# Application of artificial neural networks to segmentation and classification of topographic profiles of ridge-flank seafloor

**Bishwajit Chakraborty*, Eric Lourenco[#], Vijay Kodagali and Jennifer Baracho**

National Institute of Oceanography, Dona Paula, Goa 403 004, India
[#]Present address: 13, rue Dareau, 75014 Paris, France

**In this paper, we have utilized Artificial Neural Networks (ANN) for seafloor topographic data segmentation and roughness classification using the multibeam-Hydrosweep system (installed onboard ocean research vessel *Sagar Kanya*) data. Bathymetric profiles from three directions: central (beam 30), port side (beam 10) and starboard side (beam 50), were acquired from the ridge flank and rift valley areas of the Carlsberg Ridge and plain areas of the Central Indian Basin. Self-Organizing Map (SOM) – an ANN architecture employing unsupervised training is used for segmentation of the depth data. This neural architecture is successful in segmenting nonstationary profiles into stationary type before being used for classification. The number of segmented profiles is highest for the rift valley areas whereas for plain area profiles no segmentation is required. Another ANN architecture using supervised training – the Multi-Layer Perceptron (MLP) is applied for the classification of the segmented profiles in two steps. The MLP network was trained using simulated bathymetric profiles of known power spectral (correlation) parameter *b*, which has been used to classify the multibeam-Hydrosweep depth data. Another parameter *S* (amplitude parameter) is also being computed for classification. Estimated *b* and *S* values of the segmented profiles indicate the sedimentary origin of the plain area. Whereas the estimated parameters for ridge flank and rift values indicate that these areas have volcanic or tectonic origin.**

DYNAMIC processes related to the shape of the Earth may well be understood through use of seafloor bathymetry. Bathymetry provides the shape of the seafloor features, and its accuracy depends on the resolution of the sounding systems[1]. In order to characterize the nature of the seafloor, a detailed knowledge of the bathymetric profiles is required. Swath mapping systems produce multiple lines (typically tens to hundreds) of parallel high-resolution depth data. Using multibeam swath bathymetric systems, a complete plan view of topography, including the detailed shape and orientation of structures can be determined. The seafloor topography is influenced by tectonic processes such as faulting, folding and flexure and is often changed and destroyed by erosion and sedimentation[2]. Since much of the seafloor topography is complex, certain aspects are deterministic. This complexity is mainly the consequence of erosion processes. Therefore, a given profile of seafloor topography typically contains alternating 'smooth' and 'rough' portions, which generally implies that the topography is nonstationary. This nonstationary topography presents a sort of hindrance since most statistical analyses presume that the input is stationary. If we consider a seafloor topographic profile, a statistical analysis applied to this profile will produce results that are an unpredictable mixture of the characteristic parameters (mean, variance, probability density function) of the smooth and rough areas. It thus becomes necessary to divide the dataset into statistically homogeneous segments.

In this paper, we have employed Artificial Neural Networks (ANN) for data segmentation and subsequent classification of the seafloor depth[3]. ANN are signal-processing systems that try to emulate how a human brain classifies differing patterns. For online application, the ANN-based seafloor classifier serves as a real time classifier into different known classes. Two different types of ANN architectures, viz. Kohonen's Self-Organizing Map (SOM) and the Multi-Layer Perceptron (MLP)[4] have been employed for segmentation and classification respectively. The unsupervised architecture, SOM, finds application in segmenting a particular seafloor topographic profile. As mentioned, this segmentation is required to subdivide the bathymetric data that are received into blocks that are statistically stationary. The SOM architecture exhibits the ability to form segments suitable for further classification, that are being made based on differing patterns in the input raw data. Once the segmentation is done, classification of the seafloor based on the power spectral parameter, *b* (also known as a correlation parameter) can be determined using MLP. In order to carry out MLP-based depth data classification, one feature known as Mean of the Absolute First Difference (MAFD)[2], is generated from the simulated data for varying sets of *b* values,

*For correspondence. (e-mail: bishwajt@darya.nio.org)

which serve to train the MLP. Further, unknown depth data can be successfully classified using this trained network, to correctly identify *b* parameters. A flowchart comprising the brief functional units is given in Figure 1.

The topographic data were obtained on the Carlsberg Ridge (CR) (Figure 2) and in the Central Indian Basin (CIB) areas. These areas were surveyed using a multibeam bathymetric system – Hydrosweep (manufactured by M/s STN Atlas Electronic GmbH, Bremen, Germany) installed onboard *ORV Sagar Kanya*.

## Survey data acquisition and preliminary depth data analyses

The topographic data were obtained from the CR area. The CR is a slow-spreading ridge, similar to the Mid-Atlantic Ridge. It forms the northwest extension of the Central Indian Ridge in the Arabian Sea. A total area of 15000 sq km was surveyed with a state-of-the-art multibeam bathymetric system[5]. Depth data from two area types of the CR-flank and the rift valley were considered for this study. Bathymetric data from central beam (beam 30), beam 10 (port side) and beam 50 (starboard side) were picked up from the acquired cruise data. In addition to this, data were also collected from the CIB (polymetallic nodule-bearing area) during the cruise SK-136 (ref. 6). Thus, various topographic domains, viz. rift valley, ridge flank and plain areas were considered for the study. Ridge flank and rift valley area from the CR data were collected from the locations: (long. 63.2°E, lat. 3.24°N) and (long. 63.9°E, 3.27°N) respectively. Similarly, CIB plain area data were obtained from the location: (long. 75.4°E and lat. 13.3°S). The depth varied between 2200 m and 4500 m. The average beam footprint of these areas is found to be varying between the 177 m and 361 m. The beam-wise depth profiles used for this study areas are given in Figure 3. The depth data lengths of 30 km, 22 km and 32 km are used from CIB plain and CR flank and rift valley areas for present applications.

In order to develop understanding of the bathymetric data type used for this work, the Probability Density Function (PDF) of the beam-wise raw depth data is examined critically. The PDF characteristics of the raw data for

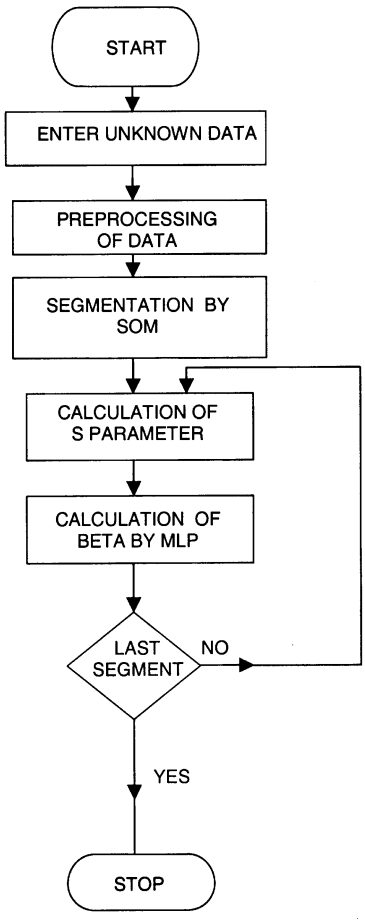


**Figure 1.** Flowchart showing the important components in the process of segmentation and classification of multibeam backscatter data.

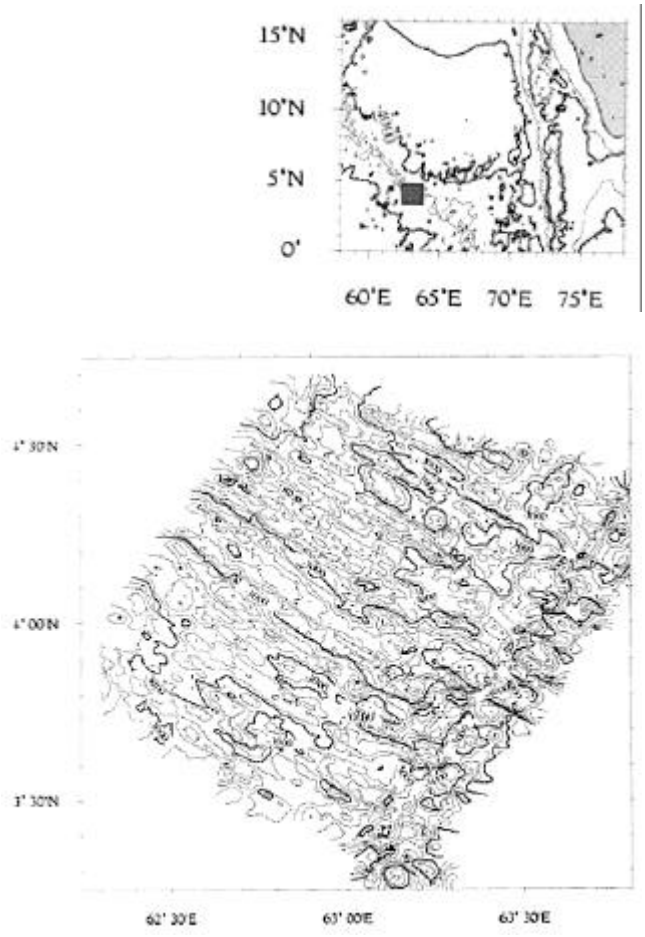


**Figure 2.** Bathymetric map of the surveyed segment of the Carlsberg Ridge (contour interval = 250 m). Topographic sample data from rift valley and Ridge flank areas are taken from this area.

central and outer beams of different geological provinces do not indicate any changes, which emphasize nonstationarity of data. However, analyses using PDF of the detrended central beam depth data show the normal distribution (Gaussian type) for the plain, flank and valley areas, which are presented in Figure 4. For a particular area dataset, the computed mean depth was utilized to normalize the entire dataset for drawing PDF curves. It was found that the plain area surveyed using the central beam shows minimum fluctuation compared to the beams 10 (port side) and 50 (starboard side) of the same area because the width of the PDF curve is observed to be significantly narrower than that of the outer beam data. Also, the widths of the central beam PDF characteristics for the ridge flank and rift valley regions are relatively higher than that of the plain area. Interestingly, for all the areas, it is observed that the central beam PDF characteristics are narrower than their outer beam characteristics. This indicates relatively less efficient functioning of the bottom-tracking unit for the outer beams of the multibeam system. This is attributed to the dominant seafloor backscatter for outer beams relative to the central beam, which is nevertheless important for rough terrain. The depth data detrending aimed at making the data similar to those of a stationary nature, however it is problematical to achieve such conversion during real time application. In this article we attempt to enable real-time application with the neural networks approach on raw nonstationary data.

## Segmentation of a bathymetric profile using SOMs

In terms of seafloor roughness, different bathymetric profiles having the same statistical parameters may look similar, but in reality they will not necessarily be the same. Therefore, it becomes difficult to quantify the roughness of such bathymetric profiles[2]. Under these circumstances, the concept of self-similarity or scale invariance has an important role to play in seafloor bathymetric studies. Considerable study has been carried out to quantify the roughness based on the properties of self-similarity. However, due to the nonstationarity of the profile, a straightforward application of self-similarity becomes strenuous. If a statistical analysis is applied to such nonstationary profiles, which are unpredictable mixtures of the known statistical parameters (mean, variance, PDF) of the smooth and rough areas, it will reduce accuracy of the results. Thus it becomes essential to divide the bathymetric data

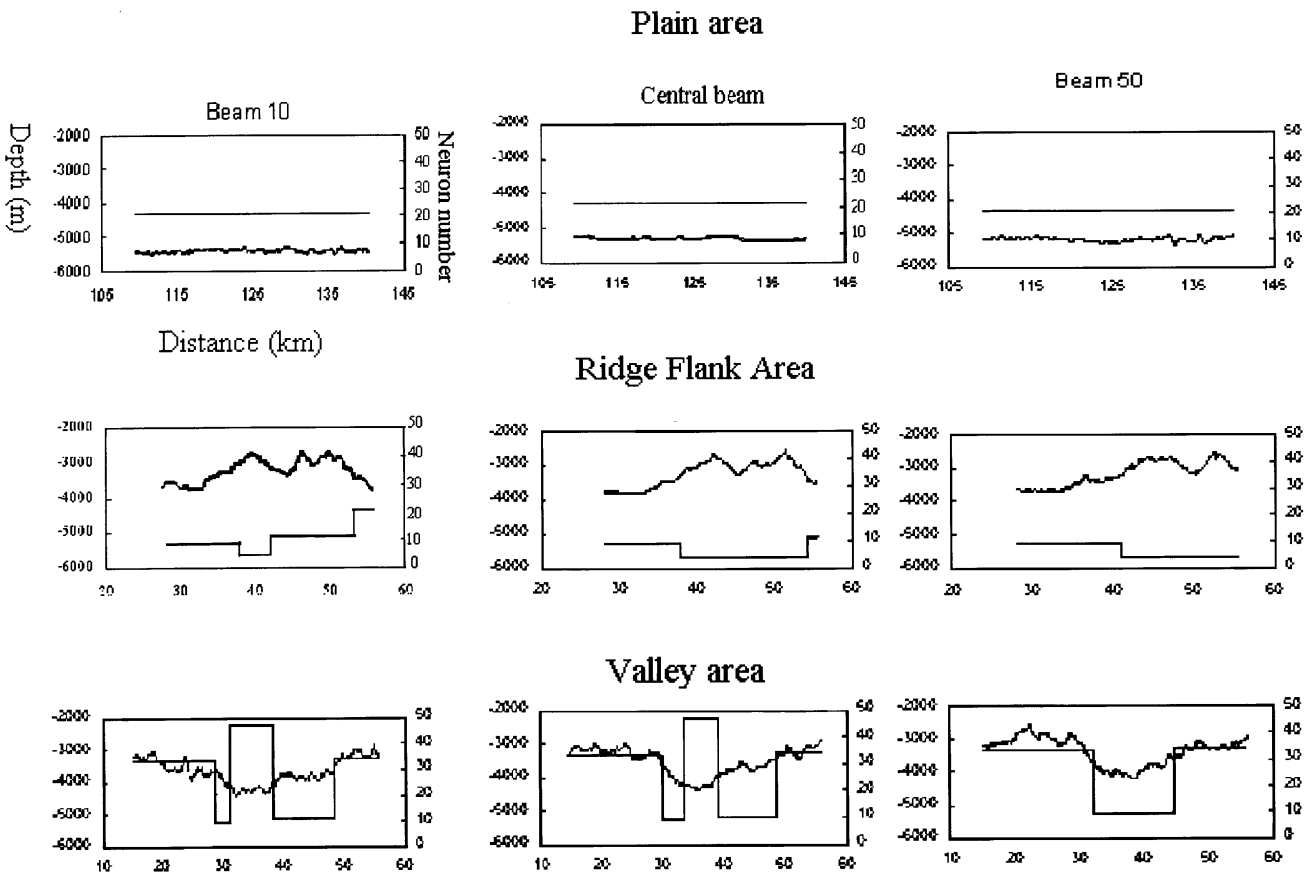


**Figure 3.** Segmentation of the bathymetric profiles for central beam (beam 30), beam 10 (port side) and beam 50 (starboard side), each for the plain area of the Central Indian Ocean Basin and the Ridge flank area and Rift valley area of the Carlsberg Ridge.

into stationary segments and then an estimate of the statistical characteristics of each segment, i.e. classification, can be made. One of the simplest ways to deal with such a nonstationary series is to segment it into a number of portions. For segmentation, the optimum gate length technique was applied by Bansal and Dimri[7]. In their method, an arbitrary length of the given profile is chosen and its time varying autocorrelation function is computed. Stewart *et al*.[8] found that if a profile was segmented into too many portions, some of the useful higher frequency components from the profile would be lost. Use of polynomial fitting to qualitatively selected segments and subtraction of the fitted data from original data is performed. This is done in order to detrend the data, i.e. to remove the nonstationarity of the profile. However, no check of the successes of the segmentation is made except based on visible distinction. Malinverno[2] performed segmentation using standard deviation and Median of Absolute Deviation (MAD) of the first difference of the data as estimates of dispersion. The suggested methods provide an approximate location for the boundaries and values of the spectral parameter ($b$). In this section, we propose to deal with the problem of segmentation using SOM[9–11]. The SOM uses unsupervised learning to train itself, where the network is unaware of the number of segments among which a particular set of depth data will be divided. For precise segmentation, we present an overview of SOM in the next section.

## *Self-organizing maps*

The SOM architecture comprises a flat one-dimensional grid of learning units called neurons, similar to those in the human brain[4]. Every input element is connected to all the neurons in the grid. When an input vector is presented to the SOM, the neurons in the grid compete among themselves in order to get activated. This is known as competitive learning. Weights of only this neuron and a few others in its neighbourhood are updated iteratively, to form a representative cluster. This is known as tuning of weights in response to a given class of input vectors. Once the SOM has been trained with given dataset, a unique neuron cluster in the output grid represents the segments of different depths. The learning algorithm used in a SOM is the Kohonen learning algorithm[9–11]. This algorithm organizes the nodes in the grid into local neighbourhoods that act as feature classifiers on the input data. A weight matrix of size ($i$, $j$) is initialized with random values between a range of + 1 to – 1. Here $i$ represents the number of elements of each input vector, while $j$ is the number of neurons. One input vector from each class is presented as a training sample from that class. For each input presented, the Euclidean distance between the input vector and the weights of each neuron in the one-dimensional grid is computed by

$$\mathrm{d}_j = \sum_{i=0}^{n-1} [x_i(t) - w_{ij}(t)]^2 ,$$

where $x_i(t)$ is the input vector and $w_{ij}(t)$ is the weight vector at time $t$. The neuron having the least distance is designated the 'winner' neuron. Now the weights of this neuron are updated and the distance matrix is computed iteratively for the same input vector to minimize the error between itself and the input presented, using the following expression

$$w_{ij}(t+1) = w_{ij}(t) + h(t)\cdot[x_i(t) - w_{ij}(t)].$$

This procedure is repeated consistently a number of times till the value of $\mathrm{d}_{\min}$ reduces below a pre-specified error value. The term $h(t)$ is a learning function ($0 < h(t) < 1$) that decreases with time, gradually reducing the magnitude of weight updating as the error is successively reduced. The neighbourhood size also decreases as time goes by, thus localizing the area of maximum activity in response to input vectors.

## *Application of SOM in segmentation*

In this study we have employed the SOM in order to segment a given Hydrosweep-multibeam bathymetric profile. The SOM architecture in this study comprises a one-dimensional grid of fifty neurons[11]. We chose the fixed number of input vectors to be ten. In our study, we optimized output neuron number $j$ to be 50. Once the network is initialized, a bathymetric profile of ten initial points is chosen and smoothing is done using the ten points moving average technique. The weight matrix described has a size of 10 × 50. Smoothing of data is basically done so that the spurious depth values in the profiles are

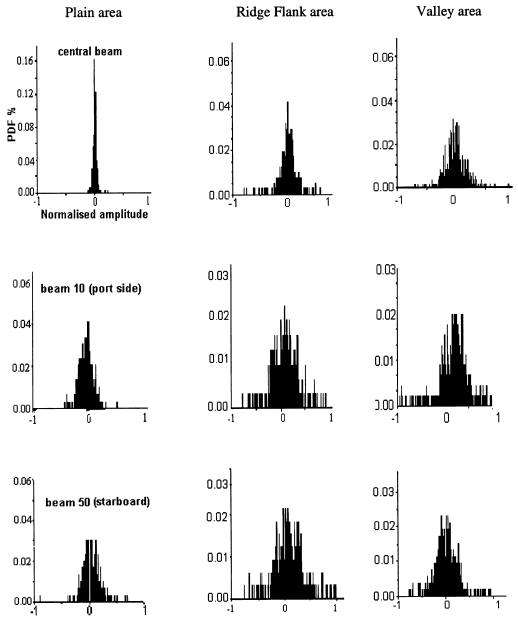


**Figure 4.** PDFs of depth data for the plain area of the Central Indian Ocean Basin, Ridge flank and rift valley areas of the Carlsberg Ridge.

removed. These spikes may cause improper segmentation while applying neural networks. The moving average of 10 data points was found to be optimum, which has been obtained by examining different number of data points from different areas. Once the smoothing of a predetermined length of the bathymetry data is done, it is then normalized between + 1 and – 1 to match the weight distribution, so that all the neurons in the output grid have an equal chance of being close to the input vectors. The winning neuron corresponding to this dataset is noted. Only the weights of the neurons that lie in this neighbourhood are updated during successive passes through the network. The learning function $h(t)$ employed here is $0.4/t^{0.20}$ and $t$ is the iteration number. During each successive iteration, the inputs and weights are used to compute the minimum distance. In this work, the neighbourhood was chosen as 5 (either side of the winning neuron), which subsequently reduces to 1 as training progresses. Training stops when the specified number of 500 iterations is reached or an error goal of $10^{-30}$ is attained for all the areas. Once training is done, the SOM is now tested with the next ten smoothed data points, for example (2 to 11). The rule is that if this set succeeds in exciting a neuron within a range (in the grid) of the winning neuron that was obtained during training, then this set is said to belong to the same segment with which the SOM was trained. If another neuron gets activated during the testing, then the same dataset which causes the misfiring is used to further train the SOM and a new winning neuron corresponding to a different segment is obtained. This procedure of training and testing is employed using successive sets of smoothed data points as mentioned above. At the end of all this successive training and testing, the final result is a profile consisting of precisely distinguishable segments (Figure 3).

Results of the segmentation study using the SOM along with the bathymetric data profile from plain areas of Central Indian Ocean Basin reveal only one segment for the bathymetric profiles of central beam (beam 30), beam 10 (port side), and beam 50 (starboard side) (Figure 3). The firing (excitation) of neuron number 21 in the output grid indicated the segments. In this figure, left and right hand side vertical axes are indicative of the depth and output neuron numbers respectively. Each segment is identified by a unique step, along with the segment number. Therefore, the plain areas surveyed with all the three beams produced only one segment. The results for the CR flank and rift valley are also given in the same figure. The CR flank area, whose depth data PDF (Figure 4) had a relatively larger width (comparatively more fluctuations) than the plain area, provides higher number of segments. The central beam in this region produces three segments shown by the excitation of neurons 9, 4, and 11. The area surveyed with beam 10 (port side) had four segments indicated by the excitation of neuron numbers 9, 4, 11 and 21, while the beam 50 (starboard side) had only two segments indicated by neurons 9 and 4. Of all the three region's central beam depth PDF, the rift valley areas central beam has the maximum PDF width, and also the highest number of segments. There are five segments indicated by neurons 33, 9, 47, 11 and 34 along the central beam profile. The valley area surveyed using beam 10 (port side) also indicates five segments shown by neurons 33, 9, 47, 11 and 34. The area surveyed with beam 50 (starboard side) showed three segments indicated by neurons 33, 9 and 35. An overall analysis showed that those areas which had the highest width of detrended bathymetric PDF had more number of segments, and thus proves a direct correlation between the PDF of the detrended (stationary) depth and neural network based SOM architecture employed for online (nonstationary) depth for segmentation purpose, ultimately allowing the conversion from nonstationary to stationary data for suitable classification.

## Classification of bathymetric profiles using MLP

After the segmentation was carried out using the SOM as explained previously, each segment of the bathymetric profile has to be classified based on its roughness. The MLP finds application here to achieve this classification by computing the spectral parameter $b$ corresponding to a particular segment. Many applications of MLP are being studied which are of geophysical and oceanic importance[12,13], due to the advantage of this network to adapt to real-time classification once training is performed. The basic architecture of the MLP and the manner in which it classifies a segmented profile using $b$ are explained in this section.

As we know, the correlation parameter, $b$ is a measure of the topographic roughness[2]. In order to estimate this $b$ parameter, feature extraction of a given dataset using MLP is an important process to be carried out. After considering and analysing a variety of features (variance, kurtosis and MAFD), the MAFD feature of the segmented data proved to be the most suitable candidate for classification. Therefore an algorithm was developed to extract this feature from each segmented bathymetric profile. Another parameter, known as amplitude parameter $S$, is being used by Malinverno[2] along with $b$ for classification. In order to compute $S$, the required MAD of the depth data is computed as

$$\text{MAD} = \text{median } |\Delta z(x) - \text{median } |\Delta z(x)||,$$

where $\Delta z(x)$ is the first difference of the depth series $z(x)$. From here, $S$ is computed using the following expression[2]

$$S = 1.4826 \times \text{MAD}.$$

For more details about the above expression on amplitude parameter ($S$), readers may refer Kleiner and Graedel[14].

### *Architecture of the MLP and classifications*

The MLP architecture designed consists of six networks, a main network and five subnetworks (Figure 5). These

networks were trained using simulated data for $b$ values, which have been incorporated in this study[2]. A white-noise series of the desired length $n$ (= 100) was randomly initialized. This series is then zero-padded to a length that is the 7th power of 2, i.e. 128, and its Fourier transform is obtained. Next, filtering is carried out by multiplying the real and imaginary parts at frequency ($f$) by $f^{-(b-2)/2}$, and taking the inverse Fourier transform. Thus a set of depth values for various $b$ values (ranging from 0.5 to 4.5) is simulated, with a frequency range of $10^{-2}$ to $10^{2}$ Hz. After inverse FFT, simulated time series (depth) data are obtained for a unique value of $b$.

Each simulated depth data for a given $b$ ranges are used for necessary training for coarse classification. In the main network, training for five main classes (ranges) is carried out, and each range is further sub-divided into more accurate ranges for finer classification (limits are given in Figure 5). When a feature of a segmented bathymetric profile is fed to the input layer of the main network, it triggers one of the five output neurons based on the value of the feature. The main network, i.e. the coarse tuning network is used to broadly classify the segment among five distinct classes of seafloor. When a bathymetric profile is thus classified, the output neuron of the subnetwork, i.e. the fine tuning network corresponding to this class is triggered to give a more accurate value of the feature.

In the process of designing a MLP network, certain important configurational parameters like the number of hidden layers and the number of nodes in each hidden layer must be fixed. We have designed MLP networks with only one hidden layer. The optimum number of hidden neurons is given by $(m \times n)^{1/2}$ (ref. 11), where $m$ is the number of output neurons and $n$ is the number of input neurons. After trial and error method on the number of nodes in the input and hidden layers, as a trade-off between efficiency of training time and sufficient separation of classes, 10 input neurons were used in our study.

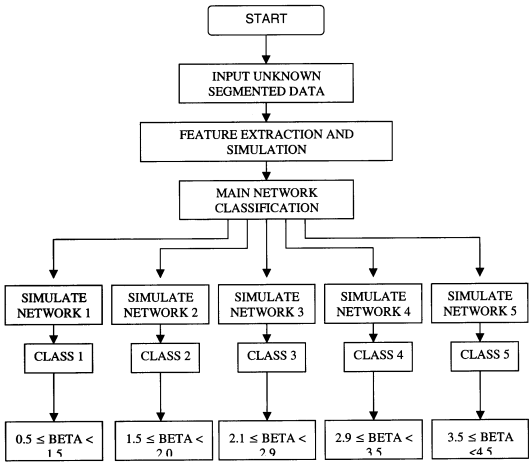


**Figure 5.** Flowchart for MLP adaptation for a two-stage classification of previously segmented data, into main classes and further sub-classes.

The output neuron number is fixed by the number of classes, i.e. 5 in this case for the main network, hence the number of hidden neurons is computed to be around 7. Similar configuration is used for the sub-networks, where input neurons are 10 and output neurons are 8, hence the hidden layer has 9 neurons. In order to train the network, we have used the Levenberg–Marquardt and Gradient-descent training algorithms[4,15,16] for the main and sub-networks respectively. The Levenberg–Marquardt algorithm is slower but gives an output matrix exactly equal to the target matrix. This principle works well for coarse classification. For the sub-networks, Gradient descent algorithm is very fast and gives an output matrix close to the target matrix. In our MLP-based classification network, the transfer functions for input, hidden and output layers were respectively fixed to purelin (the input function is directly reflected as the output without any further change), logsig (this function generates output between 0 and 1 as the neuron's net input goes from negative to positive infinity) and purelin for both main and sub-networks.

The network weights and biases between input, hidden and output neurons are initialized before the training phase to values between + 1 and – 1. For the output neurons, a target vector is assigned, where the neuron correctly represents the class. During every forward pass[17] through the network (from input to output layer), the mean squared error between the target vector and the computed network output vector is calculated for the input feature presented. Subsequent back-propagation updates the neuron weights to reduce this error. Training ceases when either the error goal of $10^{-30}$ is reached or the number of epochs (iterations) exceeds the predefined value of 1000. Once training is complete for five main classes, each class is further trained to classify sub-classes. Our observation of available $b$ values from various literature[18] indicates that more frequent $b$ values exist in the ranges ($1.5 < b < 3.5$). Based on these observations, we choose $b$ intervals of $\pm$ 0.1 at these ranges. Hence there are eight sub-classes for the three main classes corresponding to $b$ values of 1.6–2.2, 2.2–2.9 and 2.9–3.5, as these values of $b$ are more frequently occurring for depth data. However, the increment of $b$ values is maintained to $\pm$ 0.4 within the rarely occurring ranges of $0.5 > b < 1.5$ and $3.5 < b < 4.5$. For presently available $b$ values (Figure 5), only three sub-classes are sufficient. Now the main network is ready for testing with the MAFD feature extracted from a segmented depth data. The correctly representing output neuron is activated, which is further fine-tuned by testing this feature on the corresponding sub-network.

In Figure 6, the scatter plot of $b$ and $S$ parameters has been presented for each segmented depth data from three different regions. Moderately low values of the $b$ and $S$ for the Central Indian Basin plain areas are indicative of the sedimentary topography[2]. We observe different combinations of the amplitude parameter $S$ and $b$ for the CR

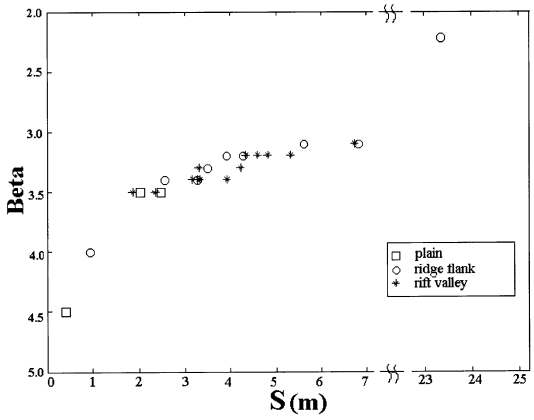


**Figure 6.** Scatter plot between the correlation parameter (*b*) and amplitude parameter (*S*) of the segmented seafloor topographic profiles for the plain area, Ridge flank area, and rift valley area are represented by the symbols square, circle and star respectively.

flank and rift valley areas. Interestingly, we also get comparable values of the estimated *S* and *b* values for given tectonic/volcanic areas from the Explorer ridge of the NE Pacific Ocean[2]. Employing ANN, areawise distinct trend is observed in estimated parameters. In general, both the estimated parameters (*S* and *b*) are moderate for the rift valley areas, though a significant variation is seen for the ridge flank areas.

## Conclusions

This article first applies neural network to segment and classify the multibeam bathymetric data. Two topologies of neural networks, namely SOM and MLP are used. The SOM is an unsupervised learning, which segments a particular dataset into statistically stationary segments. The technique of moving average was used for smoothing the data before applying SOM architecture for segmentation of the profile, its advantage being that it allows removing non-stationary part of the data. Using SOM, the multiple segments are obtained for CR flank and rift valley area, whereas for the CIOB, only one segment is seen. A distinct relationship is observed between the segmented depth data and depth PDF. The numbers of segmented profiles are the highest for the CR rift valleys, also indicated in the higher fluctuating depth PDF. Also, a number of segments for ridge flank and rift valleys are more for outer beams than the central beams, indicative of either noisy bottom-tracking of the outer beam of multibeam bathymetry system.

For classification, MLP needs extraction of suitable features from the data. The only successful feature used in this study is the MAFD of the depth profile, which is effective for the present case. The classification results are fairly meaningful in the sense that they are matching those of Malinverno[2]. The important aspect of the MLP architecture is that, it takes significantly more time to design the network and training of the network using simulated depth for known *b* values. Once network is designed and training is made, application (testing) of the real depth data takes negligible time. The MLP architecture could be improved by introducing more sub-networks having smaller intervals of *b* or simulate more *b* files at intervals smaller than 0.1. The employed technique has many applications, especially related to real time classification problems in marine geophysical applications including seafloor classification using sound signal backscatter and also in palaeomagnetism.

1. de Moustier, C., Beyond bathymetry: Mapping acoustic backscattering from the deep seafloor with Sea Beam. *J. Acoust. Soc. Am.*, 1986, **79**, 316–331.
2. Malinverno, A., Segmentation of topographic profiles of the seafloor based on a self-affine model. *IEEE JOE*, 1989, **14**, 348–358.
3. Chakraborty, B., Lorenco, E. and Kodagali, V. N., 38th Annual convention and meeting of Indian Geophysical Union – A special session on 'Ridge processes, plume ridge interaction and hydrothermal activity' Vishakhapatnam, December 2001, pp. 61–62.
4. Beale, R. and Jackson, T., *Neural Computing: An Introduction*, Institute of Physics Publishing, Bristol, 1997.
5. Mudholkar, A. V., Kodagali, V. N., Raju, K. A. K., Valsangakar, A. B., Ranade, G. and Ambre, N. V., Geomorphological and petrological observations along a segment of slow-spreading Carlsberg Ridge. *Curr Sci*., 2002, **82**, 982–989.
6. Cruise report of ORV *Sagar Kanya*, Cruise Report No. 114, 1996 (available at NIO library).
7. Bansal, A. R. and Dimri, V. P., Gravity evidence for mid-crustal domal structure below Delhi for belt and Bhilwara super group of western India. *Geophys. Res. Lett*., 1999, **26**, 2793–2795.
8. Stewart, W. K., Chu, D., Malik, S., Lerner, S. and Singh, H., Quantitative seafloor characterization using a bathymetric sidescan sonar. *IEEE JOE*, 1994, **19**, 599–609.
9. Kohonen, T., Self-organizing map. *Proc. IEEE*, 1990, **78**, 1464–1480.
10. Kohonen, T., *Self-Organizing Map*, Springer, Berlin, 2001.
11. Chakraborty, B., Kaustubha, R., Hegde, A. and Pereira, A., Acoustic seafloor sediment characterization using self-organizing maps. *IEEE TGARS*, 2001, **39**, 2722–2725.
12. Chakraborty, B., A neural network-based seafloor classification using acoustic backscatter. *Lecture Notes in Computer Science/ Lecture Notes in Artificial Intelligence* (eds Pal, N. R. and Sugeno, M.), (LNAI 2275), Springer, Berlin, 2002, pp. 245–250.
13. Michalopoulo, Z. H., Alexandrou, D. and de Moustier, C., Application of neural and statistical classifiers to the problem of seafloor characterization. *IEEE JOE*, 1995, **20**, 190–197.
14. Kleiner, B. and Graedel, T. E., Exploratory data analysis in the geophysical sciences. *Rev. Geophys*., 1980, **18**, 699–717.
15. Matlab Neural Network Toolbox, User's Guide, Version 6.0, Mathworks Inc, 2001.
16. Gill, P. R., Murry, W. and Wright, M. H., *Practical Optimization*, Academic Press, London, 1988.
17. Haykin, S., *Neural Networks: A Comprehensive Foundation*, Prentice Hall, New Jersey, 1994.
18. Berkson, J. M. and Matthews, J. E., Statistical characterization of seafloor roughness. *IEEE JOE*, 1984, **9**, 49–52.

ACKNOWLEDGEMENTS. We thank Dr Ehrlich Desa, Director, NIO, Goa, for his encouragement. We also thank Dr P. Blondel, University of Bath for his constructive review to improve this paper, and Dr Ranadip Banerjee for his help. We also acknowledge support from the Department of the Ocean Development, Government of India, New Delhi and in particular to Dr P. C. Pandey, Director, and Dr M. Sudhakar, Scientist, NCAOR, Vasco-da-Gama, for use of Ocean Research Vessel (ORV) *Sagar Kanya* to acquire multibeam-Hydrosweep bathymetric data. NIO's contribution # 3833.