JOURNAL OF THEORETICAL
AND APPLIED MECHANICS
50, 1, pp. 85-97, Warsaw 2012
50th anniversary of JTAM

MODELLING OF THE DYNAMICS OF A GYROSCOPE USING ARTIFICIAL NEURAL NETWORKS

Łukasz Łacny

Cracow University of Technology, Institute of Applied Mechanics, Kraków, Poland
e-mail: llacny@pk.edu.pl

It this paper, a neural network was utilized in order to create an emulator, which could mimic the behaviour and nonlinear dynamics of a gyroscope with two axes of freedom, subjected to both low- and high-frequency excitation. For this purpose, several known learning methods, such as the gradient and Levenberg-Margquardt method, were used. Three different models of neural networks were considered and compared for their effectiveness: NNFIR, NNARX and the recurrent network NNARMAX.

Key words: artificial neural networks, dynamical systems, emulation, gyroscopes

1. Introduction

Since their emergence, artificial neural networks have been found useful in numerous fields and applications, mostly due to the fact that they can model almost any non-linear function or process, even if it is defined by the measurements and observations only. The applications include: pattern and sequence recognition, data processing and filtering, function approximation, and identification and control of processes (Hagan *et al.*, 2002; Korbicz *et al.*, 1994; Narendra and Parthasarathy, 1990; Norgaard *et al.*, 2000; Omidvar and Elliott, 1997; Passino, 2005; Sarangapani, 2006; Żurada *et al.*, 1996). Nowadays, neural networks are considered for the control of non-linear and adaptive systems. Several teaching algorithms and models have been proposed in the literature for the use in emulators and controllers of dynamical systems, and the results have been discussed, e.g. for the non-linear dynamics of the inverted pendulum (Korbicz *et al.*, 1994).

The dynamics and behaviour of gyroscopes have been widely discussed in literature (Cannon, 1973; Koruba, 2001, 2008; Nizioł, 2005), and several classical methods of controlling them were already considered and used (Koruba, 2001). The aim of the present paper is the discussion of the effectiveness of several differing models of neural networks used for emulating the non-linear dynamics of a gyroscope. To the best of the author's knowledge, this study has not been discussed in previous literature, and it is considered as the first step in the study of a neural controller of a gyroscopic system.

Thanks to the fact that the natural frequency of a gyroscope depends on the angular velocity of the rotor, the analysis of the gyroscopic system has the advantage of allowing the assessment of different neural network models in both low- and high-frequency regimes. With the view of application in distributed-parameter systems, which generally have much higher natural frequencies, the case of high angular velocities is also studied.

The learning techniques used for training the networks include, among others, the steepest gradient and Levenberg-Margquardt algorithms. Both the dynamics of the considered systems, as well as the neural networks, are implemented and calculated using Matlab/Simulink environment.

2. Dynamics of a gyroscope

In the present paper, a gyroscope in a Cardan suspension with two axes of freedom is considered, as shown in Fig. 1. The angular velocity n of the rotor is assumed to be constant throughout the whole time of its operation. Both acceleration and deceleration of the rotor are not included in the following calculations.

The non-linear equations of motion for the gyroscope along with their derivation using d'Alembert's method are presented in the work Cannon (1973), and are given in the following form

$$
\begin{aligned}
&I_y\ddot{\vartheta} + h\dot{\psi}\cos\vartheta + (I_z - I_x)\dot{\psi}^2\sin\vartheta\cos\vartheta = M_y \\
&(I_{oZ} + I_z\cos^2\vartheta + I_x\sin^2\vartheta)\ddot{\psi} - h\dot{\vartheta}\cos\vartheta + 2(I_x - I_z)\dot{\psi}\dot{\vartheta}\sin\vartheta\cos\vartheta = M_Z
\end{aligned}
\tag{2.1}
$$

where ϑ and ψ stand for the rotation angles of the inner and the outer frame, respectively, I_x, I_y and I_z are the overall moments of inertia of the inner frame and the rotor around the x, y and z axes, I_{oZ} is the moment of inertia of the outer frame around the Z axis, while M_y and M_z are the torques applied with

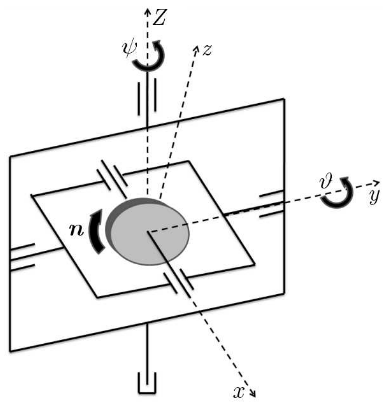


Fig. 1. Model of the gyroscope with two axes of freedom in a Cardan suspension

respect to the y and Z axis, respectively. The parameter h in the equations is the angular momentum of the rotor and is defined using the relation

$$h = I_{rx}n \tag{2.2}$$

in which I_{rx} is the moment of inertia of the rotor around its rotation axis. For this study, the values of the moments of inertia of the gyroscope where chosen as follows

$$I_x = 1.6 \cdot 10^{-4}\,\mathrm{kg\,m^2} \qquad I_y = 1.2 \cdot 10^{-4}\,\mathrm{kg\,m^2} \qquad I_z = 1.2 \cdot 10^{-4}\,\mathrm{kg\,m^2}$$
$$I_{rx} = 10^{-4}\,\mathrm{kg\,m^2} \qquad I_{oZ} = 10^{-5}\,\mathrm{kg\,m^2}$$

Additionally, in the following calculations, it is assumed that no external torque is applied to the rotation axis of the outer frame ($M_z = 0$). The value of the torque M_y applied to the rotation axis of the inner frame ranges from $-10^{-4}\,\mathrm{Nm}$ to $10^{-4}\,\mathrm{Nm}$.

The dynamics of the gyroscope given by the above equations was implemented in Matlab, and the calculations were performed using the solid-step Runga-Kutta integration method with the time step $\Delta t = 10^{-4}\,\mathrm{s}$. Figure 2 shows the response of the gyroscope to the constant torque M_y under different conditions such as the value of M_y and the angular velocity n of the rotor. This response is given in form of the angular displacement of the outer frame ψ (left graph) and the inner frame ϑ (right graph).

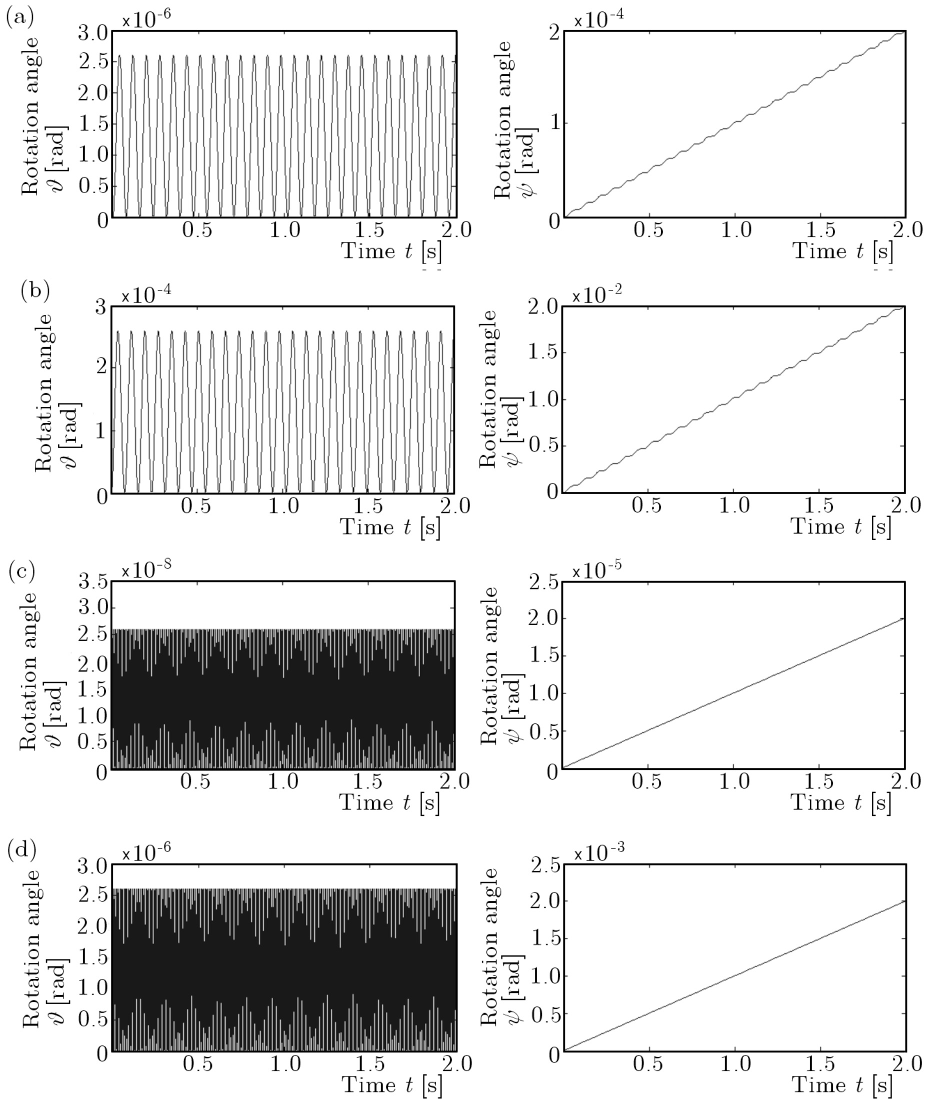


Fig. 2. Response of the gyroscope with angular velocity n to a constant torque M_y in the form of rotation angles of the outer frame ψ (left graph) and the inner frame ϑ (right graph); (a) $n = 100\,\mathrm{rad/s}$, $M_y = 10^{-6}\,\mathrm{Nm}$, (b) $n = 100\,\mathrm{rad/s}$, $M_y = 10^{-4}\,\mathrm{Nm}$, (c) $n = 1000\,\mathrm{rad/s}$, $M_y = 10^{-6}\,\mathrm{Nm}$, (d) $n = 1000\,\mathrm{rad/s}$, $M_y = 10^{-4}\,\mathrm{Nm}$

3. Basic considerations and preparations for teaching the neural network

In order for a dynamical system, such as a gyroscope, to be properly modelled and emulated using a neural network, several essential considerations were required. First of all, the decision had to be made whether to use a two- or a three-layer neural network. While the latter option seemed at first to be more accurate, the comparison proved the former one to be more advantageous in terms of computation time, with no significant loss of precision or performance.

Several already available options were considered regarding the programming and training of the neural network, such as using the functions included in the Neural Network Toolbox of the Matlab software. These however, proved problematic when attempting to train the neural network using the data set obtained from the gyroscope model using the Runga-Kutta method, mostly due to their limited capabilities and long computation time. Therefore, a decision was made to program a separate function used specifically for training the system considered in this paper, and also another one for testing its performance. These functions were based on the computational solutions given in work Norgaard *et al.* (2000) as well as the practical study presented in paper Hagan *et al.* (2002).

Figure 3 shows the structure of the neural network chosen, which has a two-layer architecture (consists of one hidden and one output layer). The operation of this neural network can be written using the following equations

$$\boldsymbol{x}_h = f_1\left(\mathbf{W}_1 \cdot \begin{bmatrix} \boldsymbol{x}_{in} \\ \mathbf{1} \end{bmatrix}\right) \qquad \boldsymbol{x}_{out} = f_1\left(\mathbf{W}_2 \cdot \begin{bmatrix} \boldsymbol{x}_h \\ \mathbf{1} \end{bmatrix}\right) \tag{3.1}$$

in which $\mathbf{W}_1$ and $\mathbf{W}_2$ are the network weight matrices, $\boldsymbol{x}_{in}$, $\boldsymbol{x}_h$, and $\boldsymbol{x}_{out}$ are its input-, hidden- and output layer values, while f_1 and f_2 stand for the activation function of the hidden and output layer, respectively. The bias values for both the hidden and output layer are already included in the respective weight matrices.

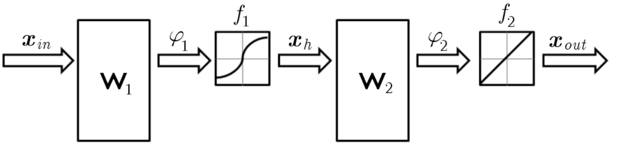


Fig. 3. Structure of the chosen neural network

The amount of neurons in this network was chosen as follows: 10 neurons for the hidden layer and 2 neurons for the output layer. These two output neurons correspond to the rotation speed of the outer and inner frame. The number of inputs, which determines size of the matrix $\mathbf{W}_1$, depends primarily on the model of the neural network chosen. The activation function, was chosen to be tangensoidal for the neurons in the hidden layer, while linear for the ones in the output one, as shown in Fig. 3.

The data required for training the network was obtained by the Runga-Kutta method from the model of the gyroscope implemented in Matlab. In order to provide the neural network with a diverse set of training data, the excitation torque M_y was chosen as a band-limited white noise, with the power spectral density (PSD) of $10^{-4}(\mathrm{Nm})^2/\mathrm{Hz}$ and the sample time $t_s = 10^{-4}$ s, in a time span of 5 s. All the data that were gathered from the model were additionally normalized before training in order for the solution to diverge faster, thus allowing for both the decrease in computation time and increase in the accuracy of the results.

Four different training algorithms were programmed and considered in Matlab: steepest gradient, Gauss-Newton, quasi-Newton and Levenberg-Margquardt algorithm (Norgaard *et al.*, 2000). Although each of these algorithms provided proper training of the network, the last one proved to be the most effective, and therefore was chosen for use in further computations.

4. Modelling the behaviour of the gyroscope using neural networks

Three nonlinear model structures based on neural networks, which are described in detail in book Norgaard *et al.* (2000), were considered for use in emulation of the dynamics of a gyroscope. Each of these is based on the well-known linear model structure and modified in such a way that would allow its use in the nonlinear system. This approach allows the decisions regarding the neural network structure to be limited to two choices only: firstly, how many inputs and of what type should be supplied, and secondly, what architecture of the neural network should be used. While the architecture has already been decided upon (two-layer network with 10 neurons in hidden and 2 in output layer), the number and type of inputs is considered for each model structure separately.

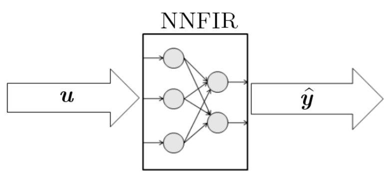
The network was trained using the back-propagation method, based on the error between the desired and predicted output value, averaged over the whole training data set. Two different approaches were used for training the neural network. The first allowed the error to be calculated for all the samples from the data set in one step, and was used for the model structures of a feed-forward type (NNFIR, NNARX). The second one was used exclusively for the model structure using the recurrent neural network (NNARMAX), and required the error to be calculated for each sample, one at a time in a step-by-step manner. In order to provide better comparison, the networks were trained using the same training data set, and the number of iterations over the whole data was also chosen to be the same for each of them.

4.1. NNFIR (Neural Network Finite Impulse Response)

The first of the presented neural network emulators is also the simplest in terms of its structure. As can be seen from Fig. 4, the inputs used for training the neural network consist solely of the excitations applied to the gyroscope model. In this case, these represent the values of torque M_y to which the inner frame of the gyroscope was subjected. The output from the neural network can be therefore described using the following relation

$$\hat{y}(t) = f(u(t), u(t-1), \dots, u(t-k)) \tag{4.1}$$

in which k is the amount of past inputs applied to the considered dynamic object.



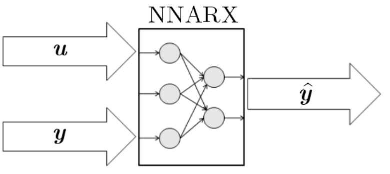
$$\boldsymbol{u} = [u(t), \dots, u(t-k)]$$

Fig. 4. Neural network model structure NNFIR

The knowledge of k past excitations only is not enough to teach the network to emulate the behaviour of the dynamic object properly. The main reason behind this is the lack of necessary information concerning the past states of the object. Due to that, the performance for this NN model cannot be improved even by increasing the number k of past excitations, and the results are far from the desired ones.

4.2. NNARX (Neural Network Auto Regressive, eXternal input)

The second model, presented in Fig. 5, is a logical extension to the NNFIR model. In this case, the artificial neural network is taught by using both past excitations and the time history of the rotation angles ψ and ϑ of the gyroscope, applied as inputs to the network.



$$\boldsymbol{u} = [u(t), \dots, u(t-k)], \quad \boldsymbol{y} = [y(t-1), \dots, y(t-m)]$$

Fig. 5. Neural network model structure NNARX

The function that describes the relation between the output and the inputs of the neural network takes the following form

$$\hat{y}(t) = f(u(t), u(t-1), \dots, u(t-k), y(t-1), \dots, y(t-m)) \tag{4.2}$$

where k is as in NNFIR, is the number of past excitations, while m is the number of the past states (outputs) of the modelled dynamic object. These states correspond to the values of rotation angles of the outer and inner frame of the gyroscope in the m subsequent moments of time.

By utilizing the NNARX model of a neural network, a good approximation of the dynamics of the analysed object can be obtained. This justifies the statement that in order to teach a neural network to simulate the behaviour of a dynamic system, the information concerning its past values of rotation angles needs to be provided.

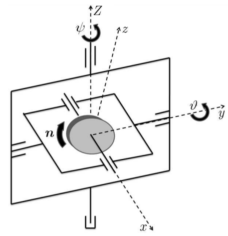
4.3. NNARMAX (Neural Network Auto Regressive, Mean Average, eXternal input)

Figure 6 shows the structure of the last of the considered emulators. It differs from the other two, as it is not a feed-forward network, but a recurrent one, in which some of the values obtained at the output of the network are redirected to its input, and play a crucial role in its operation.

The structure of this model is the same as in NNARX, but with an addition of the past r residual values (difference between the physical output y and the neural network output $\hat{y}$), used as inputs. The output function to this emulator can be written as follows

$$\hat{y}(t) = f(u(t), \dots, u(t-k), y(t-1), \dots, y(t-m), e(t-1), \dots, e(t-r)) \tag{4.3}$$

The approximation using NNARMAX model is similar in accuracy to that obtained by using NNARX. However, due to the recurrent nature of the network, several drawbacks are present. First of all, the training and recalculation



$$\boldsymbol{u} = [u(t), u(t-1), \ldots, u(t-k)], \quad \boldsymbol{y} = [y(t-1), y(t-2), \ldots, y(t-m)]$$
$$\boldsymbol{e} = [e(t-1), e(t-2), \ldots, e(t-r)]$$

Fig. 6. Neural network model structure NNARMAX

of the network needs to be performed for each step in the data set. Secondly, the matrix constructed for training the network might become singular, which could prevent the solution from being found. Lastly, the network requires to be directed towards the solution, which in this case is performed by training the network using NNARX model for the few iterations, and then expanding it to the NNARMAX model.

4.4. Comparison of the model results for low- and high-frequency excitations

The training and comparison of the models was performed for the constant angular velocity of the rotor equal to 1000 rad/s. The amount of input values for each one of the models was specified as follows:

- NNFIR: $k = 2$
- NNARX: $k = 2,\ m = 2$
- NNARMAX: $k = 2,\ m = 2,\ r = 2$

As expected, NNFIR was the only one of the three models that was not able to emulate the behaviour of the gyroscope, and therefore it was not included in the results. NNARX and NNARMAX, on the other hand, performed quite well and allowed the dynamics of the gyroscope to be modelled and emulated with good accuracy and precision.

Figure 7 shows the response of the gyroscope (solid line) and the emulator (dashed line) to a constant torque M_y of value 10^{-6}, 10^{-5} and 10^{-4} Nm, from top to bottom, respectively. The results for the emulators correspond to those obtained for the physical model as shown in Fig. 2. The difference in the response can be easily attributed to the limitations of neural network training. When the boundaries within which the network was trained are reached, a divergence between the response of the emulator and gyroscope can be seen (in Fig. 7 it happens when the rotation angle $\psi = 10^{-4}$ rad). However, up to that point, both of the responses are nearly identical. If one required better performance, it could be done by widening the range of outputs used for training the emulator.

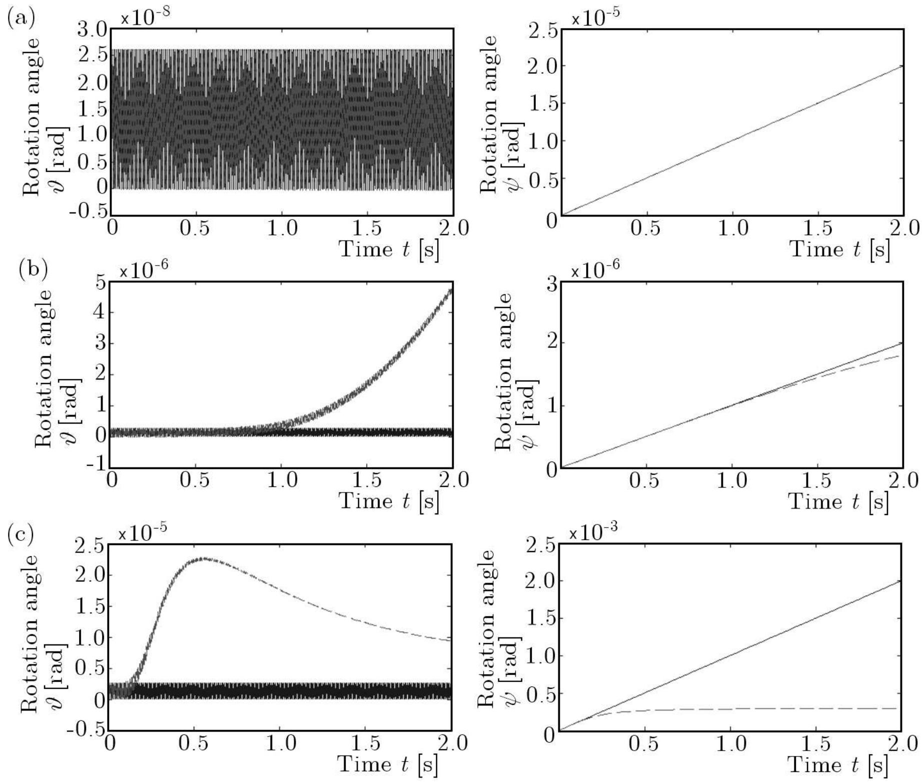


Fig. 7. Response of the gyroscope (solid line) and the emulator (dashed line) to the constant torque M_y; (a) $M_y = 10^{-6}$ Nm, (b) $M_y = 10^{-5}$ Nm, (c) $M_y = 10^{-4}$ Nm. All plots for the rotor angular frequency $n = 1000$ rad/s

In Fig. 8, the response to the torque M_y given by the sinusoidal and chirp function, is presented. As in this case, the results were identical for each model (original, NNARX emulator, NNARMAX emulator), only one result is shown for each type of excitation. It should be noted that even for a high frequency excitation, both of these models perform well and give a proper and accurate response.

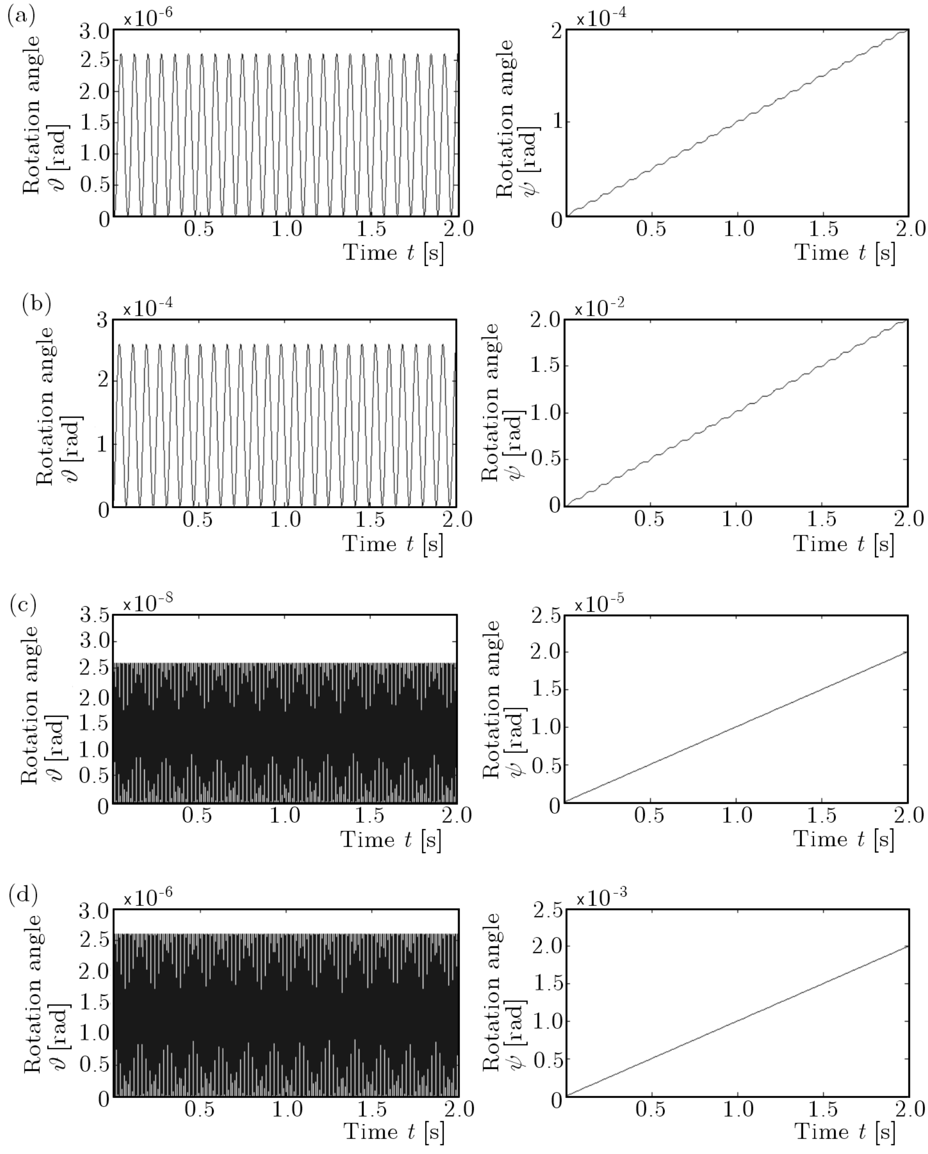


Fig. 8. Response of the emulator to the specific external torque M_y; (a) sinusoidal function, amplitude 10^{-5} Nm, angular frequency 100 rad/s, (b) sinusoidal function, amplitude 10^{-5} Nm, angular frequency 1000 rad/s, (c) chirp function, amplitude 10^{-4} Nm, frequency 0-1000 Hz in 2 s, (d) chirp function, amplitude 10^{-4} Nm, frequency 0-1000 Hz in 2 s. All plots for the rotor angular frequency $n = 1000$ rad/s

5. Summary

Three different structure types of neural networks were considered and used for emulating the dynamics of the gyroscope with two axes of freedom, subjected to an external excitation. The first model (NNFIR) lacked the necessary knowledge required for training the dynamic object. The other two models (NNARX, NNARMAX) proved to be capable of approximating and emulating the behaviour of the gyroscope. However, the latter model has shown several drawbacks that made it less useful than the former one.

Earlier studies of the application of neural networks in the emulators of nonlinear dynamical systems have been limited to low frequencies. The present analysis has shown that by utilizing available model structures of neural networks it is also possible to emulate high-frequency vibration. This is of much importance in the design of neural controllers for distributed-parameter systems (e.g. smart structures). While the results in this study were obtained for the angular velocity of 1000 rad/s, the emulators perform equally good for the lower frequencies.

Acknowledgement

The support of dr Piotr Cupiał in formulation of this subject and during the work on it is gratefully acknowledged.

References

1. Cannon R.H. jr., 1973, *Dynamika układów fizycznych*, WNT, Warszawa
2. Hagan M.T., Demuth H.B., de Jesus O., 2002, An introduction to the use of neural networks in control systems, *International Journal of Robust and Nonlinear Control*, **12**, 11, 959-985
3. Korbicz J., Obuchowicz A., Uciński D., 1994, *Sztuczne sieci neuronowe. Podstawy i ich zastosowania*, Akademicka Oficyna Wydawnicza PLJ, Warszawa
4. Koruba Z., 2001, *Dynamika i sterowanie giroskopem na pokładzie obiektu latającego*, Wydawnictwo Politechniki Świętokrzyskiej, Kielce
5. Koruba Z., 2008, *Elementy teorii i zastosowań giroskopu sterowanego*, Wydawnictwo Politechniki Świętokrzyskiej, Kielce
6. Narendra K.S., Parthasarathy K., 1990, Identification and control of dynamical systems using neural networks, *IEEE Transactions on Neural Networks*, **1**, 1, 4-27

7. Nizioł J., edit., 2005, *Mechanika Techniczna. Tom II – Dynamika układów mechanicznych*, IPPT PAN, Warszawa

8. Norgaard M., Ravn O., Poulsen N.K., Hansen L.K., 2000, *Neural Networks for Modelling and Control of Dynamic Systems*, Springer-Verlag, London

9. Omidvar O., Elliott D.L., 1997, *Neural Systems for Control*, Morgan Kaufmann Publishers Inc., San Francisco

10. Passino K.M., 2005, *Biomimicry for Optimization, Control and Automation*, Springer-Verlag, London

11. Sarangapani J., 2006, *Neural Network Control of Nonlinear Discrete-Time Systems*, CRC Press, Taylor & Francis Group, New York

12. Żurada J., Barski M., Jędruch W., 1996, *Sztuczne sieci neuronowe, Podstawy teorii i zastosowania*, Wydawnictwo Naukowe PWN

Zastosowanie sztucznych sieci neuronowych do modelowania dynamiki giroskopu

Streszczenie

W niniejszej pracy przedstawiono, w jaki sposób przy użyciu sztucznej sieci neuronowej możliwe jest stworzenie emulatora, który naśladuje zachowanie i nieliniową dynamikę giroskopu o dwóch osiach swobodnych, poddanego wymuszeniom zarówno o niskiej, jak i wysokiej częstotliwości. W celu nauczenia sieci neuronowej, wykorzystano szereg dostępnych algorytmów uczących (m.in. gradientowy, Levenberga-Margquadta). Przetestowano oraz porównano trzy różniące się od siebie modele sieci neuronowych: NNFIR, NNARX oraz sieć rekurencyjną NNARMAX.

Manuscript received November 26, 2010; accepted for print April 27, 2011