An Improved Technique in Porosity Prediction: A Neural Network Approach

Patrick M. Wong, Tamás D. Gedeon, and Ian J. Taggart

Abstract— **Genetic reservoir characterization is important in developing, for a given petroleum reservoir, an improved understanding of the total amount and fluid flow properties of hydrocarbon reserves. Application of genetic concepts involves the classification of well log data into different lithofacies groups, followed by a facies-by-facies description of rock properties such as porosity and permeability. This work contrasts the genetic and nongenetic approaches in predicting porosity values of an oil well using backpropagation neural network methods. The performance of both methods are critically evaluated. A systematic technique to optimise the network configuration using weight visualization curves is proposed, thereby enabling the amount of training time to be significantly reduced. In the example problem, the genetic approach provides superior porosity estimates to that based on a nongenetic approach.**

I. Introduction

MANY forms of heterogeneity in rock properties, such as porosity and permeability, are present in clastic reservoirs. Understanding the form and spatial distribution of these heterogeneities is fundamental to the successful characterization of petroleum reservoirs [13]. Porosity and permeability are the two fundamental rock properties which relate to the amount of fluid contained in a reservoir and its ability to flow when subjected to applied pressure gradients. While fluid saturation is an equally important parameter, its estimation is beyond the scope of this paper.

A. Genetic Reservoir Characterization

There are two broad ways in which geologists, petrophysicists and engineers approach the problem of reservoir characterization. These two methods can be described as nongenetic and genetic approaches [17]. The nongenetic approach is the older, more established process of reservoir characterization. The genetic approach is a newer concept which seeks to identify and treat each dominant lithofacies group separately. Note that our use of "genetic" in this paper refers to the geological classification of rocks, and is in no way related to the different meaning of "genetic" algorithm used in the artificial intelligence literature.

The genetic approach to reservoir characterization emphasises the importance of lithofacies in hydraulic properties, such as facies-specific relationship between porosity and permeability [15], [25] and therefore each lithofacies must be treated individually. Identifying and mapping these lithohydraulic flow units is the first step in a successful reservoir characterization process. In the genetic approach, rock types are classified into the same lithohydraulic category (or more simply lithofacies) according to following criteria: 1) similar textural features (grain size and roundness), 2) similar diagenetic histories (mineralogy and pore space changes with time), and 3) similar petrophysical properties (such as porosity and wireline log response).

The above three criteria means that sedimentary features alone, should not be used in selecting lithofacies. Once selected, different lithofacies should readily be distinguished by possessing a differing pore throat geometry and connectivity. For example, in a given sandstone reservoir, the origin deposition may have caused localised sorting and orientation of grains into fine and coarse grain structures which may have similar porosity but widely different permeabilities. In addition, different amounts of calcite, a carbonate, precipitation may have occurred (cementing sand grains together). Thus four lithofacies may exist: fine and coarse sandstone, with and without, carbonate cementation. A nongenetic approach would treat all the sandstone as one unit thus ignoring for example, specific porosity-permeability relationships which may exist for each lithofacies.

The biggest difficulty faced with adopting a genetic approach to reservoir characterization is the reliable identification of lithofacies from available data in a reservoir under consideration. Generally, suitable outcrop and/or extensive core information is not available. Instead, the geologist usually has core data only in a few selected and isolated wells, and must instead rely on available wireline log data to distinguish different facies groups. From the selected core data, a set of lithofacies can be identified based on the criteria given above. These lithofacies must be linked to the larger set of wireline log signatures in such a way that the lithofacies group can be predicted on the basis of the log signature alone. Because of the inherent difficulty in this step, and the fact that it overlaps the separate disciplines of geology, pattern recognition, petrophysics and ultimately reservoir engineering, there are only a handful of documented examples of quantitative genetic approaches to reservoir characterization in the literature [17], [22], [26].

B. Previous Work

The concept of genetic reservoir characterization is relatively new. Its acceptance has perhaps been delayed by the

Manuscript received March 29, 1994; revised October 13, 1994.

P. M. Wong is with Petroconsultants Australasia Pty. Ltd., St. Leonards, NSW 2065, Australia.

T. D. Gedeon is with the School of Computer Science and Engineering, The University of New South Wales, Sydney, NSW 2052, Australia.

I. J. Taggart is with the West Australian Petroleum Pty. Ltd., Perth, WA 6001, Australia.

IEEE Log Number 9412807.

availability of suitable computational and statistical tools. LITHO [7] and FACIOLOG [28] represent examples, which employed clustering and principle component methods to classify wireline log data into different rock types. In these papers, the rock types were not necessarily lithofacies. The biggest drawback with these methods was the need to devise suitable cluster association rules (to reduce the number of clusters) and provision of logic to classify "between clusters" entries. Identification of outliers invariably occurred in transition regions between lithofacies and resulted in the concept of electrofacies [23]. A given lithofacies may have several associated electrofacies depending on the presence of nearby boundaries and transitions. While both of these methods were capable of good performance, the training process required to identify unique clusters is usually performed with considerable manual intervention and is considered cumbersome. Ideally the classification algorithm should be self training. The supervised classification method of discriminant analysis is one such method, and has been used in a genetic context with the introduction of electrofacies [17]. The disadvantage with discriminant analysis methods is however, the ease with which the results can be further processed into facies specific porosity and permeability data. It is the ability of neural nets to be readily configured to perform supervised classification, as well as performing nonlinear regression on inputs and/or intermediate results which make them attractive to genetic reservoir classification. Furthermore, neural network methods can form more complex decision boundaries which do not necessarily the redundancies introduced by electrofacies.

A supervised neural network technique has been widely used in lithofacies classification [1], [5], [8], [20]. Supervised learning requires training data which has been labelled with the desired outcome for each pattern of inputs. In this case, the well log data is labelled with the lithofacies group names. By selecting a suitable number of log responses to a given reservoir lithofacies, this technique can be trained to recognise, and predict, a given lithofacies from an input set of wireline log readings. Unlike traditional statistical classification techniques, such as discriminant analysis, neural networks can output both discrete (lithofacies) and continuous (porosity) data. Predictions of continuous values, such as total porosity and grain density [2] and permeability [19], were previously attempted, however, none of these utilised the genetic approach to reservoir characterization.

This work contrasts the application of neural networks in both genetic and nongenetic approaches for the purpose of porosity predictions. Whilst the application of neural network theory to wireline log analysis is not new, the particular contribution made by this work is in the use of neural network methods for both lithofacies and porosity predictions. This is an area which, to the best of our knowledge, has received little attention.

The following sections will first review the basic properties of neural networks. Then, a systematic approach in optimizing the network configuration using weight visualization curves will be introduced, followed by a brief discussion on developing the training data set. In the last section, we will compare the performance of the genetic and nongenetic methods to supervised porosity predictions from a suite of wireline logs. Data comes from a real reservoir where one well provides the core data used in the training phase of both schemes. Both methods are applied to a second well, where predictions of porosity values are compared to core data withheld from the training phase.

II. Neural Network

An artificial neural network, or simply a neural net, is a computer model which attempts to mimic some parts of the workings of the human brain [4]. It can learn from examples or experience, and is extremely useful in solving pattern classification and mapping problems.

The training, or learning, is an essential part in using neural nets. This process requires training patterns which consist of a number of input signals paired with target signals. The inputs are presented to the network and the corresponding outputs are calculated, and the network parameters are modified to reduce the error. The aim of training is to minimise the differences between the output and target values (i.e., errors) for all the training pairs. By training, a set of parameters are produced and can be used for classifying data or predicting property values in situations where the actual output is unknown. Note, however, that this set of parameters are problem-specific because each training data set results in a unique neural net.

A. Basic Architecture

A typical neural net is composed of three kinds of layers: input, middle (or hidden) and output layers. The input layer nodes are different to other nodes as they only receive input signals from the outside world and no mathematical operations are performed. These inputs layer nodes are connected, via weighted links, to every node in the middle layer. Unlike the input and output layers, the number of middle layers can be any positive number (as well as zero). Recent studies show that one middle layer is generally sufficient to solve complex problems if enough nodes are available [14], [16], and hence this study was only limited to one middle layer structure (see later sections). The decision of how many nodes should be present in the middle layer, however, is difficult to determine *a priori* and is usually determined by trial and error [3], [8]. The number of nodes present in input and output layers is usually more straightforward and is usually determined by the particular problem domain. Output layer nodes receive output signals from middle layer nodes and therefore provide responses to a given set of input signals. Fig. 1 shows a schematic diagram of a simple network architecture which consists of three layers. The input layer is composed of two nodes, and the middle and output layers contain three nodes. This can be described as the 2-3-3 configuration. Mathematically speaking, this neural computation involves a transformation of an input vector with two components (X1, X2) into an output vector with three components (Y1, Y2, and Y3). The magnitude of the output vector depends on the weights on all connections which are represented by the lines as shown in the figure. Bias nodes are usually included for faster convergence and better decision boundaries [6]. The

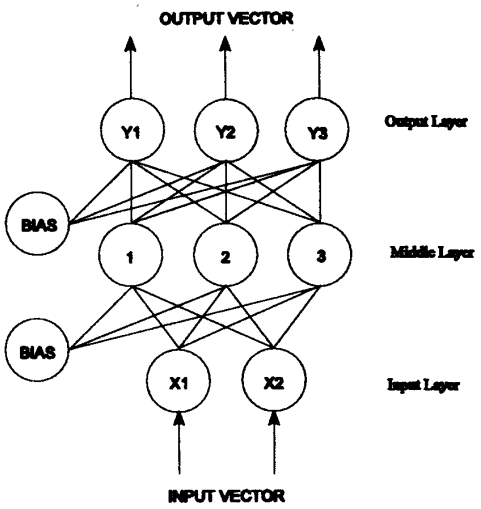


Fig. 1. Schematic diagram of a three-layer neural network.

weights on these nodes are treated the same as the others, and the input activation values of these connections are always equal to one.

B. Learning Algorithm

The backpropagation (BP) algorithm is the most widely used learning procedure for supervised neural nets. Before beginning training, some small random numbers are usually used to initialise each weight on each connection. BP requires preexisting training patterns, and involves a forward-propagation step followed by a backward-propagation step. The forward-propagation step begins by sending the input signals through the nodes of each layer. A nonlinear function, called the sigmoid function, is usually used at each node for the transformation of the incoming signals to an output signal. This process repeats until the signals reach the output layer and an output vector is calculated. The backward-propagation step calculates the error vector by comparing the calculated and target outputs. New sets of weights are iteratively calculated, by modifying the existing weights, based on these error values until a minimum overall error, or global error, is obtained. The root-mean-square error (RMSE) is usually used as a measure of the global error [6] which can be defined as

$$\text{RMSE} = \sqrt{\frac{\sum_i^{n_p} \sum_j^{n_o} (y_{ij} - x_{ij})^2}{n_p \cdot n_o}} \tag{1}$$

where n_p is the number of input/output pairs making up the training patterns, n_o is the number of nodes in the output layer, x and y are the output and target signals respectively. Details of the BP algorithm can be found in [4], [21].

In order to improve the generalization capabilities of the net, a test or validation data set (i.e., a set of known input-output pairings which were withheld from the training set) is usually used to stop training before generalization degrades. Performance of the trained network can be evaluated by some simple statistical functions such as recognition rate (i.e., percentage of the total number of correctly classified

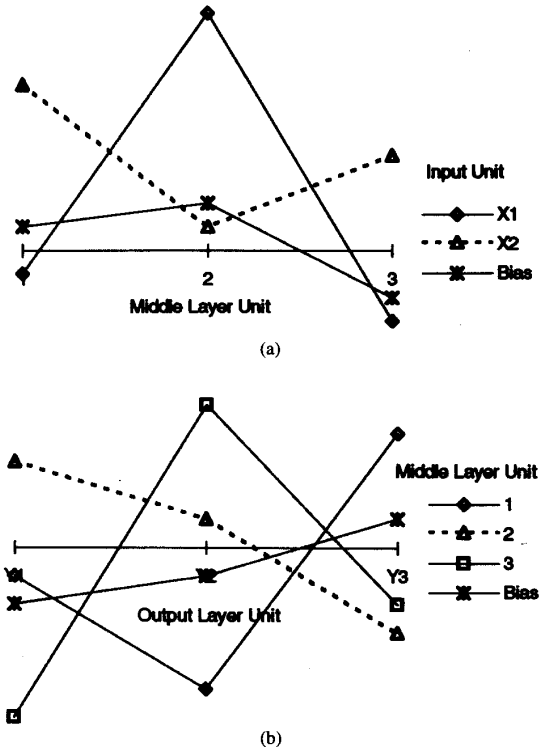


Fig. 2. WV-curves for 2–3–3 configuration in Fig. 1. Patterns of weights connecting different (a) input and middle layer units, (b) middle and output layer units.

outcomes over the number of sample points, or simply %Reco) and mean-square-error (MSE). If the error value on the test data set begins to increase, training is halted and the results are examined to determine whether they are acceptable. If the results are unacceptable, then it is possible to retrain the network, by either modifying some network parameters (e.g., the seed value for the random number generator, and the number of nodes in the middle layer), or increasing or decreasing the variations present in the training patterns. Once an acceptable error value is obtained during the test stage, the network is ready for solving real problems, such as classification of input signals (e.g., well log data) into discrete classes (e.g., lithofacies) and prediction of property values (e.g., porosity and permeability) using some input signals (e.g., well log data).

III. Training Set Development

A major drawback of the neural net approach is the problem of convergence. Convergence in the BP algorithm means that the global minimum (smallest error) of the error function is obtained in a reasonable amount of iterations, or epochs. The iterative process may require long training times of the order of several hundreds of thousands of iterations. Sometimes the network may get stuck in a local minimum during training which means that the network has failed to learn acceptably and gives large errors. Developing faster learning algorithms

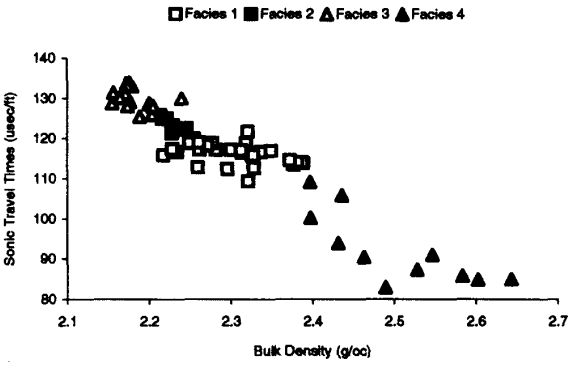


Fig. 3. Training patterns (65 data points) at Well A. Facies 1 is Mudstone, Facies 2 is Sandy Mudstone, Facies 3 is Sandstone, and Facies 4 is Carbonate Cemented Bed.

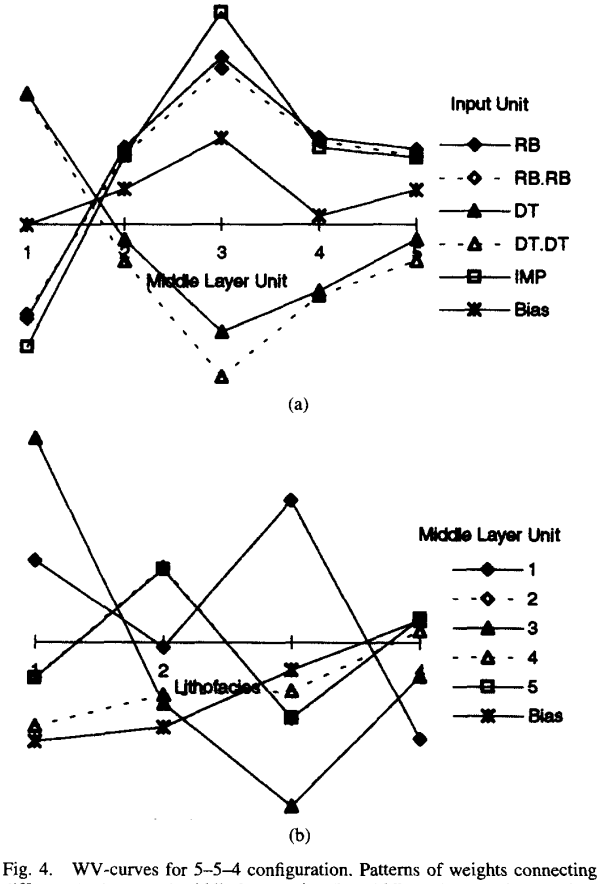


Fig. 4. WV-curves for 5–5–4 configuration. Patterns of weights connecting different (a) input and middle layer units, (b) middle and output layer units.

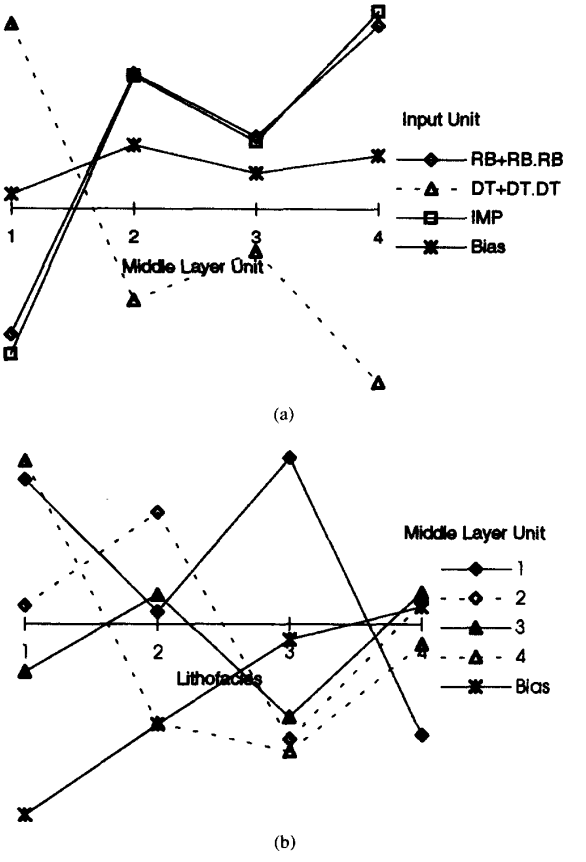


Fig. 5. WV-curves for 3–4–4 configuration (combining inputs). Patterns of weights connecting different (a) input and middle layer units, (b) middle and output layer units.

and local minimum detection and avoidance methods are active areas of research on neural nets [11], [12], [29], [30]. Careful examination of training data is crucial before the training process starts. Sets of training patterns which do not adequately distinguish between different facies groups will invariably result in either slow convergence or nonconvergence. Developing training data usually includes the choice of training variables and their representative samples and data normalization. Details of training patterns development can be found in [9], [10], [24].

Lithofacies information extracted from core samples is usually used as the training data set in classification of well log signals because of its high reliability and accuracy compared to well logs. After depth-matching core and log data, the corresponding well log signals (e.g., gamma ray, sonic travel time and bulk density) can be read for each core sample, and removal of outliers is usually done. The input data is then normalised in the interval (0, 1). The output data, however, is practically normalised in the interval (0.1, 0.9) for faster convergence [9]. There are no restrictions on how to normalise the data. The whole data set (i.e., the training and validation data) must, however, be normalised in the same manner.

Not all the variables display the distinct characteristics of each lithofacies. The most common selection rule is to choose the log variables with strong discriminating power of lithofacies. If the less selective variables are also included in the training set, the amount of training time required may increase. The advantage of applying the genetic approach (the use of lithofacies information in predicting porosity values)

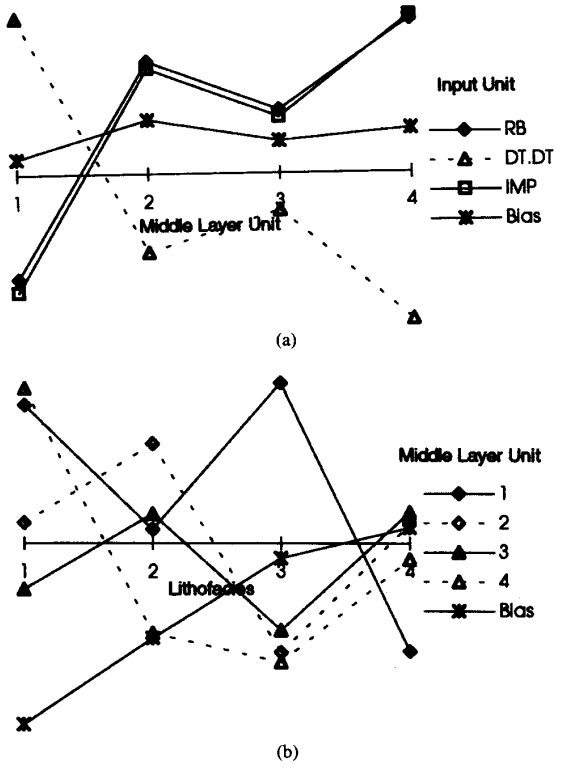


Fig. 6. Wv-curves for 3–4–4 configuration (discarding inputs). (a) Patterns of weights connecting different (a) input and middle layer units, (b) middle and output layer units.

is to provide additional information allowing the network to learn the separation of patterns in the input log data, and hence training time will be significantly reduced with high prediction accuracy. Also, the generation of nonlinear input variables [18], [27] may also reduce training time and avoid the use of more than one middle layer in some cases.

IV. Weight Visualization Curves

One of the major difficulties during neural net learning is the choice of the network parameters. This usually includes the weight initialization, the choice of input variables and the number of middle layer nodes. The choice of input variables is usually straightforward. However, if nonlinear inputs, such as $X1^2, X2^2$ or $X1 \cdot X2$ in Fig. 1 (not shown), are used, the optimum choices may be difficult. The number of middle layer nodes are also important. If too few nodes are present in the middle layer, the network will fail to learn. On the other hand, if too many nodes are used, the network will memorise the training data and fail to generalise the data trend. Therefore, techniques to interpret the trained network are essential. Weight visualization curves (WV-curves) have been used to represent the spectral character of the training data, and the possible use of these curves in feature selection has been discussed in [3]. However, the use of these curves in solving these problems have not been previously presented.

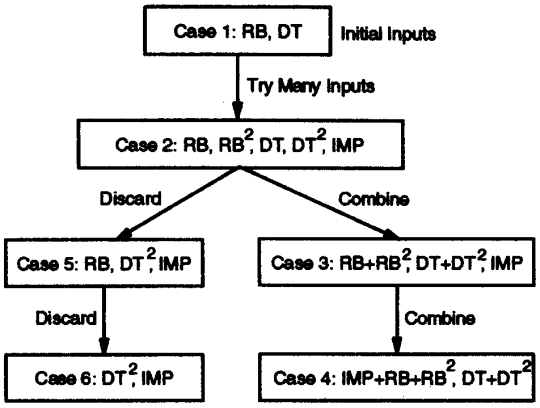


Fig. 7. Relationships of different experiments.

In this study, the presentation style of WV-curves are modified, and their application to select the number of input and middle layer nodes are also discussed. The weight values are used to calculate the average contribution of a node in a layer to a node in the next layer

$$P_{ij} = \frac{|W_{ij}|}{\sum_{k=1}^{n} |W_{ik}|} \cdot 100\% \tag{2}$$

where P_{ij} is the average contribution of a node i in a layer to a node j in the next layer, W is the weight on the connection, and n is the number of nodes in the next layer.

The meaning of (2) can be easily visualised with an example. For the 2-3-3 configuration (see Fig. 1), the WV-curves are shown in Fig. 2. In Fig. 2(a), the horizontal axis shows the node number in the middle layer, and the vertical axis (not shown) represents the size of the actual weights connecting each of these nodes to the two input nodes and the bias. The horizontal axis is located at the zero weight value level. For example, if j represents the node 2 (see horizontal axis), then P_{ij} is the highest for input node $X1$ because the weight connecting $X1$ node and node 2 in middle layer is the largest. In order words, $X1$ has the largest contribution to node 2. Similarly, node 1 in the middle layer contributes more to output node $Y3$ compared to node 3 [Fig. 2(b)].

Using (2), a similar expression can also be used to measure the average contribution of an input variable to the middle layer

$$A_i = \frac{\sum_{j=1}^{n2} |W_{ij}|}{\sum_{k=1}^{n1} \sum_{j=1}^{n2} |W_{kj}|} \cdot 100\% \tag{3}$$

where A_i is the average contribution of input variable i, $n1$ and $n2$ are the number of nodes in the input layer (including bias) and the middle layer respectively.

The use of WV-curves in optimizing the network configuration will be discussed in the later sections.

V. Case Study

A. Objective

The objective of this study is to compare the genetic and nongenetic approaches in well log analysis using backprop-

TABLE I
PERFORMANCE OF EACH NEURAL NETWORKS IN LITHOFACIES CLASSIFICATION. ALL CASES CAN ACHIEVE A RECOGNITION RATE OF 95% ON THE TEST DATA EXCEPT CASE 6 WHICH CAN ACHIEVE ONLY 88% AT 10000 EPOCHS. E AND E-AVE REPRESENT THE NUMBER OF EPOCHS FOR A PARTICULAR RUN AND THE AVERAGE NUMBER OF EPOCHS FOR THE TEN DIFFERENT RUNS RESPECTIVELY

Case	Net	RB	RB^2	DT	DT^2	IMP	RB+RB^2	IMP+RB+RB^2	DT+DT^2	E	E-ave
1	2-4-4	y		y						4982	4730
2	5-5-4	y	y	y	y	y				1568	2830
3	3-4-4					y	y		y	1324	1940
4	2-4-4							y	y	3611	2680
5	3-4-4	y			y	y				1238	1850
6	2-4-4				y	y				n/a	n/a

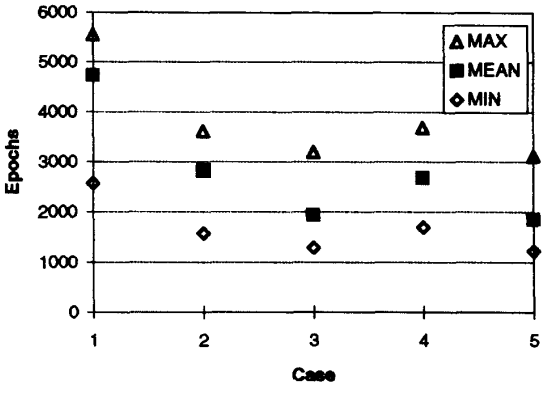


Fig. 8. Minimum, maximum and average epochs taken to achieve a classification accuracy of 95% for each case. Descriptions of each case refer to Fig. 7.

agation neural nets. The genetic approach is a two-stage process which involves the classification of log data into different lithofacies groups, and then porosity is estimated using both the log data, and the previously classified lithofacies information. The nongenetic approach involves only the use of well log data in porosity predictions. The predicted results are then compared to the core data (i.e., actual values) using these two different approaches.

B. Training and Test Data

Two wells named Well A and Well B, from an oil-bearing reservoir located at Carnarvon Basin of North West Shelf in Australia, were used to provide log and core data. The data set consisted of gamma ray (GR), deep induction resistivity (ILD), bulk density (RB), sonic travel times (DT), porosity, and lithofacies information of four different kinds: mudstone (facies 1), sandy mudstone (facies 2), sandstone (facies 3) and carbonate cemented bed (facies 4). The data set from Well A (120 core and log data) was chosen to provide the training patterns. This training set was then used to predict lithofacies and porosity values using the log data from Well B (284 data) where core data (i.e., lithofacies and porosity) was also available for comparison purposes.

The data set from Well A was carefully examined, and 55 samples were identified as outliers. These outliers were defined as the unrepresentative samples, such as data extracted from fractured core plugs and data which are affected by thin-bedding of lithofacies which give rise to "shoulder effects" on log readings. Further examination showed that neither GR nor ILD could by itself discriminate the four lithofacies groups identified from the cores, which means that each lithofacies displayed a similar range of log values and resulted in large overlapping regions. Hence, the training data set was constructed using the remaining 65 patterns at Well A. In this study, nonlinear inputs were also generated using RB^2, DT^2 and RB/DT (i.e., acoustic impedance) or simply IMP. These nonlinear inputs were constructed for faster and better convergence purposes. Using RB^2 for example, effectively modified the transfer function in the next layer of nodes for this variable only. The cross-plot of RB and DT is displayed in Fig. 3. This figure shows that there is an inverse relationship between these two variables. The input and target training data at Well A were then normalised in the interval (0, 1) and (0.1, 0.9) respectively, using simply the minimum and maximum values of the whole data set. The same scaling was also applied to the test data at Well B.

Once the lithofacies patterns were determined, porosity was estimated using the genetic approach. The results were then compared with the nongenetic case using only the log data. Core data from Well B, consisted of 66 points of lithofacies and porosity, were used to test the performance of different approaches. Also, because DT related closely to lithofacies and porosity, the predicted lithofacies and porosity profiles were plotted with this log as the indicators of locations of geological bedding planes and porosity levels.

C. Lithofacies Classification

In this section, we will discuss the use of WV-curves in optimizing the network used for lithofacies classification purposes, however, the same approach can be applied to other applications as well.

The 65 training patterns from Well A, composed of RB, RB^2, DT, DT^2, IMP and known lithofacies information, were used to classify the log data from Well B into the four lithofacies groups. One output node was assigned to each lithofacies, and hence four output nodes were required. The resulting lithofacies was determined based on the largest response on the corresponding output node. Only one middle layer was considered in this example. The number of optimum input nodes was still uncertain because nonlinear inputs were used. The number of nodes present in the middle layer was also not known. Hence, some guess work was required at the start. For each experiment, the network was trained for 10,000

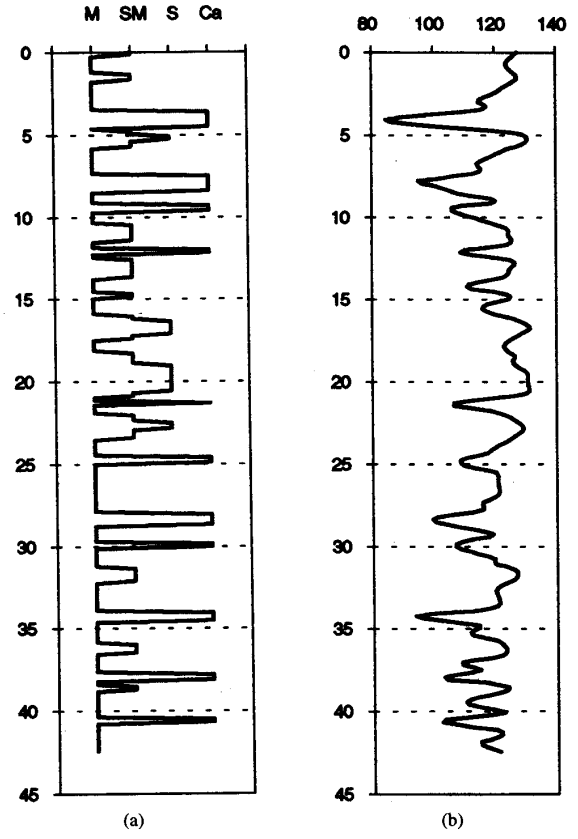


Fig. 9. Predicted Lithofacies Profiles at Well B. M is Mudstone, SM is Sandy Mudstone, S is Sand, and Ca is Carbonate Cemented Beds. Vertical axis is depth indicator. (a) Neural network results, (b) Sonic Log (usec/ft).

TABLE II
PERCENTAGE CONTRIBUTION A_i OF EACH INPUT VARIABLE (INCLUDING BIAS) IN DIFFERENT NEURAL NETWORKS

	Lithofacies Classification	Porosity Predictions	
		Nongenetic	Genetic
RB	30.2	29.3	11.4
DT	n/a	38.8	11.8
DT.DT	29.3	n/a	n/a
IMP	30.3	31.4	29.4
Lithofacies	n/a	n/a	37.3
Bias	10.1	0.6	10.0

epochs. The 66 test data from Well B were also used to record the highest recognition rate (%Reco) during training phase.

In this study, the first try was using only the two independent inputs (i.e., RB and DT). The results were optimised at 4982 epochs using four middle layer nodes, and the %Reco was 95% on the Well B data set. These results were then used to provide a minimum baseline level of performance to achieve.

The second try was using the five linear and nonlinear inputs (i.e., RB, RB^2, DT, DT^2, and IMP) and five middle layer nodes (i.e., 5-5-4). The results were maximum at 1568 epochs, and the %Reco was 95% on the Well B data set. Compared to the previous experiment, these two networks achieved the same amount of accuracy, however, the results converged at

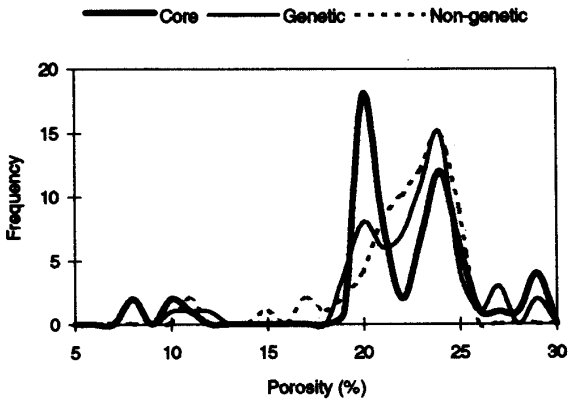


Fig. 10. Histograms of core and predicted porosity values.

less epochs. These showed that creating nonlinear inputs in network training do not necessarily give extra information in the input space, but using these inputs can certainly reduce the training time by a significant amount. The WV-curves for 5-5-4 configuration at 1568 epochs are shown in Fig. 4. These curves showed the relationships between the logs and the lithofacies groups as displayed in Fig. 3. For example, the output value at facies 3 (i.e., sandstone) was high when the middle layer unit 1 gave a high value [Fig. 4(b)]. This happens when DT was also high, and that means DT has a positive contribution to sandstone as shown in Fig. 3.

When two WV-curves are similar, there are two diametrically opposite possibilities. Either both are necessary, or we can dispense the one with a smaller contribution to the network. An example of the former case occurs in the exclusive-or (XOR) problem, where the weights of both inputs would have identical or similar sets of values [6], but both are required. The omission of one input would render the network invalid, however, the addition of the input values would produce correct results since these signals are combined linearly in the network. An example of the latter is the case of $X1$ and $X1^2$ for instance, where one of the inputs could usually be omitted. In actual use, neither case is usually as clear cut as this, hence our following experiments to discover the best means to decide what to do based only on the performance and WV-curves.

In Fig. 4(a), the weights of RB and RB^2, and DT and DT^2 to the middle layer have almost the same values. That means RB and RB^2 (or DT and DT^2) can either be replaced by their sum, or one of them omitted. Note that in this figure, the actual weight values at nodes 2 and 5 from all inputs are nearly the same. This also shows in Fig. 4(b) where middle layer nodes 2 and 5 contribute nearly identical amount to all the four output nodes. Therefore, one node can be removed from the middle layer, and a 3-4-4 configuration (i.e., RB+RB^2, DT+DT^2, IMP) was used in the next experiment. The results were maximum at 1324 epochs with the same %Reco (i.e., 95%). The WV-curves for this configuration are shown in Fig. 5. In Fig. 5(a), the WV-curves for IMP and RB+RB^2 are nearly the same. Hence the further simplification

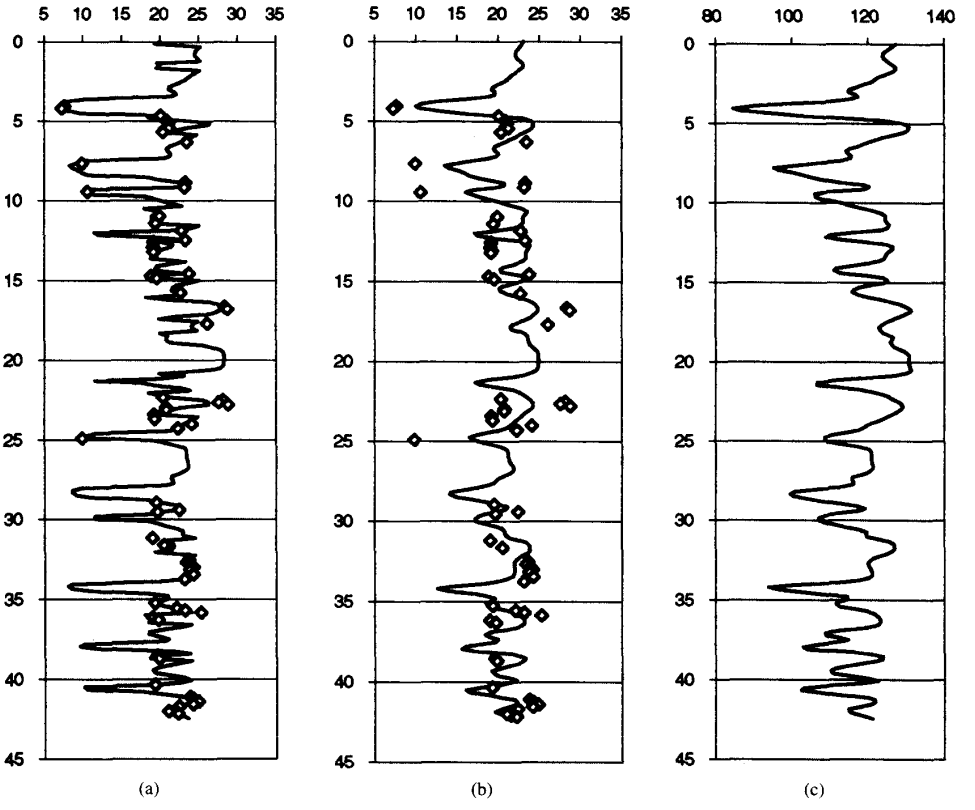


Fig. 11. Comparisons of neural network-derived porosity profiles at Well B. Diamonds indicate core porosity. Vertical axis is depth indicator. Predicted porosity profile (%) by (a) genetic approach, (b) nongenetic approach. (c) Sonic log (usec/ft).

was performed using IMP+RB+RB^2 and DT+DT^2 (i.e., 2-4-4), the results converged at 3611 epochs.

The other alternative in network simplification is used by discarding the input with smaller contribution to the network. In Fig. 4(a), RB and RB^2, and DT and DT^2 had similar WV-curves. By comparing A_i's using (3), RB and DT^2 contributed more than RB^2 and DT respectively. Therefore another network of 3-4-4 was tested using RB, DT^2, and IMP. The results of this network converged at 1238 epochs with the same %Reco. The WV-curves are displayed in Fig. 6. In Fig. 6(a), RB and IMP had similar WV-curves, and IMP contributed more than RB. Therefore, a network of two inputs (i.e., DT^2 and IMP) was further tested but the highest %Reco was only 88% on Well B data set within the 10 000 training epochs.

Fig. 7 shows the derivation relationships of the six cases. Each of these cases was done for ten runs using different initial random weights. A summary of these experiments is plotted in Fig. 8 and is also tabulated in Table I. Fig. 8 shows the minimum, maximum and average epochs taken to achieve the same classification accuracy (i.e., 95%) for all the ten runs. Note that Case 6 could only achieve a maximum accuracy of 88% in 10 000 epochs for all runs. Table I shows the epochs for one set of runs, and is used in Figs. 4 to 6 displaying the WV-curves. The final column shows the average number of epochs which is also displayed in Fig. 8. Case 1 performed the least well on average, followed by the better Cases 2 and 4, and then Cases 3 and 5. Case 2 shows that providing extra preprocessed information (i.e., nonlinear inputs) to a network can increase its learning speed by reducing the number of epochs taken to reach a desired classification level. Case 4 shows that combining (or adding) all three inputs (i.e., RB, RB^2, and IMP) which were similar on WV-curves did not particularly improve average performance. Case 3, however, shows that some combination of inputs of this nature can improve performance. Case 5 shows that keeping only the larger of a pair of similar inputs produced the best result, implying that the pair of input variables were dependent (we knew this already but have now shown this behaviour). Finally, Case 6 shows that removing one of a pair of independent inputs will adversely impact on performance (refer to our earlier discussion on the XOR problem).

The Case 5 results were used as the final configuration since it took the least epochs. The predicted lithofacies profile is displayed in Fig. 9 with the DT log at Well B. The results showed that the lithofacies profile was highly correlated with the sonic log. The contribution, A_i, of each input log to the network is listed in Table II using (3).

D. Porosity Prediction

The training data set in prediction of porosity values from well logs was the same as the one for lithofacies classification,

except lithofacies information was used as an extra input variable, and the core porosity values as target data. The inputs used in porosity predictions were RB, DT, IMP, and lithofacies. One output node (i.e., porosity) was used to develop the network configuration, and three middle layer nodes were found to produce the best results by using some statistical measurements such as correlation coefficient and mean-square-error (MSE). The Well B data set was also used during each epoch in order to prevent over-fitting. The network was trained for 10 000 epochs, and the results converged at 2345 epochs with a minimum MSE of 0.0016 and a correlation coefficient of 0.94.

The nongenetic approach used only RB, DT, and IMP as input variables. The network optimised using two middle layer nodes (i.e., 3-2-1 configuration). The MSE did not reduce below 0.01 within 10 000 epochs. The correlation coefficient was only 0.70. The lithofacies information provided additional separation of patterns of the input vector, and the removal of this variable from the training set resulted in averaging porosity values for all lithofacies. Hence, the MSE was high.

The value of the output neuron was a continuous value which was rescaled to get the predicted porosity value. The predicted porosity values are displayed in Fig. 8 as histograms. The genetic-derived porosity produced a similar distribution to the core data. The nongenetic-derived porosity, however, tended to smooth out the profile and produces a high frequency content in the 20–25% interval. The predicted porosity values are also shown in the form of a porosity log which is shown in Fig. 9(a) and (b) together with the core data and the corresponding DT log along Well B (Figs. 10–11). Once again, the results showed that the genetic-derived porosity profile was in a better agreement with the core data (correlation coefficient = 0.94 and also showed a better correlation with the DT log in terms of geological beddings, in comparison with the nongenetic-derived porosity values (correlation coefficient = 0.70). The contribution of each input variable to the system is tabulated in Table I. It shows that lithofacies was a very important variable in porosity estimations.

VI. Conclusion

The results of porosity predictions using wireline log signals were compared for genetic and nongenetic approaches in reservoir characterization. The genetic approach involves the classification of log data into different lithofacies groups, and then porosity descriptions are carried out on a facies-by-facies basis, while the nongenetic approach uses only the log data, which means that effectively only one "averaged" lithofacies is considered. A technique for optimizing the network architecture has been proposed using the weight visualization curves. Based on the results obtained from the example problem, the major findings are listed below:

1) The genetic approach in well log analysis can be implemented using a backpropagation neural network.
2) Using nonlinear input reduces significantly the amount of training time required.
3) Weight visualization curves are very useful in understanding the trained neural network.
4) The genetic approach provides superior porosity estimates to that based on the nongenetic approach.

Future studies on different formations will be required to test the generality of the conclusions drawn.

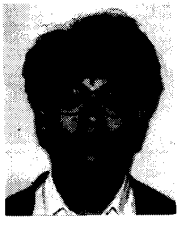
Acknowledgment

The authors would like to thank the reviewers for time and consideration. Their recommendations were very helpful in improving this manuscript.

References

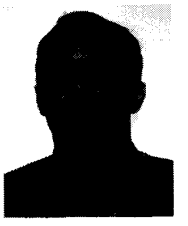
[1] J. L. Baldwin, A. R. M. Bateman, and C. L. Wheatley, "Application of neural network to the problem of mineral identification from well logs," *Log Analyst*, vol. 3, pp. 279–293, 1990.
[2] J. L. Baldwin and D. N. Otte, "Computer emulation of human mental processes: Application of neural network simulators to problems in well log interpretation," presented at the *64th Ann. Tech. Conf. Exhibition Soc. Petroleum Eng.*, 1989.
[3] H. Bischof, W. Schneider, and A. J. Pinz, "Multispectral classification of landsat-images using neural networks," *IEEE Trans. Geosci. Remote Sensing*, vol. 30, pp. 482–490, 1992.
[4] M. Caudill, "Neural networks primer: Part III," *AI Expert*, vol. 3, pp. 53–59, 1988.
[5] D. Z. Cheng, X. L. Wu, and J. A. Cheng, "Determining reservoir properties in reservoir studies using a fuzzy neural network," presented at the *68th Ann. Tech. Conf. Exhibition Soc. Petroleum Eng.*, 1993.
[6] J. E. Dayhoff, *Neural Network Architectures: An Introduction*. New York: Van Nostrand, 1990.
[7] P. C. Delfiner, O. Peyret, and O. Serra, "Automatic determination of lithology from well logs," presented at the *59th Ann. Tech. Conf. Exhibition Soc. Petroleum Eng.*, 1984.
[8] H. Derek, R. Johns, and E. Pasternack, "Comparative study of back-propagation neural network and statistical pattern recognition techniques in identifying sandstone lithofacies," *Conf. Artificial Intellig. Petroleum Exploration Production*, 1990, pp. 41–49.
[9] R. C. Eberhart and R. W. Dobbins, *Neural Network PC Tools: A Practical Guide*. San Diego, CA: Academic, 1990.
[10] T. D. Gedeon and T. G. Bowden, "Heuristic pattern reduction," *Int. Joint Conf. Neural Networks*, 1992, vol. 2, pp. 449–453.
[11] T. D. Gedeon and D. Harris, "Hidden units in a plateau," *Proc. 1st Int. Conf. Intelligent Syst.*, 1992, pp. 391–395.
[12] M. Gori and A. Tesi, "On the problem of local minima in backpropagation," *IEEE Trans. Pattern Analysis Mach. Intellig.*, vol. 14, pp. 76–86, 1992.
[13] H. H. Haldorsen and E. Damsleth, "Challenges in reservoir characterization," *AAPG Bulletin*, vol. 77, pp. 541–551, 1993.
[14] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, pp. 359–366, 1989.
[15] C. L. Hearn, W. J. Ebanks, R. S. Tye, and V. Ranganathan, "Geological factors influencing reservoir performance of the Hartzog draw field, Wyoming," *J. Petroleum Technol.*, vol. 36, pp. 1335–1344, 1984.
[16] B. Irie and S. Miyake, "Capabilities of three-layered perceptrons," *Int. Joint Conf. Neural Networks*, 1988, vol. 3, pp. 1641–1648.
[17] F. X. Jian, C. Y. Chork, I. J. Taggart, D. M. Mckay, and R. M. Barlett, "A genetic approach to the prediction of petrophysical properties," *J. Petroleum Geology*, vol. 17, pp. 71–88, 1994.
[18] R. P. Lippmann, "Pattern classification using neural networks," *IEEE Commun. Mag.*, vol. 27, pp. 47–64, 1989.
[19] D. A. Osborne, "Permeability estimation using a neural network: A case study from the Roberts unit, Wasson field, Yoakum Country, Texas," *Conf. Artificial Intellig. Petroleum Exploration Production*, 1992, pp. 89–96.
[20] S. J. Rogers, J. H. Fang, C. L. Karr, and D. A. Stanley, "Determination of lithology from well logs using a neural network," *AAPG Bulletin*, vol. 76, pp. 731–739, 1992.
[21] D. E. Rumelhart, G. E. Hinton and R. H. Williams, "Learning representations by backpropagating errors," *Nature*, vol. 323, pp. 533–536, 1986.
[22] S. Sakurai and J. Melvin, "Facies discrimination and permeability estimation from well logs for the endicott fields," *Society of Professional Well Log Analyst 29th Ann. Logging Symp.*, Paper FF, 1988.

[23] O. Serra and H. T. Abbott, "The contribution of logging data to sedimentology and stratigraphy," presented at the *55th Ann. Tech. Conf. Exhibition Soc. Petroleum Eng.*, 1980.
[24] P. Slade and T. D. Gedeon, "Bimodal distribution removal," in J. Mira, J. Cabestany and A. Prieto, Ed., *New Trends in Neural Computation*, vol. 686. Berlin: Springer Verlag, 1993, pp. 249–254.
[25] J. H. Stiles, Jr. and J. M. Huthilz, "The use of routine and special core analysis in characterizing brent group reservoirs, U. K. North Sea," *J. Petroleum Technol.*, vol. 44, pp. 704–713, 1992.
[26] W. A. Wendt, S. Sakurai, and P. H. Nelson, "Permeability prediction from well logs using multiple regression," in L. W. Lake and H. B. Carroll Eds., *Reservoir Characterization*. New York: Academic, 1986, pp. 181–222.
[27] B. Widrow and M. A. Lehr, "30 years of adaptive neural networks: Perceptron, madaline and backpropagation," *Proc. IEEE*, vol. 78, pp. 1415–1442, 1990.
[28] M. Wolff and J. Pelissier-Combescure, "FACIOLOG: Automatic electrofacies determination," *Soc. Professional Well Log Analysts 23rd Ann. Logging Symp.*, Paper FF, 1982.
[29] X. H. Yu, "Can backpropagation error surface not have local minima," *IEEE Trans. Neural Networks*, vol. 3, pp. 1019–1021, 1992.
[30] Y. X. Zhang and D. X. Wang, "Fast learning in a backpropagation algorithm with a sine-type thresholding function," *Appl. Opt.*, vol. 31, pp. 2414–2416, 1992.



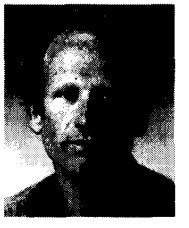
Patrick M. Wong received the B.E. degree in petroleum engineering and the Ph.D. degree at the University of New South Wales, Sydney, Australia, in 1992 and 1995, respectively.

He is currently a Petroleum Engineer with Petroconsultants Australasia Pty. Ltd., St. Leonards, Australia. His research interests include the use of artificial neural network method and geostatistical techniques in the area of reservoir characterization.



Tamás D. Gedeon received the B.Sc. and the Ph.D. degrees in computer science at the University of Western Australia, Sydney, Australia, in 1981 and 1989 respectively.

He is currently a Senior Lecturer in computer science and engineering at the University of New South Wales and a Visiting Research Fellow at the Centre for Computers in Law and Finance, Brunel University. His research interests are in program transformatin, extracting knowledge (data mining) from trained neural networks, and neural network applications.



Ian J. Taggart received the B.Math. in mathematical physics at the Newcastle University and the Ph.D. degree in Reservoir Engineering at the University of New South Wales in 1979 and 1992, respectively.

He was a lecturer in Petroleum Engineering at the University of New South Wales where he conducted research into reservoir characterization using geostatistical methods. He is currently a Reservoir Engineer with West Australian Petroleum Pty. Ltd.