ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1994

Seria: MECHANIKA z. 116 Nr kol. 1231

Wojciech CHOLEWA
Katedra Podstaw Konstrukcji Maszyn
Politechnika Śląska

Maurice F. WHITE
Division of Marine Engineering
University of Trondheim

ZASTOSOWANIE SIECI NEURONALNYCH DO ODWRACANIA MODELI DRGAŃ

Streszczenie. Celem pracy jest wskazanie przydatności modeli odwrotnych w badaniach drgań (np. do identyfikacji rozkładu niewyrównoważenia) i określenie sposobu odwracania znanych, istniejących modeli. Zamieszczono prosty przykład obliczeniowy, pokazujący, jak można określać rozkład niewyrównoważenia wzdłuż wirnika na podstawie amplitud drgań czopów w wybranych dwóch płaszczyznach, obserwowanych dla różnych prędkości wirowania.

APPLICATION OF NEURAL NETWORKS FOR INVERTING OF VIBRATION MODELS

Summary. The aim of this paper is to point out the usefulness of inverse models in rotordynamics (e.g. for identification of unbalance distribution) and to show how to invert known, existing models. The overall methodology of the approach is presented, along with a simple numerical example which shows how the distribution of unbalance may be identified from knowledge of the response in only two planes at different rotating speeds.

ПРИМЕНЕНИЕ НЕЙРОННЫХ СЕТЕЙ ДЛЯ ОТВЕРНУТИЯ МОДЕЛЕЙ КОЛЕБАНИЙ

Резюме. Цель работы - это показание пригодности оборотных моделей в исследовании колебаний (н.п. для идентификаций росположения неустройчивости ровновесия) и определения способа отвернутия известных, существующих моделей. Помещено прямой исчесленный пример, который показывает как можно определять расположение неустройчивости ровновесия вдлину ротора на базе амплитуд перемещения цапфей в избранных двух плоскостях, измеренных для разных скоростей поворачивания ротора.

1. INTRODUCTION

There exist a great number of interesting papers and books (e.g. [3], [13], [5]) which discuss how the vibration phenomena are caused by the design and technical state of machinery. It should be underlined that conclusions are quite often contrary to our intuition. It means that special tools, such as mathematical modelling have to be used to predict the shape and values of vibration. Today the prediction of vibration done by means of the finite or boundary element models with distributed masses of shafts, inertia of impellers, stiffness, internal and external damping as well as dynamic excitation from the fluid, is quite representative when compared with results of experimental measurements. This paper contains selected parts of [2].

Mathematical models in rotordynamics contain non homogeneous systems of differential equations with parameters depending on features of rotating machinery. Due to non linearity and complex form of equations the general analytical solutions are unknown. Most often it is possible only to solve them in a numerical way and calculate the values of selected vibration estimates for the given set of parameters. The models make known cause-effect links existing in modelled objects and set relations between the following groups of variables:

- independent variables representing features that are easy to define and to measure (e.g. operating conditions, general design features, ...),
- independent variables representing features that are very difficult (or even impossible) to measure (e.g. distribution of imbalance masses along the rotor, damping of rotor-bearing system, ...),
- dependent variables (results of calculations) representing features that are quite easy to measure (e.g. vibration estimates for bearings, ...),
- dependent variables (results of calculations) representing features that are no so easy to measure (dynamic forces acting on bearings and supporting structure, vibration estimates for planes away from the bearing, ...).

It seems reasonable to look for another way for modelling, where we can start from the data that are easy to measure, to obtain as a result the data that are not so easy to measure. By means of such models it will be possible for example to predict the distribution of imbalance on the rotor based on the results of measurements. From the physical point of view a solution of the problem is possible, because the distribution of unbalance determines the degree to which particular bending modes of vibration are excited (for a flexible shaft). There are two possibilities to solve the problem:

- prepare new models from scratch,
- use the existing models and try to invert them.

We assume that the second possibility is better, because the existing models contain knowledge collected over a long time. Moreover they are carefully validated with respect to results of experimental test on machinery.

2. GENERAL PROCEDURE

2.1. Formal Description

We consider an existing (and given as an algorithm or program) mathematical model M by means of which a multidimensional metric space $\{MI\}$ of values of input parameters mi_i is mapped into a multidimensional metric space $\{MO\}$ of values of output parameters mo_j. Elements of the spaces will be written as matrices (so called vectors) $MI = [mi_i]$ and $MO = [mo_j]$. Not all parameters are equally important for our task. Let us consider the projections $\{MIA\}$, $\{MIB\}$, $\{MIC\}$, $\{MID\}$ and $\{MOA\}$, $\{MOB\}$ of the spaces, such that

$$\begin{aligned} \{MI\} &= \{MIA\} \times \{MIB\} \times \{MIC\} \times \{MID\} \\ \{MO\} &= \{MOA\} \times \{MOB\} \end{aligned} \tag{1}$$

where the inputs to the model M

- $\{MIA\}$ contains unknown parameters, which should be estimated by means of an inverted model (e.g. distribution of unbalance),
- $\{MIB\}$ contains known parameters (e.g. operating conditions),
- $\{MIC\}$ contains parameters, that may be considered as constant (e.g. design of machinery, stiffness, damping),
- $\{MID\}$ contains parameters difficult to measure, that we have to leave out (e.g. misalignment),

and the outputs from the model M

- $\{MOA\}$ contains known parameters (e.g. radial vibrations of rotor in normal planes at two given positions along the shaft),
- $\{MOB\}$ contains unimportant (presently) and unknown parameters (e.g. radial vibration of rotor in other planes).

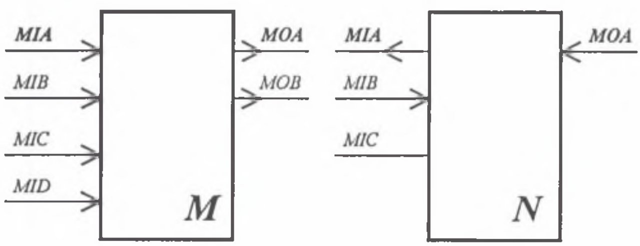

Fig. 1. Models: M - direct, N - inverted
Rys. 1. Modele: M - bezpośredni (odwracany), N - odwrócony

We formulate the basic task as follows (see Fig. 1):
Given the mapping M

$$\{MIA\} \times \{MIB\} \times \{MIC\} \times \{MID\} \xrightarrow{M:} \{MOA\} \times \{MOB\} \tag{2}$$

For a fixed matrix *MIC* find the mapping *N* such that

$$\{MIB\} \times \{MOA\} \xrightarrow[MIC=const]{N:} \{MIA\} \tag{3}$$

We do not assume that the matrix *MIC* is known. We assume only that it is constant. The remaining subspaces {*MID*} and {*MOB*} are not taken into account (in the mapping *N*).

Of course it should be clear that the mapping *N* does not exist in a general case. We are not able to expect that leaving some parameters out (subspace {*MID*}) it will be possible to find the mapping *N*. Ignoring a set of parameters results in randomness of the inverted model. The level of this randomness is strongly task-dependent. Even if we take into account all parameters, it will be (strongly) possible that the mapping *M* cannot be inverted. Such a situation can occur when for the mapping *M* there exist identical matrices *MO* for a few particular, different matrices *MI*, i.e. when mapping *M* converts different input patterns into the same output pattern (see Fig. 2).

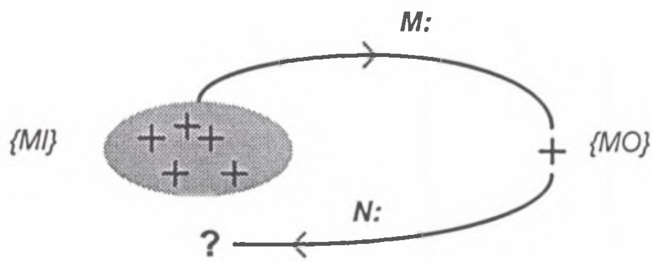

Fig. 2. It may be possible that the mapping *M* cannot be inverted
Rys. 2. Możliwe są sytuacje, w których odwzorowanie *M* nie będzie odwracalne

What to do? We should simplify the task. In general there are two possibilities: apply a classifier and/or fuzzify the data.

2.2. Fuzzy Classifier

The notion of fuzzy sets was introduced by [15]. Fuzzy set $\bar{A}$ in a space U is the set of pairs $\langle u, w_A(u) \rangle$

$$\bar{A} = \{\langle u, w_A(u) \rangle : u \in U\} \tag{4}$$

where the membership function $w_A(u)$ for the element u from the space U takes its values from the range [0,1] of real numbers. This function estimates a degree in which the given element u belongs to the set $\bar{A}$. The more closely this value is to 1.0 than more strongly the element belongs to the set.

To fuzzify the data in $\{MIA\}$ we can define transformations for particular parameters converting their real values to a small number of linguistic values - like *small, large*. The similar treatment can be done not for single parameters but for the selected subset of parameters in $\{MIA\}$. In the space spanned on these parameters we select some regions called classes and define a family of such classes as a family of fuzzy sets, given by their membership functions $\{w_1, w_2, ..., w_c\}$. For our application it is strongly recommended that the classes are defined directly and we are able to interpret what it means that parameters belong to a class. In an opposite case when the classes result from clustering of data such interpretation may be very difficult. The values of membership functions (without the direct knowledge of the parameters *MIA* used when defining the values of the functions) can be used to partition the space $\{MIA\}$ by means of the following rule

$$\text{if} \quad w_i(x) > w_j(x) \quad \text{for all} \quad j = 1,...,i-1, i+1,...,c \quad \text{then} \quad x \in class_i \tag{5}$$

To fuzzify the data expected from the discussed model N we introduce in equation (3) the mapping C

$$\{MIA\} \xrightarrow{C:} \{w_i\} \tag{6}$$

converting the real parameters to the values of membership functions.

2.3. Neural Networks

The general method to look for the mapping N is to consider the black box which is trainable (see Fig. 3). Neural networks are an appropriate tool to solve such tasks. We can generate a lot of examples by means of the model M and train the network (i.e. model N) on the examples. This approach seems to be attractive, since due to existing software the required knowledge of the theory of neural networks and training strategies is minimal.

The theory of neural networks is huge. It is not the aim of this paper to make an overview of major architecture and theoretical concepts (for an overview see e.g. [11], [7], [12], [4], [8], [6], [9]). One (basic) kind of neural network, which seems to be quite appropriate for our application is presented below. It is a simple three-layer feed forward network with back propagation of errors (Fig. 8).

Neural networks consist of linked processing units called nodes or neurones, where interconnections of nodes may be in general variable. The selected network has a single layer of hidden nodes sandwiched between the input layer and output layer of nodes, where the inputs to nodes in hidden and output layer come exclusively from respectively input and hidden layer. The input layer contains virtual nodes only (they do not do any processing). The network propagates the input data through the layers to the output layer. Each j-th node in the hidden or output layer l transforms outputs $x_{l-1,i}$ from nodes in previous $(l$-$1)$ layer, to its output $x_{l,j}$

$$x_{l,j} = f(\sum_{i=0}^{n-1} w_{l,j,i}\, x_{l-1,i} + w_{l,j,n}) \qquad (7)$$

where $w_{l,j,i}$ are weights, $w_{l,j,n}$ represents a bias and where the activation function f is any differentiable, monothonic function - most often the sigmoid function (called logistic function)

$$f(z) = \frac{1}{1+e^{-z}} \qquad (8)$$

The useful property of the sigmoid function (8) is that the derivative of the function can be calculated directly from the value of this function

$$\frac{\partial f}{\partial z} = f(z)(1 - f(z)) \qquad (9)$$

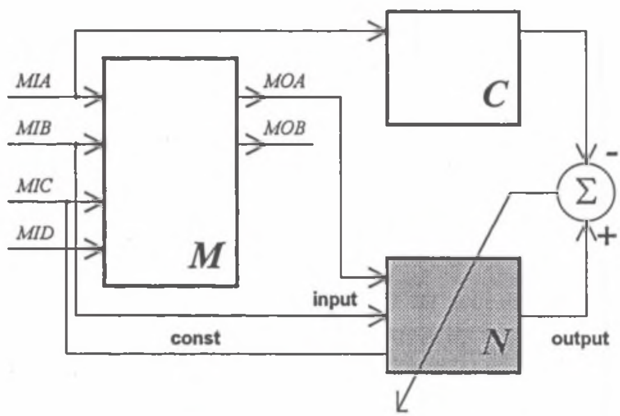

Fig. 3. Model N as a black box during training
Rys. 3. Model N jako czarna skrzynka w fazie trenowania

The operating characteristic of the net is mainly defined by the topology of the network, by the weights $w_{l,j,i}$ and does not depend significantly on the shape of activation function f. Weights $w_{l,j,i}$ are calculated in the training (supervised learning) or learning (unsupervised learning) phase. During training the known set of samples in the form of pairs consisting of values of all inputs and expected values of all outputs is given. The outputs are next calculated by the network (i.e. inputs are feed forward), using current values of weights, where initially the values of weights are set randomly. From the comparison of calculated and expected values an error results for each output.

The global measure of errors (e.g. sum of their squares) should be minimised. Due to the shape of (7) and (8) it is possible to calculate the degree to which each input to a given node contributes in its output error. Finally it is possible to propagate errors back to previous layers and using appropriate minimising strategy it is possible to update iterativly the selected weights $w_{l,j,i}$. The calculation of errors and updating of weights is repeated until the network reaches an expected level of quality. Of course the convergence of the process is not guaranteed. A neural network is a highly non-linear system and assuring the stability of such a system is difficult. To obtain the convergence and stability during training and to guarantee jumping over local extremes it is necessary to take into account some special heuristic algorithm for updating of weights (see e.g. [9] or [10]).

As the result of training we have the set of weights. The result should be tested using an independent set of data, because due to the great number of weights and non-linearities in nodes it is possible to reach a state when the network perfectly maps the training data but produces significant errors for other data.

3. NUMERICAL EXAMPLE

The discussed methodology will be illustrated with an example. It will be shown how to invert a model by means of which we predict the vibration of the shaft as a result of unbalance into a model by means of which it will be possible to identify the unbalance based on the results of vibration measurements at locations near the bearings.

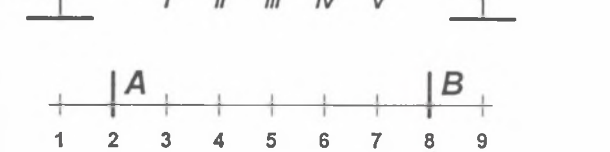

Fig. 4. Investigated object
Rys. 4. Schemat badanego obiektu

3.1. Investigated Object and Analysis Method

The investigated (artificial) object consisted of a rotating shaft with five discs (impellers) supported by two similar journal bearings. To simplify the example an axial symmetry of the supporting structures is assumed. It is assumed too, that the displacements of the shaft can be measured at two fixed points *A* and *B* (see Fig. 4). The shaft is considered as consisting of eight elements *a, b, ..., h*. Discrete unbalance may be located at the given radial and angular position on the selected discs *I, II, ..., V*.

The model *M* introduced in the equation (2) may be comprised of mathematical equations, algorithms (procedures), computer programs and so on. Vibration of the discussed rotor system was simulated with the program TURBO [14] which calculates the forced vibrations of a multiply supported rotor due to unbalance using a finite element approach. For the tests in question journal bearings were modelled using 8 linearized coefficients. This introduced speed dependent bearing stiffness $\underline{K}$ and damping $\underline{C}$ as may be expected in a practical situation.

3.2. Test-Simulation

To check the basic properties of the given object, the maximal amplitude of the shaft displacements measured in the point A (Fig. 4) were calculated for rotating speeds from 1000 to 9000 rpm and for five unbalance cases:

case U001: full unbalance on the disc *I*,
case U002: full unbalance on the disc *II*,
case U003: full unbalance on the disc *III*,
case U004: full unbalance on the disc *IV*,
case U005: full unbalance on the disc *V*.

Results of the calculations are shown in the Fig. 6. The relative values of the maximal amplitudes calculated for each rotating speed are compared in Fig. 5.

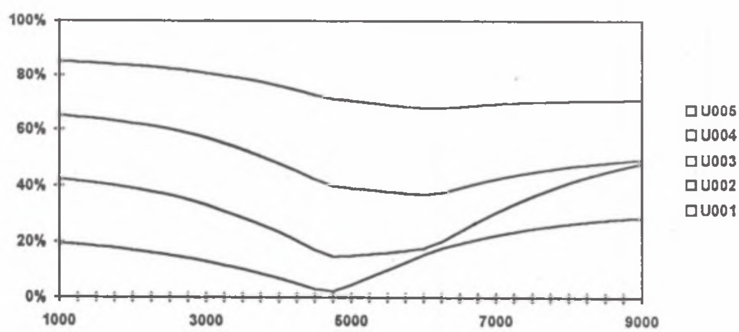

Fig. 5. Relative maximal displacement at the point *A* (Fig. 4) as a function of rotating speed
Rys. 5. Względna amplituda przemieszczeń w punkcie *A* (Rys. 4) jako funkcja prędkości wirowania

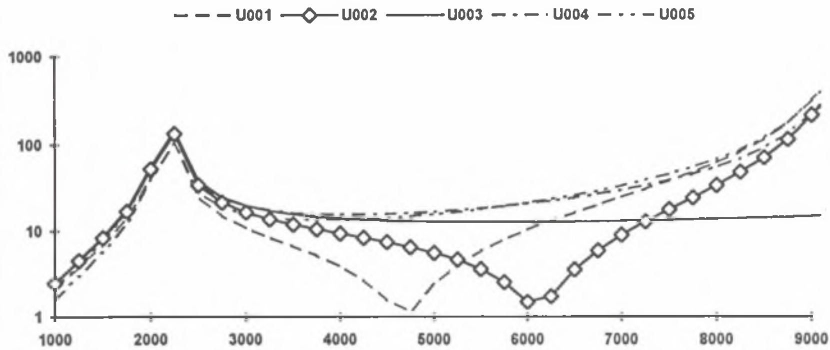

Fig. 6. Maximal displacement at the point A (Fig. 4) as a function of rotating speed
Rys. 6. Amplituda przemieszczeń w punkcie A (Rys. 4) jako funkcja prędkości wirowania

The similar calculations of relative values were done for the unbalance on the disc I (75 μ m) and on the disc III (25 μm) with the following lags of the angular location of unbalance on the discs:

case U009: $\alpha_{III}-\alpha_{I} = 0°$,
case U010: $\alpha_{III}-\alpha_{I} = 90°$,
case U011: $\alpha_{III}-\alpha_{I} = 180°$.

Results are shown in the Fig. 7.

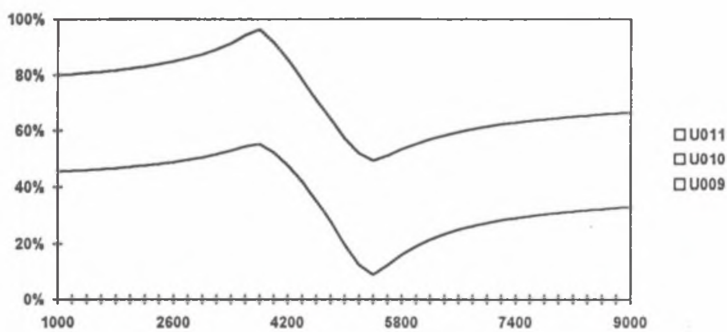

Fig. 7. Relative maximal displacement at the point A (Fig. 4) as a function of rotating speed
Rys. 7. Względna amplituda przemieszczeń w punkcie A (Rys. 4) jako funkcja prędkości wirowania

From the results of test-simulation it follows that the shape of the function

displacement (*rotating speed*)

depends on the location of unbalance along the shaft (see Fig. 5) and on the angular position of unbalance (see Fig. 7), i.e. it depends on the unbalance case. Of course if we generate a set of similar figures for other unbalance cases it should be very difficult (or even impossible) for an expert to formulate a set of heuristic rules by means of which it will be possible to identify the location of a main unbalance.

3.3. Input and Output Parameters

The investigated inverse model *N* should predict the unbalance of the rotor on the selected disc. The result is expected as a qualitative value. It was assumed that the output of the inverse model *N* should estimate the membership functions of fuzzy sets interpreted as follows (for i = I, II, ..., V)

$$\text{"Main part of unbalance is located on the disc } i \text{ ."} \tag{10}$$

It means that there will be 5 output parameters (estimates of values of membership functions) for the model *N*:

$$\hat{c}_{\mathrm{I}};\ \hat{c}_{\mathrm{II}};\ \hat{c}_{\mathrm{III}};\ \hat{c}_{\mathrm{IV}};\ \hat{c}_{\mathrm{V}}; \tag{11}$$

For the simulation of vibrations by means of the model M (program TURBO simulating the vibration of investigated object), the unbalance case will be described by the eccentricity of each disc

$$ecc_{\mathrm{I}};\ ecc_{\mathrm{II}};\ ecc_{\mathrm{III}};\ ecc_{\mathrm{IV}};\ ecc_{\mathrm{V}}; \tag{12}$$

and their angular positions

$$\alpha_{\mathrm{I}};\ \alpha_{\mathrm{II}};\ \alpha_{\mathrm{III}};\ \alpha_{\mathrm{IV}};\ \alpha_{\mathrm{V}}; \tag{13}$$

in a coordinate system rotating together with the shaft.

It is not expected that the values (12), (13) will be calculated by the inverse model *N*. To prepare the data for training of the model *N* it is necessary to establish a relationship between the parameters (12), (13) and the membership functions (11). Expected values of the membership functions (11) will be calculated (during simulations) as relative eccentricity of the given disc i (for i = I, II, ..., V)

$$c_i = \frac{ecc_i}{ecc_{\mathrm{I}} + ecc_{\mathrm{II}} + ecc_{\mathrm{III}} + ecc_{\mathrm{IV}} + ecc_{\mathrm{V}}} \tag{14}$$

It should be pointed out that we would like to predict only the location of the unbalance along the shaft and not their angular positions although it is evident e.g. from the Fig. 7 that measured vibration depends strongly on the angular position. It means that the angular position belongs to the parameters pointed out as $\{MID\}$ in (1).

We will try to design the model *N* taking into account the basic *kiss*-rule for all designing processes (***keep it simple, stupid***). It means we assume that the model will be simple, i.e. the number of layers, the number of nodes and the number of input parameters will be small. We will consider the maximal amplitudes of the shaft displacements measured in points A and B (Fig. 4) at the following rotating speeds only:

$$n_1 = 3000; \quad n_2 = 5000; \quad n_3 = 7000; \quad [1/\min] \tag{15}$$

It is important that we do not select the critical speeds. Such treatment is convenient for applications, where it will be possible to obtain the proper quality of measurements done at stabilised rotating speeds (e.g. specified in practical recommendations for run-up procedure).

From the above assumptions it follows that for each case of unbalance there will be 6 parameters (maximal amplitudes of displacement) calculated by the model M

$$a_{A,1}; \quad a_{A,2}; \quad a_{A,3}; \quad a_{B,1}; \quad a_{B,2}; \quad a_{B,3}; \quad [\mu\text{m}] \tag{16}$$

Because the inverse model N should identify the unbalance in a qualitative form, it is convenient to convert the parameters (16) into non-dimensional, relative values

$$b_{A,1}; \quad b_{A,2}; \quad b_{A,3}; \quad b_{B,1}; \quad b_{B,2}; \quad b_{B,3}; \tag{17}$$

Values (17) will be used as input parameters for the inverse model N. The simple method to obtain such values and to keep the relations between the measurements done at different rotating speeds in both measuring points is to use the formula

$$b_{i,j} = \frac{a_{i,j}}{a_{A,1} + a_{A,2} + a_{A,3} + a_{B,1} + a_{B,2} + a_{B,3}} \qquad \text{for} \quad i = a,b; \quad j = 1,2,3 \tag{18}$$

3.4. Inverse Model

The investigated model N can be designed independently for each disc. In the example we discuss the simplified version of the model by means of which it will be possible to estimate the unbalance for the first disc only. Similar steps can be done to estimate unbalance on other discs. Such steps will be necessary for the prediction of the unbalance along the shaft and for pointing out the disc with maximal unbalance by means of (5). Intuition suggests that the structure of the network should be 'adjusted' to the current problem and should take into account the specific properties of the task. In our case it is difficult to find such general features of the task.

We consider the model N for the disc I as the neural network (Fig. 8) with p=6 inputs and r=1 output. Basing on general recommendations (e.g. [9]) we decide to use

$$q \approx \sqrt{p \cdot r} \approx 3 \tag{19}$$

hidden nodes. The output node [2,0] of the network (Fig. 8) will be calculated according to (7) as a value of the sigmoid function (8). This function takes its values in the open range from 0.0 to 1.0 only and the network is not able to learn outputs outside this range. To avoid extremely large absolute values of the argument z in (8) it is required to keep output values in a restricted smaller range. We assume that the training values of the output should belong to the range from 0.15 to 0.85 . Due to this assumption the values c_i (14) and $\hat{c}_i$ (11) ranging from 0.0 to 1.0 will be converted into new values d_i and $\hat{d}_i$, by means of the following linear transformations (for $i = \mathrm{I, II}, ..., \mathrm{V}$):

$$\begin{aligned} d_i &= 0.15 + 0.7 \cdot c_i \\ \hat{d}_i &= 0.15 + 0.7 \cdot \hat{c}_i \end{aligned} \tag{20}$$

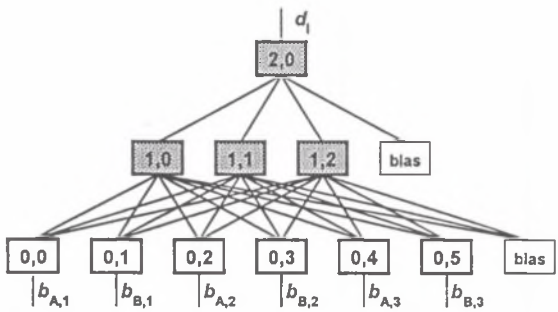

Fig. 8. Neural network (model N)
Rys. 8. Sieć neuronalna (model N)

3.5. Training Phase

To describe the network (Fig. 8) it is necessary to specify

$$s = (p+1) \cdot q + (q+1) \cdot r = 25 \tag{21}$$

weights $w_{l,j,i}$ mentioned in (7). It is clear that the number of training cases should be significantly greater than the number of unknown weights. We decide to use 95 unbalance cases denoted as U001, ..., U095. They include the different static as well as dynamic unbalances and result in different vibration modes.

Training of the network is a long term process. Before we start the training of the selected network it is reasonable to check that the task is solvable. How to check that there are any properties of the training data that make solution impossible? If we consider Fig. 2 one way is to check that different values of inputs to the model N produce the different outputs of the model. We take into account the distances between all pairs of the unbalance cases u_i and u_j (such that $i > j$) calculated in the space of input parameters

$$dist_{in}(u_i, u_j) = \sqrt{\left(b_{A,1}(u_i) - b_{A,1}(u_j)\right)^2 + \cdots + \left(b_{B,3}(u_i) - b_{B,3}(u_j)\right)^2} \tag{22}$$

and in the space of output parameters

$$dist_{out}(u_i, u_j) = \left| d_{\mathrm{I}}(u_i) - d_{\mathrm{I}}(u_j) \right| \tag{23}$$

Due to the definition of unbalance cases U001, ..., U095 and due to (20) there are possible only the following values of $dist_{out}$

$$0.15;\quad 0.325;\quad 0.5;\quad 0.675;\quad 0.85; \tag{24}$$

The minimum, mean and maximum values of distances $dist_{in}$ (22) corresponding to the above distances $dist_{out}$ (for the discussed unbalance cases U001, ..., U095) are presented in Fig. 9. From the figure it follows that training data are correct and they do not contain the cases presented in Fig. 2. It can also be expected that the quality of results will be better for values of $dist_{out}$ above 0.5 (i.e. for the main unbalance located on the disc I).

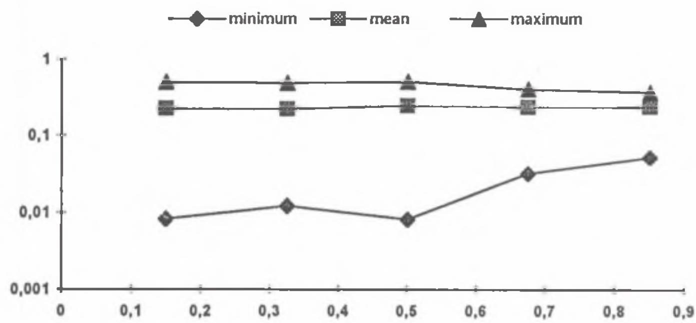

Fig. 9. Distances in the input space corresponding to the distances in output space
Rys. 9. Odległości w przestrzeni wejściowej odpowiadające odległościom w przestrzeni wyjściowej

The values $a_{A,1},\ldots,a_{B,3}$ (16) for each unbalance case were calculated by the program TURBO [14] on a mainframe computer. Training of the network (Fig. 8) was carried on a PC by the program MAS [1]. Program MAS (an expert system shell) can handle frames and uncertain decision tables. It contains an interface to couple simulation procedures. The part of MAS responsible for training of neural networks is based mainly on procedures described in [9]. The aim of calculations carried on in such an inconsistent environment was to show that it will be possible to realise the basic idea discussed in the paper, i.e. that it will be possible to invert existing, well validated models supplied in the form of closed computer programs, without any modifications of the programs.

3.6. Quality of Results

The network (Fig. 8) was trained *by epoch*. It means that output errors and corrections of network weights were calculated for a set of training cases and not for each training case individually. Such training strategy allows a stable solution to be obtained. The set of training data (cases U001, ..., U095) contain unequal numbers of cases for particular output values d_i

(20). To compensate for this, the cases U001, ..., U041 were exposed 4 times for one exposition of remaining cases.

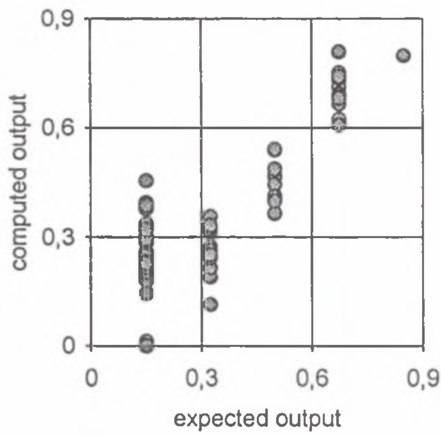

Fig. 10. Output values from the network (Fig. 8)
Results of training (cases U001, ..., U095)
Fig. 10. Wartości wyjściowe sieci neuronalnej (Rys. 8)
Wyniki fazy trenowania sieci (dane U001, ..., U095)

Results of training are presented in Fig. 10 and Fig. 11. The mean square error of the output values from the network

$$\sigma = \sqrt{\frac{1}{95-1}\sum_{i=1}^{95}\left(d_1(u_i) - \hat{d}_1(u_i)\right)^2} = 0.1015 \tag{25}$$

There are only the following 5 cases with errors greater than $2 \cdot \sigma$

Case	*expected output*	*calculated output*	*error*
U039	0,3250	0,1131	+0,2119
U044	0,1500	0,3941	-0,2441
U048	0,1500	0,3779	-0,2279
U053	0,1500	0,4546	-0,3046
U056	0,1500	0,3875	-0,2375

and there are no cases with errors greater then $3 \cdot \sigma$. From the histogram (Fig. 11) it follows that the confidence interval for the output from the network can be set as

$$\hat{d}_i - 0.2 < d_i < \hat{d}_i + 0.2 \tag{26}$$

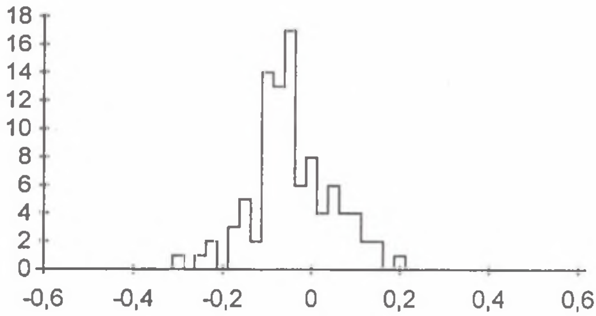

Fig. 11. Histogram of the network errors $(d_i - \hat{d}_i)$ for the cases U001, ..., U095
Rys. 11. Histogram błędów sieci neuronalnej $(d_i - \hat{d}_i)$ dla danych U001, ..., U095

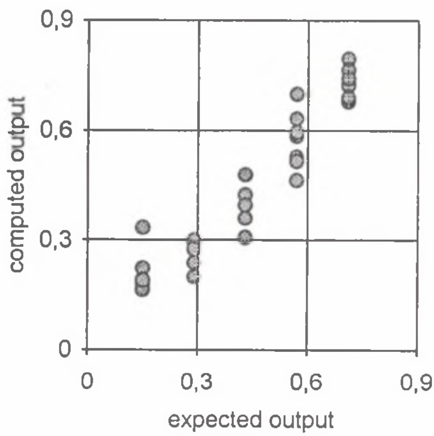

Fig. 12. Output values from the network (Fig. 8) for the cases U100, ..., U130
Rys. 12. Wartości wyjściowe sieci neuronalnej (Rys. 8) dla danych U100, ..., U130

As mentioned in section 3.5 the quality of results in Fig. 10 is better when the main unbalance is located on disc *I* (i.e. at higher output values d_I). The greater deviations for the small values of outputs (i.e. when there is little or no unbalance on disc *I*) are caused mainly by the large number of different unbalance distributions (cases U042, ..., U095) that are represented during training as single number $d_I = 0.15$. Moreover this feature of training cases was slightly amplified by the described training strategy. The discussed deviations should not result in serious errors when we look for the disc with maximum unbalance. By means of similar networks designed for the other discs (*II*,...,*V*) it will be possible to identify the qualitative distribution of unbalance along the shaft.

The network was tested after training with new data not used for training (cases U100, ..., U130). Results of the test are shown in Fig. 12. Even though the test data contained significantly different unbalance distribution, where both eccentricity and angular location were changed compared with the training data, the quality of the test results are seen to be similar to that of the training results in Fig. 10.

4. CONCLUSIONS

The results show how a neural network and inverse rotordynamic model can be trained and used to identify unbalance distribution in a multidisc shaft, by considering maximal amplitudes of shaft vibration measured at two locations for three different rotating speeds.

A numerical example demonstrates that the unbalance on a selected disc can be qualitatively evaluated even though the total distribution and angular location of unbalance along the shaft is unknown. Calculations should be continued for all discs. To obtain better quality of results one should take into account more input parameters, connected with greater number of considered rotating speeds as well as with the shape of orbits of the shaft in bearings described by the ratio of maximum and minimum displacement or by more detailed fuzzy clusters of orbits.

It was shown that the inversion of rotordynamics models given in the form of computer programs is possible. This opens the area for extensive investigations that seems to be very important for technical diagnostics. It should be possible to design inverse models for identification of serious changes in bearing clearances, stiffness and damping coefficients and other non-measurable diagnostic symptoms.

ACKNOWLEDGEMENTS

The investigations reported in this paper were partially supported by the research fellowship granted by the Norwegian Shipowners Fund at The Norwegian Institute of Technology. W.Cholewa appreciates the character of hospitality and kindness at The Division of Marine Engineering in The University of Trondheim.

REFERENCES

[1] W.Cholewa: *Szkieletowy System Doradczy MAS. Dokumentacja Użytkownika.* (wersja prototypowa 0.34). Politechnika Śląska, Gliwice 1993.

[2] W.Cholewa, M.F.White: *Inverse Modelling in Rotordynamics for Identification of Unbalance Distribution.* Machine Vibration, vol.?, 1993 (accepted for publication).

[3] F.F.Ehrich: *Handbook of Rotordynamics.* McGraw-Hill, New York 1992.

[4] J.A.Feldman, M.A.Fanty, N.H.Goddard, K.J.Lynne: *Computing with Structured Connectionist Networks.* Communications of the Association for Computing Machinery. 31 (2) 1988, pp.205-254.

[5] C.M.Harris, Ch.E.Crede: *Shock and Vibration Handbook.* McGraw-Hill, New York 1976.

[6] J.Hertz, A.Krogh, R.G.Palmer: *Introduction to the theory of neural computation.* Addison-Wesley, Redwood City 1992.

[7] T.Khanna: *Foundations of Neural Networks.* Addison-Wesley, Reading 1990.

[8] R.P.Lippman: *An Introduction to Computing with Neural Nets.* IEEE ASSP Magazine, 4, 1987, pp.4-22.

[9] T.Masters: *Practical Neural Network Recipes in C++.* Academic Press, San Diego 1993.

[10] *Neural Computing. User's Manual for Neural Works.* Neural Ware Inc., Pittsburgh 1991.

[11] D.E.Rummelhart, J.L.McClelland: *Parallel Distributed Processing.* MIT Press, Cambridge 1986.

[12] B.D.Shiver: *Artificial Neural Systems.* IEEE Computer, 21 (3), 1988.

[13] J.M.Vance: *Rotordynamics of Turbomachinery.* J.Wiley&Sons, New York 1988.

[14] M.F.White: *TURBO: Turbo-Rotor Dynamics - Calculation of Forced Response. Users Manual.* Marine Technology Centre, Trondheim 1988.

[15] L.A.Zadeh: *Fuzzy Sets.* Information and Control. 8, 1965, pp.330-353.

Recenzent: Prof. dr hab. inż. Ryszard Knosala

Wpłynęło do Redakcji w grudniu 1993 r.

Streszczenie

Modele matematyczne stosowane do opisywania procesów drgań maszyn pozwalają na wyznaczanie ocen drgań w funkcji ich przyczyn, takich jak cechy konstrukcyjne obiektu, niewyrównoważenie, parametry eksploatacyjne. Optymalizacja działania maszyny wymaga poszukiwania zmian przyczyn drgań, które prowadzić będą do ich eliminacji lub ograniczenia. Przykładem mogą być procesy wyrównoważania wirników. Bezpośrednie odwracanie znanych modeli matematycznych w celu określania przyczyn drgań jako funkcji skutków napotyka na dużo trudności.

Zaproponowano postępowanie polegające na trenowaniu modelu (rys. 3) występującego w postaci czarnej skrzynki, na podstawie danych uzyskiwanych w wyniku symulacji realizowanej za pomocą znanego, odwracanego modelu. Taką czarną skrzynką może być sieć neuronalna. Zwrócono uwagę na istnienie modeli, dla których wyznaczenie modeli odwrotnych nie będzie możliwe (rys. 2).

Zastosowanie proponowanej metody pokazano na przykładzie zadania dotyczącego wirnika z niewyrównoważeniami (rys. 4). Dane trenujące dla tego przykładu generowano za pomocą programu TURBO [14], uwzględniając między innymi zjawiska nieliniowe zachodzące w łożyskach. Program ten był odwracanym modelem M (rys. 1). Pozwolił on na wyznaczenie maksymalnych amplitud drgań wirnika względem podpór łożyskowych w dwóch płaszczyznach i dla różnych prędkości wirowania wału. Amplitudy te są zależne od rozkładu niewyrównoważenia (rys. 6), (rys. 5). Przyjęto model odwrotny N (rys. 1) w postaci prostej sieci neuronalnej (rys. 8), posiadającej trzy warstwy węzłów. Sygnałami wejściowymi tej sieci były względne amplitudy drgań (17) określane dla trzech prędkości. Oczekiwanym sygnałem wyjściowym była waga (14) określająca stopień niewyrównoważenia wybranej tarczy wirnika. Dla każdej tarczy definiowany jest jej indywidualny model w postaci odrębnej sieci neuronalnej. Trenowanie sieci przeprowadzono za pomocą programu MAS [1]. Wynikiem procesu trenowania sieci były wagi (7) opisujące jej węzły. Wynik procesu trenowania (sieć neuronalną) testowano za pomocą niezależnego zbioru danych symulacyjnych, otrzymując pozytywny wynik (rys. 11), (rys. 12).

Otrzymany model odwrotny pozwala na identyfikację rozkładu niewyrównoważenia wzdłuż wirnika na podstawie wyników pomiarów drgań w dwóch płaszczyznach. Znane metody wyrównoważania polegają na umieszczaniu, w wybranych płaszczyznach, mas kompensujących istniejące (nieznane) niewyrównoważenie i nie pozwalają na uzyskanie podobnego ogólnego rozwiązania.