ARTIFICIAL NEURAL NETWORK STUDY APPLIED TO RIGID ROTOR DYNAMIC BALANCING

Santos, F.L.
Former M.Sc. Student of the Department of Mechanical Engineering (DEMEC), Federal University of Minas Gerais (UFMG).
Av. Antônio Carlos, 6627 – Pampulha, Campus Universitário, Belo Horizonte/MG, 31270-901
email: ffabiolss@yahoo.com.br

Duarte, M.L.M.
Lecturer and Researcher of the Department of Mechanical Engineering (DEMEC), Federal University of Minas Gerais (UFMG).
Av. Antônio Carlos, 6627 – Pampulha, Campus Universitário, Belo Horizonte/MG, 31270-901
email: mlduarte@dedalus.lcc.ufmg.br

Eduardo, A.C.
Post-Doc of the Department of Mechanical Engineering (DEMEC), Federal University of Minas Gerais (UFMG).
Av. Antônio Carlos, 6627 – Pampulha, Campus Universitário, Belo Horizonte/MG, 31270-901
email: aceduard2003@yahoo.com.br

Abstract. *Artificial Neural Networks (ANN) have been used for pattern recognition in many engineering areas. One of these areas is mechanical systems monitoring and fault diagnostics. This paper deals with balancing of a rigid rotor supported by hydrodynamic journal bearings applying artificial neural network. Training and validation set are generated considering both the system unbalancing responses and the location of the correction masses provided by the dynamic balancing process employed. The ANN is trained and afterwards validated with these data in order to predict the location of the correction masses for the case when only the unbalancing responses are provided to the neural network. Levenberg-Marquardt and Backpropagation training algorithm are compared, as well as the neural network design. Training and validation data are normalized in order to improve the neural network capability when unknown data are provided to ANN. This paper shows the influence of both the ANN design and the training algorithm in neural network response. It also highlighties the importance of considering normalized data.*

Keywords: *Artificial Neural Network Design, Training Algorithm, Dynamic Balancing, Rigid Rotor.*

1. Introduction

The most common source of vibration in turbo machines is the rotor mass unbalance. The unbalance forces generated are transmitted both to the bearings and the foundations. Such forces may damage the system and in some cases even affect other equipments in the vicinity. Rigid and flexible rotors can be balanced by specific balancing techniques. Plane separation technique is normally used for balancing rigid rotors, while Modal Balancing Method and Influence Coefficient Method can be used for balancing flexible rotors (Rao, 1983; Vance, 1988; Childs, 1993).

Artificial Intelligence (AI), specifically Artificial Neural Network (ANN) techniques, has been used for monitoring and fault diagnostics of mechanical systems. The Artificial Neural Network training and validation can use either experimental or theoretical data (He *et al*., 2001; Skoundrianos and Tzafestas, 2002). Ganesan *et al.,* (1995) applied ANN on diagnostics and instability control of high speed rotating systems using analytical model with great success. The results presented by (Vyas and Satishkumar, 2001) also showed the great capability of the artificial neural network in fault prediction found in rotating machines. Despite the researches published about monitoring and fault detection using artificial intelligence techniques, there are few studies in which such techniques are used to predict the balancing corrections in mechanical systems. The study presented here works exactly on that direction.

This paper presents an Artificial Neural Network study applied to rigid rotor balancing. The Stodola-Green shaft model is used to represent the rigid rotor in which the gyroscopic moments and rotary inertia are taking into account. The rotor is supported by hydrodynamic journal bearings considered as a linearized bearing model where the damping and stiffness coefficients are computed using a finite element procedure specially devised for that (Santos, 2005). Both direct and cross-coupling coefficients are computed. Such models are presented in section 2. Training and validation sets are generated using the system unbalancing response and the correction masses provided by the plane separation balancing technique, as shown in sections 3 and 4. These sets are normalized in order to improve the neural network performance, as shown in section 5. The neural network design considering two different ANN architectures (as presented in section 5.1) are compared, as well as the Levenberg-Marquardt and Backpropagation training algorithm (as presented in section 5.2). The results in section 6, concluded in section 7, show the influence of the ANN design and the training algorithm in the neural network response, besides highlighting the importance of considering normalized data.

2. Model

The equations of motion for a rigid rotor can be obtained from the Lagrangian calculated in terms of the Euler angles (Childs, 1993). The Lagrangian is composed only by the translational and rotational kinetic energy for the rotor bearing system as given by equation (1):

$$L = T = \frac{1}{2}M(\dot{X}^2 + \dot{Y}^2) + \frac{1}{2}I_t(\dot{\alpha}^2 + \dot{\beta}^2) + \frac{1}{2}I_p(\omega^2 - 2\omega\dot{\alpha}\beta) \quad (1)$$

In this work, the rotor bearing system model uses the Stodola-Green shaft model in which a rigid rotor is supported by hydrodynamic journal bearings localized on Z = ± L/2, according to Figure 1.

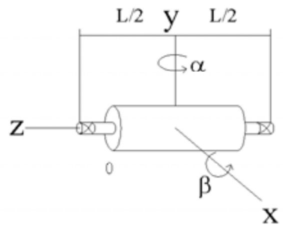

Figure 1. Rigid Rotor

The equations of motion, Eq. (2), for the rotor bearing system are obtained using Lagrange formulation (Vance, 1988), with *X*, *Y*, α and β as the generalized coordinates of the system and *F* and *M* as the generalized forces and moments, respectively. The terms involving the polar moment of inertia (I_p) are known as gyroscopic moments.

$$\begin{aligned} M\ddot{X} &= \sum F_X \\ M\ddot{Y} &= \sum F_Y \\ I_t\ddot{\beta} + I_p\omega\dot{\alpha} &= \sum M_X \\ I_t\ddot{\alpha} - I_p\omega\dot{\beta} &= \sum M_Y \end{aligned} \quad (2)$$

The generalized forces and moments are obtained by the dynamic stiffness and damping coefficients of the journal bearings and by the rotor unbalancing. The damping and stiffness forces are calculated considering the velocity and displacement in the *X* and *Y* direction, respectively. The cross-coupling stiffness (damping) coefficients are obtained assuming that a displacement (velocity) in the *X* direction produces a force in the *Y* direction, and vice-versa. It is known (Vance, 1988) that the cross-coupling stiffness coefficients have opposite sign, whereas the cross-coupling damping coefficients have the same sign. However, it is important to point out here that the coupling coefficients are normally ignored in general modeling formulations, although they have a strong influence on the obtained results. So, Eq. (3) represents the generalized forces and moments in terms of the damping and stiffness coefficients, including in the formulation the coupling terms:

$$\begin{aligned} \sum F_X &= -2C_{XX}\dot{X} - 2C_{XY}\dot{Y} - 2K_{XX}X - 2K_{XY}Y \\ \sum F_Y &= -2C_{YY}\dot{Y} - 2C_{YX}\dot{X} - 2K_{YY}Y + 2K_{YX}X \\ \sum M_X &= -C_{YY}\frac{L^2}{2}\dot{\beta} - C_{YX}\frac{L^2}{2}\dot{\alpha} - K_{YY}\frac{L^2}{2}\beta + K_{YX}\frac{L^2}{2}\alpha \\ \sum M_Y &= -C_{XX}\frac{L^2}{2}\dot{\alpha} - C_{XY}\frac{L^2}{2}\dot{\beta} - K_{XX}\frac{L^2}{2}\alpha - K_{XY}\frac{L^2}{2}\beta \end{aligned} \quad (3)$$

Therefore, the equations of motion for the system, Eq. (4), can be obtained substituting Eq. (3) into Eq. (2) and including the generalized forces and moments caused by the rotor unbalance. The latter is represented by the RHS of Eq. (4). The system is represented by a rigid rotor supported by hydrodynamic journal bearings.

$$\begin{aligned} M\ddot{X} + 2C_{XX}\dot{X} + 2C_{XY}\dot{Y} + 2K_{XX}X + 2K_{XY}Y &= \sum_{i=1}^{m} m_i\omega^2u_i\cos(\omega t + \psi_i) \\ M\ddot{Y} + 2C_{YY}\dot{Y} + 2C_{YX}\dot{X} + 2K_{YY}X - 2K_{YX}X &= \sum_{i=1}^{m} m_i\omega^2u_i\text{sen}(\omega t + \psi_i) \\ I_t\ddot{\beta} + I_p\omega\dot{\alpha} + C_{YY}\frac{L^2}{2}\dot{\beta} + C_{YX}\frac{L^2}{2}\dot{\alpha} + K_{YY}\frac{L^2}{2}\beta - K_{YX}\frac{L^2}{2}\alpha &= \sum_{i=1}^{m}(m_i\omega^2u_i\cos(\omega t + \psi_i))l_i \\ I_t\ddot{\alpha} - I_p\omega\dot{\beta} + C_{XX}\frac{L^2}{2}\dot{\alpha} + C_{XY}\frac{L^2}{2}\dot{\beta} + K_{XX}\frac{L^2}{2}\alpha + K_{XY}\frac{L^2}{2}\beta &= \sum_{i=1}^{m}(m_i\omega^2u_i\text{sen}(\omega t + \psi_i))l_i \end{aligned} \quad (4)$$

The cross-coupling stiffness terms act on the general damping levels of the system, so increasing the levels of the responses obtained (Childs, 1993; Vance, 1988). That stresses the importance of including such terms during the modeling stage, since their exclusion will underestimate the unbalance responses of the system.

3. Integration of the Equation of Motion

The fourth order *Runge-Kutta* integration method, as presented by Eq. (6), can be used to obtain the time responses of the rotor bearing system, in terms of coordinates *X*, *Y*, α and β.

$$y_{n+1} = y_n + \frac{1}{6}\left(k_{n1} + 2k_{n2} + 2k_{n3} + k_{n4}\right) \quad (5)$$

where:
$$\begin{aligned} k_{n1} &= f(t_n, y_n)\Delta t \\ k_{n2} &= f(t_n + \tfrac{1}{2}\Delta t, y_n + \tfrac{1}{2}k_{n1})\Delta t \\ k_{n3} &= f(t_n + \tfrac{1}{2}\Delta t, y_n + \tfrac{1}{2}k_{n2})\Delta t \\ k_{n4} &= f(t_n + \Delta t, y_n + k_{n3})\Delta t \end{aligned} \quad (6)$$

The smaller the time increment considered (Δt), the better the resolution obtained for the time responses. From that, a Fast Fourier Transform (*FFT*) is used to give the responses of the system in the frequency domain.

For the present study, the responses of the system are obtained for the center of the rotor and for the two journal bearings of the system.

4. Balancing Equations

A rigid rotor can be balanced by adding correction masses in any two balancing planes. One technique used for that is the plane separation technique. It is important to emphasize that, for rigid balancing, the process needs to be performed considering speeds below its first critical, typically between 100 and 600 rpm (Rieger, 1988).

Figure 2 represents the unbalancing distributions along the shaft in which two balancing planes are defined.

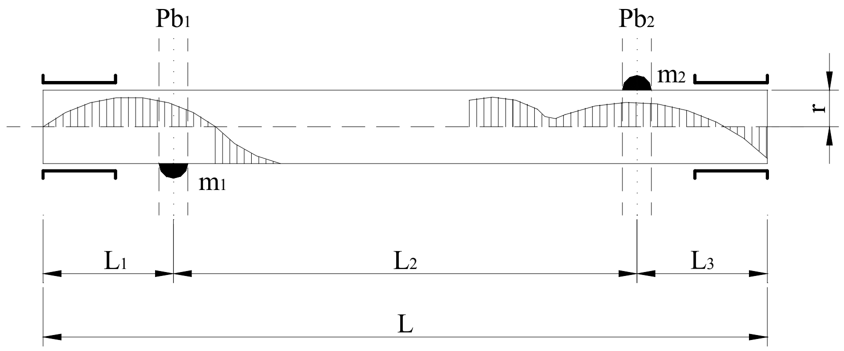

Figure 2. Balancing Planes

Equations (7) represent the force and moment equations, considering the balancing planes shown in Figure 2 obtained from the plane separation technique (Santos, 2005):

$$\begin{aligned} &\hat{R}_1 e^{-i\phi_1} + \hat{R}_2 e^{-i\phi_2} + m_1 r\omega^2 e^{i\phi_{b1}} + m_2 r\omega^2 e^{i\phi_{b2}} = 0 \\ &L\hat{R}_2 e^{-i\phi_2} + (L_1 + L_2) m_2 r\omega^2 e^{i\phi_{b2}} + L_1 m_1 r\omega^2 e^{i\phi_{b1}} = 0 \end{aligned} \quad (7)$$

Such equations when solved will reduce the unbalancing synchronous response of the rotor bearing system. The solution provides the correction masses (m_1 and m_2) and their respectively angular positions (ϕ_{b1} and ϕ_{b2}), as represent by the equations (8),

$$\begin{aligned} S_1 &= m_1 e^{i\phi_{b1}} \\ S_2 &= m_2 e^{i\phi_{b2}} \end{aligned} \quad (8)$$

5. Artificial Neural Network

A neural network is a massively parallel, self-adaptive, interconnected network of basic elements called neurons. These neurons computation is based on the brain neurobiological computational processes. Neurons have a simpler computational process but the interactions between them allow the Artificial Neural Network to learn from given input sets and their corresponding outputs (Haykin, 1999).

A Multilayer Perceptron Network is composed by: an input layer, one or more hidden layer and an output layer, as shown in Fig. 3. This kind of Artificial Neural Network has greater computational power when compared with one layer Neural Network. The most commonly used Neural Network training algorithms are the Error Backpropagation and the Levenberg-Marquardt Algorithm.

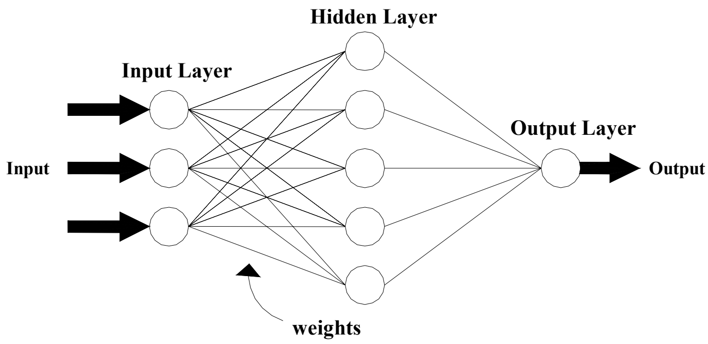

Figure 3. Multilayer Perceptron Network

Error Backpropagation training algorithm attempts to minimize the square of the error obtained between the Artificial Neural Network output and the desired output, considering a descent gradient technique to change the synaptic weights. Two steps are considered: the first one consists of a forward step through the network and the second one realizes the synaptic weight correction, according to Eq. (9):

$$w_{ji}^{(l)}(n+1) = w_{ji}^{(l)}(n) + \zeta[w_{ji}^{(l)}(n-1)] + \gamma\delta_j^{(l)}(n)y_i^{(l-1)}(n) \tag{9}$$

where : $\delta_j^{(l)}$ = represents a local gradient for a neuron j in a layer l; ζ = momentum constant; γ = learning rate.

Levenberg-Marquardt training algorithm applies a modification of Newton's method allowing a faster convergence then methods based on descent gradient form. Levenberg-Marquard algorithm employs a second order training using an approximation for the Hessian matrix based on the product of the Jacobians matrices, as shown by Eq. (10):

$$H = J^T J \tag{10}$$

where: H = Hessian matrix;
J = Jacobian matrix.

The synaptic weights are corrected using Eq. (11):

$$w(n+1) = w(n) - \left[H(w(n)) + \mu(n)I\right]^{-1}.\nabla\varphi(w(n)) \tag{11}$$

where: $H(w(n))$ = Hessian matrix for $w(n)$;
$\varphi(w(n))$ = Function to minimize $w(n)$;
$\nabla\varphi(w(n))$ = Gradient function of $\varphi(w(n))$;
μ = Determines the algorithm trend: if μ = 0 Newton's method is employed to obtain the Hessian matrix; if μ was large, gradient descendent must be employed.

Polar input data, related to the unbalancing response of the rotor bearing system, are calculated in Cartesian Coordinates according equation (12). The same procedure is performed for the output data, related to the balancing correction masses, as represented by the equation (13).

$$x_{ei} = \cos(\phi_i)\hat{R}_i$$
$$y_{ei} = \text{sen}(\phi_i)\hat{R}_i \tag{12}$$

where: x_{ei} and y_{ei} = input Cartesian coordinates;
ϕ_i = phase angle i;
$\hat{R}_i$ = System response at i position.

$$\begin{aligned} x_{si} &= \cos(\phi_{bi})m_i \\ y_{si} &= \mathrm{sen}(\phi_{bi})m_i \end{aligned} \tag{13}$$

where: x_{si} and y_{si} = output Cartesian coordinates;
ϕ_{bi} = correction angle obtained for the balancing plane i;
m_i = correction mass obtained for the balancing plane i.

In order to improve the Artificial Neural Network generalization capability, the input and output data need to be normalized. So, Eqs. (14) are employed for the input and output data normalization that will be used on the Artificial Neural Network training and validation steps (Saldarriaga and Steffen, 2003).

$$\begin{aligned} N(y) &= 2\frac{(R(y)-y_{\min})}{(y_{\max}-y_{\min})}-1 \\ N(x) &= 2\frac{(R(x)-x_{\min})}{(x_{\max}-x_{\min})}-1 \end{aligned} \tag{14}$$

where: $N(x)$ and $N(y)$ = x and y normalized coordinates;
$R(x)$ and $R(y)$ = x and y coordinates to be normalized;
$y_{\min}$ and $y_{\max}$ = maximum and minimum displacement of the coordinate y;
$x_{\min}$ and $x_{\max}$ = maximum and minimum displacement of the coordinate x.

5.1. Network Design

There are a lot of possible ANN designs to be considered. Most researches include either a single or double hidden layers. For the present study, the best ANN design was obtained as described in Table 1.

Table 1 – Artificial Neural Network Design

Artificial Neural Network Design				
Layers	input-output relationship			
6 x n x n x 4	Input	x_n(bearing 1)	Output	
		y_n(bearing 1)		x_n(balancing plane 1)
				y_n(balancing plane1)
		x_n(bearing 2)		
		y_n(bearing 2)		
				x_n(balancing plane 2)
		x_n(center of the rotor)		y_n(balancing plane 2)
		y_n(center of the rotor)		

5.2. Artificial Neural Network Parameters

For the learning procedure, the Error Backpropagation and Levenberg-Marquardt algorithm (Haykin, 1999) were employed. The artificial neural network learning parameters used in this stage are described in Table 2.

Table 2 – Artificial Neural Network Parameters

Neural Network Parameters	**Error Backpropagation**	**Levenberg-Marquardt**
Transfer function	sigmoid function	sigmoid function
Rate of learning	0,03	0,05
Increase factor of learning	1,05	1,05
Performance goal	0,001	0,001
Momentum	0,05	0,075
Iterations	3000	3000

6. Results

In this study, the rotor bearing system was simulated using theoretical data. The rotor parameters employed in the modeling process are described in Table 3. The length and diameter are such that the rotor can be considered a rigid structure and the material employed simulates a common one found in the industry.

Table 3 – Rotor Parameters

Rotor Parameters	
Length (m)	0,6
Diameter (m)	0,05
Specific mass (kg/m^3)	7800

The rotor is supported by hydrodynamic journal bearings. The damping and stiffness dynamic coefficients of the bearings employed in system simulation are presented in Table 4 (Santos, 2005).

Table 4 – Dynamic Coefficients

Journal Bearings Dynamic Coefficients			
Stiffness Coefficients (N/m)			
K_{XX}	K_{XY}	K_{YY}	K_{YX}
0,2310E+06	0,2449E+05	0,2020E+06	-0,2182E+05
Damping Coefficients (N.s/m)			
C_{XX}	C_{XY}	C_{YY}	C_{YX}
1,718E+03	1,672E+03	1,713E+03	1,672E+03

From a random unbalancing distribution along the shaft, the response of the system were obtained using the parameters given in Table 3 and the dynamic coefficients given in Table 4, considering a balancing speed of 400 rpm. The unbalancing responses were obtained at the rotor center and at the bearings. The same procedure was adopted for the bearings position and similar results were obtained.

The database was generated using the unbalancing responses of the system and the correction masses given by the balancing process. From the database, the training and validation sets were obtained.

The training set was composed by 200 elements in which the unbalancing responses of the rotor bearing system are the input data and the correction masses are the output data. The convergence of the learning procedure was terminated at 0.1 % error threshold. Twenty (20) unknown elements formed the validation set. These elements were the input data corresponding to the unbalancing responses provided to the network, and the correction masses and angles were the output data.

Error Backpropagtion (Descent Gradient) and Levenberg-Marquardt training algorithm were available. The results showed that for the type of data employed in this work, descent gradient algorithm presented a worst performance when compared to Levenberg-Marquardt training algorithm. Table 5 indicates the training algorithm results. As seem, for all topologies studied, the Levenberg-Marquardt algorithm converged for the threshold error. However, Error Backpropagtion (Descent Gradient) did not converge after a maximum number of interactions. Moreover, the goal error was higher than the threshold error established. The training set was normalized since better results were obtained using normalized data in preliminary tests for both training algorithm studied.

Table 5 – Training Algorithm Results

Architecture	Topology	Number of Interactions Levenberg-Marquardt	Goal Error	Number of Interactions Error Backpropagation	Goal Error
1	6 x 10 x 10 x 4	1500	0,001	3000	0,251
2	6 x 15 x 15 x 4	1205	0,001	3000	0,188
3	6 x 20 x 20 x 4	388	0,001	3000	0,198
4	6 x 25 x 25 x 4	492	0,001	3000	0,231

In order to validate the artificial neural network employed, a validation set composed by 20 unknown elements was provided to the net and then the network results were compared with the correct ones. The results of the validation step showed that descent gradient presented a lower power of generalization when compared with the Levenberg-Marquadt training algorithm. Figures 4 and 5 present the neural network responses for the validation set considering architecture 3 (6 x 20 x 20 x 4) that was the one with the lesser number of interactions among the architectures (see Table 5).

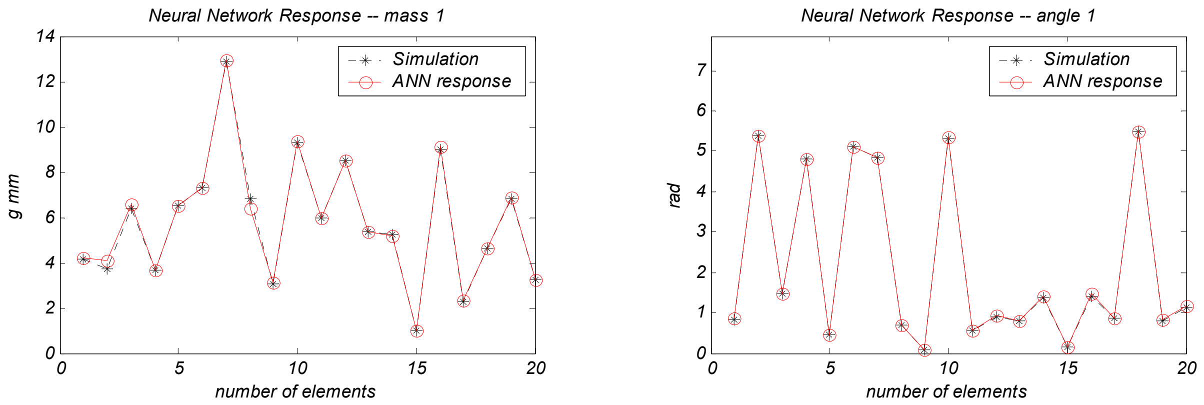

Figure 4 – Correction mass – balancing plane 1

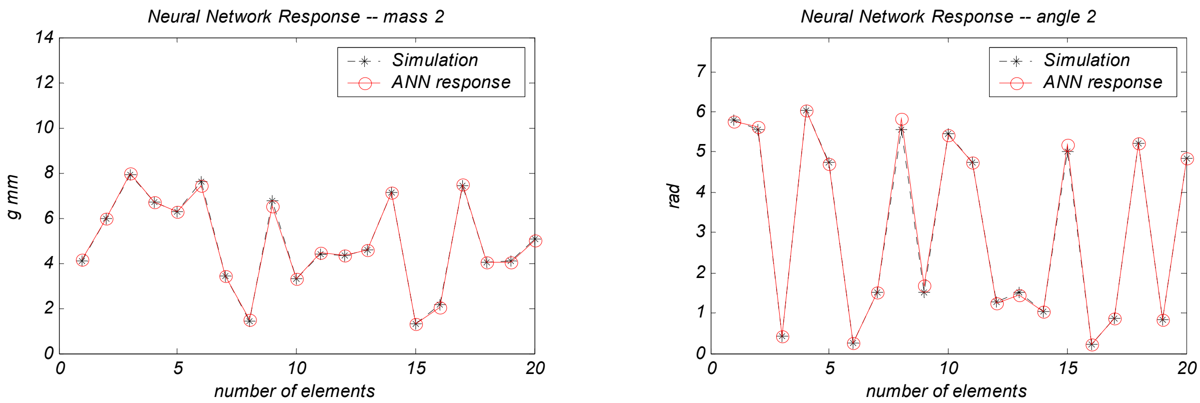

Figure 5 – Correction mass – balancing plane 2

The results show that the artificial neural networks using Levenberg-Marquardt training algorithm are able to predict the correction masses with satisfactory accuracy when the descent gradient algorithm presented low accuracy. The rate of error considering Levenberg-Marquardt algorithm was below 10% and only 4 elements of the validation set presented error rate above 5%. Nevertheless, when these 4 elements were used to balance the system, they presented balance response reductions up to 74%, confirming the great capability of the presented methodology.

7. Conclusions

A balancing methodology of rotors has been presented. The methodology applies artificial neural networks to predict the correction masses when the unbalancing responses of the system are provided to the neural network. The Levenberg-Marquardt and Error Backpropagation (descent gradient) training algorithm were compared. The Levenberg-Marquardt algorithm presented a better performance, converging for the 0,1% error threshold, when compared with the Error Backpropagation (descent gradient) algorithm. The latter presented low accuracy on training step and low generalization on validation step. Better results were obtained when the data employed in both training and validation tests were normalized. The artificial neural network, using Levenberg-Marquardt algorithm, presented a satisfactory performance when the validation set was used in order to validate the neural network. The results show that the error rate was below 10% and the reduction of the unbalancing response was up to 74%. Therefore, it may be conclude about the efficiency of the system balancing technique proposed.

8. References

Childs, D., 1993, “Turbomachinery Rotordynamics,” John Wiley & Sons, Inc, New York.

Ganesan, R., Jionghua, J., Sankar, T.S., 1995, "A classifier neural network for rotordynamic systems". Mechanical Systems and Signal Processing, Vol. 9, N. 4, pp. 397 - 414.
Haykin, S., 1999, "Neural Networks: A comprehensive foundation", Prentice Hall, New Jersey, USA, 2nd edition.
He, J. Z., Zhou, Z. H., Chen, S. F., 2001, "A general design technique for fault diagnostic systems", Proceedings INNS-IEEE International Joint Conference on Neural Networks II, Tokyo, Japan, pp. 1307 - 1311.
Rao, J. S., 1983, "Rotor Dynamics," New Delhi: Wiley Eastern Limited.
Rieger, N. F., 1988, "Principles of Balancing and of Balancing Machines – Rotordynamics 2, Problems in Turbomachinery", In: Internacional Centre for Mechanical Sciences, Courses and Lectures, N. 297. Edited by N. F. Rieger. Springer-Verlag-Wien, New York, pp. 67 - 93.
Saldarriaga, M. V., Steffen, V., 2003, "Balancing of flexible rotors without trial weights by using optimization techniques", Proceedings of COBEM – 17*th* International Congress of Mechanical Engineering. São Paulo, SP.
Santos, F.L., 2005, "Development of a Hybrid Dynamic Balancing Method for a Rigid Rotor Supported by Hydrodynamic Bearings" (Desenvolvimento de um Método Híbrido de Balanceamento Dinâmico de um Rotor Rígido Apoiado em Mancais Hidrodinâmicos, In Portuguese), M.Sc. Thesis, Federal University of Minas Gerais, Mechanical Engineering Department.
Skoundrianos, E. N., Tzafestas, S. G., 2002, "Fault diagnosis via local neural networks," Mathematics and Computer in Simulation, Vol. 60, pp. 169 - 180.
Vance, J.M., 1988, "Rotordynamics of Turbomachinery", John Wiley & Sons, Inc, New York.
Vyas, N. S., Satishkumar, D., 2001, "Artificial neural network design for fault identification in a rotor bearing system", Pergamon Mechanism and Machine Theory, Vol. 36, pp. 157 - 175.

9. Responsibility notice

The authors are the only responsible for the printed material included in this paper.

Nomenclature

X, Y, α, β = Generalized coordinates
ω = Rotor angular velocity
M = Rotor mass
I_p, I_t = Polar and transversal moment of inertia
$K_{XX}, K_{XY}, K_{YY}, K_{YX}$ = Stiffness coefficients
$C_{XX}, C_{XY}, C_{YY}, C_{YX}$ = Damping coefficients
m_i = Mass of rotor section
l_i = Length of rotor section
Δt = time increment
u_i = Eccentricity of unbalancing mass
L = Rotor length
$\delta_j^{(l)}$ = Represents a local gradient for a neuron j in a layer l;
ζ = Momentum constant;
γ = Learning rate.