Unsupervised adaptive resonance theory neural networks for control chart pattern recognition

D T Pham* and **A B Chan**
Manufacturing Engineering Centre, School of Engineering, University of Wales Cardiff, Wales, UK

Abstract: This paper describes the use of unsupervised adaptive resonance theory ART2 neural networks for recognizing patterns in statistical process control charts. To improve the classification accuracy, three schemes are proposed. The first scheme involves using information on changes between consecutive points in a pattern. The second scheme modifies the ART2 vigilance parameter during training. The third scheme merges class neurons representing the same class after training. The paper gives results which demonstrate the improvements achieved.

Keywords: neural networks, pattern recognition, control charts

1 INTRODUCTION

Control charts are employed in statistical process control (SPC) to assess whether a process is functioning correctly. A control chart can exhibit six main types of patterns: normal, cyclic, increasing trend, decreasing trend, upward shift and downward shift [**1**, **2**]. Figure 1 depicts these six pattern types. Correct identification of these patterns is important to achieving early detection of potential problems and maintaining the quality of the process under observation.

Different artificial intelligence techniques such as rule-based expert systems and neural networks have been implemented for control chart pattern recognition. An expert system was developed by Swift for the task [**3**]. Cheng [**4**] used special templates to classify control chart patterns in an expert system. Rule-based expert systems contain explicit information. Therefore, the rules can be modified easily to reflect new information. Where necessary, such systems can provide explanations to the human operator regarding the decisions that they have reached. The main drawback with rule-based systems is the difficulty in obtaining the rules to be embedded in them. This is often referred to as the knowledge acquisition bottleneck.

Supervised neural networks have also been applied to control chart pattern recognition [**5**–**8**]. The main advantage of using supervised neural networks is that explicit rules and templates are not required. Learning is undertaken automatically in a training phase with representative examples [**9**]. Although supervised neural networks are not capable of providing explicit explanations for their decisions, their autonomous learning capability makes them suitable for problems where little *a priori* knowledge is available. There is one significant disadvantage with using supervised neural networks: they are not suitable for continuous, incremental on-line training owing to a problem known as the stability and plasticity problem [**10**].

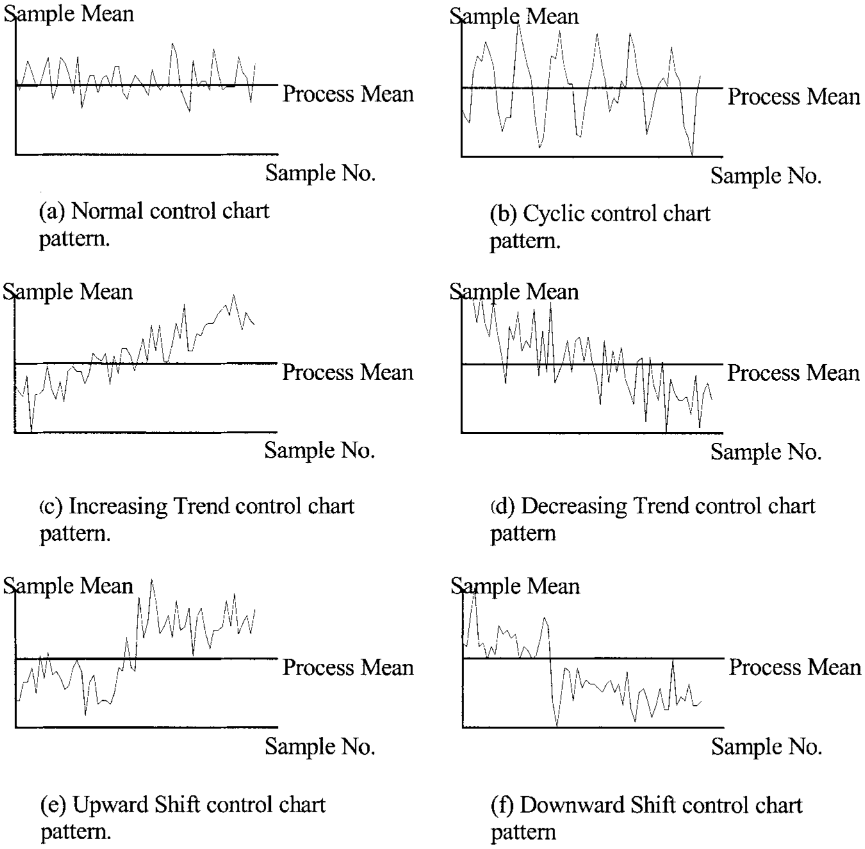
When a trained supervised neural network is required to learn new data, it tends to lose some of the knowledge obtained from previous training. While learning the new data set (i.e. being plastic), the network may forget what it previously learnt (i.e. it is unstable). To train a supervised neural network to learn additional data, a new training set is required comprising those data and the data already learnt by the network. The network then has to be retrained with this new training data set, which means that it has to be taken off line for training.

This paper describes the application of the ART2 network [**10**] to control chart pattern classification. It has been demonstrated [**10**] that ART2 networks are both plastic and stable in that they can learn new data without erasing currently stored information. Thus ART2 networks are suitable for continuous, incremental on-line training. The architecture and dynamics of a basic ART2 implementation are first explained. Classification results with this particular version of ART2 are presented. Three schemes for improving the classification performance are then detailed together with the comparative results obtained.

The MS was received on 4 June 1999 and was accepted after revision for publication on 10 April 2000.

Corresponding author: Manufacturing Engineering Centre, School of Engineering, University of Wales Cardiff, Queen's Buildings, The Parade, Newport Road, PO Box 688, Cardiff CF2 3TE, Wales, UK.

© IMechE 2001



(a) Normal control chart pattern.

(b) Cyclic control chart pattern.

(c) Increasing Trend control chart pattern.

(d) Decreasing Trend control chart pattern

(e) Upward Shift control chart pattern.

(f) Downward Shift control chart pattern

Fig. 1 Main types of control chart patterns

2 ART2 NETWORKS FOR CONTROL CHART PATTERN RECOGNITION

2.1 Architecture of an ART2 network

The architecture of an ART2 network is delineated in Fig. 2. In this particular configuration, the 'feature representation' field (F1) consists of four loops. An input pattern will be circulated in the lower two loops first. Inherent noise in the input pattern will be suppressed [this is controlled by the parameters a and b and the feedback function $f(\cdot)$] and prominent features in it will be accentuated. Then the enhanced input pattern will be passed to the upper two F1 loops and will excite the neurons in the 'category representation' field (F2) via the bottom-up weights. The 'established class' neuron in F2 that receives the strongest stimulation will fire. This neuron will read out a 'top-down expectation' in the form of a set of top-down weights sometimes referred to as class templates. This top-down expectation will be compared against the enhanced input pattern by the vigilance mechanism. If the vigilance test is passed, the top-down and bottom-up weights will be updated and, along with the enhanced input pattern, will circulate repeatedly in the two upper F1 loops until stability is achieved. The time taken by the network to reach a stable state depends on how close the input pattern is to passing the vigilance test. If it passes the test comfortably, i.e. the input pattern is quite similar to the top-down expectation, stability will be quick to achieve. Otherwise, more iterations are required. After the top-down and bottom-up weights have been updated, the current firing neuron will become an established class neuron. If the vigilance test fails, the current firing neuron will be disabled. Another search within the remaining established class neurons in the F2 layer will be conducted. If none of the established class neurons has a top-down expectation similar to the input pattern, an unoccupied F2 neuron will be assigned to classify the input pattern. This procedure repeats itself until either all the patterns are classified or the memory capacity of F2 has been exhausted.

The basic ART2 training algorithm can be summarized as follows:

(a) initializing the top-down and bottom-up long-term memory traces;
(b) presenting an input pattern from the training data set to the network;
(c) triggering the neuron with the highest total input in the category representation field;
(d) checking the match between the input pattern and the exemplar in the top-down filter (long-term memory) using a vigilance parameter;

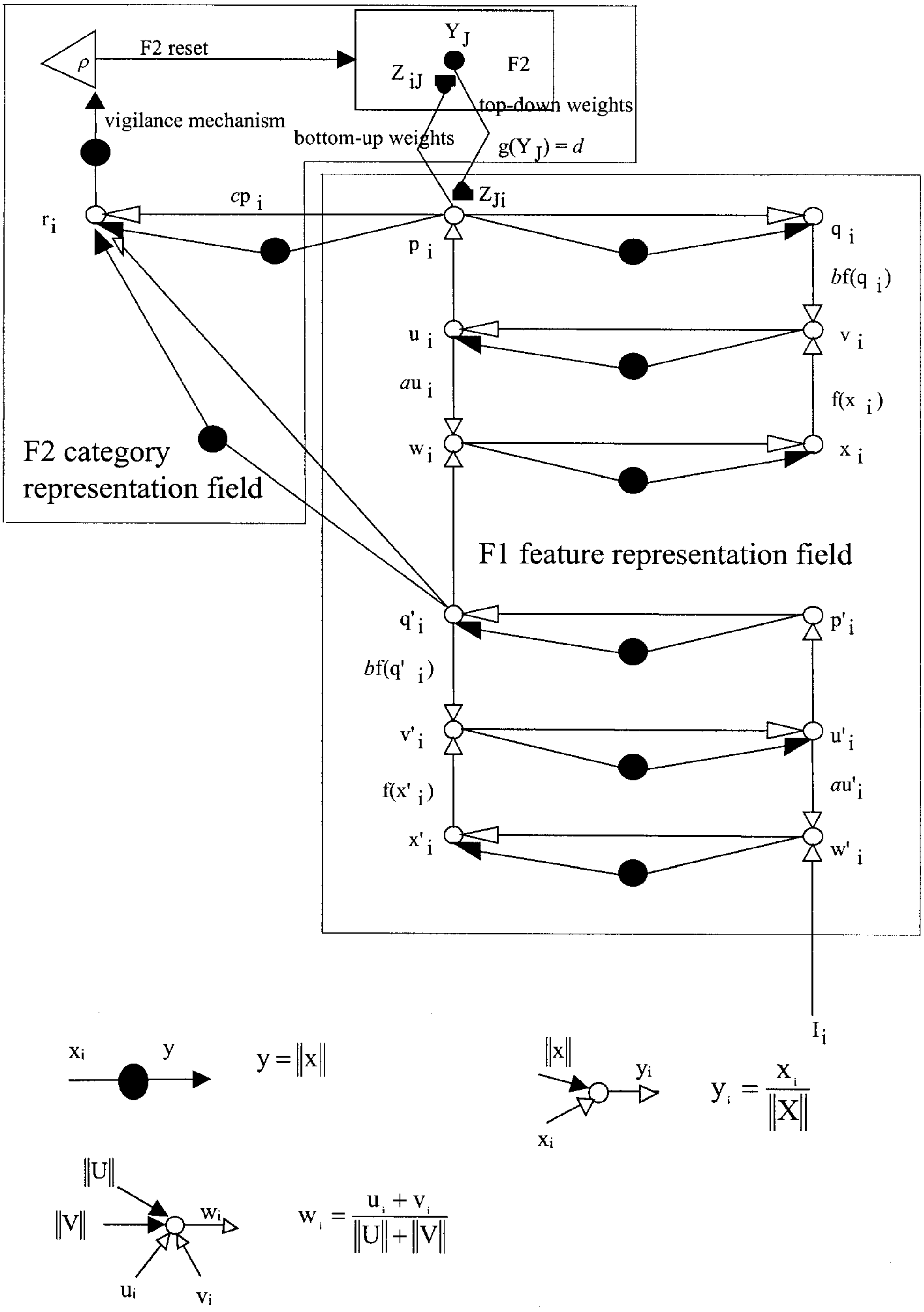


Fig. 2 Architecture of an ART2 network

(e) starting the learning process if the mismatch is within the tolerance level defined by the vigilance parameter and then going to step (h), otherwise moving to the next step;
(f) disabling the current active neuron in the category representation field and returning to step (c), going to step (g) if all the established classes have been tried;
(g) establishing a new class for the given input pattern;
(h) repeating steps (b) to (g) until the network stabilizes or a specified number of iterations is completed.

In the recall mode, only steps (b), (c), (d) and (h) will be utilized.

The equations governing the operation of an ART2 network can be found in the Appendix.

2.2 Classification results

The ART2 network described in the previous section has a set of parameters which are adjustable by the user.

Table 1 Classification results for various ART2 networks

								Classification accuracy	
Network	a	b	c	d	θ	ρ	Neurons	Train data	Test data
1	10.0	10.0	0.1	0.9	0.01	0.996	45	63.39%	60.61%
2	50.0	50.0	0.1	0.9	0.01	0.996	46	64.21%	62.12%
3	1.0	1.0	0.1	0.9	0.01	0.996	27	62.57%	59.85%
4	1.0	10.0	0.1	0.9	0.01	0.996	41	62.30%	61.36%
5	10.0	1.0	0.1	0.9	0.01	0.996	35	60.11%	57.58%
6	10.0	10.0	0.3	0.7	0.01	0.996	17	42.62%	40.15%
7	10.0	10.0	0.01	0.99	0.01	0.996	25	66.67%	69.70%
8	10.0	10.0	0.01	0.99	0.01	0.999	117	83.88%	74.24%
9	10.0	10.0	0.01	0.99	0.1	0.999	210	90.53%	84.22%
10	10.0	10.0	0.01	0.99	0.1	0.992	127	87.90%	82.35%

The training data set contains 366 vectors. The test data set contains 132 vectors. ρ is the vigilance parameter; neurons, the number of neurons established after training has stabilized; train data, the accuracy of the ART2 network when tested with the training data set; test data, the accuracy of the ART2 network when tested with the test data set.

They are a, b, c, d, θ and ρ. The first five parameters, a, b, c, d and θ, are responsible for suppressing noise and enhancing the major features in the input pattern and ρ is the vigilance parameter which determines the sensitivity of the classifier to pattern differences. There is no standard method for deciding the values of these parameters. A series of classification experiments were conducted using networks with different combinations of parameter values. The aim of the experiments was to find a good combination of values for the control chart classification problem. Each network was fed a training data set comprising 366 patterns and a test data set with 132 patterns in each of the six categories. A pattern is a time series of 60 data points. The generation of the different types of patterns for the training and test data sets is described in reference [**5**]. The results are presented in Table 1. A good classifier should have high classification accuracies with both the training and the test data sets and should involve only a small number of classification neurons by comparison with the number of patterns in the data sets.

The values of the parameters used in network 1 were chosen according to the recommendations in reference [**10**]. However, this combination of network parameters, as can be seen in Table 1, did not yield high classification accuracies.

For network 2, the values of a and b were much higher than those for network 1 while all the other parameters remained the same. The result was that both the classification accuracies and the number of established classification neurons increased slightly. This showed that increasing the parameters a and b was not an effective way of improving the classification accuracy, nor was it an effective method for reducing the total number of class neurons needed for the classification task.

In network 3, the values of a and b were set low. A significant reduction in classification neurons was obtained compared with network 1 and network 2. Although the number of neurons decreased almost by one-half, the classification accuracy only deteriorated slightly compared with network 2.

Networks 4 and 5 had a and b with different values. In network 4, a was smaller than b and, in network 5, b was smaller than a. This approach did not seem to offer better performances than the other combinations of a and b.

The results obtained for networks 6 and 7 show the effect of adjusting the balance between the parameters c and d. It was discovered that if the value for c was raised while the value for d was decreased as in network 6, the classification accuracy dropped significantly although the number of established neurons was reduced. With d taking a large value and c a much reduced value, network 7 had a higher classification accuracy while suffering a small increase in the number of established class neurons. Nevertheless, the classification accuracy attained by network 7 was still too low for it to be adopted as a control chart pattern classifier.

Network 8 was an attempt to improve the classification accuracy by raising the vigilance parameter. The classification accuracy did increase as a result, but at the expense of having many more classification neurons.

Network 9 had a larger value for θ. The classification accuracy again improved. However, this was achieved with 210 classification neurons which was far too many for the size of the training set. As there were 366 patterns in the training data set, on average each neuron in the network classified only 1.7 patterns. As an attempt to address this problem, network 10 was given a smaller vigilance parameter. The number of established classification neurons was almost halved and, although the classification accuracies decreased as well, they were still comparable to the accuracies achieved with supervised neural networks [**11**].

Thus, from Table 1, the following observations can be made about the choice of parameters of ART2 networks for control chart pattern recognition: (a) making a and b different from each other does not have too much effect on the network performance—it is useful to keep both a and b low; (b) d should be significantly larger than c; (c) θ

influences the sensitivity of the overall network—it should not be larger than 1, otherwise each input pattern would require a classification neuron which would make the task of the classifier trivial; (d) the vigilance parameter should be kept high to ensure that different patterns are separated into distinct classes.

The general validity of observations (c) and (d) can readily be confirmed by referring to the roles of parameters θ and ρ to which those observations relate (see equation (13) and relation (18) in the Appendix). Because the functions of parameters a, b, c and d are more complex, observations (a) and (b) should be regarded as applicable principally to the test cases presented in Table 1. However, with ART2, where, as mentioned previously, there is no standard method for parameter selection, these tentative observations can still prove helpful as starting points for the network designer.

2.3 New input presentation scheme

All the ART2 networks detailed in Table 1 used solely the input patterns in the process of classification. The performance might improve if the input patterns as well as their first-order differences were employed in the classification process because this would make use of the contextual information in the relationship between the neighbouring elements of an input pattern. When an input pattern is presented, its first 'derivative' would be extracted and then added to the input pattern before it is given to the ART2 network for classification.

Consider the following input pattern:

$$\mathbf{x} = [x_1, x_2, x_3, x_4]^{\mathrm{T}}$$

Its first-order difference vector is taken as

$$\mathbf{x}' = [x_2 - x_1, x_3 - x_2, x_4 - x_3]^{\mathrm{T}}$$

The overall input pattern presented to the network would be

$$\mathbf{x}_{\text{new}} = [x_1, x_2, x_3, x_4, x_2 - x_1, x_3 - x_2, x_4 - x_3]^{\mathrm{T}}$$

Therefore, if the original input pattern had n elements, the final pattern would have $2n - 1$ elements.

The results for a group of networks trained using this scheme are shown in Table 2. The other network parameters were exactly the same as those used for the networks described in Table 1.

The results for the first five networks in Table 2 again confirm that variations in parameters a and b did not have much effect on the overall classification performances. This is consistent with the observation made on the first five networks in Table 1. Networks 5 to 8 underline the point that, to achieve good accuracy in control chart pattern classification, parameter d should be larger than parameter c. As recorded in Table 2, among networks 5 to 8, the network demonstrating the best classification and generalization capability has $c = 0.01$ and $d = 0.99$. This agrees with findings reported earlier for the networks in Table 1.

Networks 8 and 9 in Table 2 show that, when θ was increased from 0.01 to 0.1, the sensitivity of the classifier to different pattern classes was enhanced. Nevertheless, the number of established class neurons increased with θ as observed previously. Also, by reducing ρ slightly, a degree of control of the neuron number was achieved. Overall, network 9 in Table 2 gave the best classification accuracies.

Comparing Tables 1 and 2, it can be seen that the average classification accuracy of the networks in Table 2 is higher than that of the networks in Table 1. Also, the average number of established class neurons of the networks in Table 2 is smaller. Therefore, using the scheme proposed in this section, an ART2 network can attain better generalization and accuracy in control chart pattern classification.

A disadvantage with this scheme is that the input patterns are higher in dimension than those used by the networks in Table 1. Consequently, the training and processing times are longer. Typically, the networks in Table 2 required twice as long to train and to process input patterns as the networks in Table 1. This could be a serious drawback in a time-critical application.

Table 2 Classification results with augmented input patterns

								Classification accuracy	
Network	a	b	c	d	θ	ρ	Neurons	Train data	Test data
1	10.0	10.0	0.1	0.9	0.01	0.996	35	68.96%	65.34%
2	50.0	50.0	0.1	0.9	0.01	0.996	38	69.08%	65.77%
3	1.0	1.0	0.1	0.9	0.01	0.996	20	64.57%	63.98%
4	1.0	10.0	0.1	0.9	0.01	0.996	37	65.48%	59.97%
5	10.0	1.0	0.1	0.9	0.01	0.996	28	63.56%	60.39%
6	10.0	10.0	0.3	0.7	0.01	0.996	15	50.03%	46.65%
7	10.0	10.0	0.01	0.99	0.01	0.996	19	70.88%	69.78%
8	10.0	10.0	0.01	0.99	0.01	0.999	69	89.93%	85.46%
9	10.0	10.0	0.01	0.99	0.1	0.999	158	91.22%	90.65%
10	10.0	10.0	0.01	0.99	0.1	0.992	73	90.35%	85.69%

3 MODIFICATIONS TO THE ART2 LEARNING RULE

3.1 Decreasing the vigilance parameter in the course of training

With the ART2 architecture described in Section 2, too many classification neurons were needed to obtain good accuracies. As already noted, the ratio of classification neurons and patterns to be classified approached the trivial value of 1:1.

For an 'intelligent' self-organizing classifier, once the representative class templates have been established, the discriminating criterion should be lowered to achieve good generalization. Therefore, the proposed training scheme involves decreasing the vigilance parameter gradually in the course of training. At the beginning of the training process, the vigilance parameter is set at a high value so that all input patterns with differing prominent features will cause new classification neurons to be formed. This establishes the most representative class templates. The vigilance parameter is allowed to decrease smoothly as training progresses to help each established classification neuron to classify a larger number of similar patterns.

The proposed modification to the ART2 training rule–scheme operates as follows. First, the user defines an initial value (ρ_start) for the vigilance parameter (ρ) before training commences. Then, whenever a new input is presented the vigilance parameter is decreased by $\Delta\rho$ until it reaches a predefined end value (ρ_end). Subsequently, the vigilance parameter will remain at ρ_end until the training phase is finished. In the test phase, the vigilance parameter will take the mean value $(\rho\_start + \rho\_end)/2$. The performances of a set of ART2 networks trained using this scheme are shown in Table 3.

3.2 Classification results for monotonically decreasing vigilance parameter scheme

Parameters a, b, c, d and θ for networks 1 to 4 in Table 3 were the same as for the best network in Table 1 (network 9). The values for ρ_start and ρ_end were chosen so that cases with vigilance parameter values above and below the constant vigilance of network 9 in Table 1 could be tested. The vigilance parameter decrement ($\Delta\rho$) was the same for the four networks.

Although the numbers of class neurons obtained for the four networks were smaller than for network 9 of Table 1, the accuracies were also much lower. These results could be attributed to the lower effective values of the vigilance parameter ρ. In the case of network 2, although the value of ρ_start was higher than the constant vigilance parameter for network 9 of Table 1, the vigilance parameter quickly decreased to a value lower than that for the latter network. Thus, effectively, the new network operated with a reduced vigilance overall and thus produced a poorer accuracy.

In networks 5, 6, 7, 8 and 9, the values of a, b, c, d and θ were varied from those of the best network in Table 1. Network 5, with θ equal to one-tenth of the value for network 9 in Table 1, had the smallest number of neurons. However, the classification acccuracies were still low.

Comparing networks 6, 7 and 8, which had the same a, b, c and d values as for network 1 in Table 1, again reveals the effect of adopting higher vigilance parameter values. Network 8 with the highest vigilance among the three networks also produced the best performance, well exceeding that of network 1 in Table 1.

Performance improved with network 9, which had the same parameters as network 5, except that $\Delta\rho$ was now lower. As $\Delta\rho$ controls the rate of decrease in vigilance during training, this effectively means a higher average vigilance was obtained with network 9.

In network 10, the value of $\Delta\rho$ was further reduced. This gave the best of all the results presented in Table 3.

Networks 8, 9 and 10 in Table 3 all had better test accuracies and fewer class neurons than the best original ART2 network (network 9 in Table 1). This shows that the proposed decreasing vigilance scheme can improve the performance of ART2. This is due to the possibility

Table 3 Classification results with the proposed learning rule modification

										Classification accuracy	
Network	a	b	c	d	θ	ρ_start	ρ_end	$\Delta\rho$	Neurons	Train data	Test data
1	10.0	10.0	0.01	0.99	0.1	0.993	0.99	0.00001	72	73.26%	65.41%
2	10.0	10.0	0.01	0.99	0.1	0.9999	0.996	0.00001	91	76.42%	54.72%
3	10.0	10.0	0.01	0.99	0.1	0.998	0.996	0.00001	94	77.15%	53.44%
4	10.0	10.0	0.01	0.99	0.1	0.996	0.99	0.00001	75	64.09%	59.50%
5	10.0	10.0	0.01	0.99	0.01	1.0	0.996	0.00005	30	62.21%	58.42%
6	10.0	10.0	0.1	0.9	0.1	0.99	0.98	0.00005	58	70.61%	63.23%
7	10.0	10.0	0.1	0.9	0.1	0.99	0.97	0.0001	44	75.39%	71.11%
8	10.0	10.0	0.1	0.9	0.1	0.996	0.99	0.00001	62	90.34%	89.99%
9	10.0	10.0	0.01	0.99	0.01	0.9999	0.996	0.00001	73	91.24%	90.52%
10	10.0	10.0	0.01	0.99	0.1	0.9999	0.996	0.000005	96	92.35%	91.67%

ρ_start is the starting vigilance parameter when training commences. ρ_end is the finishing vigilance parameter when training ends. $\Delta\rho$ is the value by which the vigilance parameter is reduced when a new input is presented.

Table 4 Classification results for various ART2 networks after class merging

								Classification accuracy	
Network	*a*	*b*	*c*	*d*	θ	ρ	Neurons	Train data	Test data
1	10.0	10.0	0.1	0.9	0.01	0.996	32	60.41%	56.72%
2	50.0	50.0	0.1	0.9	0.01	0.996	36	59.23%	57.38%
3	1.0	1.0	0.1	0.9	0.01	0.996	25	58.76%	55.27%
4	1.0	10.0	0.1	0.9	0.01	0.996	33	57.92%	56.09%
5	10.0	1.0	0.1	0.9	0.01	0.996	28	52.82%	49.28%
6	10.0	10.0	0.3	0.7	0.01	0.996	16	42.62%	39.31%
7	10.0	10.0	0.01	0.99	0.01	0.996	23	62.87%	64.22%
8	10.0	10.0	0.01	0.99	0.01	0.999	71	80.29%	76.36%
9	10.0	10.0	0.01	0.99	0.1	0.999	89	85.34%	80.73%
10	10.0	10.0	0.01	0.99	0.1	0.992	72	82.25%	78.64%

of using high vigilance parameter values to increase accuracy without expanding the number of class neurons, unlike in the fixed vigilance case.

It has been noted that the performances of the other networks in Table 3 were inferior to that of network 9 in Table 1, the best original ART2 network. This underlines a requirement common to all ART2 networks: their parameters have to be correctly selected to achieve good results, which was the case with network 9 in Table 1, but not with the poorly performing networks in Table 3.

4 POST-TRAINING REDUCTION OF CLASS NEURONS

Although good classification results were achieved with the scheme detailed in Section 3, the number of class neurons established at the end of training is still high for most cases. To solve this problem, a method of reducing the number of established class neurons while maintaining good levels of accuracy is described in this section. The proposed method is to be applied on completion of training. It involves ‘merging’ class neurons which seem to represent the same class. This can be done by comparing the template vectors of the neurons against one another using the vigilance mechanism. A neuron would be considered fit to be merged with another neuron, if, treating its weight vector as the input pattern, it passed the vigilance test against the class template of the other neuron. The same procedure is repeated for all the class neurons at the end of a training session until no further merging is possible. The vigilance parameter used in this merging process has the same value as at termination of training.

Tables 4 and 5 show the results of applying this class neuron merging method to the original ART2 network. Note that the numbers of class neurons were reduced in most cases although this was also accompanied by a decrease in classification accuracy as can be expected.

5 CONCLUSIONS

This paper has described the application of ART2 networks to control chart pattern classification. Three schemes have been proposed to improve the performance of the basic ART2 network. All schemes preserve the ART2 architecture and the essence of the ART2 learning rule. As far as the authors are aware, there have not been other attempts at improving the ART2 network without radically affecting the ART2 paradigm. The first scheme involving adding rate information to the input pattern gave good classification accuracies but lengthened training and processing times. The next scheme related to the learning rule of ART2, in particular the operation of the

Table 5 Classification results with the proposed learning rule modification after class merging

										Classification accuracy	
Network	*a*	*b*	*c*	*d*	θ	ρ_*start*	ρ_*end*	$\Delta\rho$	Neurons	Train data	Test data
1	10.0	10.0	0.01	0.99	0.1	0.993	0.99	0.00001	56	60.48%	55.82%
2	10.0	10.0	0.01	0.99	0.1	0.9999	0.996	0.00001	78	65.25%	41.76%
3	10.0	10.0	0.01	0.99	0.1	0.998	0.996	0.00001	84	73.50%	41.67%
4	10.0	10.0	0.01	0.99	0.1	0.996	0.99	0.00001	59	63.34%	42.03%
5	10.0	10.0	0.01	0.99	0.01	1.0	0.996	0.00005	24	63.39%	53.79%
6	10.0	10.0	0.1	0.9	0.1	0.99	0.98	0.00005	57	68.27%	67.56%
7	10.0	10.0	0.1	0.9	0.1	0.99	0.97	0.0001	34	64.52%	51.28%
8	10.0	10.0	0.1	0.9	0.1	0.996	0.99	0.00001	45	69.25%	57.11%
9	10.0	10.0	0.01	0.99	0.01	0.9999	0.996	0.00001	48	78.28%	77.89%
10	10.0	10.0	0.01	0.99	0.1	0.9999	0.996	0.000005	64	86.59%	84.66%

vigilance mechanism. With this scheme, the vigilance parameter changed during training. This was designed to increase classification accuracies as well as controlling the expansion in the number of class neurons. The first of these aims was achieved although the number of class neurons was still high relative to the number of patterns to be classified. The third scheme was to be applied post-training to reduce the number of class neurons by merging neurons that represent the same class. This scheme produced the expected decrease in neurons but also a deterioration in classification accuracies. There clearly is a trade-off between accuracy and required number of class neurons. Nevertheless, the accuracies achieved with the best ART2 network (Network 9 in Table 1, Network 9 in Table 2 and Network 10 in Table 3) were comparable with those obtained with supervised networks. The main advantage of ART2 networks, however, is that they do not suffer from the stability–plasticity problem of supervised networks and thus are more suitable for continuous online learning of the classification task.

ACKNOWLEDGEMENT

The authors would like to thank the European Regional Development Fund (ERDF) for their support for the work described in this paper.

REFERENCES

1 **Pham, D. T.** and **Oztemel, E.** *Intelligent Quality Systems*, 1996 (Springer, London).

2 **Pham, D. T.** and **Chan, A. B.** Control chart patttern recognition using a new type of self-organising neural network. *Proc. Instn Mech. Engrs, Part I, Journal of Systems and Control Engineering*, 1998, **212**(I2), 115–127.

3 **Swift, J. A.** Development of knowledge based expert system for control chart pattern recognition analysis. PhD thesis, Graduate College, Oklahoma State University, Stillwater, Oklahoma, 1987.

4 **Cheng, C.** Group technology and expert system concepts applied to statistical process control in small batch manufacturing. PhD thesis, Graduate College, Arizona State University, Tempe, Arizona, 1989.

5 **Oztemel, E.** Integrating expert systems and neural networks for intelligent on-line statistical process control. PhD thesis, Cardiff University, 1992.

6 **Pham, D. T.** and **Oztemel, E.** Combining multi-layer perceptrons with heuristics for reliable control chart pattern recognition. In *Artificial Intelligence in Engineering* (Eds G. Rzevski, J. Pastor and R. A. Adey), Vol. VIII, 1993, pp. 801–810 (Computational Mechanics, Southampton).

7 **Smith, A. E.** and **Yazici, H.** Backpropagation neural network approaches to process control: evaluation and comparison. Technical Report 90-22-47, Department of Engineering Management, University of Missouri-Rolla, Rolla, Missouri, 1992.

8 **Hwarng, H. B.** and **Hubele, N. F.** X-bar chart pattern recognition using neural nets. In 45th Annual Quality Congress, American Society for Quality Control, Milwaukee, Wisconsin, 1991, pp. 884–889.

9 **Pham, D. T.** Neural networks in engineering. In Proceedings of the 9th International Conference on *Artificial Intelligence in Engineering*, Pennsylvania, 1994, pp. 3–36.

10 **Carpenter, G. A.** and **Grossberg, S.** ART2: self-organisation of stable category recognition codes for analog input patterns. *Appl. Opt.*, 1987, **26**(23), 4919–4930.

11 **Pham, D. T.** and **Liu, X.** *Neural Networks for Identification, Prediction and Control*, 1995 (Springer-Verlag, London).

APPENDIX

The dynamics and learning mechanism of ART2

Network dynamics

The dynamics of the ART2 network illustrated in Fig. 2 is controlled by the following set of equations:

$$w_i' = I_i + au_i' \tag{1}$$

$$x_i' = \frac{w_i'}{||W'||} \tag{2}$$

$$v_i' = f(x_i') + bf(q_i') \tag{3}$$

$$u_i' = \frac{v_i'}{||V'||} \tag{4}$$

$$p_i' = u_i' \tag{5}$$

$$q_i' = p_i' \tag{6}$$

$$w_i = q_i' \tag{7}$$

$$x_i = w_i \tag{8}$$

$$v_i = f(x_i) + bf(q_i) \tag{9}$$

$$u_i = \frac{v_i}{||V||} \tag{10}$$

$$p_i = u_i + \sum_j g(Y_j)z_{ji} \tag{11}$$

$$q_i = \frac{p_i}{||P||} \tag{12}$$

I_i is the ith input to the network. The symbol $||X||$ represents the L_2 norm of vector X with components x_i. The output of the jth neuron in the classification layer is denoted by $g(Y_j)$. The function $f(\cdot)$ in equations (3) and (9) is a non-linear function, the purpose of which is to suppress noise in the input pattern. The definition of $f(\cdot)$ is

$$f(x) = \begin{cases} 0 & \text{if } 0 \leqslant x < \theta \\ x & \text{if } x \geqslant \theta \end{cases} \tag{13}$$

where θ is a user-defined parameter between 0 and 1.

Learning mechanism of ART2

When an input pattern is applied to the ART2 network, it will pass through the four loops constituting F1 and then stimulate the classification neurons in F2. The total excitation received by the jth neuron in the classification layer is equal to T_j where

$$T_j = \sum_i p_i z_{ij} \tag{14}$$

The neuron that receives the strongest excitation will fire by generating an output of constant value d. Therefore, for the winning neuron J, $g(Y_J)$ equals d. When a winning neuron is determined, the value d will be used to multiply the winning top-down expectation z_{Ji}. When the winning neuron fires, all the other neurons are inhibited from firing and produce zero top-down expectation. Thus, if J is the firing neuron, equation (11) becomes

$$p_i = u_i + dz_{Ji} \tag{15}$$

For other neurons, the equation simplifies to

$$p_i = u_i \tag{16}$$

The top-down expectation pattern is combined with the enhanced input pattern at point r_i before they enter the vigilance test. From Fig. 2, r_i is given by

$$r_i = \frac{q'_i + cp_i}{||Q'|| + ||cP||} \tag{17}$$

The vigilance test fails and the firing neuron will be reset if the following condition is true:

$$\frac{\rho}{||R||} > 1 \tag{18}$$

where ρ is the vigilance parameter.

On the other hand, if the vigilance test is passed (that is, the current input pattern can be accepted as a member of the class represented by the firing neuron), the top-down weights z_{ji} and the bottom-up weights z_{ij} are updated so that the special features present in the current input pattern can be incorporated into the class exemplar represented by the firing neuron. The updating equations are as follows:

$$\frac{\mathrm{d}}{\mathrm{d}t} z_{ji} = d(p_i - z_{ji}) \tag{19}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} z_{ij} = d(p_i - z_{ij}) \tag{20}$$

According to reference [**10**], all the top-down weights should be initialized with the value 0 at the beginning of the learning process:

$$z_{ji}(0) = 0 \tag{21}$$

This is to prevent a neuron from being reset when it is used to classify an input pattern for the first time [**10**].

The bottom-up weights are initialized as follows:

$$z_{ij}(0) = \frac{1}{(1-d)\sqrt{M}} \tag{22}$$

where M is the number of neurons in the input layer. This number is equal to the dimension of the input vectors. Equation (22) ensures that, after all the neurons with top-down expectations similar to the input pattern have been searched, the input pattern can easily access a previously uncommitted neuron [**10**].

Copyright of Proceedings of the Institution of Mechanical Engineers -- Part B -- Engineering Manufacture is the property of Professional Engineering Publishing and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.