Velocity inversion in cross-hole seismic tomography by counter-propagation neural network, genetic algorithm and evolutionary programming techniques

Sankar Kumar Nath, Subrata Chakraborty, Sanjiv Kumar Singh and Nilanjan Ganguly*

Department of Geology and Geophysics, Indian Institute of Technology, Kharagpur 721302, India. E-mail: nath@gg.iitkgp.ernet.in

Accepted 1999 January 25. Received 1999 January 18; in original form 1998 October 23

SUMMARY

The disadvantages of conventional seismic tomographic ray tracing and inversion by calculus-based techniques include the assumption of a single ray path for each source-receiver pair, the non-inclusion of head waves, long computation times, and the difficulty in finding ray paths in a complicated velocity distribution. A ray-tracing algorithm is therefore developed using the reciprocity principle and dynamic programming approach. This robust forward calculation routine is subsequently used for the cross-hole seismic velocity inversion.

Seismic transmission tomography can be considered to be a function approximation problem; that is, of mapping the traveltime vector to the velocity vector. This falls under the purview of pattern classification problems, so we propose a forward-only counter-propagation neural network (CPNN) technique for the tomographic imaging of the subsurface. The limitation of neural networks, however, lies in the requirement of exhaustive training for its use in routine interpretation.

Since finding the optimal solution, sometimes from poor initial models, is the ultimate goal, global optimization and search techniques such as simulated evolution are also implemented in the cross-well traveltime tomography. Genetic algorithms (GA), evolution strategies and evolutionary programming (EP) are the main avenues of research in simulated evolution. Part of this investigation therefore deals with GA and EP schemes for tomographic applications. In the present work on simulated evolution, a new genetic operator called 'region-growing mutation' is introduced to speed up the search process.

The potential of the forward-only CPNN, GA and EP methods is demonstrated in three synthetic examples. Velocity tomograms of the first model present plausible images of a diagonally orientated velocity contrast bounding two constant-velocity areas by both the CPNN and GA schemes, but the EP scheme could not image the model completely. In the second case, while GA and EP schemes generated an accurate velocity distribution of a faulted layer in a homogeneous background, the CPNN scheme overestimated the vertical displacement of the fault. One can easily identify five voids in a coal seam from the GA-constructed tomogram of the third synthetic model, but the CPNN and EP schemes could not replicate the model. The performances of these methods are subsequently tested in a real field setting at Dhandadih Colliery, Raniganj Coalfields, West Bengal, India. First arrival traveltime inversion by these algorithms from 225 seismic traces revealed a $P$-wave velocity distribution from 1.0 to 2.5 km s$^{-1}$. A low-velocity zone (1.0 km s$^{-1}$), the position of a suspected gallery in the Jambad Top coal seam, could be successfully delineated by CPNN, GA and EP schemes.

Key words: evolutionary programming, genetic algorithm, neural network, seismic tomography, simulated evolution.

* Now at: School of Earth and Ocean Sciences, University of Victoria, Canada.
INTRODUCTION

Seismic tomography, which has existed as an inverse technique for decades and has been used to generate 1-D and occasionally sparse 2-D models, has emerged during the past 20 years as a spectacular new frontier in seismology. The extension of well-known concepts of seismic inversion to three dimensions, the adaptation of imaging techniques borrowed from medicine, and the use of modern, fast computers with graphics facilities have jointly produced a powerful tool that has pervaded every branch of seismology, from global-scale Earth processes to metre-scale resource exploration.

A large number of tomographic methodologies and applications in global seismology and exploration geophysics have been reported by Kak & Slaney (1985), Nolet (1987), Natterer (1986), Iyer & Hirahara (1993) and Hardage (1992) in their books.

Computerized tomographic imaging, initially developed in the medical field (Kak & Slaney 1985), has received substantial attention in geophysics as a means of estimating subsurface velocity distributions. Of all the seismic geometries used in data acquisition, cross-well is the most like the geometry used in medical imaging.

In cross-hole tomography, one uses the first arrival travel-times of the signals transmitted between two boreholes to determine the seismic velocities in the plane between these holes. Standard cross-hole tomographic techniques, such as the simultaneous iterative reconstruction technique (SIRT) (Gilbert 1972; Dines & Lytle 1979; Ivansson 1983), algebraic reconstruction technique (ART) (Herman 1980; Natterer 1986) and conjugate gradient technique (CGT) (Paige & Saunders 1982; Bregman et al. 1989), are based on the decomposition of the subsurface area into a number of small constant-velocity cells and inversion of the time derivative matrix. It has been found that the analysis of seismic and electromagnetic tomographic data does not always give accurate information about the anomalous region. The resolution, accuracy and the degree of distortion in the estimated images vary with the type of inversion algorithm used, the location of the anomalous zones between the boreholes, and the initial model parameters assigned to the medium when using the iterative reconstruction technique (Ivansson 1986; Dyer & Worthington 1988). Ivansson (1986) showed that, by adding more diverse data (from a variety of viewing angles) and imposing constraints based on a priori information (Dines & Lytle 1979; Bregman et al. 1989), non-uniqueness can be reduced to a great extent. Furthermore, as each ray travels through only a small fraction of the total number of cells, the time derivative matrix is generally sparse in nature, causing difficulty in convergence when SIRT schemes are used.

Since the use of model-free estimators, such as pattern recognition using artificial neural networks, can enhance the speed and accuracy of cross-hole seismic tomographic imaging by approximating a function, that is by mapping the traveltime vector to the velocity/slowness vector, we moved to the domain of pattern classification. By presenting an input pattern to the trained network, the desired pattern evolves with the use of a learned reference vector from a pattern look-up system. Owing to the effectiveness of forward-only counter-propagation neural networks (CPNN) in such problems, an attempt was made to implement it in the solution of the tomographic imaging problem under consideration. The limitation of neural networks, however, lies in the exhaustive and time-consuming training required for the interpretation of cross-hole seismic data.

While optimal solutions are always aimed at, sometimes from poor initial models, global optimization and search techniques such as simulated evolution can be applied in the design of practical algorithms for solving the imaging problem. Initially, the genetic algorithm (GA) scheme is implemented in the form of a structured computer program to suit our applications. The common genetic operators, crossover and mutation, are used. A new genetic operator, 'region-growing mutation', is introduced to speed up the search process. Subsequently, we designed an algorithm based on evolutionary programming (EP) to tackle certain complex geological settings. While the GA scheme works at the chromosomal level, EP works at the level of the species. The region-growing mutation is also used in the EP scheme.

The paper is organized as follows. In the first section, we discuss the forward modelling scheme which is at the heart of the inversion algorithm. Next, we describe the implementation of the forward-only counter-propagation neural network technique for cross-hole seismic imaging. The simulated evolution techniques using GA and EP are subsequently outlined. The performances of these techniques are tested using a variety of synthetic and real field examples, with some of the results being presented in the last section. The numerical examples are chosen to simulate geological conditions for detecting a dipping interface, faulted blocks in the subsurface and multiple voids in coal seams. The first arrival times picked from a P-wave seismogram acquired at Dhandadih Colliery (23°38’N, 87°5’E), Raniganj Coalfields, West Bengal, India are interpreted for the detection of galleries in the coal seam by CPNN, GA and EP schemes. Good resolution of the raster images of the subsurface is achieved by these techniques for both the synthetic and actual field examples.

THEORY

The forward problem

Every tomographic inversion method needs a forward modelling scheme that enables the accurate calculation of the misfit function/ error-norm and its derivative with respect to model parameters. Since minimization of the misfit function is an iterative process, the forward modelling must be performed a number of times and therefore must be robust and fast. As the forward modelling algorithm based on the reciprocity principle and Fermat’s Principle uses the concepts of the dynamic programming approach (Matsuoka & Ezaka 1992; Schneider et al. 1992) to take care of the curved ray geometry of the ray paths, this is the one which is most suitable for our analysis.

The following steps are used in this algorithm.

1. The minimum traveltimes from a source point to all grid points in the cell model are computed.
2. The source and the receiver positions are interchanged and the minimum traveltimes at all the grid points are calculated in the reverse direction.
3. The traveltime data obtained from the above two steps are added to generate a two-way traveltime map over the entire mesh.
4. Fermat’s Principle is then invoked to estimate the ray path from the global minimum traveltimes.
The procedure used for the calculation of traveltimes makes use of a dynamic programming approach, whereby the traveltimes are mapped using a Markovian process. Given the geometry of the mesh, in order to calculate the first arrival time at any point \( R \), only the arrival times at the three neighbouring points are required. Two of these points \( (H, E) \) are located in the previous column (left cell boundary) and one \( (G) \) is located in the same column (right cell boundary). The situation is illustrated in Fig. 1(a) for a single cell. Calculation of the traveltime is performed using a non-linear interpolation scheme for curved ray paths (Singh et al. 1996).

In this method, the traveltime map is established over the entire grid space by considering four time values, two of which are from the neighbouring grid points, while the other two are calculated by considering the intersection of the ray path with the nearest row and column. The points of intersection are determined by the root bisection method outlined below.

With reference to the geometry of a cell in Fig. 1(a), \( d_2, d_0, d_3 \) are the distances from the source \( S (x_s, z_s) \) to the corner points \( E (x_1, z_1) \), \( F_{row} (x_0, z_1) \) and \( G (x_2, z_1) \), \( F_{row} \) is the point of intersection of the ray with the row.

If \( \Delta \) is the average slowness between the source \( S \) and the cell under consideration, then

\[
t_0^2 = \Delta^2 d_2^2, \quad t_2^2 = 3 \Delta^2 d_3^2.
\]

Subtracting \( t_2^2 \) from \( t_0^2 \) and rearranging the terms we obtain

\[
\Delta^2 = \frac{(t_0^2 - t_2^2)}{(x_1 - x_2)^2 - (x_0 - x_1)^2} = W_s \text{(say)},
\]

where \( t_2 \) and \( t_3 \) are the times along ray paths \( SE \) and \( SG \), respectively.

Again,

\[
\Delta^2 = \frac{t_0^2 + W_s [(x_1 - x_2)^2 - (x_0 - x_1)^2]}{2}
\]

Let the traveltime at \( R \) along \( SF_{row} \) be \( t_{F_{row}} \), and the time taken to reach \( F_{row} \) from the source \( S \) be \( t_0 \), then

\[
t_0^2 = \Delta^2 + W_s [(x_0 - x_1)^2 - (x_0 - x_2)^2],
\]

and

\[
t_{F_{row}} = t_0 + s \sqrt{(\Delta x)^2 + (x_2 - x_0)^2},
\]

where \( s \) is the slowness in the cell under consideration.

Differentiating eq. (2) with respect to \( x_0 \) and rearranging the terms, we have

\[
dt_0 = \frac{W_s}{t_0}(x_0 - x_1).
\]

Differentiating eq. (3) with respect to \( x_0 \), we have

\[
dt_{F_{row}} = \frac{W_s}{t_0}(x_0 - x_1) - \frac{s}{\sqrt{(\Delta x)^2 + (x_2 - x_0)^2}}(x_2 - x_0).
\]

On substitution of the value of \( dt_0/dx_0 \), eq. (5) becomes

\[
dt_{F_{row}} = \frac{W_s}{t_0}(x_0 - x_1) - \frac{s}{\Delta x}(x_2 - x_0).
\]

For \( t_{F_{row}} \) to be minimum, \( dt_{F_{row}}/dx_0 \) should be equal to zero. Therefore,

\[
s^2 (x_2 - x_0)^2 = \frac{W_s^2 (x_0 - x_1)^2}{t_0^2}.
\]

Thus we can now define a function \( \text{FUNC}_{row} \) as

\[
\text{FUNC}_{row} = \frac{s^2 (x_2 - x_0)^2}{(\Delta x)^2 + (x_2 - x_0)^2} - \frac{W_s^2 (x_0 - x_1)^2}{t_0^2}.
\]

Considering the intersection of the ray with the column at \( F_{col}(x_1, z_0) \), and proceeding in a similar manner, we can define a function \( \text{FUNC}_{col} \) as

\[
\text{FUNC}_{col} = \frac{s^2 (z_2 - z_0)^2}{(\Delta z)^2 + (z_2 - z_0)^2} - \frac{W_s^2 (z_0 - z_1)^2}{t_0^2}.
\]

such that

\[
W_s = \frac{(t_1^2 - t_0^2)}{(x_1 - x_2)^2 - (x_0 - x_1)^2} - \frac{(z_1 - z_0)^2 - (z_0 - z_1)^2}{(\Delta z)^2 + (z_2 - z_0)^2}.
\]

and \( t_0, t_1 \) are the traveltimes along \( SF_{col} \) and \( SH \), respectively.

By equating \( \text{FUNC}_{row} \) and \( \text{FUNC}_{col} \) to zero, we obtain the coordinates of the intersection points \( F_{row} \) and \( F_{col} \), and then compute the values of \( t_{F_{row}} \) and \( t_{F_{col}} \). The first arrival time at \( R \) is the minimum of \( t_1 + t_{\Delta x}, t_0 + t_{\Delta z}, t_{F_{row}} \) and \( t_{F_{col}} \).

The values obtained at all the points in a column are then stored in a floating array, which is used for calculating the values in the next column. The floating array is subsequently updated by substituting the old values with the new ones. The mapping thus shifts by one column, and the computation over the next column starts. The process continues until the right...
edge of the grid is reached. At this stage we have the first
arrival times at all the grid points starting from the source.
The process is then continued in the reverse direction by
interchanging the source and the receiver positions. The float-
ing array is updated by adding the new traveltime data while
calculating in the reverse direction. Thus a two-way traveltime
map is generated. The minimum of two-way traveltimes in the
cell matrix is the ray path for a source–receiver configuration,
and half of this time yields the time of arrival at a particular
node. An example of the performance of this forward algorithm
is depicted in the raster image of Fig. 1(b), which shows ray	race tracing through an intrusive high-velocity square model.

FORWARD-ONLY COUNTER-
PROPAGATION NEURAL NETWORK

The key factors in designing and implementing a neural net-
work application are the choice of network, data representation,
selection of training and test examples, the number of hidden
layers, and the number of processing elements. The choice of
network is problem-dependent. There are several types of net-
worls that are best suited for classification-type problems, such
as the probabilistic neural net learning vector quantization,
counter propagation (Hecht-Nielsen 1990), adaptive resonance
and self-organizing map (Kohonen 1988).

Tomographic imaging of the subsurface velocity structure
from first arrival traveltime data can be considered to be a
problem of approximating a function, that is, of mapping a
traveltime vector to the velocity matrix. This also falls under
the purview of pattern classification problems, in which, by
presenting an input pattern to the trained network, the desired
pattern is evolved using a learned reference vector from a
pattern look-up system. Since counter propagation, and in
particular forward-only counter propagation, has been devised
for pattern matching and pattern completion applications, taking
advantage of the parallel architecture of neural networks, we
have implemented the forward-only counter-propagation
neural network approach in designing and coding an algorithm
for cross-hole seismic tomography. The basic theory and
algorithm for the counter-propagation neural network are
discussed below.

Basic theory

Counter-propagation neural networks (Hecht-Nielsen 1987,
1990) are multilayer networks based on a combination of
input, clustering and output layers. A CPNN approximates its
training input vector pairs by adaptively constructing a look-
up table. In this manner, a large number of training data
points can be compressed to a more manageable number of
look-up table entries. If the training data represent function
values, the net will approximate a function. A hetero-associative
net is simply one interpretation of the function from a set of
vectors (patterns) X to a set of vectors Y. The accuracy of the
approximation is determined by the number of entries in the
look-up table, which equals the number of units in the cluster
layer of the net.

Two types of layers are generally used in CPNNs: the hidden
layer is the Kohonen layer, with competitive units which do
unsupervised learning; the output layer is the Grossberg layer,
which is fully connected to the hidden layer and is not
competitive.

Counter-propagation nets are trained in two stages. During
the first stage, the input vectors are clustered. The resulting
clusters may be based on either the dot product metric or the
Euclidean norm metric. During the second stage of training,
the weights from the cluster units to the output units are
adapted to produce the desired response.

Forward-only counter-propagation nets are a simplified
version of the full counter-propagation nets. Forward-only
nets are intended to approximate a function \( y = f(x) \) if the
mapping from \( x \) to \( y \) is well defined, but the mapping from \( y \)
to \( x \) is not.

Forward-only counter propagation differs from full counter
propagation in using only the X vectors to form the clusters
on the Kohonen units during the first stage of training. The
original presentation by (Hecht-Nielsen 1990) of forward-only
counter propagation used the Euclidean distance between
the input vector and the weight vector for the Kohonen unit.

Architecture

The architecture of a forward-only counter-propagation net is
illustrated in Fig. 2. The counter-propagation net has inter-
connections among the units in the cluster layer, which are
not shown. In general, in forward-only counter propagation,
after competition only one unit in that layer will be active and
send a signal to the output layer.

Algorithm

The training procedure for the forward-only counter-
propagation net consists of several steps. First, an input vector
is presented to the input units. The units in the cluster layer
compete for the right to learn the input vector. After the entire
set of training vectors has been presented, the learning rate is
reduced and the vectors are presented again. This continues
through several iterations.

After the weights from the input layer to the cluster layer
have been trained (the learning rate has been reduced to a
small value), the weights from the cluster layer to the output

Figure 2. Topology for a forward-only counter-propagation neural

network.

© 1999 RAS, GJI 138, 108–124
Implementation of CPNNs in cross-hole seismic tomography

Using the concept of forward-only CPNNs, a network is designed and implemented to suit the applications in cross-hole seismic tomography. The software is coded considering a network of three layers, namely the input layer, the competitive layer and the output layer.

The competitive layer has been taken as a rectangular grid. During training, the traveltime vector $T$ is presented to the input layer, and the known slowness/velocity structure is presented to the output layer. When a pattern is fed to the network, the winning unit is found in the competitive layer. The weights between the input layer and the competitive layer are adjusted, not only for the winning unit but also for other units in the neighbourhood. This is a deviation from the standard counter-propagation algorithm. The neighbourhood is taken initially large enough to cover the entire grid at the competitive layer. As the iterations progress, the neighbourhood size is decreased linearly. The amount of weight adjustment depends on the learning rates. The learning rate used for adjusting weights between the input and competitive layers ($\alpha$) also decreases as the iterations progress. The learning rate mechanism directed towards decreasing cost or increasing pay-off and have a high probability of locating the global solution optimally in a multimodal search landscape.

Genetic algorithms proceed from a loose analogy between function optimization and a biological system composed of organisms that interact in a relatively complex way and are comparatively few in number. Since a genetic algorithm tries to find an optimal answer by evolving a population of trial answers in a way that mimics biological evolution, the use of the principles of genetic optimization will be a natural choice in cross-hole seismic tomographic applications. Currently there are three main avenues of research in simulated evolution: genetic algorithms, evolution strategies and evolutionary programming. Each method emphasizes a different facet of natural evolution. The fundamental difficulty with genetic algorithms lies in the fact that the crossover operation may at times cause premature convergence. Since evolutionary programming does not use the crossover operation, it may result in a more diverse population than a single genetic algorithm run would, which may be very useful for geophysical inversion problems. With the advent of high-speed desktop computers/workstations with enormous computational capabilities and parallel processing environments, the use of genetic algorithms and evolutionary programming to search for a global solution has become much more common.

The art and science of basic genetic algorithms in the context of general optimization problems has been given in several books (e.g. Holland 1975; Goldberg 1989; Sen & Stoffa 1995) and papers (e.g. Boschetti et al. 1996). Detailed accounts of GAs for seismic waveform inversion are given by Stoffa & Sen (1991) and Sen & Stoffa (1995).

The algorithm presented here is based on the work by Holland (1975), who discussed the theory of genetic algorithms in natural and artificial adaptations. Holland's algorithm is modified for application in cross-hole seismic velocity inversion. A completely new genetic operator, called region-growing mutation, is used to speed up the search process. Subsequently, we implemented evolutionary programming using the guidelines given by Fogel (1991).

**Mathematical model and genetic algorithm scheme**

The objective of the GA here is to evolve the subsurface 2-D velocity distribution model between two boreholes that would give rise to the traveltime data observed in the field. The forward modelling routine based on the reciprocity principle is invoked by the GA program to calculate the travel times for each individual velocity model in a population for the subsequent calculation of the fitness of that individual. A brief summary of our GA scheme is given below.

**Generation of population and representation structure of an individual**

The model parameter values are coded in binary form. An individual in a population is represented as a structure consisting of three members: the slowness matrix consisting of slowness indices; the fitness of the individual; and a flag which states whether or not the individual has undergone a particular operation. The slowness matrix constitutes the elements of the structure between the boreholes. The value in each element
Figure 3. Computational steps for the forward-only CPNN scheme implemented for cross-hole seismic tomography.
points to a floating-point array containing the possible slowness values. These slowness values are read from an input file. A population consists of many such individuals, an assembly of possible solutions.

The simulated evolution in cross-hole seismic tomography includes multiple slowness/velocity distributions in the subsurface to be imaged. The slowness matrix contains the codes in the usual 8-bit binary representation. The individuals in the population are randomly generated using this binary representation of the slowness codes. Once the coding scheme is selected, all the coded model slowness values are joined together into a long bit string, analogous to a chromosome, which represents the genetic information of each individual in the population.

**Fitness function**

Without loss of generality, the fitness function of an individual is taken as the reciprocal of the RMS error between the traveltimes calculated for the individual and those observed in the field (or calculated for the original structure). The RMS error is calculated for each and every individual velocity model by calling the forward modelling routine to compute the traveltimes. If \( t_{ij}(j = 1 \text{ to } n) \) is the traveltime for the \( j \)th source-receiver pair in the \( i \)th individual velocity model, the RMS error can be expressed as

\[
e = \sqrt{\frac{1}{n} \sum_{j=1}^{n} (t_{ij} - t_{ij})^2},
\]  

(10)

where \( t_{ij} \) is the traveltime for the \( j \)th ray in the original structure. The population of velocity models is then sorted in ascending order of the RMS error to yield a population with individuals sorted in descending order of fitness value.

**Genetic operators**

In the present implementation of the genetic algorithm, three genetic operators are used, namely crossover, mutation and a filter mutation which we prefer to call the region-growing mutation.

**Crossover:** The crossover operation in our algorithm is modified slightly from that used in a simple genetic algorithm (SGA). As the population is sorted according to the fitness at every stage, the individuals at the top of the ranked list are good individuals, and those at the bottom are poor. In the present scheme, the individuals are taken one by one from the best 25 per cent of the population to be paired with an individual chosen randomly from the remaining population. The crossover point is also chosen randomly. The offspring generated by crossover replace the individuals having the lowest fitness. The entire population is re-sorted and the individuals are ranked based on their fitness values. The process is illustrated in Fig. 4(a). A crossover rate of 2 per cent is used for all operations. Since one individual is always selected from the best 25 per cent of the population for the crossover operation, traits of the good individuals always exist in the tomographic formulation of the simulated evolution algorithm, a new genetic operator is introduced to speed up the search process. The building block hypothesis (Goldberg 1989) states that ‘Genetic Algorithm combines high fitness above average schemata termed as building blocks to form better individuals in the population’. As the tomographic problem may have one system stability as the solution is approached. A mutation rate of 1 per cent is found to yield the fastest convergence and is therefore used in all the synthetic and real field examples presented here.

**Region-growing mutation:** In the present implementation of the genetic operators are used, namely crossover, mutation and a filter mutation which we prefer to call the region-growing mutation.

**Mutation:** The mutation operation is implemented in the same way as in the SGA. Depending on the mutation rate, an individual is chosen randomly from the population by leaving the best 10 per cent of the individuals intact. This ensures
or more anomalous regions of velocity/slowness, the search is speeded up by allowing the small regions that the GA tries to combine to grow faster. This is done by carrying out a ‘3 × 3’ low-pass filter operation. The process is demonstrated in Fig. 4(b). A ‘3 × 3’ window is first placed on the slowness matrix with the first element at the centre of the window. The slowness in this element is replaced by the slowness that occurs most frequently among all the elements falling within the window. As shown in the figure, the total number of elements in the window is either four (when the window is placed on a corner element), six (when it is placed at the edge) or nine (when placed over any other element). In case of a tie, that is when more than one slowness occurs most frequently, any one of them is chosen randomly. The window then moves over to the next element in the row and the operation is repeated. This operation is performed sequentially row-wise over the entire slowness matrix.

With this kind of operation, the GA starts gathering the 0s and 1s, or any other slowness indices if there are more than two slowness values in the desired portions of the individuals. The regions that the GA has started forming ‘grow’ in this operation, and hence we choose to call it region-growing mutation. Geological formations having a speckled distribution of velocities may be considered homogeneous, as the velocity of interest in that case becomes the composite velocity of the structure. As we are interested in detecting distinct regions of different velocities, the region-growing mutation becomes an indispensable tool.

For a faster convergence, only those individuals ranked between the top 10 and 20 per cent are mutated, keeping the best 10 per cent of the population intact. The mutated individuals replace the worst individuals in a sequential manner starting from the bottom-most individual in the ranked list. Once the operation is complete, the entire population is again sorted in ascending order of RMS error of the individuals.

The major computational steps are given in the flow chart of Fig. 5.

**Evolutionary Programming**

Fogel *et al.* (1966) used evolutionary programming (EP) in system identification in which finite-state machines were considered to be organisms. The parents of the organisms which gave the best solutions for a target function were allowed to reproduce, and the parents were mutated to create offspring. Holland (1975) proposed a GA scheme that used many new concepts, such as coding, crossover, etc. and Fogel (1988) applied it to the travelling salesman problem. Fogel & Fogel (1989) re-examined the EP algorithm, and Fogel (1991) outlined an EP technique appropriate for function optimization in geophysical inversion. Fogel points out that the crossover operation, which is unique to GA schemes, may lead to premature convergence. This is because, after successive generations, the entire population converges to a set of coding such that the crossover no longer generates any new chromosomes. This may happen even before an optimal solution has been found. Although mutation allows for diversity, the mutation rate is usually low, so that practically no improvement can be attained in the final generations of the algorithm.

Like a GA, an EP scheme starts with a population of models (let $n$ be the number of models in the population), and the fitness is evaluated for each of the models in the population. Next, an equal number of offspring are generated by perturbing each member of the population by a Gaussian distribution whose variance is obtained by properly scaling the fitness function of the model. Thus a poor model is allowed to move far from its current position, while improvements are sought in the close neighbourhood of a good model. Next, tournament selection is applied and a score is assigned to each of the $2n$ models. Each model is allowed to compete with a subset (e.g. $k$) of the $2n$ models, drawn at random based on their fitness. If the current model wins, it scores 1; otherwise, it is assigned a score 0. The competition is probabilistic, and thus the worse models have a finite probability of winning. Note, however, that no temperature-type control parameter such as that used in simulated annealing is used in the algorithm. The maximum score that any model can obtain is $k$. Next, the $2n$ models are ranked in decreasing order of their fitness values. The first $n$ of the $2n$ models are now considered parents, Gaussian mutations are allowed to create $n$ new offspring, and the process is repeated until the population converges to a high fitness value (Sen & Stoffa 1995).

Fogel (1991) gave several examples of function optimization with EP and showed that in many cases EP performed better than GAs. Again it is very difficult to say whether such performance can be expected for every optimization problem. Since this method does not use the crossover operation, it may result in a more diverse population than a GA, which may be quite useful for the purpose of cross-hole seismic velocity inversion.

**Mathematical model and algorithm of evolutionary programming**

The implementation of evolutionary programming in cross-hole seismic tomography (Singh *et al.* 1997) broadly followed the guidelines set by Fogel (1991). The crossover operation is entirely omitted.

Mutation being the main genetic operator in EP schemes, the operation is performed using Gray coding. The advantage of Gray coding is that the binary representation of two consecutive Gray-coded numbers differs by only one bit, thereby allowing a smooth variation of the model slowness. In the simple binary coding, however, we may need to change several bits to increase a model slowness value by only one unit. Gray codes are generated by forming bitwise exclusive or of an integer $i$ with the integer part $i/2$.

In the EP scheme, the mutation is performed in the following manner.

(1) The top 10 per cent of the ranked individuals are not considered for mutation as these are the fittest of the population.

The individuals to be mutated are selected randomly depending upon the user-specified mute rate. The number of cells to be flipped in the slowness matrix of any such individual is calculated using a Gaussian distribution function given as

$$
p = \frac{1}{\sigma \sqrt{2\pi}} e^{-1/2} \left( \frac{\varepsilon - \mu}{\sigma} \right)^2, \quad (11)
$$

where $\varepsilon$ is the RMS error of the selected individual and $p$ is the probability of mutation. The limits of the distribution are
Figure 5. Computational flowchart for the genetic algorithm scheme.

taken as $10^{-1}$ and $10^{-8}$, where the values of $p$ are 1.0 and 0.01, respectively. These limits define the values of $\mu$ and $\sigma$ for the distribution curve.

(2) At each mutation operation, a cell of the slowness matrix of the selected individual is chosen randomly.

(3) The value of the variable slowness code of the cell, say ‘$slo$’, is then converted into the Gray code.

In this representation of the variable, a position is randomly selected at which the bit has to be flipped. A mask is generated by placing 1 in the rightmost position and left-shifting this 1 to the randomly selected position for bit flipping. The bitwise exclusive or is then performed on the variable with the generated mask. Thus the desired mutated result is achieved and the generated ‘$slo$’ variable is converted back to binary form. This is the new slowness value that replaces the slowness of the cell selected for mutation. When block-wise mutation is to be performed, the new slowness value replaces the slowness of all the cells of the selected block.

Finally, the ranked individuals are sorted for inplace region-growing mutation to speed up the search process.

The major computational steps are given in the flow chart of Fig. 6.

RESULTS AND DISCUSSION

In this section, we present three synthetic data examples and one real field case study in which the performances of CPNN, GA and EP cross-hole traveltime tomographic algorithms are tested and compared.

The initial learning rates for the CPNN are chosen as 0.6 for ‘$a$’ and 0.1 for ‘$a$’, as suggested by (Hecht-Nielsen 1990). At the start of the first run in the CPNN, the weight file is...
created randomly from the range $-0.3$ to $+0.3$, obtained by trial and error. The CPNN is trained by a number of patterns, namely homogeneous velocity models with varying $P$-wave velocities between $0.33$ and $2$ km s$^{-1}$, vertical beds with variable $P$-wave velocities, horizontal beds, dipping interface between two layers with $P$-wave velocities of $1$ and $2$ km s$^{-1}$, and single and multiple voids in coal seams. Optimum population sizes of $50$ and $200$ are assigned to the GA and EP schemes, respectively. A threshold error level of $10^{-8}$ or an iteration limit of $10000$ is set as the termination criterion for CPNN, GA and EP schemes.

The tomograms presented here are obtained by cubic spline interpolation to $512 \times 512$ pixels from $9 \times 9$ and $17 \times 17$ grids for the synthetic case studies and $39 \times 39$ grids for the real field data example for better representation, continuous dithering of the image between $0$ and $255$ grey levels and visual clarity of the evolved subsurface models in continuous velocity domain.

**Synthetic data examples**

The testing of imaging methods on synthetic data is the most important component of every application. Hence we have chosen three synthetic models in which the cross-hole geometries comprise squares with sides of $8$ m in the first two examples and $16$ m in the third one. The left-hand side corresponds to the well in which the sources are placed at depth intervals of $2$ m—four positions for the first two models and eight positions for the third model. The right-hand side corresponds to the other well, where the receivers are placed at depth intervals of $2$ m—four locations for the first two models and eight locations for the third one. The source-receiver positions are indicated in the diagrams. Thus the numerical examples consist of $16$ seismic traces for the first two cases and $64$ traces for the third model. The synthetic seismograms are generated using finite element simulation (Nath et al. 1993), and the picked first arrival times are cross
Figure 7. (a) Raster image of a diagonally orientated velocity contrast bounding two constant-velocity areas. Tomogram of the dipping model estimated by (b) the CPNN after 77 iterations, (c) the GA after 56 iterations, and (d) EP after 225 iterations. (e) Plot of the normalized RMS error versus number of iterations during the reconstruction of the model by the CPNN, GA and EP schemes.
Figure 8. (a) Raster image of a faulted layer in a homogeneous background. Tomogram of the fault model estimated by (b) the CPNN after 95 iterations, (c) the GA after 272 iterations, and (d) EP after 121 iterations. (e) Normalized RMS error versus iteration plot for the CPNN, GA and EP tomographic reconstructions of the fault model.
checked using the forward modelling by reciprocity principle (Singh et al. 1996).

The first model consists of a diagonally orientated velocity contrast bounding two constant-velocity areas with $P$-wave velocities ($V_p$) of 1 and 2 km s$^{-1}$, as shown in the raster image of Fig. 7(a).

The tomograms of the model obtained by the CPNN after 77 iterations, the GA after 56 iterations and EP after 225 contrast bounding two constant-velocity areas with $P$-wave velocities ($V_p$) of 1 and 2 km s$^{-1}$, as shown in the raster image of Fig. 7(a).

The tomograms of the model obtained by the CPNN after 77 iterations, the GA after 56 iterations and EP after 225

---

**Figure 9.** (a) Raster image of five voids in a coal seam. Tomogram of the model generated by (b) the CPNN after 6075 iterations, (c) the GA after 2140 iterations, and (d) EP after 4450 iterations. (e) Error curves for the detection of five voids in a coal seam by the CPNN, GA and EP schemes.

iterations are presented in Figs 7(b), (c) and (d), respectively. A plot of normalized RMS error versus number of iterations for all three methods is given in Fig. 7(e). Velocity tomograms evolved by the CPNN and GA present plausible images of the dipping model, but EP failed to replicate the original. In the CPNN and GA, the error dropped down to zero, whereas in EP it levelled off at 0.000025 after 225 iterations. Of these three methods, the GA is found to yield faster convergence with less CPU elapsed time.

The second synthetic model represents a faulted layer with \( V_p = 5.0 \text{ km s}^{-1} \) in a homogeneous background with \( V_p = 2.5 \text{ km s}^{-1} \), as shown in the raster image of Fig. 8(a). The tomogram of the structure after 95 iterations by the CPNN overestimated the vertical displacement between the blocks, as evident from Fig. 8(b). The GA evolved a model after 272 iterations, as shown in Fig. 8(c), which is very close to the original one. Some spurious velocity anomalies can be observed in this tomogram, which may be due to premature convergence in the imaging process. EP replicated the model after 121 iterations, as shown in the tomogram of Fig. 8(d). The convergence of these three methods is represented in the form of error curves in Fig. 8(e).

The third synthetic model simulates a practical geological problem wherein we try to detect multiple voids in coal seams. The physical configuration of the synthetic model is presented as a raster image in Fig. 9(a). There are five voids in the coal seam. The tomograms of the model reconstructed by the CPNN after 6075 iterations (RMS error of 0.000002), by the GA after 2140 iterations (RMS error of 0.0) and EP after 4450 iterations (RMS error of 0.000008) are presented in Figs 9(b), (c) and (d), respectively. The GA replicated the velocity model. The CPNN could generate a model similar to the actual one, but EP estimated only four voids. The error curves are plotted in Fig. 9(e).

Owing to the limited number of training patterns, CPNN could not replicate the exact velocity model in the second and third examples. The diversity of population in EP helped in achieving an exact solution in the case of the faulted model but not in the other two cases. The GA, on the other hand, could detect the dipping interface and the multiple voids successfully and gave a very close solution for the fault model. However, as a whole, the results produced by each technique are noteworthy.

Now we apply these techniques to a real cross-hole data set acquired at Dhandadih Colliery, Raniganj Coalfields, West Bengal for delineating galleries in the coal seam.

Real data example

There is a history of surface cave-in, mine fires and inundation in the Raniganj Coalfields, West Bengal, India caused by a number of old unstable, abandoned and uncharted underground colliery workings under a shallow cover of about 50 m or less. In order to be able to control the consequent environmental, social and mine hazards, it is imperative that these underground workings are stabilized by techniques such as hydro-pneumatic sand-slurry filling of the underground voids.

However, in the absence of accurate mine plans, it is not possible to locate boreholes for packing and stabilizing such old workings. Dhandadih is one such colliery in the Raniganj Coalfields used in the present investigation as a test case.

In Dhandadih (23°38'N, 87°5'E) Colliery, the 6.9 m thick Jambad Top coal seam is being mined by the opencast method along its outcrop. The depth to the bottom of the coal seam is 23–24 m and its dip is shallow and rolling. One edge of the quarry has approached the outer limit of an underground board and pillar working in the same area which was abandoned and water logged over 50 years ago. The quarry has gradually progressed into the area worked earlier and has punctured galleries of the old workings. This site has been selected in order to enable the locations of underground pillars and galleries to be pinpointed before the progress of opencast mining.

The acquisition geometry for the cross-well survey and the major geological units are shown in Fig. 10. Two vertical wells BH-I and BH-II are separated by a horizontal distance of 37.6 m. A verification well (VW) is drilled 14.5 m from the tomographic borehole BH-I, as shown in the figure. The litholog of this verification well depicts the position of a gallery between 18.8 and 21.8 m depth in a coal seam about 7 m thick between 15.8 and 22.8 m depth. The objective of the cross-hole experiment was to locate this gallery in the coal seam. A cross-hole survey was conducted using a shear-wave-generating hammer system, a three-component geophone with a natural frequency of 40 Hz, and a 24-channel BISON seismograph. A sample interval of 0.1 ms and frequency band of 30–250 Hz were used for the data acquisition. Both the shots in BH-I and receivers in BH-II are located at depths ranging from 10 to 38 m at intervals of 2 m (15 shot points, each firing into 15 receiver points). The 225 seismic traces (Seguin et al. 1992–95)
are presented in Fig. 11(a). The first arrival time for each shot–receiver pair is hand picked to yield a database consisting of 225 $P$-wave arrival times. This database of arrival times is presented in Fig. 11(b) in the form of a traveltime surface (Pratt et al. 1993). In this raster display, each pixel corresponds to one source–receiver pair. The source position decreases in depth from left to right along the top of the plot; the receiver depths decrease from the top to the bottom of the plot. Each pixel has a grey level corresponding to the reduced traveltime minus the straight-ray arrival time in a homogeneous medium with a $P$-wave velocity of $2.48 \text{ km s}^{-1}$ calculated by the back projection method (Wong et al. 1983). The horizontal velocity model shown in Fig. 11(c), constructed by simply using the traveltimes corresponding to shot points and receivers located at the same depth, also depicts a higher average velocity of the order of $2.5 \text{ km s}^{-1}$ in the subsurface section. The traveltime surface of Fig. 11(b) gives an initial assessment of the data suggesting slower and faster rock masses in the subsurface, the slower being represented by the lighter grey levels and the faster by the darker ones.

The subsurface region $37.6 \times 38 \text{ m}$ in the horizontal and depth directions is divided into $39 \times 39$ grids. The $P$-wave velocity in this region is assumed to vary within the range $0.33–2.5 \text{ km s}^{-1}$. The lower limit is chosen with the assumption that the gallery may be air-filled. The upper limit is taken from the horizontal velocity model of Fig. 11(c). After 1306 iterations, the CPNN scheme had evolved a model in which the RMS error attained a steady value after a significant decrease from the initial one. The raster image of this model is shown in Fig. 12(a). We observe a variation in the $P$-wave velocity distribution from 1.0 to $2.5 \text{ km s}^{-1}$. A low-velocity region with $V_p = 1.0 \text{ km s}^{-1}$ can be identified in the tomogram between 17 and 20 m depth and between 12.5 and 17.5 m in the horizontal direction. The subsurface velocity models evolved by the GA after 1500 iterations and by EP after 1050 iterations are presented as raster images in Figs. 12(b) and (c), respectively. In both cases the velocity variation in the evolved models is the same as that obtained by the CPNN (i.e. $1.0–2.5 \text{ km s}^{-1}$). The low-velocity zones ($V_p = 1.0 \text{ km s}^{-1}$) in both these velocity tomograms could be identified between depths of 17.5 and 22.5 m for the horizontal spread of 13–17 m. As depicted in the verification well litholog of Fig. 10, these low-velocity zones indicate the position of the suspected gallery in the coal seam. Because the velocity contrasts between the coal seam and overlying sandstone and the underlying shale are very poor, the litho units are not all totally delineated in the tomograms of the subsurface region between the boreholes BH-I and BH-II. The gallery, on the other hand, shows a remarkable velocity contrast with respect to the surrounding

Figure 11. (a) Acquired cross-hole BH-I to BH-II seismogram comprising 225 seismic traces at Dhandadih. (b) Raster display of the residual traveltime surface generated from BH-I to BH-II $P$-wave first arrival data. (c) Horizontal velocity model of the subsurface region enclosed between the boreholes.

© 1999 RAS, GJI 138, 108–124
Figure 12. Raster image of the subsurface region (37.6 m × 38 m) between the boreholes BH-I and BH-II as estimated by (a) the CPNN after 1306 iterations, (b) the GA after 1500 iterations, and (c) EP after 1050 iterations. (d) Error curves for the detection of voids/galleries at Dhandadih by the CPNN, GA and EP schemes.
lithology and hence is distinctly identified in the raster images evolved by the CPNN, GA and EP schemes. The error curves for all three tomographic imaging methods are plotted in Fig. 12(d) for the sake of comparison. Note that between 0 and 10 m depth there were no source-receiver pairs and hence there is feeble illumination in this region.

This case study undoubtedly establishes the efficiency, efficacy and usefulness of the CPNN, GA and EP tomographic schemes presented in this paper.

CONCLUSIONS

Using traveltime tomography, seismic velocity functions have been analysed in a cross-swell geometry by a forward-only counter-propagation neural network, genetic algorithm and evolutionary programming. Both the synthetic examples and the real field case study demonstrated the accuracy of these tomographic tools.

From these analyses we can appreciate the inherent complexity of the problem. While medical tomography has a simple nature, geophysical tomography encounters a diverse set of problems, such as non-linearity, an ill-defined domain to be imaged, non-availability of full ray-path coverage of the area to be scanned due to limitations in the source-receiver geometry, the high cost involved in data acquisition and processing, etc. As tomographic imaging is a problem of approximating a function, CPNN, a proven tool for pattern mapping and pattern completion applications, could act as an efficient alternative approach for subsurface velocity/slowness estimation. While most of the existing tomographic tools are versatile enough to cover a range of earth models, the simulated evolution techniques are found to work over a large domain of problems. Their beauty lies in the fact that they start with a random assumption of velocity distribution without any a priori structural information and still guide the process to a very accurate solution.

This opens up a major avenue of research in the field of geophysical tomography. Our investigation is a step in that direction.

ACKNOWLEDGMENTS

The authors wish to thank the anonymous referees for their critical review, comments and helpful suggestions which greatly improved the quality of the manuscript leading to much clearer exposition. The permission given by IDRC, Canada and MGMI, Calcutta to use cross-hole data in the Raniganj Coalfields is gratefully acknowledged.

REFERENCES