Active Control of Vibration Using a Neural Network

Scott D. Snyder, Associate Member, IEEE, and Nobuo Tanaka

Abstract—Feedforward control of sound and vibration using a neural network-based control system is considered, with the aim of deriving an architecture/algorithm combination which is capable of supplanting the commonly used finite impulse response filter/filtered-x least mean square (LMS) linear arrangement for certain nonlinear problems. An adaptive algorithm is derived which enables a stable adaptation of the neural controller for this purpose, while providing the capacity to maintain causality within the control scheme. The algorithm is shown to be simply a generalization of the linear filtered-x LMS algorithm. Experiments are undertaken which demonstrate the utility of the proposed arrangement, showing that it performs as well as a linear control system for a linear control problem and better for a nonlinear control problem. The experiments also lead to the conclusion that more work is required to improve the predictability and consistency of the performance before the neural network controller becomes a practical alternative to the current linear feedforward systems.

I. INTRODUCTION

THE active control of sound and vibration using feedforward control techniques has been the topic of much research in recent years [1]. Some of the research programs are now coming to fruition: for example, active noise control systems for air handling ducts and for automotive interiors (from Nissan, in a limited number of cars for the Japanese market) are commercially available. The majority of the present control systems, both research and commercial, use modified adaptive signal processing algorithms and architectures to derive the control inputs to the structural/acoustic system. The most common form of adaptive algorithm/architecture combination is a transversal filter-based controller (FIR (finite impulse response) or IIR filter) used to train a gradient descent-type algorithm, the utility of which has been widely demonstrated over the past decade. There is, however, one potential drawback to this arrangement: by design, such a system is limited to linear control problems. In other words, the control input signal, as well as the associated measured error signal used in the adaptation process, must be linear functions of the reference signal used by the adaptive filter to derive the control signal. This restriction has not proved to be too constraining in the past, as the vast majority of structural/acoustic system arrangements targeted for feedforward active control do represent linear control problems. There are, however, several instances where a nonlinear adaptive control scheme may prove superior to a linear one. These include cases where the control actuator exhibits nonlinear response characteristics, such as: when it excites both the frequency of interest and its associated harmonics, and it is desired to have the control system compensate for, and overcome, this detriment (a common complaint when using surface-mounted piezo-electric actuators to actively control vibration or structural-acoustic radiation); where a tonal reference signal is provided to the control system, such as from a shaft encoder on a piece of rotating machinery which is responsible for the primary noise disturbance, but where the actual primary disturbance to be controlled comprises both the tone and harmonics; and where it is desired to directly minimize some power or intensity-based error signal, such that the error signal will be twice the frequency of the control and primary disturbance signals (since intensity is based on the product of two quantities, both sinusoidal for sinusoidal excitation).

What is needed in these instances is some nonlinear filter architecture/adaptive algorithm combination which can supplant the currently used linear one. One possible architectural candidate is a feedforward neural network [2], [3]. There are two issues which must be addressed in regard to this suggestion: What are the details of an algorithm which can be used to adapt a feedforward neural network for such an implementation, and what are the practical performance characteristics of a neural network-based system in this implementation. The work to be presented in this paper aims to address these two issues.

The ability of an artificial neural network to be "trained" to perform some desired task using the gradient descent-based backpropagation algorithm has been well documented [4]. It would seem likely, therefore, that such an architecture/algorithm combination could be employed to perform the previously mentioned nonlinear active control tasks, where the neural network would be trained to derive an output signal which would "cancel" the unwanted disturbance. As with the linear filter based systems, however, implementation of a gradient descent algorithm in an adaptive feedforward active control system is not straightforward. Referring to Fig. 1, the reason for complication is a (system dependent) cancellation path transfer function between the control signal input to the system and the associated error measurement output. This transfer function incorporates the frequency response characteristics of the control actuator(s) and error sensor(s), as well as the response characteristics of the structural/acoustic system which separates them, including delays due to the finite distance between source(s) and sensor(s). It is intuitively
obvious that the existence of this transfer function must be taken into account when adapting the control system if algorithm stability is to be maintained, a fact which is well documented in both adaptive signal processing [5], [6] and active noise and vibration control [1], [7]—[12] literature, and recently restated in regard to neural network based systems [13]. For the (most common) linear FIR filter-based active noise or vibration control arrangement, this leads to a version of the (most common) gradient descent-based least mean square (LMS) algorithm referred to as the filtered-x LMS algorithm [1], [6], [8], [9]. As will be discussed later, stability is maintained in this adaptive algorithm by "filtering" the reference signal, which had been used in deriving the control signal, through an estimate of the cancellation loop transfer function before it is used by the adaptive algorithm to update the weights in the FIR filter. A similar arrangement can be utilized to enable the replacement of the FIR filter with an IIR filter [10], [11].

The use of neural networks for nonlinear feedforward control problems is certainly not new (see, for example, [13]—[17]). The work to be presented here, however, has two distinguishing features: First, it provides details of a specific implementation targeted at supplanting an existing, commonly used linear feedforward adaptive control arrangement with a nonlinear controller for some applications. Therefore, the neural network must "fit" into the physical and conceptual space defined by the linear controller, with no additional requirements. The adaptive algorithm to be formulated for such an arrangement will be shown to be an extension of the linear filtered-x LMS algorithm, and while it falls within a recently described general framework [13] is perhaps better viewed as an extension of the use of a feedforward neural network for nonlinear signal processing [18]—[20], in much the same way the linear arrangement which it aims to supplant is viewed as an extension of a linear adaptive signal processing arrangement. Second, performance characteristics will be assessed experimentally rather than in simulation, since for the architecture/algorithm combination considered here to have the potential to supplant the "standard" linear one it must be able to provide control while contending with real-world limitations of low-cost microprocessor performance, nonideal peripherals such as sensors, actuators and filters, and background noise.

This paper therefore begins by detailing an adaptive algorithm for implementation of a feedforward neural network in an active noise or vibration control system. The algorithm is then compared to the common filtered-x LMS algorithm normally used in such implementations, showing it to be simply a generalization of this linear methodology. Following this, experimental testing of the control arrangement and comparison with linear controller results will be described. The paper concludes with a qualitative discussion of the performance of the neural network implementation in the feedforward active control arrangement.

II. ALGORITHM DEVELOPMENT

Development of the algorithm for adapting the neural based controller will be undertaken with reference to Fig. 2. With the arrangement shown here, a reference input sample at time \( k \), \( x_{in}(k) \), which is in some way related to (but not necessarily linearly correlated with) the impending primary disturbance, \( p(k) \), is used to derive a set of control signals, \( x_c(k) \), via the neural network controller. Each control input to the system is modified by some system dependent cancellation path transfer function to produce the feedforward control signal, \( s(k) \), as measured at the output of the error sensor. In a linear active control problem, this transfer function is normally modeled for time domain adaptive filtering as a finite impulse response function, but in this nonlinear implementation will be modeled as a second neural network. In deriving \( s(k) \) from the control signal \( x_c \) the transfer function model utilizes both present and past control signals (via a tapped delay line), which enables the modeling of explicit system time delays to maintain causality within the control scheme. Each error signal is then the sum of the primary and control components (superposition of the signals in the structural/acoustic environment)

\[
e_j(k) = p_j(k) + s_j(k).
\]

It will be assumed in the following analysis that the neural network model of the cancellation path transfer function has already been formulated with some "reasonable" accuracy.
(extrapolating from linear filter results [5], [12], [21]-[23], “reasonable” will be taken to mean within 90 degrees of the correct phase response), and the aim is to use the model to facilitate stable adaptation of the neural controller, via a gradient descent algorithm, to minimize the system error criterion. At first glance this problem may seem trivial, as application of the standard backpropagation algorithm will perform the outlined task. Owing to the inclusion of the tapped delay line input to the transfer function model, however, the standard backpropagation algorithm cannot be used directly in this arrangement as it must backpropagate through a tapped delay line. Therefore, the standard gradient descent algorithm must be modified to enable adaptation of the neural controller for use in feedforward active control systems.

One point which should be stressed here is that it is not possible to simultaneously adapt both the cancellation path transfer function model neural network and the controller neural network, owing to the inherent differences in their function. The former neural network is a model, whose error is based on the difference between its output and some desired signal (here that desired signal is the system response measured at the error sensor output to the control signal input). The controller neural network is inherently a “phase inverse” model, whose error signal is defined as the sum of its output and the signal whose phase inverse is desired (for the feedforward control scheme this amounts to the superposition of the control signal and the primary noise disturbance in the structural/acoustic domain). Therefore, these two networks, the cancellation path transfer function model and the controller, must be adapted separately. It is not necessary to derive an algorithm here for adapting the cancellation loop transfer function neural network as the common backpropagation algorithm [4] is capable of performing the task. Once converged, the cancellation path transfer function model is then simply used as a tool to facilitate stable adaptation of the controller neural network and is not modified itself in this process.

The error criterion which the controller is to minimize is the sum of the mean square values of error signals from Ne error sensors

$$
\Xi(k) = \sum_{n=0}^{N_e-1} e_n^2(k) = \sum_{n=0}^{N_e-1} E\{e_n^2(k)\}
$$

(2)

to be facilitated by the introduction of Nc control inputs. The gradient descent algorithm which will be employed acts to adjust the weights of the control system to achieve this objective by adding to each a portion of the negative gradient of the error criterion with respect to the weight of interest

$$
w(k+1) = w(k) - \mu \Delta w(k)
$$

(3)

where \(\mu\) is the convergence coefficient, or portion of the negative gradient to be added. What is required, therefore, is an expression for the gradient with respect to each weight in the neural network controller so that the generic gradient descent algorithm of (3) can be used to facilitate “tuning” of the system. For practicality the stochastic approximation \(\xi(k) \approx e^2(k)\) is used, so that the error criterion actually employed in the algorithm derivation is

$$
\text{minimize } \sum_{n=0}^{N_e-1} e_n^2(k).
$$

(4)

Noting that the primary disturbance \(p(k)\) is in no way a function of the weights of the control system or cancellation path transfer function model, the gradient estimate used in the adaptive algorithm is

$$
\Delta w \approx \sum_{n=0}^{N_e-1} \frac{\partial e_n^2(k)}{\partial w} = 2 \sum_{n=0}^{N_e-1} e_n(k) \frac{\partial e_n(k)}{\partial w} = 2 \sum_{n=0}^{N_e-1} e_n(k) \frac{\partial s_n(k)}{\partial w}.
$$

(5)

The aim of the following analysis, based upon the preceding framework, is to find a solution to (5) for each weight in the controller neural network. The solution can then be substituted into the generic gradient descent algorithm of (3) to adapt each weight in the control system.

Assuming that the system itself is linear (as opposed to the control problem being linear, which is not assumed), the cancellation path transfer functions can be considered individually for each control signal. This enables the system to be modeled as shown in Fig. 2, where a separate neural network is used to model the cancellation path transfer function between any given control signal and each of the Ne error sensors (each neural network model has Nc inputs, from a tapped delay line, and Ne outputs, one for each error sensor). Note that if the system is not linear then some simple modifications to the following analysis can be made, where a single cancellation path transfer function model neural network, incorporating all control signals as inputs, is used. The advantage to separating the transfer functions as shown in Fig. 2 is an implementation one; for several control signals and several error sensors, it was found to be easier to obtain several small transfer function models as opposed to a large, all-inclusive one.

As mentioned, the impediment to utilizing the standard backpropagation algorithm for adapting the controller neural network is the inclusion of a tapped delay line in the transfer function model. To derive a modified algorithm, the error signal(s) can first be backpropagated from the transfer function model output to the tapped delay line providing the model

$$
\delta_{\text{node}}(k) = \begin{cases} 
2e(k)f'[\text{netoutput}(k)] & \text{layer} = \text{output} \\
2 \sum_{j=1}^{N_{\text{nodes (layer +1)}}} e_j(k)w_{i,j} & \text{layer} = \text{hidden}
\end{cases}
$$

(6)
input using the standard algorithm [4] shown in (6), found at the bottom of the previous page, where $\delta_{\text{node}}$ is the backpropagated error for the node of interest, $N_{\text{nodes}}[\text{layer}+1]$ is the number of nodes in the layer immediately following ("downstream") the layer of interest, and $\text{net}_{\text{node}}(k)$ is the result of the multiply/accumulate operation at the node of interest at time $k$.

Consider now the output layer of the controller neural network. To derive the gradient of the error criterion with respect to the weights of the $c$th output node, it must be noted that the samples in the delay chain input to the cancellation loop transfer function model associated with it are related to the weights of this node by

$$x_{c,0,k}(k) = f_{0,c}[w_{c,0}(k)x_{hf}(k)]$$

Taking into account all of the elements in the delay chain, the gradient $\Delta w_{0,c}$ can be expressed

$$\Delta w_{0,c} = \sum_{j=0}^{N_{-1}} \delta_{0,c}(k-j) x_{hf}(k-j)$$

where

$$\delta_{0,c}(k-j) = f_{0,c}[\text{net}_{0,c}(k-j)] \sum_{n=1}^{N_{c,hi}} \delta_{c,hi,n}(k) w_{c,hi,n,j}(k)$$

$N_{c,hi}$ is the number of nodes in the initial hidden layer (denoted by $hi$) of the transfer function model and $w_{c,hi,n,j}$ is the weight of node $n$ in the initial hidden layer of the neural network transfer function model associated with control signal $c$ which weights the input from stage $j$ in the tapped delay line.

Deriving the gradient estimate for the weights coefficients in any other layer of the controller neural network is now a trivial task. Consider the layer immediately preceding the output layer, which is the final hidden layer as denoted by the subscript $hf$. The gradient for the weight coefficient vector associated with node $t$ in this layer, $w_{hf,t}$, can be written directly as

$$\Delta w_{hf,t} = \sum_{j=0}^{N_{c}-1} \delta_{hf,t}(k-j) x_{hf-1}(k-j)$$

where

$$\delta_{hf,t}(k-j) = \sum_{\gamma=0}^{N_{c}-1} \delta_{c,hi}(k-j) w_{c,hi,t}(k-j) f_{hf,t}[\text{net}_{hf,t}(k-j)]$$

and $N_{c}$ is the number of nodes in the controller neural network output layer, equal to the number of control outputs. Summarizing, the gradient estimate for any nodal weight coefficient vector in the controller neural network can be expressed as

$$\Delta w_{\text{layer,node}} = \sum_{j=0}^{N_{c}-1} \delta_{\text{layer,node}}(k-j) x_{\text{layer-1}}(k-j)$$

where (see (13) at the bottom of the page) $t$ node denotes the cancellation loop transfer function model associated with the control output node.

One point to note is that although it is the error signal which is thought of as being “backpropagated” through the delay chain input to the transfer function model, it is in fact past and present versions of the nodal outputs which are used in updating the controller network weights, and not past and present values of the error signal. This makes sense intuitively, as the error signal is a function of past and present nodal outputs, and not vice versa. Interestingly, the algorithm presented in [3] for adapting a neural network implemented in a feedforward active control system was derived directly from consideration of this fact, and although the approach taken was different than the one employed here the end result is the same.

### III. COMPARISON TO THE FILTERED-X LMS ALGORITHM

It will be useful at this stage to compare the algorithm derived in this section for adapting the neural controller with the filtered-x LMS algorithm, used for adapting an FIR filter used in a feedforward active control system. Referring to Fig. 3, the single input, single output version of the filtered-x LMS algorithm is [1], [6]-[9]

$$w(k+1) = w(k) - 2\mu e(k)f(k)$$

where $w(k)$ is the vector of weight coefficients in the filter at time $k$, $\mu$ is the convergence coefficient, $e(k)$ is the error signal at time $k$, and $f(k)$ is the “filtered” reference signal vector

$$f(k) = [f(k) \ f(k-1) \ \cdots \ f(k-(n-1))]^T$$

whose elements are produced by “filtering” the reference signal samples through an estimate of the cancellation path transfer function. If the cancellation path transfer function is modeled as an $n$-order finite impulse response function (vector) $t$, this filtering is accomplished by the convolution

$$f(k) = x^T(k)t$$

where $x(k)$ is the vector of reference signal samples in the filter at time $k$

$$x(k) = [x(k) \ x(k-1) \ \cdots \ x(k-(n-1))]^T.$$
Fig. 3. Outline of FIR filter/filtered-
LMS algorithm controller implementa-
tion.

Note that the filter output, \( y(k) \), which is the control signal, is formulated by
\[
y(k) = z^T(k)w(k).
\]  
(18)

The neural network controller/cancellation path transfer function model combination being considered here can be made architecturally equivalent to the FIR filter controller/cancellation path transfer function model combination by representing both neural networks as having only a single output layer with a single node, having a linear nodal output function \( f \). If this is the case then the backpropagated error signal at the output of the transfer function model is
\[
d_0(k) = 2e_0(k)f'[net_{0,0}(k)]=2c(k).
\]  
(19)

where \( e_0 \) denotes error signal 0, and \( t0 \) denotes the transfer function model associated with control output zero. If this is now backpropagated to the controller network the gradient estimate given in (12) becomes
\[
\Delta w_{0,0}(k) = \sum_{j=0}^{N-1} d_{0,0}(k-j)z(k-j)
\]  
(20)

where
\[
d_{0,0}(k-j) = \sum_{n=0}^{N_{0,0}-1} e_0(k)w_{0,0,0}(k).
\]  
(21)

Substituting (21) into (20), the resultant expression can be written in matrix form as
\[
\Delta w_{0,0} = \sum_{j=0}^{N-1} e(k)w_{0,0,0}^T(k)\mathbf{x}(k-j)
\]  
(22)

which can be expressed as
\[
\Delta w_{0,0} = c_0(k)f(k)
\]  
(23)

where \( f(k) \) is as defined in (15). When this is substituted into the generic gradient descent algorithm format of (3) the result is simply the filtered-x LMS algorithm. It can therefore be surmised that the algorithm derived here is simply a generalization of the filtered-x LMS algorithm.

IV. EXPERIMENTAL IMPLEMENTATION

Now that an algorithm has been derived for using a neural network in a feedforward active control system, what remains is to implement it to see if it can perform the desired tasks. The testbed chosen for this was a simple cantilever beam, depicted in Fig. 4. The dimensions of the beam are 1100 mm x 32 mm x 5 mm, and the material is plain carbon steel. The low frequency response characteristics of the beam are shown in Fig. 5. The primary noise disturbance and control signal are both supplied to the beam by point forces, provided by electrodynamic shakers, located 958 mm and 650 mm from the end of the beam, respectively. The primary disturbance is sinusoidal at 53 Hz, with this signal also being supplied to the controller as a reference signal. The error signal is beam displacement 1085 mm from the clamped end of the beam as measured by a gap sensor. This experimental arrangement was selected both for convenience (it was part of another set of experiments) and simplicity, so that the performance characteristics of the neural network control system will be clearly elucidated. No examination of the overall beam response to the control input was undertaken; the aim of the experiments was purely to see if the neural network controller was capable of minimizing the measured error signal based on the reference signal it was given, the common requirement of adaptive feedforward controllers.

A nonlinearity was introduced into the experimental arrangement by not physically attaching the control source shaker to the beam; rather, the shaker was simply pushed up against the beam, and the resulting preload used to maintain contact between it and the beam. Consequently, if the control source is driven "lightly" the resultant error signal autospectrum is reasonably clean, as shown in Fig. 6 for the 53 Hz
pure tone input. If the control source is driven harder the resultant autospectrum becomes increasingly "noisier," as also illustrated in the figure. Although this represents an exaggeration of the problem, this form of nonlinearity is common in active vibration control systems, where misalignment of shakers and backlash in stinger ball joints is often responsible for the excitation of higher frequency harmonics. To further exaggerate the relative importance of the harmonics in the spectrum, in some tests the error signal is passed through a bandpass filter as shown in Fig. 4. The rolloff of this filter was 50 dB/octave, with the center frequency adjustable, as will be outlined. It should also be noted that both the error and control signals were high and low pass filtered at 15 Hz and 400 Hz, respectively, for all tests.

The control system was implemented on a transputer array, with the neural network controller responsible for deriving a control signal implemented on a single T805 32 bit floating point transputer. Reference signal input and control signal output was handled by separate link-connected T222 16-bit fixed point transputers, with memory mapped 12-bit A/D and D/A converters. The sampling rate for the experiments presented here was 800 Hz. While various combinations of neural network layer and node numbers were used in the experiments, as will be outlined, for all tests there was only a single output node. Two types of nodal output function were used: For the output node, the function was linear, providing the control signals with the capacity to vary over the positive/negative range required for control. All other nodes used a sigmoidal nonlinear output function

$$f(x) = \frac{1}{1 + e^{-x}}.$$  \hspace{1cm} (24)

When implementing the neural network controller, calculation of the actual nonlinear nodal output was replaced by a lookup table/interpolation scheme to increase the speed of output calculation. A separate T805 transputer was used to calculate new weights for the controller neural network. These calculations were done in parallel with the derivation of the control signal and passed to the controller network when completed. The weight update calculations were done using a neural network cancellation path transfer function which had been adapted prior to the start of control. As with the controller network, a variety of layer/node combinations in the transfer function model were used in the experiments, as will be outlined, but always a single linear output was used, with all other nodes having a nonlinear sigmoidal output function.

For comparative purposes, a six tap FIR filter controller, adapted using the filtered-x LMS algorithm given in (14), was also implemented. The microprocessor layout of the control system was identical to that described for the neural controller, with the filter and algorithm running on separate microprocessors. Modeling of the cancellation path transfer function was done on-line using the extended least squares approach discussed in [10] and [24], also using a six tap FIR model.

The first case to consider is a simple linear one, where the control source is capable of suppressing the primary disturbance without shaking hard enough to loose contact between the shaker and beam, and the bandpass filter on the error signal was not used. The error signal autospectrum during primary excitation and under control for this case is shown in Fig. 7. Significant attenuation, 56.6 dB, of the tone can be seen when applying the "standard" linear adaptive control arrangement, as would be expected for such a simplistic problem (the residual 50 Hz peak is due to electrical noise). The control achieved by using a 6x6x1 neural network controller (six inputs, six (nonlinear) hidden layer nodes, and one (linear) output node), combined with a 6x1 cancellation path transfer function model (six inputs and one (linear) output node) looks identical to the linear control case, with 52.4 dB of attenuation. In fact, the output from the neural network controller is purely sinusoidal, as illustrated in Fig. 8. (Note that a summary of the principal attenuation results is provided in Table I.)

While this may seem a somewhat trivial test, using a (relatively) computationally intensive nonlinear adaptive control system to provide linear control, it is important for two reasons: First, it demonstrates that the architecture/algorithm combination works, at least for a simple linear case. Second, if it is possible that sometime in the future neural networks do supplant linear filters in adaptive feedforward active control systems, they must be able to handle linear control problems effectively, as these make up the majority of situations encountered. If they are not capable of effectively providing
control with linear problems then their potential utility is greatly reduced.

Having demonstrated that the neural network/adaptive algorithm combination is capable of achieving control when the problem is linear, nonlinear control problems must be used to assess its overall capabilities. In the first experiment, the previous problem will be altered slightly in two ways: by increasing the driving force of the primary disturbance and by bandpass filtering the error signal to provide a slight bias to the higher frequency harmonics, with the center frequency of the filter set to 85 Hz. The resultant error signal autospectrum for primary excitation is shown in Fig. 9. Comparing this with the original case of Fig. 7, it can be seen that the primary tone is slightly reduced in level, even though the primary force has been increased (the effect of bandpass filtering). Also, the spectrum has become noisier, owing to the fact that the increased levels of beam vibration cause the control shaker to rattle.

The effect of driving the primary source harder is that the control shaker must also drive harder to suppress the primary disturbance, which in this case will cause the shaker to rattle as it loses contact with the beam. The result of this is evident in Fig. 9 in the residual error autospectrum when applying a control force via the FIR filter control arrangement. By comparing this with the primary spectrum, it can be seen that significant levels of attenuation, 39.8 dB, of the 53 Hz primary tone were achieved. This, however, was offset to some degree by a 16.1 dB increase in the level of the first harmonic, at 106 Hz. This is comparable to the result obtained when deriving a control input via a 4x6x4x1 neural network (four inputs, six nodes in the first hidden layer, four nodes in the second hidden layer, and a single output node), adapted using a 4x4x1 neural network cancellation path transfer function model, also shown in Fig. 9. While the attainment of the primary tone, 33.6 dB, is not quite as large as was achieved with the linear filter, there was no increase in the first harmonic level. In fact, there was even a (practically insignificant) decrease of 0.2 dB. The control signal produced by the neural network is shown in Fig. 8. It is clearly evident that there is some slight distortion of the waveform, which combats the nonlinearity of the control actuator. There is no mechanism in the linear filter/algorithm combination for forming this distortion, hence the inferior control performance in this instance.

To continue with this line of experimentation, the error signal can now be bandpass filtered with the center frequency of the filter set at the frequency of the first harmonic, 106 Hz, giving even more emphasis to the higher frequency components. The resultant error signal autospectrum during primary excitation and under control is shown in Fig. 10. Under primary excitation the level of the 106 Hz first harmonic is now approximately 15 dB above that of the 53 Hz tone. When using a control input derived by the FIR filter controller, there was a small decrease in the 53 Hz primary tone level, 13.6 dB, but virtually no change in the level of the dominant 106 Hz first harmonic. Presumably the reduction in the 53 Hz component would improve over an extended period of time, this time required as its level is effectively in the noise floor of the spectrum. No improvement in the level of attenuation of the first harmonic can be anticipated, however. This result can be contrasted to that of applying a control signal derived from a 4x8x4x1 neural network controller, adapted using a 4x4x1 neural network cancellation path transfer function model. Here the attenuation in the level of the primary 53 Hz tone is reduced only 9.3 dB, but this is more than compensated for.
by the 24.4 dB reduction in the dominant 106 Hz harmonic. In terms of "real" levels of control, as would be perceived by a human monitor, the neural network controller performance is vastly superior to that of the linear controller.

One more source of nonlinearity introduction into the structural/acoustic system will be examined in this section, that being due to a distortion of the reference signal. Such a case can occur in a system where the reference signal is measured by an acoustic or vibration transducer, and the levels are large enough that the transducer amplifier clips the signal. For the purposes of the experiment, the 53 Hz reference signal was intentionally clipped, as illustrated by its waveform shown in Fig. 11. The primary disturbance, however, is still generated by the unclipped sinusoidal signal, resulting in the primary disturbance error signal autospectrum depicted in Fig. 12. For this experiment there was no filtering of the error signal.

Consider first the effects of applying a control signal generated by the linear FIR filter system, as depicted by the residual error autospectrum of Fig. 12. The performance of this control arrangement is not particularly outstanding for controlling a purely sinusoidal primary tone, only 23.9 dB attenuation. This is accompanied by, and offset to a large degree by, an increase in the noise levels of the residual spectrum. This can be compared to the residual error autospectrum when using a 4x6x4x1 neural network controller, adapted using a 6x6x1 cancellation path transfer function model, also shown in the figure. Not only is the reduction in the primary tone level superior, now 41.9 dB, but the noise in the spectrum is also reduced (the residual 50 Hz peak is electronic noise).

The reason for the differences in the residual error autospectrums can be understood in part by examining the two control signals produced by the two different arrangements, which are plotted together in Fig. 13. In comparing these, the increased "noise" in the linear controller produced signal is clearly evident, while the neural network controller produces a smoother, albeit still imperfect, signal. The end result is that while the performance of the neural control scheme is still not ideal, it is superior to that of the linear FIR filter controller.

One final point which should be mentioned concerning the experimental data presented in this section is convergence time. In many ways "raw" convergence time is an unfair comparison between the two systems, for several reasons: First, the FIR filter-based controller must perform both on-line cancellation path transfer function modeling as well as performing weight update calculations for the control filter. This is an extra computational burden not placed upon the neural network controller, which was adapted using a model of the cancellation path transfer function obtained prior to start-up. Second, the weight update calculations of the neural
network are more multiplication intensive than those of the FIR filter-based controller, due to the size of both the transfer function model and controller, and the transputer is relatively slow in multiplication. Therefore, a different hardware arrangement might result in different relative convergence times. Third, there is no way of knowing how "ideal" the convergence coefficient used in updating the neural network is in terms of speed performance, so that an alternative value of convergence coefficient could be faster or slower (in fact, later work using a time varying convergence coefficient as described in [25] for linear adaptive filters showed significant changes in convergence speed in some cases). Regardless of these factors, in terms of "raw" convergence time, the FIR filter-based controller was always quicker to converge, taking on the order of 10 seconds from startup to convergence. The neural network convergence time ranged between approximately 15 seconds and one minute, depending on both the test problem and the individual run. Linear control problems where on the short end of this scale, and the most nonlinear on the long end.

V. DISCUSSION

The experimental results presented in the previous section clearly demonstrate the ability of the neural network controller/algorithm scheme developed in the beginning of the paper to provide significant levels of disturbance attenuation in a feedforward active control scheme. Referring to the summary of results provided in Table I, this attenuation was seen to be the (practical) equal of a linear controller for a linear control scheme and to be superior to the linear control scheme when the control problem has some inherent nonlinearity. The neural network controller was shown to be able to compensate for the introduction of harmonics by the control actuator in Figs. 9 and 10 by producing a control signal, derived from a pure tone reference signal, which contains some level of harmonics as illustrated in Fig. 8. This ability is extremely attractive, especially in active noise control where higher frequency components of sound are usually perceived as louder than equal magnitude lower frequency components. Also, the neural controller was seen to be able to compensate for a distorted reference signal in a manner superior to that of a linear controller in Figs. 12.

The question remaining to be answered is, does this mean then that the neural network controller/algorithm arrangement utilized here is capable of supplanting the linear filter/algorithm combinations commonly used in feedforward active control schemes? The subjective assessment, after having completed the experiments in the previous section, is no. At least, not yet. The principal problem, not evident from the experimental data, is a lack of predictability and consistency as to what the neural network controller will do. Currently, the level of knowledge concerning what influence various structural/ acoustic system parameters have upon the performance of the linear filter/algorithms is sufficiently advanced to enable the design of stable adaptive feedforward control systems. If a parameter in the system is changed (for example, additional error sensors being included in the system), it can be directly compensated for in the adaptive algorithm used in the linear feedforward control system to maintain stability and performance. Also, if a linear control system is effective under one set of conditions at a given instant in time, it can be assumed that it will be equally effective under the same set of conditions at some other point in time. No such statements could be made regarding the neural controller used in these experiments. While for a given set of conditions the neural network controller would be consistently stable, the levels of control obtained would not be consistent. Sometimes the controller would simply turn itself off, presumably because there was a local minima in the error criterion for this result and the random initial weights were not sufficient to enable the algorithm to avoid getting trapped in it. Also, how control was achieved was inconsistent. Will convergence be fast or slow? Will the control signal be clean or distorted in some unpredictable manner? These questions could not always be answered based on previously acquired knowledge.

The most predictable cases were found to be linear control problems, but even these can occasionally have unforeseen results. Consider for a moment the results of Figs. 7 and 8. When controlling a pure tone the neural controller performed very well, producing 52.4 dB of attenuation as a result of the "beautiful" sine wave control signal shown in Fig. 8. It was said in describing the test that no bandpass filtering of the error signal was used. To pose a question, what would be expected to happen if the previously described bandpass filter was used on the error signal, with the center frequency set at 53 Hz such that (practically) only the fundamental tonal component of the error signal was available to the adaptive controller? If the controller was linear this would have no detrimental effect, as the control output is constrained to be some linear function of the 53 Hz reference signal provided to it. This constraint is not placed upon the neural network controller, however, and so the addition of filtering is responsible for the control signal depicted in Fig. 14. This control signal produces 48.0 dB attenuation of the 53 Hz tone, but from a subjective point of view the end result was disastrous. The neural network controller, however, does not know that fact.

Therefore, based on the experimental work conducted and presented in this paper, the following can be said: The neural network controller/algorithm scheme presented here for feedforward active control shows the potential to equal to performance of a linear control scheme for a linear control.
problem and to have (far) superior performance for nonlinear problems. For this potential to be fully realized, however, a great deal more work needs to be done, work directed toward constraining the network not only to perform the desired task, but to perform it consistently and within some acceptable bounds of "side effects," such as convergence speed and signal distortion.

VI. CONCLUSIONS

A neural network controller/adaptive algorithm combination has been developed for use in feedforward active control schemes. The potential of this control scheme has been demonstrated experimentally, and compared to a linear control scheme was found to be equal in performance for a linear control problem, and superior for a nonlinear one. The neural network controller was shown to be able to compensate for the introduction of harmonics by the control actuator by producing a control signal, derived from a pure tone reference signal, which contains some level of harmonics. Also, the neural controller was seen to be able to compensate for a distorted reference signal in a manner superior to that of a linear controller. The main drawback concerning the use of the nonlinear neural network controller was a lack of consistency and predicability concerning its performance. Despite this, the results presented here are extremely encouraging.

REFERENCES


Scott D. Snyder (A’91) received the B. E. (Hons.) and Ph.D. degrees in mathematical engineering from the University of Adelaide, Australia, in 1988 and 1991, respectively. During 1992 he was a Visiting Research Fellow at the Mechanical Engineering Laboratory, Tsukuba City, Japan. Since 1993 he has been a Research Fellow at the University of Adelaide. His research interests include theory and implementation of active noise and vibration control systems.

Nobuo Tanaka received the M.S. and Ph.D. degrees in mechanical engineering from Tokyo Metropolitan University, Japan, in 1973 and 1977, respectively. From 1975–1976, he was on study leave at the University of California at Berkeley, USA. Since 1981 he has been a Senior Research Scientist in the Vibration Engineering Division at the Mechanical Engineering Department of the University of Adelaide, Australia. His research interests include theory and practice of active vibration and wave control.