IDENTIFICATION AND CONTROL OF ROTORCRAFT HUB LOADS USING NEURAL NETWORKS

Sesi Kottapalli
Rotorcraft Aeromechanics Branch
NASA Ames Research Center
Moffett Field, California

Abstract

The objective of the study was to develop a robust neural network based controller to minimize vibratory hub loads. A metric consisting of five vibratory hub load components (with their sine and cosine components) obtained from a wind tunnel test of a four-bladed rotor with individual blade control was used to characterize the hub loads. The neural network control procedure was bound by the following ground rules: the controller must converge quickly in six iterations or less and gradient based optimization techniques must not be used. A simple and straightforward iterative procedure for neural control was applied. Two neural networks were used in the procedure requiring a plant model (using a radial-basis function neural network) and an "inverted neural network for control" model (using a back-propagation neural network). A simple half-interval calculation which halves the metric was used in order to speed up convergence. The neural network control procedure successfully achieved the objective within the given constraints. Finally, a limited-scope comparison of the results from the present neural control procedure with those from a one-step deterministic controller showed that the two control methods were roughly comparable, with neural control being slightly more robust.

Notation

A	Rotor disk area, πR^2
A_2	Experimental 2P control amplitude input, deg
$A_{2N,\, i}$	Neural network controller 2P control amplitude input for the i'th iteration, deg
A_{2N}	2P control amplitude input predicted by inverted neural network for control, deg
A_m	Amplitude of mP IBC input, deg
c	Representative chord of the blade
C_T	Rotor thrust coefficient, thrust nondimensionalized by $\rho A(\Omega R)^2$
i	Blade number; i=1 implies ψ=0 for blade at helicopter tail
HHC	Higher Harmonic Control
IBC	Individual Blade Control
m	Harmonic number for IBC input
N_b	Number of blades
R	Rotor blade radius
RBF	Radial-basis function
[T]	transfer-function matrix, of size 10 x 2
V	Wind tunnel airspeed, knots
VHLM	Vibratory hub loads metric, made up of five hub load components with their sine and cosine components
{ z }	vector of measured vibratory hub loads, sine and cosine components, of size 10 x 1
α_s	Rotor shaft angle, positive nose up, deg
{ θ }	vector of 2P blade pitch control inputs, of size 2 x 1
{ θ^* }	vector of optimal 2P blade pitch control inputs, of size 2 x 1
θ_{im}	IBC contribution to blade pitch, m'th harmonic pitch for i'th blade

Presented at the American Helicopter Society 53rd Annual Forum, Virginia Beach, Virginia, April 29 - May 1, 1997.

μ advance ratio, $V/(\Omega R)$

σ Rotor solidity, $N_b c/\pi R$

Φ_2 Experimental 2P control phase input, deg

$\Phi_{2N,\,i}$ Neural network controller 2P control phase input for the i'th iteration, deg

Φ_{2N} 2P control phase input predicted by inverted neural network for control, deg

ψ Rotor azimuth angle, deg

Ω Rotor rotational speed

Introduction

The development and implementation of a robust active control system for helicopter aeromechanics must include a method for accurate identification of aircraft parameters and a robust scheme to generate optimal control inputs. For example, rotorcraft hub loads and vibration almost always behave nonlinearly with respect to the phase of a higher harmonic control (HHC) pitch input. This behavior is evident in: wind tunnel test results (Kottapalli, et al., Ref. 1); flight test results (Miao, et al., Ref. 2); and analytical results (O'Leary, et al., Ref. 3). In the preceding HHC cases and other similar cases, the controller's task would be to identify the nonlinear, vibration-related helicopter parameters and subsequently generate optimal HHC pitch control inputs that result in the lowest vibration level.

Neural network based techniques are attractive nonlinear methods for control of nonlinear systems. Neural networks do not necessarily require large amounts of computational resources or central processor time. Additionally, they appear easy to apply and understand. A successful neural network application enables the accurate nonlinear identification of important rotorcraft parameters and subsequent calculation of the optimal control inputs. An efficient neural network application can enable the hardware implementation of feedback-driven control systems. In the present context, hardware implementation refers to the complete control system and its functions (which include modeling, predicting, optimizing, and controlling).

The application of neural networks to rotorcraft dynamics control is still relatively new; therefore, available literature is limited (for example, Kottapalli, et al., Refs. 4 and 5). However, the following references have been useful for the present investigation. The handbook on intelligent control by White and Sofge (Ref. 6) covers neural and other approaches. Miller, et al. (Ref. 7), Wasserman (Ref. 8), Werbos (Ref. 9), Omatu (Ref. 10), and Pham and Liu (Ref. 11) discuss various neural control approaches. Werbos (Ref. 9) classifies existing neural control approaches into five types: supervised control systems, direct inverse control, neural adaptive control, back-propagation-through-time, and adaptive critic methods. Other classifications exist (for example, Omatu, et al., Ref. 10). Psaltis, et al. (Ref. 12) discuss architectures associated with training neural networks for neural control. The survey paper by Hunt, et al. (Ref. 13) covers neural networks for control systems. Faller, et al., (Ref. 14) consider a fixed-wing aerodynamics application of neural control, namely, real time identification and control of 3-D unsteady separated flow.

The objective of the present study is to develop a robust neural network based control to minimize vibratory hub loads. A metric consisting of five vibratory hub load components (with their sine and cosine components) obtained from a wind tunnel test of a four-bladed rotor with individual blade control is used to characterize the hub loads. The neural controller is required to be relatively quick in its execution and not be computationally intensive. Thus, the present neural network control procedure is bound by the following ground rules: the controller must converge in six iterations or less and gradient based optimization techniques must not be used.

This study begins with a description of two types of plant models: single-input, single-output and multi-input, multi-output. Next, the neural network control procedure (based on an existing, neural control technique called the "direct inverse" method) is described. The experimental data are then described. Next, five cases are examined. The first four cases pertain to neural control and for these cases, results are presented in the following sequence: plant model results, inverted neural network for control results, and finally, the neural controller's convergence sequence. The fifth case looks into the performance of a traditional, one-step deterministic controller and, in a limited manner, compares its results with those from the neural controller. Finally, observations derived from this study are presented.

Plant Model

Kottapalli, et al. (Refs. 4 and 5) have developed procedures for deriving neural network plant models that can be used in the present rotorcraft dynamics (controls) application. Single-input, single-output (SISO) and multiple-input, single-output (MISO) applications were considered in Ref. 4, and the multiple-input, multiple-output (MIMO) application was considered in Ref. 5. Background material on these applications is given in Ref. 4. The SISO applications are briefly described below.

Single-Input, Single-Output Plant Model

In the first of two SISO applications, the network training input is the 2P control phase input Φ_2. In the second SISO application, the network training input is the 2P control amplitude input A_2. In both applications, the network output is the vibratory hub loads metric, VHLM. Accurate plant modeling was obtained for the 2P control phase application by using a two-hidden-layer radial basis function (RBF) type of neural network depicted by "1-12-4-1 RBF network." In this depiction, the leading and trailing 1's refer to the single input and the single output, and the 12 and 4 refer to the number of processing elements in the first and second hidden layers, respectively.

Neural Network Control Procedure

The present control procedure was finalized after conducting a survey of neural network control techniques and trying several control procedures. In the present SISO case it is possible to obtain accurate modeling and prediction for simulating the plant model being controlled. Therefore, the present control procedure does not include the plant itself. This simplification results in a control procedure that may be classified as "direct control" (Narendra and Parthasarathy, Ref. 15).

A modified version of the "direct inverse control" method in neural control (Wasserman, Ref. 8) is used in this study. This approach is attractive in the present context because of its simplicity and straightforward implementation. This approach does not involve gradient based optimization techniques. This method assumes that the plant model is invertible, i.e., a unique dependent (y-axis) value exists for a given independent (x-axis) value. Mathematically, Ref. 8 describes the application of the method in the following manner. Let **F** be the plant model and $\mathbf{F}^{-1}$ its inverse. The combined system includes the inverse model followed by the plant model. A desired system response that is input to $\mathbf{F}^{-1}$ would ideally give the desired system response as the output of **F** since $[(\mathbf{F}^{-1}) \mathbf{F}] = [\mathrm{I}]$. Therefore, to achieve a desired system output, it is only necessary to provide the specific desired system output as an input (Ref. 8). Without feedback, serious questions can arise regarding the robustness of this method (Ref. 13).

The non-feedback direct inverse method is too simple for general purpose applications. In order to apply the direct inverse control method to the present application, modifications and enhancements to the method are necessary. Therefore, the present implementation of this method additionally includes an iterative, feedback loop and a simple halving technique in order to speed up the neural controller's convergence.

Also, the present plant models of interest are nonlinear and hence noninvertible. Consider the following example. For the linear system $y = x$, the resulting variation is a straight line; this system is invertible. For the nonlinear system $y = x^2$, the resulting variation is a parabola. This system is noninvertible because when the axes are inverted, a given "new" x-value does not result in a unique y-value. Recall that the direct inverse control method assumes that the plant model is invertible. The following subsections describe how the presently modified direct inverse control method is applied to rotorcraft dynamics controls applications of interest that involve nonlinear (noninvertible) plant models.

Neural Control Implementation

A block diagram of the present overall neural network control procedure is shown in Fig. 1. For illustrative purposes, this diagram is drawn using the control phase as the relevant feedback parameter. In reference to the plant model, the 2P control phase input is the x-axis value and the vibratory hub loads metric is the nonlinear y-axis value. If the axes were to be simply inverted, the vibratory hub loads metric becomes the x-axis value and the 2P control phase input becomes the y-axis value. In this inverted system, a given metric value does not result in a unique 2P control phase input value. The present control procedure calls only for inverting the axes of the plant model data. This inverted-axes control modeling step "inverts the axes" by using the vibratory hub loads metric as the independent, or input, parameter on the x-axis and the neural network predicted 2P control phase output parameter on the y-axis. Subsequently, an appropriate type of neural network is trained using this inverted-axes data set, thus completing the inverted-axes control modeling step. This step yields a unique x-y relationship corresponding to the input-output relationship of the appropriate neural network. For clarity and brevity, the neural network associated with the inverted-axes control modeling step is referred to as the *inverted neural network for control.*

The present scheme works for the cases considered because the back-propagation network representing the inverted neural network for control is always able to capture the appropriate functional form of a unique y-axis value for any x-axis value. This is achieved in part since the scheme exploits an artifact of the back-propagation neural network: for extrapolative calculations where training data do not exist, the network output is an approximate average of the existing neighboring data points.

Neural Control Procedure Initiation. Since it is possible to perform accurate identification in the present application, the plant is not included in the neural control procedure (“direct” procedure). The present neural control procedure as outlined in Fig. 1 can be initiated in two ways: 1) inputting a desired metric into the inverted neural network for control or 2) inputting a starting value of the control phase input into the plant model. In this study, the neural control procedure was initiated by inputting a control input (2P control phase input or amplitude input) into the plant model.

Neural Control Implementation Details. A simple half-interval calculation which halves the current VHLM metric is used at each iterative step (Fig. 1). The iterative control procedure is terminated when the metric has converged, that is, has reached a global minimum. This approach can also accommodate the application where a desired metric is specified in advance. This paper does not include results for such an application.

Present Application

Data used in this study were obtained from the second U.S./German Individual Blade Control wind tunnel test (Jacklin, et al., Ref. 16). The test article was a four-bladed BO-105 hingeless rotor system fitted with IBC electro-hydraulic actuators. The rotor system was tested in the NASA Ames 40- by 80-Foot Wind Tunnel. Five vibratory hub loads (axial, side, and normal forces, and pitching and rolling moments) obtained from the Rotor Test Apparatus steady/dynamic rotor balance in the fixed system. In this study, all loads were referenced to the rotor hub. These loads were combined into a single metric (by taking the square root of the sum of the squares of each load with equal weighting for each load component). The test condition used in this study is one of simulated descent at an airspeed of approximately 65 knots ($\mu = 0.15$) and $C_T/\sigma = 0.075$. Other test parameters are: $\alpha_s = 2.9$ deg, $\Omega = 425$ RPM, with the hub pitching and rolling moments trimmed to 1600 ft-lb and -350 ft-lb, respectively. This descent condition is equivalent to an approximate 5.6 deg glide slope angle.

The m'th harmonic IBC pitch input for the i'th blade is defined as:

$$\theta_{im} = A_m \sin [m (\psi_i+90 \text{ deg}) + \Phi_m]$$

The present application includes five cases and the results from the first four cases help in directly assessing the neural controller's convergence behavior, robustness, and accuracy. The fifth case considers the performance of a traditional, one-step deterministic controller as "re-applied" to the first two cases. The five cases are outlined below.

Case 1. Basic (Benchmark): This case considers the variation of the metric with 2P control phase input Φ_2 with a constant control amplitude $A_2 = 1$ deg. The IBC2 data base for this case has 12 data points ($\Phi_2 = 0$ to 330 deg at 30 deg intervals). During the neural network plant modeling step, periodicity of the metric is ensured by including an additional 13th point at 360 deg. For this case, the variation of the metric has two minimums (Fig. 2). The neural control procedure is initiated with a 2P control phase input $\Phi_{2N,\,0} = 0$ deg.

Case 2. Starting Point Sensitivity: Using the neural controller from Case 1, a parametric study that assesses the neural controller's robustness is conducted in Case 2. The effect on neural controller convergence of initiating the iterative control procedure using four different starting values of the 2P control phase input (0, 180, 240, and 270 deg) is studied.

Case 3. Reduced Data Base: Here, the benchmark 12-point data set is split into two smaller 6-point data sets based on odd- and even-numbered selections. These two "reduced data base" cases are called Cases 3a and 3b, respectively. Case 3 is important since the results can be used to assess the impact of reducing the number of training data points made available to that part of the neural control procedure which provides an updated estimate of the 2P control phase input.

Case 4. Amplitude Variation: For a constant 2P control phase input ($\Phi_2 = 210$ deg), this case considers the effect of varying 2P control amplitude input A_2 from 0 to 2 deg. Five IBC2 test data points are available, all of which are used for neural network training purposes. In this case, the variation of the metric has one minimum. Case 4 is important because it considers a control amplitude input variation which, in general, along with a control phase input variation represents a complete "control space" (within which the controller must operate). Variations which include both control phase and amplitude inputs are not considered in this study; thus, this case and Case 1 represent a first step towards the consideration of the case with multi-dimensional inputs.

Case 5. One-Step Deterministic Controller: In addition to the results from the present neural controller, results from a traditional, one-step deterministic controller (Johnson, Ref. 17) were calculated for two cases corresponding to Cases 1 and 2. The one-step deterministic controller cases are called Cases 5a and 5b, respectively.

Results

The application of neural network control was conducted using the neural networks package NeuralWorks Pro II/PLUS (version 5.2) by NeuralWare (Ref. 18). The Pro II/PLUS package was installed on an ACER Acros personal computer with an Intel 486DX2/66 central processor. All network applications required approximately two minutes of clock time in order to complete the training step.

Inverted Neural Network for Control

Substantial time and effort were expended in this study to gain insight into the best type of neural network (RBF, back-propagation, etc.) and network architecture (number of hidden layers and number of processing elements) that are needed for an appropriate inverted neural network for control (Fig. 1). This study was also conducted to ensure that the final, selected neural network is the simplest network that can be used as the inverted neural network for control. The network input is the metric VHLM and the network output is either the predicted 2P control phase input Φ_{2N} (Cases 1 to 3) or the 2P control amplitude input A_{2N} (Case 4). The inverted neural network for control is trained with 12 input data points in Case 1 and Case 2, six points in Case 3, and five points in Case 4 . Overall, the best inverted neural network for control can be determined only after the neural control results for Cases 1 to 4 are computed. At present, the overall conclusion is that a 1-2-3-1 back-propagation neural network can be successfully used as the inverted neural network for control for all the neural control cases considered in this study.

Results for the neural control cases are presented in the following sequence: plant model results, inverted neural network for control results, and finally, the neural controller's convergence sequence.

Case 1. Basic (Benchmark)

Plant Model. Figure 2 shows the vibratory hub loads metric's variation with the 2P control phase input Φ_2 for a constant 2P control amplitude A_2 = 1 deg. In the figure, the solid squares represent metric values derived from the IBC2 test data. The baseline metric (no IBC, A_2 = 0 deg) is 578 and the IBC2 test-based minimum metric is 211 (at Φ_2 = 240 deg), Fig. 2. Figure 2 also shows the plant modeling results obtained from the 1-12-4-1 RBF neural network that was trained using 13 input data points. The 1-12-4-1 RBF network was trained for 10,000 iterations; the final RMS error was 0.0389. This 1-12-4-1 RBF neural network is also used as the plant model in Cases 2 and 3.

Inverted Neural Network for Control. The outputs from the 1-2-3-1 back-propagation neural network used as the inverted neural network for control are shown in Figs. 3a and 3b. This neural network was trained with 12 IBC2 test data points. In training this network, the data set axes are inverted compared to the plant model (see subsection on "Neural Control Implementation"). The 1-2-3-1 back-propagation network was trained for 20,000 iterations; the final RMS error was 0.2339. Figure 3a shows the network output for a metric range of 0 to 10,000. As can be seen from Fig. 3a, the network output is almost constant for large values of the metric ("large-metric behavior"). This type of large-metric behavior is desirable and appears to depend on the type of network. Figure 3b is the same as Fig. 3a except that the x-axis plotting scale is 0 to 2000; the neural computations are exactly the same for the two plots. For future reference, the network output behavior for metric values within this scale is called "small-metric behavior."

Figure 3b shows that the shape of the predicted network output appears to be close to that of a sigmoid function. This could be an attribute of the present inverted neural network for control that is due to an inherent property of the specific type of neural network that is being used, namely, the back-propagation neural network. As noted earlier, several types of neural networks (RBF, back-propagation, etc.) were tried out in an attempt to make the present neural control scheme work. As described in the following section, in general, the above mentioned behavior of the 1-2-3-1 back-propagation neural network is sufficient to make the neural control scheme successful.

Neural Controller Convergence. In the present case, the controller cycle is initiated with a 1 deg 2P control amplitude with phase $\Phi_{2N,\,0}$ = 0 deg where the first subscript refers to 2P control phase input, the second subscript refers to the neural (analytical) controller, and the third subscript refers to the iteration index. Figures 4a and 4b show results from the present neural network controller. The neural controller produces a converged global minimum metric (VHLM = 195) in four iterations (Fig. 4a). The corresponding converged optimal 2P control phase input ($\Phi_{2N,\,4}$) predicted by the neural controller is 261 deg (Fig. 4b).

For other applications, if the as-presently-formulated inverted-axes control modeling step is not valid, one may have to resort to a radically different approach. This approach may involve deriving special, new neural networks customized for individual applications and possessing the extrapolative and interpolative characteristics required for a good inverted neural network for control. Nevertheless, the results from this one example are very promising.

Case 2. Starting Point Sensitivity

In Case 2, the starting values of the 2P control phase input $\Phi_{2N,\,0}$ are varied and specified as 0 deg (same as in Case 1), 180, 240, and 270 deg. These values were selected to give the neural controller freedom in determining the best solution. The Case 1 2P control phase input starting value (0 deg) may be the worst starting point for the neural controller since the optimal 2P control phase input from the IBC2 test is 240 deg, and the controller has a "long way to go" before reaching the optimal solution. The plant model and the inverted neural network for control are the same as those used in Case 1.

Neural Controller Convergence. Figures 5a and 5b show the neural controller convergence sequences for the metric and the 2P control phase input for the four starting values. The table below shows the corresponding numerical values in which the halving step is also noted. The identified halved metric is the input to the inverted neural network for control at the next iterative cycle. The 0 deg starting point subcase ($\Phi_{2N,\,0} = 0$ deg) required four iterations to converge whereas the other three subcases required three iterations to converge. The metric converged to a value of 195 for all four subcases (Fig. 5a). The corresponding predicted, converged 2P control phase input is 261 deg for all four subcases (Fig. 5b). These results appear to indicate that the present neural controller is insensitive to a starting point for the current study.

Table 1. Starting Point Sensitivity, Neural Control

(PM: plant model, INNC: inverted neural network for control)

Iteration No.	2P control phase input into PM	Identified/ Halved Metric	2P control phase input from INNC deg
Starting Point = 0 deg			
0	0	1130/565	217
1	217	443/222	256
2	256	198/99	261
3	261	195/98	261
4	261	195	
Starting Point = 180 deg			
0	180	507/254	254
1	254	200/100	261
2	261	195/98	261
3	261	195	
Starting Point = 240 deg			
0	240	261/130	260
1	260	195/98	261
2	261	195/98	261
3	261	195	
Starting Point = 270 deg			
0	270	209/105	260
1	260	195/98	261
2	261	195/98	261
3	261	195	

Case 3. Reduced Data Base

This case has two subcases, Cases 3a and 3b. During the inverted neural network for control modeling step, these two subcases use two different "reduced data base" 6-point training data sets. In Case 3a, the six odd-numbered training points from the original 12-point training data set (IBC2 test data) are used; similarly, in Case 3b the six even-numbered points are used. The plant models for Cases 3a and 3b are the same as those used in Case 1 (13-point training data set).

Case 3a. Odd-Numbered, Six Point Data Base

Inverted Neural Network for Control: Figures 6a and 6b show the outputs of the 1-2-3-1 back-propagation neural network used as the inverted neural network for control. This network was trained using the odd-numbered six point data base. The 1-2-3-1 back-propagation network was trained for 20,000 iterations; the final RMS error was 0.1802. The large-metric behavior of this 1-2-3-1 back-propagation network (Fig. 6a) has the desirable constant trend noted earlier in Case 1. The Case 3a large-metric behavior exhibits a small negative value and hence the y-axis in Fig. 6a does not start from zero. This type of large-metric behavior of the 1-2-3-1 back-propagation network does not affect the neural control procedure. Figure 6b (small-metric behavior) shows that the appropriate functional form for the output of the inverted neural network for control can be generated from six, odd-numbered training points instead of the baseline twelve points (Fig. 3b).

Neural Controller Convergence: The controller iterations were initiated with a 1 deg 2P control amplitude with a starting 2P control phase $\Phi_{2N,\,0} = 0$ deg. Figure 7a shows that the present neural controller globally converges in five iterations to a value of 224. Figure 7b shows that the corresponding converged optimal 2P control phase input $\Phi_{2N,\,5}$ predicted by the neural controller is 275 deg. This compares favorably to a converged VHLM of 195 at a 2P control phase input of 261 deg from Case 2.

Case 3b. Even-Numbered, Six Point Data Base

Inverted Neural Network for Control: Figures 8a and 8b show the outputs of the 1-2-3-1 back-propagation neural network used as the inverted neural network for control. This network is trained using the even-numbered six point data base. The 1-2-3-1 back-propagation network was trained for 20,000 iterations; the final RMS error was 0.2617. The large-metric behavior of this 1-2-3-1 back-propagation network has the desirable constant trend (Fig. 8a). Figure 8b (small-metric behavior) shows that the appropriate functional form can be generated from six, even-numbered training points instead of the baseline twelve points (Fig 3b).

Neural Controller Convergence: The controller iterations were initiated with a 1 deg 2P control amplitude with a starting 2P control phase $\Phi_{2N,\,0} = 0$ deg. Figure 9a shows that the present neural controller globally converges in five iterations to a metric of 207. (even-numbered six point case). Figure 9b shows that the corresponding converged optimal 2P control phase input $\Phi_{2N,\,5}$ predicted by the neural controller is 269 deg .

The preceding results from Cases 1 to 3 indicate that the inverted neural network for control modeling step is sufficiently robust and accurate for the present neural control purposes.

Case 4. Amplitude Variation

Plant Model. Figure 10 shows the vibratory hub loads metric variation with 2P control amplitude input A_2 for a constant 2P control phase input $\Phi_2 = 210$ deg. In the figure, the solid squares represent metric values derived from the 5-point IBC2 test data base. The IBC2 test-based minimum metric is 328 (at $A_2 = 0.5$ deg). Figure 10 also shows the plant modeling results obtained from a 1-5-1 RBF neural network that was trained using the five points. The 1-5-1 RBF network was trained for 22,000 iterations; the final RMS error was 0.0137.

Inverted Neural Network for Control. Figure 11 shows the small-metric output of the 1-2-3-1 back-propagation neural network (used as the inverted neural network for control). The 1-2-3-1 back-propagation network was trained for 35,000 iterations; the final RMS error was 0.2589. In Case 4, a larger number of iterations was required to train the 1-2-3-1 back-propagation neural network compared to training the 1-2-3-1 back-propagation networks in Cases 1 and 3 (35,000 compared to 20,000). During the training process for the 1-2-3-1 back-propagation neural network in Case 4, the network's RMS error variation with the number of iterations took longer to reach a steady state. The final output in Case 4 has the appropriate functional form (Fig. 11), i.e., the shape of the predicted network output appears to be close to that of a sigmoid function. Though not included in this paper, the large-metric trend from this 1-2-3-1 back-propagation neural network was found to be around $A_{2N} = 2$ deg.

Neural Controller Convergence. For this one-minimum case, the neural controller converged in three iterations (iterations initiated with $A_{2N,\,0} = 0$ deg) to a metric of 346 (Fig. 12a). The corresponding predicted 2P control amplitude $A_{2N,\,3}$ is 0.57 deg (Fig. 12b). Comparing this result with Fig. 10, the neural controller has acceptably converged to a minimum VHLM.

Case 5. One-Step Deterministic Controller

Using linear transfer-function identification theory and the quadratic performance function formulation (Johnson, Ref. 17), the performance of a one-step deterministic controller is assessed in Case 5.

Linear Transfer-Function Matrix Identification. The vector of responses (vibratory hub loads) is assumed to vary linearly with the control input as given below:

$$\{ z \} = [T] \{ \theta \}$$

Here, $\{ z \}$ is the response vector, $[T]$ is the linear transfer-function matrix, and $\{\theta \}$ is the control input vector. In the present case, the vectors $\{ z \}$ and $\{ \theta \}$ are made up of experimental vibratory hub load components and 2P control sine and cosine phase inputs, respectively.

The present one-step deterministic controller applications include ten vibratory hub load components (sine and cosine components of five hub load components) and twelve measurements (2P control phase input varying from 0 to 330 deg in 30 deg increments). Separate single harmonic sine and cosine least-square fits are used to determine the elements of the T-matrix.

Optimal One-Step Deterministic Control Input. The optimal control input is calculated based on a quadratic performance function with all responses equally weighted. For the present one-step deterministic controller, the optimal control input vector is calculated using Equation A1 (Appendix). In Equation A1, the 2 x 1 vector of optimal control inputs $\{ \theta^* \}$ consists of the sine and cosine components from which the 2P control optimal phase input is calculated. In the following, the subscript "s" refers to the starting condition for the one-step deterministic controller.

Case 5a. Baseline Results: In Case 5a, the starting condition is the baseline condition: $\{ \theta \}_s = \{ 0 \}$ (corresponding to $A_2 = 0$ deg), and with the starting response vector $\{ z \}_s$ taken as the baseline experimental hub loads vector (ten vibratory hub load sine and cosine components). Here, the metric is calculated using the plant model of Case 1 and requiring that the 2P control amplitude input is 1 deg. The one-step deterministic controller predicts a metric of 281 and an optimal 2P control phase input of 237 deg. Comparing this one-step deterministic controller result to the neural control result of Cases 1 and 2 (converged metric is 195, corresponding 2P control phase input is 261 deg), the conclusion is that the two control methods are comparable. Note that the starting conditions are different for these two methods: the one-step deterministic controller starts out with the baseline condition whereas the neural controller starts out with a 2P control phase input (Cases 1 and 2).

Case 5b. Starting Point Sensitivity: In Case 5b, each of the four sets of $\{ \theta \}_S$ and $\{ z\}_S$ vectors is separately determined by the following four 2P control phase input values: 0, 180, 240, and 270 deg, each with A_2 = 1 deg. The control input vector $\{ \theta \}_S$ is directly obtained from the 2P control phase input under consideration, and the starting response vector $\{ z \}_S$ is taken as the experimental hub loads vector corresponding to the particular 2P control phase input under consideration. The table below shows the results of the sensitivity study which evaluates the performance of the one-step deterministic controller performance for the four starting points.

Table 2. Starting Point Sensitivity, One-Step Deterministic Control, A_2 = 1 deg

Starting 2P control phase input, deg	Predicted Metric	Predicted 2P control phase input, deg
0	196	262
180	207	269
240	205	268
270	337	230

The above "starting point sensitivity" one-step deterministic controller results can be compared to those from the neural controller (Case 2). This limited-scope comparison shows that the two control methods are roughly comparable, with neural control being slightly more robust. In particular, the one-step deterministic controller yields relatively poor results for the 270 deg starting condition (VHLM = 337) as compared to the corresponding neural network result (Table 1, VHLM = 195).

Concluding Remarks

The application of neural networks to rotorcraft dynamics control is still relatively new. The objective of the present study was to develop a robust neural network based controller to minimize vibratory hub loads. A metric consisting of five vibratory hub load components (with their sine and cosine components) obtained from a wind tunnel test of a four-bladed rotor with individual blade control was used to characterize the hub loads. The present single-input, single-output neural network control procedure was bound by the following ground rules: the controller must converge in six iterations or less and gradient based optimization techniques must not be used. A simple and straightforward iterative procedure for neural control was applied. Two neural networks were used in the procedure requiring a plant model (using a radial-basis function neural network) and an inverted neural network for control model (using a back-propagation neural network). The training of each network required approximately two minutes of clock time. A simple half-interval calculation which halves the metric is used in order to speed up convergence.

The neural network controller was successful in achieving convergence within a limited number of iterations while being robust and computationally efficient. Specific findings are given below.

1. Radial-basis function neural networks were successfully used for the plant modeling step in the present neural control procedure.

2. A simple back-propagation neural network can be used as the inverted neural network for control. This back-propagation neural network has two hidden layers with the first and second hidden layers having two and three processing elements, respectively.

3. Four neural control application cases were considered in this study. The first three cases involved variation in the individual blade control 2P control phase input (2P control amplitude fixed) and the fourth case involved a variation in the 2P control amplitude input (2P control phase fixed). The present findings are:

    a. The present neural network control procedure successfully achieved the objective of converging to the global minimum in all four cases within six iterations without using gradient based optimization techniques.

    b. Results from the first three application cases showed that the present neural control procedure is robust.

4. A limited-scope comparison of the results from the present neural control procedure with those from a one-step deterministic controller showed that the two control methods were roughly comparable, with neural control being slightly more robust.

Acknowledgments

The author wishes to thank Donald Soloway and Chuck Jorgensen (NeuroEngineering Group, Computational Sciences Division, NASA Ames) for their invaluable help.

References

1. Kottapalli, S., Swanson, S., LeMasurier, P., and Wang, J., "Full-Scale Higher Harmonic Control Research to Reduce Hub Loads and Noise," American Helicopter Society 49th Annual Forum, St. Louis, Missouri, May 1993.

2. Miao, J., Kottapalli, S.B.R., and Frye, H.M., "Flight Demonstration of Higher Harmonic Control (HHC) on S-76," American Helicopter Society 42nd Annual Forum, Washington, D.C., June 1986.

3. O'Leary, J.J., Kottapalli, S.B.R., and Davis, M., "Adaptation of a Modern Medium Helicopter (Sikorsky S-76) to Higher Harmonic Control," 2nd Decennial Specialists Meeting on Rotorcraft Dynamics, NASA Ames Research Center, Moffett Field, California, November 1984.

4. Kottapalli, S., Abrego, A., and Jacklin, S., "Application of Neural Networks to Model and Predict Rotorcraft Hub Loads," American Helicopter Society Second International Aeromechanics Specialists Conference, Bridgeport, Connecticut, October, 1995.

5. Kottapalli, S., Abrego, A., and Jacklin, S., "Multiple-Input, Multiple-Output Application of Neural Networks to Model and Predict Rotorcraft Hub Loads," Sixth International Workshop on Dynamics and Aeroelastic Stability of Rotorcraft Systems, Los Angeles, California, November 1995.

6. Handbook of Intelligent Control (Neural, Fuzzy, and Adaptive Approaches), edited by White, D. A. and Sofge, D.A., Van Nostrand Reinhold, New York, 1992.

7. Neural Networks for Control, edited by Milller, W.T., Sutton, R.S., and Werbos, P.J., The MIT Press, Cambridge, Massachusetts, 1990.

8. Advanced Methods in Neural Computing, Wasserman, P.D., Van Nostrand Reinhold, New York, 1993.

9. Werbos, P.J., "Neurocontrol and Supervised Learning: An Overview and Evaluation," Handbook of Intelligent Control (Neural, Fuzzy, and Adaptive Approaches), edited by White, D. A. and Sofge, D.A., Van Nostrand Reinhold, New York, 1992.

10. Neuro-Control and its Applications, Omatu, S., Khalid, M., and Yusof, R., Advances in Industrial Control, Springer, 1996.

11. Neural Networks for Identification, Prediction and Control, Pham, D.T. and Liu, X., Springer, 1995.

12. Psaltis, D., Sideris, A., and Yamamura, A.A., "A Multilayered Neural Network Controller," *IEEE Control Systems Magazine*, April 1988.

13. Hunt, K.J., Sbarbaro, D., Zbokowski, R, and Gawthrop, P.J., "Neural Networks for Control Systems-A Survey," *Automatica,* Vol. 28, No. 6, 1992, pp. 1083-1112.

14. Faller, W.E., Schreck, S.J., and Lutges, M.W., "Real-Time Prediction and Control of Three-Dimensional Unsteady Separated Flow Fields Using Neural Networks," AIAA Paper 94-0532, 32nd Aerospace Sciences Meeting & Exhibit, Reno, Nevada, January 1994.

15. Narendra, K.S. and Parthasarathy, K., "Identification and Control of Dynamical Systems Using Neural Networks," *IEEE Transactions on Neural Networks*, Vol. 1 (1), March 1990, pp. 4-27.

16. Jacklin, S., Blaas, A., Kube, R., and Teves, D., "Reduction of Helicopter BVI Noise, Vibration, and Power Consumption through Individual Blade Control," American Helicopter Society 51st Annual Forum, Ft. Worth, Texas, May 1995.

17. Johnson, W., "Self-Tuning Regulators for Multicyclic Control of Helicopter Vibration," NASA Technical Paper 1996, March 1982.

18. NeuralWorks Manuals:
    a. Reference Guide
    b. Neural Computing
    c. Using NeuralWorks

NeuralWare, Inc., Pittsburgh, Pennsylvania, 1995.

Appendix

One-Step Deterministic Controller Equations

Optimal Control Input

The optimal one-step deterministic controller control input vector used in the present study is derived as follows. From Johnson (Ref. 17), the optimal control input is:

$$\{ \theta^* \}_n = [C] (\{ z \}_{n-1} - [T] \{ \theta \}_{n-1})$$

where the subscript "n" refers to a time step and the controller gain [C] is given by:

$$[C] = - ([T]^T [T])^{-1} [T]^T$$

Substituting for [C], the optimal control input is expressed as:

$$\{ \theta^* \}_n = - ([T]^T [T])^{-1} [T]^T (\{ z \}_{n-1} - [T] \{ \theta \}_{n-1})$$

The present, one-step deterministic control study is an off-line type of application, and the preceding equation is presently used in the following manner in order to calculate the optimal control input vector:

$$\{ \theta^* \} = - ([T]^T [T])^{-1} [T]^T (\{ z \}_s - [T] \{ \theta \}_s) \qquad \text{(A1)}$$

where the subscript "s" refers to the one-step starting condition.

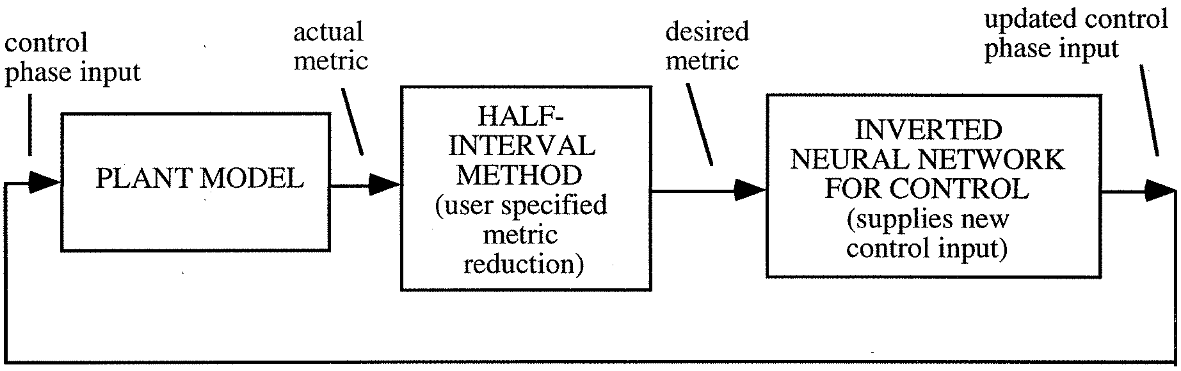

Fig. 1. Overall neural network control procedure.

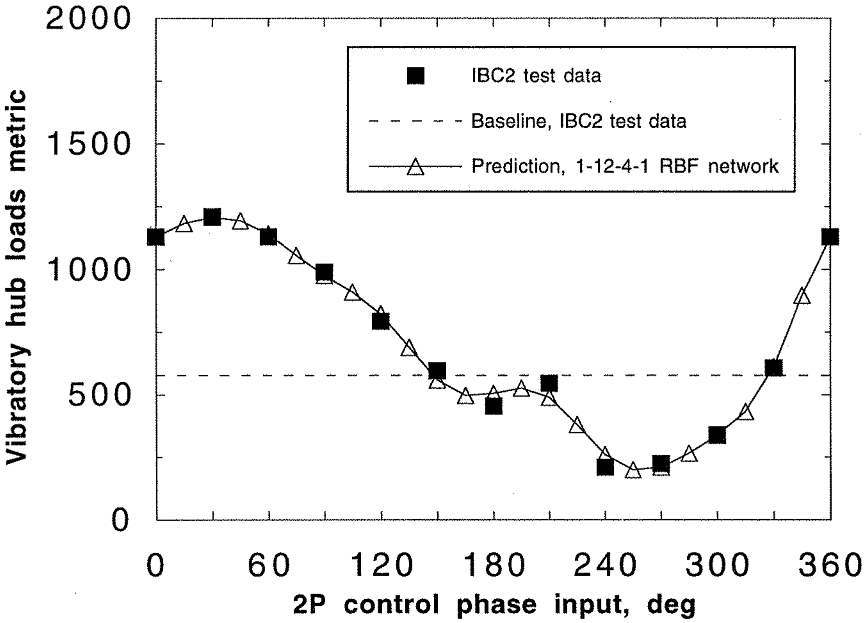

Fig. 2. Experimentally derived metric and identification (plant modeling) by neural network for Cases 1 to 3.

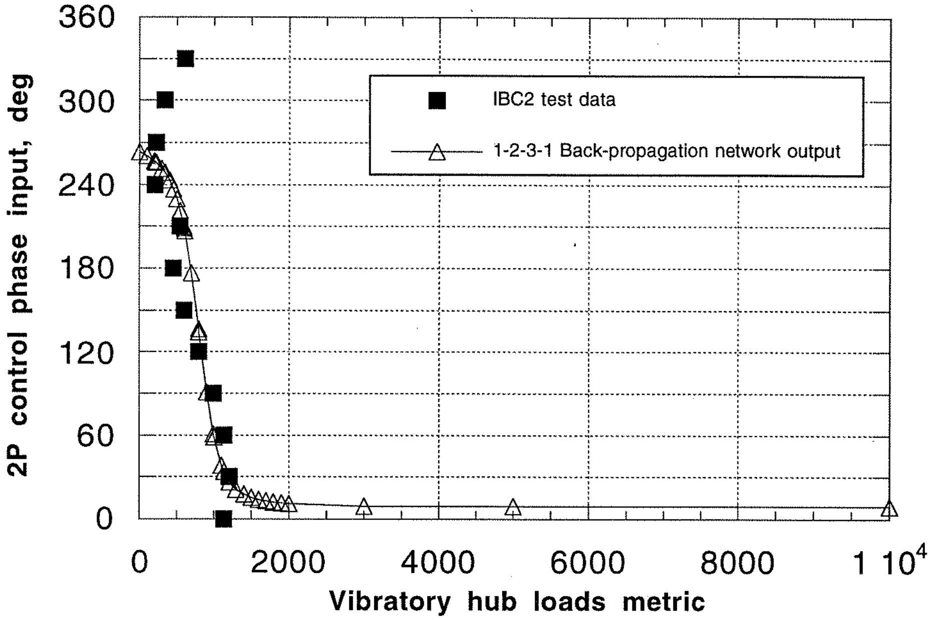

Fig. 3a. **Case 1 (Basic) Benchmark, output of inverted neural network for control, 12 training points, large-metric scale.**

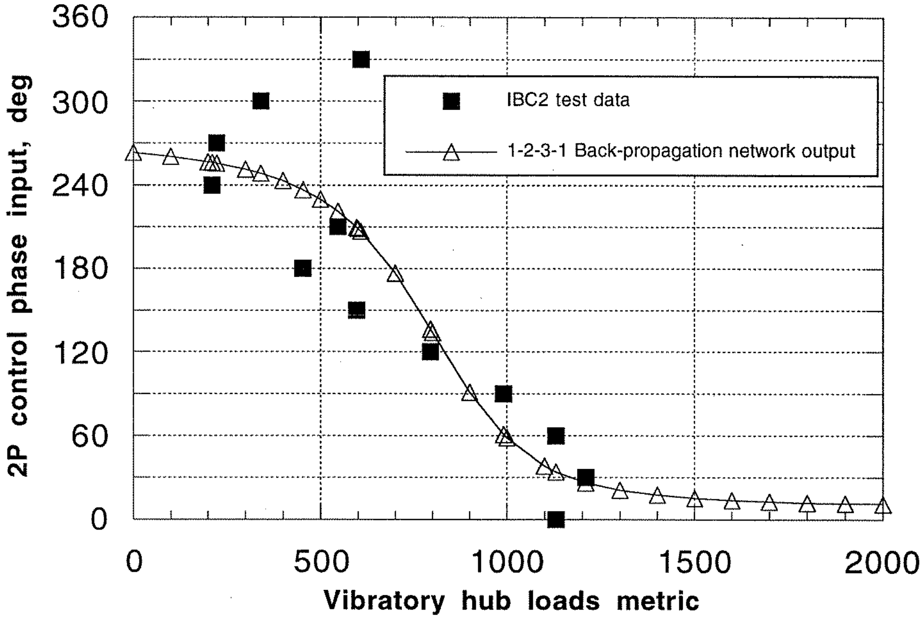

Fig. 3b. **Case 1 (Basic) Benchmark, output of inverted neural network for control, 12 training points, small-metric scale.**

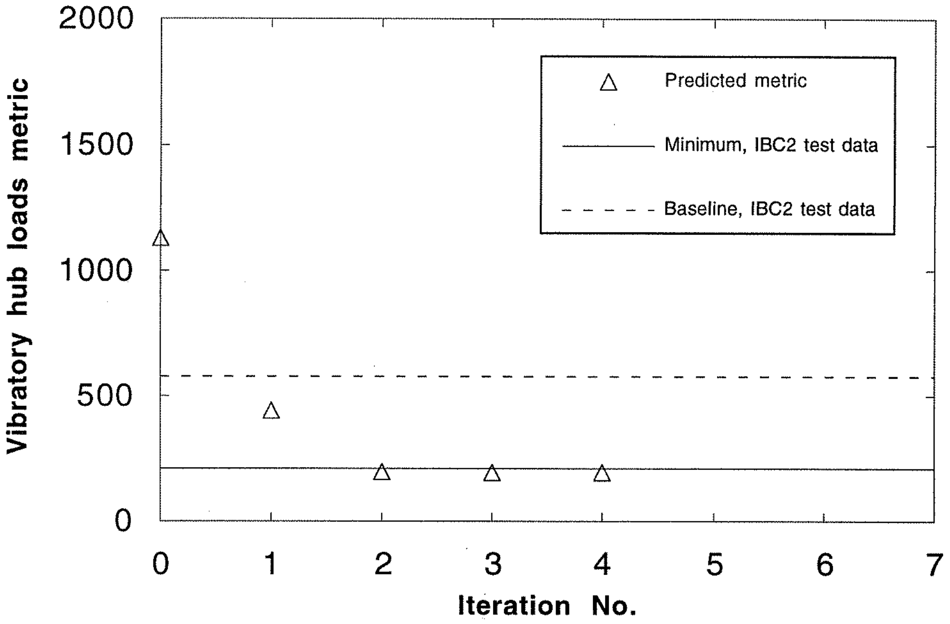

Fig. 4a. Case 1 (Basic) Benchmark, neural control of hub loads metric.

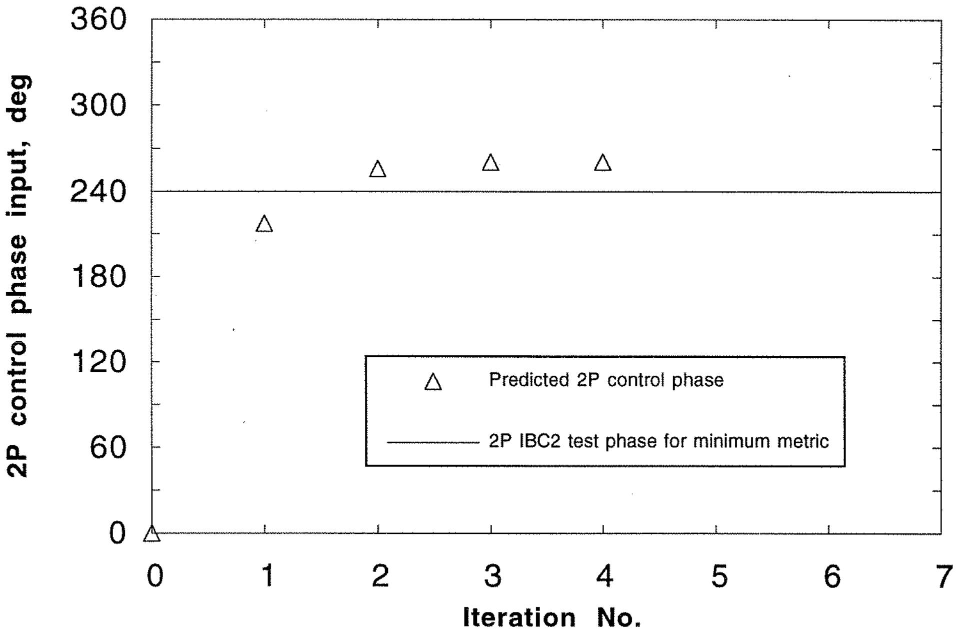

Fig. 4b. Case 1 (Basic) Benchmark, convergence of 2P control phase input (neural control, Fig. 4a shows corresponding metric).

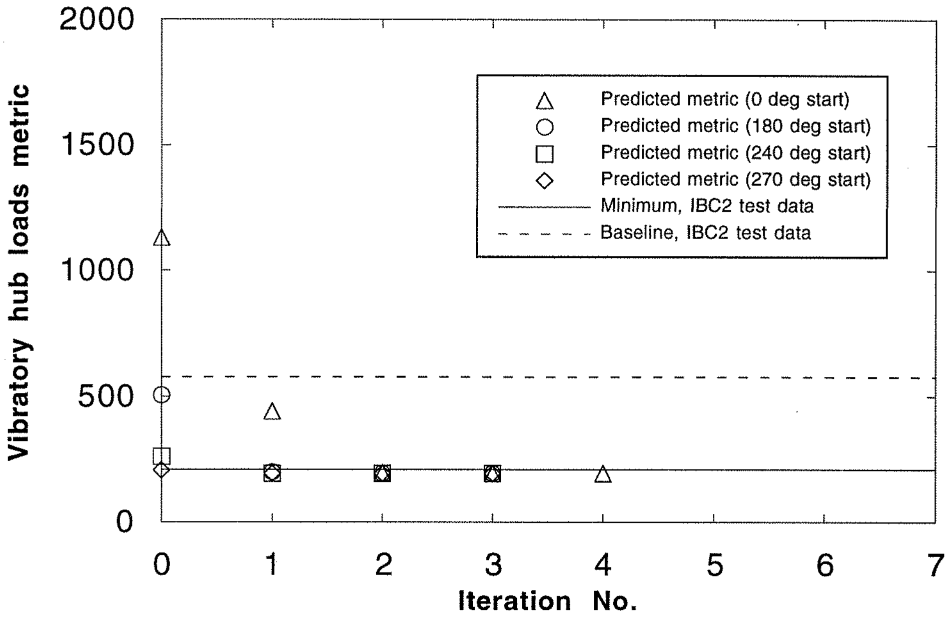

Fig. 5a. Case 2 (Starting Point Sensitivity), convergence of hub loads metric (neural control).

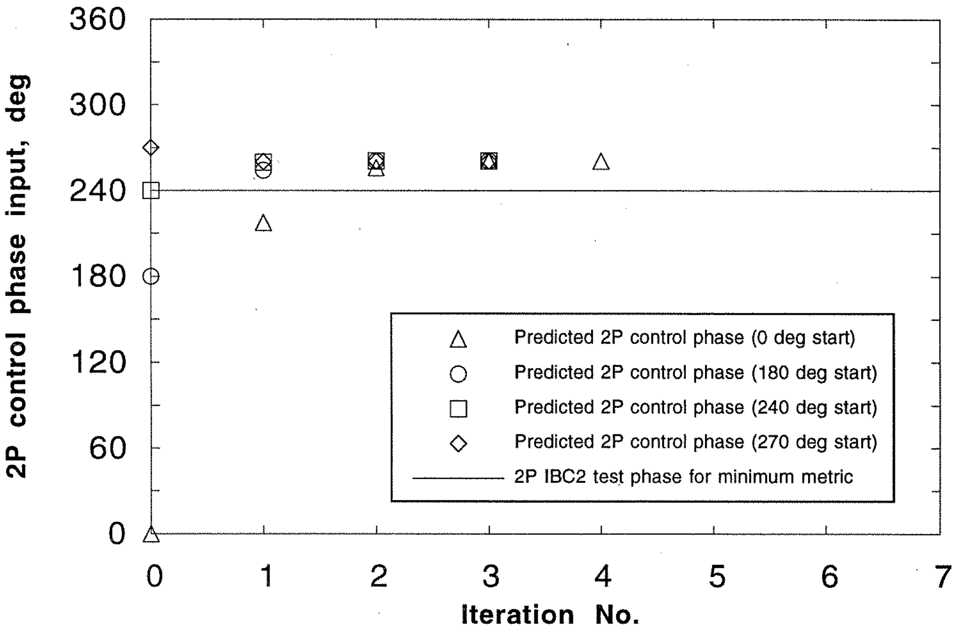

Fig. 5b. Case 2 (Starting Point Sensitivity), convergence of 2P control phase input (neural control, Fig. 5a shows corresponding metrics).

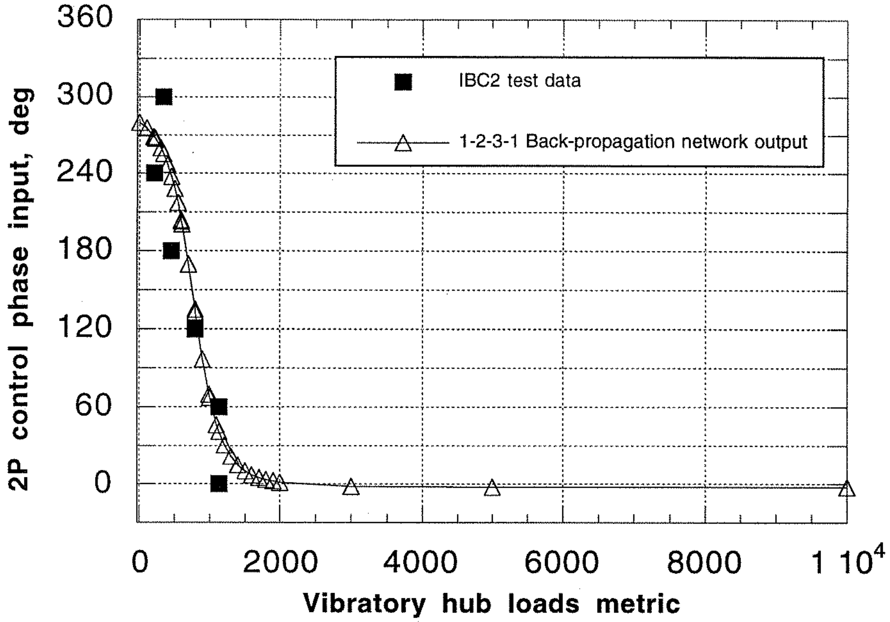

Fig. 6a. Case 3a (Odd-Numbered, Six Point Data Base), output of inverted neural network for control, large-metric scale.

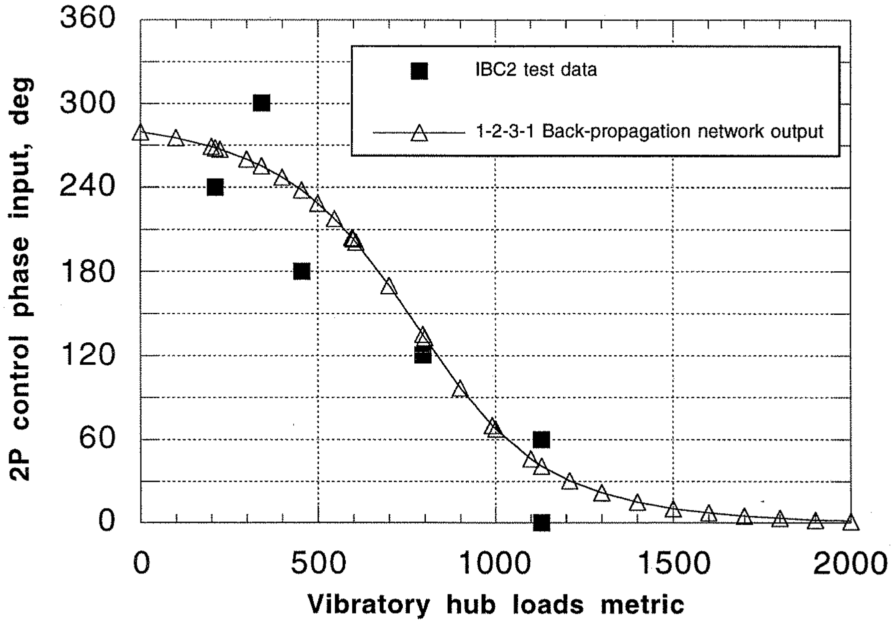

Fig. 6b. Case 3a (Odd-Numbered, Six Point Data Base), output of inverted neural network for control, small-metric scale.

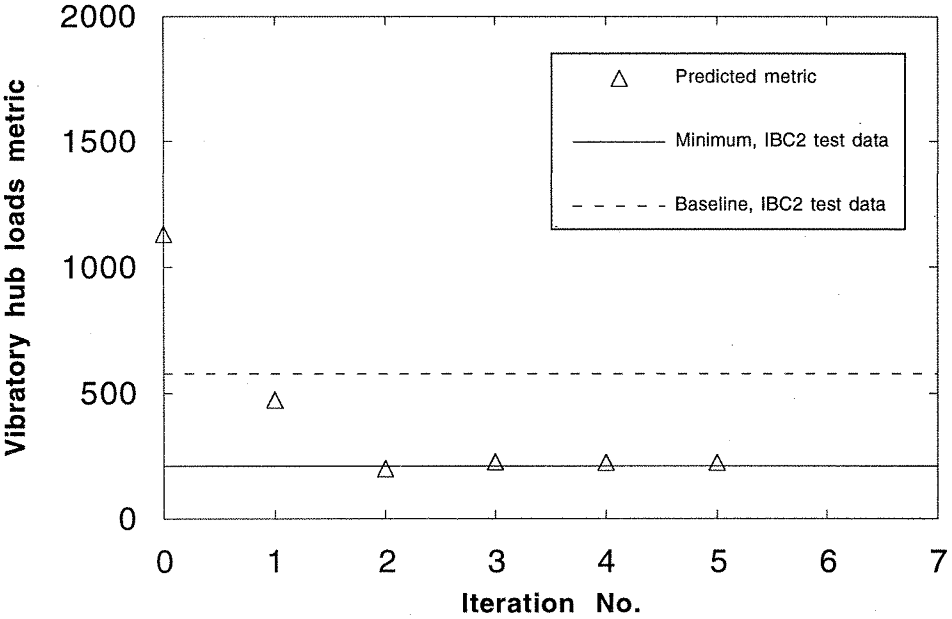

Fig. 7a. Case 3a (Odd-Numbered, Six Point Data Base), neural control of hub loads metric.

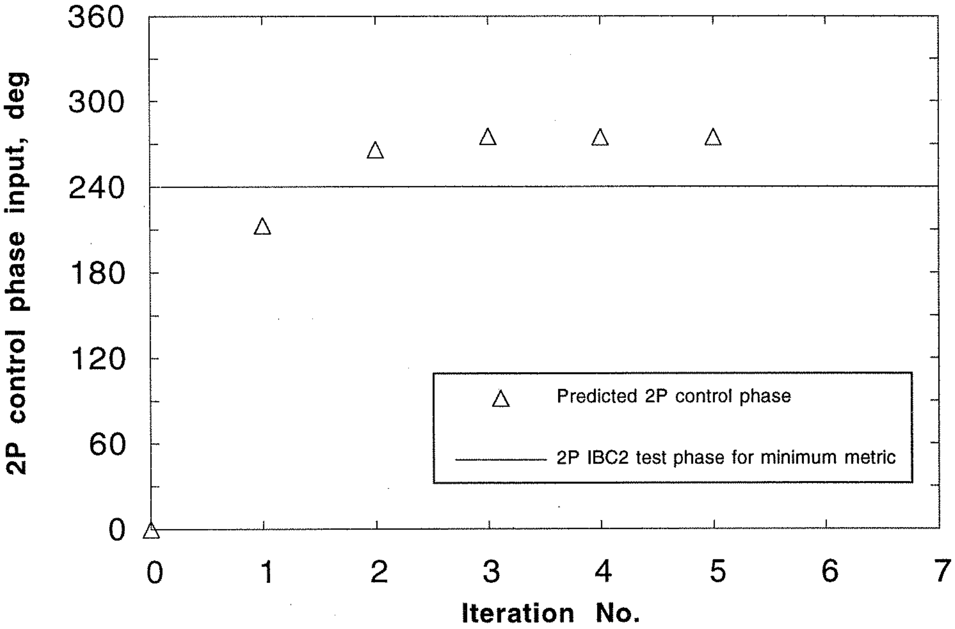

Fig. 7b. Case 3a (Odd-Numbered, Six Point Data Base), convergence of 2P control phase input (neural control, Fig. 7a shows corresponding metric).

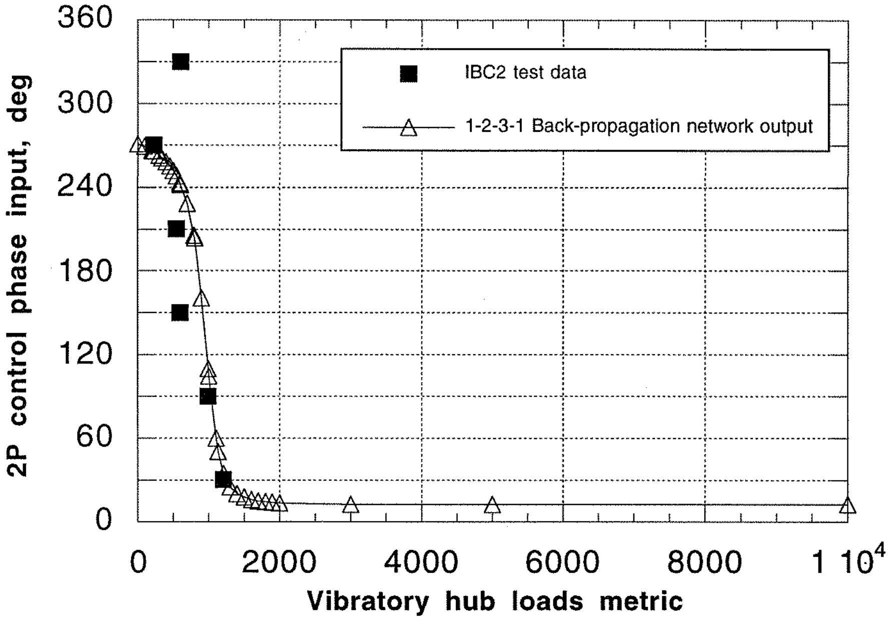

Fig. 8a. Case 3b (Even-Numbered, Six Point Data Base), output of inverted neural network for control, large-metric scale.

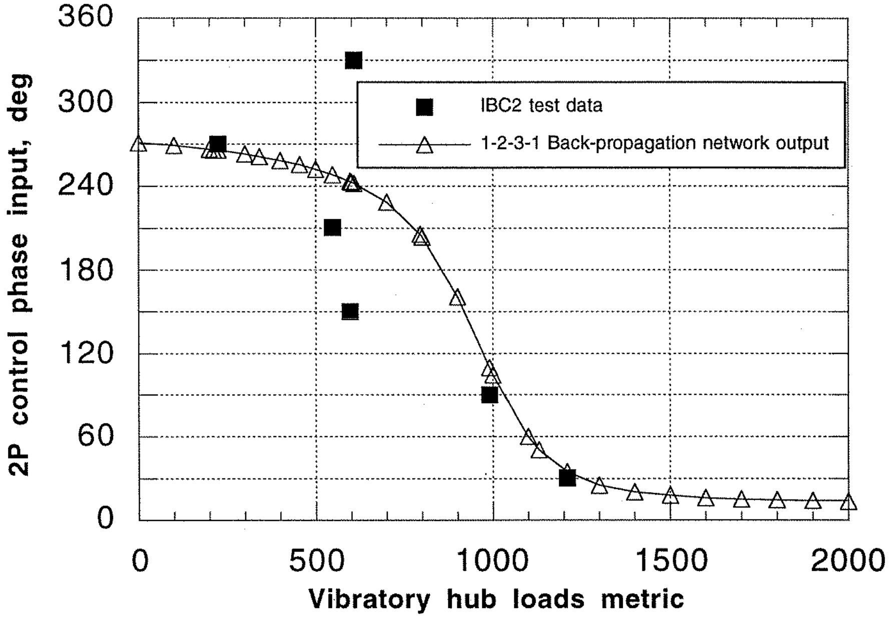

Fig. 8b. Case 3b (Even-Numbered, Six Point Data Base), output of inverted neural network for control, small-metric scale.

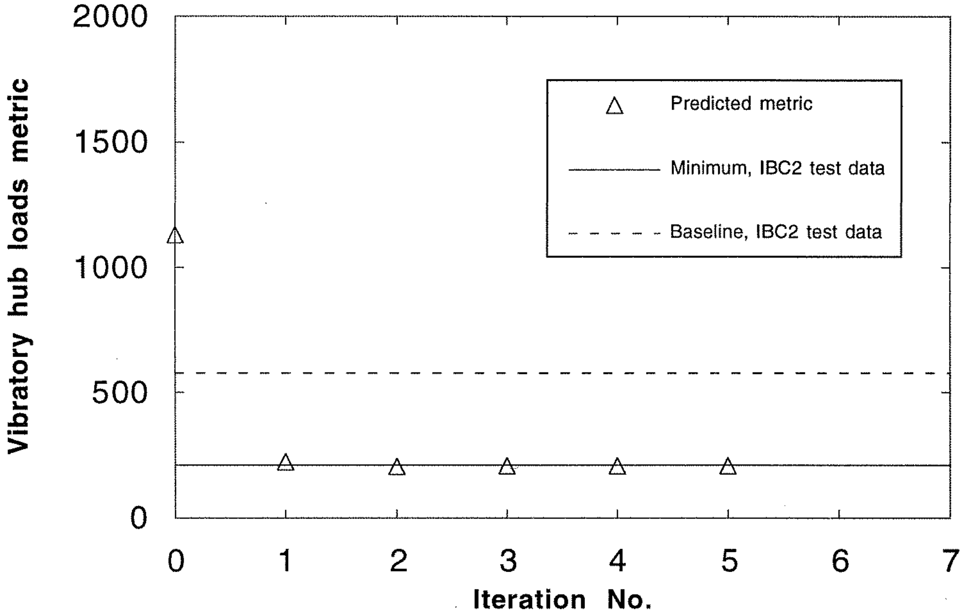

Fig. 9a. Case 3b (Even-Numbered, Six Point Data Base), neural control of hub loads metric.

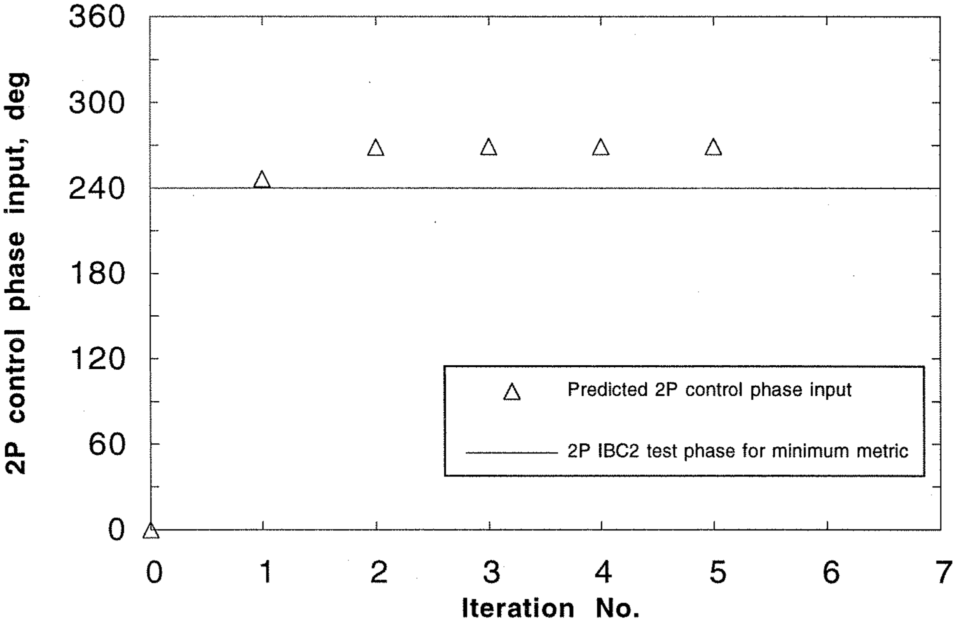

Fig. 9b. Case 3b (Even-Numbered, Six Point Data Base), convergence of 2P control phase input (neural control, Fig. 9a shows corresponding metric).

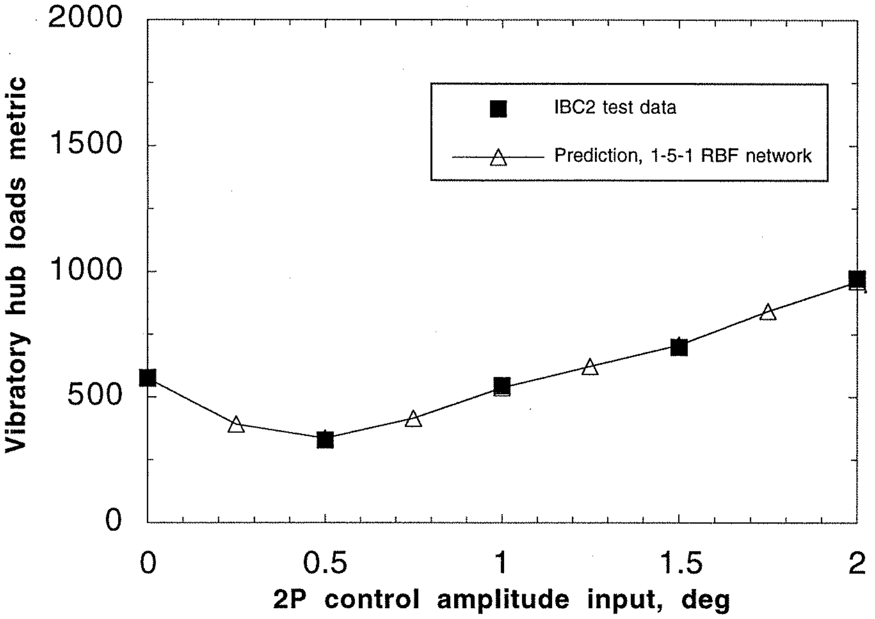

Fig. 10. Experimentally derived metric for amplitude variation case (Case 4) and identification (plant modeling) by neural network.

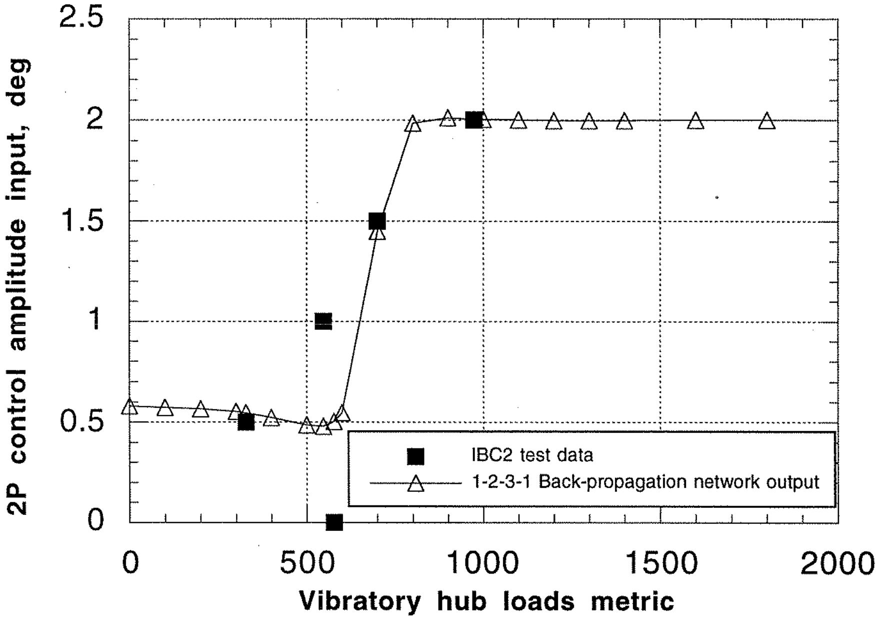

Fig. 11. Case 4 (Amplitude Variation), output of inverted neural network for control, five training points, small-metric scale.

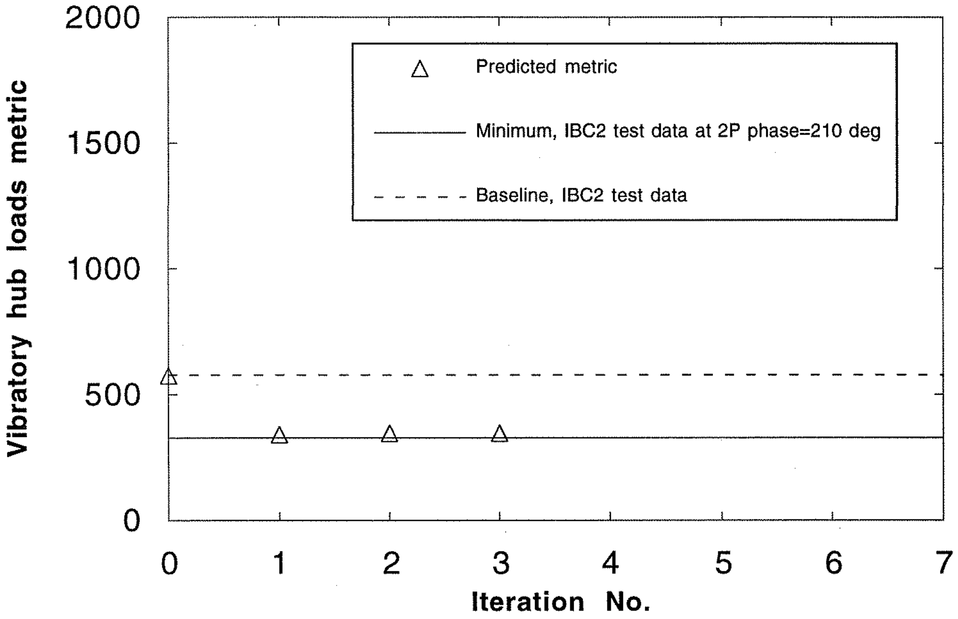

Fig. 12a. Case 4 (Amplitude Variation), neural control of hub loads.

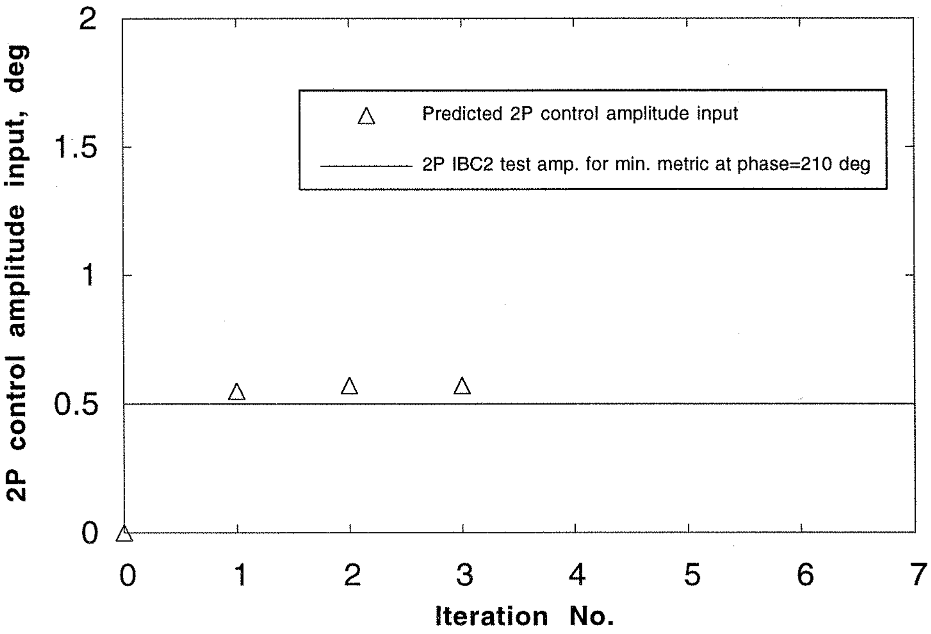

Fig. 12b. Case 4 (Amplitude Variation) convergence of 2P control amplitude input (neural control, Fig. 12a shows corresponding metric).