Field Experiment of Refracted Waves and Its Travel Time Interpretation Using Neural Networks

Sadeghi, Hossein
Faculty of Science, Kyushu University

Suzuki, Sadaomi
Faculty of Science, Kyushu University

Field Experiment of Refracted Waves and Its Travel Time Interpretation Using Neural Networks
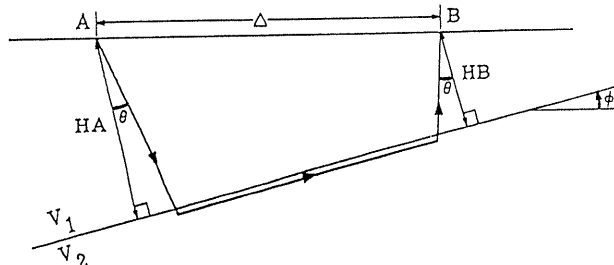
Hossein SADEGHI and Sadaomi SUZUKI

Abstract

We study neural networks for interpretation of travel time data of refracted waves in a field experiment. The designed multilayer network consists of an input layer, two hidden layers and an output layer. Fifty examples of simple velocity structure consisting of two layers seperated by a plane interface dipping less than 10 degrees are used to train the neural network. Surface layer velocity, travel times and distances between two sources and five receivers are used as the network inputs. Target output is the expected velocity structure model. The trained neural network produces results that are close to the actual output of the new examples not included in the training set. The trained neural network also gives acceptable results for travel time data of field experiment. We demonstrate that neural networks are able to perform transformation giving the correct connections between data and model parameters in seismic refraction surveys.

Introduction

The expression for travel time of seismic refracted wave through a two layered velocity structure, whose interface consists of a plane dipping surface, as shown in Fig. 1, can be represented in several ways. One formulation is

$$t = \frac{HA \cos \theta}{V_1} + \frac{HB \cos \theta}{V_1} + \frac{\Delta \cos \phi}{V_2} \qquad (1)$$

where HA and HB are the prependicular distances from the shot A and the receiver B to the interface. V_1 and V_2 are the seismic wave velocities in the upper and lower layers, respectively. θ is the critical angle or $\sin \theta = \frac{V_1}{V_2}$ and ϕ is the dip angle of the interface.

The neural network applications is an important technique that solve

Fig. 1. Ray path along a dipping refractor interface from shot A to receiver B (after SCHEIDEGGER and WILLMORE, 1957).

problems without specifying an algorithm. In order to solve a problem the neural network goes through a training process with some examples or patterns. Then the neural network with patterns supervision approximates the functional relationship between the input and output spaces.

Neural networks have been used for inversion of seismic data by ROTH and TARANTOLA (1994) and seismic travel time tomography by LU and BERRYMAN (1990). Neural networks have also been used in other seismological studies, *e.g.*, for seismic data trace editing (McCORMACK *et al.*, 1993), first arrival picking by MUART and RUDMAN (1992) and also by McCORMACK *et al.* (1993), in seismic event classification by PULLI and DYSART (1990).

There are some traditional methods for using equation (1) in refraction travel time analysis. For example SCHEIDEGGER and WILLMORE (1957) have introduced the time-term method.

In this paper, we will describe the application of artificial neural networks to analyze refracted wave travel time data for determining the depth of interface and refractor velocity. The designed network has been trained with a set of numerical models, based on equation (1). Finally, the accuracy of trained network will introduce by comparison of its results with numerical models, not present in the trainig set, and field experiment and also with the Mereu's iterative method for solving the time-term equations (1966).

Field experiment

On 27 November 1994 a simple seismic experiment was arranged in the Hakozaki campus of Kyushu University (Fig. 2). Seismic refraction profile was 40 meters with 1 meter recording space.

In this seismic refraction measurment, hammer was the signal source and two geophones, one on the source point and another on the receiver point, were finding out seismic energy arriving. Difference between signal receiving times at source point geophone and receiver point geophone is travel time.

Digital recorder (TEAC DR-F1) was recoding the output of geophones after amplification, in floppy disk. After completion data of forward and reverse direction, these data were analyzed by PC-9801 in order to find each arrival time
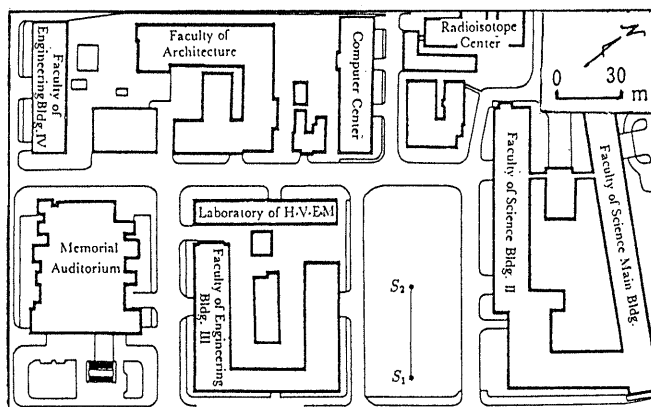

Fig. 2. Map showing location of field experiment in Hakozaki campus of Kyushu University.

and make record sections or the display of seismic traces in a time-distance graph, by using shot time as zero point for each trace. Fig. 3 shows record sections of forward and reverse directions.

We have analyzed the first refracted arrival times by an iterative method for solving time-term equations that MEREU (1966) has presented. In time-term analysis for the above-mentioned field experiment, because record sections of stations on 35 and 38 meters of reverse direction have not a clear arrival time,
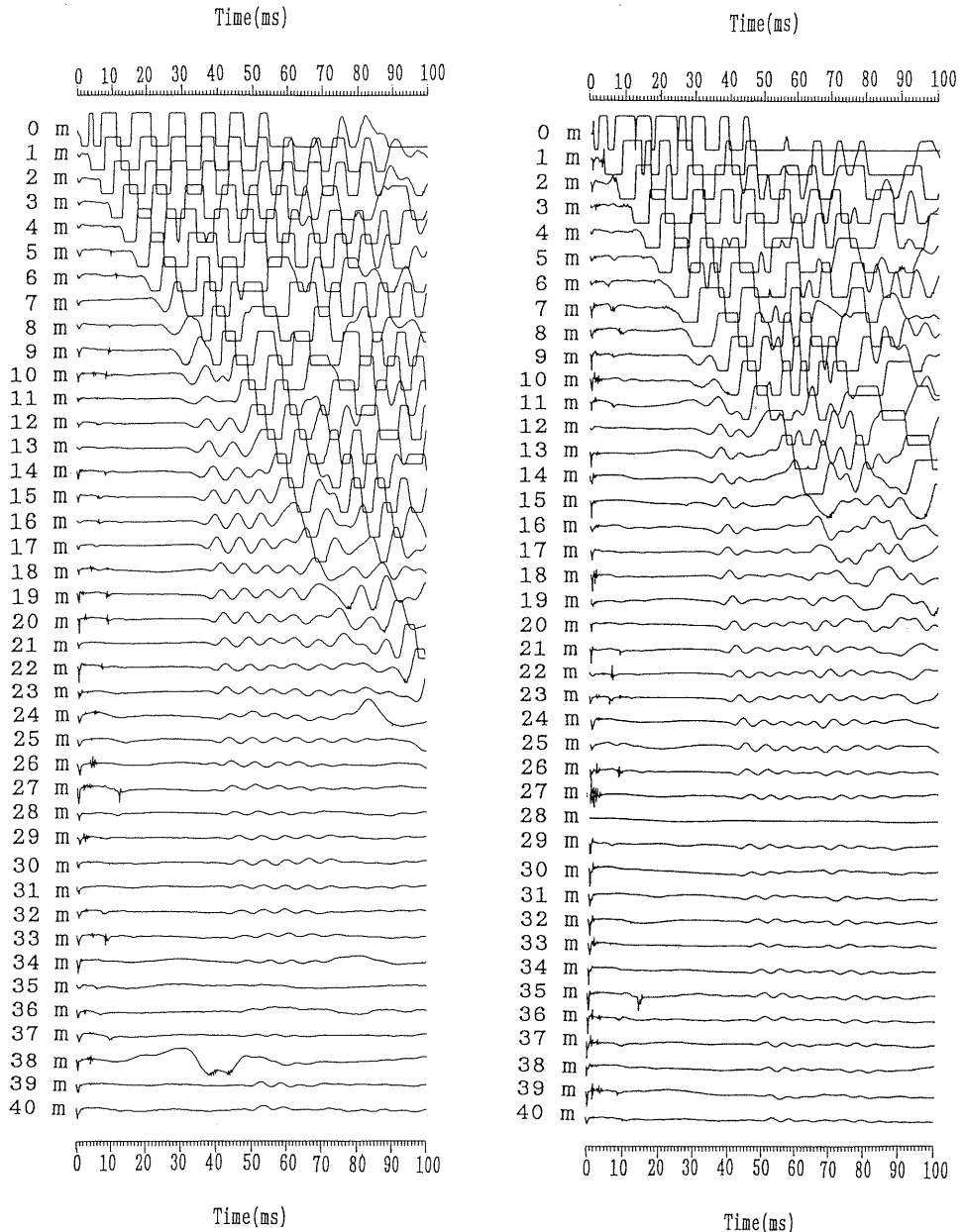

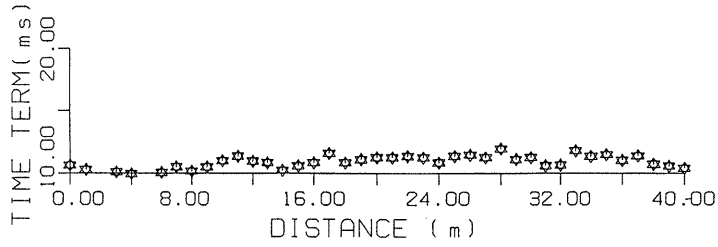Fig. 3. Record sections of forward (right) and reverse (left) directions.
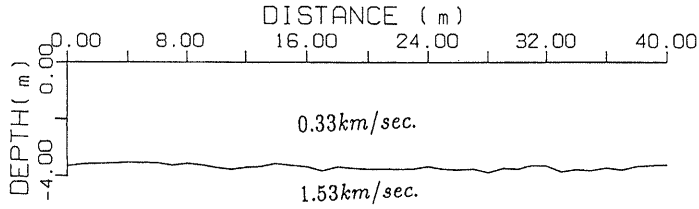
Fig. 4. Time-terms of stations.

Fig. 5. Model of seismic P wave velocity structure.

these stations have been omitted. The values of time-terms are shown in Fig. 4 and refractor velocity is; $V_2=1.53km/sec$. Fig. 5 shows the seismic P wave velocity structure model. The perpendicular distance from each station to the interface is calculated by the time-term of station and first layer velocity, $V_1=0.33km/sec$.

Principles of multilayered neural networks

Artificial neural networks, which we simply call neural networks, can be characterized most adequately as 'computational models' with particular properties such as the ability to adapt or learn, to generalize, or to cluster or organize data (KROSE and SMAGT, 1993). In general a neural network can be thought of black box and consists of a pool of simple processing elements which communicate to each other over a large number of weighted connections. Fig. 6 illustrates a typical neural network. Each processing element in a neural network collects the values from all of its input connections and produces a single output value:

$$\text{NET}_j^h = \sum_{i=1}^{n} W_{ji}^h \ I_i + \theta_j^h \qquad (2)$$

where n is the number of neurons in input layer. I_i, W_{ji}^h and θ_j^h are input neuron, connection weight and an external input or bias in hidden layer, respectively. Upper suffix h means the hidden layer.

Using continuously differentiable functions, such as Sigmoid function, the network can learn practically any nonlinear mapping to any desired degree of accuracy,

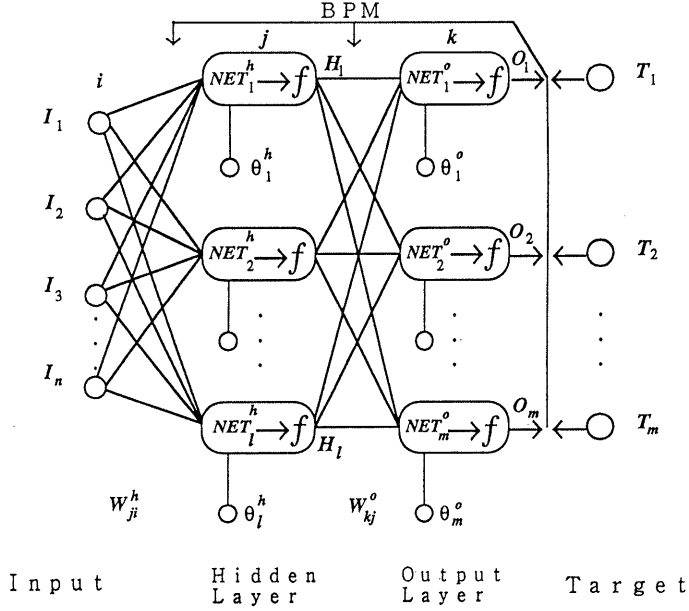$$H_j = f\left(\text{NET}_j^h \right) \qquad (3)$$

Fig. 6. Explanation of a neural network (see text in detail).

where H_j is output of the neuron j in the hidden layer and f is the activation function.

The output of hidden layer neurons are then transmitted to the input of the output layer neurons through another network weights denoted by W^o_{kj}. Thus the output of the output layer, O_k, is expressed as follows:

$$\text{NET}^o_k = \sum_{j=1}^{l} W^o_{kj} H_j + \theta^o_k \tag{4}$$

$$O_k = f\,(\,NET^o_k\,) \tag{5}$$

where l is the number of neurons in hidden layer, upper suffix o means the output layer and θ^o_k is a bias in output layer.

The error, E^p, is the summed square error for pattern p and is defined to be

$$E^p = \frac{1}{2} \sum_{k=1}^{m} (\,T_k - O_k\,)^2 \tag{6}$$

where the variable T_k is the desired or target output and m is the number of neurons in output layer.

Learning is used to determine the values of connection weights which minimize the error between the computed and desired output. In one common technique for learning in multilayer networks, back propagation method (BPM), neural network goes through an iterative period of training for adjustment of weights of a neural network by a method called *gradient descent*. The idea is to make a change in the weight proportional to the negative of the derivative of

error as measured with respect to each weight:

$$\Delta W_{kj}^o = -\eta \frac{\partial E^p}{\partial W_{kj}^o} \tag{7}$$

where η is learning rate coefficient. The derivative is

$$\frac{\partial E^p}{\partial W_{kj}^o} = \frac{\partial E^p}{\partial NET_k^o} \frac{\partial NET_k^o}{\partial W_{kj}^o} \quad ; \quad \frac{\partial NET_k^o}{\partial W_{kj}^o} = H_j \tag{8}$$

When we define

$$\delta_k = -\frac{\partial E^p}{\partial NET_k^o} = -\frac{\partial E^p}{\partial O_k} \frac{\partial O_k}{\partial NET_k^o} = (T_k - O_k) f'(NET_k^o) \tag{9}$$

from equations (7), (8) and (9) we get an update rule resulting in a gradient descent on the error:

$$\Delta W_{kj}^o = \eta \delta_k H_j \tag{10}$$

Therefore the weights W_{kj}^o and W_{ji}^h can be adjusted to minimize error for the set of training patterns:

$$\Delta W_{kj}^o = \Delta W_{kj}^o + \eta \delta_k H_j \tag{11}$$

$$\Delta W_{ji}^h = \Delta W_{ji}^h + \eta \sigma_j I_i \tag{12}$$

where

$$\sigma_j = (\sum_{k=1}^{m} \delta_k W_{kj}^o) H_j', \quad H_j' = \frac{\partial H_j}{\partial NET_j^h} \tag{13}$$

Neural networks experiments

Neural networks can find a set of weights that can successfully associate or map input vectors with output values. The process of finding the weights is called *training*. In order to train, some external teachers (also called patterns) are required to find the weights which minimize the difference between the computed and desired output pattern, for each of the input patterns.

We trained the multilayer network by 50 patterns of numerical models. These models consist of two layers seperated by a plane interface (see Fig. 7). V_1 ranges from $0.15km/sec$ to $7.5km/sec$ and V_2 ranges from $0.4km/sec$ to $8.4km/sec$. The dip of the interface is less than 10 degrees and measured from the horizontal in up or down direction. Travel times have been computed from two shots on the two ends of models. The number of receivers between two shots has been chosen to be five. Of course if we select more receivers the solution will be better. But we found from our experiment that five receivers are enough. The other condition that we impose on the models is; the maximum depth of interface should be equal to half of the surface distance between shots. As two shot points and all receiver stations corresponding to each model have a fixed distance, Fig. 7, each model introduces 10 travel times of refracted waves, based on equation (1), with respect
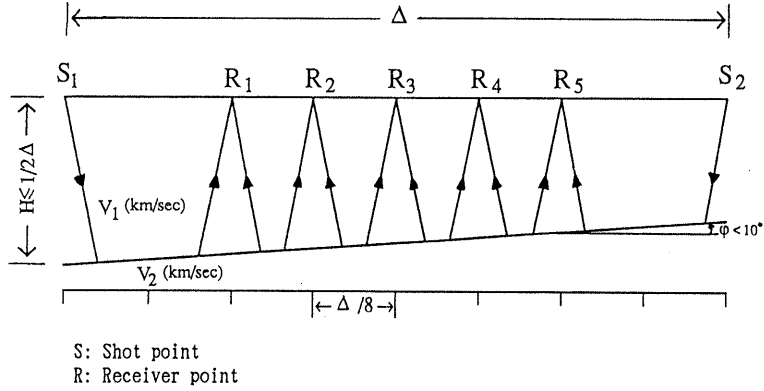
S: Shot point
R: Receiver point

Fig. 7. Conditions for pattern models.

to 10 distances between shots and receivers. These 20 data and first layer velocity are the input vector. The results consist of 7 depths of the interface from five stations and two shots, and one velocity of the refractor.

The input neurons correspond to elements of the input vector. The output layer produces the results. Therefore there are 21 neurons in input layer and 8 neurons in output layer. Choosing the number of hidden layers and the number of neurons in each hidden layer is usually done through experimentation and there are no strict rules. Too large a number of units in the network will give slow convergence, and may not generalize well; too few units will not allow a solution to the problem (ANTHONY et al., 1992).

At first one hidden layer was employed but was not sufficient and the neural network could not find the solution. Two hidden layers with 18 neurons in first and 14 neurons in second layer have been found to be suitable for our problem. We also used three hidden layers but did not increase the benefit. Use of more than three hidden layers gives no benefit (WILLIAMS and GUCUNSKI, 1995 : WARD SYSTEMS GROUP, 1993).

The neural network was trained with the normalized inputs and targets. The velocity was normalized with respect to a maximal value of $10km/sec$. The maximum value for normalizing depths of interface and distances between shots and receivers was 2.5Δ. Travel times were also normalized with maximum value of 2.5Δ, whose dimension is *msec or sec* where the dimension of Δ is *meter or km*, respectively.

The activation functions, $f(NET)$, of three layers are chosen to be the sigmoidal function, which takes the form (WANG and TENG, 1995)

$$f(NET) = \frac{1}{1 + exp(-\mu NET)} \tag{14}$$

where μ controls the slope of the semi-linear region (see Fig. 8). We chose μ =0.2 because of a minimum *TSSE* as shown in Figure 9. The weights are adjusted based on the error function E^p with the learning rate η =0.8.

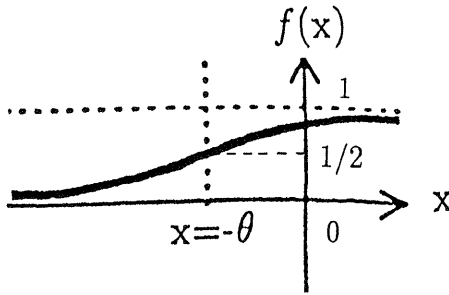The ability of the network to converge is shown by total sum square error, *TSSE*, which is defined as

Fig. 8. Sigmoid function, $\dfrac{1}{1+exp-\mu(x+\theta)}$, θ and μ are parameters which relate to the location and slope of the sigmoid, respectively.
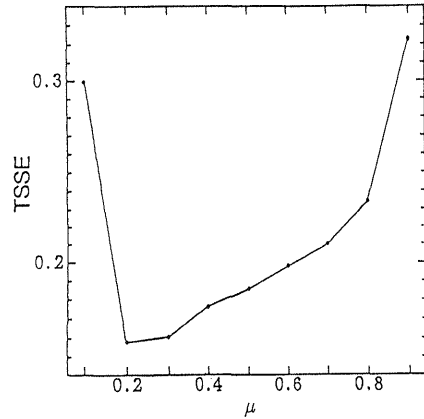

Fig. 9. Total sum square error, *TSSE*, change for different values of μ after 1000 iterations.

$$TSSE = \sum_{p=1}^{N^p} E^p \tag{15}$$

where N^p is the number of patterns. Fig. 10 shows *TSSE* against number of iterations. The network has kept iterating untill the *TSSE* has been reduced to 0.0045, with 50000 iterations.

After the training process, for testing the generalization capacity of the network, 10 new examples under before mentioned restrictions for numerical models, set to the trained networks. Fig. 11 gives the comparison of the computed and desired new velocity models, not used for training, by sum square error for each one. This figure introduces the high accuracy and capability of the proposed system for interpretation of a simple structure as the presented models.

Testing the accuracy of the trained network with the field experiment we have used the travel time data of five receivers on 10, 15, 20, 25 and 30 meters from shot 1. Table 1 compares the model results computed by the presented
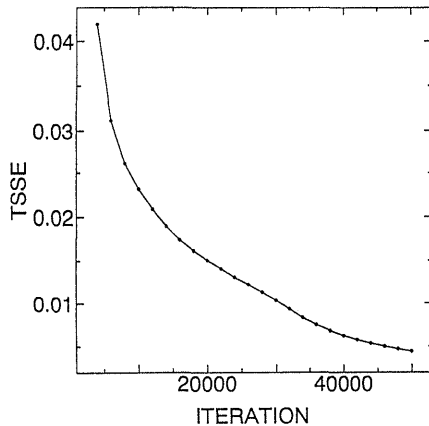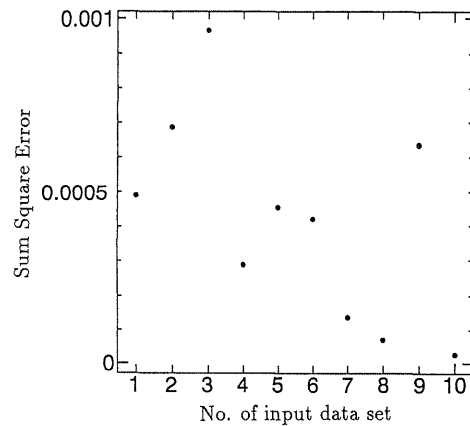

Fig. 10. *TSSE* against number of iterations.

Fig. 11. Sum square error, E^p, for 10 examples of input data sets.

Table 1. Comparison of results by time-term method and neural network.

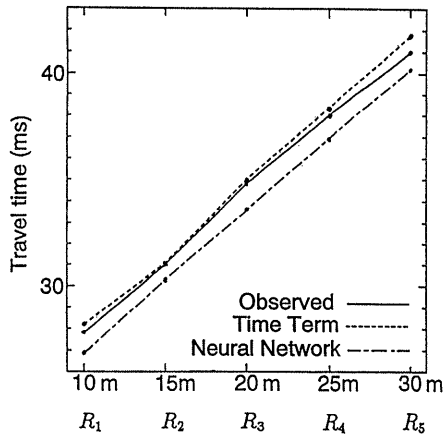| Station	Interface Depth (m)		Distance
	Time Term Method	Neural Network	from S_1(m)
S_1	3.60	3.38	
R_1	3.72	3.50	10
R_2	3.59	3.56	15
R_3	3.81	3.58	20
R_4	3.84	3.61	25
R_5	3.89	3.62	30
S_2	3.52	3.28	40
Velocity (V_2)	1.53 (km/sec)	1.54 (km/sec)	

Fig. 12. Comparison of travel times calculated by time-term method and the neural network with observed ones.

trained network and those of calculated by the time-term method. Fig. 12 shows the plots of calculated travel time from structure results of time-term method and the neural network and also observed travel times. Note that the values derived from the time-term method are not only based on the travel times of five mentioned receivers but also the rest of travel time data from other receivers in field experiment. Thus these values are expected to be more accurate. In addition, we trained the neural network only by plane interface examples. In other words, the accuracy of the neural network when the interface is not plane will not be the same as plane interface models. Therefore, although Fig. 12 demonstrates a little lower fitting of the neural network results to observed travel times, neural networks expected can interprete more satisfactory field data if networks are trained by more input travel time data and also by uneven interface models.

Conclusions

We have trained a multilayer neural network to construct a simple velocity structure from travel time data. The results show the high accuracy of the neural network for interpretation of travel times calculated for a simplistic structure. It seems that neural networks can also provide a useful technique for the analysis of field travel time data. This result is preliminary. More detailed investigation

may be able to construct a more complicated structure and using neural network techniques may provide alternative methods for interpretation of seismic refraction data.

Acknowledgments

References

ANTHONY, D. M., HINES, E. L., HUTCHINS, D. A. and MOTTRAM, J. T. (1992): Ultrasound tomograph imaging of defects using neural networks. *Neural Computation,* **4**, 758-771.

KROSE, B. J. A. and PATRICK VAN DER SMAGT, P. (1993): *An introduction to neural networks,* University of Amesterdam, 13-34.

LU, S. and BERRYMAN, J. G. (1990): Inverse scattering, seismic traveltime tomography, and neural networks. *Int. Jour. Imaging Sys. Techno.,* **2**, 112-118.

McCORMACK, M., ZAUCHA, D. and DUSHEK, D. (1993): First-break refraction event picking and seismic data trace editing using neural networks. *Geophysics,* **58**, 67-78.

MEREU, R. F.(1966): An iterative method for solving the time-term equation. In *The Earth beneath the Continent* , *AGU Monograph No. 10,* 495-497.

MURAT, M. E. and RUDMAN, J. R. (1992): Automated first arrival picking: A neural network approach. *Geophysi. Prospect.,* **40**, 587-604.

WARD SYSTEMS GROUP (1993): *Neuroshell 2 manual.,* Frederick, Inc., Mass.

PULLI, J. J. and DYSART, P. S. (1990): An experiment in the use of trained neural networks for regional seismic event classification. *Geophys. Res. Let.,* **17**, 977-980.

ROTH, G. and TARANTOLA, A. (1994): Neural network and inversion of seismic data. *Jour. Geophys. Res.,* **99**, [B], 6753-6768.

SCHEIDEGGER, A. E. and WILLMORE, P.L. (1957): The use of a least squares method for the interpretation of data from seismic surveys. *Geophysics,* **22**, 9-22.

WANG, J. and TENG, T. (1995): Artificial neural network-based seismic detector. *Bull. Seism. Soc. Am.,* **85**, 308-319.

WILLIAMS, T. P. and GUCUNSKI, N. (1995): Neural networks for backcalculation of moduli from SASW test. *J. Computing Civ. Engrg.,* **9**, 1-8.