

ARTIFICIAL NEURAL NETWORK BASED SIGNAL PROCESSING FOR PERSPECTIVE ONBOARD SYSTEMS

E. Efimov, T. Shevgunov, A. Valaytite, E. Sadovskaya

Moscow Aviation Institute (National Research University), Moscow, Russia

Keywords: *neural networks, adaptive elements, radar image processing, position location*

Abstract

The state-of-art signal processing based on artificial neural network (ANN) technology is discussed in regards to modern, developing and newly-created onboard systems. The concept of simple adaptive elements is introduced as a way to the design of networks that would be used as essential models for the variety of system identification procedures rather than approximating black-box evaluation or a kind of surrogate modeling.

Target detection is presented as an exemplary problem of radar imaging processing which solution is obtained by means of parameter estimation approach performed with radial basis function ANN. The multilayer perceptron is shown to be applied as the core for the robust phase estimation procedure to perform time delay estimation engaged by any target localization technique relying on target radio emission.

The revealed advantages of ANN-based techniques are their adaptability to the environment and the inherent possibility to reconfiguration, although achieved by larger computation cost in comparison with conventional signal processing methods.

Acknowledgement

The reported study was partially supported by RFBR, research project No. 14-01-31399 МОИ_а

1 Introduction

The application of artificial neural networks (ANN) is considered as a hopeful method for obtaining solutions to a number of outstanding

problems in design and developing of modern aircraft systems. The techniques engaging the ANN-based models take their places not only in almost all aircraft lifecycle steps, e.g., design, maintenance, repairmen, services but also in all aircraft subsystems such as glider, engines, flight and warfare control etc.

The particular role the ANN plays for the avionics. The first application can be marked as primarily algorithmic which means that neural networks are considered as the core technique behind the particular digital signal processing algorithm transforming information flows. The application of artificial neural networks to signal processing performed by avionics deals with such problem as target selection, attack planning or steering, and track-before-detect and the list is not exhaustive. The large collected experience is a strong foundation that allows formulating the best practice of ANN application [1, 2]. Thus better performance can be achieved if the target selection and the track-before-detect are approached by the application of a feed-forward back-propagation network but the attack planning or steering is solved via parallel processing neural network, which can also be implemented on fast systolic-array-type neural chips.

The flight control is the essential part of attack planning or steering decision making process. The well-described instance of linear matrix inequalities framework for designing reconfigurable flight controllers with sigmoidal nonlinearities is presented in [3]. The ANN-based approach is directed to addressing the need for flight controllers that provide strong safety and performance guarantees under linear-parameter varying flight conditions, and that can adapt rapidly in the event of changes or failures

that cannot be accounted for a priori. The choosing neural networks as the core provides the learning capabilities and flexibility required to accommodate for a variety of situations and maneuvers. The control system is comprised of sigmoidal neural network where all of the network parameters are updated as soon as a new observation of the flight state becomes available. A constrained training technique is developed as online or interactive learning procedure to prevent forgetting prior control knowledge during incremental training.

Motion forecasting models based on ANN have been also presented in [4] to predict future position of aircraft and ground vehicles around the airport. These models allow for automated monitoring of the movements, ensuring that all operations are performed correctly. Integrated in a surveillance system these modules for position prediction possess the remarkable ability to detect threatening runway incursions.

The important point which makes the neural network models successful is the availability of accurate historical information like data encompassing actions taken by the controller and the pilot, since that information is used to train the networks. The option of using neural networks to model complex environments appears very attractive because of their ability to capture the essence of decentralized decision making and learn the operational environment that exists at the airport and its immediate vicinity.

The notable approach based on ANN will lead to the remarkable progress in the development of avionics testing technology [5]. This heralds major advances for the so called 'No Fault Found' phenomenon and invites the question of extension this technology to be exploited and applied to benefit other maintenance, repair and overhaul activities.

The problem of Fault Detection and Isolation is also covered in [6]. This paper presents the full cycle of design, development and integration of a Fault Detection and Isolation architecture for an air data computer based on Artificial Neural Networks. The proposed architecture was tested on a live flight data and showed good performance in identifying fault occurrences.

The important issue raised in the application of ANN-based technology in electronic system is the platform choosing of their implementation. There are two generic types of ANN assemblies. The first one is their implementation as the algorithm, realized by special software module where the neural networks itself are constructed virtually as the objects in the memory of onboard computer. The main advantage of this bearing are the possibility to construct the networks of almost any complexity, random access to their internal signals and the ability to reconfigure them dynamically with preserving all intermediate states. The greatest disadvantage of software implementation consists in that each active network requires quite large processor time for its normal functioning, while the learning process could be extremely computationally consuming and demands huge processor time.

The second type of networks is the networks implemented on the specialized hardware platforms often referred as neurocomputers [7]. More thoroughly hardware implementation of neural networks algorithm is covered in [8], where the authors proposes simple synapse multiplier with high precision and large linearity range. It is shown that proposed approach is suitable for VLSI implementation of neural network. Since the most widely used approach is taken feed-forward back-propagation networks a special, fast path generation chip is also developed.

Although a low-cost, high-speed, compact solution to a number of avionics functions is available through neural networks this paper presenting the results of researches concentrated more on the tasks of the target detection problem than others mentioned above.

One of the issues raised in the target detection is target identification through the analysis of radar image where a possible solution can be achieved by means of scattering analysis of radar targets. The conceptual idea is a decomposing the image under processing into point scatterers [9] as simpler elements possessing clear physical explanation. Each scatterer could be reproduced in the linked structure of the appropriate neural network.

Another task of target detection process is known as position location that determines one's ability to estimate the coordinates of the particular target using its radio emission. There are several techniques to solve the positioning but angle of arrival (AOA) and time difference of arrival (TDOA) are only vital to be sprung into the solution of the task ready for the avionics. Since one can easily get [10, 11] that there is nothing but time delay estimation is in the center of both techniques. This makes important the developing of a robust algorithm of time delay estimation based on feature extraction. Direct time delay evaluation originated on those features nonlinear combining can be successfully performed in the ANN-basis.

The authors of present paper have found [12] that existing widespread approach to neural network design requires some modifications in order to consider the neural networks as identification technique rather than approximation model only. The neural networks organized as identification models provide one with the powerful tool to perform effective parameter estimation procedure. The complex model can be decomposed into simpler blocks, which are mirrored in the appropriate units in the network graph. It means that well-trained network can be used not only for representing the revealed dependency but for extracting the values of internal parameters of its elements. Afterwards, these values can be reversely mapped into the values of the original model parameters. In simpler cases, there would be one-to-one reference between model parameters and some of network parameters.

The rest of this paper is organized as follows. The section 2 introduces the short outlook on the theory of simple adaptive elements applied to neural networking. The ANN-based solution for the problem identification of multiple scatterers is presented in section 3. The time delay estimation via ANN technique performing the mapping of the discrete cosine transform coefficients into delay value is described in section 4. The paper ends with conclusion depicting the further development of the signal processing based on ANN and applied to modern onboard systems.

2 Network Synthesis Theory

A brief introduction to the basic theoretical principals of neural networks build of adaptive elements will be given in this section. Afterwards, there is detailed explanation of adaptive element structure and interface, two-directional connections, neural network architecture and their role during the overall process of particular neural network synthesis. The structural schemes of regular neurons and radial basis function neurons (RBF-neurons) are also considered while essential features of learning process are discussed at the last part of the section.

An artificial neural network can be considered in most general way as a mathematical model inspired by the biological neural cells possessed by animal. As a virtual model, ANN has its inputs, its outputs and a transformation that can be explained in term of transfer functions in the most of practical cases. There are two opposite directions for information travelling inside the network depending on the network state. The process known as feed-forwarding means signals are propagating inside the network in the forward direction from its inputs to its outputs. These signals are typically the object which the neural network realizes some complicated transformation for. The other process is a back-propagation and during this process a special internal signals move back from the outputs to the inputs implementing the learning process. The back-propagation process can be explained pictorially using back-propagation diagrammatic representation [13].

There are few approaches to synthesize artificial neural networks [14–17], and the most commonly used approach of neural network synthesis operates with regular neurons which are sometimes referred as McCulloch-Pitts model. In this paper, however a quite different approach is used, it implements required characteristics and behavior of the networks through sets adaptive elements. We propose to use simpler basic elements than neurons which can be referred as adaptive elements. The original outlook based on [13, 18] is developed by the authors in papers [12, 19].

Being as simple as they can possibly be and linked in a proper way adaptive elements can form a network of necessary complexity. Thus the description of adaptive elements is of particular interest and their properties are given below.

2.1 Adaptive elements

An adaptive element is a simple transfer element able to react to a given stimulus and produce a corresponding response. As in the whole network itself, the signal within an adaptive element can propagate in two directions, hence, adaptive elements have two transfer functions: for the forward (\mathbf{T}_f) and for the backward (\mathbf{T}_b) signal propagation:

$$\begin{aligned} \mathbf{y}_f &= \mathbf{T}_f(\mathbf{x}_f, \boldsymbol{\theta}), \\ \mathbf{y}_b &= \mathbf{T}_b(\mathbf{x}_b, \boldsymbol{\theta}), \end{aligned} \quad (1)$$

where \mathbf{y}_f and \mathbf{y}_b are output signals for feed-forwarding and back-propagation, \mathbf{x}_f and \mathbf{x}_b are input signals for feed-forwarding and back-propagation correspondingly; \mathbf{T}_f is transform function for feed-forward process and \mathbf{T}_b is transform function for back-propagation. Finally, term $\boldsymbol{\theta}$ describes the vector combining parameters to be adjusted which belong to the particular adaptive element.

The following basic types of most widely used adaptive elements can be defined:

- 1) a summing element,
- 2) a splitting element,
- 3) an amplifying element,
- 4) an element possessing arbitrary functional transform.

In order to be able to build a network of arbitrary complexity, one needs to reestablish connection entities. The structural scheme of the connection between elements is shown in fig. 1. It is important to highlight that direct connections between two elements was chosen that allows signals passing through in both directions and does not introduce any disturbance, noise, interference or other transform to the signal because all of these things are done by the adaptive elements.

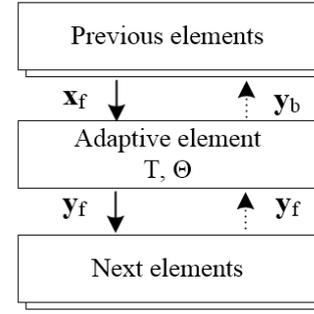


Fig. 1: Structural scheme of the two-directional connection between adaptive elements.

Combining above-mentioned simple adaptive elements makes possible building extremely complex neural network. Some types of neurons such as regular neuron or RBF-neuron can also be synthesized upon the basic adaptive elements foundation.

The next subsection overviews the learning process in regards to the adaptive elements and signal propagation within neural network. This will clarify the desirable features of adaptive element concept.

2.2 Training process

The essential property of neural networks is its ability to extract generalizing information from the data presented to them and to store it in as the value of parameters in networks' structure. This process is usually called learning (or training) since it looks in some sense similar to the learning of primitive animal species. There are few approaches to perform the training of ANN [16]: supervised, unsupervised, reinforced, etc. and a supervised training is used in this work. This means that a well-defined set of training samples must be acquired in order to make the network learn. A single sample of this set consists of pair of input and output data vectors which are assumed to be known for the trainer.

An objective function should be defined for qualitative estimation of network performance during training. In this research, authors rely on mean square error (MSE) function for that purpose:

$$E = \frac{1}{2N} \sum_{n=1}^N (t_n - z_n)^2, \quad (2)$$

where E is overall error (MSE) measured on N training samples, z_n and t_n are network evaluated output value and original output value for the n -th training sample correspondingly. The goal of the training process is easier reformulated in terms of minimization problem of objective function in the space of parameters Θ .

Basically training process contains of the cyclically repeated steps:

- 1) preparing input data samples,
- 2) presenting some samples to the network in feed-forward mode,
- 3) performing back-propagation process,
- 4) adjusting adaptive parameters.

These set of steps are iterated until required performance measured by the network objective function is achieved or any stopping criteria take place. There are a few variants of how to perform last three steps. It is possible to adjust parameters of the adaptive elements after each training sample is presented and the back-propagation process performed or to adjust the parameters once after a bunch of training samples. In this work we will use the latter option known as batch training mode [15].

The parameters adjustment of the adaptive elements can also be performed in various ways. The set of rules prescribing how to use the results of back-propagation process in order to make changes to the values of the parameters is defined by the learning method. In this work, the group of autonomic methods of the first order is considered since they require the values available within the same element only. This key feature leads the light on the paradigm of the adaptive elements. The basic underlying method for all methods of the first order is the gradient decent. For that method the adjustment value $\Delta\theta_m$ for the parameter θ_m of m -th adaptive element is calculated using the local gradient value $\delta\theta$ obtained during the back-propagation process and its history according to the expression below:

$$\Delta\theta = -\varepsilon \cdot \delta\theta, \quad (3)$$

The gradient descent method itself contains parameter ε called a learning rate. The effect of this parameter is significant since the lower value of ε causes slow training process makes required performance achieve later. In the other

words, the computational costs of the training process increases. But the larger values of ε are risky since they may lead to the essential destabilization of the training process.

In order to suffice requirements of supervised learning with back-propagation, the transfer functions of adaptive elements must be defined accordingly. Here are transfer functions of some basic adaptive elements presented. Thus transfer functions for summing and splitting elements have to be defined as follow:

$$\begin{aligned} y_{f, \text{sum}/b, \text{splitter}} &= \sum x_{i, f, \text{sum}/i, b, \text{splitter}}, \\ y_{f, \text{splitter}/b, \text{sum}} &= x_{f, \text{splitter}/b, \text{sum}}, \end{aligned} \quad (4)$$

where output signals are denoted as y and input signals are x ; the indices determine the directions: f is for feed-forwarding while b is for back-propagation.

Since the amplifying element is, in fact, a special case of arbitrary functional transform element (i.e. transform with a given amplifying factor) the following equations define transform functions of that element:

$$\begin{aligned} y_f &= wx_f, \\ y_b &= wx_b, \end{aligned} \quad (5)$$

where x_f and y_f stand for feed-forwarding input and output accordingly, x_b and y_b stand for back-propagation input and output accordingly and w does for amplifying coefficient which is equivalent to the synaptic weight.

Four adaptive elements described above are engaged in building various types of neurons such as classical neurons and neurons for Radial Basis Functions (RBF) networks. In order to implement necessary activation functions for the neurons of each subtype, arbitrary functional transform elements are used. The most popular and widespread activations functions are sigmoidal, hyperbolic tangent, Gaussian, square, linear etc.

2.3 Neuron Assembling

Using adaptive elements as a starting point makes synthesizing the most classes of neurons possible. Such approach provides network structure to be plain and simple yet allows creating complex transfer functions. This implements the paradigm known as network's

point of view which means all elements are quite identical in terms of signal propagation that also allows introducing new types of elements easily while any new elements would be able to represent various systems considering their parts as covered (or masked) sub-networks.

The distinctions in the synthesis of two types of neurons: regular and RBF-neuron is overviewed below.

2.3.1 Regular neuron

A regular neuron (or linear separating neuron) can be synthesized by means of a set of amplifying elements, a summing element and a functional transform element with required transformation function. Each amplifier is connected to a certain summing element input thus implementing synoptic weight.

The structural scheme of a regular neuron with 3 inputs is shown in fig. 2. Its synaptic weights are represented by the amplifiers in the left-hand part of the scheme, than weighted signals are summed and transmitted to the functionally transforming element that embeds the necessary activation function.

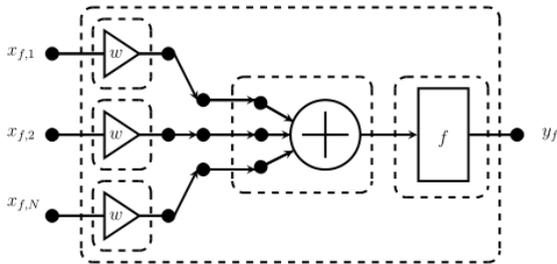


Fig. 2: Structural scheme of regular neuron with three inputs during feed-forwarding.

The overall transform functions of regular neuron define as follow:

$$\mathbf{y}_f = f\left(\sum_{i=1}^N (x_{f,i} w_i)\right), \quad (6)$$

According to the back-propagation characteristics of the underlying elements the back-propagation function and the structural scheme in case of back-propagation for regular neuron define as shown in the fig. 3.

The transform function in that case has the formula defined in the expression below:

$$\mathbf{y}_{b,i} = x_b f' \left(\sum_{i=1}^N (x_{f,i} w_i) \right). \quad (7)$$

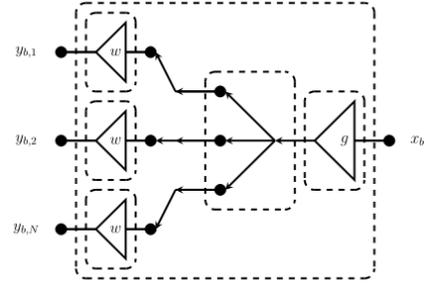


Fig. 3: Structural scheme of regular neuron with three inputs during back-propagation.

2.3.2 RBF-neuron

An RBF-neuron is a special type of neuron that not only uses a radial basis function as its activation function but also has another input combiner. Structural scheme of RBF neuron during feed-forwarding is shown in the fig. 4. In case of working with two-dimensional input data such neuron will normally have 3 inputs: two for coordinates x and y and another one for bias, the latter marked with “+1”.

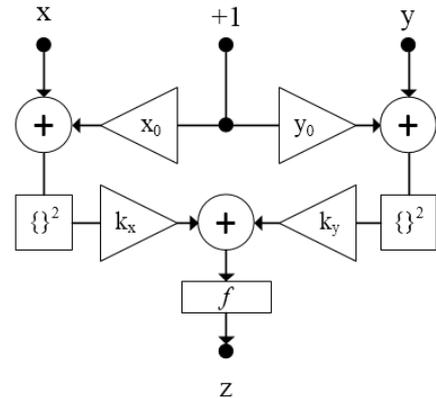


Fig. 4: Structural scheme of RBF neuron for feed-forwarding.

The transformation function can be easily written from the scheme of the neuron in fig. 4, thus the output signal is defined by the following expression:

$$z = f\left(k_x(x_0 + x)^2 + k_y(y_0 + y)^2\right) \quad (8)$$

The argument inside the parenthesis of f -term in the transfer function above can be rewritten in the form of canonical equation for an ellipse (or an ellipsoid if the dimension would be greater than two):

$$z = f(w(x, y)), \quad \frac{(x-x_1)^2}{c^2} + \frac{(y-y_1)^2}{d^2} = 1. \quad (9)$$

Where the following substitutions are applied:

$$\begin{aligned} x_1 &= -x_0, y_1 = -y_0, \\ c &= \sqrt{w/k_x}, d = \sqrt{w/k_y} \end{aligned} \quad (10)$$

In the case of Gaussian function taken for the activation function f , the output signal of one RBF-neuron will be equal to unity at the central point and rapidly decreasing bell around.

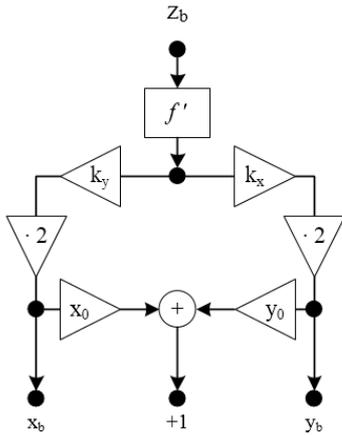


Fig. 5: Structural scheme of RBF neuron for backpropagation.

The individual knowledge of the adaptive elements behavior during their back-propagations opens the option for constructing structural scheme of RBF neuron for back-propagation (see the fig. 5).

As it can be seen from the overview given in this section the concept of the adaptive elements perfectly corresponds to the theoretical basis of neural network processing. Complex yet simple in their nature basic elementary neural structure can be successfully built with decomposing approach using adaptive elements. The strict definition of input and output interfaces of the adaptive element allow to keep it relatively isolated. This makes implementation of custom algorithm based on first-order training methods possible. The next sections explain some application of neural networks built according to the proposed concept.

3 Scatter based target identification

The ability of neural networks to operate on distorted, noised and incomplete data sets alongside with the properties of RBF-neurons

allows them to be applied in the scatter based target identification which is reviewed in this section. The radar image processing using ANN in order to extract information about the target depicted on is presented below.

The image to be processed is assumed to be obtained by a radar system performing azimuthal scanning and having a high resolution for both distance and angle. The system operating in centimeter wavelength range emits coherent pulses with the same antenna working with time division both for transmission and echo receiving receiver. The scanning is carried out in a presence of white Gaussian noise. The main task of the scatterer identification is the estimating parameters of multiple scatterers involved in target representation such as the coordinate of their centers within the target as the most important information to characterize its geometrical form in a further possible processing. This acquired information could be passed to the specialized classification systems that form the anchors for the process of the automatic target identification.

The proposed approach in general consists in using specifically-synthesized RBF-neural network to approximate sampled radar image given to its input. In the case of successful approximation, the Cartesian coordinates of the scatterers are taken from the parameters of neurons directly. The numerical calculations results are presented in order to estimate practical viability.

3.1 Radar image model

The enhanced model of complex radar target [20] considers the radar target represented as a set of individual scatterers mounted on a stiff backbone. This model supposes that echo-response signals received during observation are to be determined as a superposition of the responses from each individual scatterer. Each of these individual responses possesses in the first consideration the form of the probe pulse emitted by the radar system.

Such model has a few deliberately introduced simplifications:

1) non-linear distortions caused by signal reflection from a scatterer are compensated;

2) changings in the position of the scatterer during antenna moving are neglected;
 3) the inaccuracy of reflection process could be represented in the model by means of the additive noise component.

The model of space-time radar echo response signal of complex radar target is used to generate test input radar image. Since a high resolution in both distance (denoted by ρ) and angle (denoted by φ) for the observed radar target is assumed, the signal could be described with following expression:

$$\begin{aligned} \dot{x}(t, \theta) &= \sum_{p=1}^P \dot{x}_p(t, \theta) + n(t) \\ &= \sum_{p=1}^P \dot{a}_p \cdot \dot{s}(t - \tau_p) \cdot f_A^2(\theta - \varphi_p) + n(t), \end{aligned} \quad (11)$$

where $\dot{x}_p(t, \theta)$ stands for complex-valued space-time radar signal echoed from p -th individual scatterer, $\dot{s}(t - \tau_p)$ is the slice of the radar image across the distance (in fact, the form of this slice will correspond to the form of probe pulse), $f_A^2(\theta - \varphi_p)$ is term for squared antenna pattern, φ_p determines the main beam direction of the antenna. The term $n(t)$ describes the additive white Gaussian noise which is uniform in the frequency bandwidth used by radar system. The schematic structure of such signal is shown in fig. 6.

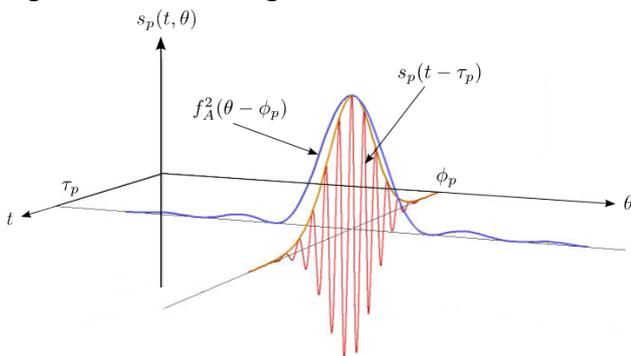


Fig. 6. The structure of space-time radar echo response signal of complex radar target.

The instance of a typical radar image with three individual scatterers generated according to the introduced model for the complex radar target is shown in fig. 7. This image is noted to be visualized by the absolute value of originally complex-valued 2D function (11).

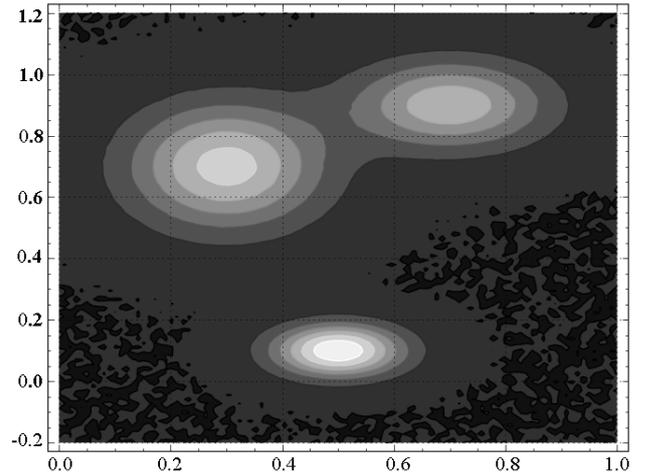


Fig. 7. The instance of radar image with three scatterers.

As one can note in fig. 7, two pulses are close to each other, so they are partially overlapping – this was chosen intentionally. The third pulse has the greatest intensity and located separately from the others. The probe pulse range waveform and antenna cross-range pattern are both assumed to be of Gaussian curve.

3.2 Neural network design

The coordinate of scattering centers are considered [21] to be the most relevant parameter for the target identification. In works [22, 23] authors have proposed the way to the identification using parametric methods for pole estimation in the frequency domain. The coordinates of the poles on the virtual complex plane can be used then to evaluate geometrical centers of the scatters. Although this approach demonstrates high accuracy and has proven suboptimal nature, it suffers from high calculation cost and the requirement to perform accurate deconvolution of the radar image that is naturally ill-conditional problem.

Author of the present work have proposed in [24] alternative solution based on ANN-framework since it is not difficult to notice that described in the previous section Radial Basis Function (RBF) neural networks perfectly correspond to the model (11). Proposed approach contains the following stages:

- radar image sampling;
- forming training set from the radar image samples;
- RBF neural network synthesizing;

- using this set as the input data during neural network training process;
- using RBF neurons parameters to directly calculate coordinates of individual scatterers (in case of success training).

The structural scheme of RBF neural network is shown in fig. 8. It consist of input signals x and y representing coordinates of a point belonging to the image to be processed and output z representing the intensity at this point; block marked with “+1” introduces bias input.

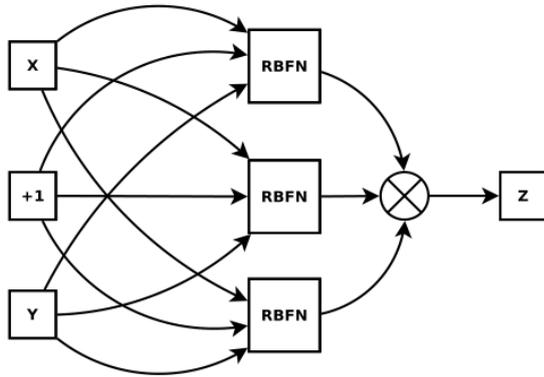


Fig. 8. RBF neural network.

Training set consists of samples which x and y coordinates on the image are used as input data and the signal intensity in that point is used as required output data. Hence, the problem can be now redefined as the task of approximation of the target radar image with neural network.

Output signal generated by neural network is defined by the following equation:

$$x_a(\rho, \varphi) = \sum_{p=1}^P g_p(\rho, \varphi) \quad (12)$$

where x_a is output network signal (i.e. approximated radar image), ρ and φ stand for distance and antenna azimuthal angle correspondingly, P denotes the number of RBF neurons within the network, g_p is partial output signal taken as weighted output of p -th RBF neuron.

The important point to highlight is that single RBF neuron is targeting a particular part of the image which is best corresponding to the neurons output signal. Thus by selecting activation function of the neuron in accordance to the probe pulse form and antenna pattern, one can expect that each single RBF neuron will

after the successful training process target one individual scatterer.

The parameters of the neurons are available after the training and can be directly used to calculate the estimated parameters of the scatterers, e.g., the coordinates of their centers and the effective widths.

3.3 Numerical simulation

Since any strict theoretical investigation of on the estimation ability of ANN are always challenging, the numerical simulation is a helpful tool that is used in order to estimate practical accuracy of the proposed procedure. The described above allocation of three individual scatterers was used to synthesize radar image of complex target.

The radar image shown in fig. 7 undergoes sampling. The discrete points, or samples, of the image are shown in fig. 9 with circles, which squares are proportional to their intensity. Activation function is chosen Gaussian in accordance to the form of probe pulse and antenna pattern possessed by model (11). Since the number of RBF-neuron corresponds to the number of scatterers, this quantity is assumed known and is equal to three.

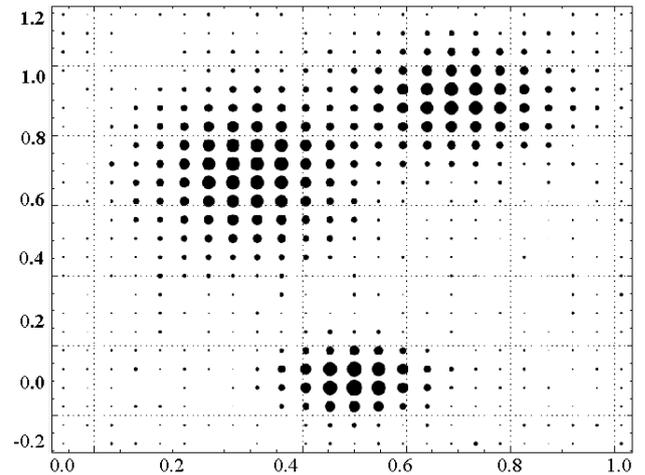


Fig. 9. Sampled radar image.

All sample formed training set was used to train RBF network in batch mode with gradient descent method. The objective MSE function representing there the difference between source radar image and approximated one evaluates network performance during the training. The learning curve during the training is shown in

fig. 10 as the value of MSE plotted against the number of iteration or epoch.

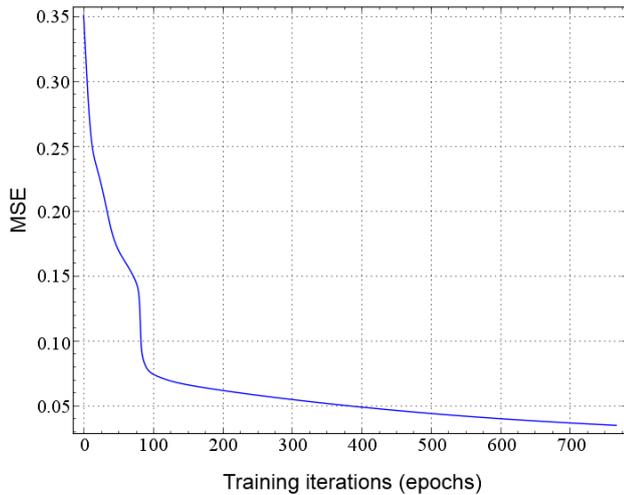


Fig. 10. Training process.

The low level of MSE is achieved after 100 training iterations; hence the approximated radar image accurately. Parameters of the scatterers are immediately taken from the adaptive elements the neurons consist of. The reconstructed scatterers are shown in the fig. 11 alongside with the true ones.

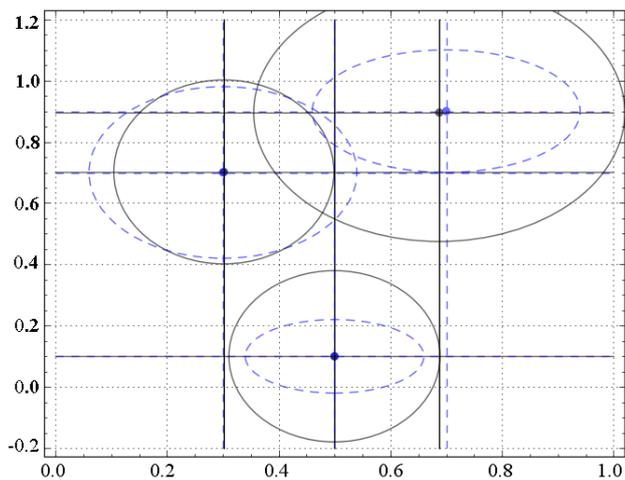


Fig. 11. Original and reconstructed scatterers.

It is clear that the centers of all scatterers are accurately estimated despite the fact that two of them were partially overlapping. However effective widths of the scatterers are not estimated accurately. This phenomenon could be explained by their lesser influence on the overall MSE value. Besides, fig. 10 shows that the value of MSE is still decreasing at the end of the training process but notably slower than it took place at the beginning.

The series of simulation indicate that RBF neural network can be successfully used for accurate estimating the centers of the individual scatterers with signal-to-noise ratios down to 5 dB.

4 Neural based target localization

In this section the application of the neural network to the task of target localization is discussed. A particular problem of estimating constant delays as informative parameters embedded into received signals that are noisy and damped image of the known reference signal.

The offered approach uses Discrete Cosine Transform (DCT) to obtain a set of DCT coefficients from the received signal. Then, selection is used to reduce number of coefficients in the selection, hence getting Reduced Cosine Transform Coefficients (RDCTC) set. The phase shift embedded in the original signal will also be encoded into RDCTC set. Since the form of the reference signal is known a series of RDCTC sets can be synthesized in advance and used as training data to the Neural Network. Reduced size of the DCT coefficient set guarantees relatively small size of the network itself.

Using these synthesized RDCTC sets to train network one can obtain the neural network designed specifically for the known reference signal; since the neural network is featured with the ability to handle noisy and incomplete data this solution becomes both accurate and efficient in terms of calculation cost.

4.1 Theory framework and model

One of the important problems within position location task is calculating delay between two signals received at the specially separated points. The prime concept of angle-of-arrival (AOA) estimation technique is shown in fig. 12.

Signal $s(t)$ radiated by a radio source (RS) is received at two spatially separated points which signals can be described in the form:

$$\begin{aligned} s_1(t) &= K_1 \cdot s(t - T_1) + n_1(t), \\ s_2(t) &= K_2 \cdot s(t - T_2) + n_2(t), \end{aligned} \quad (13)$$

where K_1 and K_2 are attenuation ratios, T_1 and T_2 are time delays and n_1 and n_2 are noises for the first and second points correspondingly.

Considering the simple instance of mono-harmonic signal $s(t)$ with circular frequency ω , the points separated by distance d not farther than half of its wavelength and noises uncorrelated with each other as well as with the signal one can obtain the following expressions for received signals

$$\begin{aligned} x_1(t) &= A \cdot \cos(\omega \cdot t + \varphi_1), \\ x_2(t) &= A \cdot \cos(\omega \cdot t + \varphi_2), \end{aligned} \quad (14)$$

where amplitude A is the same while phases φ_1 and φ_2 can be used for evaluation of angle of arrival:

$$\cos \alpha = \frac{c(\varphi_2 - \varphi_1)}{\omega d}, \quad (15)$$

where c is phase speed e.g. speed of light.

There are few approaches to determine phase shift between two harmonics. The most widely used technic is cross-correlation. One of its efficient applications is shown in [26]. But not all approaches involves cross-correlation, the algorithm of direct phase estimation is based on neural networks and was originally introduced in [27] and developed by authors of this research in [28].

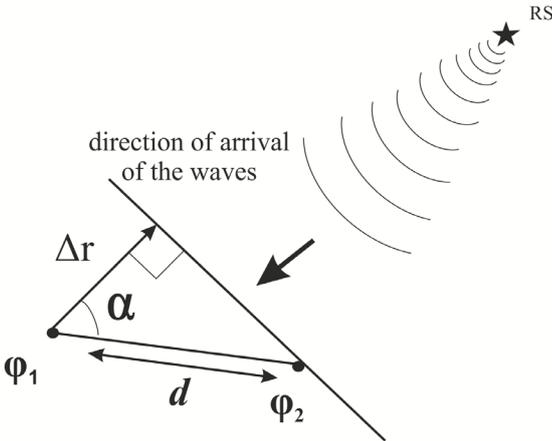


Fig. 12. Receiving signals at two spatially separated points.

In order to implement this sort of algorithms one needs at first to determine features of the signal that will most effectively embed the information about the delay or phase shift. Such features must obviously be very

sensitive to the changes in the delay. It was shown in [27, 28], that a subset of Direct Cosine Transform coefficients can be used for that purpose. This subset can be used to decode the value of phase shift in order to calculate time delay.

Discrete Cosine Transform is a transform with even harmonic functions basis. There are 8 types of DCT due to the symmetry property; in this paper we use the most thoroughly researched type described in [25] as normalized DCT-II and defined by the following equations:

$$\begin{aligned} X[k] &= \sqrt{\frac{2}{N}} \cdot \beta[k] \cdot \sum_{n=0}^{N-1} \Psi_{n,k}, \quad 0 \leq k \leq N-1, \\ \Psi_{n,k} &= x[n] \cdot \cos\left(\frac{\pi \cdot k \cdot (2n+1)}{2N}\right), \end{aligned} \quad (16)$$

$$\text{where } \beta[k] = \begin{cases} \frac{1}{\sqrt{2}}, & k = 0; \\ 1, & 1 \leq k \leq N-1. \end{cases}$$

That transform was chosen among the others because of its property known as “energy compression” [25]: coefficients of the sequence concentrated at some range of indices with higher density rather than in Discrete Fourier Transform (DFT) or any other types of DCT. This property is illustrated by 3D-plot in fig. 13 where the actual values of DCT coefficients are plotted against coefficient number and values of the time delay measured in the part of harmonic period.

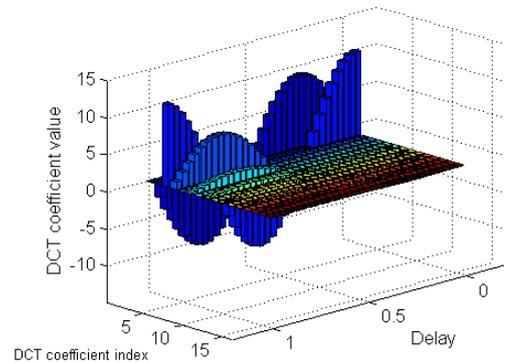


Fig 13. DCT energy compression.

The results shown in fig. 13 indicate that some of coefficients are less sensitive to the phase shift; hence the usage of the whole set of DCT coefficients is redundant. The criteria of how to choose the most sensitive of the phase

shift coefficients must be defined. In paper [27], the criteria of class L_1 based on sum of absolute difference (SAD) is used:

$$SAD = \sum_{n=0}^{N-1} |R(\varphi_{n+1}) - R(\varphi_n)|, \quad (17)$$

where R is DCT coefficient under analysis, φ_n are possible phase values of the received signal defined with some predefined steps.

Alternatively to SAD we have used L_2 criteria [28]: the sequential selection of the coefficients containing biggest share of energy, although those criteria led to the same results in selecting the most sensitive DCT coefficients (R) in order to form RDCTC sets. These RDCTC sets are used later as a training data for the neural network.

4.2 Numerical simulation

The conceptual graph of feed-forward neural network known as multi-layer perceptron (MPL) is shown in fig. 14. We will use this type of network, which can be easily build using adaptive elements, to solve task of estimation of the constant delay. The network is built from neurons united into two layers traditionally called hidden and output layers. Each neuron embeds the summing element and the functional transform; for the hidden layer hyperbolic tangent function is used. The output layer is strip of its functional transform element and contains only summation element.

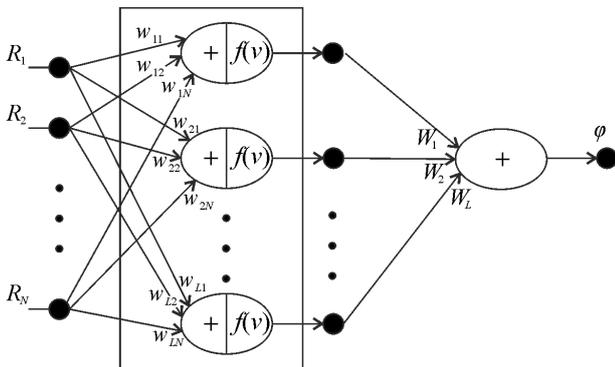


Fig. 14. Multilayer perceptron as neural network for angle evaluation.

As input signal $s(t)$ for the numerical modeling a single harmonic of frequency Ω_0 and sampling rate T was chosen. The noiseless signal was delayed with constant delay D . A

series of DCT transforms were performed on input signal for various delay values.

The results collected into a single plot are shown in fig. 15 as a dependency between of DCT coefficients with smaller index value from the delay.

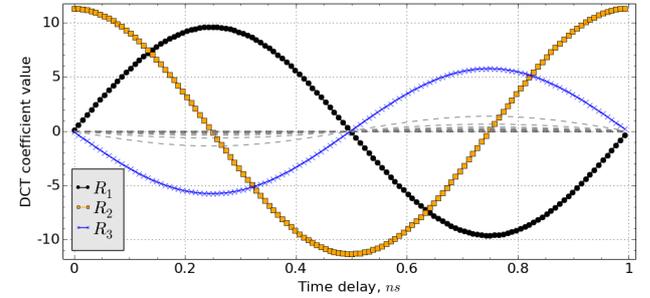


Fig. 15. The dependency between DCT values and the phase shift embedded into signal.

Fig. 15 clarifies the point that despite the fact that all DCT coefficients are sensitive to the value of the delay in the signal, the most sensitive to that change are only a few coefficients marked as R_1 , R_2 and R_3 . These results also indicate that no matter how sensitive any coefficient to the value of the delay is, it is sensitive only for the changes in some range, for example R_1 coefficients is only sensitive to the change in ranges $[0.18 - 0.4]$ and $[0.6 - 0.86]$. That means R_1 will improve quality of the estimation only for the values within these ranges; outside of them the coefficient will be useless. In order to cover all possible values of the time delay changes we need to pick a few DCT coefficients so they ranges of sensitivity overlap. In this particular case, we need at least 3 coefficients to obtain appropriate estimation procedure.

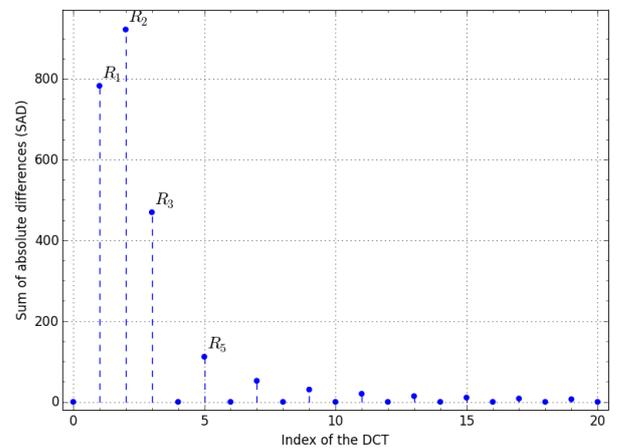


Fig. 16. SAD evaluated for DCT coefficients.

The integral outlook of the ability could be presented by the dependency of SAD, written in equation (17), from the number of DCT coefficient as it shown in fig. 16.

The next step after the indexes of the most sensitive DCT coefficients have been determined is selecting optimal neural network topology [12] in order to make it efficient in the estimation of phase-shift value. The numerical modeling was performed to choose the topology leading to lower value of Mean Square Error (MSE). The topology parameter to optimize is the size of the hidden layer. A series of 40 experiments was performed for each size of hidden layer in order to rule out the effect on random initial synaptic weights. In each experiment the value of MSE for neural network with the hidden layer of given size was estimated, taken values were averaged afterwards. The obtained dependency between MSE and the hidden layer size is shown in fig. 17.

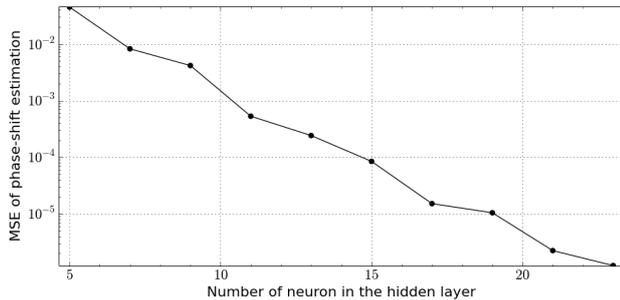


Fig. 17. The dependency between MSE and the number of neurons in the hidden layer.

More experiments were done to estimate the influence of both the size of the hidden layer and the number of the most sensitive DCT coefficients. The results are compiled into a diagram shown in the fig. 18; the value of MSE are depicted with intensity of grayscale.

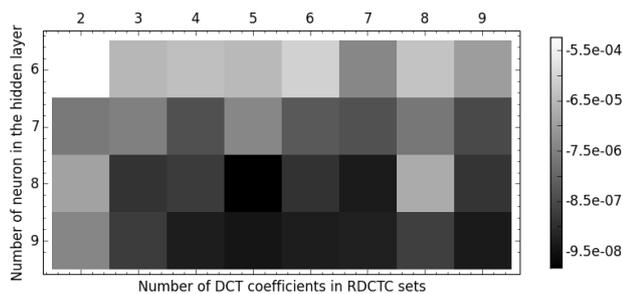


Fig. 18. The dependency of MSE from the number of neuron in the hidden layer and the number of DCT coefficients in the RDCTS sets.

The analysis above shows that it is enough to take 5 most sensitive DCT coefficients with 8 neurons in the hidden layer. The results of the numerical simulation indicate that MSE generally decreases as the number of DCT coefficients in the RDCTC sets or the number of the neurons in the hidden layer increases. It is important to note that the other side of the increase of either size of hidden layer or the size of RDCTC sets is the increase of the calculation cost.

5 Conclusion

The present paper reflects the current advances in the neural network based signal processing in regards to tasks related to the modern onboard systems. The concept of adaptive elements forming the framework to synthesize neural network of various types is described. Two practical tasks solved using the proposed approaches with neural network are presented: scatterer based target identification and target localization.

The artificial neural networks based on simple adaptive elements are described and the two significant advantages of this framework are discussed. The first is the clear potential to the architecture extension and incorporating the difficult and implicit parts of the model into blocks belonging to the networks representing the model under investigation with the required degree of adequacy and highlighting the details of the interest. The second is the straight way to the building system identification scheme achieving with open access to the values of internal parameters possessed by underlain network blocks. This allows one to implement the complex parameter estimation for monitoring, predicting and control problems on the powerful learning procedures developed for several decades.

The significant results of the paper is demonstrating a possible way how neural networks can be used for automated radar image analysis which is the essential part for a set of tasks forming target detection problem. The proposed algorithm carries out the system identification approach reached via neural network learning procedure. Thus the radar

image is firstly approximated by RBF networks where each RBF-neuron preserves the information about the point scatterer of the possible targets. The adaptive element concept chosen for ANN synthesis is extremely suitable for the second stage when the values of parameters are being extracted and further transformed into parameters of the multiscatterer model.

It was shown that the neural networks give the strong framework for effective time delay estimating algorithm. The estimation procedure relies on the feature extraction technique carried out by discrete cosine transform which appropriate type allows obtaining the compact energy representation of signal undergone to the digital processing. Since time delay estimation is the core of commonly used position location technique – angle of arrival and time difference of arrival – ANN provide the basement for integrated procedure. The feed-forward neural network, also known as multilayer perceptron, appears to be the most effective architecture. The optimization problem for the number of character enough for signal representation and for the number of elements in the hidden layer was also solved as the illustrative example in the current research.

The research, which some of results are presented in this paper, is in progress and will hopefully continue. The main finding is that neural networks can no doubt be successfully applied to the signal processing performed by onboard electronic systems. These systems will be able to solve different task in the similar way by means of the entire ANN framework with the modification required for particular problem.

6 References

- [1] Seidman A. Neural networks and digital avionics. *Proceedings of 9th Digital Avionics Systems Conference, IEEE-AIAA-NASA*, 1990.
- [2] Bowman C, DeYong M and Eskridge T. Role of neural networks for avionics, *Proc. SPIE 2566, Advanced Imaging Technologies and Commercial Applications*, 1995.
- [3] Ferrari S and Jensenius M. Robust and Reconfigurable Flight Control by Neural Networks. *AIAA*, 2005.
- [4] Doshi A. *Aircraft Position Prediction Using Neural Networks*. Thesis, MIT, 2005.
- [5] Huby G, Cockram J, Baker J. Developing the Use of Neural-Network Test Technology from Avionics to Wiring Systems. *Aircraft Airworthiness and Sustainment Conference*, Brisbane, 2012.
- [6] Garbarino L, Zazzaro G. Neural Network based architecture for Fault Detection and Isolation in air data systems. *32nd IEEE/AIAA Digital Avionics Systems Conference (DASC)*, pp. 1-28, East Syracuse, NY, 2013.
- [7] Shukla K. *Neuro-Computers: Optimization Based Learning*. Narosa, 2003.
- [8] Bingxue S. CMOS PWM VLSI implementation of neural network. *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks (IJCNN)*, Como, Vol. 3, pp. 485-488, 2000.
- [9] Rihaczek A and Hershkowitz S. *Theory and Practice of Radar Target Identification*. Norwood, Artech House, MA, USA, 2000.
- [10] Knapp C and Carter G. The Generalized Correlation Method for Estimation of Time Delay. *IEEE Transactions on Acoustic, Speech and Signal Processing*, 1976, Vol. 24, No. 4, pp. 320-327.
- [11] Dubrovin A and Sosulin Y. One-stage estimation of the position of a radio source by a combined passive system. *Journal of Communications Technology and Electronics*, 2007, Vol. 52, No. 4, pp. 415-430.
- [12] Efimov E and Shevgunov T. Feed-forward neural network synthesis using adaptive elements. *Journal of Radio Electronics*, 2012, No. 8, IRE RAS, Moscow.
- [13] Wan E and Beaufays F. Diagrammatic Methods for Deriving and Relating Temporal Neural Network Algorithms. *Adaptive Processing of Sequences and Data Structures, Lecture Notes in Computer Science*, 1998, pp. 63-98.
- [14] Haykin S. *Neural Networks: A Comprehensive Foundation*. Macmillan Coll Div, 1994.
- [15] Samarasinghe S. *Neural Networks for Applied Sciences and Engineering: From Fundamentals to Complex Pattern Recognition*. Auerbach Publications, 2006.
- [16] Hassoun M. *Fundamentals of Artificial Neural Networks*. The MIT Press, 1995.
- [17] Hagan M, Demuth H and Beale M. *Neural Network Design*. Martin Hagan, 2002.
- [18] Narendra K and Parthasarathy K. Identification and Control of Dynamical Systems Using Neural Networks. *IEEE Transactions on Neural Networks*, Vol. 1, No. 1, 1990, pp. 4-27.
- [19] Efimov E and Shevgunov T. “Development and Analysis of the Technique for the Building of Artificial Neural Networks Based on Adaptive Elements”, *Trudy MAI*, No. 51, 2012.

- [20] Henderson F and Lewis A. *Principles and applications of imaging radar. Manual of remote sensing*. 3rd edition, Vol. 2, John Wiley and Sons, Somerset, NJ USA, 1998.
- [21] Chen C and Andrews H. Target motion induced radar imaging. *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-16, No. 1, pp. 2–14, 1980.
- [22] Chen V and Ling H. *Time-Frequency Transforms for Radar Imaging and Signal Analysis*. Norwood, MA: Artech House, 2002
- [23] Konovaluk M, Gorbunova A, Kuznetsov Y and Baev A. Algorithm of the extraction of information from the complex radar image, *4-th All-Russian conference Radiolocation and radio communication*, Moscow, IRE RAS, Moscow, 2010.
- [24] Efimov E and Shevgunov T. Identification of target scatterers in radar images using radial basis function neural networks. *Trudy MAI*, No. 68, 2013.
- [25] Oppenheim A and Shafer R. *Discrete-Time Signal Processing*. Prentice Hall, 2nd ed., 1999, 870 pp.
- [26] Chan Y, So H and Ching P. Approximate maximum likelihood delay estimation via orthogonal wavelet transform. *IEEE Transactions on Signal Processing*, Vol. 47, No. 4, pp. 1193-1198, 1999.
- [27] Shaltaf S. and Mohammad A. Neural Networks Based Time-Delay Estimation using DCT Coefficients. *American Journal of Applied Sciences*, 6(4), pp. 703-708, 2009.
- [28] Valaytite A, Efimov E and Shevgunov T. Investigation of an Algorithm for Signal Time-Delay Estimation Based on Discrete Cosine Transform with Artificial Neural Network Processing. *Int. conf. on the Radio-Electronic Devices and Systems for the Infocommunication Technology (REDS-2014)*, Moscow, pp. 44-48, 2014.

7 Contact Authors Email Address

For general questions, please, mailto:
shevgunov@gmail.com

Evgeny Efimov: omegatype@gmail.com
Timofey Shevgunov: shevgunov@gmail.com
Alina Valaytite: alinavalaytite@gmail.com
Elena Sadovskaya: elena_93@list.ru

Copyright Statement

The authors confirm that they, and/or their company or organization, hold copyright on all of the original material included in this paper. The authors also confirm that they have obtained permission, from the copyright holder of any third party material included in this paper, to publish it as part of their paper. The authors confirm that they give permission, or have obtained permission from the copyright holder of this paper, for the publication and distribution of this paper as part of the ICAS 2014 proceedings or as individual off-prints from the proceedings.