Journal of Intelligent Manufacturing (1993) **4**, 79–94

Process monitoring using auto-associative, feed-forward artificial neural networks

P. J. C. SKITT*, M. A. JAVED†, S. A. SANDERS and A. M. HIGGINSON

School of Industrial Automation, Faculty of Engineering and Computer Technology, University of Central England, Perry Barr, Birmingham, B42 2SU, UK
†*Engineering Division, Southampton Institute of Higher Education, Southampton, UK*

The potential of using artificially simulated neural networks as intelligent, adaptive process-monitoring devices is discussed. The investigation is considered as a method for automatic, intelligent exception reporting for quality control applications. The technique is also compared with the conventional statistical approaches of principal component analysis and Kohonen's feature map. The applications of the technique in aerospace and manufacturing environments are presented and a possible extension of the method to incorporate a diagnostic function is discussed.

Keywords: Neural networks, condition monitoring, resistance welding, acoustic emission

1. Introduction

In this report two distinct applications of neural networks are described which demonstrate the benefits of using these networks as intelligent monitoring systems for industrial applications. The applications described are firstly, incipient fault detection in aircraft engines[1] and gearboxes and secondly, monitoring the quality of welded joints in an automated resistance spot welding environment. The aim is to devise an intelligent system which can learn the individual characteristics of its own application environment and operational routines, without detailed knowledge of the likely faults and their associated symptoms. It is also desirable that the monitor be capable of self-adapting to the long-term changes in its application environment, due to ageing and maintenance.

A neural networks-based approach seems appropriate for devising these monitoring systems, primarily because of their intrinsic capabilities to self-organize, and extract characteristics or features from the available data, particularly in noisy environments. The auto-associative networks are used because supervised learning is only feasible if an adequate number of examples spanning the entire domain of both satisfactory and unsatisfactory operating conditions are available. Data representing satisfactory process operations can readily be captured; however, data enveloping all possible unsatisfactory behaviours are difficult to generate in an experimental programme. It is also uncertain whether examples of every possible fault can be simulated to a reasonable level of accuracy, either experimentally or numerically. It is therefore preferred to avoid a conventional supervised learning strategy. The use of auto-associative neural networks appears to be a feasible approach.

The auto-associative network is employed, initially to extract important features from the input pattern, without supervision, which are then presented to a more intelligent system, operating at a higher level. At this second level, which may well be a more advanced neural net, the experience gained from processing of the satisfactory behaviour forms the basis for highlighting any subsequent deviations from it and its possible causes. The application can therefore be considered as a form of intelligent exception reporting providing an initial indication of an unsatisfactory behaviour pattern.

The system as reported here only provides an intelligent condition-monitoring device without any fault-identification capability. However, the speed at which trained neural networks operate certainly provides an

*Seconded from Cheltenham and Gloucester College of Higher Education as a Royal Society/SERC Research Fellow at Smith's Industries Aerospace and Defence Systems, Bishop's Cleeve, Cheltenham, UK.

[1]Smiths Industries patent applied for.

opportunity for some diagnostic capability to be incorporated in the system. The work concerning identification of different faults and subsequent initiation of appropriate remedial actions has not yet reached a stage where results can be reported. It is nevertheless felt that the technique has a broad potential for applications in monitoring the performance of intricate processes without involving costly human evaluation, especially where the nature of the evaluation testing is likely to be complex or even destructive.

The main reason for using a neural network is to take advantage of its ability to self-organize the data providing an alternative, objective analysis with no pre-conceived expectation of the form of potential faults. An important secondary reason is that the adaptive nature of the learning process involved in neural networks allows a monitor to adapt to the individual characteristics of its host machinery and its operating environment.

2. Characteristics of data

Data for the first application were obtained by digitizing the broad-band recordings of the acoustic emission of a Rolls-Royce GEM engine (Hewitt *et al.*, 1989; Witcomb *et al.*, 1989). The acoustic information in the form of spectral components was obtained by performing a Fast Fourier Transform (FFT) using a Hanning window on raw microphone signals. The FFT favours the partitioning of the signal into 2^n frequency bins and it was necessary to balance the high computational load caused by a large value of n against the loss of details in the information for a small value of n. Sixty-four bins gave an acceptable compromise, where the calculations of both the FFT and neural network could be performed in real time while retaining the major features in the spectral frames. Nine adjacent frames were averaged to produce spectral frames at the rate of five frames per second. The following are typical characteristics of the data.

(1) There is potentially an infinite number of input patterns, ruling out any prospect of target learning which would, in any case, defeat the self-adaptive aim of the system. There is also a high degree of correlation between the inputs, especially those closer together in the time domain;

(2) Some inputs represent short-lived, transient engine conditions, such as those experienced during the acceleration phase. Others such as those encountered during cruise flight are of longer duration. The inputs, therefore, cannot be presented to the network in any organized sequence or at any controlled frequency. The system must be capable of distinguishing between these states;

(3) The noise and natural variability change in a complex manner across the input set (Hewitt *et al.*, 1989; Witcomb *et al.*, 1989). Such complexity has a profound effect on the learning strategy.

3. Artificial neural networks

A number of methods for simulating neural networks are available (Lippman, 1987), and three of these models, namely Kohonen (1989), Carpenter and Grossberg (1987) and Rumelhart *et al.* (1987), were tried. The networks suggested by Kohonen (1989) and Carpenter and Grossberg (1987) showed some useful learning behaviours for both applications. The results, however, were found to be less satisfactory for the first application, primarily because these models led the networks towards a form of vector quantization. Also, data involved in the applications placed special requirements on the networks.

Most satisfactory results were, however, obtained by employing a feed-forward layered network with error back-propagation methodology derived by Rumelhart *et al.* (1987), often referred to as multi-layered perceptron (MLP). A brief summary of this methodology is outlined in Appendix A. After some experimentation, a symmetrical five-layer perceptron, shown schematically in Fig. 1, was used. The number of cells in each layer was chosen empirically.

The network employed the error back-propagation learning strategy to reproduce its input on the output layer. The outputs from the cells in the middle layer were used as an encoded form of the input. The number of cells in the middle layer controlled the dimensionality of this encodement. The reduction in dimensionality also had the effect of removing noise from the input signal in a manner very similar to the method of principal components and resulted in a characteristic 'acoustic signature' as discussed in the next section. This arrangement circumvented the necessity of 'supervised learning', bringing the benefits of automation and removing the need for prior knowledge of the engine's acoustic signature. Additionally a controllable learning threshold was used to inhibit the network from re-adaptation on already learned patterns, such as those representing less frequent engine conditions.

It is perceived that without such intervention in the learning process, the extremes of the persistent input subset, corresponding to the engine cruise state conditions, would map onto the extremes of the internal representations at the expense of the transient subset features. Kuczewski *et al.* (1987) have suggested a similar technique which they call a 'dead zone'. This incorporates a chosen discrepancy between the inputs and the reconstructed outputs and new learning is only initiated

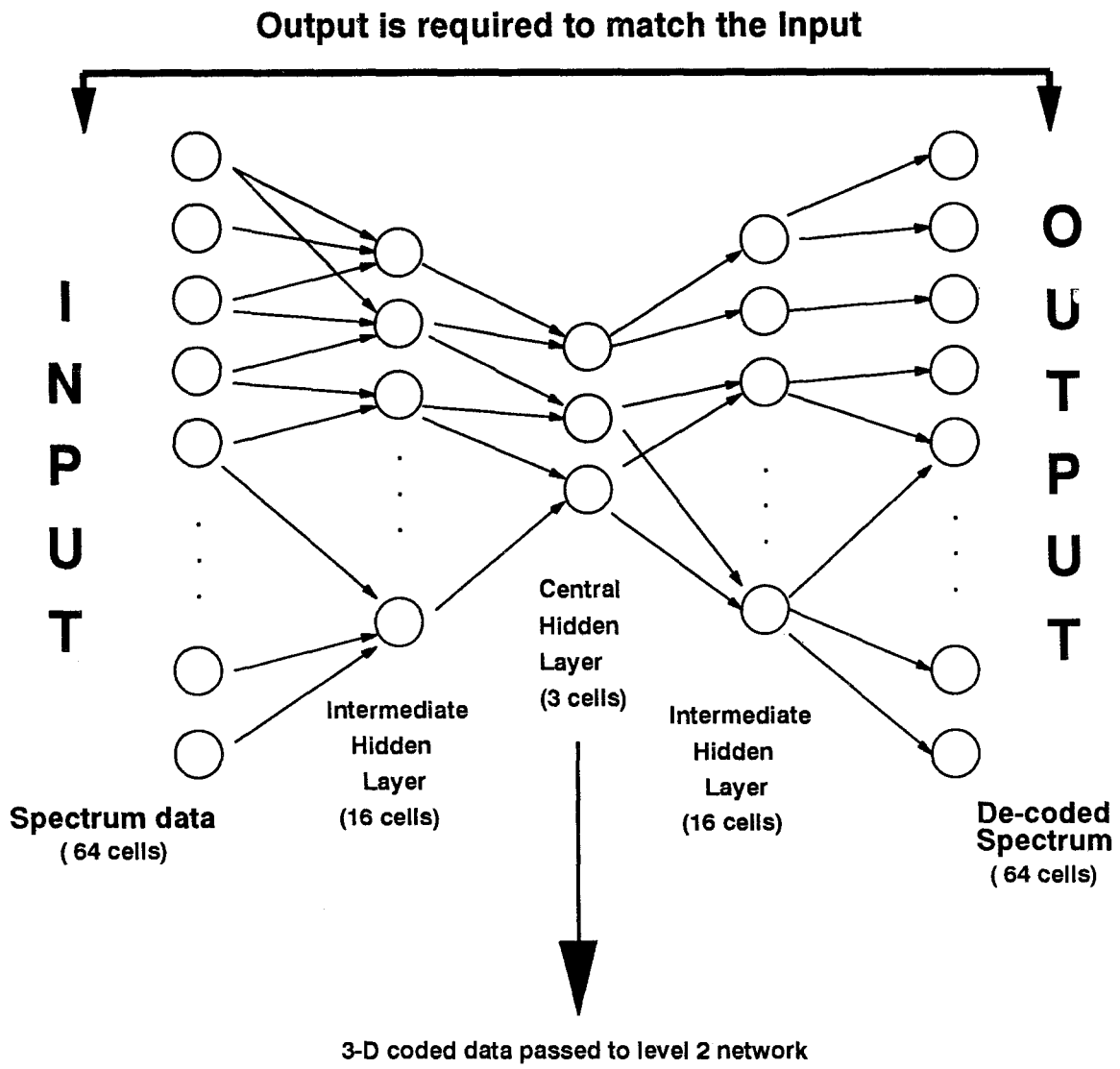

Fig. 1. The 'hour glass' arrangement for the auto-associative multi-layer perceptron.

for discrepancies outside this dead zone. A multiplicative, rather than a constant, tolerance in discrepancy was employed in the present application, the dead zone being some fixed proportion of the input values. With up to a 10% threshold value, satisfactory learning performance was achieved.

For the second application a Kohonen (1989) network seems appropriate as the patterns to be learned and subsequently identified at the second level by the system can be considered as discrete, but related. In a Kohonen (1989) network, as outlined in Appendix B, the 'neurons' or 'nodes' are arranged in a two-dimensional grid and when activated they influence each other by stimulating close neighbours. This stimulation affects the learning process. Consequently the neuron synapses adapt to a closer match to the stimulating pattern so that neighbouring cells learn to recognize input patterns which are 'near' to one another. The result is that the network produces a reduced dimensional mapping of the input space. This mapping can preserve some measure of distance observed in the input pattern space, whilst each cell provides an exemplar pattern acting as a vector quantizer.

This model provided results which were similar to the five-layer MLP network described above, with the advantage of a faster learning rate. The technique has its limitations, however, as the number of desired clusters needs to be specified prior to its application. The system, as mentioned earlier, is intended to operate in an unknown pattern space, and such a requirement may restrict its scope in a broader sense. There is also no mechanism built into the model which can curtail its tendency to re-adapt on less frequent patterns, such as the patterns representing the first few welds produced by

a new welding electrode. Kohonen-type networks are further explored for both of the problems described, but the work is not yet at a stage where conclusions may be drawn.

4. Machine condition monitoring using neural networks

The application of an auto-associative neural network to condition monitoring approaches the problem from a new standpoint. Conventional condition monitoring assumes some knowledge of the fault conditions and associated symptoms. The purpose of a neural network-based monitor, as outlined earlier, will be to extract information from the experience gained by monitoring the satisfactory operating conditions and then to signal any future deviations from these satisfactory states. Helicopter gearboxes and the health of jet engines (Hewitt *et al.*, 1989; Witcomb *et al.*, 1989) have been used as the basis for research but the approach is applicable to a wide range of problems. This implies that the monitoring system could be applied to many types of system, with the restriction that the network must be able to learn from experience spanning the entire range of satisfactory states.

A block diagram of the monitor is given in Fig. 2, which shows two levels of the intended intelligent system. In operation the first-level monitor would perform the following functions. The network is trained by showing it the input patterns for the satisfactory operating behaviour only and the learning parameters are frozen. With the connection weights fixed, inputs to the MLP would be matched to the corresponding output, which can only occur if the features were encountered during training. Then, if an unsatisfactory match is obtained, the reduced dimensional outputs from the central hidden layers are passed to the second-level monitor for analysis in the context of other parameters and dynamic effects. This second-level analysis is performed, at present, using conventional statistical approaches. The adaptive nature of the neural networks would be exploited by retraining the networks at intervals as the individual machines to which they are connected are modified or age. A detailed description of the first level of operation follows.

The first level uses acoustic information in the form of spectral components. The neural network encoder, a symmetrical five-layer MLP operating in an auto-associative mode, reduces the dimensionality of the data at the central hidden layer, thus forming an internal representation of the spectral data which constitutes the acoustic signature.

The second half of this hour-glass arrangement effectively decodes the signature back to the archetype spectral representation. The acoustic inputs to the monitor are processed to reduce the volume of the data and to monitor the mismatch between the input–output pattern simultaneously. The mismatch is construed as an unsatisfactory operating behaviour. In the case of a mismatch,

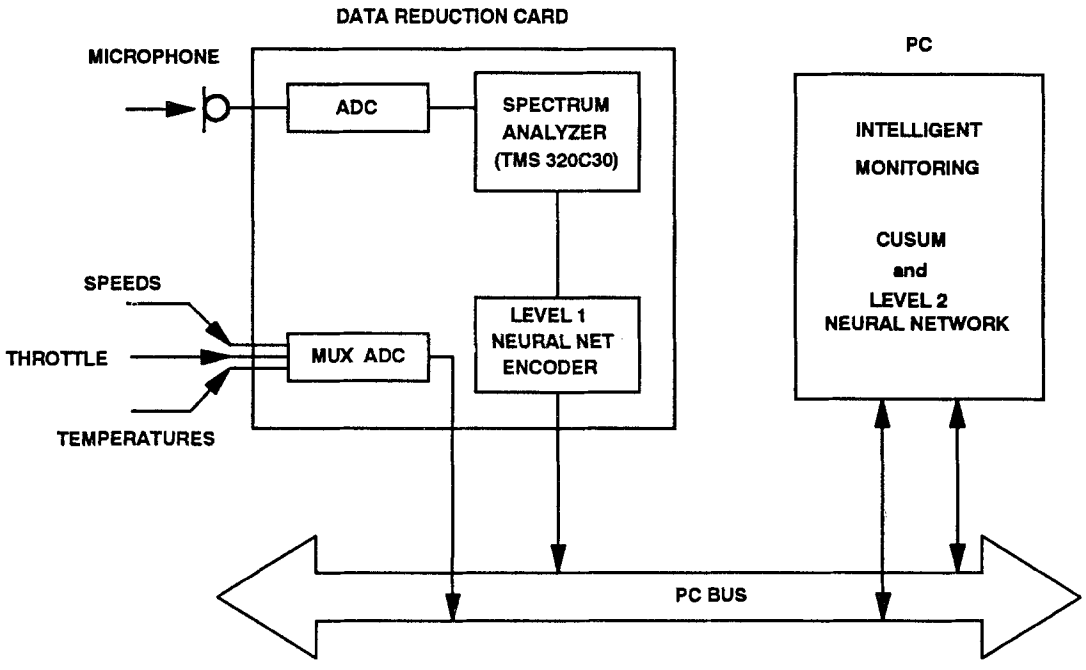

Fig. 2. Block diagram for the engine-monitoring system, showing the two levels of neural network processing.

the resulting acoustic signature is presented to the second level of the system. At the second level other inputs, representing both control and response parameters, are then concocted together to establish the auto-correlation with the acoustic signature.

The reduction in dimensionality is achieved by finding a set of connection weights which characterize features of the inputs, whilst evading modelling of the noise present on the signal. It is difficult to decipher this encodement in relation to the physics of the problem, as it is not clear how the non-linear combination of weights across the hidden layers achieves this characterization and what it physically symbolizes. The numerical validity of this encodement can, however, be established.

An example showing the performance of the encodement is presented in Fig. 3 which uses intensity (light to dark) to indicate increasing acoustic energy level so that the moving peaks do not obscure one another. Acoustic data in the form of spectral frames, as shown in Fig. 3a, are transformed by the MLP into the three values depicted in Fig. 3b. The output spectra reconstructed by the MLP from these three values is shown in Fig. 3c. The reconstructed patterns are visually indistinguishable from the input patterns, showing that the main features have been captured and used in reconstruction.

The encodement of the central layer shows that spectra that occur close in time are encoded into a close representation, and this three-dimensional representation can be thought of as a path in time or a trajectory. An alternative demonstration of these values from the central hidden layer is provided in Fig. 4, where a projection of the path defined by the three values is shown in a unit cube defined by the possible values cells may take in the central hidden layer. This illustrates the way in which the continuous, high-dimensional input is mapped onto a continuous low-dimensional encodement.

There are two distinguishing features of this application. First, the states to be monitored are not discrete, that is to say the acceptable states are connected in a continuous fashion and occupy a restricted volume in their high-dimensional vector space. Figure 3a illustrates this, where the acoustic emission at 64 frequencies constitutes the measurement space and the transition between the end points forms a well-defined path. The difference between neighbouring states depends on the speed of transition and sampling rate, and in the limit would become continuous. In this context, vector quantization may give poor results as a discrete set of reference vectors will give inaccurate matches at boundaries, unless the number of reference vectors is large, in which case the calculation load may then become excessive.

The second feature is the very restricted availability of data for fault condition – not many aircraft engines are tested to destruction – hence the need to establish a monitoring method based on the experience of acceptable performance. The function of the monitor is to signal potential problems early enough to allow a sophisticated diagnostic technique to be applied, and consequently minimize the down-time. It is important that the potential problems do not have to be pre-specified, although as a result some new conditions will be signalled which are not necessarily faults.

The ability to detect incipient faults has been tested under laboratory conditions. The adaptive, localized nature of the monitor allows it to concentrate on the characteristics of an individual machine eliminating inter-machine effects. The simple fault-detection capability of this approach has been tested by exposing the first-level monitor to acoustic data in which the operation of the engine has been modified, but which has not been included in the training set. Changes in the reheat nozzle angles, and opening of the bleed valves, can be detected instantaneously using conventional statistical approaches at the second level.

As an example, the detection of a small disturbance to the acoustic input is described and illustrated in Fig. 5. The original GEM data were used to generate a larger test set, by sampling as a random walk and overlaying uniform noise in the range $\pm 10\%$. A second test set was then generated by adding a disturbance of 0.1% of the total input signal to the 40th frequency bin. The MLP was exposed to both test sets and a cumulative summation of errors (CUSUM) (Page, 1954) used to measure the input–output match. The CUSUM is a simple measure of fit, calculated by accumulating, through time, the input–output difference across all frequencies. Under the assumption that the error term is distributed with zero mean, i.e. the acoustic signature is centred on the actual signature, the expectation of the CUSUM is zero, and hence consistent deviation from zero indicates that some change has occurred. The simplicity of the CUSUM approach avoids the need to estimate the distribution and associated parameters of the errors. In the example of Fig. 5c the CUSUM is shown for both test sets after 10 minutes' simulation. The broken line indicates that the CUSUM for the test set with the disturbance is consistently diverging from zero.

4.1. *Comparison with conventional methods*

The statistical method of principal component analysis (PCA) can also be used to reduce dimensionality in problems containing redundant information, and may provide a performance benchmark. Baldi and Hornik (1989) have shown that a linear network with no thresholding performs a task similar to PCA. The logistic activation function and thresholds of the MLP extend this theme so that the auto-association may be considered similar to a non-linear PCA, with a correspond-

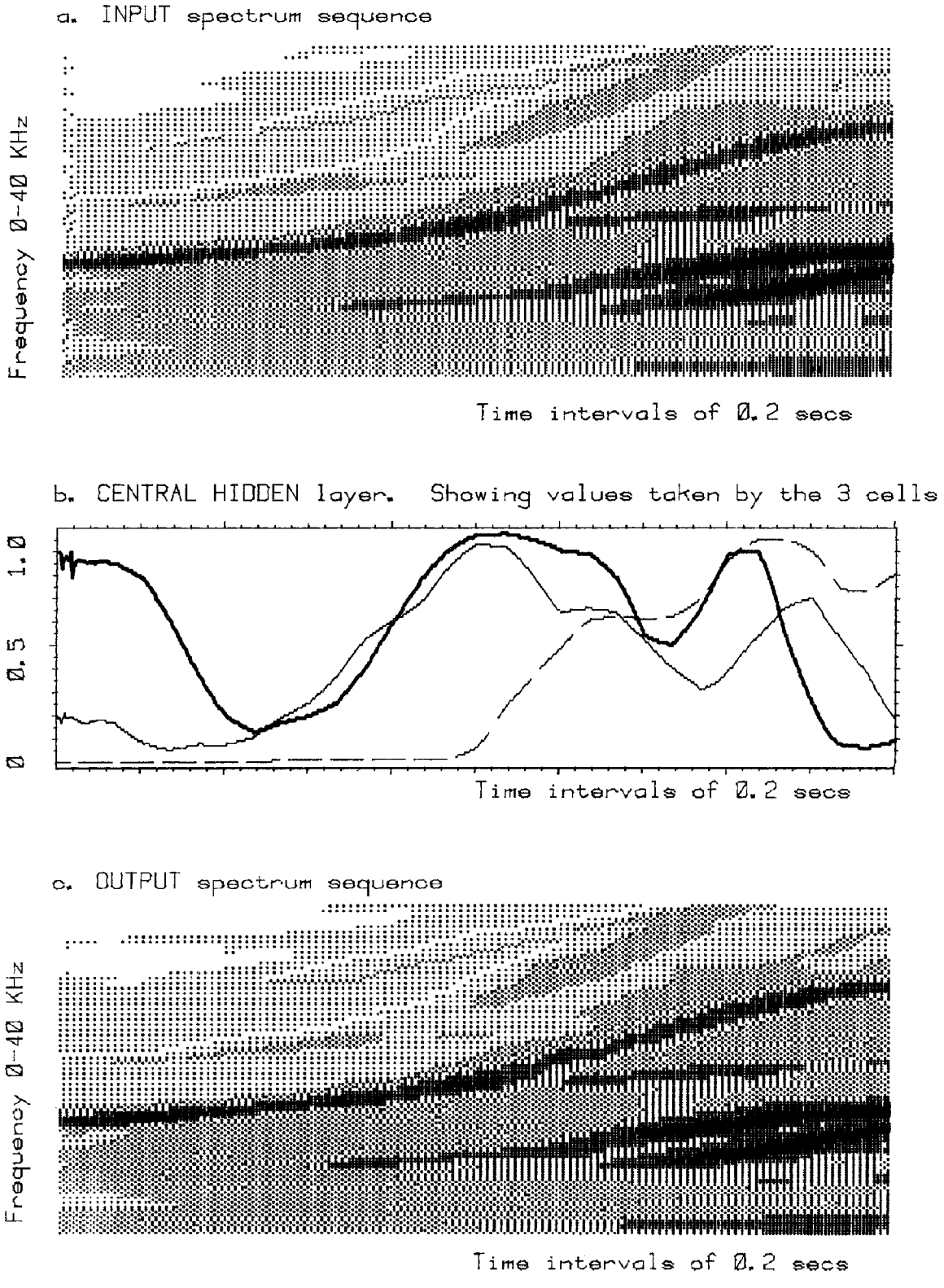

Fig. 3. A sequence representing the acceleration transient of an aircraft engine. The actual input spectra (a) are shown as an intensity (light–dark) corresponding to low–high level; 64 frequencies are shown coded into three values (b) at the central hidden layer and decoded at the output (c).

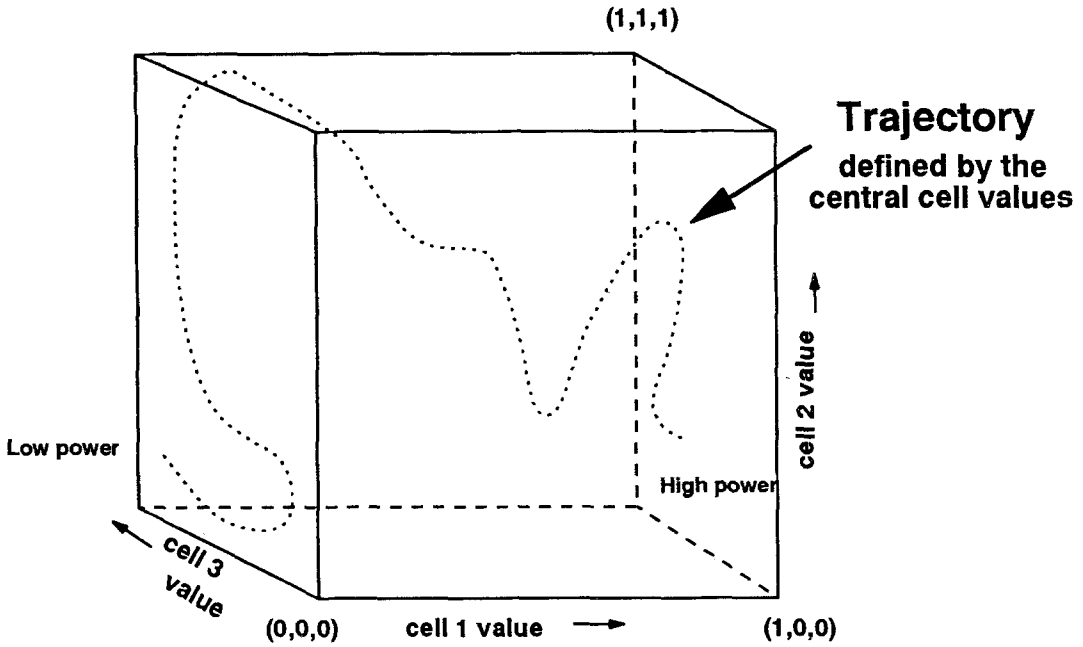

Fig. 4. The three values at the central hidden layer, corresponding to Fig. 3b showing the contiguous path. The input spectra are encoded onto this path which represents the 'acoustic signature'.

ing inversion. Deleeuw *et al.* (1976) have described an alternating least-squares algorithm which obtains combinations of non-linear transforms to perform a non-linear PCA. This algorithm, implemented on the Statistical Analysis Systems (SAS) computer package, was used as a comparison for the performance of the MLP. Table 1 shows the results for a number of transforms, with the MLP outperforming the non-linear PCA by a small margin. Linear PCA fall far short of this performance with the first six components accounting for only 94% of the variance.

The results above show that it is possible to train a neural network self-adaptively to characterize the normal operating behaviour of its host machine, using the available information which may contain some redundant information. Small deviations from normal operation can then be detected. The function performed by the neural network, in this case, is to derive a continuous mapping from the input space onto a lower-dimensional space. The methodology differentiates the reported approach from more usual application of neural networks to classification problems, where decision surfaces are derived which partition the input space. The adaptive nature of the neural network, and the simple auto-associative training method, allows highly specific, automated development of monitoring systems.

It is believed that this complementary approach of anomaly detection will allow a wide range of manufacturing and commercial systems to be monitored which previously were considered too complex or data intensive. For example, quality control may be achieved by identifying small changes quickly so that early corrective action may be taken.

Table 1. Comparison of the performance of the MLP encoder with linear and non-linear PCA

Method of analysis	*Number of components*	*% variance explained*
Linear PCA	3	82.5
	6	93.7
Non-linear PCA – cubic spline	3	97.5
– monotone cubic spline	3	96.5
Auto-associative MLP	3	98.9

5. Non-destructive testing of weld strength

The second application describes the use of neural networks in an automated production environment to monitor the quality of joints achieved by resistance spot welding. Resistance spot welding has long proved adequate for the bulk of mass production applications such as car manufacturing and aerospace industries. The spot welding process is a complex interaction of electrical, mechanical and metallurgical phenomena. There are many variable factors involved in the process: welding current, voltage, shape of the electrode tip, force, time,

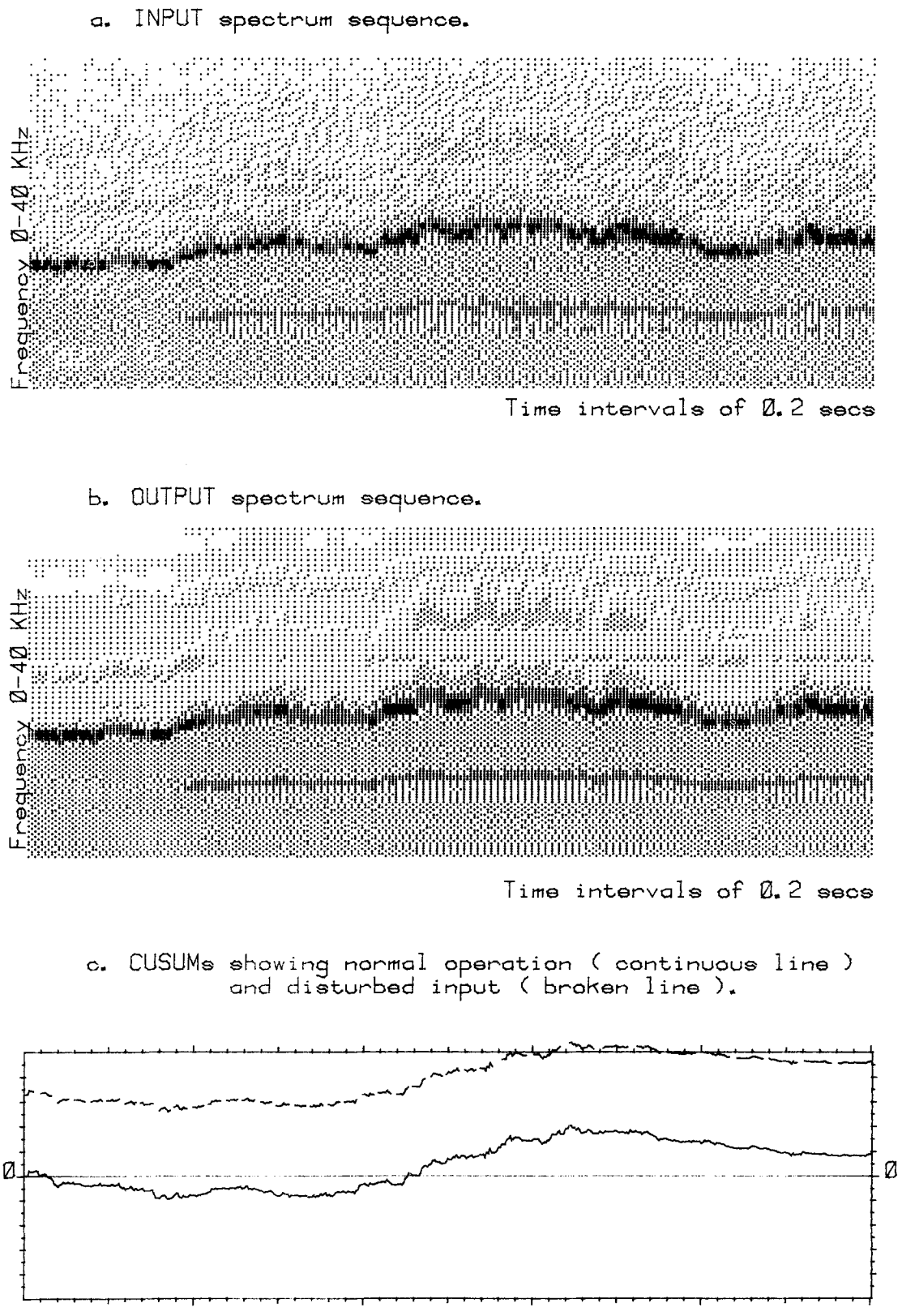

Fig. 5. Example of CUSUM detection of a small (0.1%) disturbance after 10 minutes' simulation. The cumulative differences between input spectra (a) and output (b) are plotted (c) where the disturbance has caused its CUSUM (broken line) to diverge from zero.

fit-up of parts, and so on. Some of these factors can be measured and incorporated into a weld qualification procedure. The principal weld qualification procedure customarily used in industry has been based on the estimations from a destructive peel or chisel test. The weld is destroyed to evaluate the nugget size, which is related to the strength of the welded joint. A minimum nugget size is then specified in order to meet the required minimum strength. More recently dynamic resistivity profiles (Holden and Sanders, 1989) of the components being welded have been studied to formulate the basis of weld quality control procedures.

The increasing use of galvanized steel to enhance the corrosion protection of the exposed parts in the manufactured products has given rise to the need for a more sophisticated weld qualification procedure. The traditional technique based on the dynamic resistivity profiles has its limitations for coated materials, primarily because the wear of the welding electrode and electrode tip corrosion have a profound effect on these profiles. The technique can therefore cause misleading assessments. A non-destructive procedure capable of qualifying the welded joints is a particularly difficult but important task for the automated manufacturing industries. This automated resistance welding qualification procedure is the subject of the present discussion.

It has been noted that the zinc coating on steel alters the thermal and electrical properties (McGregor, 1983) of the electrode–sheet and sheet–sheet interfaces. Zinc being a better conductor of electricity than steel implies that in order to generate the same amount of heat at the faying surface, as would be required for uncoated steel, the welding current must be increased. Zinc is also a better conductor of heat, so more of the generated heat is conducted away from the area of weld formation. The consequence of these effects implies that coated steel requires a larger welding current and a smaller welding lobe, i.e. it has a reduced tolerance to variations in welding conditions. The requirement of a larger supply current needed to produce an acceptable nugget size helps intensify the brassing effect at the electrode tips as a consequence. All this contributes towards reduced electrode life (Tanaka *et al.*, 1985; Holden and Sanders, 1989), but more importantly it is difficult to quantify and monitor the quality of welds in a real-time environment with high confidence.

5.1. *Analysis of data*

The welding process as a whole has a number of measurable control parameters such as current, voltage, pressure and time. There are however a multitude of factors relating to physical and metallurgical properties of the material being welded, e.g. the fit-up of the parts, state of the weld tips, and so on, which can cause an unacceptable weld joint. The objective of this application is to identify acceptable welds using measurable data.

The raw data consisting of current, voltages (AC and DC components), time and weld tip age were measured for 0.8 mm hot-dipped galvanized mild steel (Holden and Sanders, 1989). The current was measured using a toroid on the transformer secondary and the voltage was measured using a sampling technique (Holden and Sanders, 1989). The tests consisted of performing 95 welds on one sheet of material. The welds were then broken and the nugget size recorded. It was hence established that for the given material a nugget size of 3.5 mm diameter was the minimum required size to achieve an adequate strength level. The measured data were processed using a MATLAB software package to remove bursts of periodic noise and to combine into an empirically determined time-dependent function called 'weld signature'. It is simply calculated by dividing the measured voltage by the supply current. Figure 6 shows some typical weld signatures in the life-time of a typical electrode.

5.2. *Results and discussions*

There is a similarity between this and the former application in that the fault conditions were excluded from the learning process: firstly, because there are fewer such samples available and the validity of simulated faults for the purpose of teaching the network is highly dubious and may not bear any resemblance to the real-time faults; secondly, because allowances for changing conditions such as weld tip age must be incorporated.

The differences between the two examples are

(1) The welding examples are not connected in a continuous fashion like the adjacent engine states in the former example;

(2) A diagnostic element was deemed necessary.

The last point is complicated by the fact that the conventional dynamic resistivity profile criterion cannot be employed beneficially to distinguish a faulty weld signature from a good weld signature. The objective of the investigation is then initially to identify weld signatures which do not fit into the satisfactory pattern and to identify the doubtful welds in real time. In the latter case further investigation may be appropriate or a pre-defined remedial action could be taken. The relationship between weld quality and weld signature is established by considering the measured nugget size in comparison with the average value of the weld signature as illustrated in Fig. 7. It is readily seen that the unsatisfactory welds fall into a separate region, but are only distinguishable if the nugget size is known. The correlation coefficient relating the average weld signature and nugget size is observed to

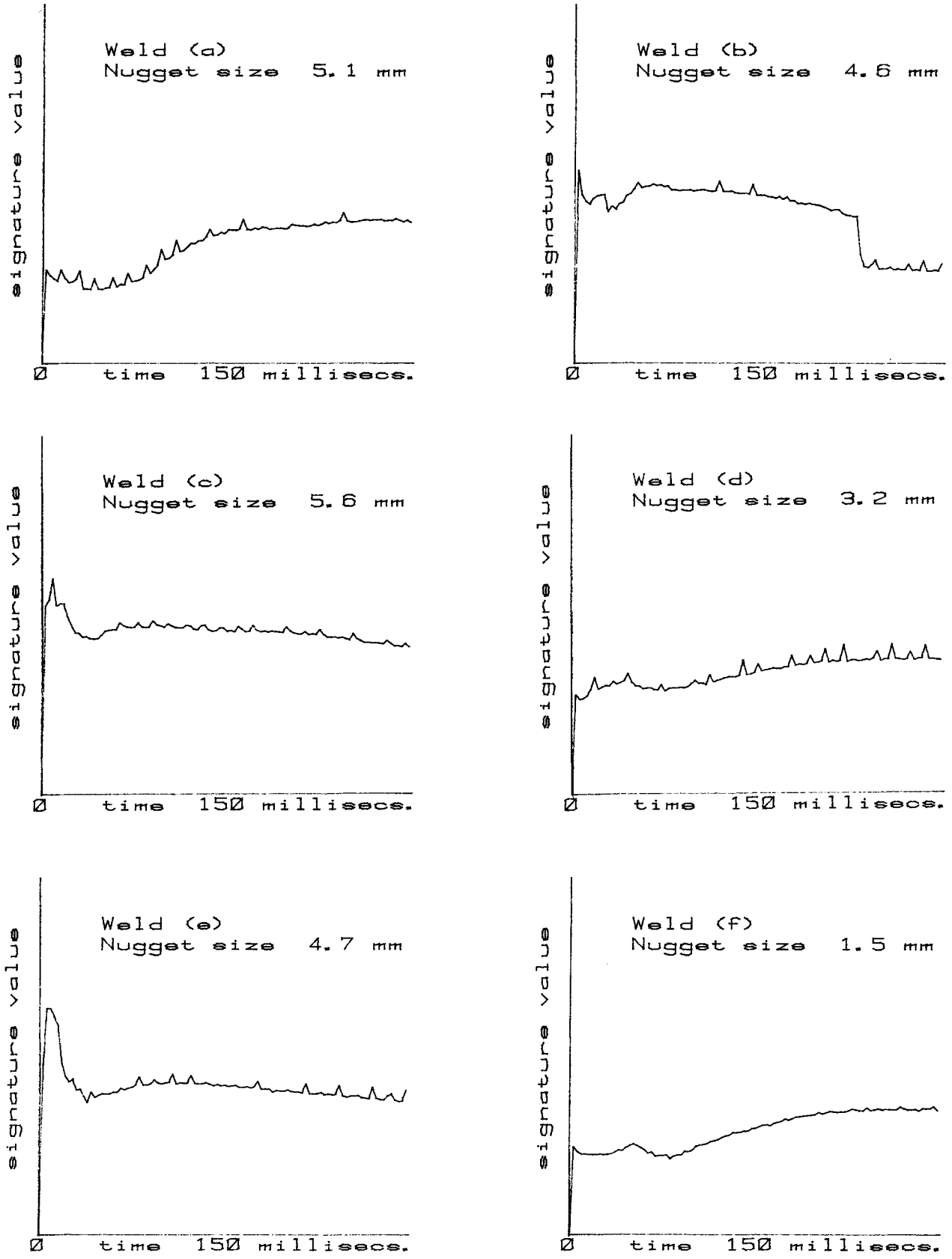

Fig. 6. Examples of weld signatures during the life of a pair of weld tips.

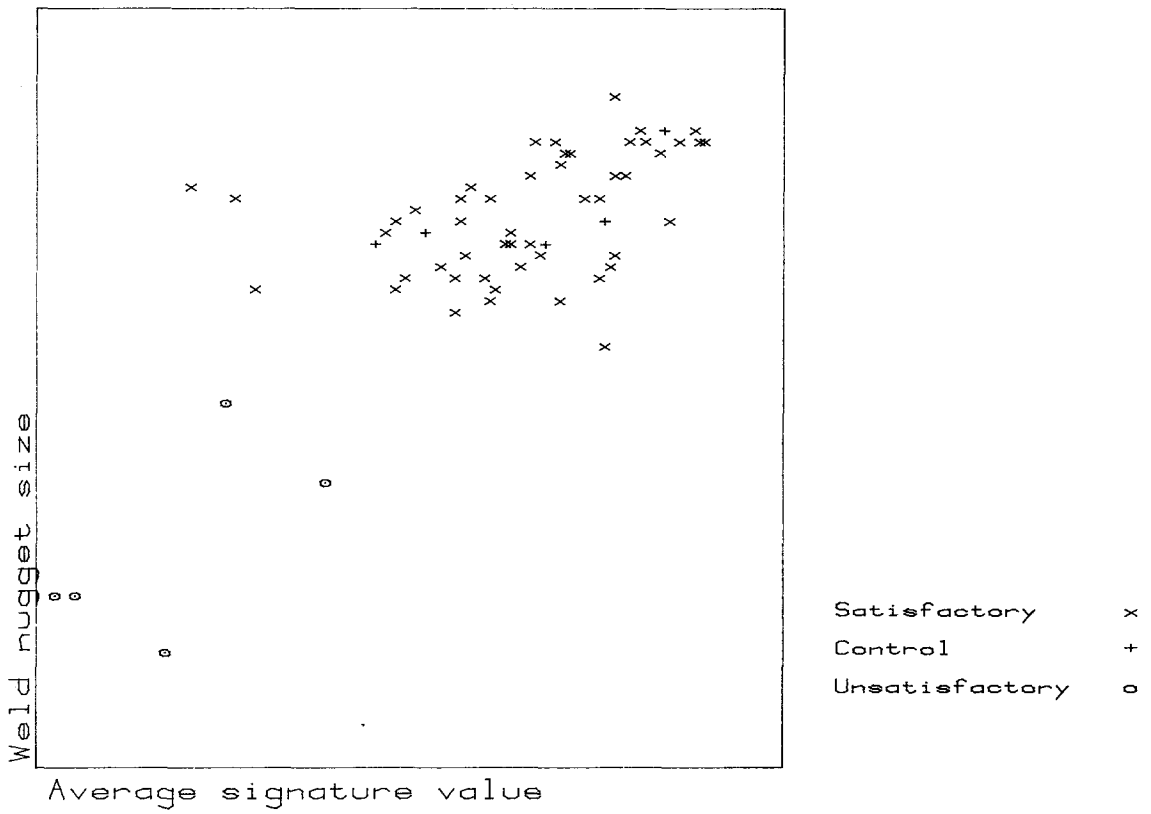

Fig. 7. The average signature values plotted against weld nugget size show that there is a relationship between them that may be useful for non-destructive testing, but that better discrimination is required.

be 0.73. There is also an overlap between the classes of satisfactory and unsatisfactory welds.

A three-layer MLP was taught to associate the weld nugget size with the weld signature and a correlation coefficient of 0.85 between target and predicted weld size was obtained. This gave an indication that there was additional information in the time domain that could help discriminate between the classes.

Supervised learning as such was avoided since it would have implied knowledge of all possible circumstances. An auto-associative feedforward five-layer MLP network, based on the previous model of engine monitor, was employed to avoid this problem and to pass the data through a two-dimensional central layer. As before the input–output mismatch was used as the measure of difference.

The network was trained on 50 satisfactory weld signatures, and 10 weld signatures (five satisfactory and five unsatisfactory) were used for testing purposes. The results are shown in Fig. 8. It is seen that the five faulty welds are distinguishable from the good welds while the five good welds occupy the same region as other good welds. Figure 8 shows that a mismatch between the input and output signals a signature outside the experience of the network.

Although the changes in the weld signature are gradual and predominantly reflect the steady wear and contamination of the electrode tips, yet the data may not necessarily be restricted to a continuous path through the input space in a manner similar to the jet engine example. There are other sources of discontinuities involved in the data such as the effects of a weld splash. Techniques based on a vector quantization-type methodology may therefore be considered more suitable for welding application as compared to the jet engine application.

The two-dimensional feature map, as suggested by Kohonen (1989), was employed to explore and compare its discrimination capabilities with the five-layer symmetrical MLP. The results for a 10×10 output map are displayed in Fig. 9 and are typical of the results obtained by employing different sizes of output maps, e.g. having 6×6, 8×8, 10×10, and 12×12 nodes in the output map, and giving a range of 36 to 144 reference vectors. The discrimination between satisfactory and unsatisfactory welds is quite reasonable, provided the nugget size is known. The prime objective of the exercise is to devise a monitor which can predict a good weld on the basis of the measured parameters, i.e. voltage and current.

The classification performed by the MLP has proved

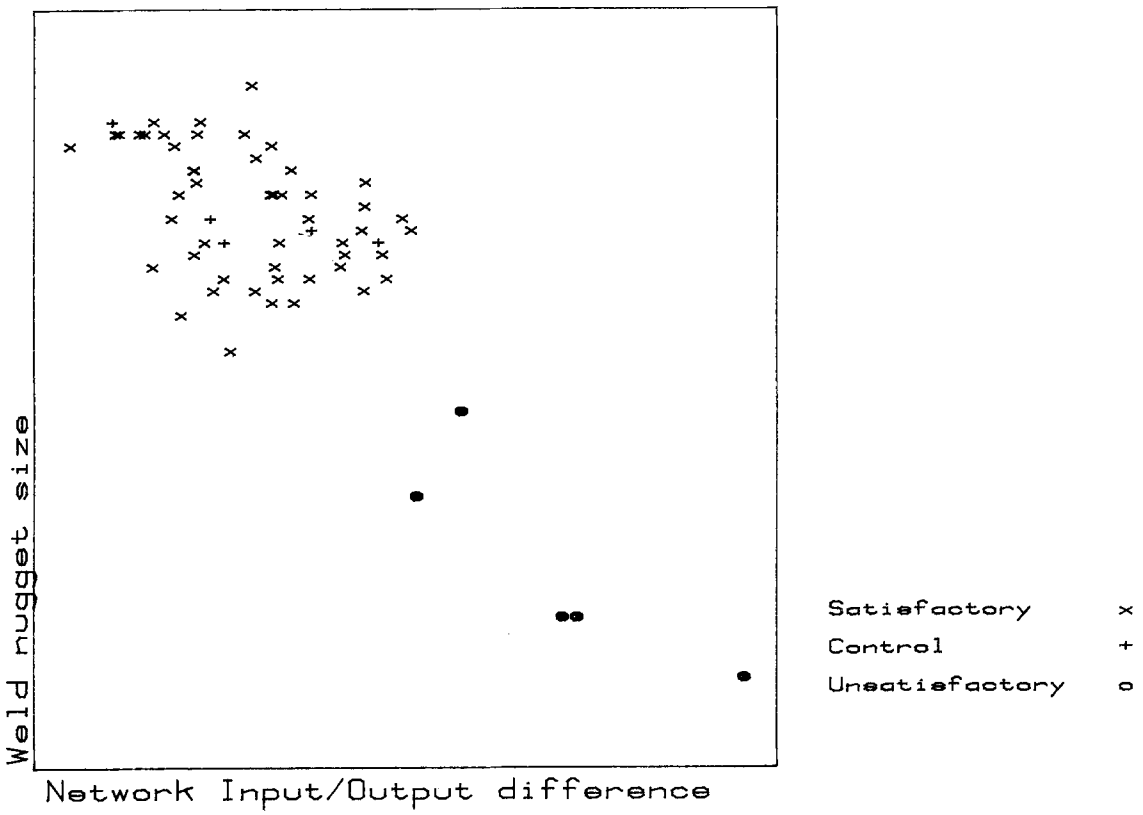

Fig. 8. The input/output difference of the weld signature shows that the MLP has extracted some time-dependent information to improve discrimination.

Fig. 9. The Kohonen network does not provide adequate discrimination.

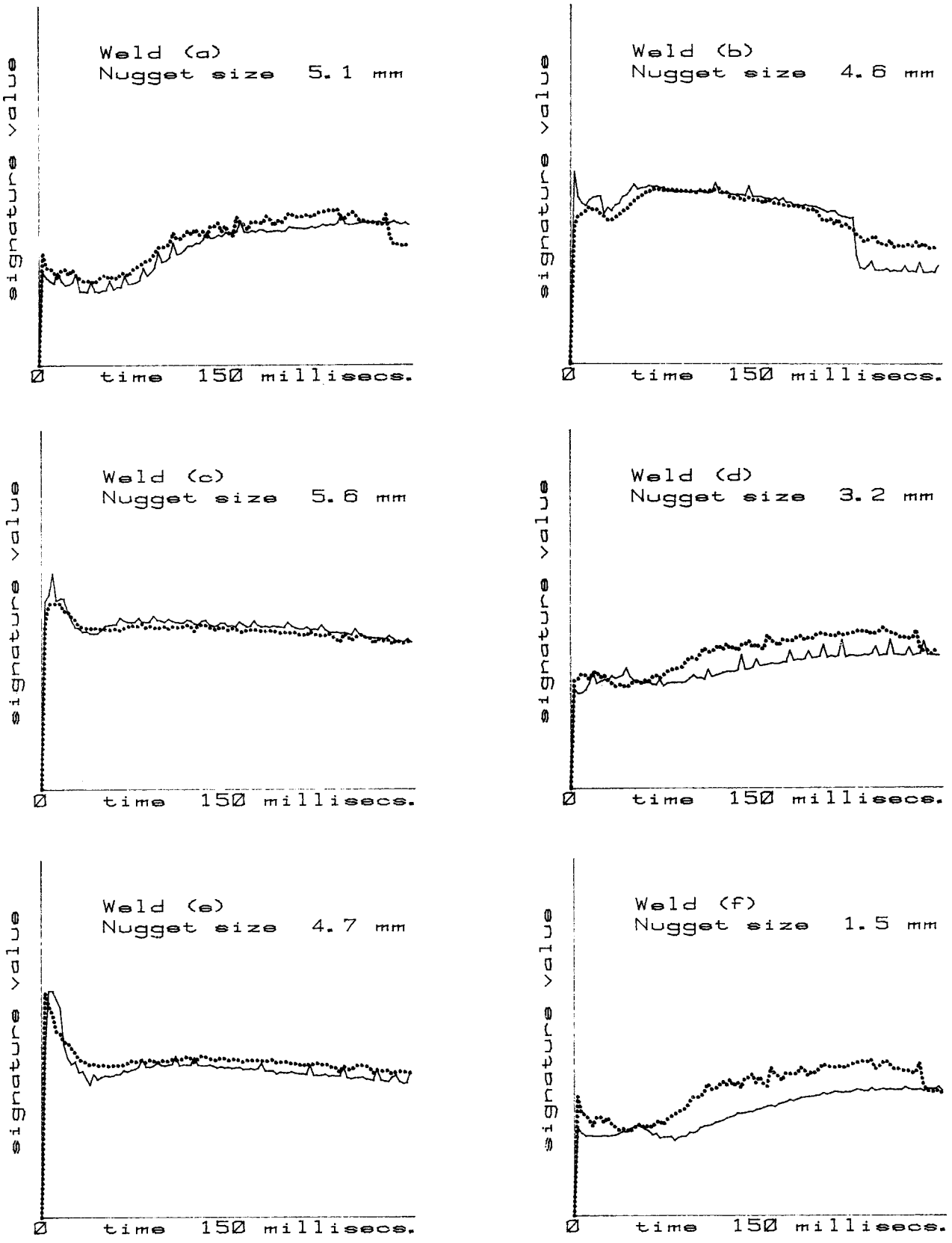

Fig. 10. The six example welds (solid line) from Fig. 6 are shown with the corresponding neural network output (broken line). The poor welds (d) and (e) are identifiable by the difference between the lines.

more satisfactory towards meeting this goal and it can be implemented readily in real-time environments. The MLP can be trained on the weld signatures corresponding to the entire domain of good welds and then employed to identify the signatures corresponding to the bad welds. A mismatch between the input–output values implies that the signature is outside the domain of its experience. An example is shown in Fig. 10, where weld signatures (d) and (f) are successfully identified as bad welds, based on the expected and actual values of the weld signature. This technique can easily incorporate a decision mechanism on the basis of area under the signature curves. If the area under the actual curve is less than that under the expected curve, taking into account the learning threshold parameter relating to the 'dead zone' as explained earlier, the weld is identified as a bad weld. An appropriate corrective measure such as a current-stepping mechanism can then be initiated. The application essentially amounts to non-destructive testing to evaluate the quality of welds in real time.

6. Conclusions

Neural networks have been shown to be a suitable method of process monitoring. Features of such applications in quality control of a manufacturing process (resistance welding) and a condition monitoring (aircraft engine) have been described. MLP-type neural networks operating in auto-associative mode can perform intelligent exception reporting equally well in situations involving discrete states as well as continuous states. More importantly the potentials of neural network-based monitoring schemes for processes with large intractable data have been demonstrated.

There are obvious similarities with the statistical method of principal component analysis and vector quantization. Neural networks however automatically perform feature extraction tasks in complex non-linear problems in an adaptive way. This gives them an advantage over purely analytical techniques provided the long learning times are not an obstacle. It is envisaged that the symptoms highlighted by the monitor will eventually be used by a higher level of intelligence to perform diagnostic functions, but this must be based on some experience of the representations derived by the network being linked to observable causes. A database could be built over an extended period of time for this purpose.

Non-linear PCA and vector quantization with interpolation offer analytic solutions which could achieve similar results. However, the neural network appears to perform equally well and have an advantage in flexibility of application.

Acknowledgement

The authors wish to express their gratitude to Dr R. C. Witcomb of Smiths Industries and Prof. F. Arther, Dean of the faculty, for their constant encouragement and support during the course of these studies. Thanks are also due to Mr N. Holden for his efforts and contribution in obtaining the experimental data.

Appendix A

Computing with the MLP

The multi-layer perceptron (MLP) comprises of a densely interconnected system of parallel distributed simple computational elements usually called nodes. These nodes are typically arranged as layers of processing elements. There are three types of layer: input layer, output layer, and hidden layers, as shown in Fig. A1. The hidden layers do not interact with the outside world directly and their role is to transform the input pattern and to associate it with the output pattern. The number of nodes in the input and output layers depends upon the dimensionality of the observation pattern and the exemplar pattern. The number of nodes in the hidden layer and the number of hidden layers depends upon the complexity of the task, but can be chosen arbitrarily and an appropriate topology derived empirically.

The principle of computation is similar to an analogue computer. The input pattern is presented to the nodes in the input layer and these nodes generate an output. The output is propagated forward, layer by layer, through the entire network. The output thus produced by the network is compared with the exemplar pattern. The resulting error is propagated backwards layer by layer and the weights adjusted in order to minimize the error.

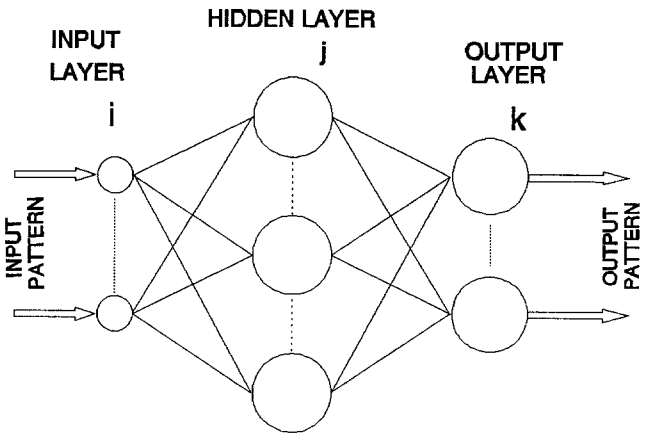

Fig. A1. Topology of a neural network.

This error minimization is achieved by the so-called error back-propagation strategy.

The nodes in each layer have the same basic structure. The computational functions performed by a typical node are depicted in Fig. A2. Each unit consists of three main components:

(1) The weights which adjust the strengths of the incoming signal from units in the previous layer.

$$O_{j,k-1} \times W_{i,j,k}$$

where $O_{j,k-1}$ is the output of the jth node in the $(k-1)$th layer; $W_{ij,k}$ is the weight between the jth node in the $(k-1)$th layer and the ith node in the kth layer.

(2) The summation block which adds all the incoming weighted signals to the threshold value, say β, of that node and forms the total input to that node.

$$NET_{i,k} = \sum_{j=0}^{n} (W_{i,j,k} \times O_{j,k-1}) + \beta_{i,k}$$

where n represents the number of nodes in the $(k-1)$th layer; $\beta_{i,k}$ represents the threshold of the ith node in the kth layer.

(3) The output activation function which produces the output signal of that node as a function of its total input. A typical form of activation function is the sigmoid.

$$O_{i,k} = \frac{1}{(1 + e^{-NET_{i,k}})}$$

The weight values and the threshold values are chosen arbitrarily at the start of the learning process. Networks using sigmoid transfer functions can be trained by learning algorithms. The generalized delta rule (Rumelhart *et al.*, 1987) has been used extensively for this purpose and shown to work efficiently for non-linear transformation between input and output patterns.

The training procedure consists of presenting the input pattern to the input layer and calculating the output pattern at the output nodes using the current set of learning parameters. This set of learning parameters of the system consists of the weights and the node threshold values. The output pattern is compared with the exemplar pattern and the error is calculated by computing the distance between the actual and desired output. The process is repeated for the entire set of input–output patterns and the total error, E, determined from:

$$E = \tfrac{1}{2} \sum_{c} \sum_{i} (d_i - o_{i,n})^2$$

where index c represents summation over all the input–output cases used for training; index i represents summation over all nodes in the output layer; d_i represents the desired output at the ith output node; $o_{i,n}$ represents the actual output at the ith output node; n represents the total number of nodes in the output layer.

The learning procedure aims to drive the error E to zero or close to zero by adjusting the learning parameters suitably. This essentially is a minimization problem which the generalized delta rule attempts to solve by gradient descent technique. The calculation of the error gradient with respect to learning parameters is performed by propagating the error backwards through the network and involves simple local computations at nodes in the same layer. Once the gradient is calculated the learning parameters are adjusted using the gradient descent method. The change in the learning parameters is made as follows

$$\Delta W_i = -\eta \frac{\partial E}{\partial W_i}$$

where η represents a positive step size, usually termed as learning rate; w_i is a learning parameter; E represents the total error.

Rumelhart *et al.* (1987) suggest that the expression can be modified as

$$(\Delta w_i)_{(m)} = -\eta \left(\frac{\partial E}{\partial w_i} \right)_{(m)} + \alpha (\Delta w_i)_{(m-1)}$$

where m is used to indicate the mth iteration; α is a small positive step, usually termed as momentum rate.

The momentum term is used to specify that the changes in weights at the mth step should be similar to the changes undertaken at the $(m-1)$th step. In this way some inertia is built into the procedure and the momentum in the rate of change is conserved to some degree.

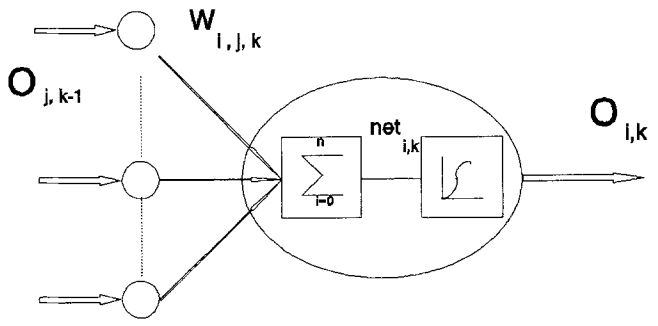

Fig. A2. The neural net cell.

Appendix B

Kohonen's networks

The Kohonen (1989) self-organizing feature map is based on an algorithm which performs a form of vector

quantization. The mapping is usually from a high-dimensional input space onto a two-dimensional output. In operation it is similar to *k*-means clustering algorithm, but produces an ordered mapping of the input patterns onto the two-dimensional array of output nodes. A schematic diagram of the network is shown in Fig. B1. The ordering occurs as a result of the output nodes having feedback connections between them. The feedback is invariant and follows the so-called 'Mexican hat' function, where close neighbours are stimulated and more distant nodes are inhibited.

Kohonen has shown that when the network is presented with an input pattern and allowed to become stable over a period of time, a 'bubble' of active output nodes will emerge. The radius of this bubble (neighbourhood) depends on the lateral feedback function, and the centre will be the node which responded most strongly when the pattern was first presented. Kohonen then obtains the following shortcut learning algorithm, for a network with N inputs and M outputs.

(1) Set the values of the weight vectors $\mathbf{m}_i$ ($i = 1, M$) for each of the output nodes to random values, and set the radius of the neighbourhood set of nodes $\mathbf{N}_c$;

(2) Present the network with an input vector $\mathbf{x}$;

(3) Find the centre of the bubble $\mathbf{c}$ where:

$$\|\mathbf{x} - \mathbf{m}_c\| = \min\{\|\mathbf{x} - \mathbf{m}_i\|\}$$

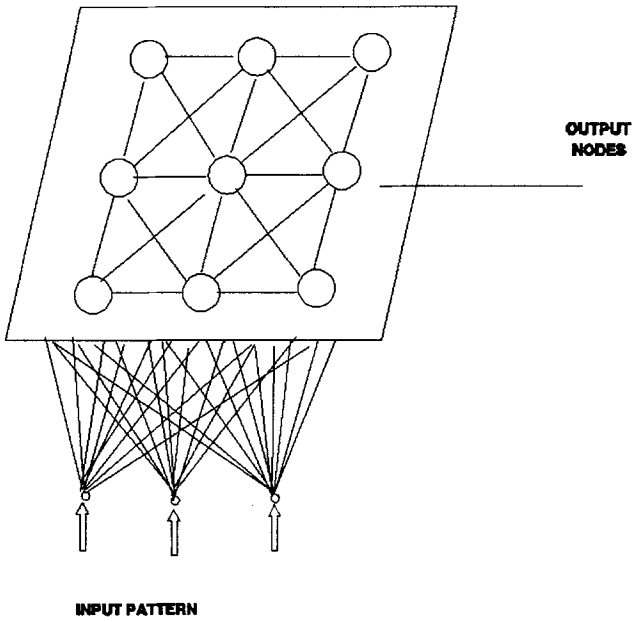

Fig. B1. Two-dimensional array of output nodes used to form feature maps.

(4) Update the weight vectors of all nodes in the neighbourhood of $\mathbf{c}$ using gain α:

$$m_i = m_i + \alpha \cdot (\mathbf{x} - \mathbf{m}_i)$$
$$\text{for} \quad \mathbf{i} \in \mathbf{N}_c$$

(5) Repeat from step 2.

Kohonen states that for good self-organizing results, both the radius of the neighbourhood and the gain α should decrease as the number of presentations of input vectors increases.

References

Baldi, P. and Hornik, K. (1989) Neural network and principal component analysis, *INNS Neural Networks*, **2**(1), 53–58.

Carpenter, G. A. and Grossberg, S. (1987) A massively parallel architecture for a self organising pattern recognition machine, *Computer Vision, Graphics and Image Processing*, **37**, 54–115.

Deleeuw, J., Young, F. W. and Takani, Y. (1976) Additive structure in quantitative data; an alternating least squares method for optimal scaling, *Psychometrica*, **41**, 471–503.

Hewitt, P. D., Skitt, P. J. C. and Witcomb, R. C. (1989) A self organising feedforward network applied to acoustic data, *IOP (Physics World)*, November.

Holden, N. and Sanders, S. (1989) New opportunities from medium frequency welding, in *Proceedings of ISATA Conference*, Wiesbaden.

Kohonen, T. (1989) *Self-organisation and Associative Memory*, Springer Verlag (3rd edition).

Kuczewski, R. M., Myers, H. and Crawford, W. J. (1987) Exploration of backward error propagation as a self organisational structure, *Proceedings of IEEE 1st International Conference on Neural Networks*, San Diego, CA, Vol. 2, pp. 89–95.

Lippman, R. P. (1987) An introduction to computing with neural nets, *IEEE ASSP Magazine*, April.

McGregor, G. (1983) The resistance spot welding of metallic coated steel, *Metals Australasia*, March.

Page, E. S. (1954) Continuous inspection schemes, *Biometrika*, **41**, 1054, 100–115.

Rumelhart, D. E., McLelland, J. L. and PDP Group (1987) *Parallel Distributed Processing*, MIT Press, **1**, pp. 318–362.

Tanaka, Y., Sakaguchi, M., Shirasawa, H. *et al.* (1985) Electrode life in resistance spot welding of zinc plated steel sheets, *International Journal of Vehicle Design*, **6**(4/5).

Witcomb, R. C., Skitt, P. J. C. and Hewitt, P. D. (1989) Adaptive acoustic monitoring of aircraft engines, in *Proceedings of First International Congress on Condition Monitoring and Diagnostic Engineering Management*, Birmingham, UK, September, p. 194–198.