



Combining multi-layer perceptrons with heuristics for reliable control chart pattern classification

D.T. Pham & E. Oztemel

Intelligent Systems Research Laboratory, School of Electrical, Electronic and Systems Engineering, University of Wales College of Cardiff, P.O. Box 917, Cardiff CF2 1XH, UK

ABSTRACT

A hybrid system for on-line control chart pattern classification is presented. The system comprises three different pattern classification modules, a rule-based module and two multi-layer perceptron modules. Each module is set up, initialised and trained independently. The outputs of the hybrid system are produced by a decision making module which synergistically combines the outputs of the individual modules.

INTRODUCTION

Statistical process control charts can exhibit patterns which reflect the long term behaviour of the process being monitored. These patterns can indicate that the process is operating normally ("normal" patterns) or that an abnormal situation is taking place, for example, the process is limit cycling ("cycles") or there is drift in the process ("increasing trends" or "decreasing trends"), or a more sudden step change ("upward shifts" or "downward shifts").

A number of techniques have been developed for control chart pattern recognition. As will be seen in the next section, these techniques belong to two main groups: those employing heuristic rules and those based on neural networks. This paper describes a hybrid system adopting techniques from both groups. The system comprises three separate classification modules. One of the classification modules is a program implementing heuristically-derived rules and the other two modules are based on neural networks. The neural networks employed are multi-layer



perceptrons (MLPs). The purpose of using different pattern recognition modules is to ensure that they act as "specialists" with different pattern recognition skills. A decision making module then synthesises the outputs of the hybrid system by combining the outputs of these individual pattern recognition specialists. As will be seen later, the synergy achieved in the coordinated work of the team of specialists has resulted in better performances than obtainable with each specialist individually.

PREVIOUS WORK

Previous work on automatic recognition of control chart patterns has used either expert systems or neural networks. The information encapsulated in expert pattern recognition systems typically consists of special templates (Cheng[1]) or statistical hypotheses and heuristics (Swift[2], Pham and Oztemel[3]). An advantage of this kind of information is its explicit nature. It can therefore be readily examined, for example, to find out how the pattern recogniser operates. If necessary the information can also be modified or updated with relative ease. However the information has to be supplied by a human expert in the first instance and extracting it from him can be a complex and time consuming process. Another drawback with expert pattern recognition using pre-defined templates and rules relates to the handling of arbitrary patterns which have not previously been encountered. Usually, the pattern recognition system would not be able to classify such patterns. Neural-network-based pattern recognisers (Hwarng and Hubele[4], Pham and Oztemel[5], Pham and Oztemel[6]), on the other hand, perform identification and classification with minimum process knowledge requiring only to be trained with examples. They can generalise from the given examples, which enables arbitrary patterns to be classified. However, a problem with these pattern recognisers is that the information they contain is implicit and virtually inaccessible. This creates difficulties when the information has to be examined, for example, to determine how a particular classification decision has been reached. Another problem is that there is no systematic way to select the correct topology and structure for a neural-network-based pattern recogniser. In general, this has to be found empirically, which can sometimes be a lengthy process.

HYBRID PATTERN CLASSIFICATION SYSTEM

The general structure of the hybrid system is shown in Figure 1. This section presents the implementation details of the pattern recognition modules and describes the operation of the decision making module shown in the figure.

Rule-based module

Rule-based programs generally embody a set of heuristic rules about a particular problem domain. These rules incorporate common sense information that is largely incapable of proof. The rules used in the rule-based module contain information regarding

the statistics expected of each type of pattern (for example, what the mean of a pattern should be in relation to the mean of the process parameter being monitored for it to be a normal pattern).

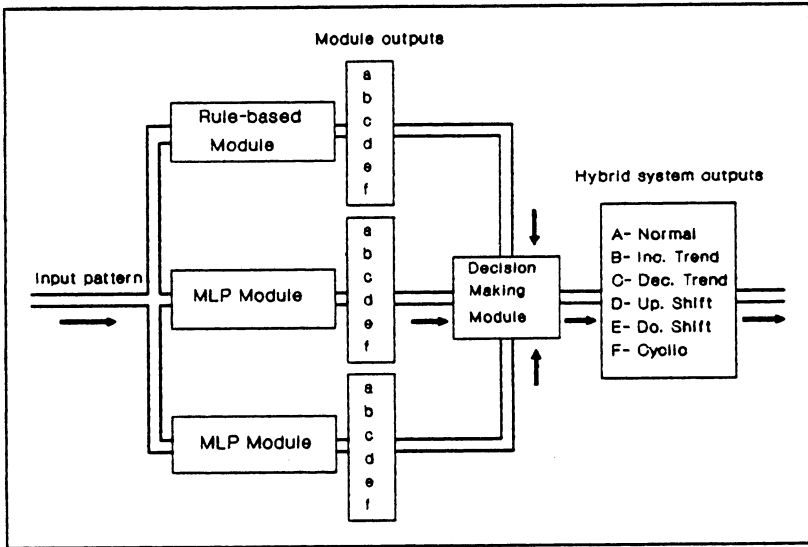


Figure 1. General structure of the hybrid system

In addition to rules, a rule-based program also has data and factual information. This includes, for instance, the mean value and standard deviation of the process parameter, the process mean and standard deviation, the maximum allowed deviation from the process mean (process mean deviation threshold) for a normal pattern, the minimum slope (slope threshold) for a trend and the mean-square linear-regression error (error threshold) for trends, cycles, and normal patterns.

Finally, a rule-based program usually also includes procedures for mathematical and statistical computations. For example in the rule-based module, there are procedures for computing the mean of the pattern and performing linear regression analysis.

The classification rules for different pattern types are summarised below.

Normal patterns. The mean of a normal pattern should not be much different from that of the process. In addition, a good fit should be obtainable for a straight line with a slope below the slope threshold for a trend-type pattern. Thus to detect if the pattern is normal, statistical mean analysis and linear regression analysis are undertaken. If the mean of the points in the pattern is not significantly different from the process mean and both the



slope of the fitted straight line and the regression error are below the respective thresholds, then the given pattern is classified as normal.

Trend-type patterns. If the slope of the fitted straight line is above the slope threshold and the regression error is less than the error threshold, then a trend is present. A positive slope yields an increasing trend and a negative slope, a decreasing trend.

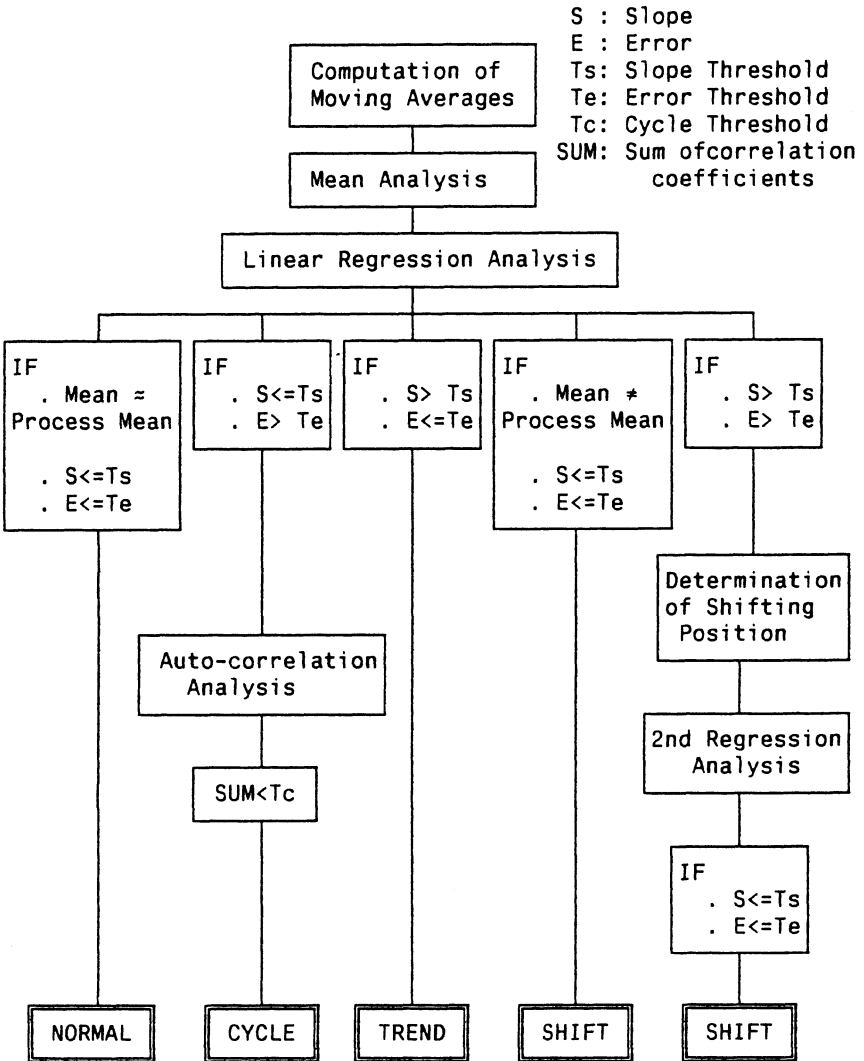


Figure 2. Rule-based pattern classification procedure

Shift-type patterns. A shift occurring at or near the beginning of the pattern is indicated by the following conditions:-

- .the pattern mean being significantly different from the process mean;
- .the slope of the least-square straight line fitted to the pattern being below the slope threshold;
- .the least-square regression error being less than the error threshold.

For a shift at some intermediate point in the pattern, the following conditions hold:-

- .the slope of the least-square straight line fitted to the entire pattern exceeds the slope threshold;
- .the regression error for the above straight line is higher than the error threshold;
- .the slope of the least-square straight line fitted to the part of the pattern after the shift position is below the slope threshold.

Cyclic patterns. If the least-square straight line fitted to a given pattern has a slope below the slope threshold and the linear regression error exceeds the error threshold, that pattern is likely to exhibit a cyclic behaviour. Auto-correlation analysis is then carried out on the pattern to compute the correlation coefficients for it. If the sum of these coefficients is nearly zero (ie. the auto correlogram for the pattern is cyclic), the pattern is confirmed as cyclic.

The decision tree for the rule-based module is shown in Figure 2.

Multi-layer perceptron module

The principles of multi-layer perceptrons (MLP) are described in Rumelhart *et al.*[7]. The MLP modules adopted in this work consisted of three layers: an input layer, a hidden layer and an output layer (see Figure 3). The input layer which received the pattern to be identified had 60 neurons, one for each point in the pattern. (The pattern was a time series comprising 60 consecutive points). The hidden layer which extracted features from the input pattern comprised 35 neurons. That number was arrived at following experimentation with hidden layers of various sizes. The output layer, which processed extracted features to obtain the pattern class, had 6 neurons, one dedicated to each of the available classes.

The neurons in the input layer had unity activation (or transfer function) and simply transmitted the scaled values of the pattern points directly to the hidden layer. The processing by the neurons in the hidden and output layers was implemented with semi-linear (sigmoidal) activation functions (Rumelhart[7]). Inputs to the network were continuous and in the range 0-1. The network outputs were also continuous and in the same range. When one output value was above a threshold (set at 0.8) the



input pattern was considered correctly classified if it belonged to the class represented by that output.

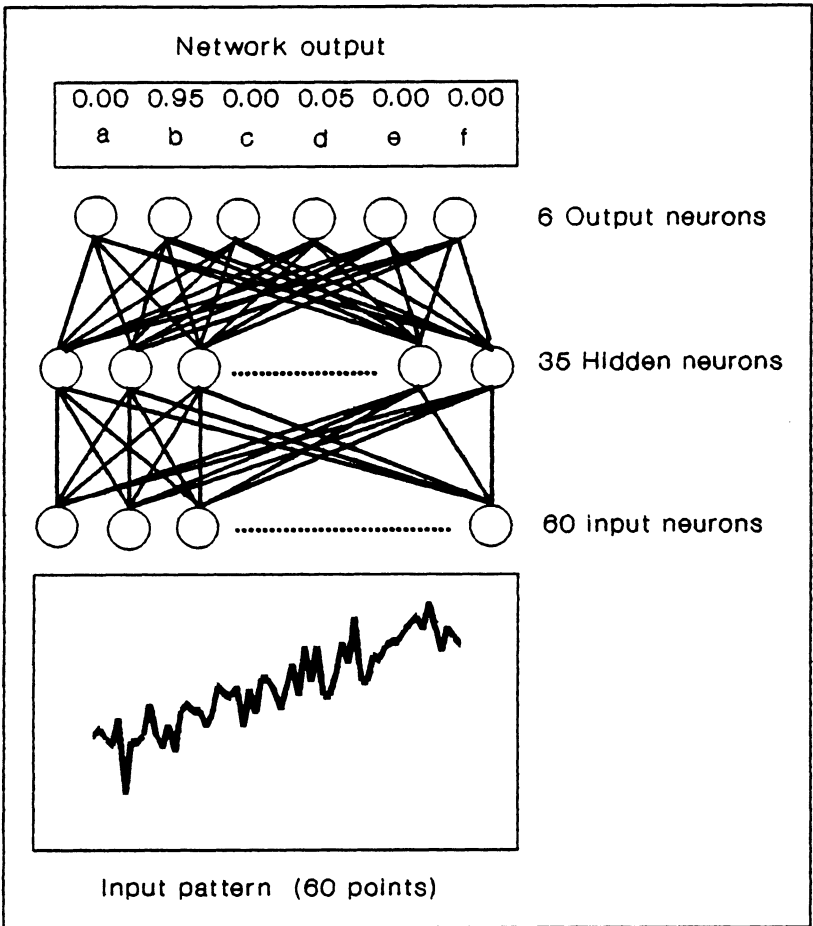


Figure 3. Structure of a multi-layer perceptron module

Two pairs of MLP pattern recognition modules were developed. These had identical structures but were trained in different ways. The first pair was taught a data set comprising 498 patterns of 6 types (83 patterns of each type), both MLP modules being shown the same training data. With the second pair, each MLP module had to learn a different data set. The two data sets were of the same size and also contained 498 patterns of 6 types. The data presentation was random for the first pair and followed a predetermined order for the second pair. In both cases the data sets were presented to each module two hundred times.

All MLP modules were trained with a learning rate of 0.3 and a momentum coefficient of 0.8. The weights of the connections in the MLP modules in the first system were initially randomly set to values between -1 and 1. The connection weights for the MLP modules in the second system had initial values in the range -0.1 to 0.1.

Decision making module

The decision making module computed the final six outputs of the hybrid system from the outputs of the individual pattern recognition modules as follows:-

(i) The corresponding outputs of the three pattern recognition modules were summed up (eg. outputs "a" of modules 1,2 and 3 were added together). This produced six sums.

(ii) If S_{\max} , the largest of the sums computed in step (i), exceeded a given threshold τ_1 (set after experimentation at 2.0) and all the other five sums were below τ_1 , the system output corresponding to the module outputs that produced S_{\max} was set to 1; the other system outputs were set to 0. (For example, system output "A" would be set to 1 and system outputs "B"- "F", to 0, if S_{\max} was the sum of outputs "a" of modules 1,2 and 3.) Otherwise the next step was taken.

(iii) The corresponding outputs of the modules were added in pairs (eg. outputs "a" of module 1 and module 2 were added). For each of the six groups of corresponding outputs, three "pairwise" sums were thus obtained (eg. the sums Σ_{a12} , Σ_{a13} , Σ_{a23} , of outputs "a" of modules 1 and 2, modules 1 and 3, and modules 2 and 3).

(iv) If the overall largest pairwise sum Σ_{\max} produced in step (iii) was above a threshold τ_2 (empirically set at 1.5) and the largest sums for individual groups (except the group that produced Σ_{\max}) were all below τ_2 , the system output corresponding to the module outputs that produced Σ_{\max} was set to 1; the other system outputs were set to 0. (For example, if Σ_{\max} was produced by outputs "a" of modules 1 and 2, system output "A" would be set to 1 and system outputs "B"- "F", to 0.) Otherwise the next step was taken.

(v) Each system output was set to half the largest pairwise sum produced by its group of corresponding module outputs, that is the average of the largest two corresponding module outputs in the group. (For example, system output "A" would be set to $0.5 \Sigma_{a12}$ if Σ_{a12} was the largest among the pairwise sums Σ_{a12} , Σ_{a13} and Σ_{a23} of outputs "a" of modules 1,2 and 3.)

RESULTS

The individual modules and the hybrid system were evaluated on a test set including 1002 previously unseen patterns (167 patterns of each type). The classification accuracy of a module and that of the hybrid system are calculated as:-



$$\text{Classification accuracy(\%)} = \frac{\text{Number of test patterns correctly classified}}{\text{Total number of test patterns presented}} * 100$$

As shown in Table 1, the classification accuracies of the rule-based and MLP modules were 94.8%, 95.2% and 95.3% respectively when the MLP modules were trained with the same data set and 94.8%, 95.2% and 94.3% respectively when different sets were employed. Table 1 also shows that the hybrid system performed better than its individual pattern recognition modules. The hybrid system was able to classify 97.7% of the patterns in the test set correctly when the two neural network modules were trained with the same data. This accuracy level increased to 98.2% when each neural network was shown a different training data set.

CONCLUSION

This paper has described a hybrid system for control chart pattern recognition. The system clearly exhibited a superior performance compared to its individual pattern recognition modules. The latter acted as "specialists" with different backgrounds working together to solve a given pattern classification problem. The synergy arising from collaboration between these "specialists" could be regarded as the main reason for the enhanced performance of the hybrid system.

Module	Classification Accuracy (%)	
	Same data	Different data
Heuristic module	94.8	94.8
MLP module 1	95.2	95.2
MLP module 2	95.3	94.3
Hybrid system	97.7	98.2

Table 1. Performances of the hybrid system and its components

A simple way of obtaining different specialists from a basic neural network module is to train it with different data sets. Where there is insufficient data to construct different sets, non-identical "specialists" could still be trained by varying the training conditions, thus causing the networks to converge to different solution points. This explains why for the case where only one



data set was employed the data was presented randomly during training and the range of the initial weights was chosen to be larger than for the case where two data sets were used.

Being a rule-based program, one of the pattern recognition modules was indeed very different from the other two modules. The rules embodied in the program were simple heuristics derived by examining the available data. As a component of the hybrid system, the rule-based program enabled the majority (approximately 95%) of classification decisions to be explained. The handling of arbitrary patterns, which ordinary rule-based programs are incapable of, was made possible by adopting neural networks as the other remaining pattern recognition modules in the hybrid system.

ACKNOWLEDGEMENTS

The authors would like to thank the ACME Directorate of the Science and Engineering Research Council, STS Ltd and Performance Vision Ltd for supporting this work. E. Oztemel would also like to thank Sakarya University Engineering Faculty for sponsoring his doctoral studies.

REFERENCES

1. Cheng C., *Group technology and expert system concepts applied to statistical process control in small batch manufacturing* Ph.D dissertation, Graduate College, Arizona State University, Tempe, AZ, 1989
2. Swift J. A., *Development of a knowledge based expert system for control chart pattern recognition analysis* Ph.D dissertation, Graduate College, Oklahoma State University, Stillwater, Oklahoma, 1987
3. Pham, D.T. and Oztemel, E. 'A knowledge-based statistical process control system,' pp. INV-4.2.1 - INV-4.2.6, *ICARCV'92, Proceedings 2nd International Conference on Automation, Robotics and Computer Vision*, Vol. 3, Singapore, 16-18 September 1992.
4. Hwang, H.B. and Hubele, N.F. 'X-Bar chart pattern recognition using neural nets,' pp. 884-889, *45th annual quality congress. American Society for Quality Control*, Milwaukee, 20-22 May 1991.
5. Pham, D.T. and Oztemel, E. 'Control chart pattern recognition using neural networks' *Journal of Systems Engineering, Special issue on neural networks* 2(4), pp. 256-262, 1992
6. Pham, D. T. and Oztemel, E. 'Control chart pattern recognition using learning vector quantisation neural networks' *Submitted to International Journal of Production Research*.



7. Rumelhart, D.E., Hinton, G.E. and Williams, R.J. 'Learning internal representation by error propagation' *Parallel Distributed Processing* eds. Rumelhart D.E. and McClelland J.L., Vol. 1, pp. 312-362, MIT Press, Cambridge, MA, 1986.