

A NEURAL-NETWORK-BASED APPROACH TO SPEECH SIGNAL PREDICTION

RUI J.P. DE FIGUEIREDO

Key words: Neural networks, Linear predictors.

In this paper a new approach to speech signal prediction is presented. The approach deploys an EKF (Extended Kalman Filter)-based learning algorithm for this application. Simulation results show that the proposed neural network approach leads to better performance than the well-known linear predictive coding (LPC) approach which uses the Levinson-Durbin algorithm for predictor design.

1. INTRODUCTION

Prediction refers to the problem of accurately forecasting the short-term evolution of a system. In modeling, on the other hand, one tries to find a description that accurately captures features of the long-term behavior, while characterization attempts to determine fundamental properties of the time series. Prediction and modeling may seem to have quite similar objectives, but they do not necessarily lead to the same solutions. Long-term modeling is based on the assumption that a time series has been generated by a set of deterministic governing equations in a state-space (also called phase space), and that there is a deep connection between time-lagged vectors and the underlying dynamics. This philosophy provides the basis for the approach to time series prediction known as state-space reconstruction by time-delay embedding [1, 2, 3].

However, the governing equations may not always constitute the most reliable way to determine the parameters for good short-term predictions. The underlying process responsible for generating the data may be genuinely stochastic, and in addition, many physical signals encountered in practice exhibit non-stationary characteristics. These two properties are inconsistent with the state-space approach. Signals that are both nonlinear and non-stationary can in many cases be successfully represented by adaptive nonlinear models in which the parameters are allowed to be continuously adjusted.

Dept. of EECS, "Henry Samueli" School of Engineering, University of California, Irvine, California 92697-2800, Email: rui@uci.edu

In this paper a new approach is presented to speech signal prediction using a particular multi-layer artificial neural network structure. The proposed network deploys an EKF- based learning algorithm, [4, 5]. For this reason we denote the new application-specific neural network as EKF-ML-ANN.

The EKF-ANN represents an instance of the generic Functional Artificial Neural Networks (FANNs) introduced by the author in 1998 [8]. These networks can be viewed as generic ANNs with “dynamical memories”, and therefore the author has referred to them as Dynamical Functional Artificial Neural Networks, or simply D-FANNs.

The new neural predictor has three layers. The first one is the input layer, the second one is referred to as the hidden layer, and the third layer is called the output layer.

The EKF-ML-ANN predictor and its learning algorithm are introduced in Section II. In Section III we illustrate some simulation results. Finally, a brief conclusion is given in Section IV.

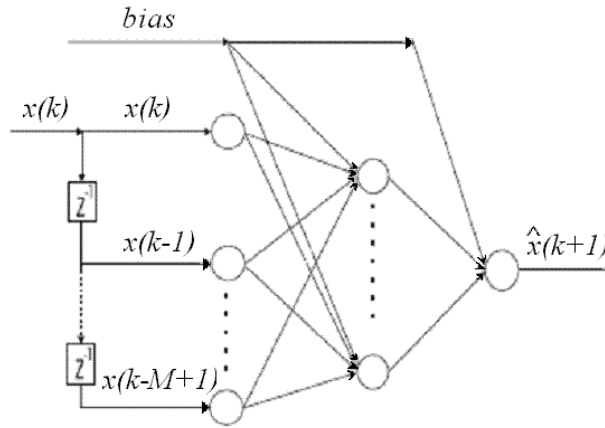


Fig. 1 – The EKF-ML-ANN predictor.

2. EXTENDED-KALMAN-FILTER-BASED MULTI-LAYER ARTIFICIAL NEURAL NETWORK (EKF-ML-ANN)

2.1. NEURAL NETWORK ARCHITETURE

The EKF-ML-ANN predictor (Fig. 1) has three layers. The first layer is called the input layer. The input elements are the present and the previous $M - 1$ samples of the speech signal. Similarly, the third layer is called the output layer and

the output is the prediction of the next speech sample based on the previous speech samples. The second layer is referred to as hidden layer. The activation function of the hidden and output layers is denoted by $\phi(\cdot)$.

Let the j^{th} neuron in the n^{th} layer be denoted as *neuron* (n, j). The following notations are used throughout this paper. For the time instant k , L is the total number of layers, N_n is the total number of neurons in the n^{th} layer, $x(k)$ is the input vector to the neural network, $v_j^n(k)$ is the input to the *neuron*(n, j), $y^n(k)$ is the output vector of the n^{th} layer, $y_j^n(k)$ is the output of the *neuron*(n, j), $d_j(k)$ is the desired output of the j^{th} neuron in the output layer, $y_j^n(k)$ is the synaptic-weight from the *neuron*($n-1, j$) to the *neuron*(n, j) and $\beta_j^n(k)$ is the bias for the *neuron* (n, j).

The output of the *neuron*(n, j), $y_j^n(k)$, is given by evaluating the activation function

$$y_j^n(k) = \phi \left(\sum_{i=1}^{N_{n-1}} \omega_{ji}^n(k) y_i^{(n-1)}(k) + \beta_j^n(k) \right). \quad (1)$$

Moreover, the bias for each neuron can be treated as an extra synaptic-weight from the previous layer by using an additional virtual neuron at the previous layer with its value equal to one. So the equation (1) becomes

$$y_j^n(k) = \phi \left(\sum_{i=1}^{N_{n-1}+1} \omega_{ji}^n(k) y_i^{(n-1)}(k) \right), \quad (2)$$

with $y_{N_{n-1}+1}^{(n-1)}(k) = 1$ and $\omega_{j(N_{n-1}+1)}^n(k) = \beta_j^n(k)$.

The output of the EKF-ML-ANN, y^L , is denoted as $\hat{x}_I(k+1)$ in Fig. 1 and it is the prediction of $x(k+1)$ based on previous M samples.

2.2. LEARNING ALGORITHM

EKF is a method of state vector estimation [4, 6]. Thus, the synaptic-weights of the network are put as a state vector,

$$W = \left[(W^2)^T, \dots, (W^L)^T \right]^T, \quad (3)$$

where

$$W^n = \left[(W_1^n)^T, (W_2^n)^T, \dots, (W_{N_n}^n)^T \right]^T, \quad (4)$$

$$W_i^n = [\omega_{i,1}^n, \omega_{i,2}^n, \dots, \omega_{i,N_{n-1}+1}^n]^T. \quad (5)$$

Then the ML-ANN can be expressed by the nonlinear system equations

$$W(k-1) = W(k), \quad (6)$$

$$d(k) = c(x(k); W(k)) + v(k) = y^L(k) + v(k), \quad (7)$$

where $c(\cdot)$ is a nonlinear functional and $v(k)$ is a zero-mean white noise process with correlation matrix $R(k)$. Applying the EKF to the nonlinear model given by equations (6, 7) gives the following real time algorithm:

$$\hat{W}(k) = \hat{W}(k-1) + K(k)[d(k) - \hat{y}^L(k)], \quad (8)$$

$$K(k) = P(k-1)C(k)^T [C(k)P(k-1)C(k)^T + R(k)]^{-1}, \quad (9)$$

$$P(k) = P(k-1) - K(k)C(k)P(k-1), \quad (10)$$

where $P(k)$ is the inverse of $R(k)$, $K(k)$ is the Kalman gain, and “ $\wedge$ ” denotes the estimate of the corresponding vectors based on the observations up to time $k-1$ (see EKF in [6] for details). $C(k)$ is constructed by

$$\begin{aligned} C(k) &= \left(\frac{\partial y^L(k)}{\partial W} \right)_{W=\hat{W}(k-1)} = \\ &= [C_1^2(k), \dots, C_{N_2}^2(k), C_1^3(k), \dots, C_{N_3}^3(k), \dots, C_1^L(k), \dots, C_{N_L}^L(k)], \end{aligned} \quad (11)$$

with the definition

$$C_i^n(k) = \left(\frac{\partial y^L(k)}{\partial W_i^n} \right)_{W=\hat{W}(k-1)}. \quad (12)$$

$C_i^n(k)$ is computed by

$$C_i^n(k) = \delta_i^n(k) \hat{y}^n(k)^T, \quad (13)$$

where $\delta_i^n(k)$'s are computed with the following recursion

$$\delta_i^n(k) = \begin{cases} \varphi'(v_i^L(k)) \cdot \left[0, \dots, 0, \overset{i^{th}}{1}, 0, \dots, 0 \right]^T, & n = l \\ \varphi'(v_i^L(k)) \sum_m \delta_m^{n+1}(k) \omega_{mi}^{n+1}(k), & \text{otherwise,} \end{cases} \quad (14)$$

where $\varphi'(\cdot)$ denotes the derivative of $\varphi(\cdot)$.

The correlation matrix $R(k)$ and its inverse $P(k)$ are unknown *a priori*. Thus, $P(k)$ is initialized as $\gamma^{-1}I$, where γ is a small constant and I is the identity matrix. Using this initialization and the Matrix Inversion Lemma, the following *algorithm* is created:

1) Define the scalars

$$\xi_i^n(k) = \hat{y}^n(k)^T P_i^n(k-1) \hat{y}^n(k), \quad (15)$$

$$\rho_i^n(k) = \delta_i^n(k)^T \delta_i^n(k). \quad (16)$$

2) Define the Kalman gain $K_i^n(k) = \frac{1}{\lambda + \xi_i^n(k) \rho_i^n(k)} \psi_i^n(k) \delta_i^n(k), \quad (17)$

where λ is a constant less than or equal to one, and $\psi(\cdot)$ is given by

$$\psi_i^n(k) = P_i^n(k-1) \hat{y}^n(k). \quad (18)$$

3) Update the synaptic-weights and the inverse of the correlation matrix, $P(k)$, using

$$\hat{W}_i^n(k) = \hat{W}_i^n(k-1) + \frac{\delta_i^n(k)^T (d(k) - \hat{y}^L(k))}{\lambda + \xi_i^n(k) \rho_i^n(k)} \psi_i^n(k), \quad (19)$$

$$P_i^n(k) = P_i^n(k-1) - \frac{\theta_i^n(k)}{\lambda + \xi_i^n(k) \rho_i^n(k)} \psi_i^n(k) \psi_i^n(k)^T. \quad (20)$$

3. SIMULATION RESULTS

We have compared the performance of the EKF-ML-ANN predictor with the well-known one which uses linear predictive coding (LPC) based on the Levinson-

Durbin algorithm. We have tested both predictors with different lengths of previous samples. Female speech signal sampled at 8 kHz was used to test both systems, Fig. 2. We have summarized the results in Table 1 for convenience. For a quantitative comparison, we used the following performance measure [7].

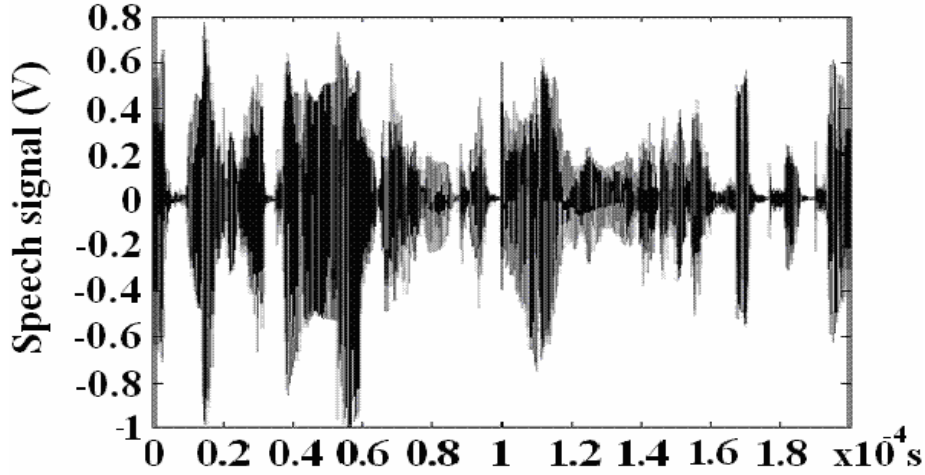


Fig. 2 – Female speech signal sampled at 8 kHz.

$$R_p = 10 \log_{10} \left(\frac{\delta_s^2}{\delta_p^2} \right), \quad (21)$$

where δ_s^2 is the mean square value of the incoming signal, and δ_p^2 is the corresponding value of the prediction error.

Table 1

Comparison of LPC and EKF-ML-ANN

Number of previous samples	LPC Rp	EKF-ML-ANN Rp
3	6.58	8.35
4	7.71	8.27

4. CONCLUSIONS

We proposed a new nonlinear adaptive speech signal predictor based on a multi-layer neural network architecture. The new application-specific network uses an EKF-based learning algorithm. The simulation results showed that the proposed

approach performed better than the well-known linear predictive coding (LPC) approach, which uses the Levinson-Durbin algorithm for predictor design.

Received on 9 August 2008

REFERENCES

1. F. Takens, *Detecting strange attractors in turbulence*, in *Dynamical Systems and Turbulence*, Lecture Notes in Mathematics, **898**, pp. 336-381, Berlin, Springer, 1981.
2. T. Sauer, *A noise reduction method for signals from nonlinear systems*, Physica D, **58**, pp. 193-201, 1992.
3. J. Theiler, *Estimating fractal dimension*, J. Opt. Soc. Am. A, **7**, 6, pp. 1055-1073, 1990.
4. E. Akay, R.J.P. de Figueiredo, *Digital Satellite Channel Equalization using an Extended Kalman Filter Multilayer Artificial Neural Network (EKF-ML-ANN)*, Intl. Workshop on Nonlinear Dynamics of Electronic Systems 2002, pp. 5.5-5.8, June 2002.
5. Y. Iiguni, H. Sakai, H. Tokumaru, *A Real-Time Learning Algorithm for a Multilayered Neural Network Based on the Extended Kalman Filter*, IEEE Transactions on Signal Processing, **40**, 4, pp. 959-966, April 1992.
6. S. Haykin, *Adaptive Filter Theory*, Prentice Hall, Third Edition, 1996.
7. S. Haykin and L. Li, *Nonlinear adaptive prediction of non-stationary signals*, IEEE Trans. Signal Processing, **43**, 2, pp. 526-535, 1995.
8. R. J. P. de Figueiredo, *Optimal interpolating and smoothing functional artificial neural networks (FANNs) based on a generalized Fock space framework*, Circuits, Systems, & Signal Processing, **17**, 2, pp. 271-87, 1998.