Neural Comput & Applic (2014) 24:1327–1340
DOI 10.1007/s00521-013-1348-4

ORIGINAL ARTICLE

Influence of the optimization methods on neural state estimation quality of the drive system with elasticity

Teresa Orlowska-Kowalska · Marcin Kaminski

Received: 5 April 2012 / Accepted: 25 January 2013 / Published online: 16 February 2013

Abstract The paper deals with the implementation of optimized neural networks (NNs) for state variable estimation of the drive system with an elastic joint. The signals estimated by NNs are used in the control structure with a state-space controller and additional feedbacks from the shaft torque and the load speed. High estimation quality is very important for the correct operation of a closed-loop system. The precision of state variables estimation depends on the generalization properties of NNs. A short review of optimization methods of the NN is presented. Two techniques typical for regularization and pruning methods are described and tested in detail: the Bayesian regularization and the Optimal Brain Damage methods. Simulation results show good precision of both optimized neural estimators for a wide range of changes of the load speed and the load torque, not only for nominal but also changed parameters of the drive system. The simulation results are verified in a laboratory setup.

Keywords Neural networks · State estimation · Electrical drive · Two-mass system · Training methods · Bayesian regularization · Optimal Brain Damage method

1 Introduction

In most electrical drives, the elasticity of the shaft between a driving motor and a load machine must be taken into account. In order to obtain drive response to a reference signal with high dynamics, and to minimize torsional vibrations, different control methods of the drive system with elastic joint, based on control theory, like PI/PID methods, state controller-based methods, sliding-mode, and adaptive or predictive control methods [1–6] are used. All these control methods require feedbacks from different mechanical state variables of the system (load side speed, torsional torque, load torque). These mechanical variables can be measured, but only in laboratory environments. In the real drive systems, in industry, torsional or load torque can not be measured, as the torque transducer is never mounted between the driven motor and the loading machine because lack of space and generation of additional (high) cost. Similarly, the load side speed is hardly measured because lack of place for additional speed transducer and additional cabling, which is troublesome. In such a case, only estimation of those state variables is the solution for the industry conditions. This is the reason why we have to estimate the torsional torque and the load side speed of a two-mass system.

In many applications connected with electrical drives, algorithmic methods are applied for the non-measurable state variables estimation, for example, the Kalman filters [4, 5] and the Luenberger observers [6]. However, the algorithmic estimators require the mathematical model and parameter knowledge of the system, which could change during the system operation—so to obtain the good estimation quality the parameters of the state estimators must be tuned on-line (by on-line plant parameters' identification or estimation). Alternative ways of solving this problem are estimators based on neural networks (NNs). Such estimators do not need a mathematical model and parameters of the system, only the training data are required [7–9] for the estimator design. Moreover, the generalization

T. Orlowska-Kowalska (✉) · M. Kaminski
Institute of Electrical Machines, Drives and Measurements,
Wroclaw University of Technology, Wroclaw, Poland
e-mail: teresa.orlowska-kowalska@pwr.wroc.pl

M. Kaminski
e-mail: marcin.kaminski@pwr.wroc.pl

ability causes that neural estimators are less sensitive to parameters or measurement signals uncertainties.

However, in the case of NN applications in state variable estimation, the determination of NN structure for a specific task is one of the most important problems. This structure should be carefully chosen to obtain good estimation quality also in the case of NN input data different than those used in the training procedure. It means that a suitable generalization ability is required. Data generalization is one of the main advantages of the NN and consists in the possibility of solving a given task by a trained network in case the elements of the input vector are not taken into account in the NNs training process. In the technical literature, many methods for the improvement of the NN generalization properties are presented. It is possible to distinguish three main trends [7]:

- impact on the length of the learning process (early stopping) [10],
- application of regularisation method [11],
- modification of neural networks topology (growing or pruning) [12, 13].

Many methods for NN structure optimization are presented in the literature. Most of them require the initial choice of NN structure, and then, selected neural connections are eliminated. One of the simplest ways to choose a specific inter-node connection for elimination is the analysis of absolute values of NNs' weights. Another method consists in checking the influence of each connection on the generalization error. In this case, the generalization errors before and after the elimination each weight factor are compared [7, 14].

Very good results are obtained with the sensitivity methods. These algorithms are based on the analysis of sensitivity of the cost function to deletion of individual connections. The most important methods in this category are the Optimal Brain Damage (OBD) [15, 16] and Optimal Brain Surgeon [17, 18] methods.

In many techniques, genetic algorithms are also applied for pruning the inter-neural connections [19].

The other solution is adding the regularization element to the cost function [17]. It consists in the modification of the objective function used in the training algorithm, which is next minimized in any iteration. In the extended form of such cost function, elements dependent on values of the inter-neural connection weights are added to the standard cost function; then, the problem of the selection of regularization parameters in the modified objective function appears. In this work, the regularization method based on the Bayesian interpretation of NNs is applied. This algorithm gives analytical formulas for automatic computation of optimal regularization parameters [20, 21].

It is reported in the literature that the Bayesian regularization method can significantly improve the quality of state variable estimation. So in this paper, the effectiveness of this method is compared with the previously used OBD method (which is rather complicated in practice [16]) for the NN state estimators of a two-mass drive system.

This paper presents neural estimators of the torsional torque and the load machine speed for a drive system with elastic joints. These neural estimators are trained with classical Levenberg–Marquardt method [7] and next they are optimized using OBD and Bayesian regularization methods. The obtained estimators are tested in the open-loop and closed-loop control structure with additional feedback adjusted suitably for damping the torsional vibration of the drive system with elastic coupling between the driven motor and the load mechanism.

The paper is divided into seven sections. After a short introduction, the mathematical model of the two-mass drive system is presented. Then, the speed control structure with a state controller and feedbacks from the motor speed, shaft torque, and the load speed are described. These two last state variables are estimated by the tested NN, and the motor speed is measured directly as well as the motor current, which form the input vectors of NN estimators. In the next part, the discussion of the NN input vector selection for the analyzed task is presented. In the forth part, the chosen methods for the improvement of the NN generalization properties are described. This paper is focused on two methods: the Bayesian regularization and the OBD method. The designed NN estimators are next implemented in the control structure and tested under simulation (section five) and experimental tests (section six). The paper is completed with short conclusions.

2 Design of the state controller for a two-mass drive system

The electrical drive with anelastic joint can be described by different mathematical models, depending on the exactness of the elastic shaft modeling. Usually, such drive is analyzed as a system composed of two masses connected by an elastic shaft, where the first mass represents the moment of inertia of the drive and the second mass refers to the moment of inertia of the load side (see Fig. 1). It is assumed that value of the moment of inertia of elastic shaft J_c is much smaller than the moments of inertia of the driving motor J_1 and the load machine J_2. This assumption involves the neglecting of the moment of inertia of the elastic shaft.

For the further considerations, the damping coefficient D of this elastic shaft is assumed as equal to zero, which

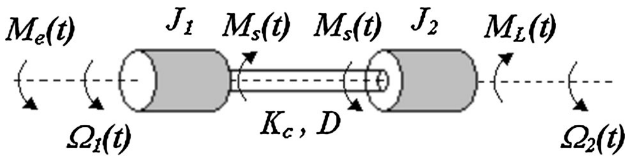


Fig. 1 The schematic diagram of the two-mass system

leads to enlarging the influence of elasticity of the shaft on the drive system operation. Moreover, the nonlinear phenomena, like friction and backlash, are omitted; thus, the mechanical part of the considered two-mass drive can be described by the following state equation, in the per unit system, using the following notation of new state variables [2]:

$$T_1 \frac{\mathrm{d}\omega_1(t)}{\mathrm{d}t} = m_e(t) - m_s(t), \tag{1}$$

$$T_2 \frac{\mathrm{d}\omega_2(t)}{\mathrm{d}t} = m_s(t) - m_L(t) \tag{2}$$

$$T_c \frac{\mathrm{d}m_s(t)}{\mathrm{d}t} = \omega_1(t) - \omega_2(t) \tag{3}$$

with:

$$\omega_1 = \frac{\Omega_1}{\Omega_N}, \omega_2 = \frac{\Omega_2}{\Omega_N}, m_e = \frac{M_e}{M_N}, m_s = \frac{M_s}{M_N}, m_L = \frac{M_L}{M_N} \tag{4}$$

where Ω_1, Ω_2, Ω_N—motor speed, load side speed, and nominal speed of the motor (rad/s), M_N—nominal torque of the motor (Nm), ω_1, ω_2—motor and load speeds, m_e, m_s, m_L—electromagnetic, shaft, and load torques in the per unit system.

The mechanical time constant of the motor—T_1, the load machine—T_2 and the stiffness time constant—T_c are thus given as:

$$T_1 = \frac{\Omega_N J_1}{M_N}, T_2 = \frac{\Omega_N J_2}{M_N}, T_C = \frac{M_N}{K_c \Omega_N}. \tag{5}$$

where ω_1, ω_2—the motor and load speeds, m_s, m_L—the shaft and load torques, T_1, T_2—the mechanical time constants of the motor and load machine, T_c—the stiffness time constant.

The block diagram of such system with elastic connection between the motor and the load machine is shown as the part of Fig. 2 (dashed-line rectangular).

The classical cascade control structure of such drive system consists of two major control loops: the inner control loop contains the current controller, the power converter, and the motor. After optimization, the current (or torque) control loop can be replaced by the first-order inertial block with small time constant. During the design process of the speed loop, the dynamics of the torque loop is very often neglected [2]. In most cases, the PI speed controller is used in the external control loop. In this paper, the state-space controller with an integral action for steady-state error elimination is applied for the speed control of the drive system with elasticity (see Fig. 2).

Taking into account the equation for the required value of the electromagnetic torque, generated by the motor (with neglected dynamics of the torque loop and negative feedbacks from all state variables):

$$m_e = K_i \int (\omega_r - \omega_2)\mathrm{d}t - k_1\omega_1 - k_2 m_s - k_3\omega_2 \tag{5}$$

Introducing the Laplace transform for the mathematical model of the drive system (1–3) and (5), we obtain:

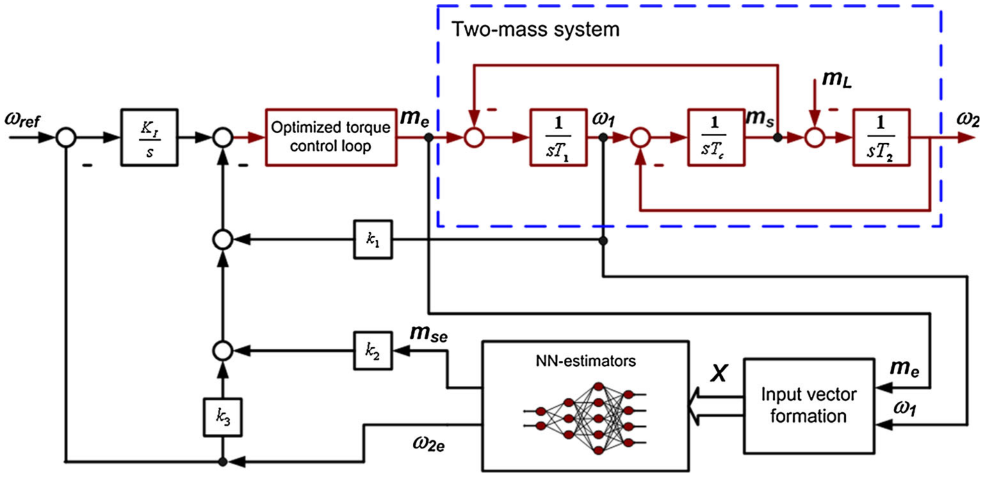
$$T_1 s\omega_1 = m_e - m_s \tag{6}$$

$$T_2 s\omega_2 = m_s - m_L \tag{7}$$

$$T_c s m_s = \omega_1 - \omega_2 \tag{8}$$

$$m_e = R(\omega_r - \omega_2) - k_1\omega_1 - k_2 m_s - k_3\omega_2 \tag{9}$$

Fig. 2 A block diagram of the control structure with a speed state controller and neural estimators for two-mass drive



where

$$R = \frac{K_i}{s}. \tag{10}$$

and s—operator of the Laplace transform.

Including (9) in (6), the following set of equations is obtained:

$$T_1 s\omega_1 + m_{\mathrm{s}} = R(\omega_r - \omega_2) - k_1\omega_1 - k_2 m_s - k_3\omega_2 \tag{11}$$

$$m_{\mathrm{s}} = T_2 s\omega_2 + m_{\mathrm{L}} \tag{12}$$

$$T_c s m_{\mathrm{s}} = \omega_1 - \omega_2 \tag{13}$$

Introducing (12) in (11) and (13), we obtain:

$$\begin{aligned} T_1 s\omega_1 + T_2 s\omega_2 + m_{\mathrm{L}} = R(\omega_r - \omega_2) - k_1\omega_1 - k_2 T_2 s\omega_2 \\ - k_2 m_{\mathrm{L}} - k_3\omega_{22} \end{aligned} \tag{14}$$

$$\omega_1 = T_c T_2 \omega_2 s^2 + T_c m_{\mathrm{L}} s + \omega_2 \tag{15}$$

then transforming this set of equations, Eq. (16) is obtained:

$$\begin{aligned} &\omega_2\left(T_1 T_2 T_c s^3 + T_1 s + T_2 s + R + k_1 T_c T_2 s^2 + k_1 + k_2 T_2 s + k_3\right) \\ &\quad = R\omega_r - k_1 T_c s m_{\mathrm{L}} - k_2 m_{\mathrm{L}} - m_{\mathrm{L}} - T_1 T_c s^2 m_{\mathrm{L}}, \end{aligned} \tag{16}$$

which enables the determination of the transfer function of the closed-loop control system with R replaced by (10):

$$\frac{\omega_2}{\omega_r} = \frac{K_i}{s^4 T_1 T_2 T_c + s^3 k_1 T_c T_2 + s^2(T_1 + T_2 + k_2 T_2) + s(k_1 + k_3) + K_i} \tag{17}$$

The characteristic equation of this transfer function has the following form:

$$\begin{aligned} H(s) = s^4 + s^3\frac{k_1}{T_1} + s^2\left(\frac{1}{T_2 T_c} + \frac{1}{T_1 T_c} + \frac{k_2}{T_1 T_c}\right) \\ + s\left(\frac{k_1}{T_1 T_2 T_c} + \frac{k_3}{T_1 T_2 T_c}\right) + \frac{K_i}{T_1 T_2 T_c}. \end{aligned} \tag{18}$$

In order to calculate the expressions defining gains of the designed state controller, the characteristic equation of the closed-loop system (18) has to be compared to the reference polynomial of the same order. The following form of this polynomial was taken into account:

$$\begin{aligned} H_{\mathrm{ref}}(s) &= \left(s^2 + 2\xi_r\omega_o s + \omega_o^2\right)\left(s^2 + 2\xi_r\omega_o s + \omega_o^2\right) \\ &= s^4 + s^3(4\xi_r\omega_o) + s^2\left(2\omega_o^2 + 4\xi_r^2\omega_o^2\right) + s(4\xi_r\omega_o^3) \\ &\quad + \omega_o^4, \end{aligned} \tag{19}$$

where ξ_r, ω_o—are the required damping factor and resonance frequency of the closed-loop system.

Comparing the elements with the same power of the Laplace operator s, the following expressions for the suitable gains of the state controller can be obtained:

$$K_I = T_1 T_2 T_c \omega_o^4. \tag{20}$$

$$k_1 = 4T_1\zeta_r\omega_o, \tag{21}$$

$$k_2 = T_1 T_c\left(2\omega_o^2 + 4\zeta_r^2\omega_o^2 - \frac{1}{T_c T_2} - \frac{1}{T_c T_1}\right), \tag{22}$$

$$k_3 = \omega_o^2 k_1 T_2 T_c - k_1, \tag{23}$$

Feedbacks from all mechanical state variables of the two-mass system are introduced to the external control loop, so the information about the shaft torque m_{s}, motor speed ω_1, and load speed ω_2 is needed. Measurement of the motor speed ω_1 is simple and trouble-free, but the measurement of the shaft torque and the load speed can be difficult or expensive in the industrial practice. In this case, we can use special estimation structures based on neural networks to estimate these variables, based on easily measurable driven motor speed and current (electromagnetic torque of the driven motor is proportional to this current). So in the control structure, we will use the measured motor speed ω_1 and the estimated variables, like shaft torque m_{se} and load speed $\omega_{2\mathrm{e}}$ (see Fig. 2).

3 Neural network based state variables estimators

As it was said before, the mechanical state variables required for feedback signals in the control structure of the drive system with an elastic joint have been estimated by NN-based estimators. For this research, the feed-forward NNs were selected. The previous research shows that this type of NN can give a high precision of the state variables estimation of the two-mass drive system [8], but the selection of proper NN structure is difficult and usually done by trial and error, which is a time-consuming method. To avoid this problem, we have selected some structure of NN (after a few preliminary simulation tests) and next tried to optimize this structure using two optimization methods, well known from the neural networks theory.

Starting structures are the same for both presented estimators—for the load side speed $\omega_{2\mathrm{e}}$ and the shaft torque m_{se}: {6-10-12-1}—6 inputs, 10 neurons in the 1st hidden layer, 12 neurons in the 2nd hidden layer, and 1 neuron in the output. For the hidden layers, the nonlinear tangensoidal activation functions are applied. The linear activation function is selected as the output function of the considered neural estimators.

The proper selection of elements of the input vector of neural network is very important for correct realization of the required task. The selection of input elements in the design process of neural estimators should take into account the properties of NNs and practical aspects of the analyzed implementation. It should be noticed that the expansion of the input vector of NN can influence the structure of the net, in result it influences on its practical implementation (e.g., using FPGA—for the time of calculation and consumption of resources). At the same time, in the case of an expanded input vector, results of NN calculation can improve slightly, or—on the contrary—they can be even worse. From the engineering point of view, signals included in the NN input vector should be selected carefully to fulfill the following conditions:

- give an important information about changes of the state variables of the process
- they should be easily measured in the real system.

According to these requirements, the input signals of the neural networks in our case are the motor speed ω_1 and the electromagnetic torque m_{e} (or stator current) of the driven motor.

In the presented application of neural estimation, MLNN (multi-layer neural networks) were implemented. It should be noted that the NNs analyzed in the described application are static systems; they do not have internal feedbacks or memories. On the other hand, the presented application is focused on dynamical signals of the drive system, quickly changing in time. Therefore, to take into account the dynamics of the processed signals and to obtain better quality of the state variable estimation, the input vector of MLNN was extended with the delayed samples of input variables (motor speed and electromagnetic torque). So the form of the assumed input vector is described by the following equation:

$$\mathbf{X} = [\omega_1(k), \omega_1(k-1), \omega_1(k-2), m_{\mathrm{e}}(k), m_{\mathrm{e}}(k-1), m_{\mathrm{e}}(k-2)] \tag{24}$$

The number of historical samples was selected experimentally. For the analyzed data, type of NNs and number of iterations in the training process, the best results were obtained for input vector described by expression (24). Tests of estimators without delayed samples of input signals in the processed vector lead to much worse results. On the other hand, increasing the number of historical samples [in comparison with Eq. (24)] only slightly influences the precision of estimation and is not necessary from the point of view of practical implementation.

The Levenberg–Marquardt (LM) learning algorithm is used to train the NN state estimators. The value of each weight coefficient is adjusted according to LM, while the error backpropagation (EBP) is used to calculate the Jacobian matrix of the cost function with respect to the weight values. The updating rules of NN weights $\mathbf{w}$ are presented below:

$$\Delta\mathbf{w} = -(\mathbf{J}^T\mathbf{J} + \eta\mathbf{I})^{-1}\mathbf{J}^T\mathbf{e} \tag{25}$$

where $\mathbf{J}$—Jacobian matrix of the cost function E with respect to the weight values, η—learning factor, $\mathbf{I}$—identity matrix, $\mathbf{e}$—difference between target output of the training data and the network output.

Next, the previously selected structure of NN estimators was optimized using the Bayesian regularization and OBD methods. The effectiveness of these methods in the described task has been compared and evaluated.

4 Optimization methods used for neural estimators

4.1 Bayesian regularization method

The neural networks training process can be defined as a minimization of the objective function. In the considered case, the analyzed cost function is described by a following equation:

$$F = \beta E_D + \alpha E_W \tag{26}$$

where element E_D is a sum of squares of NN calculation errors for each input sample, and E_W is an additional regularization term presented below:

$$E_D = \sum_{j=1}^{M} (d_j - y_j)^2, \tag{27}$$

$$E_W = \sum_{i=1}^{W} w_i^2, \tag{28}$$

where d_j—desired output values; y_j—actual output values of the neuron; M—dimension of the vector $\mathbf{d}$, w_i—weights; W—the total number of weight and biases in the network.

In relation to the objective function (26), the problem of selecting parameters α and β appears. The regularization parameters describe the influence of suitable terms E_R and E_D on the cost function. The first one decides about NN exactness in respect to the training data, and the second one enforces the smoothness of NN output [20, 21]. If α is relatively significant in comparison with β, the training error is smaller and the effect is like in a classical algorithm. In the other case, the training process gives smaller weights and leads to a smoother network output. Therefore, the optimal values for those factors are extremely important to achieve good estimation quality. In many cases, these parameters can be chosen using cross-validation techniques, but this procedure is time consuming. In the Bayesian interpretation of NNs, the optimization of inter-neural weights corresponds to the increase of probability:

$$P(\mathbf{w}|\mathbf{D},\alpha,\beta,A)=\frac{P(\mathbf{D}|\mathbf{w},\beta,A)P(\mathbf{w}|\alpha,A)}{P(\mathbf{D}|\alpha,\beta,A)} \tag{29}$$

where **w**—weight coefficient vector, **D**—training data, A—structure of the neural network, $P(\mathbf{D}|\alpha,\beta,A)$—normalization element, $P(\mathbf{w}|\alpha,A)$—describes the information on the weights' values before introducing the training data, $P(\mathbf{D}|\mathbf{w},\beta,A)$—probability of obtaining the established response of the NN for suitable inputs, depending on parameters of the network.

Under the assumption that noise in the input data (measurements) used in the process of NN training is a Gaussian and the probability of weight distribution is also a Gaussian, suitable elements in Eq. (29) are described by the following formulas:

$$P(\mathbf{D}|\mathbf{w},\beta,A)=\frac{1}{Z_D(\beta)}\exp(-\beta E_D) \tag{30}$$

and

$$P(\mathbf{w}|\alpha,A)=\frac{1}{Z_W(\alpha)}\exp(-\alpha E_W), \tag{31}$$

where

$$Z_D(\beta)=\left(\frac{\pi}{\beta}\right)^{\frac{M}{2}} \text{ and } Z_W(\alpha)=\left(\frac{\pi}{\alpha}\right)^{\frac{W}{2}}, \qquad (32)-(33)$$

thus, we obtain:

$$P(\mathbf{w}|\mathbf{D},\alpha,\beta,A)=\frac{\frac{1}{Z_D(\beta)}\frac{1}{Z_R(\alpha)}\exp(-(\alpha E_W+\beta E_D))}{P(\mathbf{D}|\alpha,\beta,A)} \tag{34}$$

For the optimization of α and β parameters in the objective function, the following equation is taken into account:

$$P(\alpha,\beta|\mathbf{D},A)=\frac{P(\mathbf{D}|\alpha,\beta,A)P(\alpha,\beta|A)}{P(\mathbf{D}|A)}. \tag{35}$$

Under the assumption that distribution of regularization coefficients α and β is uniform, maximal values of the probability $P(\alpha,\beta|\mathbf{D},A)$ are obtained for the biggest values of the element $P(\mathbf{D}|\alpha,\beta,A)$. Probability $P(\mathbf{D}|A)$ is independent of the required parameters. After suitable transformations [20, 21], equations describing α and β parameters for the minimum of the objective function are obtained:

$$\alpha=\frac{\gamma}{2E_W(\mathbf{w}_{\mathrm{MP}})} \tag{36}$$

and

$$\beta=\frac{M-\gamma}{2E_D(\mathbf{w}_{\mathrm{MP}})}, \tag{37}$$

where

$$\gamma=W-2\alpha\,\mathrm{trace}(\mathbf{H})^{-1}. \tag{38}$$

and $\mathbf{w}_{\mathrm{MP}}$—minimum point of the objective function, **H**—hessian matrix of the cost function.

The parameter γ means an effective number of parameters of the NN; however, W is a number of all parameters in the NN.

4.2 Optimal Brain Damage method

The neural networks training leads to the minimization of the cost function defined as a mean square error between estimated and real value.

The cost function, for p-elements learning vector, is described in the following way:

$$E=\frac{1}{2}\sum_{j=1}^{p}\sum_{i=1}^{M}(d_i(j)-y_i(j))^2. \tag{39}$$

Differentiability and continuity of the cost function (39) make possible to use the gradient methods for its minimization. The first step in this method is an expansion of the cost function into Taylor series around the actual solution:

$$\Delta E=\sum_i g_i\Delta w_i+\frac{1}{2}\left[\sum_i h_{ii}[\Delta w_{ii}]^2+\sum_{i\neq j}h_{ij}\Delta w_i\Delta w_j\right]+\mathrm{O}\left(\|\Delta w\|^3\right) \tag{40}$$

where Δw_i-changes of i-th weight;

$$g_i=\frac{\partial E}{\partial w_i}; \tag{41}$$

$$h_{ij}=\frac{\partial^2 E}{\partial w_i\partial w_j}. \tag{42}$$

In the OBD algorithm, the weight coefficients are eliminated after full training of the net, so we can assume that elements related to the gradient are equal zero and skip them in Eq. (40). The hessian matrix is diagonally dominant, which makes it possible to include only diagonal elements h_{ii} of this matrix in the presented algorithm. The quadratic approximation assumes that the cost function is quadratic, so the third element in the Eq. (40) can be neglected. Following the above assumptions, the saliency coefficient is described by the following relation [15]:

$$S_i=\Delta E=\frac{1}{2}\sum_i h_{ii}[\Delta w_{ii}]^2 \tag{43}$$

These coefficients give the information about influence of the respective connections in NN on the training process. The weights with the smallest saliency parameter are eliminated. The algorithm of OBD method is thus presented as follows:

1. Choice of reasonable topology of neural network.
2. Full training of the net.
3. Computing diagonal elements of the h_{ii}.
4. Evaluation of the *saliency* parameters S_i for every weights coefficients.
5. Deleting the elements with the smallest saliency.
6. If weights connections were deleted, go back to the second point with reduced topology of neural network.

5 Simulation results

The NN estimators are tested in the control structure presented in Fig. 2. The main parameters of the drive system are as follows: $T_1 = T_2 = 203$ ms and $T_c = 2.6$ ms. The assumed values of resonant frequency and the damping factor of the speed closed loop of the drive system are, respectively: $\omega_o = 45\ \text{s}^{-1}$ and $\xi_r = 0.7$. For disturbance reduction, which are caused by high dynamics of inputs signals and measurement noise, the low-pass filters are used with time constant $T = 5$ ms.

The first results are presented for NNs trained with the Levenberg–Marquardt algorithm, without any additional techniques. The neural estimators are tested first in the open control loop, which means that control structure of the two-mass system is based on state variables obtained directly from the drive mathematical model, and signals estimated by the designed NNs are not used in this structure. The obtained results are shown in Fig. 3.

In order to evaluate the quality of estimation of the load machine speed ω_{2e} and shaft torque m_{se}, the estimation errors of NNs are calculated, using the following formula:

$$\mathrm{Err} = \frac{\sum_{i=1}^{n} |x_i - \hat{x}_i|}{N} \times 100 \tag{44}$$

where x_i—real value, $\hat{x}_i$—estimated value, N—number of samples.

The estimation errors (average error per sample) calculated for transients presented in Fig. 3, are, respectively, 5.77 for the load speed and 0.63 for the shaft torque.

Next, the estimated signals were introduced into the control structure and obtained results are demonstrated in Fig. 4. As can be seen from those transients, neural estimators prepared using the Leveneberg–Marquardt algorithm and next tested in the closed control loop failed.

The NNs are not considered during the designing process of the control structure. The coefficient values (20–23) in the suitable feedback loops are calculated with the assumption that we have the exact knowledge of feedback signals, so in the case of estimated variables, these "ideal" coefficients intensify dynamical estimation errors and thus

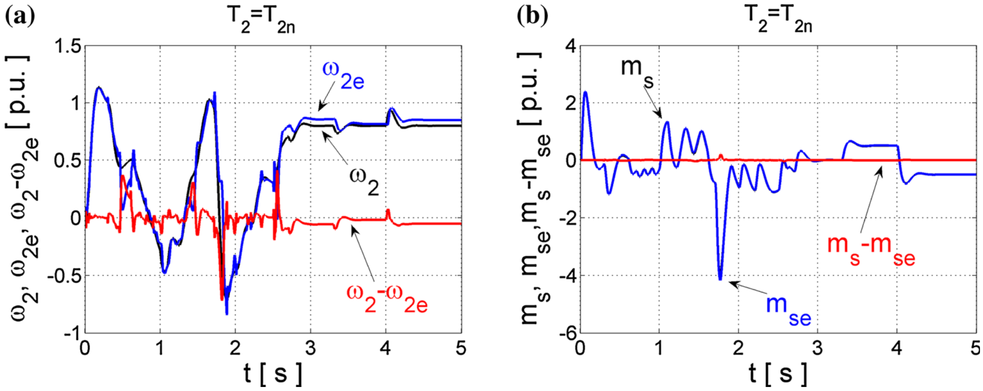


Fig. 3 Transients of the real and estimated load speed ω_2 (**a**) and torsional torque m_s (**b**) and their estimation errors obtained for NN trained with LM algorithm and tested in an open loop

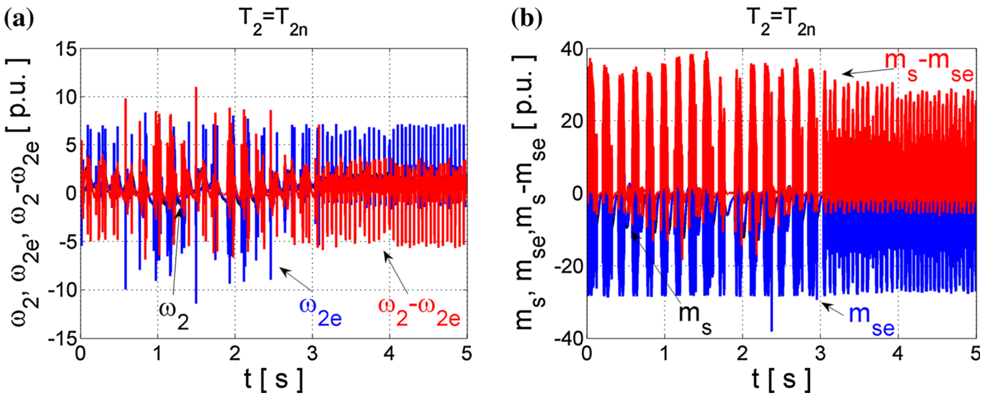
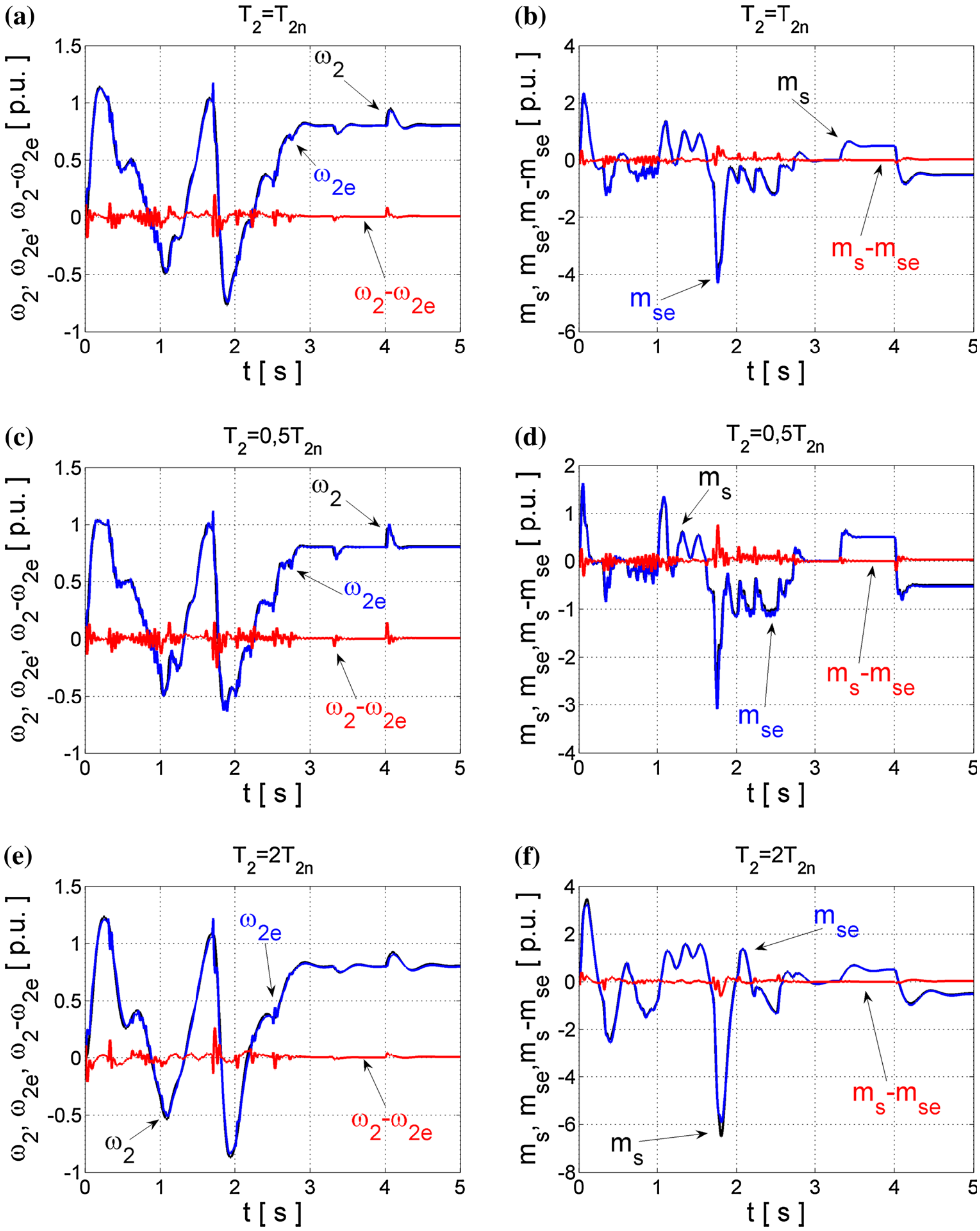


Fig. 4 Transients of the real and estimated load speed ω_2 (**a**) and torsional torque m_s (**b**), and their estimation errors obtained from NNs trained with LM algorithm and tested in the closed-loop system

Fig. 5 Transients of the real and estimated load speed ω_2 (**a**, **c**, **e**) and torsional torque m_s (**b**, **d**, **f**) and their estimation errors obtained from NN trained with LM algorithm and the Bayesian regularization, and tested in the closed loop for different values of T_2 time constant



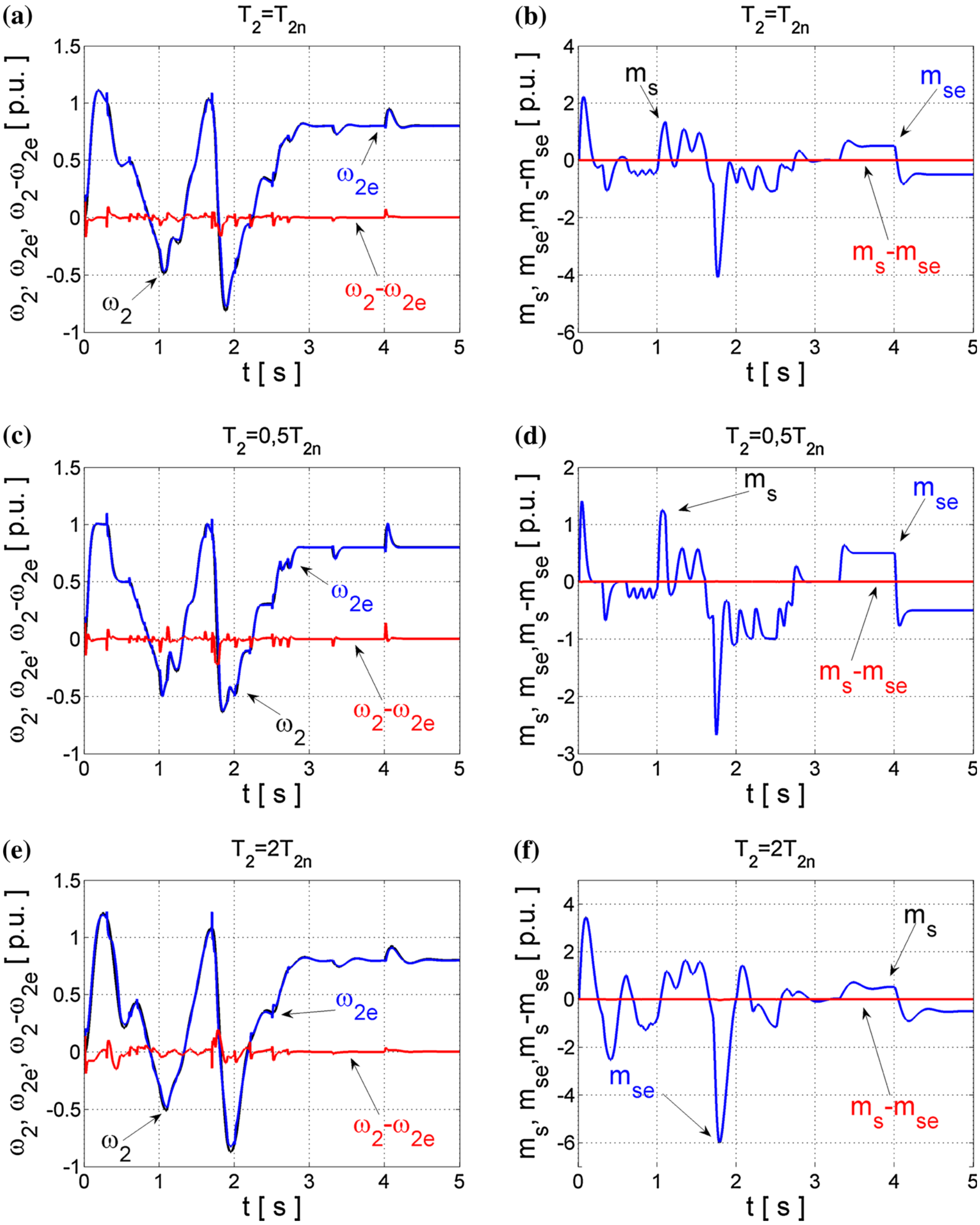
quite significant interferences appear. Oscillations of state variables are excited in the closed-loop structure, so the proper control is impossible. These phenomena may cause damages of coupling elements between the motor and load machine. So the high quality of state variables estimation is necessary for the correct operation of the closed-loop drive system. Thus, the optimization methods for NNs are introduced.

First, the application of Bayesian regularization in neural estimators is tested, and operation of the obtained NNs implemented in the closed-loop system is presented in Fig. 5, also in the case of changeable load side time constant T_2.

The obtained results are very good, and the usage of the modified cost function (26–28) during the training procedure of NN can eliminate too big weight coefficients of the designed neural estimators and thus prevent oscillations appearing previously in the closed-loop operation. The correct operation of the designed estimators in the closed control loop can be assured even for changeable values of the load mechanism time constant T_2 (Fig. 5c–f).

However, the best quality of the state estimation is achieved for NN structures optimized with the OBD method. The obtained results of closed-loop operation are demonstrated in Fig. 6.

Fig. 6 Transients of the real and estimated load speed ω_2 (**a**, **c**, **e**) and torsional torque m_s (**b**, **d**, **f**) and their estimation errors obtained from NNs trained error with LM algorithm and the OBD method, tested in closed loop for different values of T_2 time constant



Neural estimators optimized with this method can calculate precisely the suitable state variables of the two-mass drive system even in the presence of T_2 changes. It is important that changes of the time constant of the load machine are not taken into account during the training process in both tested cases. The neural estimators have the initial structure containing 215 synaptic coefficients. After OBD method applied for the shaft torque estimator, 80 synaptic connections are deleted and for the load speed estimator, 140 connections are eliminated, respectively. The decision about stopping the optimization process is based on the analysis of the estimation error. The examples of the Hinton diagrams for the load speed estimators are presented in Fig. 7. The Hinton diagrams visualize matrices of bias and weights values. Each value is represented by a rectangle, which size is associated with the weight magnitude, and each color indicates the sign (a positive—red, a negative—green). The OBD algorithm eliminates individual inter-neural connections; however, there are neurons completely eliminated after this optimization process (Fig. 7d, e).

In the Table 1, the comparison of the estimation errors, calculated according to (44) for both tested methods, is shown. Both described methods enable the preparation of neural estimators which give the correct results after implementation in the closed-loop structure of the two-mass drive system.

The OBD method can give better results, but for this method, higher computational power is required and the

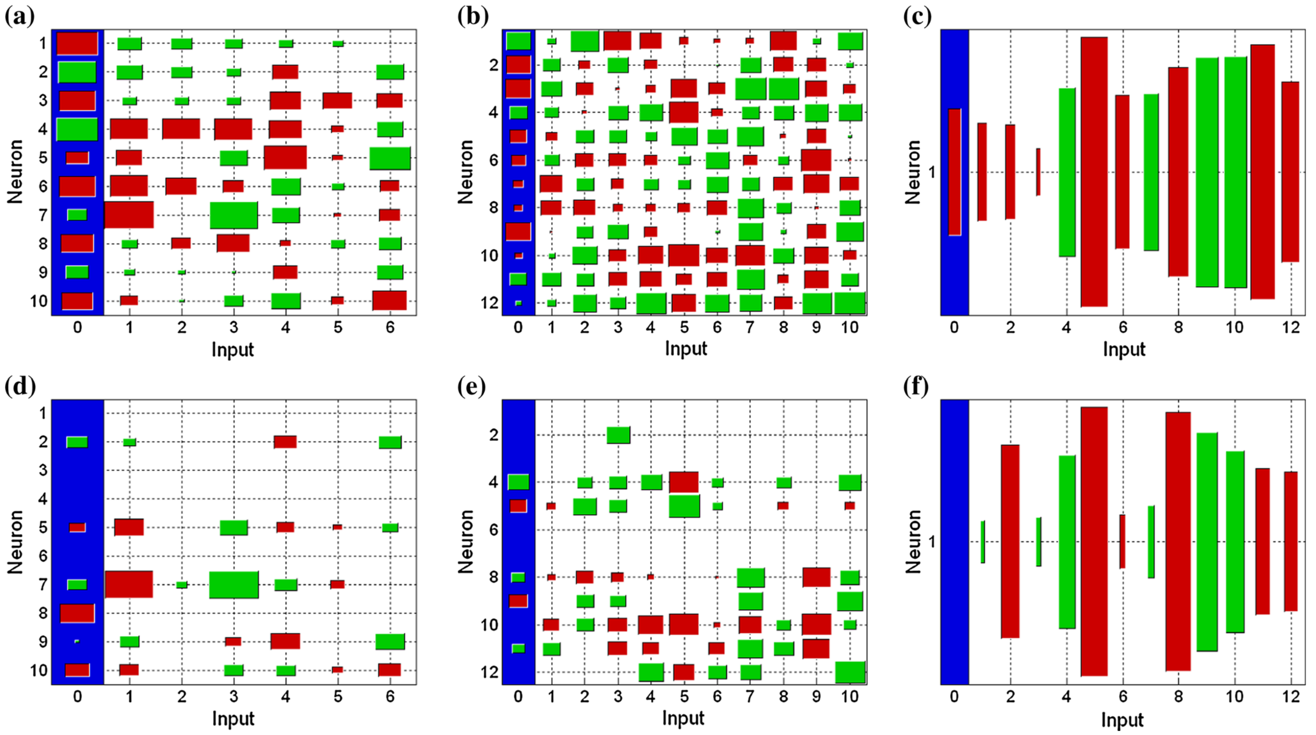


Fig. 7 Hinton diagrams for the load speed estimator illustrating weight values between input and first hidden layer (**a**, **d**), between two hidden layers (**b**, **e**) and between second hidden and output layers (**c**, **f**) before OBD method (**a**–**c**) and after weight elimination (**d**–**f**)

Table 1 Errors for neural estimators tested in the closed loop for changes of the T_2 time constant

Method	T_2 time constant		
	$T_2 = T_{2N}$	$T_2 = 0.5T_{2N}$	$T_2 = 2T_{2N}$
Estimator m_s			
Bayesian regularization	4.60	5.05	5.05
OBD	0.05	0.05	0.11
Estimator ω_2			
Bayesian regularization	2.16	2.44	2.19
OBD	1.28	1.28	2.12

training process is much slower than for the Bayesian regularization method. The pruning methods are also important when the hardware implementation of the neural estimators is considered. It is possible to reduce the structure of the NN which leads to the simplification of the realization algorithm and in result to save the hardware resources (e.g., in the case of FPGA implementation).

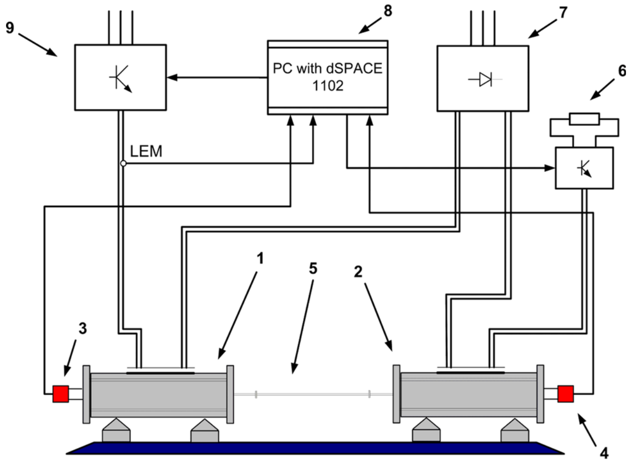


Fig. 8 Schematic diagram of the experimental setup, where *1*—motor machine, *2*—load machine, *3*,*4*—encoders, *5*—shaft, *6*—resistor, *7*—rectifier, *8*—control structure, *9*—power converter

6 Experimental results

The tested drive system with an elastic joint is emulated with two DC machines (0.5 kW each) connected by an elastic shaft (a steel shaft of 5 mm diameter and 600 mm length). The stiffness of the connection depends on the shaft diameter. The motor is fed by a power converter. The control algorithm and neural state estimators are implemented in DSP placed in the dSPACE 1102 card. The load machine in the drive system is also controlled using the DSP (see Fig. 8). Basic parameters of the drive system are presented in the Table 2.

Table 2 Parameters of the two-mass system

Parameter	Value	Unit
Power	500	W
Nominal motor voltage	220	V
Nominal speed	1,450	rev/min
Motor mechanical time constant	0.203	s
Load mechanical time constant	0.203	s
Shaft length	600	mm
Shaft diameter	6	mm
Stiffness time constant	0.0026	s

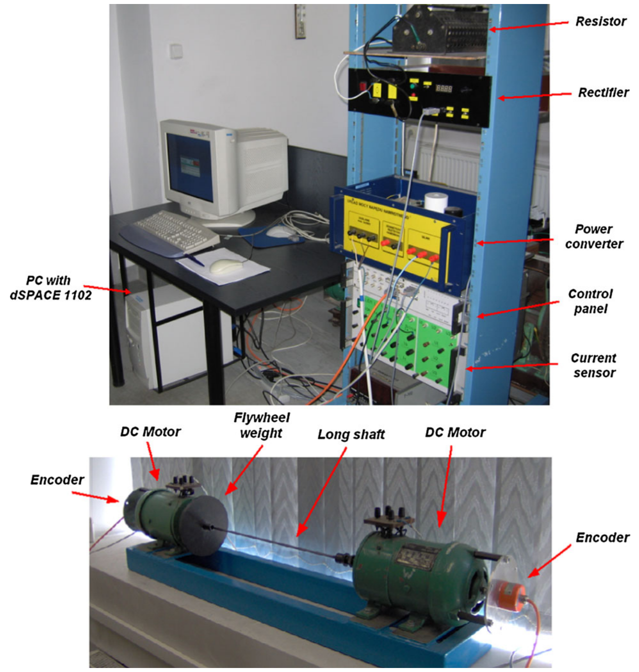


Fig. 9 Laboratory test bench

The speeds of both DC machines are measured by incremental encoders (36 000 pulses per rotation); however, the measurement of the loading machine is used only for comparison with the estimated value. In the laboratory setup the LEM sensors for current measurements are implemented. There is no shaft torque sensor in the laboratory setup. Therefore, in order to check the estimated shaft torque shape, the Kalman filter is applied [4, 22]. In Fig. 9, pictures of the laboratory test bench are presented.

Exemplary transients of the state estimation in the closed-loop drive structure, obtained before and after NNs optimization, are presented in the Fig. 10.

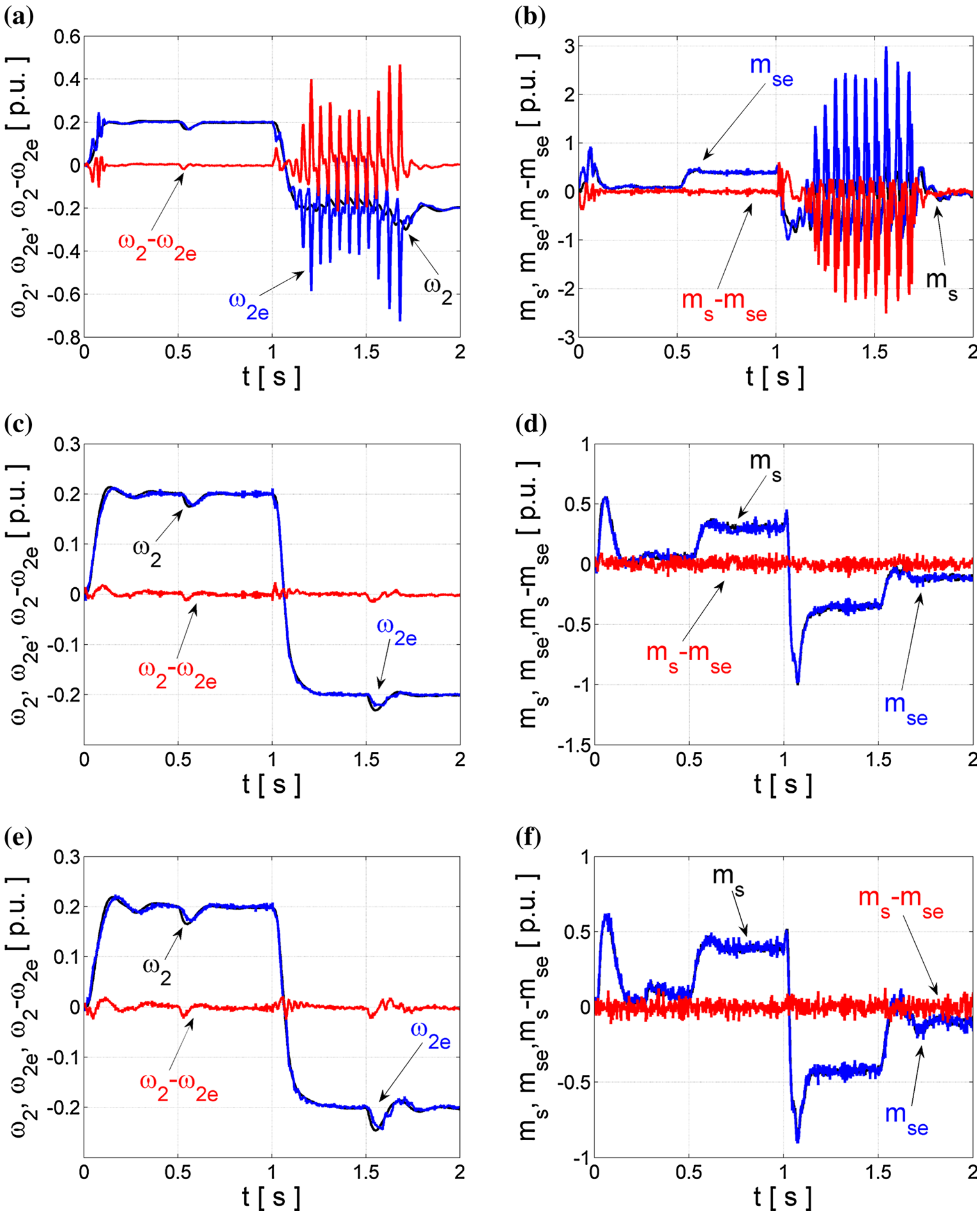
The tests are realized for the reference speed that equals 20 % of its nominal value; after one second, the reverse operation of the drive system is forced. In the period $t \in (0.5–1.5)$s, the nominal load torque is applied. Next, this load is taken off. Similarly to the simulation, neural estimators trained with Levenberg–Marquardt algorithm are generating noises and instability of the closed-loop drive system operation (Fig. 10a, b). After optimization with the described algorithms, both estimators work properly. The estimation errors (44) for both optimization algorithms are similar in the case of experimental tests: for the NN optimized with Bayesian regularization, the load speed error is 0.37 and the torsional torque error is 2.77. The inaccuracy of the load speed and torsional torque reconstruction for neural estimator designed with the OBD method are, respectively, 0.49 and 2.99. Additional tests for twice bigger T_2 time constant were conducted. The results are presented in the Fig. 11.

As in simulation tests, the changes of the two-mass system parameter are not taken into account during neural estimator training, also coefficients in the control structure are calculated for nominal value of T_2. Estimation error of the load speed is equal 0.67 in the case of the Bayesian regularization method and 0.83 for the OBD method. Torsional torque reconstruction using neural estimators optimized with Bayesian regularization method presents error equal to 2.76 and after implementation of the OBD method is 2.95. Comparing changes of the errors for different values of T_2 in the drive system similar values can be observed. The conclusion is that obtained neural estimators are robust against changes of the tested drive parameter.

7 Conclusion

Application of neural estimators in the drive system with elastic coupling enables very good estimation quality. Neural estimators do not require the knowledge of mathematical model parameters on the contrary to the algorithmic methods of state variable estimation. Disturbances from the estimated signals connected as additional feedbacks in the two-mass drive control structure can lead to the speed oscillation or even problems with system instability. So the good quality of the estimation is very important. After the implementation of Bayesian regularization or OBD, the obtained precision of calculations in NN is much better. Presented estimators are also robust to changes of the mechanical parameters of the drive, like the load side time constant T_2. The OBD method can give slightly better results, but it should be noticed that for this method higher computational power is required and the training process is much slower than for the Bayesian regularization method. Thus, this last method can be recommended for practical implementation. Correct work of the designed estimators was confirmed not only by

Fig. 10 Experimental transients (closed-loop operation for nominal T_2 value) of the real and estimated load speed ω_2 (**a**, **c**, **e**) and torsional torque m_s (**b**, **d**, **f**) and their estimation errors obtained for NN trained with LM algorithm (**a**, **b**), and after the Bayesian regularization (**c**, **d**), and the OBD method (**e**, **f**)



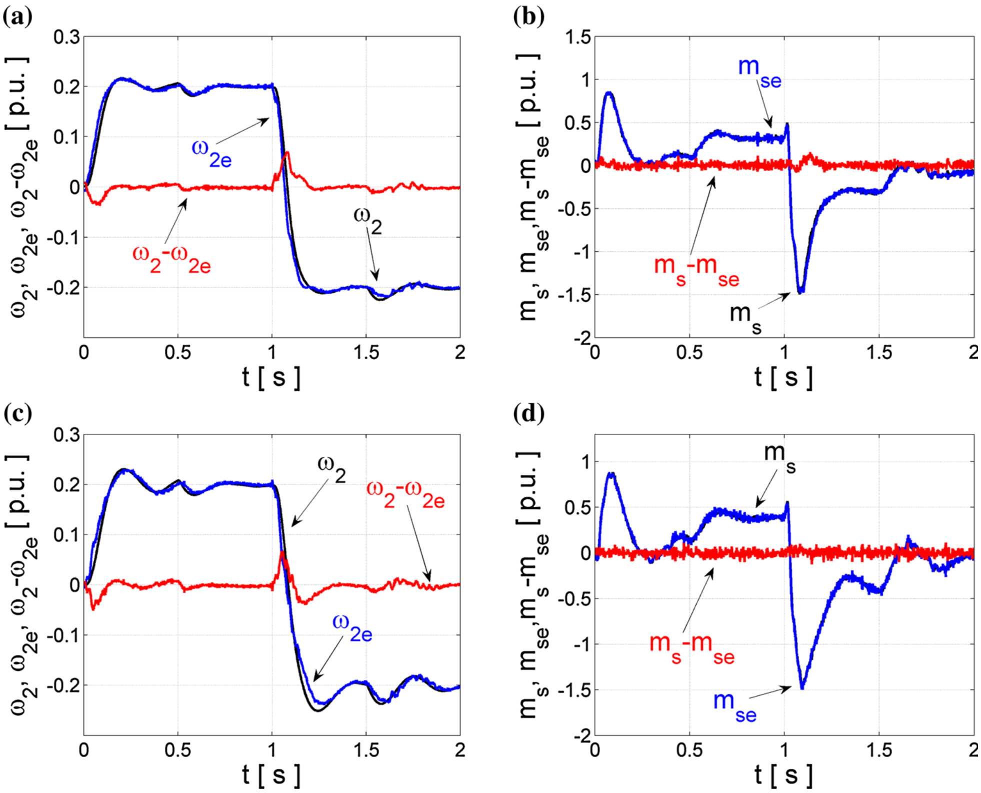


Fig. 11 Experimental transients (closed-loop operation for $T_2 = 2T_{2N}$) of the real and estimated load speed ω_2 (**a**, **c**) and torsional torque m_s (**b**, **d**) and their estimation errors obtained for NN after optimization with the Bayesian regularization (**a**, **b**), and the OBD method (**c**, **d**)

simulations but also by experiments in the real drive system in the laboratory.

Acknowledgments This research work was supported in part by the National Science Centre (Poland) under Grant 2011/01/B/ST7/04632.

References

1. Zhang G, Furusho J (2000) Speed control of two-inertia system by PI/PID control. IEEE Trans Ind Electron 47(3):603–609
2. Szabat K, Orlowska-Kowalska T (2007) Vibration suppression in two-mass drive system using PI speed controller and additional feedbacks—comparative study. IEEE Trans Ind Electron 54(2): 1193–1206
3. Vukosovic SN, Stojic MR (1998) Suppression of torsional oscillations in a high-performance speed servo drive. IEEE Trans Ind Electron 45(1):108–117
4. Szabat K, Orlowska-Kowalska T (2006) Analysis of the algorithmic methods of state variables estimation for drive system with elastic joint. In: Nawrowski R (ed) Computer applications in electrical engineering. Poli-Graf-Jak, Poznań, pp 99–115
5. Akin B, Orguner U, Ersak A, Ehsani M (2006) Simple derivative-free nonlinear state observer for sensorless AC drives. IEEE ASME Trans Mechatron 11(5):634–643
6. Baraki S, Suryanarayanan S, Tomizuka M (2005) Design of Luenberger state observers using fixed-structure H∞ optimization and its application to fault detection in lane-keeping control of automated vehicles. IEEE ASME Trans Mechatron 10(1):34–42
7. Bishop CM (1996) Neural networks for pattern recognition, 1st edn. Oxford University Press, Oxford
8. Orlowska-Kowalska T, Szabat K (2007) Neural-network application for mechanical variables estimation of a two-mass drive system. IEEE Trans Ind Electron 54(3):1352–1364
9. Nho HC, Meckl P (2003) Intelligent feedforward control and payload estimation for a two-link robotic manipulator. IEEE ASME Trans Mechatron 8(2):277–282
10. Neuneier R, Zimmermann HG (1998) How to train neural networks. In: Orr GB, Müller KR (eds) Neural networks: tricks of the trade. Springer, Berlin, pp 373–423
11. Girosi F, Jones M, Poggio T (1995) Regularization theory and neural networks architectures. Neural Comput 7(2):219–269
12. Fahlman SE, Lebiere C (1990) The cascade-correlation learning architecture. In: Touretzky DS (ed) Advances in neural information processing systems 2. Morgan-Kaufmann, Los Altos, CA
13. Hassibi B, Stork DG (1993) Second order derivatives for network pruning: optimal brain surgeon. In: Hanson SJ, Cowan JD, Giles CL (eds) Advances in neural information processing systems, vol 5. Morgan Kaufmann, San Jose, pp 164–172
14. Orlowska-Kowalska T, Kaminski M, Szabat K (2007) Optimization of the neural state variable estimators for the two-mass drive systems. In: Proceedings of 16th EDPE conference, Slovak Republic (on CD)
15. Le Cun Y, Denker JS, Solla SA (1990) Optimal brain damage. AT&T Bell Laboratories, Holmdel, NJ
16. Orlowska-Kowalska T, Kaminski M (2009) Application of the OBD method for optimization of neural state variable estimators of the two-mass drive system. Neurocomputing 72(13-15):3034–3045
17. Hassibi B, Stork DG, Wolff GJ (1992) Optimal brain surgeon and general network pruning. CRC-TR-9235, RICOH California Research Center, Menlo Park, CA

18. Orlowska-Kowalska T, Kaminski M (2009) Effectiveness of sensitivity-based methods in optimization of neural state estimators of the drive system with elastic coupling. IEEE Trans Ind Electron 56(10):4043–4051
19. Hancock PJB (1992) Pruning neural nets by genetic algorithm. In: Proceedings of international conference on artificial neural networks, Brighton, UK, pp 991–994
20. MacKay DJC (1992) A practical Bayesian framework for back-propagation networks. Neural Comput 4(3):448–472
21. Foresee FD, Hagan MT (1993) Gauss-Newton approximation to Bayesian learning. IEEE Int Conf Neural Netw 3:1930–1935
22. Szabat K, Orlowska-Kowalska T, Dyrcz K (2006) Extended Kalman filters in the control structure of two-mass drive system. Bull Pol Acad Sci Tech Sci 54(3):315–325