Split shear-wave analysis using an artificial neural network?

Hengchang Dai[1,2] and Colin MacBeth[1]

Introduction

Artificial neural networks (ANNs) are simple models that attempt to simulate the operation of neurons in the brain. Although ANNs are relatively new in seismology, their origins can be traced back to the 1940s when psychologists began developing models of human learning. One of the most exciting developments in ANNs was the advent of the *Perceptron*, the idea that a network of elemental processors arrayed in a manner reminiscent of biological neural networks might be able to learn how to recognize and classify patterns in an autonomous manner. However, in 1969, Marvin Minsky, one of the founding fathers of artificial intelligence, proved mathematically that perceptrons were incapable of solving many simple problems. After over a decade of being in the scientific wilderness, ANNs have once again become a popular tool for many applications requiring algorithms with pattern recognition capability as those mathematical difficulties have been overcome by the introduction of more complex neural network architectures in the 1980s. These new network designs offer increased flexibility and robustness. They are particularly attractive as, unlike conventional methods that incorporate a fixed algorithm to solve a particular problem, ANNs utilize a learning scheme to develop an appropriate general solution, making them flexible and adaptive to different datasets. ANNs are now starting to be used in seismic applications for event picking, or correlation of seismic horizons with sparse well-log data, and are likely to find other applications of great value.

Here, we explore a new area where ANNs may prove of benefit: the recognition of the vector motion of seismic waves. For convenience we will confine ourselves to the identification of the particular patterns of the particle motion recorded in the horizontal plane, which arise from split shear-waves: the interfering fast shear-wave *qS1* and slow shear-wave *qS2*, and investigate whether the pattern recognition capability is sufficiently powerful to be used to extract their characteristic parameters. Recognition and quantification of these split shear waves in seismic data can provide information about the effective anisotropy of the rockmass from which information regarding the distribution of internal heterogeneity may be determined. This does not restrict the more general relevance of our conclusions to other types of vector particle motion. Our incentive is that the visual analysis of such waves is time-consuming and individual recordings are more difficult to interpret. It is usual to group the recordings according to the acquisition geometry to make the measurements more robust. However, even experienced interpreters may still be subjective and inconsistent in their results and human investigation is still required to inspect the data initially and perform the pre-conditioning. Here, we concentrate on two parameters for estimating this splitting: the polarization direction of the faster shear-wave and the time-delay between split shear-waves. Can the neural network, a natural choice for this type of problem, satisfactorily distinguish between different splitting characteristics, and what parts of the particle motion does it weight in order to do this?

What are artificial neural networks?

In the brain, a neural cell, or neuron, receives input from many other neurons via interconnections called axons. If the energy level of the combined input exceeds a threshold level, then the neuron transmits an output electrical signal depending on its active function to other neurons through electrical and chemical transport mechanisms. The output signal strength is modified by a special connection called a synapse before entering another neuron. It is believed that certain forms of learning occur when the synapses are trained to assume certain strength or weights by repeated exposure to the same stimulus. An ANN consists of nonlinear processors called nodes (corresponding to neurons) linked to each other by interconnections analogues to the axons. Each interconnection has an associated scalar weight (corresponding to a synapse) whose value can be modified during the learning phase of the neural network.

Although the concept of a neural network appears initially to be quite straightforward, there is a bewildering array of different kinds of ANNs which now exist to solve different problems; Fig. 1 gives a classification.

‘British Geological Survey, Murchison House, West Mains Road, Edinburgh EH9 3LA, Scotland, UK.
[2]Department of Geology and Geophysics, University of Edinburgh, West Mains Road, Edinburgh EH9 3JW, Scotland, UK.

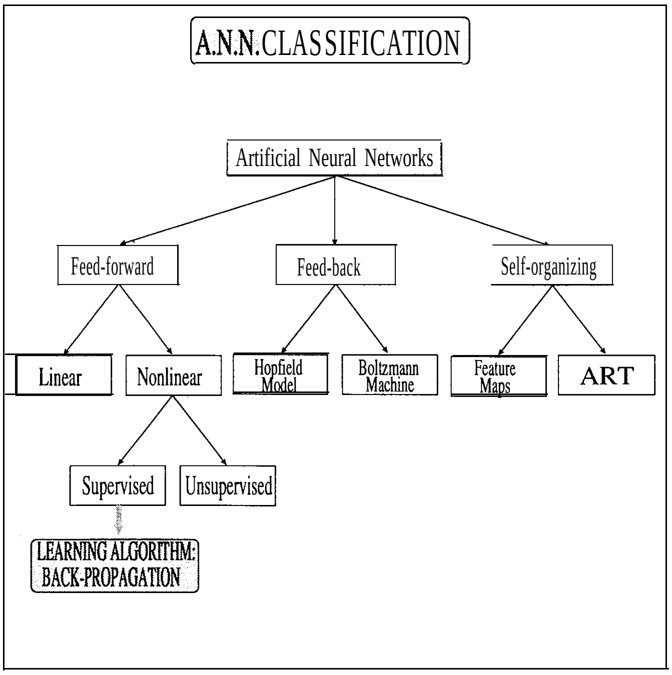

Fig. 1. General artificial neural network classification. The neural networks are divided into three main categories: feed-forward, feed-back and self-organization neural networks. In each category, there are many kinds of neural network, but we only list two to demonstrate the classification.

Different ANNs have different topological structures for the neurons and their connections, and use different learning algorithms based on different philosophies. The main categories for this classification are listed below.

- Feed-forward neural networks-the output of each layer feeds the next layer of units;
- Feed-back neural networks-the input information defines the initial activity state of a system. The first output of the system is taken as the new input, which produces a new output and so on;
- Self-organizing neural networks-where neighbouring cells compete in their activation by means of mutual lateral interactions, and develop adaptively into specific detectors of different signal patterns.

Each of them has applications in solving different types of problem. The ANN used in this study is defined as nonlinear, multilayer, and feed-forward, being trained by the back-propagation of error (Rumelhart *et al.* 1986). This is the most popular type of ANN in use for pattern recognitions, as it is well understood and appears quite versatile. ANNs are used to solve a diversity of geological and geophysical problems. For example, Palaz and Weger (1990) use ANNs to recognize waveforms in seismic data. ANNs have also been used to combine synthetic spontaneous potential (SP) and resistivity logs (MacCormack 1991) to estimate lithology logs. Poulton *et al.* (1992) used ANNs to estimate offset, depth, and conductivity-area product of a conductive target given an electromagnetic image of the target. ANNs also have been used in the first-break picking of surface seismic data (Murat and Rudman 1992; McCormack *et al.* 1993). Moreover, other kinds of neural networks are also used in seismic event classification (Leggett *et al.*, 1993). This kind of network represents our first and simplest choice of network architecture to solve our problem; will it be successful?

Our chosen network is composed of sets of *nodes* arranged in *layers*: one input layer, two intermediate hidden layers, and one output layer. This architecture is illustrated schematically in Fig. 2, where we have drawn nodes as solid circles and interconnections as straight lines. The node forms the basic processing unit of the network, and the various components of this unit are shown in Fig. 3, alongside a real neuron of the human brain. The first (input) layer just passes the signals to the various nodes in the second layer through the interconnections. The nodes in the hidden layers and the output layer receive and sum the signals and output according to their activation functions. As not all nodes need be given the same priority, a weighting is applied to each input signal before a summation procedure. These weights are applied as simple multiplicative scalars and

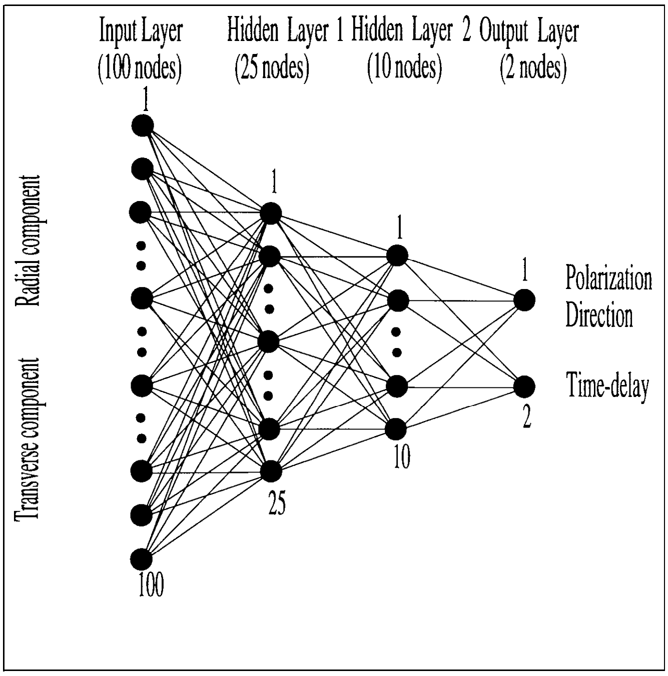

Fig. 2. Neural network structure used for analysing shear-wave splitting in VSP data. Four layers are used: the first layer has 100 nodes and inputs the radial and transverse components in the order of $r_1, r_2, \ldots, r_{50}, t_1, t_2, \ldots, t_{50}$. There are 25 and 10 nodes in the first and second hidden layers respectively. Only two output nodes are present, corresponding to the polarization and time-delay estimates.

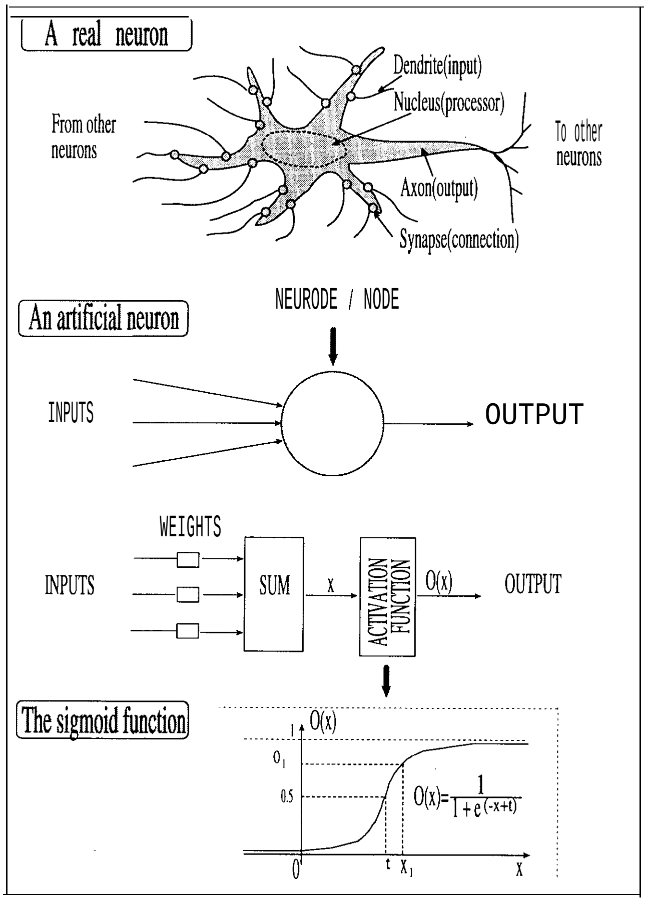

Fig. 3. Schematic diagram of various constituents making up an artificial node in a neural network, compared to a real neuron in the human brain. The symbol, x represents the summed input to the node, *t* the threshold, and *O*(*x*) the output of the node after being conditioned by a non-linear activation function.

effectively amplify or attenuate the signals. This weighting simulates synaptic connections to the neuron body. With the exception of the nodes in the input layer, the net input to each node is the sum, x, of the weighted outputs of nodes in the previous layer. This node transmits an output signal, $O(x)$, to other nodes when the energy level of the combined input to a particular node exceeds a threshold level, t, depending upon the characteristic activation function. The activation function may assume some smooth S-shape, which may be represented mathematically by the sigmoidal function (although other functions are available). The outputs of each node in the input layer is a component of the input pattern.

In our example we input two horizontal recordings of seismic shear waves into the network and output the values of two parameters: ϕ, the polarization of the fastest shear-wave, and $\Delta\tau$, the differential travel-time of the split shear-waves. No assumption is made about the polarization of the slower wave, the differential amplitude, or the source wavelet shape. The input layer has 100 nodes (50 for radial component input and 50 for transverse component input), the first hidden layer 25 nodes, the second hidden layer 10 nodes, and the output layer 2 nodes. This provides a window size of 196ms, with 4ms sample intervals for the data. The actual numbers of hidden layers and nodes in each layer are somewhat arbitrary, but do depend on external constraints from the physical problem such as the number of input and output nodes and also on the desired system error, pattern error and the nature of the training samples. There is no fixed relationship between these various factors for this type of neural network, although there are certain other network designs where this is the case (Kusuma and Brown 1992). We are guided by the knowledge that generalization is increased and memory is reduced by limiting the number of hidden nodes (Dowla *et al.* 1990). Too few hidden nodes will lead to a long learning process or no convergence. Too many hidden nodes will introduce noise in the output. In this experiment, our final choice was made after trial and error with different configurations.

The choice of a programming language is more critical than in other situations due to the computation demands of neural networks. Whatever language is used, it is advisable to seek a version that has been optimized for numeric data. The C language has become something of a de facto standard for neural network programming. Our neural network was set up using a C program written on a VAX4000 under VAX/VMS^{TM} operation system.

Training the network

In our application, the ANN may be likened to an analyst, and must be trained by presenting many different patterns of shear-wave particle motion recorded by orthogonal horizontal geophones with known parameters. In our case these are provided from synthetic seismograms, but they could also be field data interpreted in an analyst's style or, ideally, several analysts' styles. After training is accomplished, it is hoped that the ANN can estimate the splitting parameters from a variety of field VSP seismograms, consistently mimicking the expert interpretation. The ability of a neural network to recognize new patterns depends on the training patterns used as references. The learning occurs when the weights are trained to assume certain values by repeated exposure to the same stimulus. The most popular way to adjust the weights and threshold values of the sigmoidal functions is to use a back-propagation learning algorithm, or *Delta Rule.* A good mathematical description of this is given by Pao (1988). This method attempts to find the most suitable solution (numerical values of thresholds and weights) for global minimization of the mismatch between the desired output pattern and its actual value for all of the training examples. The degree of mismatch for each input-output pair is quantified by solving for the unknown parameters between the hidden and output layer and then by propagating the mismatch backwards through the network to adjust the parameters between the input layer and hidden layer. The first input pattern is presented to an initially randomized network, and the weights and thresholds adjusted in all the links. Other patterns are then presented in succession, and the weights and thresholds adjusted from the previously determined values. This process continues until all patterns in the training set are exhausted. It is generally accepted that this procedure is independent of the order in which the example patterns are presented. However, a final check can be performed by looking at the *pattern error* which is defined as the square of difference between desired output and neural network output for each pattern and the *system error* which is defined as the average of all pattern errors, to determine whether the final network solution satisfies all of the patterns presented to it within a certain error. The set of weights and thresholds in the network are now specifically tailored to 'remember' each input and output pattern, and can consequently be used to recognize or generate new patterns given an unknown input. The network is now trained, and can be used in subsequent analyses. Figure 4 summarizes the various stages of training and implementation.

The training patterns we used were generated from synthetic seismograms computed for a zero-offset VSP acquisition geometry using the anisotropic reflectivity method (Taylor 1992) because they were simple, regular, noiseless and give known splitting parameters. The horizontal X and Y recordings were computed for shear-waves generated by a horizontal point source along the Y direction, propagating through a uniform anisotropic half-space which simulates a range of *qS1* polarization directions from −X80°Y to X80°Y with an interval of 10°, and time-delays from Oms to 40ms in an interval of 4ms. We fed the seismograms, not their attributes, directly into the ANN as it is difficult to know which attributes give suitable information for solving the problem. The advantages of using seismograms is that we do not need to enforce a preconceived wavefield model. The disadvantage is that not all mathematical functions can be simulated by the ANN, and so we may exclude certain attributes. The radial component R(t) = $\{r_1, r_2, \ldots, r_{50}\}$ and transverse component T(t)= $\{t_1, t_2, \ldots, t_{50}\}$ were fed into the ANN in the order: $\{r_1, r_2, \ldots, r_{50}, t_1, t_2, \ldots, t_{50}\}$.

Although only 161 patterns in total were used in the training, the training procedure was slow (4507 iterations in about one and a half hours CPU time on a VAX4000), indicating that the global optimum may not be well-defined. The procedure did converge, however, to an acceptable system error of 8.02 x 10^{-6}, with all output patterns achieving an error of less than 10^{-4} with their ideal form. As a final check on this solution, the training data were fed back into the trained neural network. The resultant outputs indicated that all, except one pattern, lie within a tolerable maximum polarization error of 2°, and a maximum time-delay error of lms. A further check was made by inputing new synthetic data with polarizations differing by 5"; Fig. 5 displays a portion of these results. The results indicate that the network solution does appear internally consistent, and should in principle be capable of recognizing similar patterns in field data with unknown characteristics, but not lying too far from the experience of the network. One important advantage of ANNs is that once trained, they are very fast in generating the desired output. With

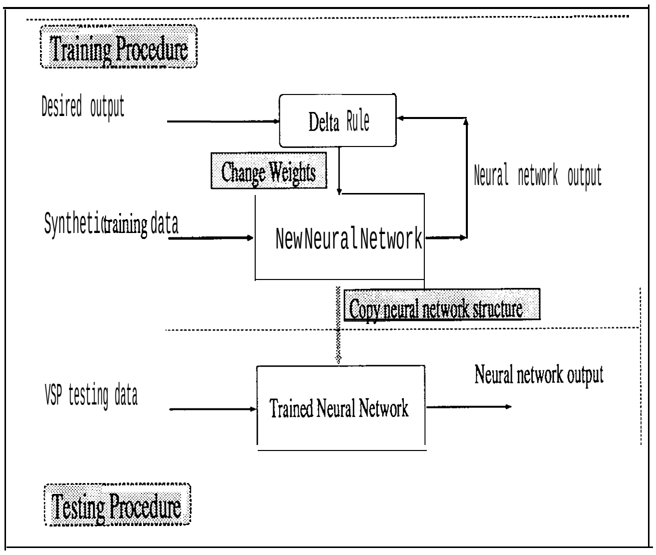

Fig. 4. Various component parts for neural network training and testing

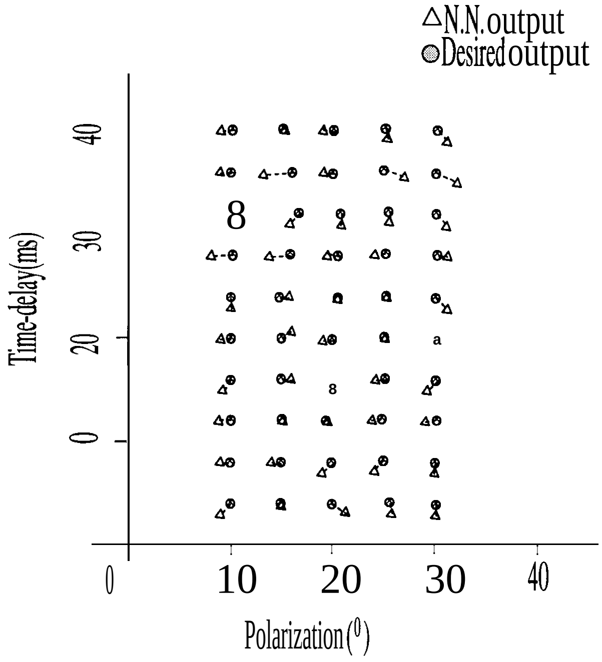

Fig. 5. Comparison of test results for trained network applied to various different synthetic VSP data, demonstrating an apparent internal consistency.

our particular solution, it took only 15 s of interactive CPU time to process the 161 training datasets.

Is the network successful with real data and what lessons can we learn?

Accepting that the trained neural network could perform its intended task on synthetic data, we now attempted to analyse real data. A three-component shear-wave VSP dataset generated by cross-line horizontal vibrators at a borehole site in the Paris Basin is chosen as it has been studied extensively (Cliet and Lefeuvre 1989). The depth surveyed in the borehole ranged from 1100m to 2060m with recording levels every 15m, and for our work we tested recordings of the 272m source offset made between 1325m and 2060m. The horizontal geophone components were re-oriented using the P-wave offset data prior to our study. The sampling rate is also reduced from 2ms to 4ms so that we could accommodate a large enough time-window (196ms) for the analysis. After this initial rearrangement of the data, the seismograms were processed by the neural network very quickly, taking only about seven seconds to process 50 recordings. The shear-wave parameters ϕ and $\Delta\tau$ output from the network compared favourably with those determined by the single source technique of Zeng and MacBeth (1993) (Fig. 6). A selection of five observed particle motions are shown in Fig. 7, alongside the corresponding synthetic motions chosen by the network as being the closest fit. Perhaps the most noticeable feature of these comparisons was that the direction of the initial onsets of the motions picked by visual examination did not correspond to either the picked polarization estimates for the observed data, or the actual polarization estimates for the observed data, or the actual polarization estimates for the synthetic data, whilst the network results agree with both. This appears to challenge our conceptual understanding of such motions: the initial onset should principally contain energy from the *qS1* wave and hence, depending upon the time-delay, give a direct indication of the polarization direction. In fact, it appears that the overall general character, including mainly the lobe of energy in the region of interference between *qS1* and *qS2*, is matched by the network and is the actual indicator for the choice of particle motion.

Given this disparity between visual examination and the network results, a further insight is required to view the mechanism by which the neural network performs its classification. This is achieved by investigating the magnitudes of the different weights in the completely trained network. Figure 8 displays weight maps of the neural network trained on the synthetic seismograms

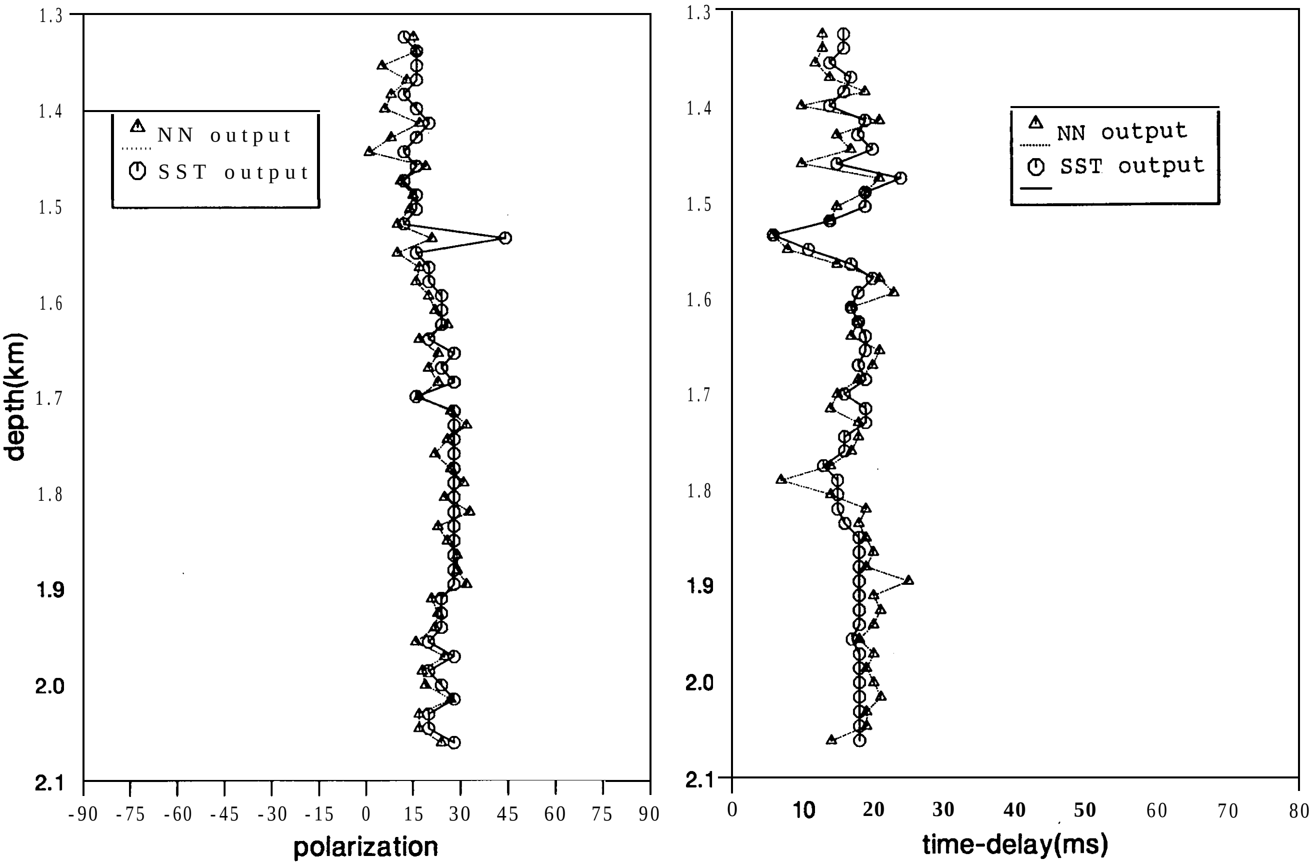

Fig. 6. Test results for Paris Basin VSP data compared with the results determined by the single source technique (SST).

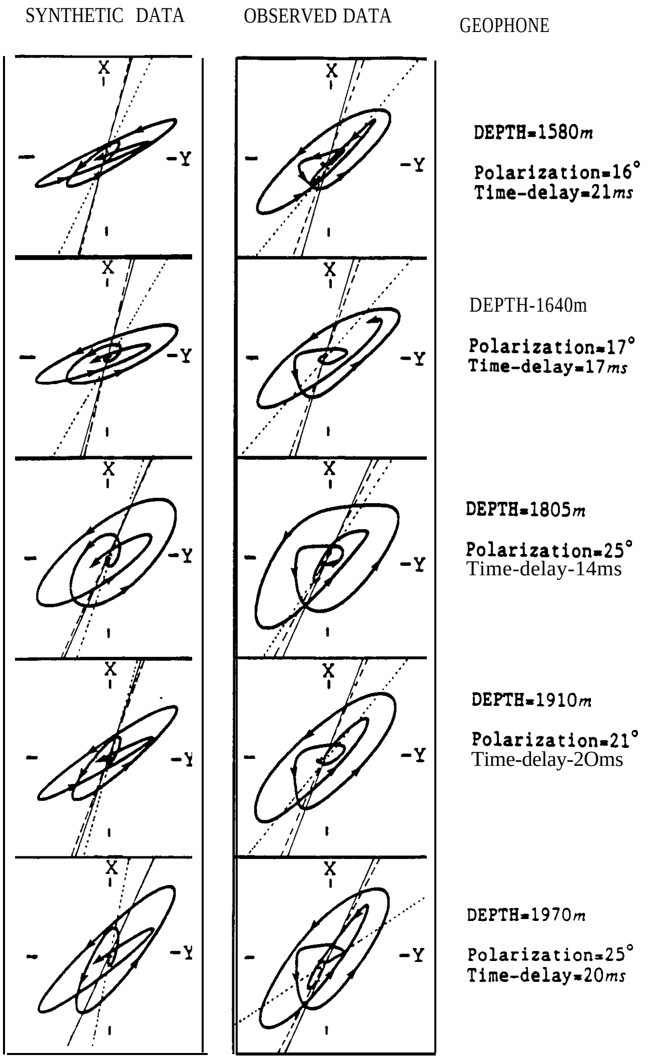

Fig. 7. Selection of five observed particle motions in the horizontal plane and corresponding synthetic particle motions chosen by the network as having the closest fit. Lines on each plot show the polarization azimuth chosen by the network (dashed line), polarization azimuth chosen by the single source technique (solid line on the observed data) and actual model polarization (solid line on the synthetic data), and polarization direction chosen by visual examination (dotted line).

used in this study. These maps show the magnitudes of each weight connecting the different layers in the network in Fig. 2. In each weight map, solid circles are shaded on a grey-scale corresponding to a magnitude range indicated by the key, with the far right row of circles in each map representing the nodal threshold. Although is it no possible to fully understand the logic underlying the network solution by visually inspecting these weight patterns, the weight map may still lend some perspective as to which aspects of the seismic waveform are more relevant to the solution than others, and thus be of benefit to further processing schemes or studies.

Before training, the weights and thresholds are initialized randomly between -0.5 and +0.5, after training these values range from -3.7 to +4.2. It appears that the weights connecting the input to first hidden layer concentrate on the portion of the signal between samples 14 to 40 (52ms to 156ms), corresponding to the main energy in the input signal, the *qS1–qS2* interference lobe. A curious feature revealed by Fig. 8 is that weights of large magnitude appear only for the radial component. This result suggests that the network is somehow sensing the departure from isotropic behaviour, as in an isotropic medium the radial component would be zero. Large negative and positive weights correspond to negative and positive signal values, yielding a large positive product, x. Due to the nature of the activation function in Fig. 3, which has in its denominator a negative exponential with the exponent being the summation of the product of weights and signal input, these large values produce in turn a large negative exponent, thus accentuating $O(x)$. The input weights separate into two distinct groups, both concentrated around the central lobe, which in turn connect separately (but not totally independently) with the weight distributions for the output polarization and time-delay.

To gain more understanding of the various weighted combinations of the input signal, which ultimately determine ϕ and $\Delta\tau$, we trained the network using split shear-waves with a source wavelet which is a single pulse s(t). In this case, each training example has four non-zero sample values, two of which correspond to delayed secondary arrivals. The training dataset is created for the same range of polarization and time-delay as in the previous example. Figure 9 gives the weight maps for this trained network and correctly shows that there are large values only for the range of signal amplitudes in the training dataset (nodes 20 to 30 and nodes 70 to 80 inclusive) corresponding to the maximum time-delay of 40ms. Note that in Fig. 9, the particular input signal example has a 20ms time-delay corresponding to input nodes 25 and 75. This does not reflect the range of time delay in the network which we initially set by presenting all training examples. It is interesting to note that both components are now strongly weighted. The weight combinations for this map are not easily traced as the parameter estimates appear to be more widely spread amongst the connections, especially for the second hidden layer. It must be concluded that the connections between the input layer, first and second hidden layers appear to be source-wavelet dependent and are probably being trained for the specific dataset. This agrees with our findings that if the source wavelets used in the synthetic training data are too dissimilar from the field data, the neural network fails to recognize the field data.

Several other indicators of these difficulties have also arisen during sensitivity tests of our results. For example, we find the output of the trained network is

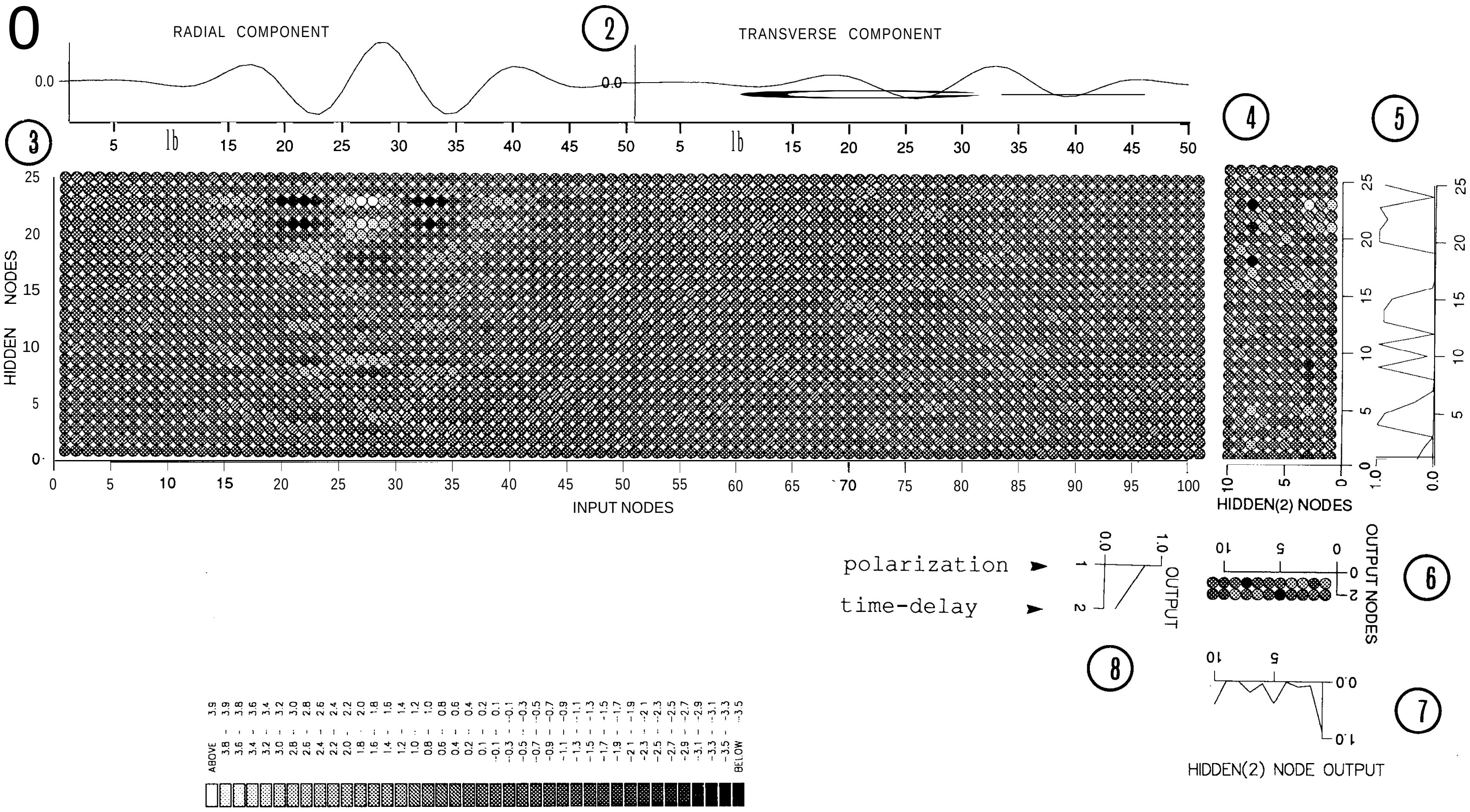

Fig. 8. Weight map for the trained neural network and a sample of synthetic training dataset. (1) and (2) correspond to the input traces ($r_1, r_2, \ldots, r_{50}$, and $t_1, t_2, \ldots, t_{50}$); (3) the mapping between the input layer and the first hidden layer (the weights having been reordered); (4) the mapping between the first hidden layer and the second hidden layer; (5) the output from the first hidden layer; (6) the mapping between the second hidden layer and the output layer; (7) the output of the second hidden layer; and (8) the final output. In each weight map, the right hand row of circles represent the node thresholds. The contrast in each circle indicates the value of a weight.

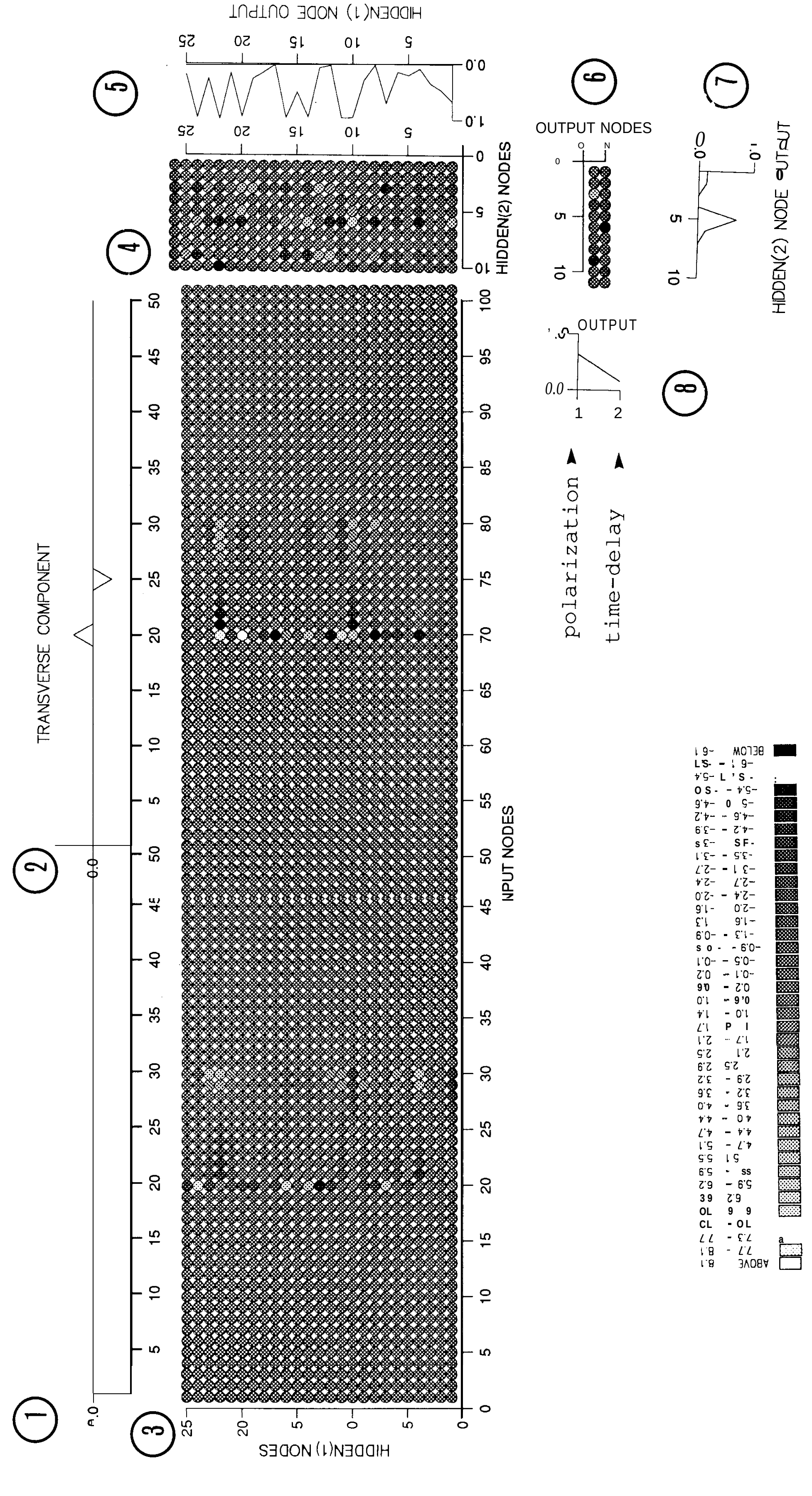

Fig. 9. As in Fig. 8, but for neural network trained by a single delta function δ(*t*) as an input source wavelet.

sensitive to the size and position of the waveforms in the time-window. Shifting the position of the window may occasionally cause large changes in the neural network output. The shear-wave must be carefully picked to ensure accurate processing, which necessitates a pre-analysis algorithm for picking (which has been attempted for both exploration and earthquake data). It appears that the ability of ANN also depends on the type of training data, and a critical factor is the choice of training datasets to match the field data for source and medium characteristics. Consequently, we must inspect the field data first, before selecting suitable parameters to create the training data. To make the neural network more robust, we need to add more training data, exploring different wavelets, coherent noise, source direction and offsets. The major drawbacks with this approach are the excessive increase in training time (which increases exponentially!) and a large and unmanageable network structure.

Conclusions

It is widely known that neural networks are ideally suited to situations where standard algorithms cannot be used as the mathematical relationship is uncertain, provided a supply of many examples of what is required can be generated. Once trained using the example patterns, it can recognize new patterns using its memory. In this respect, the approach is simple and adaptive to any problem. However, there are some limitations in this philosophy, which we have encountered in our application of our chosen type of network. The performance depends on the training data and its ability to generate solutions cannot lie too far outside its experience-in our case the solution is a highly non-linear function of wavelet and its shape cannot deviate too far from the actual source wavelet. Although, in general, extra training sets should help with this problem, for this application it is too time-consuming to train the network to recognize more complicated wavelets, and pre-processing for the source function appears necessary. Alternatively, we might choose a different network architecture from Fig. 1. On a more positive note, this ANN application has been of value in exploring the emphasis placed on each component part of the seismogram for this particular problem; ANNs have the ability to discover a relationship, no matter how complex, without *prior* statistical concepts. Consequently, application of a network can often be used as a tool to highlight relationships which exist between certain quantities, and help gain a greater insight into the physical process.

Acknowledgements

This research was sponsored by Global. Seismology Research Group (GSRG), British Geological Survey (BGS) of the Natural Environmental Research Council (NERC), and is published with the approval of the Director of the BGS (NERC). We thank the staff and students of GSRG for their help and thank the French VSP Consortium for the VSP data.

References

Cliet, C. and **Lefeuvre,** F. 1989. Characterization of the micro-structure of the sedimentary rocks in the Paris Basin from shear-wave splitting analysis. *Presented at the 51st Annual Meeting, EAEG.*

Dowla, F.U., **Taylor,** S.R., **Anderson,** R.W. 1990. Seismic discrimination with artificial neural networks: preliminary results with regional spectral data. *Bulletin of the Seismological Society of America, 80,* 13461373.

Kusuma, T., and **Brown,** M. 1992. Cascade-correlation architecture for first-break picking and automated trace editing. *Proceedings of SEG 62nd Annual International Meeting, Society of Exploration Geophysics,* 10–13.

Leggett, M., **Sandham**, W.A., and **Durrani,** T.S. 1993. Seismic event classification using a self organising Kohonen network. *Proc. IEE/IEEE int Workshop on Neural Algorithms in Signal Processing,* 14–16 November 1993, Chelmsford, UK.

McCormack, M.D. 1991. Neural Computing in Geophysics. *Geophysics: The Leading Edge of Exploration,* January 1991, 1 l-15.

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

Murat, M. and **Rudman,** A. 1992. Automated first arrival picking: A neural network approach, *Geophysical Prospecting 40, 587-604.*

Palaz, I., and **Weger,** R.C. 1990. Waveform recognition using neural network. *Geophysics: The Leading Edge of exploration,* March 1990, 28-3 1.

Pao, Y.H. 1988. *Adaptive Pattern Recognition and Neural Networks.* Addison-Wesley Publishing Company Inc., New York.

Poulton, M.M., **Sternberg,** B.K and **Glass,** C.E. 1992. Location of subsurface target in geophysical data using neural network. *Geophysics 57,* 1534-l 544.

Rumelhart, D.E., **McCleland**, J.L. and the PDP Research Group, 1988. *Parallel Distributed Processing: Exploration in the microstructure of Processing, Volume I: Foundations.* MIT Press, Cambridge, MA.

Taylor, D.B. 1992. *ANISEIS v4.6 Manual.* Applied Geophysics Inc., Houston, TX.

Zeng, X. and **Macbeth,** C. 1993. Algebraic processing techniques for estimating shear-wave splitting in near-offset VSP data: theory. *Geophysical Prospecting* 41, 1033-1066.