APPLICATION OF NEURAL NETWORKS IN ESTIMATION OF MAXIMUM FREQUENCY IN TRANSCRANIAL DOPPLER SIGNAL

Wojciech M. Zabolotny*,***, Przemyslaw Wollk-Laniewski**, Elzbieta Luczywek***, Ewa Fersten***

* Institute of Electronics Fundamentals, Warsaw University of Technology, ul. Nowowiejska 15/19, 00-665 Warszawa, POLAND

**Child's Health Center, Al. Dzieci Polskich 20, Warszawa, POLAND

***Department of Neurosurgery, Medical Research Center, Polish Academy of Sciences, ul. Dworkowa 3, 00-784 Warszawa, POLAND

E-mail: wzab@ipe.pw.edu.pl

ABSTRACT

The paper presents a new approach to the estimation of a maximum frequency in the Transcranial Doppler signal. The neural network is trained to calculate the maximum frequency from other, easier to calculate parameters of the signal. The new algorithm is compared to two "traditional" algorithms of maximum frequency estimation showing smaller mean-square estimation error, and better correlation between real and estimated maximum frequency even for poor SNR.

KEYWORDS

Transcranial Doppler, maximum frequency estimation, digital signal processing, neural networks

1. INTRODUCTION

Transcranial Doppler (TCD) is an important tool for non-invasive investigation of blood circulation in cerebral vessels. The continuous monitoring of instantaneous maximum blood flow velocity in one of cerebral arteries is an important way of examination of cerebral blood flow in such conditions as brain oedema, cerebral vasospasm or head injury [1]. The maximum blood flow velocity is proportional to the maximum frequency of TCD signal (f_{max}), and therefore reliable estimation of this parameter is of big importance in neurosurgical diagnostics.

Because of stochastic nature of the TCD signal [2], [3] the estimation of its maximum frequency is not an easy task. All previously published methods of f_{max} estimation [4], [5], [6] were based on theoretical prediction of relationship between f_{max} and different signal parameters, usually the power spectrum density function.

In this work we have proposed another approach. We use the neural network for approximation of unknown function associating f_{max} with other, easy to estimate, parameters of the TCD signal [7].

To train the neural network for calculation of the f_{max} value, we need a TCD signal with known parameters. The only way to get such a signal is to simulate it with an appropriate model.

2. TCD SIGNAL SIMULATION

Our model simulates the formation of the TCD signal. We assume, that blood flows in the cylindrical vessel and the blood flow velocity depends on the distance from the vessel's center according to the formula (1) [2]

$$V(r) = V_{max}\left(1 - \left(\frac{r}{R}\right)^k\right) \tag{1}$$

for $0 < r < R$ and $2 \leq k < \infty$, where R is the vessel's radius, V_{max} is the blood flow velocity in the center of the vessel, and k is the coefficient describing the "flatness" of the blood flow velocity profile. We have divided the blood stream in the vessel in N thin concentric layers with thickness $\Delta r = R/N$. The blood flow velocity in each layer is given by the equation (1), so the frequency of the TCD signal originating from each layer is given by the following formula:

$$f_i = f_{max}\left(1 - \left(\frac{r_i}{R}\right)^k\right) \tag{2}$$

for $0 < i < (N-1)$ and $2 \leq k < \infty$, where $r_i = (i+0.5)\Delta r$ is the mean radius of the layer. The TCD signal component originating from each layer is given by the formula:

$$X_i(t) = I_i \cos(2\pi f_i t) + Q_i \sin(2\pi f_i t) \tag{3}$$

The I_i and Q_i are random variables with the distribution $N\left(0, \sqrt{\beta\nu}\right)$, where β is the constant coefficient describing the intensity of ultrasound scattering, and $\nu = 2\pi r_i$ is proportional to the layer's volume. To get the simulated TCD signal we sum contributions from all layers (in this work we have used 1000 layers) and finally add a white noise to obtain the desired SNR level.

For training and testing of the neural network, and for comparison with other "traditional" algorithms, we have generated two sets of simulated signals. The signals with length of 256 samples were generated for 100 different f_{max} values from range $[0.02 f_N, 0.9 f_N]$, 10 different SNR values from range $[0\text{ dB}, 10\text{ dB}]$ and 10 different values of the k parameter from range $[2, 10]$ – resulting in 10000 signals in each set. The

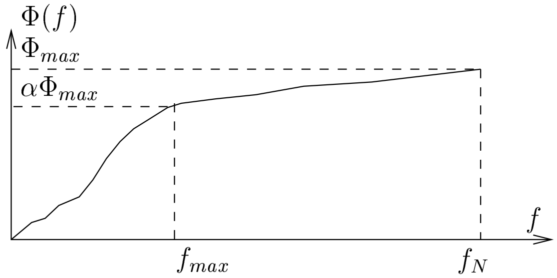

Fig. 1. Percentile method of f_{max} estimation.

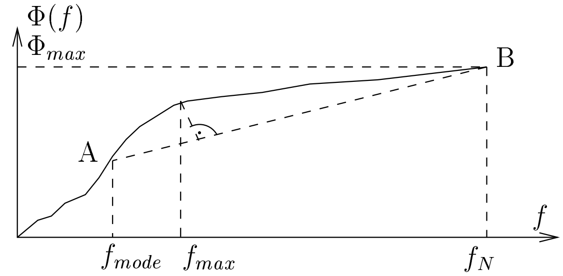

Fig. 2. "Geometrical" method of f_{max} estimation.

first set was used for training of the neural networks, and the second one for testing of "neural" and "traditional" algorithms.

3. THE REAL TCD SIGNALS

A set of real TCD signals have been recorded in patients and healthy volunteers, to allow testing of all algorithms with the real signal. The signals were recorded on tape, and then digitized with sampling frequencies from 10 kHz to 22 kHz and with 16-bit resolution.

4. OTHER ALGORITHMS USED FOR COMPARISON

We have chosen two commonly used methods of estimation of maximum frequency in the TCD signal: the "percentile" algorithm (with FFT based spectral estimation) and the "geometrical" algorithm (with 10th order AR spectral estimation), to compare them to our new algorithm. Both algorithms are based on so called "integrated power spectrum", defined by the formula:

$$\Phi\left(f_x\right) = \int_0^{f_x} G\left(f\right) df \tag{4}$$

The $\Phi_{max} = \Phi(f_N)$ is equal to the total power of the signal (f_N is the Nyquist's frequency).

In the "percentile" algorithm [5] the maximum frequency is defined as the frequency where $\Phi(f) = \alpha\Phi_{max}$. The α coefficient depends on SNR of TCD signal and is typically chosen from $[0.8, 0.9]$ range. The idea of this algorithm is explained in the figure 1.

In the "geometrical" algorithm [6] first we find the frequency f_{mode}, where the power density function reaches its maximum. Then we connect two points in the plot of integrated power spectrum: A - corresponding to the f_{mode} and B - corresponding to the f_N frequency. The maximum frequency corresponds to the point most distant from the AB line. The idea of this algorithm is explained in the figure 2.

5. TCD SIGNAL PREPROCESSING

The raw signal samples contain a lot of information not necessary for f_{max} estimation. For example the total power of the signal does not provide any information about f_{max} value, as well as the phase relationships between different frequency components are not correlated with f_{max}. Some simple operations (for example normalization), necessary to remove such superfluous information, may be very difficult for neural network, and may lead to big complexity of the network. Therefore we need to subject the raw TCD signal to some kind of preprocessing, which will filter out as much needless information as possible. The general idea of the neural network based method is presented in the figure 3. In this work we tested the following preprocessing methods:

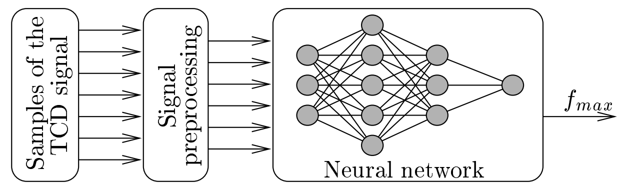

Fig. 3. General idea of the neural network based f_{max} estimation.

- Normalized power spectrum (NPS) (128 samples)
- Normalized integrated power spectrum (IPS) (128 samples)
- Normalized autocorrelation function (NAF) (10 samples)
- Parameters of the 10th order AR model (PAR)
- Reflection coefficients of the 20th order AR model (REFL)

6. ARCHITECTURE AND TRAINING OF THE NETWORK

In this work we have used the "feed forward" networks, which are well suited to the approximation of a function. We have tested three different architectures:

- plain feed forward (FF) networks
- functional link (FL) networks
- cascade correlation (CC) networks

The first experiments showed that it is impossible to estimate the maximum frequency using the neural network with one hidden layer. Therefore we concentrated on feed forward networks with two hidden layers. We used two methods for determining the proper size of the FF network. In the first ap-

Architecture & method	Preprocessing				
	NPS	**IPS**	**NAF**	**REFL**	**PAR**
Plain FF, big net, pruned after training	Unsuccessful training	Unsuccessful training	Good results, largest net	Good results, largest net	Unsuccessful training
Plain FF, gradually increased	Unsuccessful training	Unsuccessful training	Good results, smallest net	Good results, smallest net	Good results, smallest net
Cascade correlation	Unsuccessful training	Unsuccessful training	Good results, medium net	Good results, medium net	Unsuccessful training
Functional link network	Not tested (too many inputs)	Not tested (too many inputs)	Good results, network reduced to the plain FF net	Good results, network reduced to the plain FF net	Good results, two product inputs used

Table 1. Results of training obtained for different architectures, different preprocessing and training methods.

Method	Traditional algorithms				Neural network based algorithms		
	Percentile $\alpha = 0.7$	**Percentile** $\alpha = 0.8$	**Percentile** $\alpha = 0.9$	**Geometrical**	**norm. autocorr. function**	**refl. coeff. of AR model**	**param. of AR model**
Corr. coeffs. between real, and estimated f_{max}	0.8991	0.7436	0.3982	0.5951 0.9872*	0.9944	0.9956	0.9947
Mean-square estimation error for testing set	0.1152	0.1920	0.3361	0.2303 0.0562*	0.0365	0.0254	0.0269

Table 2. Comparison of "neural" and "traditional" algorithms based on the processing of test signal set. Values marked with an asterisk (*) are calculated after rejection of obviously false, zero results given by the geometrical algorithm.

proach we trained the network which was apparently to big, and then, after successful training we pruned it with "Non-contributing Units" and "Skeletonization" methods available in the SNNS package [8]. This method gave the reasonable results only for the networks with NAF and REFL preprocessing. In the second method we started with the very small network and gradually increased its hidden layers, learning and testing after each increase. This approach gave smaller networks, and additionally provided the successful training of the network with PAR preprocessing.

The next architecture tested was the cascade correlation (CC) network. The networks obtained with the cascade correlation algorithm were smaller than standard FF networks obtained with the first method, but larger than FF networks given by the second approach. Again, it was impossible to obtain the network estimating the f_{max} with NPS and IPS preprocessing. Additionally the estimation error of network with PAR preprocessing started to increase after adding of 26th unit.

The last tested network's architecture was the functional link (FL) network with product inputs. Because using of product inputs causes significant increase of inputs' amount, testing was not performed for networks with NPS and IPS preprocessing. For all three remaining kinds of preprocessing we achieved the reasonable results, however for NAF and REFL preprocessing the pruning algorithms reduced the product inputs, so finally we got just a plain FF network. The only exception was the network with PAR preprocessing, where the product inputs a_1^2 and $a_1 a_2$ were preserved by the pruning algorithm. Results of training of networks with different architectures are summarized in the table 1.

7. COMPARISON WITH "TRADITIONAL" ALGORITHMS

We estimated the f_{max} for test signals with known parameters, generated with model described in the section 2., using the different estimation algorithms. Then we calculated the mean-square estimation error and correlation coefficient between the real and estimated f_{max}. The obtained results are presented in the table 2. It can be seen, that the neural algorithms assure the best correlation coefficient between real and estimated f_{max} values and the least mean-square estimation error. Additionally we analysed the scatter plots of estimated versus real f_{max} (three of them are presented in the figure 4). The percentile algorithm gives too high f_{max} for poor SNR, and the geometrical algorithm tends to give false near to zero results for higher f_{max} and poor SNR. No such anomalies were observed for "neural" algorithm.

8. TESTING WITH REAL SIGNALS

When testing with the real TCD signals, we could not compare the real and estimated f_{max}, because the real f_{max} was unknown. Therefore we could only evaluate reconstruction of the blood flow pulse

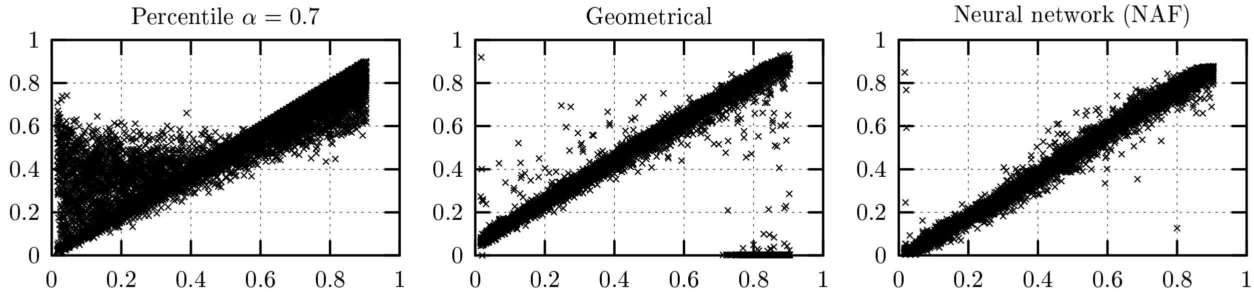

Fig. 4. Scatter plots of estimated f_{max}/f_N (Y-axis) versus real f_{max}/f_N (X-axis), obtained for different algorithms.

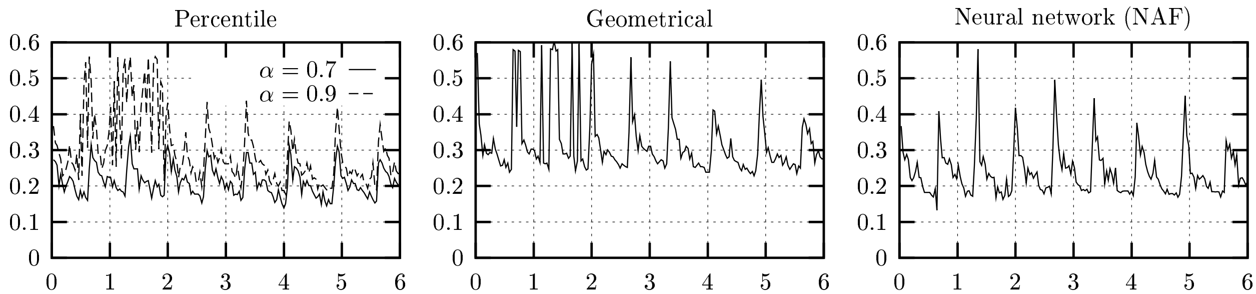

Fig. 5. Results of processing the real, clinically recorded TCD signal with different algorithms. X-axis: time in seconds, Y-axis: f_{max}/f_N

wave in the disturbed real signals. Sample results are shown in the figure 5. The tests performed with the real TCD signals have also proved that the neural network based algorithm usually provides better estimate of f_{max} then other tested algorithms, and is less sensitive to interference present in the signal.

9. CONCLUSIONS

The neural network based algorithms presented in this paper are very attractive methods of estimation the maximum frequency in the TCD signal. Their main advantages are:

- Good performance even for poor SNR
- Small computational complexity
- Ease of implementation in the DSP processors, because the algorithm contains mainly additions and multiplications.

Additionally structure of feed forward networks allows for easy implementation in systems with parallel processing (except of cascade correlation networks). The developed neural network based algorithms have been implemented in a specialized PC extension card with digital signal processor DSP 56002, and successfully tested in clinical practice [9].

REFERENCES

[1] M. Czosnyka. Testing of cerebral autoregulation in head injury by waveform analysis of blood flow velocity and cerebral perfusion pressure in head injury. *Acta Neurochirurgica Suppl.*, (31):468–471, 1994.

[2] B. A. J. Angelsen. A theoretical study of the scattering of ultrasound from blood. *IEEE Transactions on Biomedical Engineering*, BME-27(2):61–67, 1980.

[3] L. Y. L. Mo and R. S. C. Cobbold. A stochastic model of the backscattered doppler ultrasound from blood. *IEEE Transactions on Biomedical Engineering*, BME-33(1):20–27, 1986.

[4] T. D'Alessio. "objective" algorithm for maximum frequency estimation in doppler spectral analysers. *Medical & Biological Engineering & Computing*, 23:63–68, 1985.

[5] L. Y. L. Mo, L. C. M. Yun, and R. S. C. Cobbold. Comparison of four maximum frequency estimators for doppler ultrasound. *Ultrasound in Med. & Biol.*, 14(5):355–363, 1988.

[6] K. Marasek and A. Nowicki. Comparison of the performance of three maximum doppler frequency estimators coupled with different spectral estimation methods. *Ultrasound in Med. & Biol.*, 20(7):629–638, 1994.

[7] W. M. Zabolotny, P. Wollk-Laniewski, E. Fersten, and E. Luczywek. Neural network based algorithm for digital processing of transcranial doppler signal. In *Proceedings of the XX-th National Conference Circuit Theory and Electronic Networks*, pages 451–456, Kolobrzeg, 1997.

[8] SNNS Stuttgart Neural Network Simulator, User Manual, Version 4.1. Provided in electronic form with SNNS package.

[9] W. M. Zabolotny, W. Zaworski, and D. Czarkowski. Low cost PC plug-in card for digital processing of transcranial doppler signal. In *Proceedings of the XX-th National Conference Circuit Theory and Electronic Networks*, pages 457–461, Kolobrzeg, 1997.