span to network	NSE during training	NSE during testing	Input span to network	NSE during training	NSE during testing
900	**0.50035**	**0.52817**	1040	**0.46845**	**0.48579**
920	**0.29102**	**0.3338**	1060	**0.55745**	**0.60161**
940	**0.7251**	**0.77473**	1080	**0.32492**	**0.36255**
960	**0.53705**	**0.60396**	1110	**0.27872**	**0.27945**
980	**0.44059**	**0.43435**	1120	**0.647**	**0.67306**
1000	**0.69436**	**0.74189**	1140	**0.43074**	**0.45841**
1020	**0.41532**	**0.4848**	1200	**0.44298**	**0.51461**

Generally, choosing the initial and spectral radius of hidden layer is identified as the role of input signal compared to the eigenvalue of weight matrix in network. By choosing low value of this number the sensibility of network to input noise is increased. In the second simulation section, spectral radius of network is reduced to 0.1 and value of number of neuron in hidden layer is increased to value of 30. Teacher and tracking signals network for training and testing step are plotted in Figs. 11 and 12, respectively. The rate of convergence in ESN network is defined as in (21):

$$\text{D}\ (h_t,y_t) <= \text{a}\lambda^t \tag{21}$$

where λ is lower of one and parameters of *a* is defined by matrix of *W* in network. This implies convergence rate of network that is obtained by exponential rate. Network size and leaking rate are important factors for both increasing capacity of network and evolving the neuron weight over running the algorithm time. From the second simulation results, it is evident that despite of increasing of neurons in hidden layer, an influence of spectral radius is more vital in noisy and nonlinear input condition. For showing this subject, we increased the span of input from 900 to 1200 and error of tracking in different training and testing processes is obtained and explained in Table III.

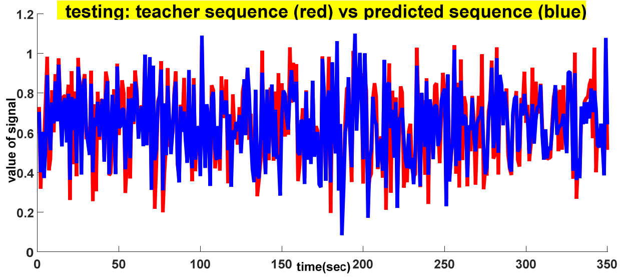

Fig. 11 Teacher and tracking signal for network during training process at second simulation

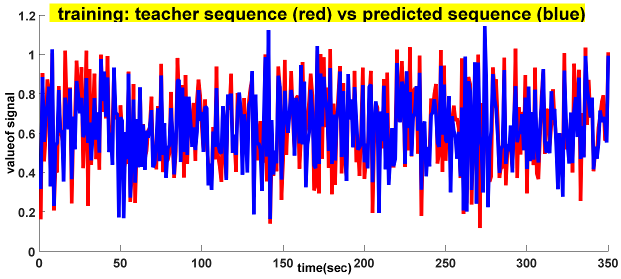

Fig. 12 Teacher and tracking signal for network during testing process at second simulation

IV. SIMULATION RESULTS FOR MLP NEURAL NETWORK APPROACH

In this section, different input and output signals are used for evaluating EKF training approach for MLP neural network. For first simulation, we proceed with MLP neural network with having 10 and one sigmoid neurons function and also linear basis function for hidden and output layer of network, respectively. For updating p(k) in KALMAN filter, back propagation error for updating is used. Also, in this section for comparing the performance of different set up for networks, the NSE criterion is used and it can be defined as in (22):

$$\text{NSE}=\sqrt{\sum_{K=1}^{100}(y_k - g_k^{\wedge}(h_k,x_k)} \tag{22}$$

These nonlinear signals elationship for input and output signals of network in first and second simulation is as in (23), (24) also for third and fourth simulation is according to (25), (26):

$$\text{H (t, 1)} = 0.5\times\text{h (t-1, 1)} + 12\times\cos(9.2(\text{t-1}) + \eta_k \tag{23}$$

$$\text{Y (t, 1)} = \frac{h(t,1)^2}{20} + 21\times\cos(9.2(\text{t-1}) + z_k \tag{24}$$

$$\text{h(t,1)}= 0.5\times\text{h(t-1,1)}+\frac{25\times\text{h(t-1,1)}}{1+h(t-1,1)^2}+8\times\cos(1.2(\text{t-1})+\frac{h(t,1)^2}{20}+\eta_k \tag{25}$$

$$Y(t, 1) = \frac{h(t,1)^2}{20} + 20 \times \cos(1.2(t-1) + 0.5 \times h(t-1, 1) + z_k \quad (26)$$

h (t) denotes input signal and y (t) denotes output signal over 100 time samples. The first related input and output signal for (23) and (24) are plotted in Figs. 13 and 14, respectively and the first related input and output signal for (25) and (26) are plotted in Figs. 20 and 21, respectively. In these simulations the Gaussian noise distribution with variance deviation of 0.1 for process noise signal and periodical deviation distribution with amplitude of 4 is considered for measurement noise signal. These results are plotted in Fig. 15 for first and second simulation and then for third and fourth simulation are plotted in Fig. 22. Next simulation result for true and tracking signal by neural networks at different simulation steps are plotted in Fig. 16 for first simulation, Fig. 18 for second simulation, Fig. 23 for third simulation and Fig. 25 for fourth simulation. Finally, covariance computation with algorithm during computation is plotted for first simulation in Fig. 17, for second simulation in Fig. 19 and for third simulation in Fig. 24, respectively.

In tracking measurement with different noise distribution, KALMAN filter is an estimator with having feature of minimum variance minimization. So this approach leads to lower tracking error and better weight for network, so balancing between tracking and reducing error in network can be gained. In simulation when periodicity behavior of measurement and process noise in second simulation is largely corrupted then, tracking performance of network does not change substantially but when the irregularity and nonlinearity of signal in input and other sources signals to network are increased, the performance of network is reduced. EKF errors are explained in Table V for these conditions. In these simulations it is evident that tracking measurement signal, when variance is suddenly varied, is reduced and network cannot track signals. In this condition due to the local accuracy of EKF model, the increasing of neural node in hidden layer cannot enhance performance of the neural network. Also no smoothness pattern of covariance in simulation by increasing the nonlinearity is evident in simulation results and this is a disadvantage of EKF approach in nonlinear and chaotic conditions that convergence cannot be obtained. Totally, in the first and second simulations, when input and output signal has periodical pattern regardless of measurement and process noise, the tracking performance is smoother and can enhance the training of MLP neural networks for prediction. In addition, having artificial patterns or data series with having more uncertainties like irregular variances, chaotic features and high input dimensions or distribution can cause reducing the desired performance and limitation of the approaches to track these zones [26], [27]. So, in these circumstances, the iteration of algorithm should be increased or new criteria or other robust approaches should be combined with algorithm in effort to enhance the identification or tracking in these noisy zones [26].

TABLE V
DIFFERENT TRACKING OF NETWORK IN DIFFERENT SIMULATION RESULTS

	EKF error	Neurons in the hidden layer
SIMULATION I	1.909493e+01	10
SIMULATION II	3.930239e+01	10
SIMULATION III	9.609864e+01	10
SIMULATION V	1.145457e+02	50

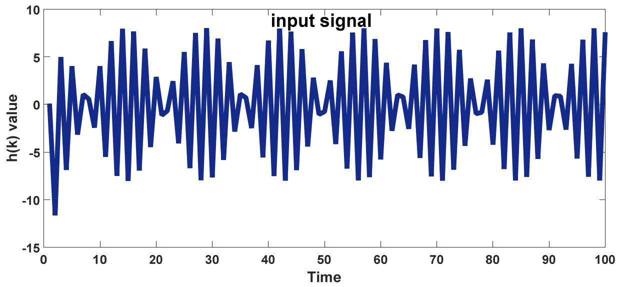

Fig. 13 Input signal for neural network at first and second simulation

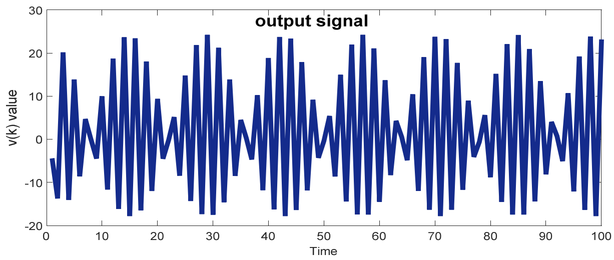

Fig. 14 Output signal for neural network at first and second simulation

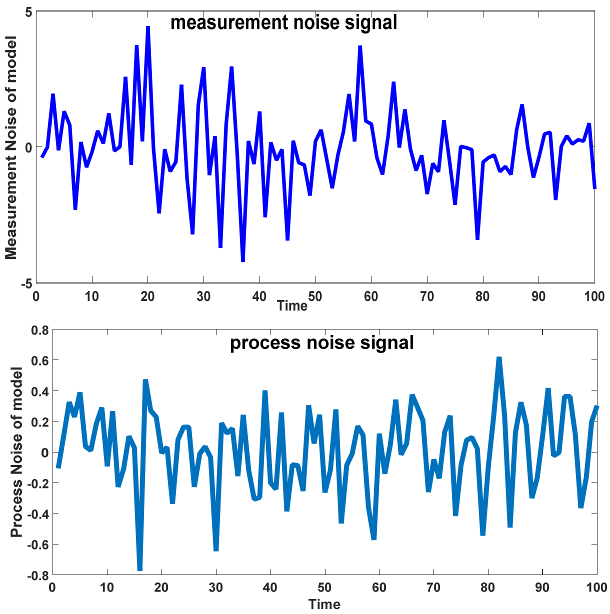

Fig. 15 Different predefined signal for MLP neural network consists of measurement and process noise

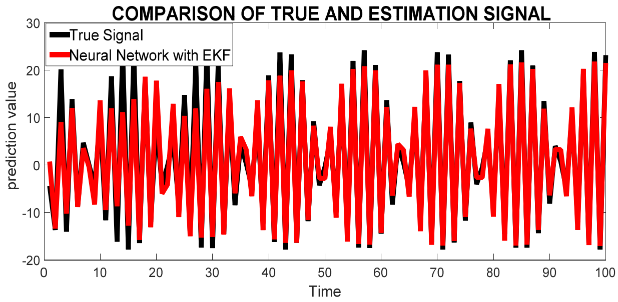

Fig. 16 Simulation result for true and tracking signal by neural networks at first simulation

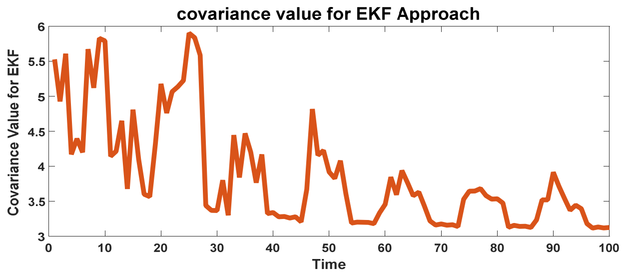

Fig. 17 Covariance computation with algorithm at first simulation

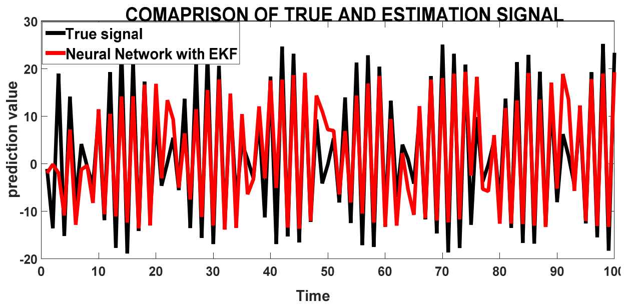

Fig. 18 Simulation result for true and tracking signal by neural networks at second simulation

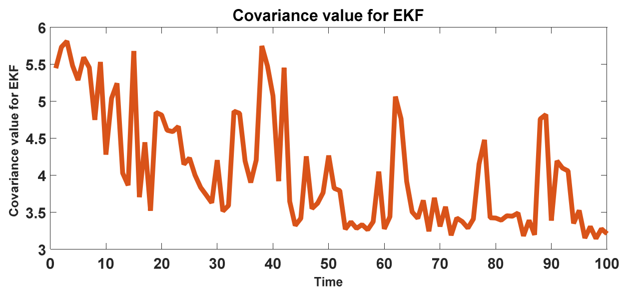

Fig. 19 Covariance computation with algorithm at second simulation

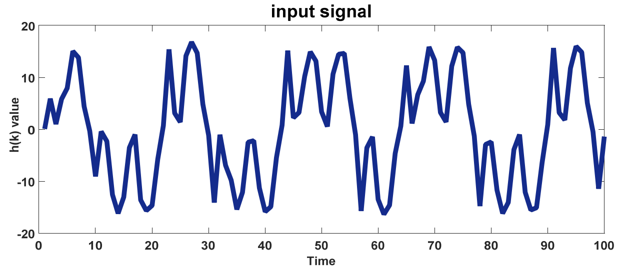

Fig. 20 Input signal for neural network at third and fourth simulation

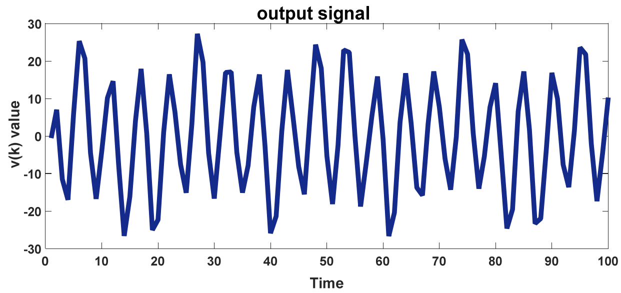

Fig. 21 Output signal for neural network at third and fourth simulation

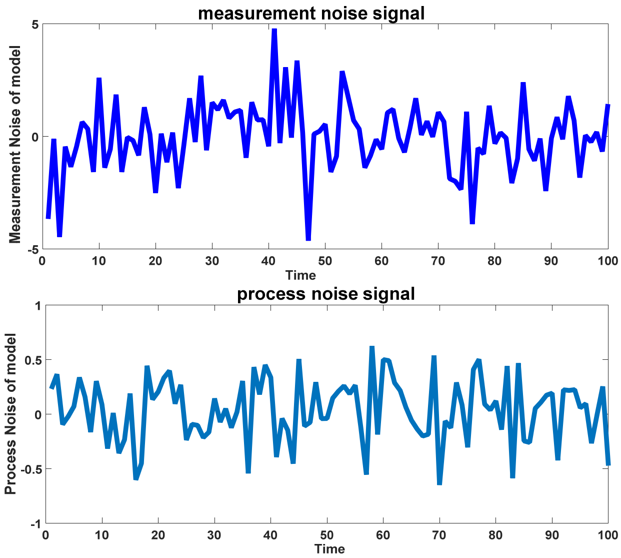

Fig. 22 Different predefined signal for MLP neural network consists of measurement and process

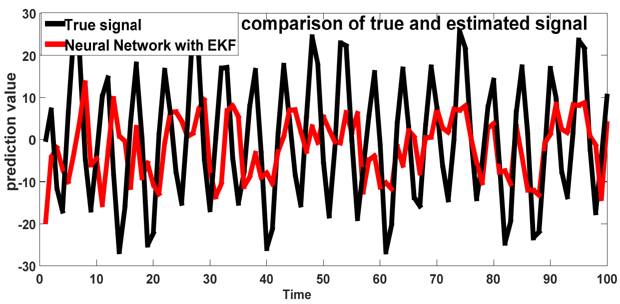

Fig. 23 Simulation result for true and tracking signal by neural network at third simulation

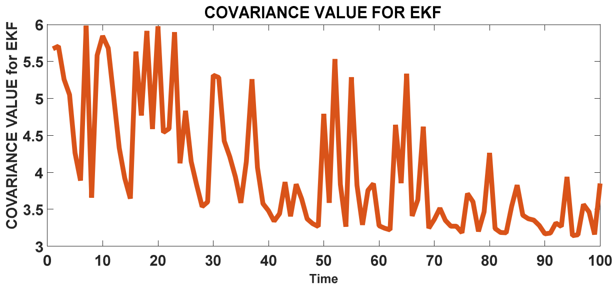

Fig. 24 Covariance computation with algorithm at third simulation

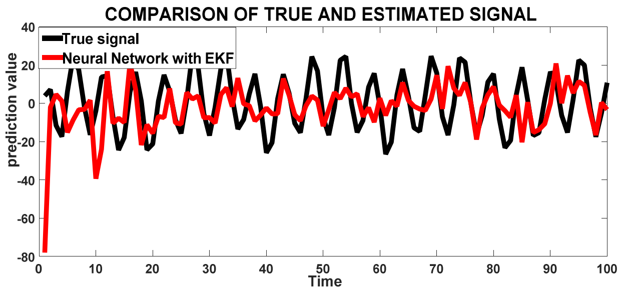

Fig. 25 Simulation result for true and tracking signal by neural networks at fourth simulation

V. Conclusion

Many engineering and physical modeling leads to the nonlinear and chaotic signal so prediction of these signals is vital. In this study the performance of ESN and MLP neural networks for tracking nonlinear signal is proposed and developed. Different simulation results for both neural networks are designed and error of tracking in conditions when key parameters of modeling are changed is investigated as well. With these simulations, limitations and increasing the ability of neural network for tracking is discussed in detail according to the results.

References

[1] R. Laje, D. V. Buonomano, "Robust timing and motor patterns by taming chaos in recurrent neural networks," *Nature neuroscience*, 2013, 16(7), pp. 925.

[2] M. Lukoševičius, "A practical guide to applying echo state networks," *Neural networks: Tricks of the trade*: Springer, 2012. pp. 659-86.

[3] F. Asadi, M. J. Mollakazemi, S. A. Atyabi, I. Uzelac, A. Ghaffari, "Cardiac arrhythmia recognition with robust discrete wavelet-based and geometrical feature extraction via classifiers of SVM and MLP-BP and PNN neural networks," in *Computing in Cardiology Conference (CinC), 2015* 2015, IEEE.

[4] F. Asadi, M. J. Mollakazemi, S. Ghiasi, S. H. Sadati, "Enhancement of life-threatening arrhythmia discrimination in the intensive care unit with morphological features and interval feature extraction via random forest classifier," in *Computing in Cardiology Conference (CinC), 2016* 2016, IEEE.

[5] S. A. S. Mousavi, X. Zhang, T. Seigler, J. B. Hoagg, "Characteristics that make dynamic systems difficult for a human to control," in *American Control Conference (ACC), 2016* 2016, IEEE.

[6] F. Matveeva, S. A. S. Mousavi, X. Zhang, T. Seigler, J. B. Hoagg, "On the effects of changing reference command as humans learn to control dynamic systems," in *Decision and Control (CDC), 2016 IEEE 55th Conference on* 2016, IEEE.

[7] A. Hamidisepehr, M. P. Sama, "A low-cost method for collecting hyperspectral measurements from a small unmanned aircraft system," in *Autonomous Air and Ground Sensing Systems for Agricultural Optimization and Phenotyping III* 2018, International Society for Optics and Photonics.

[8] A. Hamidisepehr, M. P. Sama, A. P. Turner, O. O. Wendroth, "A Method for Reflectance Index Wavelength Selection from Moisture-Controlled Soil and Crop Residue Samples," *Transactions of the ASABE*, 2017, 60(5), pp. 1479-87.

[9] A. Ghaffari, M. J. Mollakazemi, S. A. Atyabi, M. Niknazar, "Robust fetal QRS detection from noninvasive abdominal electrocardiogram based on channel selection and simultaneous multichannel processing," *Australasian physical & engineering sciences in medicine*, 2015, 38(4), pp. 581-92.

[10] M. Mollakazemi, F. Asadi, M. Tajnesaei, A. Ghaffari, "Fetal QRS Detection in Noninvasive Abdominal Electrocardiograms Using Principal Component Analysis and Discrete Wavelet Transforms with Signal Quality Estimation," *Journal of Biomedical Physics and Engineering*, 2016, pp.

[11] M. J. Mollakazemi, S. A. Atyabi, A. Ghaffari, "Heart beat detection using a multimodal data coupling method," *Physiological measurement*, 2015, 36(8), pp. 1729.

[12] D. Brezak, T. Bacek, D. Majetic, J. Kasac, B. Novakovic, "A comparison of feed-forward and recurrent neural networks in time series forecasting," in *Computational Intelligence for Financial Engineering & Economics (CIFEr), 2012 IEEE Conference on* 2012, IEEE.

[13] M. Khashei, M. Bijari, "A novel hybridization of artificial neural networks and ARIMA models for time series forecasting," *Applied Soft Computing*, 2011, 11(2), pp. 2664-75.

[14] A. Petrenas, V. Marozas, L. Sörnmo, A. Lukosevicius, "An echo state neural network for QRST cancellation during atrial fibrillation," *IEEE Transactions on Biomedical Engineering*, 2012, 59(10), pp. 2950.

[15] F. Asadi, M. Khorram, S. A. A. Moosavian, "CPG-based gait planning of a quadruped robot for crossing obstacles," in *Robotics and Mechatronics (ICROM), 2015 3rd RSI International Conference on* 2015, IEEE.

[16] F. Asadi, M. Khorram, S. A. A. Moosavian, "CPG-based gait transition of a quadruped robot," in *Robotics and Mechatronics (ICROM), 2015 3rd RSI International Conference on* 2015, IEEE.

[17] B. Choubin, S. Khalighi-Sigaroodi, A. Malekian, Ö. Kişi, "Multiple linear regression, multi-layer perceptron network and adaptive neuro-fuzzy inference system for forecasting precipitation based on large-scale climate signals," *Hydrological Sciences Journal*, 2016, 61(6), pp. 1001-9.

[18] D. Sussillo, "Neural circuits as computational dynamical systems," *Current opinion in neurobiology*, 2014, 25pp. 156-63.

[19] D. Sussillo, M. M. Churchland, M. T. Kaufman, K. V. Shenoy, "A neural network that finds a naturalistic solution for the production of muscle activity," *Nature neuroscience*, 2015, 18(7), pp. 1025.

[20] M. Lukoševičius, H. Jaeger, "Reservoir computing approaches to recurrent neural network training," *Computer Science Review*, 2009, 3(3), pp. 127-49.

[21] Z. Shi, M. Han, "Support vector echo-state machine for chaotic time-series prediction," *IEEE Trans Neural Networks*, 2007, 18(2), pp. 359-72.

[22] M. Čerňanský, P. Tiňo, "Predictive modeling with echo state networks," in *International Conference on Artificial Neural Networks* 2008, Springer.

[23] Y. Xia, B. Jelfs, M. M. Van Hulle, J. C. Príncipe, D. P. Mandic, "An augmented echo state network for nonlinear adaptive filtering of complex noncircular signals," *IEEE Transactions on Neural Networks*, 2011, 22(1), pp. 74-83.

[24] M. J. Mollakazemi, F. Asadi, A. Ghafouri, "The evaluation of the performance of different filtering approaches in tracking problem and the effect of noise variance," *simulation*, 2015, 10pp. 12.

[25] F. Asadi, M. J. Mollakazemi, "Investigation on Performance of Change Point Algorithm in Time Series Dynamical Regimes and Effect of Data Characteristics," *World Academy of Science, Engineering and Technology, International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering*, 2015, 8(10), pp. 1787-93.

[26] F. Asadi, M. Mollakazemi, A. Ghafouri, "The influence of parameters of modeling and data distribution for optimal condition on locally weighted projection regression method," *Accepted and oral presentation in ICMSE*, 2014, pp. 27-8.

[27] M. J. Mollakazemi, F. Asadi, "Real Time Adaptive Obstacle Avoidance in Dynamic Environments with Different DS," *Accepted and oral presentation in ICARM*, 2014, pp. 27-8.