

DESIGN OF NEURAL-NETWORK FILTERS FOR LOW-FREQUENCY FILTRATION OF SIGNALS

Naila Allahverdiyeva

Azerbaijan State Oil Academy, Baku, Azerbaijan, nailaallahverdiyeva@yahoo.com

Neural networks are most effective approximator of functions. It directly is connected to natural development of the signals filtration theory from a linear theory, where quantity of the prior information about desired signals and noises is great enough, to the non-linear theory. And the vehicle of neural network with reference to a problem of signals filtration encompasses and extends enough plenty of attempts to troubleshoot of non-linear processing by other means. The neural network can solve problems of signals filtration for composite and unknowns of the characteristics of a desired signal, frequently of variables in time, composite and variables in time noise characteristics.

With the help of a graphical interface NNTool of a software package Matlab-7.0 design a neural network for low frequency filtration of a noisy measuring signal with given parameters [1].

Let's select a multilayer network consisting of one perseptron with two inputs. In a learning process of a network on its inputs is giving the input data, and the comparison of value obtained on an output, with target (desired) is made. Ground of outcome of matching (deviation of the obtained value from desirable) are calculated values of change of weights and displacement reducing this deviation.

Before creation of a network it is necessary to prepare a set of the teaching and target data.

The task of network is filtration of an additive mix of harmonic desired signal and noise of a following kind:

$$x_{input}(t) = A \sin(\omega t + \varphi) + \varepsilon(t).$$

Here $A \sin(\omega t + \varphi)$ is harmonic desired signal with given amplitude and frequency, $\varepsilon(t)$ is a random noise.

Let's prepare the target data, having entered in the main window NNTool (fig. 1) in the field Targets- > New Data expression: $5 * \sin(0:0.1:10)$.

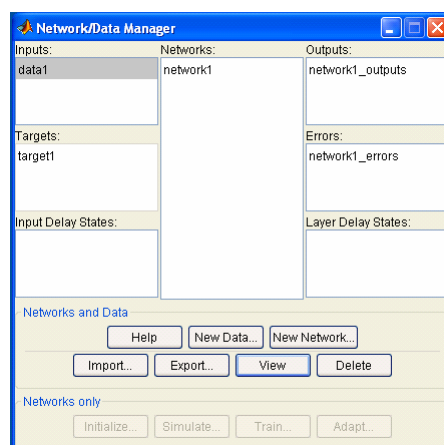


Fig. 1. The main window of a graphical interface NNTool

This curve represents a section of a harmonic signal with amplitude 5 and frequency, equal 1.

Now we shall prepare a set of the training data, having set their with the following expression:

$5 * \sin(0:0.1:10) + \text{randn}(1,101)$. Here, *randn* is a generator of normally (Gaussian) distributed random numbers.

Let's select perceptron (Feed-Forward Back Propagation algorithm) by thirteen sigmoid neurons (TANSIG) for a hidden layer and one linear neuron (PURELIN) for an output layer. The obtained network looks like, shown on a fig. 2.

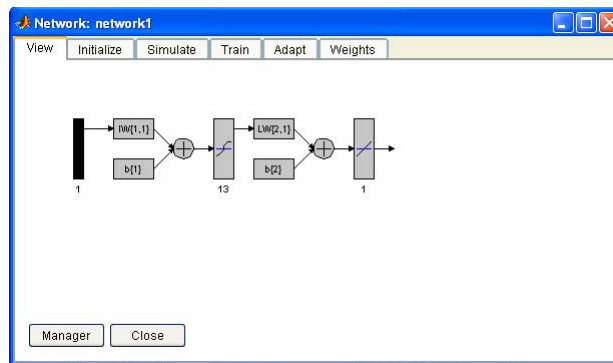


Fig.2. The architecture of a network for the neural-network filter synthesis

Our purpose - to construct a neural network, which executes a function of a low frequency filtration of input data. Apparently, it is impossible to count on that at once after a stage of creation of a network it will provide exact outcome (proper ratio "input / output"). For achievement of the purpose the network is necessary properly for training that is to choose eligible values of parameters. In MATLAB the majority of known algorithms of training of neural network are realized. We shall make training, using Levenberg-Marquardt algorithm, which one will be realized with a function TRAINLM. As an error function we shall indicate MSE (Mean Square Error).

It is possible to complete a learning process, were guided by various criterions. It is possible the situations, when is preferential to stop training, supposing sufficient some time period. On the other hand, objective criterion is error level. In a learning process the network can not supply the precise solution that is to reduce an error to zero point. However, is receives an approximation, which one can be watched on a learning curve in a fig. 3.

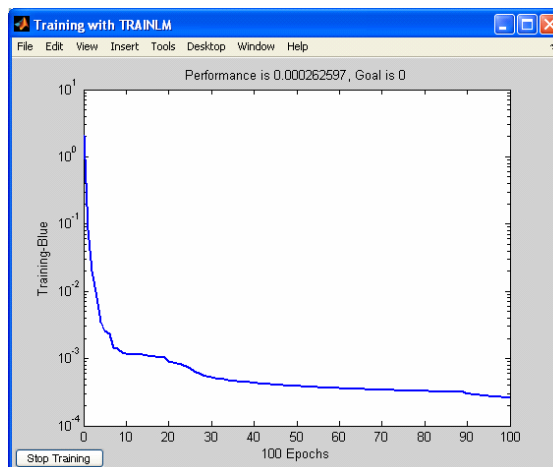


Fig. 3. Dynamics of change of network training error (100 epochs)

As a result of training, the error was minimized up to rather small value, which it is possible to consider practically equaling zero point.

The fig. 4 illustrates a difference between the test data and the output of neural-networks filter. In fig.5 the surface of training is shown.

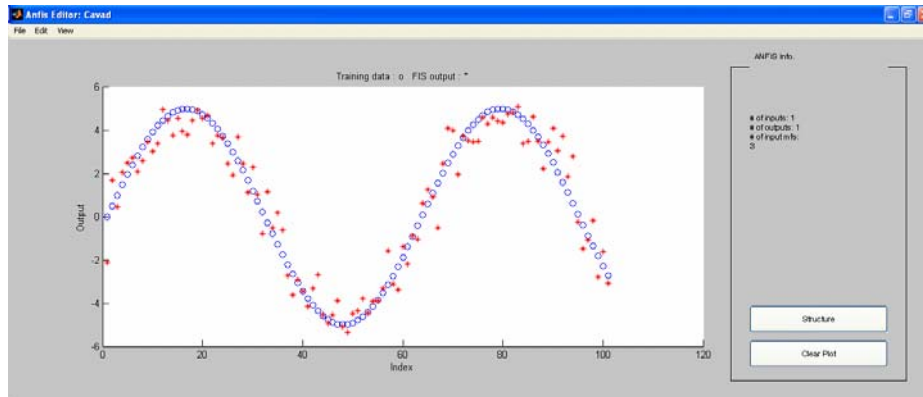


Fig. 4. Outcomes of neural-network filter testing

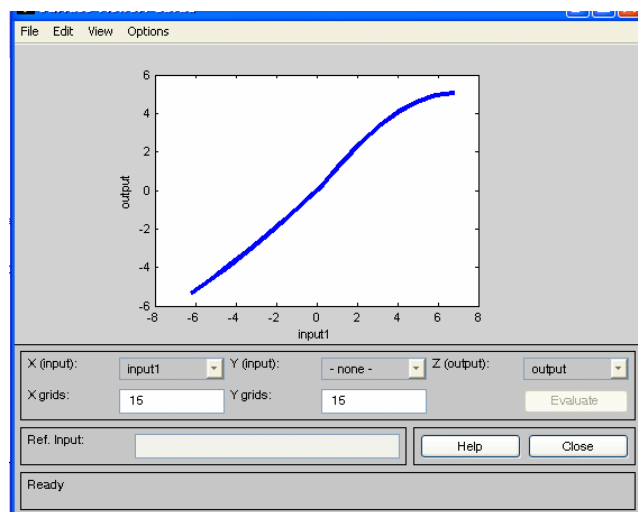


Fig. 5. Surface of training

For testing of neuro-network filter activity the skeleton diagram of neuro-fuzzy controller was prepared (fig. 6).

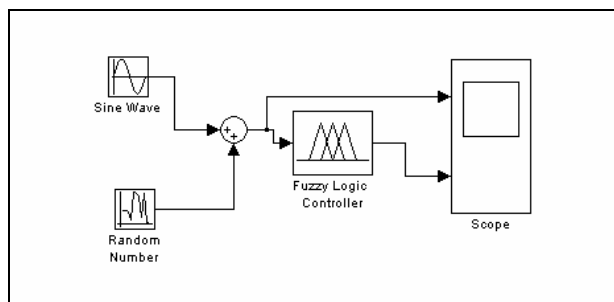


Fig. 6

The outcomes of activity of neuro-network filter executing a low frequency filtration of an input noisy signal, are figured in a fig. 7. From a figure it is visible, that the input signal in a

sufficient measure is cleaned of a high frequency noise. The evidence of it is a reduction of noise dispersion in the output of neuro-network filter almost in 15 times.

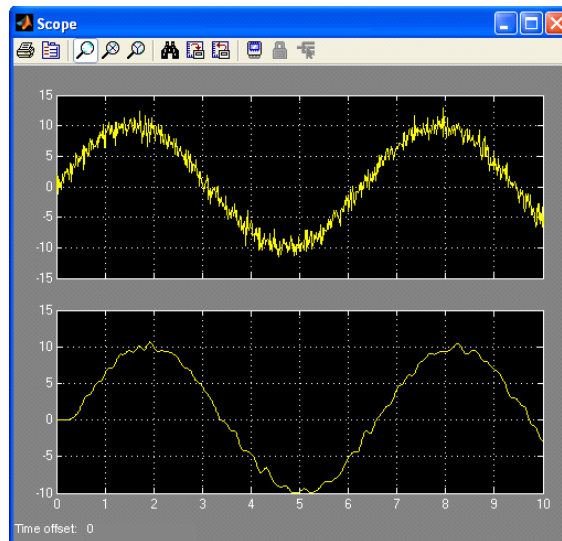


Fig. 7. Outcomes of activity of low pass neuro-network filter

Thus, on the basis of neural-network the synthesis of a digital filter is carried out, outgoing from the prior information on a signal of the known functional shape and additive noise signal, in a form of normally distributed (Gaussian) random noise. Thus is taken into account, that there can be some sets of weighting coefficients of filters for desired signals of different complexity and accordingly calculated with different computing complexity.

Literature

1. V.Ivannikov, A.Lanne. Matlab for DSP. Neural Networks: graphical user interface (In Russian), www.chipnews.ru/html.cgi/arhiv/01_08/1.htm