Geophys. J. Int. (2003) **153,** 409–423

The application of artificial neural networks to magnetotelluric time-series analysis

C. Manoj and Nandini Nagarajan

National Geophysical Research Institute, Hyderabad, India 500 007. *E-mail: manoj_c@rediffmail.com*

Accepted 2002 October 24. Received 2002 October 10; in original form 2001 September 7

SUMMARY

Magnetotelluric (MT) signals are often contaminated with noise from natural or man-made processes that may not fit a normal distribution or are highly correlated. This may lead to serious errors in computed MT transfer functions and result in erroneous interpretation. A substantial improvement is possible when the time-series are presented as clean as possible for further processing. Cleaning of MT time-series is often done by manual editing. Editing of magnetotelluric time-series is subjective in nature and time consuming. Automation of such a process is difficult to achieve by statistical methods. Artificial neural networks (ANNs) are widely used to automate processes that require human intelligence. The objective here is to automate MT long-period time-series editing using ANN. A three-layer feed-forward artificial neural network (FANN) was adopted for the problem. As ANN-based techniques are computationally intensive, a novel approach was made, which involves editing of five simultaneously measured MT time-series that have been subdivided into stacks (a stack = 5 $\times$ 256 data points). Neural network training was done at two levels. Signal and noise patterns of individual channels were taught first. Five channel parameters along with interchannel correlation and amplitude ratios formed the input for a final network, which predicts the quality of a stack. A large database (5000 traces for pattern training and 900 vectors for interchannel training) was prepared to train the network. There were two error parameters to minimize while training: training error and testing error. Training was stopped when both errors were below an acceptable level. The sensitivity of the neural network to the signal-to-noise ratio and the relative significance of its inputs were tested to ensure that the training was correct. MT time-series from four stations with varying degrees of noise contamination were used to demonstrate the application of the network. The application brought out the ability of the network to pick out signals even in a high-noise environment. This suggest the possibility of automating the editing of MT time-series with artificial neural networks.

Key words: artificial intelligence, artificial neural networks, data processing, magnetotellurics, time-series.

1 INTRODUCTION

Magnetotelluric (MT) transfer functions are calculated from measurements of horizontal electric and magnetic fields at the surface of the Earth and can image the subsurface electrical conductivity (Cagniard 1953). Because of the skin depth effect of electromagnetic fields, high-frequency MT transfer functions carry information concerning shallow conductivity structure of the Earth, whereas low-frequency MT transfer functions carry information concerning deeper structure. The relationship between the electric and magnetic fields at the surface of the Earth can be written as

$$\begin{bmatrix} E_x(\omega) \\ E_y(\omega) \end{bmatrix} = \begin{bmatrix} Z_{xx}(\omega) & Z_{xy}(\omega) \\ Z_{yx}(\omega) & Z_{yy}(\omega) \end{bmatrix} \begin{bmatrix} H_x(\omega) \\ H_y(\omega) \end{bmatrix}, \quad (1)$$

where $\boldsymbol{Z}$ is the complex transfer function tensor, $\boldsymbol{E}$ and $\boldsymbol{H}$ are the Fourier transforms of the electric and magnetic time-series and ω is the radian frequency. $\boldsymbol{Z}$ is usually represented by the MT apparent resistivity

$$\rho_{xy} \approx 0.2T|Z_{xy}|^2 \text{ ohm m}, \quad (2)$$

and phase,

$$\phi_{xy} = \tan^{-1}[\mathrm{Im}(Z_{xy})/\mathrm{Re}(Z_{xy})], \quad (3)$$

for the period T in s, $\boldsymbol{E}$ in mV km^{-1} and $\boldsymbol{H}$ in nT.

A reliable estimation of magnetotelluric transfer functions depends greatly on the amount of noise in the measured electric and magnetic time-series (Orange 1989). It is often difficult to obtain noise-free measurements as the method relies on highly variable natural electromagnetic fields. Severe problems are caused by civilization, which produces all kinds of electromagnetic noise. Many methods have been proposed to obtain stable and bias-free MT transfer functions in the presence of noise in measurements (Gamble *et al.* 1979; Park & Chave 1984; Chave *et al.* 1987; Chave & Thompson 1989). Though these methods can reduce errors caused by outliers, a substantial improvement is possible when the data presented are as clean as possible (Junge 1996). One approach to clean the data involves the manual inspection of the time-series and the removal of abnormal parts. The visual inspection/editing of magnetotelluric time-series data is time consuming, cost-intensive and subjective in nature. The purpose of this paper is to understand this process and to automate it using artificial neural-network-based techniques.

2 DATA

2.1 Magnetotelluric time-series processing

Magnetotelluric time-series consist of five simultaneously measured components of the Earth's electromagnetic field, i.e. two orthogonal components of horizontal electric fields (E_x, E_y) and three orthogonal components of the magnetic field (H_x, H_y, H_z). The measurements are usually made in wide bands of overlapping frequency ranges, with different sampling intervals. The measurements continue until a sufficient number of stacks (one stack generally consists 128–1024 data points) are obtained. The goal of MT data processing is to obtain the optimum estimates of the elements of $\boldsymbol{Z}$ (minimum bias, minimum variance). The use of traditional spectral analysis together with least-squares estimation is warranted only if the input channels (magnetic fields) are noise-free, the output channel noise has a Gaussian distribution and the MT time-series is stationary (Banks 1998). In reality most data usually show gross departures from the idealistic model described above. The main sources of noise are geomagnetic phenomena, thunderstorms, cultural interferences and instrument problems. This can result in oscillatory and biased estimates of MT transfer functions. The remote reference technique (Gamble *et al.* 1979) is one way to deal with the local noise in magnetic fields. The reference magnetic field is recorded at a site, which is far outside the coherence range of the local noise. The bias in the magnetic field is eliminated by using the cross-spectrum with the remote site instead of the autopower. Problems caused by non-stationarity of time-series can be addressed by subdividing the time-series (Egbert & Booker 1986; Banks 1998) into small segments, estimating transfer functions for each and averaging in a way that discriminates against noisy data segments. Non-Gaussian noise is detected by analysing the distribution of residuals between the observations and the prediction of the least-squares model. Robust estimation of MT transfer functions (Egbert & Booker 1986; Chave *et al.* 1987) down weights the data sections with such large residuals. However, this technique can still give erroneous results if strongly correlated noise is present during most of the recording time. To deal with such coherent noise, a robust multivariate errors-in-variables (RMEV) estimate was developed by Egbert (1997). Correlated and uncorrelated noise are separated iteratively, using data from multiple stations. Variants of this technique such as the signal–noise separation (SNS) method and the SNS remote-reference method were presented by Larsen *et al.* (1996) and Oettinger *et al.* (2001). The present paper does not deal with the statistical estimation of MT transfer functions from time-series. Instead, it concentrates on the manual editing of subdivided MT time-series, which usually precedes the estimation of MT transfer functions. Once the time-series has been edited, the estimation of transfer functions is carried out with a conventional technique (discussed in Section 5). We restrict ourselves to long-period MT time-series sampled at a 1 s interval with each stack containing 256 data points.

2.2 Signal and noise in magnetotelluric time-series

An exact classification of signal and noise characteristics in magnetotelluric time-series is difficult as their sources may have a similar spectral content. Unfortunately, there are often more noise sources than signal sources. Still, a generalized classification is possible according to the origins of both the signal and the noise. At periods longer than 1 s, the natural electromagnetic field originates in the upper ionosphere and magnetosphere. Random bursts of energy originate in charged particles from the Sun and induce sinusoidal electromagnetic waves in the magnetosphere and ionosphere. The signal amplitude and frequency may vary with the energy and type of activity and there may be a long gap between the arrival of two signal trains. Fig. 1(a) shows some typical magnetotelluric signal patterns.

Magnetotelluric noise sources have been classified as both active and passive (Santarato & Spagnolini 1995) and as mechanical, equipment and electromagnetic (Hatting 1989). Cultural noise arises from various sources; a major contributor is related to electrical power. This includes current leaks from power lines, electric pumps and electric railways. The leaked current loops back through the ground, thus forming an electromagnetic (EM) dipole source. In longer-period magnetotelluric data, spikes and step-like bursts can result from power sources. Electric channels are more prone to this type of noise. Movements of ferrous metals or other magnetic material in the vicinity of the magnetic field sensor can introduce noise into the magnetic channels. Loosely connected cables can result in spikes and/or dead traces. Wind or sea-generated vibration of the magnetic field sensor will cause random variations in the magnetic channels. Junge (1996) gives a useful

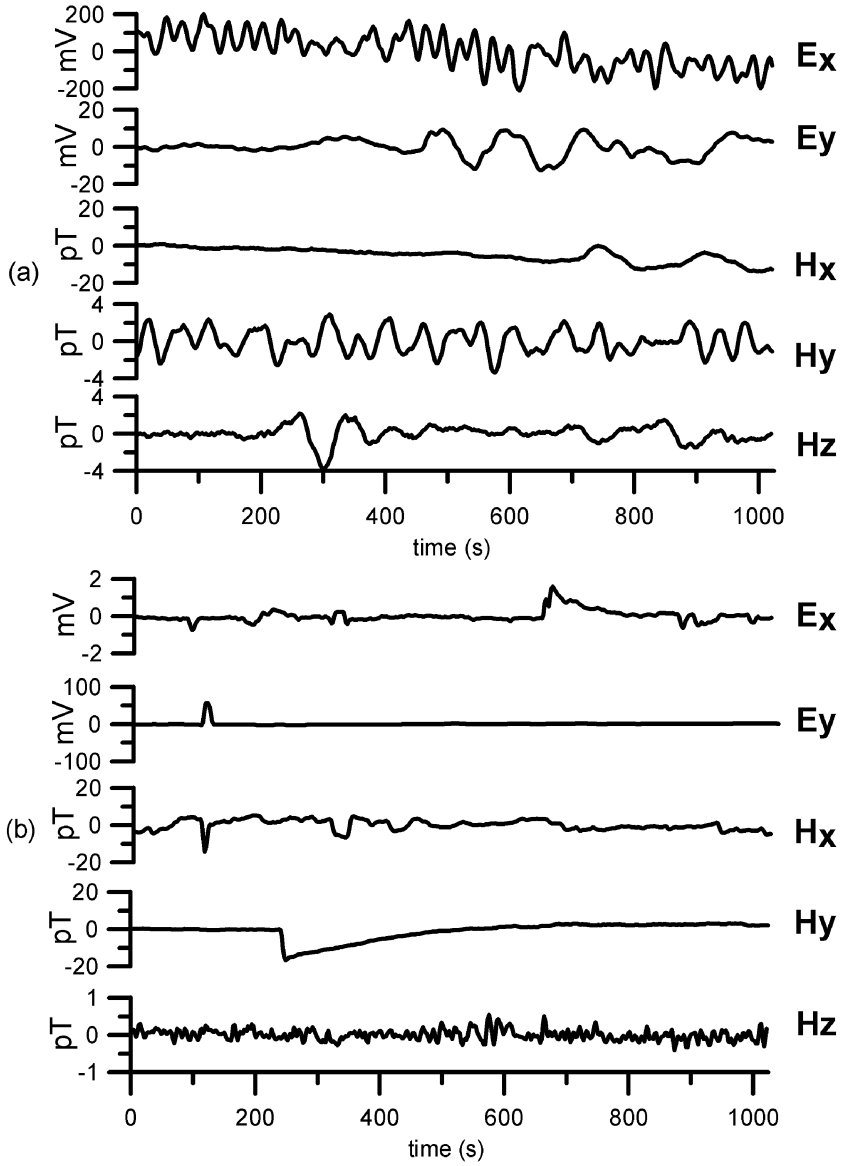

Figure 1. Common signal and noise patterns in long-period MT time-series. Samples are collected from different sites. Note the change in amplitude. (a) Signal patterns; samples of E_x and H_y shows the geomagnetic pulsations. Other channels also show signals but at longer periods. (b) Noise patterns; E_y and H_x show different types of spikes. A step and its decay is shown in H_y. A sample of random noise is shown in H_z.

review on electromagnetic cultural noise characterization and correction. Any or all of the above noise sources may be present together with the MT signal, resulting in compounded responses in the recorded time-series. Some of the most common MT noise patterns are presented in Fig. 1(b).

2.3 Visual inspection (editing) of magnetotelluric time-series data; why automation?

Manual editing is often the first step in magnetotelluric data processing. The editor examines each stack of time-series and labels it as either good or bad according to its signal/noise character. The bad stacks are removed from further processing. The task involves an intensive amount of pattern recognition. Experience provides a judicial balancing of signal characteristics such as shape, amplitude, frequency and correlation. This process is subjective in nature and the same editor may output differently over long sequences of editing. A rule of thumb is that 'if in doubt throw it out'. If questioned concerning a particular decision, however, the editor may offer a few rules for guidance but can give no obvious systematic reasoning. This constitutes the major time and human resource used in MT data processing. The number of stacks of time-series recorded sometimes runs into hundreds. With several such sessions of recordings from a station and many such MT stations occupied in each survey, there is a pressing need to provide a more robust alternative, which is less time consuming and more objective.

2.4 Magnetotelluric noise characterization

Automation of MT time-series editing requires a systematic evaluation of the task performed by the editor. What parameters influence an editing decision? How much importance does the editor give to each parameter? Can it be quantified? As there is no information available in this regard, we undertook an editing evaluation exercise. It involved discussion with different editors and re-analysis of previously edited data and identification of the influence of different parameters in magnetotelluric time-series editing. From the study, we found that the following factors reasonably represent the criteria applied in editing:

(1) signal pattern;
(2) signal amplitude;
(3) correlation of different field components;
(4) general noise level;
(5) quantity of MT data available.

When the general noise level is very high and/or the quantity of data available for editing is limited, compromises are made in editing. As these are exceptional cases, we did not include them in our analysis. The first three factors were analysed and each was given an influence percentage. The influence percentage for a particular factor was found by preferential editing using that criterion and comparing results with the regular mode of editing.

2.4.1 Patterns of signal/noise

Pattern controls a major part of decision making. The signal pattern is transient overlapped sinusoids. The noise patterns can be classified as follows.

(1) Spikes: high amplitude, maximum duration for a few data samples. Sometimes a spike is followed by a transient decay. These are commonly power related.
(2) Noise bursts: plateau or step-like appearance spanning hundreds of data samples.
(3) Noisy trace: random fluctuations as a result of wind, seismic effects and or low signal.
(4) Muted or dead trace: instrument problem such as a broken cable or a failure in the electronics.

In most cases, the quality of signal can be deduced from the signal waveform. This, according to our study, influences 60 per cent of the editing decisions.

2.4.2 Amplitude of signals

Although naturally varying electromagnetic fields exhibit a large variation in their strength, a broad range can be specified. In most cases, the amplitude ratio of orthogonal electric and magnetic signals was found to be a good discriminator. In long-period time-series, the electric field fluctuates within ± 100 mV km^{-1} and the magnetic field fluctuates within ± 0.5 nT in a noiseless environment. The channel amplitude may be increased many times in the presence of certain types of noise. The amplitude criterion often gives the most reliable information if the contaminated signal has the same pattern as noise-free signals but an enhanced amplitude. This was found to influence 20 per cent of the decisions.

2.4.3 Correlation between simultaneously measured channels

The electric and magnetic fields are related by a transfer function defined in Section 1. In the ideal case the MT signals should be highly correlated and random noise will reduce the correlation. However, noise can be highly correlated between E and H channels. Noise from power lines especially near 60/50 Hz is an example. In longer-period data, we found the correlation coefficients between orthogonal electric and magnetic fields (E_x–H_y and E_y–H_x) could be a signal discriminator. Correlation was found to influence another 20 per cent of the editing decisions.

As the above-described parameters influence the bulk of editing decisions, they were selected as the basis for automation. An automation scheme was developed using an artificial neural network for classification/editing of time-series data.

3 ARTIFICIAL NEURAL NETWORK

3.1 Why an artificial neural network?

Optimal conventional automation requires a statistical characterization of the noise. This requires an estimation of certain statistical parameters from the data. In other words, the likelihood ratio of signal/noise is replaced by sufficient statistics on the data. The method is simple and appealing. It works very well when the target signal is known and the noise has a normal distribution. However, in most cases noise does

not have a normal distribution and the likelihood ratio is a complicated non-linear function of the input data. Garth & Poor (1994) classify geophysical signals as non-Gaussian, unstructured signals as they involve the least amount of detail. Classification of such signals is best done by labelling according to their signal content and presenting them to a pattern recognition scheme. Artificial neural networks (ANN) are an emerging tool that have been applied in many areas of science and engineering where pattern recognition is involved, such as speech and character recognition. The learning and adaptive capabilities of these models make them attractive for application to some problems in geophysics (Calderon-Macias *et al.* 2000). The most documented application of ANN in geophysics has been for the automation of seismic data processing and interpretation (Murat & Rudman 1992; McCormack *et al.* 1993; Fish & Kusuma 1994; Dai & MacBeth 1995). ANN have also found other applications in geophysics, such as in interpretation of well log data (Winer *et al.* 1991), locating subsurface targets from electromagnetic field data (Poulton *et al.* 1992) and prediction of upper atmospheric and ionospheric activities (Lundstedt 1996; Altinay *et al.* 1997; Koons & Gorney 1991). More recently, artificial neural networks have been used in magnetotelluric inversion (Zhang & Paulson 1997; Spichak & Popova 2000).

3.2 ANN theory

An artificial neural network is an information processing system composed of a large number of processing elements called neurons, which are modelled on the functions of neurons in the human brain. ANN differ from conventional pattern recognition techniques in their ability to adaptively discriminate or learn through repeated exposure to examples and in their robustness in the presence of high-noise levels. ANN do not require *a priori* knowledge concerning the noise distribution of the process under study, as do its statistical counterparts. Unlike conventional methods, which incorporate a fixed algorithm to solve a particular problem, ANN perform a mapping, usually non-linear, between the input and output data, which allows the network to acquire important information on the problem being solved. It is these characteristics of neural networks that motivated us to investigate their use in MT data processing.

One of the most widely used types of ANN, the feed-forward artificial neural network (FANN) was used in the present study. Its architecture is outlined in Fig. 2. It consists of a layer of neurons that accept various inputs (the input layer). These inputs are fed to further layers of neurons (hidden layers) and ultimately to the output layer, which produces a response. The aim of the technique is to train the network such that its response to a given set of inputs is as close as possible to a desired output. A number of algorithms are available for training a neural network. Back propagation is the most popular training algorithm (Werbos 1990) and was used in the current study. During FANN training, each hidden and output neuron process inputs by multiplying each input by its weights. The products are summed and processed using an activation function, here, a sigmoid function

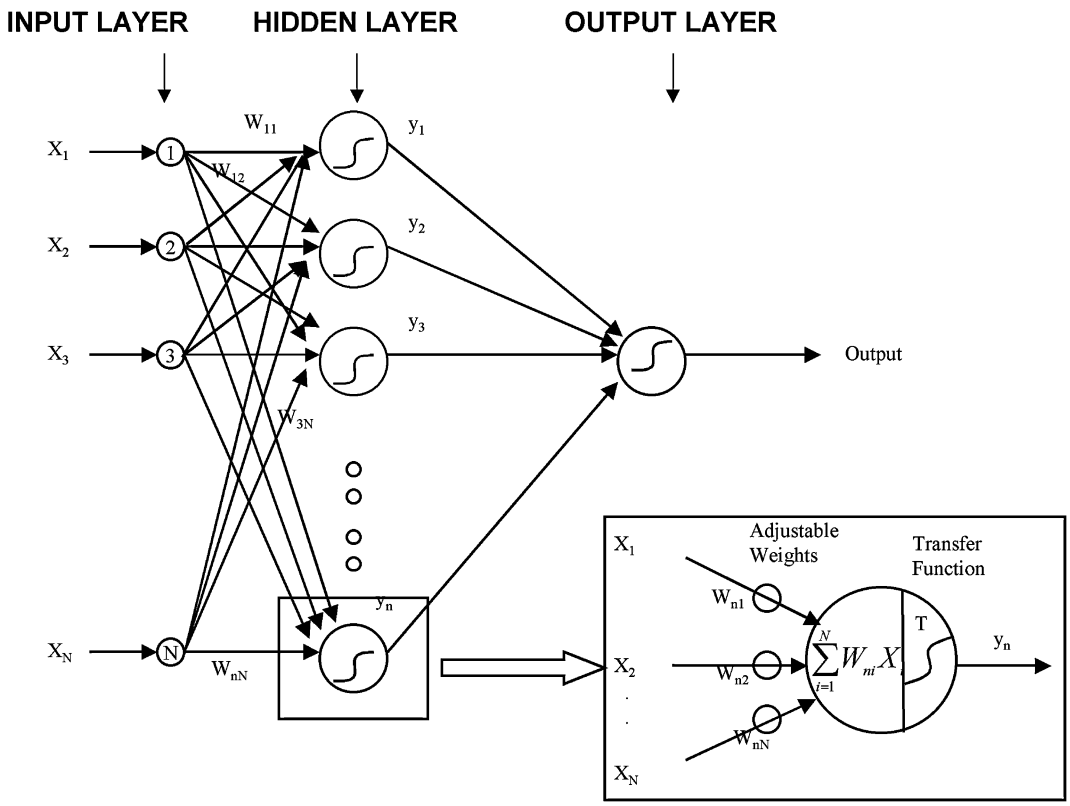

Figure 2. A simple three-layer feed-forward neural network. The data are processed at each neuron in the layers. Each neuron performs a summing of inputs multiplied with a weight parameter and outputs the data through its sigmoid transfer function.

$$f(x) = 1/(1 - e^{-x}), \tag{4}$$

to produce an output with reasonable discriminating power. The neural network learns by modifying the weights of the neurons in response to the errors between the actual and targeted output values. For a given set or vector of N inputs $(x_1, x_2, \ldots, x_n)$, the output of node j is computed as

$$y_j = f\left(\sum W_{ji} x_i\right), \tag{5}$$

where W_{ji} is the weight of the connection between the ith and jth neurons. The learning rule for the adjustments in the weight between neurons i and j is expressed as

$$\Delta W_{ji} = \eta \delta_j o_i, \tag{6}$$

where o_i is either the output of node i or an input, η is a positive constant called the learning rate and δ_j is the error term of node j. Thus

$$\delta_j = \delta E / \delta_{\text{net}\,j}, \tag{7}$$

where

$$E = \frac{1}{2} \sum (y_j - o_j)^2 \tag{8}$$

and

$$\text{net}_j = \sum W_{ji} o_i. \tag{9}$$

Here, y_j is the target value for the jth node and o_j is the output for the jth node. The value δ_j is computed as

$$\delta_j = o_j(1 - o_j) \sum \delta_k W_{kj} \tag{10}$$

if the node is not an output unit. To improve the convergence characteristics, a momentum gain β is added to the weight correction term, which stabilizes oscillations during the learning process (Boadu 1998), i.e.

$$\Delta W_{ji}(n + 1) = \eta \delta_j o_i + \beta \Delta W_{ji}(n), \tag{11}$$

where n is the iteration index. The training of the network is complete if the convergence of weighting coefficients has been achieved. The convergence criterion requires that the sum square error at the output must be less than a desired tolerable error (Luo & Unbehauen 1997).

While training a network, we start with arbitrary values for the weights W_{ji}. It is usual to choose random numbers in the range -1 to 1. Next, we calculate the outputs (O) and errors (E) for that set of weights. Then we calculate the derivatives of E with respect to all of the weights (eq. 7). If increasing a given weight would lead to more error, we adjust the weight downwards. If increasing a weight leads to a reduced error, it is adjusted upwards. After adjusting all the weights up or down, we start all over again, and keep going through this process until the error is close to zero. The sequence of presenting the entire training database, calculating the network response, comparing the result with the assigned class, propagating the error backwards and adjusting the weights is called an epoch. A few thousand such training epochs are usually required by a neural network to reach zero error. However, if the number of training patterns exceeds the number of weights in the network it may not be possible for the sum squared error (SSE) to reach zero.

4 DATA ANALYSIS

4.1 Network engineering

As a the massive interconnectivity and inherent non-linearity of the neural network requires significant computing resources, dedicated mainframe computers and workstations were traditionally used for neural network training. Most of the applications utilizing ANN in geophysics deal with single-channel data using a small sliding window moving along the time-series. The current application, where five channels of data, each containing 256 points, were to be classified simultaneously, posed a major challenge. A novel approach was made to accommodate multichannel data so that all the computing could be done on a PC. Fig. 3 gives a schematic representation of the data flow. This method can be used for any multivariate signal detection scheme. As the signal shape and pattern controls a major part of the editing decisions, attention was focused on this part of the analysis. We also included interchannel parameters, such as amplitude ratios and correlation coefficients for neural network training. The signal detection scheme was divided into two steps.

(a) Detection of the patterns of individual channels: patterns of individual channels were evaluated using a neural network. Within each stack (256 points) of five channels, we classified the pattern of each channel successively. Thus pattern detection of a single stack resulted in five values corresponding to the five channels.

(b) Detection of interchannel parameters: amplitude ratios (E_x/H_y, E_y/H_x) and correlation coefficients between channels were computed. These parameters, along with the pattern quality data from step (a), formed the inputs for another neural network. Output from this network indicates the overall quality of that data stack.

This separation provided us with the flexibility and ease of operation on a small computer, without having to compromise on network performance.

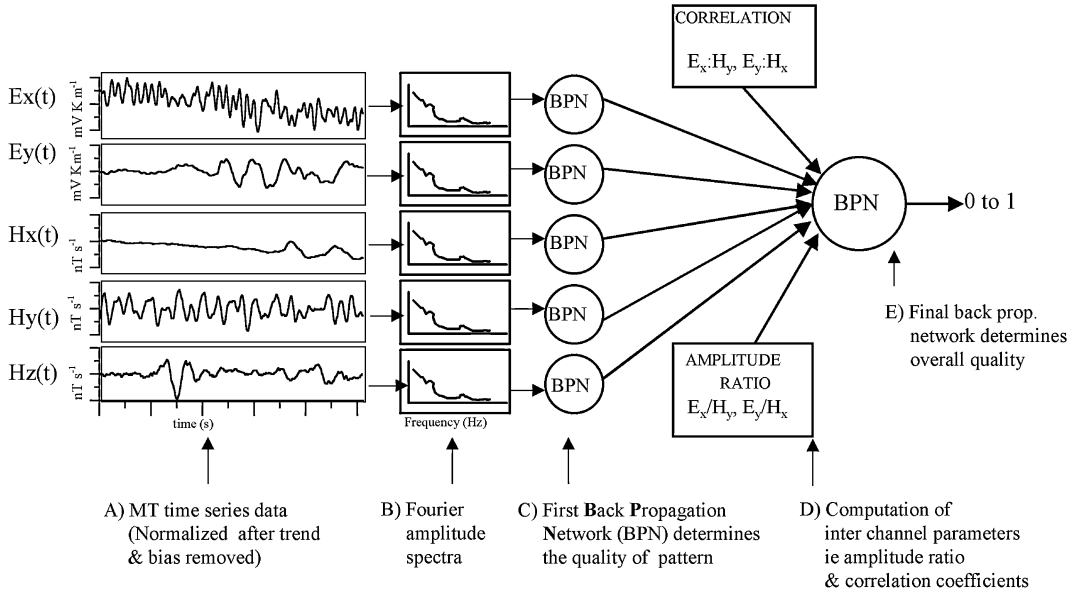

Figure 3. Data flow through the feed-forward artificial neural network-based editing scheme.

4.2 Pattern training

4.2.1 Data used

Selecting an appropriate training data set is one of the most critical steps in successful training. How many patterns are required in training? How many patterns should be dominated by signal and how many by noise? A rule of thumb is that the patterns in the training set should cover the main categories of signal and noise. The signal pattern should represent the typical features of a signal with different frequency characteristics. Obviously, there are more noise patterns than signal patterns (Zhao & Takano 1999).

With an input data space of 256, the number of exemplars required to train a neural network was large ($>$1000 sets). This required us to search for and select a large number of time-series segments. 1200 sets of MT long-period time-series data stacks (256 s length for each stack) were collected from different data sets giving 6000 traces.

4.2.2 Pre-processing

Each channel was corrected for trend and bias and normalized between 1 and 0. The data segments were then manually classified and assigned a value between 0.9 and 0.1, depending on quality. The label varied between 0.1 (bad) and 0.9 (good) depending on the pattern quality of the data. There is no fixed generic relationship between the label and quality (one could also have set the labels in the reverse order). Out of the classified time-series segments, 5000 extreme cases were selected for training. This set contained 2500 very good and 2500 very bad data samples. Care was taken to include all categories of signal and noise patterns generally found in MT time-series. The database was shuffled to have a random distribution of good and bad data samples for training. From this database 3000 exemplars (training vectors) were kept for training and the rest, for testing.

4.2.3 FANN training

The network was presented with the training database of 3000 time-series segments. Training was done with different values of learning rate (η) and momentum gain (β) for different *epochs* (defined in Section 3.2). Training typically took 60 min to complete 1000 epochs on a 500 MHz PC. The longer time for training hampered effective interaction with the process and limited an exhaustive search for the optimum network configuration (i.e. to optimize η, β and the number of hidden neurons). A transform of the input data was sought, which would preserve the essential information concerning the signal while reducing the dimensionality of the input space. For this, the time-series was Fourier transformed, after applying a cosine taper to both ends. To test whether the amplitude spectra alone could be used as a discriminator, the spectra of good and bad signals in the entire training database were stacked separately. As can be seen in Fig. 4, there is a clear difference between the two types. Noise generally raises the spectral power at higher frequencies, whereas the signal has more power at lower frequencies. Amplitude spectra were used for further training. A fast Fourier transform (FFT) enabled us to reduce the number of input data from 256 to 128. Training was attempted with different subsets of spectral harmonics by removing the highest-frequency elements successively. The first 100 harmonics were found to be sufficient and the training time was considerably reduced (20 min). The sum squared error as a function of epoch for the final training is plotted in Fig. 5(a) (see Section 4.3.2). The SSE reached a minimum of 33.73 for 2000 training samples. In

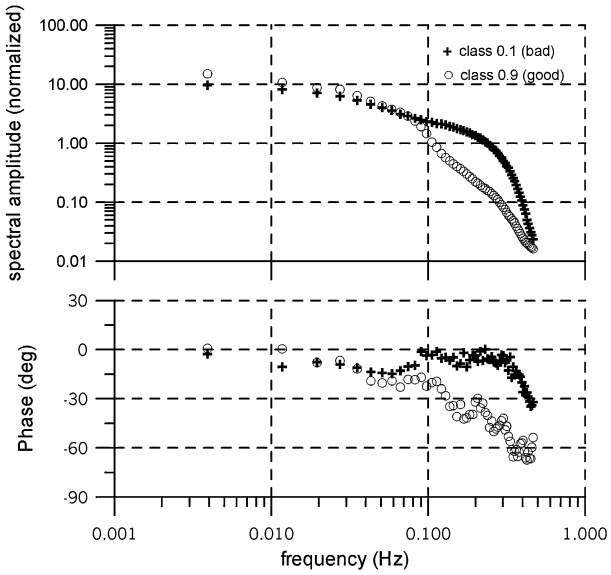

Figure 4. Stacked amplitude and phase spectra of the training database. The spectra of noisy data (class 0.1) clearly different from signals (class 0.9).

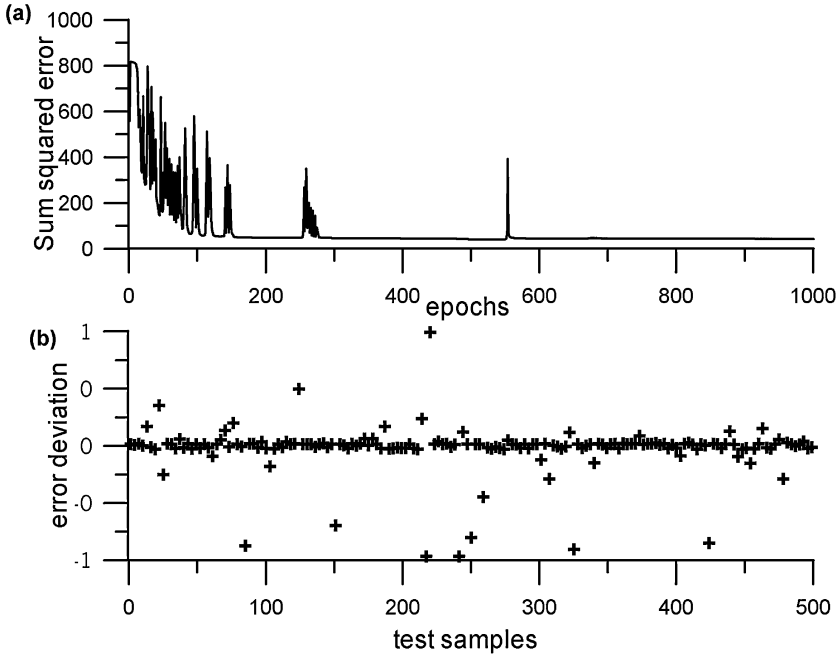

Figure 5. Results from the pattern training. (a) The SSE as a function of epoch. The error reached the minimum floor after 400 epochs. The stability of convergence is demonstrated up to 1000 epochs. (b) The deviation between manually classified and network-predicted classes for 500 test time-series segments. The scattered points show the major deviations. The correct picking constitutes 94 per cent.

neural network training, more importance is given to how the network performs on novel (non-training) data than the SSE of the training itself. FANN on test samples (samples that were not used for training) gave 94 per cent (472/500) correct classification (Fig. 5b) with $\eta = 0.09$, $\beta = 0.1$ and 10 hidden neurons. As observed by Dai & MacBeth (1995) although this solution was considered optimal for the current application, further architecture optimization could undoubtedly be achieved by a more exhaustive search procedure on a more powerful computer.

4.2.4 Sensitivity analysis

The sensitivity of the neural network to the signal-to-noise ratio was examined using the following analysis. A time-series of 256 points with high signal content was mixed with a normally distributed random noise series. The trained neural network was assigned to classify the signal. In each run the signal content was changed with a small increment so that it covered the range 0–100 per cent. The network steadily gave values near to 0 (Fig. 6) until the signal content reached 60 per cent. The output changes to higher values as the signal content exceeds 60 per

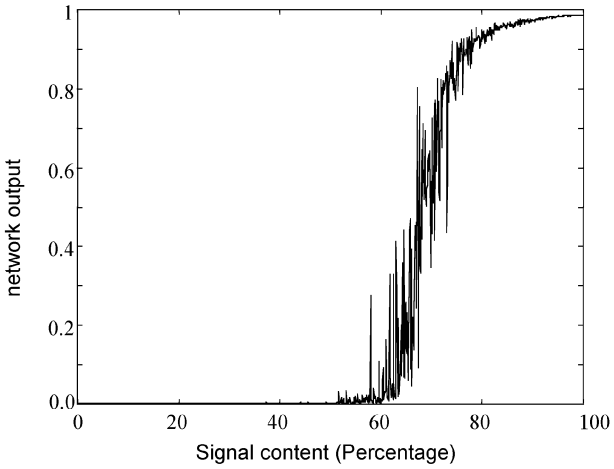

Figure 6. Network output versus signal content. The network was simulated by inputs with varying signal content. A narrow region of high variance exists when the signal content is between 65 and 75 per cent.

cent and asymptotes to 1 after it is 80 per cent. It is now evident that the network is able to detect signals if the signal content is more than 70 per cent. Another interesting aspect is the narrow region of high variance (when the signal content is between 65 and 75 per cent) in the output. A thorough analysis with noise of other distributions was beyond the scope of the present study.

4.3 Interchannel training

Here, a separate neural network was trained with the interchannel parameters (explained in Section 4.1). As we wanted to keep this as the final step in determining the overall quality of the stack, we also included the five pattern quality values predicted by the first step as an input. The inputs to the network were:

(1) five pattern quality values from the earlier network;
(2) two amplitude ratios (E_x/H_y and E_y/H_x); and
(3) two correlation parameters (E_x–H_y and E_y–H_x).

4.3.1 The data

To train the network, 900 stacks were selected from different data sets (five channels, with 256 data points each). Care was taken to include an equal number of signal and noise segments. Each stack was manually inspected to assign a class vector 0.1 or 0.9 depending on its overall signal quality, as explained previously. A training database was constituted as follows.

(1) We used the trained FANN to classify the *patterns* of the five simultaneously measured channels within each stack. The output of this processing varied between 0.0 (bad) and 1.0 (good). Fig. 7(a) shows the pattern classes for E_x and E_y channels versus the stack numbers. The broken bar below the graph indicates the manually assigned stack class (black good; white, bad). An excellent correlation between the manually assigned stack class and the pattern class of individual channels computed from FANN processing is evident. This justifies our earlier comment that signal discrimination largely depends on the pattern of time-series signals.

(2) As the *amplitude ratios* between the orthogonal electric and magnetic field were found to be another signal discriminator, the two ratios were included in the training database. Furthermore, in order to keep the values within 0 and 1.0 (as necessitated by the sigmoid function) they were normalized as

$$A_{ExHy} = \frac{E_x/H_y}{E_x/H_y + E_y/H_x} \tag{12}$$

and

$$A_{EyHx} = \frac{E_y/H_x}{E_x/H_y + E_y/H_x}, \tag{13}$$

where E_x, E_y, H_x and H_y are simple ranges of amplitude (maximum–minimum) of respective channels for a stack. As plotted in Fig. 7(b), the ratios vary considerably and a direct correlation with the signal class is impossible. We retained this parameter for training, as it adds another dimension to the input data.

(3) *Correlation* coefficients between orthogonal electric and magnetic fields (E_x–H_y and E_y–H_x) were calculated for each stack. The correlation coefficient $r(\tau)$ (Molyneux & Schmitt 1999) between two vectors X and Y ($t = 0, 1, 2, \ldots, n$) is given by

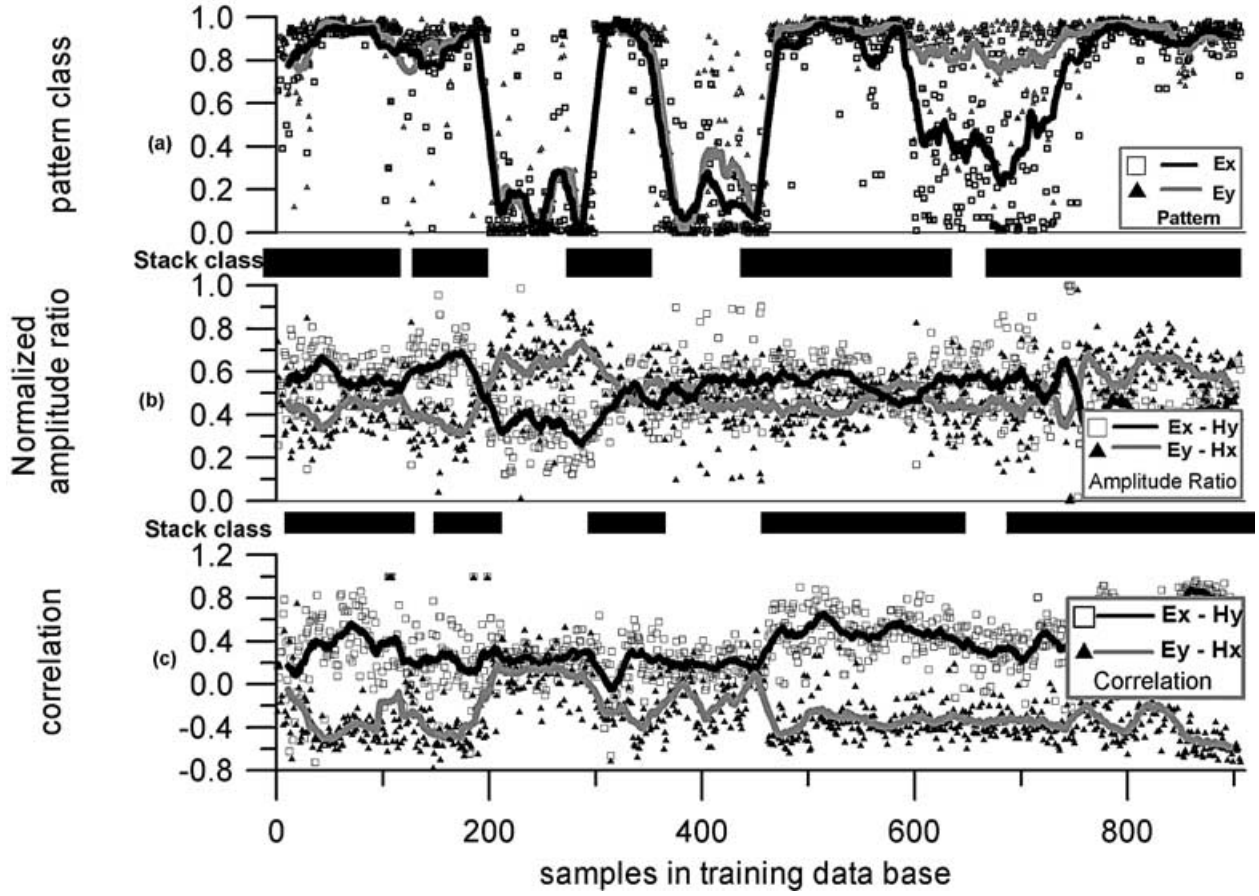

Figure 7. Pattern, amplitude ratios and correlation coefficients of 900 stacks that form the database for interchannel training and testing. The thick line is the running average over 10 points. The broken bar indicates the overall stack quality: black good (0.9) and white bad (0.1). (a) E_x (squares) and E_y (triangles) pattern quality predicted as a function of the stack number. (b) E_x–H_y (squares) and E_y–H_x (triangles) amplitude ratio. (c) The correlation coefficients of E_x to H_y (squares) and E_y to H_x (triangles).

$$r(\tau) = \frac{n\sum_{t=0}^{t=n} X(t)Y(t+\tau) - \sum_{t=0}^{t=n} X(t)\sum_{t=0}^{t=n} Y(t+\tau)}{\sqrt{n\sum_{t=0}^{t=n} X(t)^2 - \left[\sum_{t=0}^{t=n} X(t)\right]^2}\sqrt{n\sum_{t=0}^{t=n} Y(t+\tau)^2 - \left[\sum_{t=0}^{t=n} Y(t+\tau)\right]^2}}. \tag{14}$$

A value of $r = 1$ indicates perfect positive correlation between two vectors; $r = 0$ indicates no correlation, meaning the vectors are not similar; $r = -1$ indicates anticorrelation (meaning two vectors are of same shape but of opposite polarity). The correlation coefficients for all the 900 windows are plotted in Fig. 7(c). The squares represent E_x–H_y correlations and diamonds represent the E_y–H_x correlations. Most of the E_x–H_y coefficients are distributed between 0.2 and 0.6, whereas the E_y–H_x distribution is between -0.2 and -0.6. It can be clearly seen that the signal class is 0.9 (a good quality signal) when E_x–H_y and E_y–H_x coefficients are well separated (when they are closer to ± 1).

4.3.2 FANN training

The training database thus prepared was a 9×900 matrix. The rows were shuffled to produce a random distribution. Around 70 per cent of the database was used for training the neural network and the remaining 30 per cent was used for testing. As the input vector is only of length 9, the training procedure was rather simple compared with the pattern training. A three-layer feed-forward network was trained with nine neurons in the input layer, two nodes in the hidden layer and one node in the output layer. The momentum gain β was set to 0.1 and we experimented with the learning rate (η) and the number of hidden-layer nodes. Each training consisted of 5000 epochs and took 7–10 min to complete. As the training progressed, the number of hidden layer nodes was reduced to one. The training stopped when the sum squared error was 7.9 for 650 samples with $\eta = 0.09$. Fig. 8(a) shows the SSE as a function of epoch for the final training. On simulating the network using the test data, SSE was 3.72 (i.e. 0.1222 per vector). Fig. 8(b) shows the deviation of the network-predicted signal class values from the manually classified values. It can be seen that the network classification was successful.

4.3.3 Relative significance of the input

The network was fully trained to classify MT signals. To find the relative importance of the nine inputs (five-pattern classes, two correlations and two amplitude ratios) to the FANN, a validation training was carried out. On each run, the particular input of interest was set to nil

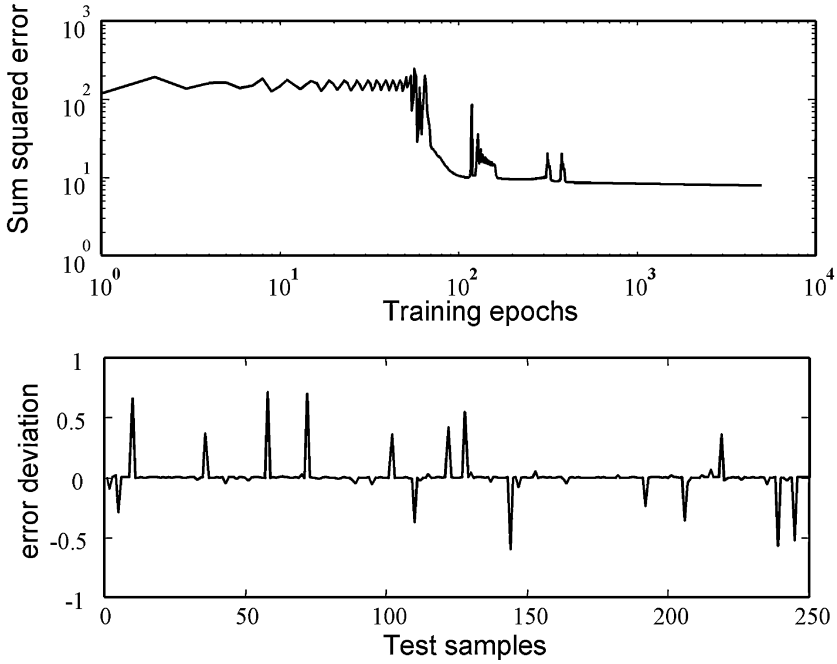

Figure 8. Results from interchannel training. (a) The SSE as a function of the training epoch. (b) Deviation between manually classified stack quality and the network-predicted quality for 250 stacks. 235 stacks were classified similarly to manual classification.

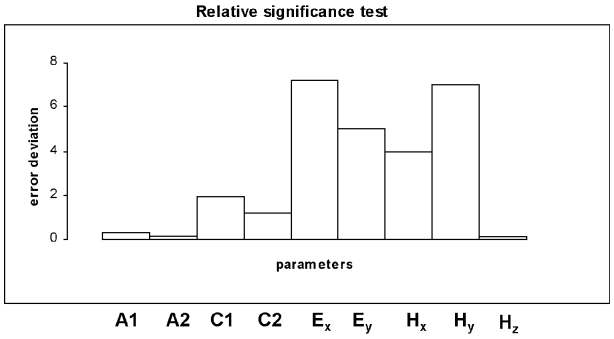

Figure 9. Relative significance of various inputs to the networks, namely amplitude ratios (A1 and A2), correlation coefficients (C1 and C2) and five-pattern qualities (E_x, E_y, H_x, H_y and H_z). The error deviation against each input is a measure of its significance to the neural network.

throughout the run. This resulted in a greater error compared with the original training using non-zero inputs. The extent of departure from the previous error is considered as the relative significance (Altinay *et al.* 1997) of that particular point for the neural network. As can be seen in Fig. 9, patterns of E_x, H_y, E_y and H_x are dominant in the neural network response, followed by the correlation parameters. The pattern of H_z and amplitude ratios have the least influence on the neural network. This roughly agrees with our earlier classification of the factors influencing editing decisions, but with two surprises. The first is the relative insignificance of E_y as compared with E_x. As the database for training was equally biased to all five channels, it was not the result of faulty training. The excellent performance of the neural network on test data also proves this. Second is the low influence of the amplitude parameters. While formulating the problem, we gave both amplitude and correlation equal weight, but the training results disprove it. As the testing sessions proved the discrimination capability of FANN, we stopped further training.

5 APPLICATION

We applied the FANN-based signal detection scheme to four MT long-period time-series sets collected in different noise environments (G12, VP12, JN10 and TT8). Station G12 is the least contaminated with noise, followed by VP12 and JN10. Station TT8 is situated in a major industrial belt in South India (Harinarayana *et al.* 2001) and is the most affected by noise. All four data sets were subjected to three types of editing: (1) blind editing, (2) FANN-based editing and (3) manual editing by a third person. Matrices of auto- and cross-spectra were computed for each target frequency over a small frequency band. To compute average MT transfer functions from the set of available observations, we used the coherence between measured and predicted channels as a discriminator. An acceptance factor of 0.7 was used, which resulted in selecting 70 per cent of the available matrices, in the order of highest coherence. The choice of the somewhat old-fashioned algorithm is

justified, as it allowed us to show the efficacy of neural-network-based editing. Moreover, the algorithm does not interfere too much with the data selected. MT apparent resistivity (ρ) and phase (ϕ) derived from the three modes of editing are plotted for comparison.

G12:The station was located over the basaltic province in western India. 192 stacks were available for processing. The time-series was generally noise-free with long-period geomagnetic pulsations (sinusoids). The occasional spikes were smaller than the waveforms within which they occur. Fig. 10(a) shows MT apparent resistivity and phase computed from all available stacks (192). The curve is quite smooth and without deviation, suggesting that the time-series were relatively noise-free. Neither FANN-based editing (selected 157 stacks) nor manual editing (selected 150 stacks) improved the curve significantly. Results are given in Figs 10(b) and (c). A small number of noisy segments were easily rejected by the coherence-based estimator, without any need of editing.

VP12: This station was located in the granulite province of South India. A total of 352 stacks were recorded. Almost half of the stacks carried spikes and step-like features originating from submersible electric pumps and switching of power supplies. The MT apparent resistivity and phase computed from all the stacks are given in Fig. 11(a). The resistivity (ρ) values, especially between 1.0 and 0.1 Hz are scattered and the phase (ϕ) is poorly resolved, with both *xy* and *yx* modes being equally affected. Fig. 11(b) shows the results of FANN-based editing

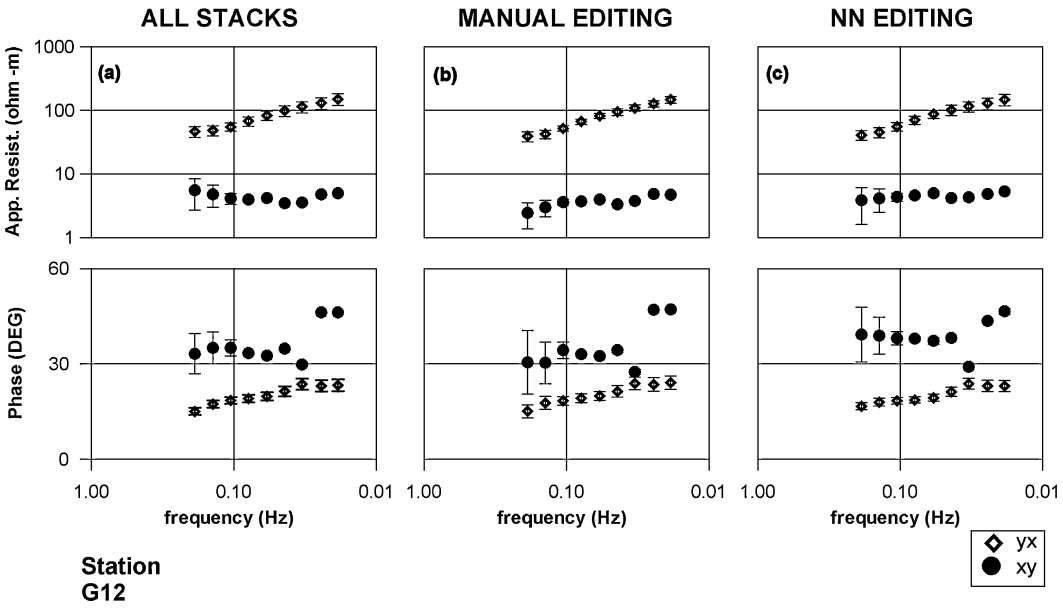

Figure 10. Comparison of MT apparent resistivity and phase computed from different modes of editing of data from site G12. Filled circles represent *xy* and diamonds represent *yx* components. (a) Using all stacks available. (b) By neural network editing. (c) By manual editing.

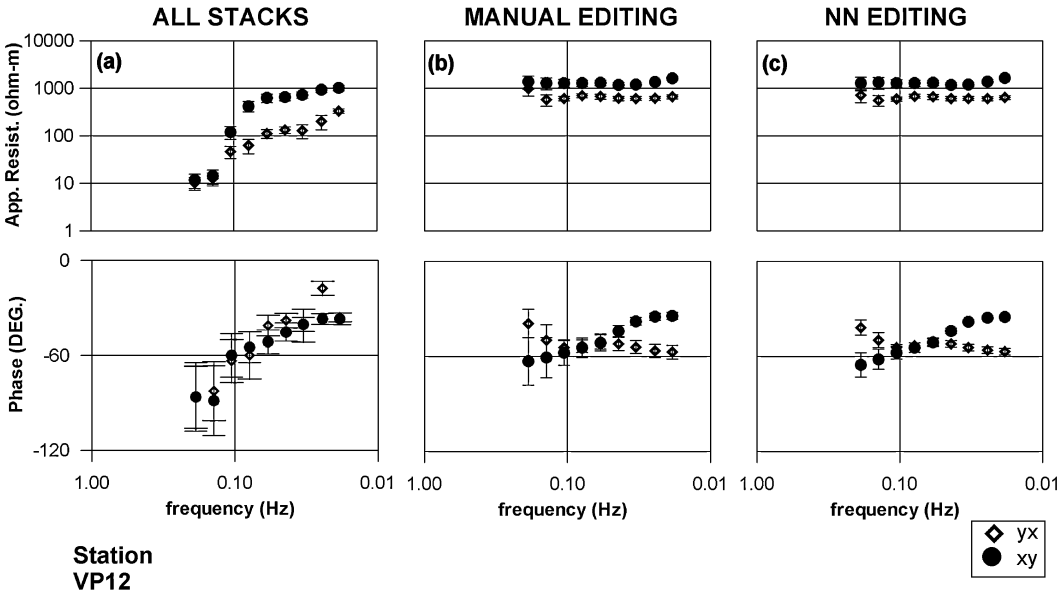

Figure 11. Comparison of MT apparent resistivity and phase computed from different modes of editing of data from site VP12. Filled circles represent *xy* and diamonds represent *yx* components. (a) Using all stacks available. (b) By neural editing. (c) By manual editing.

(140 stacks selected), where, both apparent resistivity and phase are better resolved. An improvement is clearly evident in the 1.0–0.1 Hz range. Manual editing resulted in selecting 127 stacks and the computed apparent resistivity and phase are similar to FANN editing (Fig. 11c).

JN10: this site was also located in the same region as VP12, but with more noise in the data. The effect of the noise is evident in the apparent resistivity curves computed from all the stacks (Fig. 12a). The phase is also affected in the range 1–0.1 Hz. The neural editing picked 80 stacks out of 256 available and gave a better estimate, as shown in Fig. 12(b). A less rigorous manual editing picked 107 stacks and gave a similar result (Fig. 12c).

TT8: This station was the most affected by noise. Both electric channels, especially E_x, were affected by noise originating from an industrial belt nearby. Increased spike activity was observed in all channels. The MT apparent resistivity and phase computed (Fig. 13a) from all 256 stacks available gave a very distorted picture. Both *xy* and *yx* components were poorly resolved. Significant improvement was made by FANN editing (Fig. 13b), which selected 67 stacks out of 256. The *yx* component is now smooth and relatively error-free. The *xy* component

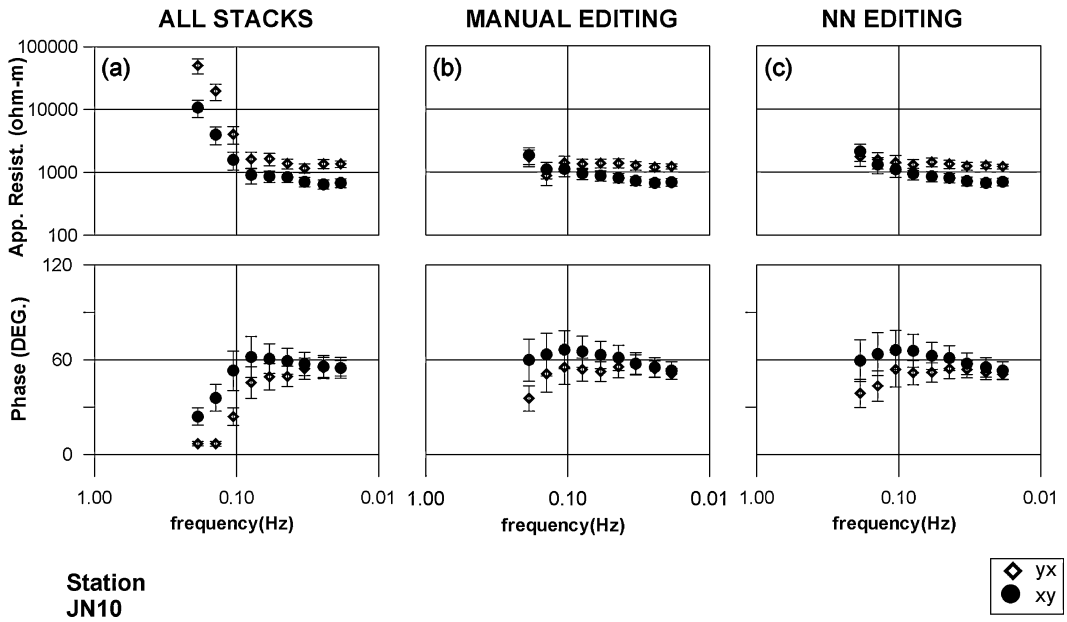

Figure 12. Comparison of MT apparent resistivity and phase computed from different mode of editing of data from site JN10. Filled circles represent *xy* and diamonds represent *yx* components. (a) Using all stacks available. (b) By neural editing. (c) By manual editing.

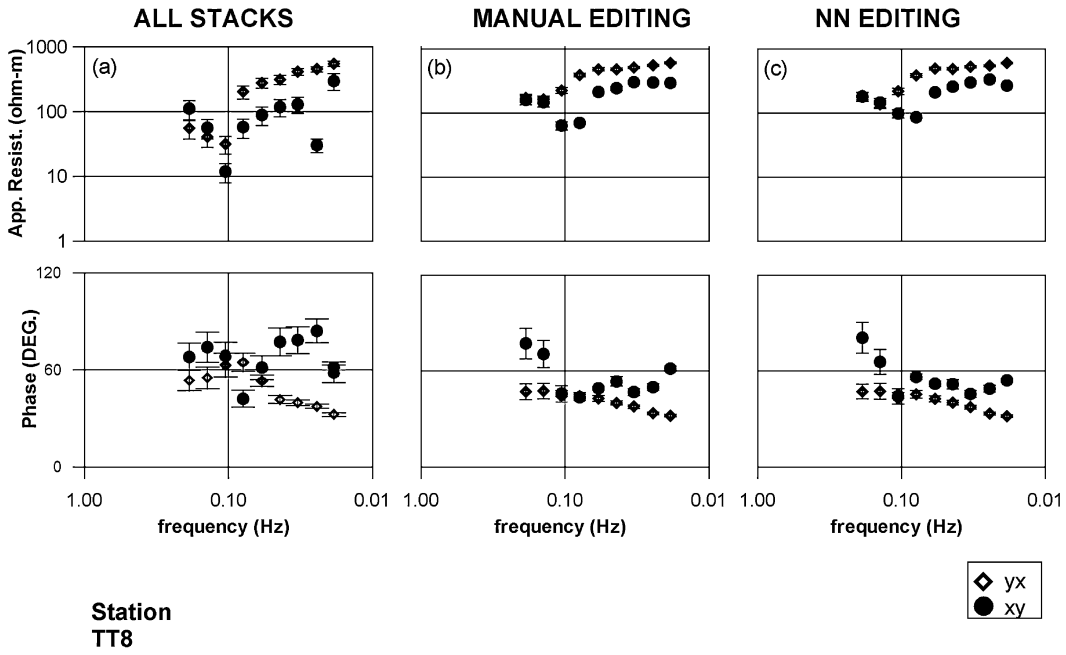

Figure 13. Comparison of MT apparent resistivity and phase computed from different mode of editing of data from site TT8. Filled circles represent *xy* and diamonds represents *yx* components. (a) Using all stacks available. (b) By neural editing. (c) By manual editing.

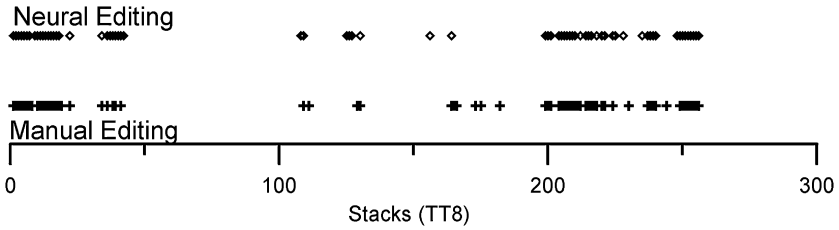

Figure 14. Comparison of manual and neural signal picking for site TT8. The diamonds present the neural picking and crosses the manual ones.

is also improved except near 0.1 Hz. Almost the same result is produced by manual editing (69/256) as shown in Fig. 13(c). The stacks picked by neural network and manual editing are compared in Fig. 14. Overall the pickings match quite well. Deviation between the two editing schemes is evident for a few picks between stack numbers 150–180. Between these there are stacks with moderate to low signal content, which were accepted by manual editing but rejected by neural editing.

6 DISCUSSION AND CONCLUSION

The application of FANN-based editing to magnetotelluric time-series has shown interesting results. In a low-noise environment such as station G12, the network editing produces results almost similar to blind editing using all stacks. On these data, a simple coherence-based estimator produces satisfactory results. The advantage of FANN-based editing is demonstrated in moderate to high-noise environments such as stations VP12, JN10 and TT8. The inherent ability of a neural network to pick out signal patterns in high-noise environments is demonstrated here. In such cases it approximates human intelligence, established from the fact that the neural-network-based editing gives a result similar to manual editing. These results satisfy the objective of the current work, i.e. to provide a robust alternative to manual editing of magnetotelluric time-series. However, different norms of signal classification are needed in each band of measurement. Based on this experiment, we believe that neural network editing has the potential to automate MT time-series editing, efficiently and reliably. The scheme could be adapted to cover the entire MT frequency range and to be incorporated into routine processing to save human time and increase reliability in editing. The current scheme performs approximately 70 000 floating-point operations (flops) per stack for classification. An FFT of five channels alone uses 40 000 flops. If the further computation uses the same spectra, some of the computational redundancy can be removed. It points to the possibility of integrating FANN-based editing into real-time processing of MT data.

ACKNOWLEDGMENTS

The Director, NGRI, India is gratefully acknowledged for his permission to publish this paper. The authors wish to thank S. V. S. Sarma and T. Harinarayana for their encouragement and for the useful discussions. Suggestions of P. S. Moharir, especially on the optimization of network parameters were crucial in developing the editing scheme. Reviews by John Stodt and Xavier Garcia greatly improved the submitted version. Editorial remarks by Martyn Unsworth also contributed to the improvement. Colleagues in the MT group, NGRI are thanked for help during the field campaign. MT data from the integrated studies over Southern Granulite Terrain, sponsored by the Deptartment of Science and Technology, Government of India (project no ESS/16/118/1997) were used in this study.

REFERENCES

Altinay, O., Tulunay, E. & Tulunay, Y., 1997. Forecasting ionospheric critical frequency using neural networks, *Geophys. Res. Lett.,* **24,** 1467–1470.

Banks, R.J., 1998. The effects of non-stationary noise on electromagnetic response estimates, *Geophys. J. Int.* **135,** 553–563.

Boadu, F.K., 1998. Inversion of fracture density from field seismic velocities using artificial neural networks, *Geophysics,* **63,** 534–545.

Cagniard, L., 1953. Basic theory of the magneto-telluric methods of geophysical prospecting, *Geophysics,* **18,** 605–635.

Calderon-Macias, C., Sen, M.K. & Stoffa, P.L., 2000. Artificial neural networks for parameter estimation in geophysics, *Geophys. Prospect.,* **48,** 21–47.

Chave, A.D. & Thompson, D.J., 1989. Some comments on magnetotelluric response function estimation., *J. geophys. Res.,* **94,** 14 215–14 225.

Chave, A.D., Thomson, D.J. & Ander, M.E., 1987. On the robust estimation of power spectra, coherencies and transfer functions, *J. geophys. Res.,* **92,** 633–648.

Dai, H. & MacBeth, C., 1995. Automatic picking of seismic arrivals in local earthquake data using an artificial neural network, *Geophys. J. Int.,* **120,** 758–774.

Egbert, G.D., 1997. Robust multiple station magnetotelluric data processing, *Geophys. J. Int.,* **130,** 475–496.

Egbert, G.D. & Booker, J.R., 1986. Robust estimation of geomagnetic data processing, *Geophys. J. R. astr. Soc.,* **87,** 173–194.

Fish, B.C. & Kusuma, T., 1994. A neural network approach to automate velocity picking, in *Society of Exploration Geophysicists, 64th Annual International Meeting; Technical Program, Expanded Abstracts with Authors' Biographies,* **64,** 185–188.

Gamble, T.D., Goubau, W.M. & Clarke, J., 1979. Magnetotellurics with a remote reference, *Geophysics,* **44,** 53–68.

Garth, L.M. & Poor, H.V., 1994. Detection of non-Gaussian signals: a paradigm for modern statistical signal processing, *Proc. IEEE,* **82,** 1060–1095.

Harinarayana, T. *et al.*, 2001. Magnetotelluric investigations, in *Modeling the Tectonic Evolution of Southern Granulite Belt of the Indian Shield using Coincident Seismic Reflection/Refraction, Geological, Geochemical, Geochronological, Gravity/Magnetic, Magnetotelluric and Deep Resistivity Studies Along the Southern Geotransect,* Technical report No. NGRI-2001-EXP-317, 84–111.

Hatting, M., 1989. The use of data-adaptive filtering for noise removal on magnetotelluric data, *Phys. Earth planet. Inter.,* **53,** 239–254.

Junge, A., 1996. Characterization of and correction for cultural noise, *Surv. Geophys.,* **17,** 361–391.

Koons, H.C. & Gorney, D.J., 1991. A neural network model of the relativistic electron flux at geosynchronous orbit, *J. geophys. Res.,* **96,** 5549–5556.

Larsen, J.C., Mackie, R.L., Manzella, A., Fiordelisi, A. & Rieven, S., 1996. Robust smooth magnetotelluric transfer functions, *Geophys. J. Int.,* **124,** 801–819.

Lundstedt, H., 1996. Solar origin of geomagnetic storms and prediction of storms with the use of neural networks. *Surv. Geophys.,* **17,** 561–573.

Luo, F. & Unbehauen, R., 1997. *Applied Neural Networks for Signal Processing,* Cambridge University Press, Cambridge, p. 123.

McCormack, M.D., Zaucha, D.E. & Dushek, D.W., 1993. First-break refraction event picking and seismic data trace editing using neural networks, *Geophysics,* **58,** 67–78.

Molyneux, J.B. & Schmitt, D.R., 1999. First-break timing: arrival onset times by direct correlation, *Geophysics,* **64,** 1492–1501.

Murat, M.E. & Rudman, A.J., 1992. Automated first arrival picking; a neural network approach, *Geophys. Prospect.,* **40,** 587–604.

Oettinger, G., Haak, V. & Larsen, J.C., 2001. Noise reduction in magnetotelluric time-series with a new signal–noise separation method and its application to a field experiment in the Saxonian Granulite Massif, *Geophys. J. Int.,* **146,** 659–669.

Orange, A., 1989. Magnetotelluric exploration for hydrocarbons, *Proc. IEEE,* **77,** 287–317.

Park, J. & Chave, A.D., 1984. On the estimation of magnetotelluric response functions using the singular value decomposition, *Geophys. J. R. astr. Soc.*, **77,** 683–709.

Poulton, M., Stenberg, B. & Glass, C., 1992. Location of subsurface targets in geophysical data using neural networks, *Geophysics,* **57,** 1534–1544.

Santarato, G. & Spagnolini, U., 1995. Canceling directional EM noise in magnetotellurics, *Geophys. Prospect.,* **43,** 605–621.

Spichak, V. & Popova, I., 2000. Artificial neural network inversion for magnetotelluric data in terms of three-dimensional earth macroparameters, *Geophys. J. Int.,* **142,** 15–26.

Werbos, P.J., 1990. Backpropagation through time: what it does and how to do it, *Proc. IEEE,* **78,** 1550–1560.

Winer, J., Rojers J.A., Rijers J.R. & Moll, R., 1991. Predicting carbonate permeabilities from wireline logs using back propagation networks, 61st SEG meeting, Houston, Expanded abstract, pp. 285–288.

Zhang, Y. & Paulson, K.V., 1997. Magnetotelluric inversion using regularized Hopfield neural networks, *Geophys. Prospect.,* **45,** 725–743.

Zhao, Y. & Takano, K., 1999. An artificial neural network approach for broadband seismic picking, *Bull. seism. Soc. Am.,* **89,** 670–680.