TIME DOMAIN RECURSIVE DIGITAL FILTER MODELING BASED ON RECURRENT NEURAL NETWORK TRAINING

Stela Angelova Stefanova

Technology School "Electronic System" associated with the Technical University of Sofia, Mladost 1, 1750 Sofia, Bulgaria, +359 2 8750040, e-mail:sstefanova@elsys-bg.org

Keywords: recursive digital filters, neural networks, optimization

In this paper a time domain recursive digital filter model, based on recurrent neural network is proposed. This problem can be considered as a training procedure of two layer recurrent neural network. The proposed neural network training algorithm is based on determination of the sensitivity coefficients of the recurrent system. The dynamic model of two layer recurrent neural network described by system of recurrent equations is considered. Time domain modeling approach has been applied to design the Nyquist recursive digital filter. Digital filter parameters are obtained by optimization procedure when the requirements to the impulse response in time domain are given.

Modern devices for digital signal processing in particular digital filters represent dynamical systems described with the difference equations. This fact allows the application of neural networks for defining the design problem in time domain, for modeling and realization of the non-recursive (finite impulse response – FIR), recursive (infinite impulse response – IIR) and adaptive digital filters (DF). One approach for the 1-D FIR digital filter design based on the weighted mean square method and neural network to state the approximation problem is proposed in [1]. Some methods for the non-linear digital filters design using neural networks are considered in [2]. Basic results related to the discrete dynamical systems approximation using neural networks are discussed in [3].

In this paper a 1-D IIR digital filter neural network model is proposed. Training sequences of input excitations and corresponding filter responses are generated for the training procedure of this model with given neural network structure. The digital filter model has been trained in such a way that with given predetermined input signal, the output variable approximates the target function in mean square sense. Time domain modeling approach has been applied to design the Nyquist recursive digital filter. Digital filter parameters are obtained by optimization procedure when the requirements to the impulse response in time domain are given.

1. RECURSIVE DIGITAL FILTER MODELING BASED ON NEURAL NETWORK

The neural network structure used for the recursive digital filter modeling is shown in Figure 1.

The recursive or IIR digital filter can be considered as a digital dynamic system described in time domain with n – order difference equation that has been stated using the delayed samples of the excitation input signal and the response signal at the output:

$$v(kT) = \sum_{i=0}^{n} a_i u(kT - iT) - \sum_{i=1}^{n} b_i v(kT - iT) \tag{1}$$

where $a_i, b_i, i = 1..n$ are the IIR digital filter coefficients, $u(kT), v(kT)$ - are the excitation input signal, respectively the response signal at the filter output.

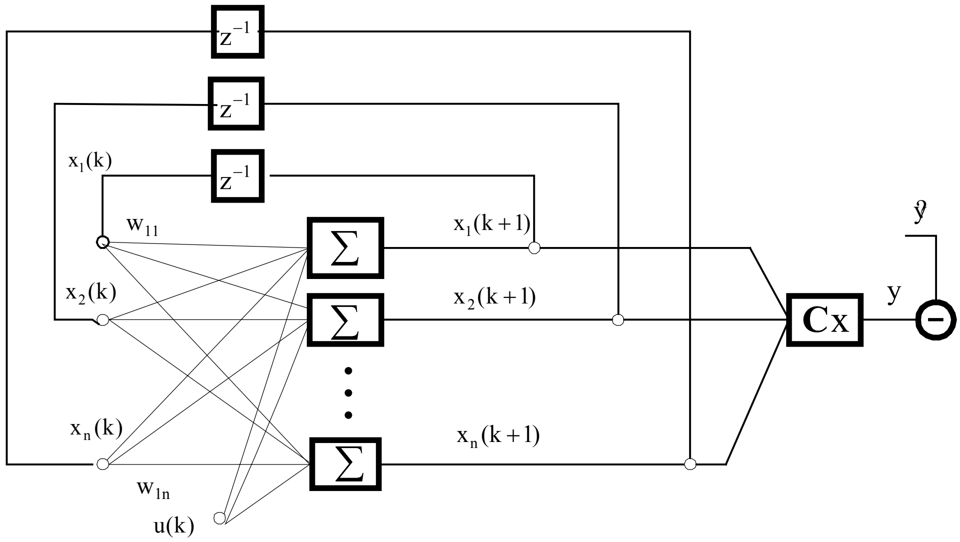

Figure 1. Recurrent neural network structure for digital filter modeling

Transforming the difference equation (1), the state space description of the recursive digital filter can be obtained as a system of n - number first order equations in the form:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ \cdots \\ x_n(k+1) \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} \cdots & w_{1n} \vdots \ldots w_{1n_u} \\ w_{21} & w_{22} \cdots & w_{2n} \vdots \ldots w_{2n_u} \\ \ldots\ldots\ldots & \ldots\ldots\ldots & \ldots\ldots \vdots \; \ldots\ldots \\ w_{n1} & w_{n2} \cdots & w_{nn} \vdots \ldots w_{nn_u} \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ \ldots\ldots \\ x_n(k) \\ u_1(k) \\ \ldots\ldots \\ u_{n_u}(k) \end{bmatrix} \tag{2}$$

$$\mathbf{x}(k+1) = \mathbf{W}\,\mathbf{z}(k) \quad , \quad \mathbf{z} = [x_1, x_2, ..., x_n, u_1, u_2, ..., u_{n_u}]^T \tag{3}$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) \tag{4}$$

where $\mathbf{x} \in \mathbf{R}^n$ is a state space vector, $\mathbf{y} \in \mathbf{R}^m$ and $\mathbf{u} \in \mathbf{R}^{n_u}$ are the dynamic system vectors of the output signals, respectively of the input signals and

$\mathbf{W} \in \mathbf{R}^{n \times n_u}$, and $\mathbf{C} \in \mathbf{R}^{m \times n}$ are matrixes of the system coefficients;

n - a number of neurons at the first layer

n_u - a number of input excitation signals

$n_z = n + n_u$; $\mathbf{C} = \mathrm{diag}(c_1, c_2, ..., c_m)$

The equations (3), (4) can be used to describe the recurrent neural network shown in Figure 1.

The following additional vectors are defined:

- a vector of the neural network weighting coefficients

$$p = \left\lfloor w_{11}\ w_{12}\ldots w_{1n_z}, w_{21}\ w_{22}\ldots w_{2n_z}\ldots . w_{n\,n_z}, c_1, c_2\ldots c_m \right\rfloor \tag{5}$$

where the elements of matrix $\mathbf{W} \in \mathbf{R}^{n \times n_u}$ are introduced row by row;

- a vector of the first layer inputs of neural network

$$z_j = \begin{cases} x_j(k)\,;\, j = 1,2,\ldots,n \\ u_j(k)\,;\, j = n \ \ +1,\ldots,n_z \end{cases} \tag{6}$$

$$\mathbf{z} = [x_1, x_2, \ldots x_n, u_1, u_2, \ldots, u_{n_u}]^T$$

- a vector of the neural network outputs

$$\mathbf{y}(k) = [y_1(k), y_2(k)\ldots y_m\ \ (k)]^T; \tag{7}$$

Mean square error objective function is defined in following form:

$$J(\mathbf{p}) = \ \frac{1}{2}\sum_{i=1}^{m}\ \sum_{k=k_0}^{k_t}[y_i(k) - \hat{y}_i(k)]^2 \tag{8}$$

where k_t is a number of samples and $\{\hat{y}_i(k)\}$ is a target function (the set of experimental data).

Using the results published in [3] the recursive digital filter can be modeling by the one layer recurrent neural network (see Figure 1) with number of neurons corresponding to the digital filter transfer function order. The training procedure of the digital filter model is realized applying the algorithm of the Lagrange multipliers.

2. ALGORITHM OF LAGRANGE MULTIPLIERS

The algorithm of the Lagrange multipliers is used as a training procedure of the digital filter neural network model. The main problem in the neural network training process is the gradient calculation of the mean square objective function (8) with respect to weights w.

The vector of Lagrange multipliers is defined as:

$$\boldsymbol{\lambda}(k) = [\lambda_1(k), \lambda_2(k), \ldots, \lambda_n\ \ (k)]^T$$

and the Hamiltonian of the optimization problem (3),(4), (8) is stated in the form:

$$H = \frac{1}{2}\sum_{j=1}^{m}(y_j(k) - \hat{y}_j(k))^2 + \boldsymbol{\lambda}^T(k+1)\ [x_1(k+1), x_2(k+1), \ldots, x_n\ \ (k+1)\ ]^T + \boldsymbol{\Gamma}^T(k+1)\mathbf{p}(k) \tag{9}$$

Using (9) the conjugated system is composed as follows:

$$\boldsymbol{\lambda}(k) = \frac{\partial H}{\partial x(k)} \qquad \boldsymbol{\lambda}(\kappa_f) = 0 \tag{10}$$

$$\mathbf{\Gamma}(k) = \frac{\partial H}{\partial \mathbf{p}(k)} \qquad \mathbf{\Gamma}(\kappa_f) = 0 \tag{11}$$

The objective function (8) gradient is calculated using the following algorithm:

Step 1. Calculate and store the set of values $\{\mathbf{x}(k)\}$ from (3)

for $\mathbf{x}(k_0) = \mathbf{x}_0;\ k = k_0, k_0 + 1, ..., k_f - 1$

Step 2. Solve the conjugate system (10) и (11)

for $k = k_f - 1, ..., k_0$ (backwards in time)

Step 3. Obtain the objective function gradient from the solution of (11) for k=0

$\nabla J(\mathbf{p}) = \quad \mathbf{\Gamma}(k_0)$

The objective function (8) is minimized applying the standard optimization procedure.

The conjugated system (10), (11) can be written in matrix form. For this purpose it is necessary to define the following sub-matrix of weighting coefficients matrix W:

$$\mathbf{W}_x = \begin{bmatrix} w_{11} & w_{12} \cdots & w_{1n} \\ w_{21} & w_{22} \cdots & w_{2n} \\ \cdots & \cdots & \cdots \\ w_{n,1} & w_{n,2} \cdots & w_{n,n} \end{bmatrix} \tag{12}$$

The error vector from the objective function (8) can be written as:

$$\mathbf{e} = [e_1, e_2, ..., e_m], \quad e_i = y_i - \hat{y}_i, \quad i = 1,2,...,m \tag{13}$$

Equation (10) is stated in matrix form as follows:

$$\boldsymbol{\lambda}(k) = \mathbf{C}\mathbf{e}(k) + \mathbf{W}_x^T \boldsymbol{\lambda}(k+1) \tag{14}$$

where C is the matrix from (4) , $\mathbf{W}_x$ is the matrix from (12) and e is an error vector.

Equation (11) can be stated in the following form:

- for the first layer of the neural network

$$[\mathbf{\Gamma}^w_{col,1}(k), \mathbf{\Gamma}^w_{col,2}(k), ..., \mathbf{\Gamma}^w_{col,n}] = [\mathbf{\Gamma}^w_{col,1}(k+1), \mathbf{\Gamma}^w_{col,2}(k+1), ..., \mathbf{\Gamma}^w_{col,n}] + \mathbf{z}(k)\boldsymbol{\lambda}^T(k+1) \tag{15}$$

- for the second layer of the neural network

$$\mathbf{\Gamma}^c_{col,1}(k) = \mathbf{\Gamma}^c_{col,1}(k+1) + \mathrm{diag}(x_1(k), x_2(k), ..., x_m(k))\mathbf{e}(k) \tag{16}$$

Matrix form statement of the conjugated system (10), (11) allows effective realization of the algorithm for the objective function (8) gradient determination with respect to neural network weights.

After gradient determination the standard optimization procedure is used for minimization of the objective function (8).

3. MODELING RESULTS

The effectiveness of the proposed algorithm is demonstrated by modeling of Nyquist recursive digital filter. Nyquist filters play an important role in digital data transmission for its intersymbol interferece (ISI)-free property. Also they can be adopted in decimation or interpolation multirate systems. To achieve zero ISI, Nyquist filters must satisfy some criteria in time domain that they should have zeros equally spaced in the impulse response coefficients except one specified. Infinite impulse response (IIR) Nyquist filters have lower orders than FIR filters, but their impulse responses are more difficult to keep the zero-crossing time constraint property ant the problem of filter stability should also be examined [4].

Impulse response $h(n)$ of the (IIR) Nyquist filter with the time domain constraints is defined in the form [4]:

$$h(K+kN) = \begin{cases} \frac{1}{N} \neq 0, & \text{if } k=0 \\ 0, & \text{otherwise} \end{cases}$$

where K and N are integers

The transfer function $H(z)$ of the (IIR) Nyquist filter can be expressed as:

$$H(z) = b_K z^{-K} + \frac{\sum\limits_{i=0,\, i\neq kN=K}^{N_n} b_i z^{-i}}{\sum\limits_{i=0}^{N_d/N} a_{iN} z^{-iN}}, \quad b_K = \frac{1}{N}$$

where N_n, N_d are integers, all the filter coefficients a_i, b_i are real, $a_0 = 1$, N_d is the multiple of N.

The impulse response of the IIR Nyquist filter is used as target function in the training procedure of the neural network model. The IIR Nyquist filter with $N_n = 15, N_d = 4$, $N = 4, K = 9$ is considered. The impulse response coefficients of the IIR Nyquist filter are given in Table 1 [4].

Table 1. The impulse response coefficients of the IIR Nyquist filter

b_0	0,003737	b_{11}	0,252947
b_1	0	b_{12}	0,197365
b_2	-0,011246	b_{13}	0,134028
b_3	-0,021744	b_{14}	0,068797
b_4	-0,021263	b_{15}	0,027277
b_5	0	a_0	1,0
b_6	0,0466062	a_1	0
b_7	0,1162339	a_2	0
b_8	0,1904965	a_3	0
b_9	0,25	a_4	0,536111
b_{10}	0,2710056		

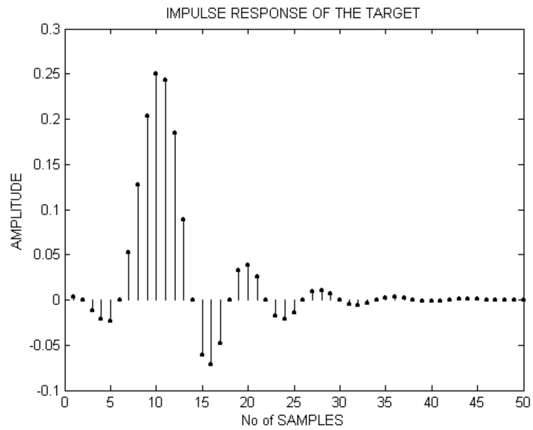

Fig 2a. Impulse responses of the target IIR filter

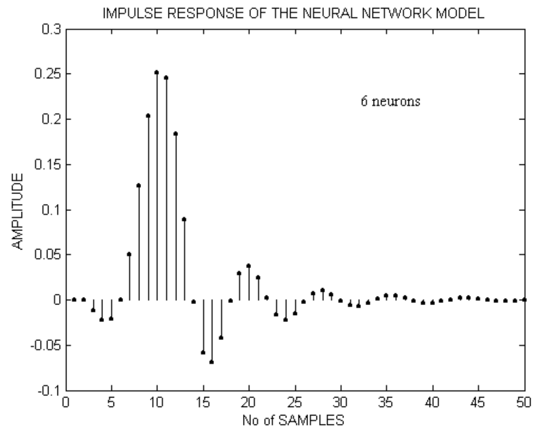

Fig. 2b. Impulse response of the neural network model

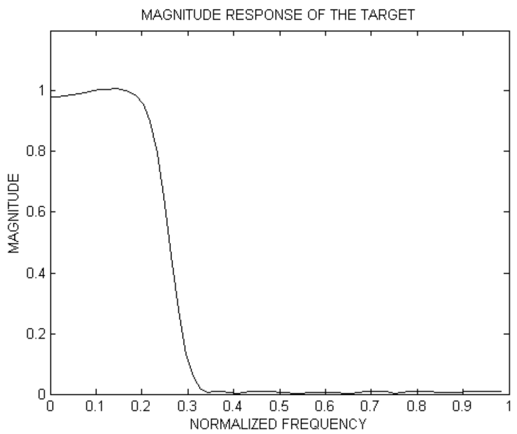

Fig 3a. Magnitude responses of the target IIR filter

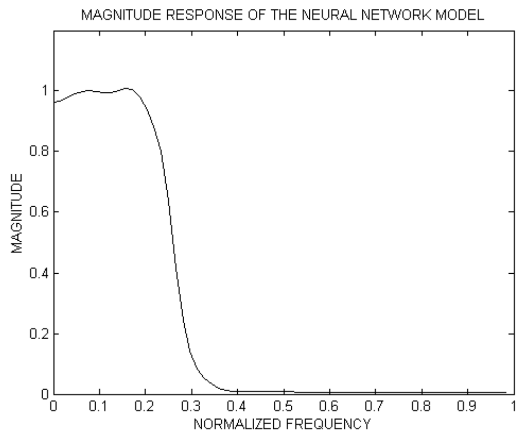

Fig. 3b. Magnitude response of the neural network model

The impulse responses of the target IIR Nyquist filter and the neural network model with 6 neurons are shown in Fig. 2a and Fig.2b respectively. The magnitude responses in frequency domain of the Nyquist filter and the neural network model with 6 neurons are shown in Fig. 3a, Fig. 3b respectively.

4. REFERENCES

[1]. Bhattacharya D. and A. Antoniou, *Real-time Design of FIR Filter by Feedback Neural Network*, IEEE Signal Processing Letters, Vol.3, No 4, May 1996, pp. 158 – 161.

[2]. Pedersen M., *Optimization of Recurrent Neural Networks for Time Domain Series Modeling*, Ph.D Thesis, Department of Mathematical Modeling, Technical University of Denmark, 1997.

[3]. Jin L., P. Nikiforuk, M. Gupta, *Approximation of Discrete-Time State-Space Trajectories Using Dynamical Recurrent Neural Networks*, IEEE Transactions on Automatic Control, Vol.40, No 7, 1995.

[4]. Pei S., Ch. Hsu, P. Wang, *Design of a class of IIR eigenfilters with time and frequency domain constraints.* IEEE Trans. on Circuits and Systems-II Analog and Digital Signal Processing, vol.49, No 2, February 2002, pp 145 - 151.