Application of artificial neural networks to assessment of ship manoeuvrability qualities

Tomasz Abramowski, Ph. D.
Szczecin University of Technology

ABSTRACT

This paper presents an attempt to applying neural networks for assessment of parameters of standard manoeuvrability tests, i.e. circulation test and zig-zag test. Methodological approach to application of neural networks as well as applied network structures and neuron activation functions are generally presented. Also, results of simulations performed by means of the elaborated networks are given in comparison with test cases selected at random. In order to analyze and reveal general trends, correlation relationships between results from network simulations and test cases were calculated and are presented as well.

Keywords: neural networks, ship manoeuvrability qualities

INTRODUCTION

One of the elements subjected to hydromechanical analysis during ship design are its manoeuvrability qualities. In 2002 IMO adopted its MSC.137(76) resolution on "Standards for ship manoeuvrability" which imposed relevant requirements on manoeuvrability parameters of new ships. Therefore it seems reasonable to elaborate tools for assessing such qualities. In this work a method based on application of neural networks was proposed. Such networks have been already used for similar purposes. Clausen *et al*. [3] proposed to use neural networks to determine ship main dimensions in preliminary design stage. Koushan [7, 9] presented possible approximations of propeller-generated pressures and rudder forces, by using neural networks. Cepowski [2] applied the method in question to assessment of ship sea-keeping qualities.

In this work were analyzed possible application of neural networks to two typical manoeuvres performed during sea trials of single propeller ship, namely : the circulation test with ship rudder laid right by 35° and the 10°/10° zig-zag test . The choice of the tests was conditioned by the fact of their standardization in IMO regulations. The presented work was aimed at elaboration of a method making it possible to perform multi-variant, simultaneous analysis of design solutions without necessity of multiple repetition of calculations by means of a simulator.

To simulate and form the networks the MATLAB environment with application of the Neural Network Toolbox package was used [9]. A dynamic simulator being a part of Tribon Initial Design software was used as a source of learning data. The Tribon System is based on numerical solving the well-known differential equations of two-dimensional motion of ship:

$$(m + m_x)(\dot{u} - v\omega - X_G\omega^2) = X_H + X_P + X_R + X_E$$
$$(m + m_y)(\dot{v} - u\omega - X_G\dot{\omega}) = Y_H + Y_R + Y_E \qquad (1)$$
$$(I_z + J_z)\dot{\omega} + mX_G(\dot{v} + u\omega) = N_H + N_R + N_E$$

The right-hand sides of the equations represent the inertial terms including added masses, and their left-hand sides – the hydrodynamic forces : acting on ship hull – (X_H, Y_H, N_H), on propeller – (X_P), on rudder – (X_R, Y_R, N_R), as well as due to such disturbances as wind and wave – (X_E, Y_E, N_E). Despite the learning data were supplemented by available sea trial results, their prevailing part was provided by the simulator. Hence the elaborated networks can be considered to be a simplified algebraic form of the Eqs. (1).

The full set of network learning and testing data comprised 260 cases having systematically changeable parameters. The set was divided in random into the learning set which took part in the process of calculation of network weighing factors and which comprised 230 cases, and the testing set which served solely to assess network quality and which did not take part in learning process and comprised 30 cases. The analysis of parameters in question covered the data presented in Tab.1.

Parameter	Maximum value	Minimum value
L_{pp} [m]	260.0	90.0
B [m]	50.0	12.0
T [m]	15.0	4.6
C_B [-]	0.85	0.6
L_{pp}/B [-]	7.9	4.1
B/T [-]	6.0	2.2
L_{pp}/T [-]	29.8	14.1

The propeller pitch ratio P/D was equal to 1 for all learning cases. The learning data relate to manoeuvres executed in water of unlimited depth. For both the manoeuvres the same rate of laying the rudder, equal to 2°/s, was assumed.

The input data were normalized within the interval (1,-1) in accordance with the following relationship:

$$y = \frac{(y_{max} - y_{min}) \cdot (x - x_{min})}{(x_{max} - x_{min}) + y_{min}} \tag{2}$$

where:

y_{max}, y_{min} - assumed limits of normalization interval, in the considered case: $y_{max} = 1$, $y_{min} = -1$

x_{min} - minimum value in a given row of data,

x_{max} - maximum value in a given row of data.

For all elaborated networks one neuron activation function was applied. It has the following form:

$$\tan \text{sig}(x_i) = \frac{2}{(1 + e^{-2 \cdot x_i}) - 1} \tag{3}$$

where:

x_i - input value.

The run of the applied activation function is shown in Fig. 1.
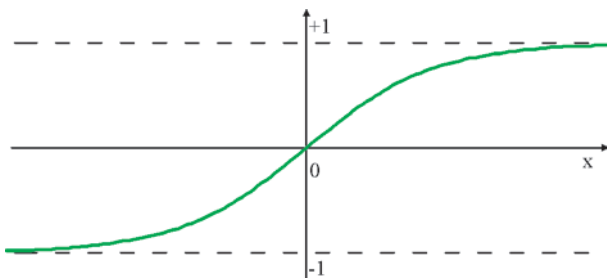

Fig. 1. The applied neuron activation (saturation) function.

ARTIFICIAL NEURAL NETWORKS

There is no commonly accepted, unambiguous definition of artificial neural network (SSN). It can be said that the SSNs are an attempt to connect human brain possibilities of parallel processing with speed and precision of computer processors. Such connection provides, to the networks, capability of learning, memorizing, calling and comparing the nonolinear, multi-dimensional structures of data in order to interpolate and extrapolate them appropriately. The traditionally applied modelling methods consist in collecting data and attempting to combine them into known physical, chemical or mechanical relationships. In such case to linearize mathematical models is necessary to make them simpler for analyzing. The method based on neural networks consists in an attempt to understand a modeled system as an entity of relations occurring within its structure and it searches all linear and nonlinear relations which can influence its operation. Some neural networks constitute models of biological neural structures,but the other ones do not. However historically, most inspirations in the field of neural networks come from a desire of creating an artificial system capable of executing complex, possibly 'intelligent' calculations similar to those regularly done by human brain.

Neural network structure is comprised of several processing units capable of mutual communicating by means of connections having different weighing factors. Generally, not every neural network has its structure similar to that shown in Fig. 2., which consists of the following elements:

★ Input layer – used for receiving and processing external signals,

★ Hidden layers (or only one layer)- whose signals remain within the network,

★ Output layer – used for sending signals outside.

Each layer can be composed of several single neurons, and number of neurons in input and output layers is determined by a kind of task to be solved (number of variables and results).
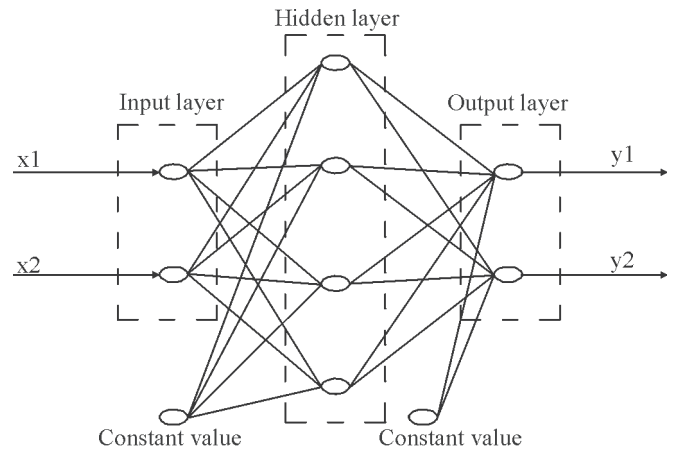

Fig. 2. Example structure of neural network

Neurons of particular layers can be additionally connected with external elements which serve as constant values and improve effectiveness of network learning process. In order to clearly present the operational principle of neural networks, making use of idea of a single neuron is most effective (Fig. 3):
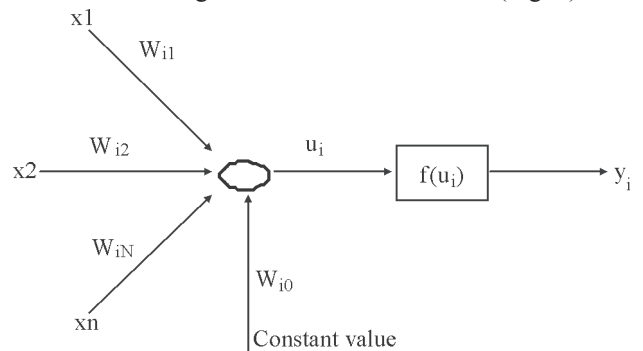

Fig. 3. Single neuron.

Calculation of the output value y_i for a single neuron is performed in accordance with the following relationship:

$$y_i = F\left(\sum_{j=1}^{N} W_{ij} \cdot x_j + W_{i0}\right) \tag{4}$$

where:

F - applied activation function

W_i - weighing factors for particular connections
x_j - input value
W_{i0} - constant values attributed to particular connections.

For given constant values and weighing factors it is easy to generalize the above presented relationship for an arbitrary number of layers and number of neurons in a given layer by using linear algebraic equations. The basic task in applying neural networks is to calculate constant values and weighing factors for a given network structure. The task is realized during learning process. The often used learning method is that called back-propagation one, applied in this work and presented in detail by Osowski [11].

The first operation in the learning process is to prepare two sequencies of data : learning one and verifying one. The learning sequence constitutes a set of such data which relatively exactly characterize a given problem. A single portion of the data is called the learning vector. It contains the input vector, i.e. the input data delivered to network input points, and output vector, i.e. the data expected to be generated by the network at its output points. After processing the input vector, the teacher compares the obtained values and expected ones and informs the network on whether the response is correct or not, if not – on how large error of the response resulted. The error is then propagated within the network but in the opposite sequence than that of the input vector (i.e. from output layer to input one) and on its basis an appropriate correction of weighing factors is introduced to each neuron in order to obtain a lower response error as a result of the repeated processing of the same input vetor. Such procedure is repeated again and again until the error generated by the network is lower than that assumed. Then the successive input vector is delivered to the network input points and the described operations are repeated. After processing the entire learning sequence (called the epoch) the error for the epoch is calculated and the entire cycle is repeated until the error drops under its allowable value. As mentioned above, the SSNs show certain tolerance for discontinuities, random disturbances or even small shortages in the learning set. It just results from their capability of generalizing the knowledge.

The verification process of network operation follows the learning process. In this moment it is important to put in the network some specimens not coming from the training set, in order to check if the network is capable of effective generalizing the learned task. To this end is used the verifying sequence which has the same features as the learning one, i.e. its data charecterize the problem exactly and exact responses are known. However it is important to select the data which have not been used earlier for learning. This way the verifying sequence is presented, however the difference is that errors generated in the process are not propagated back but only number of correct responses is recorded and on this basis the statement is made if the network in question satisfies imposed requirements i.e. to which extent it has been learned.

The initial weighing factors with which a given network starts learning, are numbers generated at random. After learning process it is always favourable to repeat all the procedure beginning from generation of initial weighing factors, in order to check the obtained results.

In the case of large networks and learning sequences consisted of many thousands of learning vectors, number of calculations to be executed during all the learning cycle is gigantic, hence time-consuming. Also, it does not happen that a network is correctly built at once but always it becomes a result of many trials and errors. Moreover it can be never quarranteed that even a correct network does not come to a local minimum instead to continue finding global one. Therefore

algorithms realizing the SSNs are fitted with mechanisms which make it possible to control speed and quality of learning. These are the so called learning and momentum coefficients. They influence steepness of activation function and control speed of influence of change of weighing factors on learning process.

CALCULATIONS FOR CIRCULATION TESTS

To simulate the circulation manoeuvre a network of 9x12x7 structure was elaborated. The input values to the network (arguments) were in turn the following : the ship length between perpendiculars, L_{PP}, ship breadth B, ship draught T, block coefficient of ship hull underwater part, C_B, longitudinal position of ship centre of gravity, L_{CG}, propeller diameter D, ship speed at the beginning of the circulation manoeuvre, V_{APP}, rudder aspect ratio λ and the total rudder area A_R.

The neural network outputs were the following : transverse and head translations (transfer and advance) at ship course change by 90°, tactical diameter of circulation and head translation for the course change by 180°, steady circulation diameter, total decrease of ship speed at course change by 360°, V/V_{APP}, as well as the drift angle β during steady circulation.

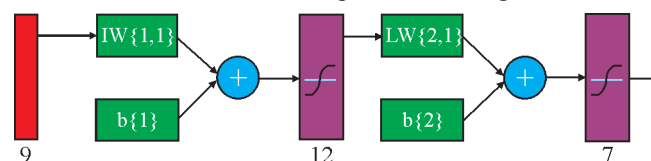The elaborated structure is presented in Fig. 4.

Fig. 4. *Neural network for circulation manoeuvre*

In Fig. 5 the run of the network learning process is shown. It can be observed that after about 300 epochs of learning the network did not exhibit any further improvement of approximation quality.
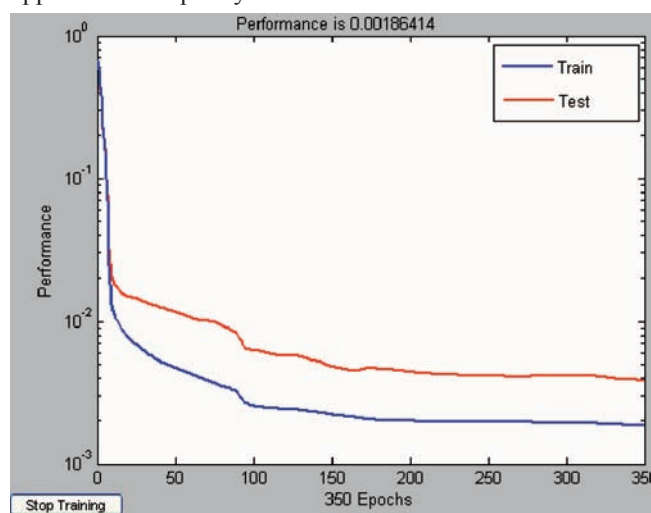

Fig. 5. *Run of network learning process for circulation manoeuvre*

The simulation of the manoeuvre was performed on the basis of a case selected from the testing set. The results of the simulation compared with the expected values are presented in Fig. 6.

It can be observed that for the course angle change by 90° the network error is of the order of the ship breadth. In the case of the course angle change by 180° the error is somewhat smaller - equal to about a half of the ship breadth. The drift angle at steady circulation is of a similar order. The difference of diameters at steady circulation is almost imperceptible. For purposes of this presentation one case characterized by the greatest errors was selected out of 30 testing ones.
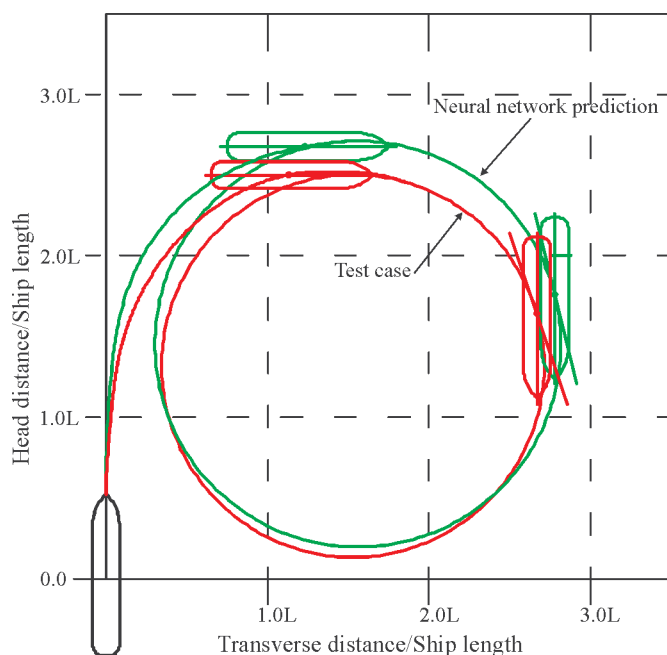
Fig. 6. *Results of simulation of neural network for circulation manoeuvre*

In Fig. 7. is presented the correlation between the drift angle values determined by means of neural networks and its
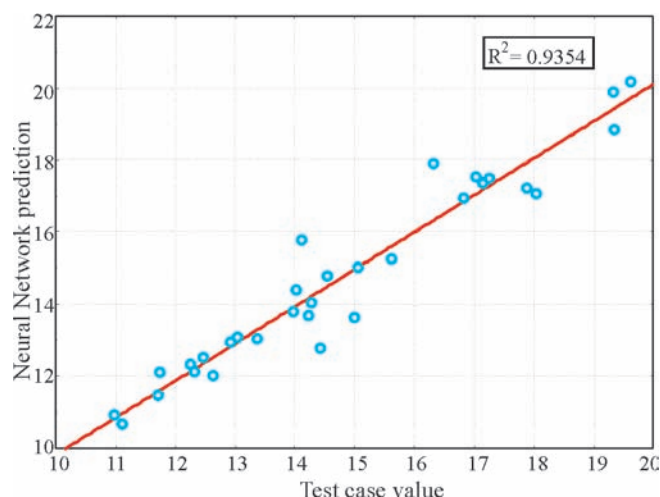

Fig. 7. *Correlation between values of the drift angle at steady circulation, obtained from the network, and relevant testing values.*

testing values, and in Fig. 8 – the correlation of values of the head translation at the course change by 90°.

Example results of network prediction values compared with relevant testing ones are shown in Tab. 2. The values concerning the translations (distances) are related to the ship length. Apart from the tests of correlation between results obtained from the network and testing cases, also capability

Tab. 2. *Example simulation quality of neural network used for assessment of circulation parameters*

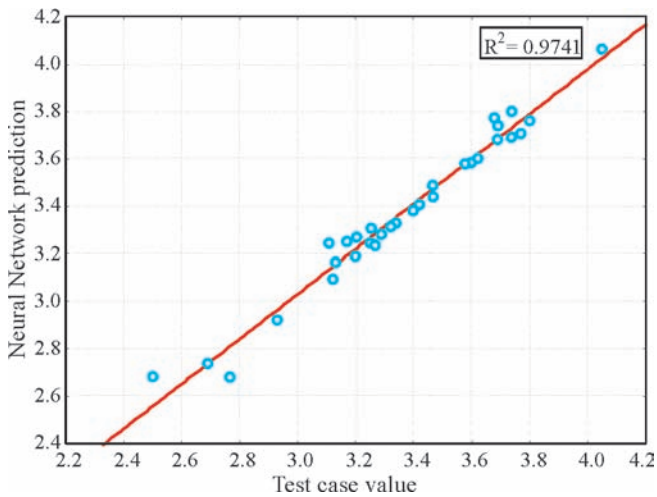|	Course change	Parameter	Values							
Testing values	**90º**	**Advance/L [-]**	3.12	3.27	3.77	3.70	3.69	3.40	2.69	3.11
		Transfer/L [-]	1.54	1.67	1.98	1.89	2.01	1.62	1.19	1.29
	180º	**Tactical diameter [-]**	3.34	3.58	4.19	3.99	4.38	3.52	2.62	2.82
		Advance/L [-]	1.90	2.13	2.34	2.30	2.15	2.35	1.96	2.10
	Steady circulation	**Diameter [-]**	2.62	2.32	2.86	2.64	3.34	2.31	1.65	1.92
		V/Vapp [-]	0.53	0.38	0.44	0.42	0.56	0.37	0.33	0.41
		Drift angle [º]	13.13	15.29	12.96	14.50	12.34	17.42	18.91	16.96
Neural network	**90º**	**Advance/L [-]**	3.09	3.23	3.71	3.6	3.74	3.38	2.74	2.92
		Transfer/L [-]	1.47	1.65	1.94	1.88	2.03	1.62	1.22	1.3
	180º	**Tactical diameter [-]**	3.21	3.57	4.12	4.01	4.41	3.55	2.67	2.85
		Advance/L [-]	1.99	2.09	2.32	2.29	2.19	2.32	1.99	2.06
	Steady circulation	**Diameter [-]**	2.63	2.34	2.86	2.7	3.31	2.38	1.62	1.96
		V/Vapp [-]	0.55	0.38	0.46	0.43	0.55	0.39	0.33	0.43
		Drift angle [º]	13.06	15.63	12.95	13.97	12.25	17.13	19.34	16.84
	Relative error in [%] of a given value		1	1	2	3	1	1	2	7
			5	1	2	1	1	0	3	1
			4	0	2	0	1	1	2	1
			5	2	1	1	2	1	1	2
			0	1	0	2	1	3	2	2
			4	1	5	1	2	5	1	4
			1	2	0	4	1	2	2	1

Fig. 8. *Correlation between values of the head translation at the course change by 90°, obtained from the network, and relevant testing values.*

of the elaborated networks, related to modeling the general trends of circulation parameters, was analyzed. In Fig. 9 is presented an example of the testing of influence of the ship hull block coefficient C_B on circulation parameters by using the elaborated networks.
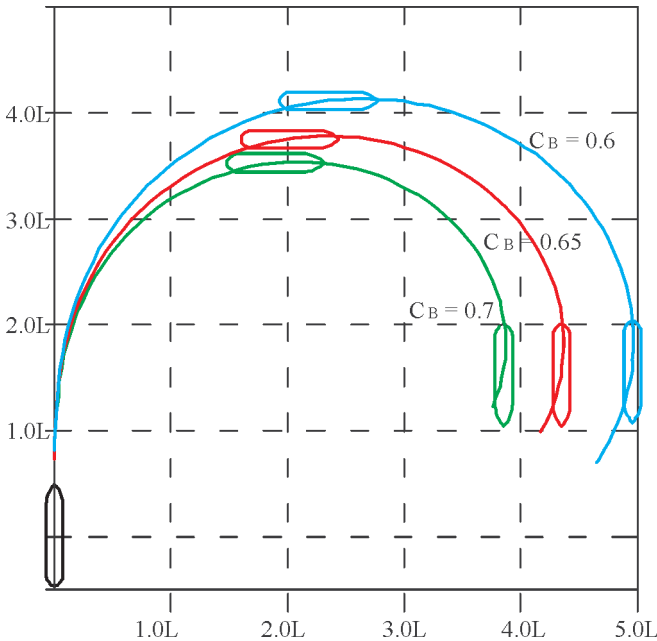

Fig. 9. *Testing of the trend representing capability of the elaborated networks. Influence of ship hull block coefficient.*

CALCULATIONS FOR ZIG-ZAG TESTS

To simulate the zig-zag test the network of 9 x 11 x 5 structure was elaborated. Inputs to the network were identical as for the circulation test. It was the following parameters: duration time of the course change by 10° (initial turning time), t_a; 1st and 2nd overshoot angle, $\Delta\psi_1, \Delta\psi_2$; 1st and 2nd time to check yaw (time of rudder inertia angle), t_1, t_2. The network structure is shown in Fig. 10. The run of the network learning process is shown in Fig. 11. The learning process was terminated after 600 epochs.

In Fig. 12 are presented results of the zig-zag test simulation compared with those of the test case. Errors in representing particular time values and overshoot angles are contained within a few percent interval of relevant values. In Fig. 13 and 14 is

presented correlation between the test values for the 1st and 2nd overshoot angle and the relevant values approximated by the neural network.
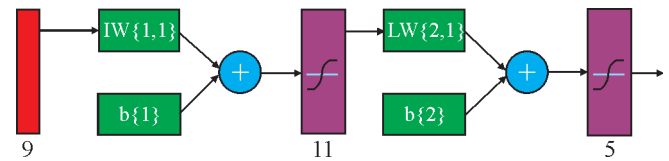

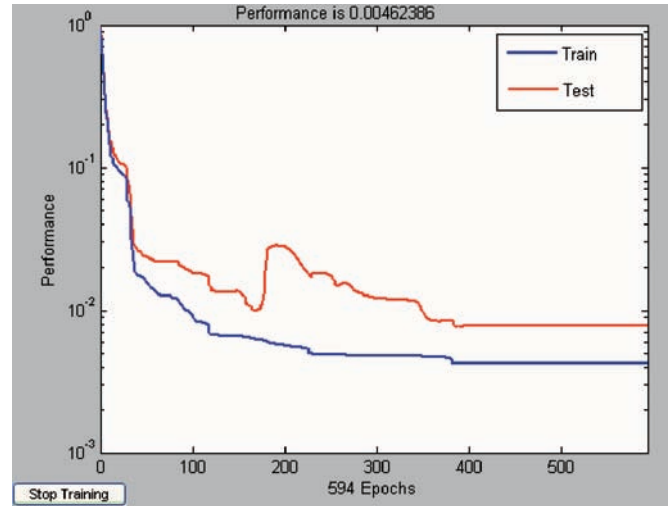**Fig. 10.** *Neural network for simulation of zig-zag test*

Fig. 11. *Run of learning process of the network for zig-zag test*
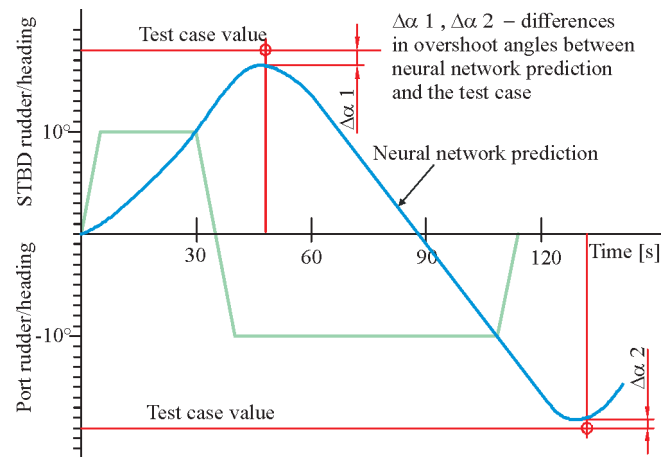

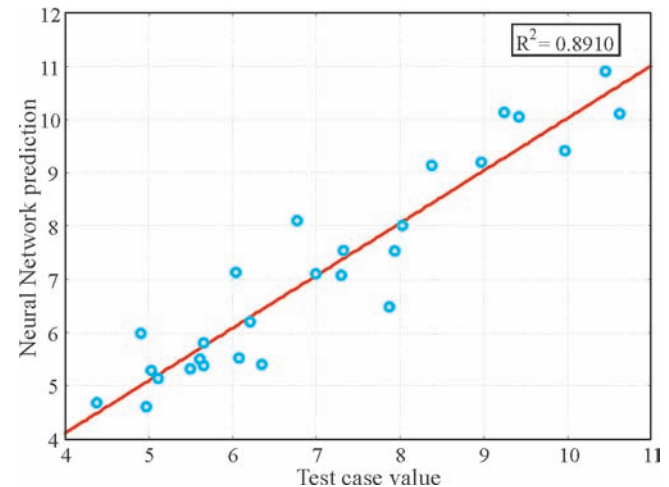**Fig. 12.** *Results of zig- zag test simulation by using the neural network*

Fig. 13. *Approximation quality of the neural network, based on the correlation for the 1st overshoot angle*

The example results of the zig-zag test prediction, compared with relevant test values are given in Tab. 3. The obtained prediction quality is somewhat worse than that in the case of simulation of circulation manoeuvre.

Tab. 3. Example simulation quality of the neural network used for assessment of zig-zag test parameters

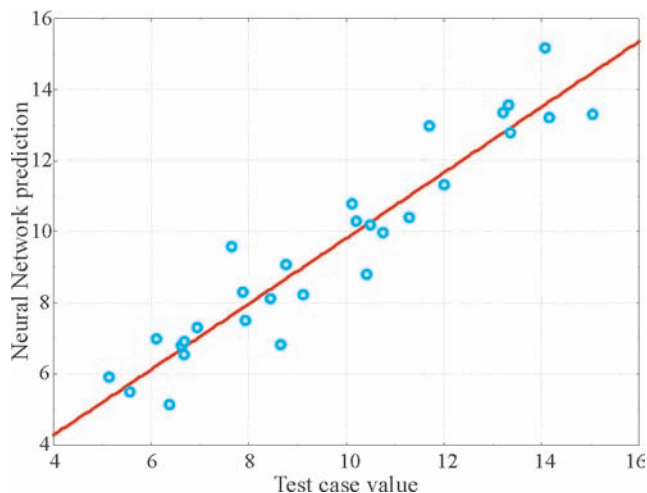|	Parameter	Values							
Test values	1st overshoot angle. $\Delta\psi_1$ [°]	7.87	7.30	6.05	7.33	6.21	11.45	8.38	5.66
	2nd overshoot angle. $\Delta\psi_2$ [°]	9.12	10.49	8.78	10.76	8.44	13.36	11.69	6.68
	Initial Turning Time. t_a [s]	32.00	38.00	62.00	36.00	38.00	28.00	34.00	32.00
	1st Time to check yaw. t_1 [s]	18.00	24.00	34.00	24.00	20.00	22.00	26.00	16.00
	2nd Time to check yaw. t_2 [s]	20.00	28.00	42.00	28.00	22.00	24.00	30.00	16.00
Neural network	1st overshoot angle. $\Delta\psi_1$ [°]	6.50	7.06	7.13	7.55	6.21	9.56	9.16	5.39
	2nd overshoot angle. $\Delta\psi_2$ [°]	8.22	10.23	9.08	9.97	8.10	13.54	12.96	6.54
	Initial Turning Time. t_a [s]	30.00	38.00	58.00	37.00	37.00	28.00	33.00	33.00
	1st Time to check yaw. t_1 [s]	16.00	24.00	34.00	23.00	20.00	20.00	24.00	15.00
	2nd Time to check yaw. t_2 [s]	18.00	28.00	40.00	27.00	22.00	24.00	28.00	16.00
Relative error in [%] of a given value:		21	3	15	3	0	20	9	5
		11	3	3	8	4	1	10	2
		6	0	7	2	3	1	3	3
		13	2	1	4	0	9	8	8
		12	0	5	5	2	1	7	0

Fig. 14. Approximation quality of the neural network, based on the correlation for the 2nd overshoot angle

CONCLUSIONS

❍ Essence of the presented method consists in its possible applications to optimization and multi-criterial automatic design of ships. A difficulty in applying traditional manoeuvrability simulators is their low capability of fast and simultaneous analyzing many design solutions within a similar time interval. This is the feature which must characterize optimization algorithms. Sometimes the feature is more desirable than a high accuracy of results achieved at the expense of large time consumption. In the case of optimization algorithms it is necessary first of all to distinguish a better solution from worse one.

❍ The presented method makes it possible to simultaneously determine parameters of analyzed manoeuvres depending on design parameters of ship. The presented correlation diagrams show that the networks correctly recognize trends and that the obtained results are not accidental.

❍ The simulation quality for circulation manoeuvre is a little better than that for zig-zag test, though it would be possible to improve the simulation quality by supplementing the learning set.

❍ It should be highlighted that the presented method is not aimed at improving ship motion modeling quality but only at improving and adjusting the existing algorithms to demands of design process.

❍ The presented neural networks are capable of exact interpolating however their capability in extrapolating have not been sufficiently recognized so far, hence during application of the method any results obtained for values of the ship parameters being beyond their intervals given in Tab. 1, should be considered very cautiously.

❍ However from another side the traditional manoeuvrability simulators make as a rule use of regression relationships to determine force coefficients. The method in question makes it possible to continue research in the presented direction, e.g. by taking into account corrections associated with water depth, ship trim or application of multi-propeller propulsion systems.

❍ It is also possible to use neural network for dynamic simulation of manoeuvre; in this case for a given ship as

network inputs would be used its kinematic parameters at a given time instant t together with taking into acount instantaneous position of rudder and screw propeller setting, and network outputs – kinematic parameters after a given time interval Δt.

○ Certainly further investigations on the method would be carried out in the direction of bettering the learning data by supplementing them by results from real sea trials, and with a view of presenting example optimization results.

BIBLIOGRAPHY

1. Abramowski T.: *Application of Artificial Neural Networks for Determination of Propeller's Crash-ahead, Crash-back and Backing Performance,* Ship Technology Research, Vol.48, No.4, November 2001
2. Cepowski T.: *Optimization of transport ship design parameters with a view of its sea-keeping qualities* (in Polish), Doctor thesis, Szczecin University of Technology, Szczecin 2002
3. Clausen H.B., Lutzen M., Friis-Hansen A., Bjorneboe N.: *Bayesian and Neural Networks for Preliminary Ship Design.* Marine Technology, vol. 38, no. 4, October 2001
4. IMO Resolution, *Standards for ship manoeuvrability.* MSC.137(76), adopted on 4 December 2002
5. Inoue S., Hirano M., Kijama M., Takashina: *A practical calculation method of ship manoeuvring motion,* Int. Shipbuilding Progress, vol. 28, no 325, 1981
6. McClelland, J., and D. Rumelhart: *Explorations in Parallel Distributed Processing*, MIT Press, Cambridge 1988
7. Koushan, K.: *Prediction of propeller induced pressure pulses using artificial neural networks,* 1st Int. Conf. Computer Applications and Information Technology in the Maritime Industries, Potsdam 2000
8. Koushan K., Mesbahi E.: *Empirical Prediction Methods for Rudder Forces of a Novel Integrated Propeller-Rudder System,* Proc. OCEANS'98, Nice 1998
9. Demuth H., Beale M., Hagan M.: Matlab *User's Guide: Neural Network Toolbox 5*. The MathWorks, 2007
10. Mesbahi E., Atlar M.: *Artificial neural networks: applications in marine design and modelling,* 1st Int. Conf. Computer Applications and Information Technology in the Maritime Industries, Potsdam 2000
11. Osowski, S.: *Neural networks in algorithmic form* (in Polish). Scientific Technical Publishing House (WNT), 1996
12. Tadeusiewicz R.: *Neural networks* (in Polish). Academic Publishing House (Akademicka Oficyna Wydawnicza), Warszawa 1993
13. Sha O.P., Ray T., Gokarn R.P.: *An artificial neural network model for preliminary ship design*, ICCAS 94, 8th Int. Conf. on Comp. Applications in Shipbuilding, Bremen 1994

CONTACT WITH THE AUTHOR

Tomasz Abramowski, Ph. D.
Faculty of Marine Technology,
Szczecin University of Technology
Al. Piastów 41
71-065 Szczecin, POLAND
e-mail: tomasz.abramowski@ps.pl

Photo: C. Spigarski