Developing cost models by advanced modelling technology

D Stockton[1] and **Q Wang**[2]*
[1]Faculty of Computing Sciences and Engineering, De Montfort University, Leicester, UK
[2]School of Engineering, University of Durham, UK

Abstract: The aim of this paper is to examine the use of artificial neural network (ANNs) in the development of cost models. Although such advanced modelling techniques have been highly successful in many engineering areas, this success has been strongly dependent on the ability to choose the correct ANN structure. In this respect, choosing the most suitable structure for the individual processing elements that make up the ANN is essential. The research reported in this paper, therefore, makes use of the Taguchi methodology to identify best and worst structural elements for ANN processing elements. In order clearly to determine the accuracy of the models developed, cost information has been generated using a published cost model of a turning process. The cost information generated from this model has been used to train ANNs and test the resulting model for estimating accuracy. In order to measure accuracy, the 'percentage average absolute error' value has been adopted. Using this measure, the accuracy of models developed using the best and worst ANN structural elements have been compared with the use of regression analysis. The results indicate that the use of ANN to develop cost models is superior to regression analysis, although both methods fail to develop models that provide useful accuracies when large numbers of variables are involved.

Keywords: cost modelling, artificial neural networks, Taguchi methodology

1 INTRODUCTION

In order to support the product and process development needed to meet market expectations such as greater choice of products, greater choice of manufacturing process and greater emphasis on minimizing overall life cycle costs of products, it is expected that the quantity, type, accuracy and complexity of cost information will need to be greatly increased. These changes will have a dramatic effect on the cost estimating process. This effect will arise owing to the nature of the cost estimating process, which involves both costly and time consuming tasks that require high levels of process and product expertise to arrive at valid cost estimates. Cost models are an essential part of the overall cost estimating process in that they are important methods of deriving cost and process time information. The changes affecting the market environment will have similar effects on both the cost estimating and cost modelling processes, as shown in Table 1.

The MS was received on 30 May 2003 and was accepted after revision for publication on 14 October 2003.
**Corresponding author: School of Engineering, University of Durham, South Road, Durham DH1 3LE, UK.*

2 COST ESTIMATING

Cost estimating [1] is the process of calculating the expected cost resources, i.e. labour, material and overhead costs, that are required to accomplish a manufacturing task or to manufacture or purchase a specific product. The type of estimate, normally characterized by the type of resource costed, and the level of detail in the cost data output and its accuracy depend primarily on the type of decision requiring cost data and the availability of data from which to derive the estimate. For example, cost data are used within a variety of decision types including when designing products, manufacturing products and making process improvements. In general, higher levels of estimating accuracy are normally associated with greater levels of data detail. In terms of product design, for example, the amount of data available for use in cost estimating is normally dependent upon where in the development process the product design is, i.e. at the concept stage few data are available while at the detailed engineering drawing stage much greater amounts are available.

The basic guidelines for cost estimation in manufacturing businesses are provided by Ostwald [1], Cunningham and Dixon [2] and Chang [3]. In general, the process

Table 1 Example of effects on cost estimating/modelling processes

	Limitations of cost estimating/modelling processes				
Changes occurring in the market environment	Greater number of cost models required	Less historical cost data available	Less time available to develop model	Greater product and process complexity	Less process expertise available
Greater choice of products	✓	✓			✓
Greater amounts of product customization	✓	✓	✓	✓	✓
Greater choice of materials	✓	✓		✓	✓
Greater choice of manufacturing processes	✓	✓		✓	✓
Reduced product development cycles		✓	✓		
Greater emphasis placed on life cycle costs	✓	✓		✓	

of developing a cost estimate can be divided into the basic steps shown in Fig. 1. This basic process can vary depending on the characteristics of the cost estimate required or limitations, such as data availability, that

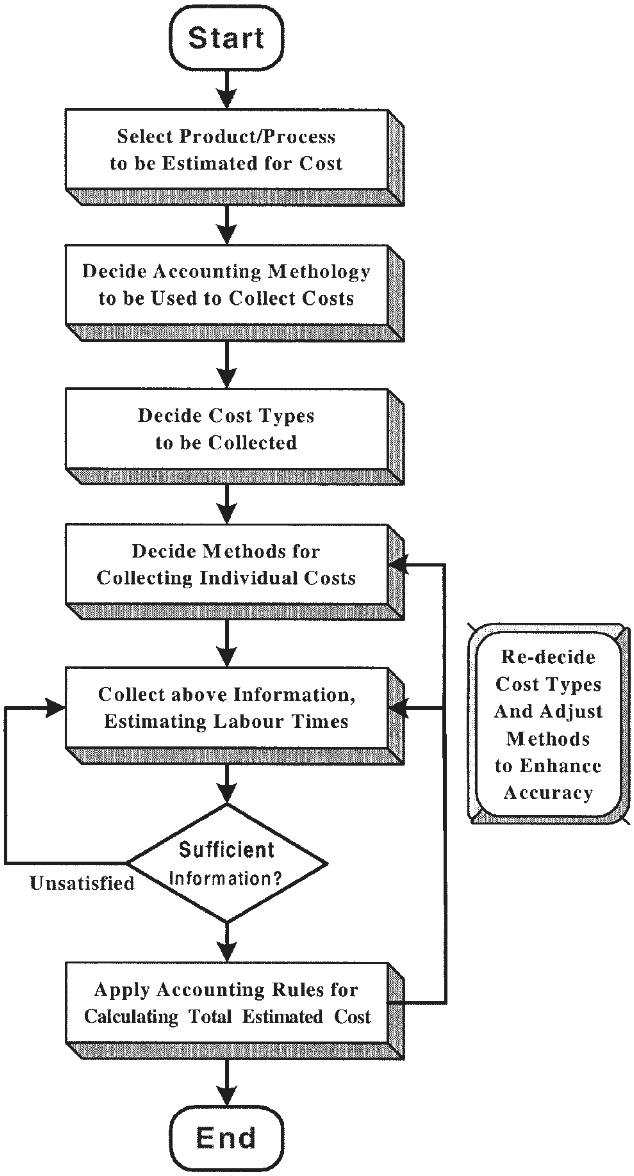

Fig. 1 Basic cost estimating process

may restrict the characteristics of the resulting cost estimate. In this respect the primary characteristics of cost estimates affecting the estimating process are their accuracy, precision, the personnel involved and the type and level of detail of the cost data provided. An example of the relationships between estimating process and model characteristics arises when a high level of accuracy is required which prevents the use of subjective judgement by experienced cost estimators being employed, whereas when only low levels of data are available the use of experienced estimators is essential.

The initial tasks involved in cost estimation are dependent on the products made, the manufacturing processes used and the specific accounting methods employed by individual business organizations. Within manufacturing industry there is a wide variety of manufacturing processes and accounting models in use. Personnel undertaking a cost estimating exercise must ensure they have expertise in the particular processes and accounting methods used by the specific company for which the exercise is being undertaken.

The functional areas, such as engineering, manufacturing and procurement, that are required to perform the work must then be identified, a schedule of the work must be prepared and the level of effort must be defined and identified. This stage also involves selecting an appropriate cost/process time estimating methodology, the main categories of which are shown in Fig. 2 [**4**], and using this method to estimate the man hours, material costs and other cost generating variables. In addition, the elapsed time required to perform each detail of the work must be determined. Cost rates are then identified and used to establish the relevant costs of work elements.

In terms of manufacturing, each of these methods requires an analysis of the work task to be performed, and each can vary in the level of detail required in the definition of the work to be performed. To determine the costs of manufacturing a product or the costs involved in operating a specific production process, it is necessary to identify cost elements and choose an appropriate method for estimating each type of cost. In this respect the majority of cost estimates are normally compiled by utilizing a combination of past similar

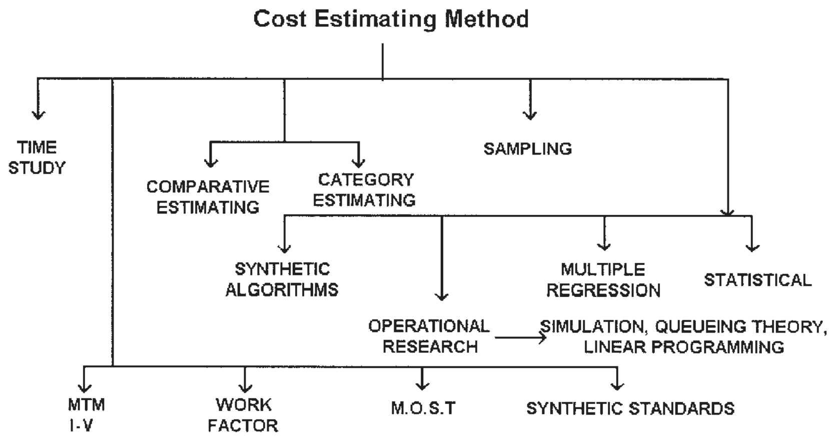

Fig. 2 Cost estimating methods [**4**]

product costs, established in-house cost knowledge and/or published cost information. Published cost information can reduce the cost and time of establishing costs. Such information includes cost indexes, which allow estimates to be adjusted to present industrial environments. For example, Marshall and Swift equipment cost indexes [**5**] are compiled for over 40 different industries. However, existing cost information is of restricted use when attempting to estimate the cost of new manufacturing technology. Recently, artificial neural networks have generated much research interest in the manufacturing area, although many of the applications reported in the literature are either laboratory experiments or preliminary applications.

3 TURNING PROCESS

The objective of the current research was to examine the ability of artificial neural networks (ANNs) to develop cost models; i.e. this ability was measured in terms of the resulting accuracy of the cost models developed. In order to ensure that the level of estimating accuracy of the ANN model could be clearly determined, a spreadsheet version of a known cost model, i.e. that published by Boothroyd and Reynolds [**6**], was developed and used to generate the cost data information used to train and test the ANN models. Smith and Mason [**7**] adopted this method of generating costing examples since it provided the advantage of knowing, for certain, what the true underlying relationships are between predictor variables and costs. Hence, the performance of an ANN in predicting these relationships could be measured with a high level of certainty. In addition, this model was chosen as suitable for use within the experimentation since it contained a relatively large number of predictor variables, i.e. 16, and both linear and non-linear relationships between these variables and process costs.

The model developed by Boothroyd and Reynolds [**6**] can be broken down into the following major elements:

1. The machining time for roughing operations, t_{mp} (s), assuming that maximum power is used, is given by

$$t_{\mathrm{mp}} = \frac{60 r_{\mathrm{v}} p_{\mathrm{s}} W}{d_{\mathrm{m}} a W^{b}} = \frac{60 r_{\mathrm{v}} p_{\mathrm{s}}}{d_{\mathrm{m}} a} W^{1 \cdot b} \tag{1}$$

where

r_{v} = proportion of the initial volume
P_{s} = specific cutting energy or unit power for the work material (hp min/in^3)
W = weight of the workpiece (lb)
d_{m} = density of the work material (lb/in^3)
a = constants
b = constants

2. The non-productive time, t_{np} (s), is given by

$$t_{\mathrm{np}} = \frac{t_{sa} + n_{\mathrm{t}} t_{sb}}{\mathrm{BS} + t_{\mathrm{ln}} + n_{\mathrm{o}} t_{\mathrm{pt}}} \tag{2}$$

where

t_{sa} = basic set-up time for the machine
n_{t} = number of tools
t_{sb} = set-up time per tool
BS = batch size
t_{ln} = loading and unloading time
n_{o} = number of operations
t_{pt} = tool positioning time per operation

3. The finish machining time, t_{mc} (s), will then be given by

$$t_{\mathrm{mc}} = \frac{60A}{R_{\mathrm{sg}}} \tag{3}$$

where

A = effective area to be machined
R_{sg} = machinability factor

4. The effective area to be machined, A, assuming that all surfaces on the workpiece are to be finished,

including a through bore, is given approximately by

$$A = 3.7\left(\frac{1 \cdot r_{\mathrm{i}}}{2\mathrm{LDR}} + 1 + r_{\mathrm{i}}^{0.5} \cdot \frac{r_{\mathrm{e}}}{1.5}\right)\mathrm{LDR}^{0.33}\frac{W^{0.67}}{d_{\mathrm{m}}^{0.67}} \quad (4)$$

where

r_{i} = proportion of material removed by internal machining
r_{e} = proportion of material removed by external machining
LDR = length/diameter ratio of the workpiece

5. Worn tool replacement costs can be significant when operating under optimum (minimum cost) conditions. It has been shown that these costs can be allowed for by modifying equation (3) as follows:

$$t'_{\mathrm{mc}} = \frac{60A}{R_{\mathrm{sg}}}\frac{1}{1 \cdot n} \quad (5)$$

where

n = Taylor tool life index

6. When machining is carried out at maximum power, the corrected value of t_{mp} (s) to allow for tool replacement costs is

$$t'_{\mathrm{mp}} = t_{\mathrm{mp}}\left[1 + \frac{1}{1 \cdot n}\left(\frac{t_{\mathrm{mc}}}{t_{\mathrm{mp}}}\right)^{1/n}\right] \quad (6)$$

When $t_{\mathrm{mc}}/t_{\mathrm{mp}} < 1$, this correction, however, will be small unless the maximum power conditions are close to the recommended conditions, which will not usually be the case for finishing operations.

The variables used in the above turning cost model, and used as variables within the experimentation, are defined as follows:

n = Taylor tool life index
r_{v} = proportion of the initial volume (m^3)
P_{S} = specific cutting energy or unit power for the work material (hp min/in^3)
W = weight of the workpiece (lb)
d_{m} = density of the work material (lb/in^3)
R_{sg} = machinability factor
r_{i} = proportion of material removed by internal machining
r_{e} = proportion of material removed by external machining
LRD = length/diameter ratio of the workpiece
t_{sa} = basic set-up time for the machine (s)
n_{t} = number of tools
t_{sb} = set-up time per tool (s)
BS = batch size
t_{ln} = loading and unloading time (s)
n_{o} = number of operations
t_{pt} = tool positioning time per operation (s)

Table 2 Relationships between variables and cost

Variables	Linear relationship	Non-linear relationship	Both types of relationship
n			✓
r_{v}			✓
P_{S}			✓
W		✓	
d_{m}			✓
R_{sg}		✓	
r_{i}		✓	
r_{e}		✓	
LDR		✓	
t_{sa}	✓		
n_{t}	✓		
t_{sb}	✓		
BS		✓	
t_{ln}	✓		
n_{o}	✓		
t_{pt}	✓		

An analysis of the relationships between each of the above variables and process costs is shown in Table 2 in terms of whether they represent linear and/or non-linear types.

4 ARTIFICIAL NEURAL NETWORKS

From its initial development in the early 1940s [**8**], ANN technology has advanced tremendously in terms of its ability to identify complex relationships. An ANN [**9–12**] consists of a number of computer processing elements. The processing element forms the heart of the ANN, and it is the functions associated with these elements that provide the ANN with the ability to model a wide variety of relationships between input and output variables. An ANN, therefore, consists of many processing elements joined together in the above manner. Processing elements are usually organized into groups called layers, with full or random connections between successive layers. There are typically two layers that possess connections to the outside world, i.e. an input layer where data are presented to the network, and an output layer which holds the response of the network to a given input. Layers distinct from the input and output buffers are called hidden layers.

The ANN structure shown in Fig. 3 [**13**] illustrates how the ANN is made up of these three basic types of layer:

1. The input layer accepts information from external sources and assigns weighted values to these depending on their relative importance as cost drivers.
2. The hidden layer processes this input information and converts it to the required output data.
3. The output layer outputs cost data from the ANN.

The number of processing elements contained in a layer can be varied, as can the number of hidden layers within any individual network. Processing elements within layers are normally 'fully connected'; i.e. an individual

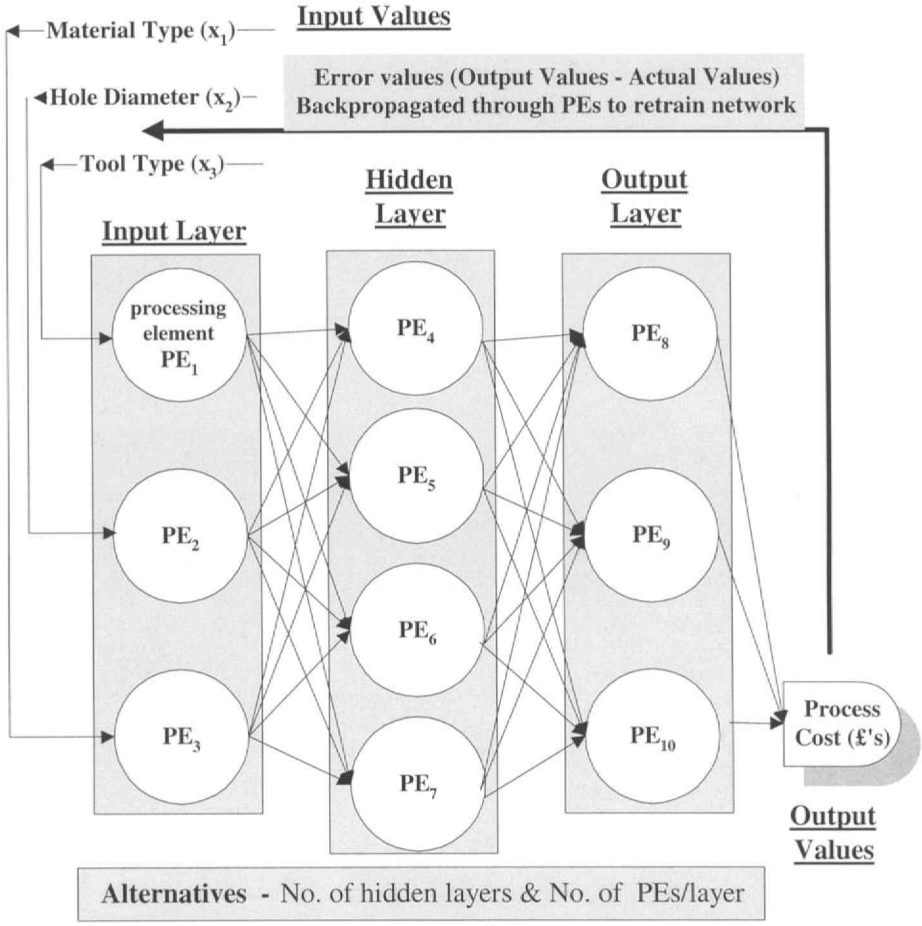

Fig. 3 Artificial neural network structure

processing element within a layer is connected to all processing elements in both the preceding and succeeding layers. Processing elements within the same layer are, however, not connected.

As values for process variables are input into the ANN, the processing elements within the input, hidden and output layers are modified such that the difference between the output cost values and actual cost values, i.e. the error, is gradually minimized. This process, termed 'training the network', is performed within the current research work using 'back-propagation'; i.e. this technique calculates an error between actual values and output values and propagates the error information back through the network to each node (i.e. processing element) in each layer. This back-propagated error then drives the learning at each node. Learning is therefore the process of adapting or modifying the connection weights in response to stimuli being presented at the input buffer and optionally at the output buffer.

Processing elements (PEs) [**14**, **15**] contain a number of mathematical functions, as shown in Fig. 4. These functions act in a sequential manner to transform input values into output values. Input values can either be externally derived input values of predictor variables or outputs from processing elements in a preceding layer. The functional steps within this sequence are as follows:

Step 1: weighted summation function. Weightings are applied to each of the variable values input into a PE, and the summation function then sums these weighted variable values. Two methods of summing the weighted inputs have been examined in the current research:

1. The *sum* is the traditional sum of the effective inputs.
2. The *majority* counts the number of effective inputs greater than zero and subtracts the number of effective inputs less than or equal to zero.

Step 2: transfer function. The result of the weighted sum is transformed into a working output or 'transfer' output by the transfer function. Four types of transfer function have been examined in the current research:

1. In the *linear function* the transfer value is simply the input value.
2. The *sigmoid function* maps inputs into values between 0 and 1.
3. The *sine function* transfers the trigonometric sine of the input value.

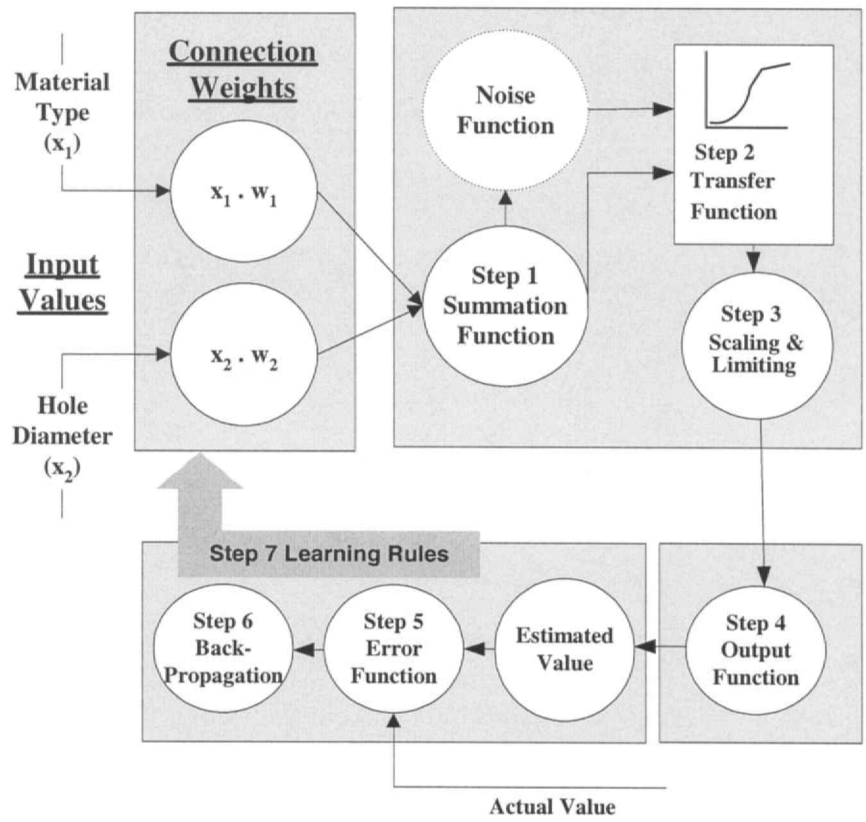

Fig. 4 Function of a processing element

4. *Tanh (hyperbolic tangent)* is similar to the sigmoid function but maps input values into the range · 1 to +1.

Prior to applying the transfer function, uniformly distributed random noise may be added. Three types of noise function have been examined in the current research:

1. In the case of *uniform noise*, a random value is applied to each PE in a layer.
2. In the case of *Gaussian noise*, again a random value is applied to PEs, but these random values are normally distributed.
3. In the case of *no noise*, no noise function is applied.

Step 3: scaling and limiting. Scaling is used to perform a linear transformation on the result of the transfer function. After scaling is applied, the transfer function is clipped to the upper and lower limits.

Step 4: output function. This provides a method of allowing PEs within a layer to compete with each other. Competition can occur to determine which PEs provide outputs to PEs in succeeding layers and/or to determine which PEs will participate in the learning or adaptation process. Three methods of determining the participation of PEs have been examined in the current research:

1. In the *direct* method there is no competition between PEs.
2. In the *select* method, if a PE has 'learned', then the output value for a single PE is set equal to the current transfer value for a single PE. If a PE has never 'learned', the output value for a single PE is set equal to zero.
3. In the *one highest* method, when processing elements compete for output, only the first winner will learn or adapt and no other PEs in the layer will adapt its weight.

Step 5: error function. Three methods have been examined in the current research for transforming the raw error, i.e. the difference between the current output and the desired output:

1. In the *standard* function no transformation takes place.
2. In the *quadratic* function the error is squared but retains its sign.
3. The *cubic* function cubes the error.

The latter two functions increase the importance of large errors. A scale is then applied to the resulting error function in order either to increase or to decrease the error associated with a particular PE. The resulting value is termed the 'current error'.

Step 6: back-propagation. The process of back-propagation consists of multiplying connection weights by a specific value and then adding the resulting value to the error field in the source PE. Depending on the network type, the back-propagated value is either the current error, the current error scaled by the derivative of the transfer function or the desired output.

Step 7: learning rules. Variable connection weights are modified according to a learning rule, of which four have been compared in the current research:

1. According to the *Hebb rule*, if both the desired output and the input are above a threshold value, then the connecting weight is incremented by a set learning rate.
2. The *perceptron rule* uses a derived amount to change the connection weights between PEs. Individual weight changes depend on actual and desired output values of PEs. A simple rule is used to determine if weight changes should be applied: if the output from an individual PE is active and is intended to be active, then do not apply the weight changes, otherwise apply the weight changes.
3. According to the *delta rule*, the error between the desired output and the actual output transformed by the derivative of the transfer function is 'back-propagated' to prior layers until the first layer is reached.
4. According to the *Ext DBD rule*, if the error is less than the previous minimum error, the weights are saved in memory as the current best. However, if the current error exceeds the minimum previous error, all connection weight values revert stochastically to the stored best set of weights in memory and, in addition, the learning rate and momentum rate are decreased to begin recovery.

If some form of competition for learning is in effect, only the weights belonging to the 'winning' processing element will be updated. Learning rules will use one or more of the learning coefficients from the learning and recall schedule. These coefficients will have different meanings depending on the particular learning rule.

5 TAGUCHI METHODOLOGY

Taguchi's methodology, often termed the 'robust design method', provides the designer with a systematic and efficient approach for conducting experimentation to determine near-optimum settings of design parameters for performance and cost [**16–20**]. The mathematical tools that form the methodology are primarily based on the statistical theory underpinning the concepts of 'design of experiments' [**20–22**]. The primary advantage of using the Taguchi methodology is its ability to minimize the number of experiments required to identify the effect, on estimating accuracy, of each of the alternative mathematical functions that make up the structure of the individual ANN processing elements. The use of the Taguchi methodology is justified since there is only one output variable, i.e. cost. The technique is not being used to determine the effect of continuous variables; i.e. alternative mathematical techniques are being compared, and the decisions required from the method are merely to identify the best and worst of these alternatives. Essentially, the Taguchi methodology enables a series of experiments to be designed that will enable identification of the optimal quality characteristics for a specific objective. In terms of product and process design, therefore, the methodology provides a systematic approach for determining the optimum configuration of design parameters for performance, quality and cost.

The basis of the Taguchi methodology is the use of orthogonal arrays (OAs) which are employed to select a range of experiments capable of efficiently studying parameter spaces that contain a large number of decision variables. The choice of an OA [**23**] depends on the number of degrees of freedom required for studying the main and interaction effects.

6 APPLICATION OF TAGUCHI METHODOLOGY

The main function of an ANN-based cost modelling tool is to generate cost estimates that are accurate and inexpensive to generate in terms of the type and amount of input data needed to train the network. The quality characteristic to be observed is 'estimating accuracy', and maximizing accuracy is the objective. The objective function to be minimized is the 'percentage average absolute error'. Also of importance within the cost estimating area is the range of error. This is measured, in the current research, using 'standard deviation of the percentage average absolute error' values.

The controllable design factors (i.e. parameters) to be considered, along with their alternative levels, are listed in Table 3. In the case of designing ANN structures, the alternative 'level' of the design factors will be represented by alternative types of mathematical function

Table 3 ANN design factors and factor levels

Processing element function	Level 1	Level 2	Level 3
Summation function	Sum	Majority	
Noise function	Uniform noise	Gaussian noise	None
Transfer function	Linear	tanh	Sine
Output function	Direct	Select	One highest
Error function	Standard	Quadratic	Cubic
Learning rules	Ext DBD	Perceptron	Delta rule

Table 4 Total number of degrees of freedom

Processing element function	Number of levels	Degrees of freedom
Mean		1
Summation function	2	1
Noise function	3	2
Transfer function	3	2
Output function	3	2
Error function	3	2
Learning rules	3	2
Total number of degrees of freedom = 12		

used within processing elements. These factor levels define the experimental region to be studied.

To select the appropriate orthogonal array to fit a specific case study, it is necessary to count the total degrees of freedom to find the minimum number of experiments that must be performed to reach a near-optimum parameter set [**20**, **22**, **24**]. One degree of freedom is associated with the overall mean regardless of the number of control factors. This is added to the degrees of freedom associated with each control factor, which is equal to one less than the number of levels. The number of degrees of freedom has been calculated in Table 4 as an example for the turning experimentation.

Therefore, it is necessary to conduct at least 12 experiments to reach a near-optimum case. This fits into Taguchi's standard L18 array, shown in Table 5. In order for an array to be a viable choice, the number of rows must at least be equal to the degrees of freedom required [**20**]. The L18 array has 17 degrees of freedom and hence can manage eight factors at three levels. Since there are only six control factors, two of the columns of the array will remain empty. Orthogonality is not lost by keeping one or more columns of an array empty [**20**, **24**].

Table 6 Number of variables experimentation

Experiment numbers	Number of variables	Number of data points	Model types tested
1 to 15	1	150, 300, 450, 600, 750	Best ANN, worst ANN, regression
16 to 30	2	150, 300, 450, 600, 750	Best ANN, worst ANN, regression
31 to 45	4	150, 300, 450, 600, 750	Best ANN, worst ANN, regression
46 to 60	6	150, 300, 450, 600, 750	Best ANN, worst ANN, regression
61 to 75	9	150, 300, 450, 600, 750	Best ANN, worst ANN, regression
76 to 90	16	150, 300, 450, 600, 750	Best ANN, worst ANN, regression

Table 7 Number of data points experimentation

Experiment numbers	Number of data points	Number of variables	Model types tested
1 to 18	150	1, 2, 4, 6, 9, 16	Best ANN, worst ANN, regression
19 to 36	300	1, 2, 4, 6, 9, 16	Best ANN, worst ANN, regression
37 to 54	450	1, 2, 4, 6, 9, 16	Best ANN, worst ANN, regression
55 to 72	600	1, 2, 4, 6, 9, 16	Best ANN, worst ANN, regression
73 to 90	750	1, 2, 4, 6, 9, 16	Best ANN, worst ANN, regression

The experiments listed in Tables 6 and 7 were carried out to determine the effects, on the accuracy of ANN-based cost estimating models, of the following:

(a) the number of variables used to construct models and
(b) the number of data samples used to develop the cost models.

Table 5 L18 orthogonal array turning cost models

Experiment numbers	Summation functions	Noise function	Transfer function	Output functions	Error function	Learning rules
1	Sum	Uniform noise	Linear	Direct	Standard	Ext DBD
2	Sum	Uniform noise	tanh	Select	Quadratic	Perceptron
3	Sum	Uniform noise	Sine	One highest	Cubic	Delta rule
4	Sum	Gaussian noise	Linear	Direct	Quadratic	Delta rule
5	Sum	Gaussian noise	tanh	Select	Cubic	Ext DBD
6	Sum	Gaussian noise	Sine	One highest	Standard	Perceptron
7	Sum	None	Linear	Select	Standard	Delta rule
8	Sum	None	tanh	One highest	Quadratic	Ext DBD
9	Sum	None	Sine	Direct	Cubic	Perceptron
10	Majority	Uniform noise	Linear	One highest	Cubic	Ext DBD
11	Majority	Uniform noise	tanh	Direct	Standard	Perceptron
12	Majority	Uniform noise	Sine	Select	Quadratic	Delta rule
13	Majority	Gaussian noise	Linear	Select	Cubic	Perceptron
14	Majority	Gaussian noise	tanh	One highest	Standard	Delta rule
15	Majority	Gaussian noise	Sine	Direct	Quadratic	Ext DBD
16	Majority	None	Linear	One highest	Quadratic	Perceptron
17	Majority	None	tanh	Direct	Cubic	Delta rule
18	Majority	None	Sine	Select	Standard	Ext DBD

As a comparison, models were also developed using the LINEST regression analysis function within a Microsoft Excel spreadsheet [**25**].

7 RESULTS AND DISCUSSION

In order to identify the relative effect on cost modelling estimating accuracy of the number of variables and number of data points, the experiments listed in Tables 6 and 7 were carried out, and the results are shown in Figs 5 to 10. From these figures the following effects can be observed. Increasing the number of variables increases the estimating accuracy of the resulting models both in terms of the percentage average absolute error and the standard deviation of percentage average absolute error:

1. This effect becomes more pronounced as the number of data points used to construct the model decreases. However, both increasing numbers of variables and increasing numbers of data points lead to improvements in estimating accuracy.
2. The estimating accuracy of the regression-based models, in general, increases with increasing numbers of variables, but there appears to be no marked increase when the number of data points used to construct models increases.
3. In all cases the estimating accuracy obtained when using the ‘worst’ network structure is poor when compared with that of the ‘best’ structure and in many cases the regression models have similar results. In addition, the ability of such models to estimate

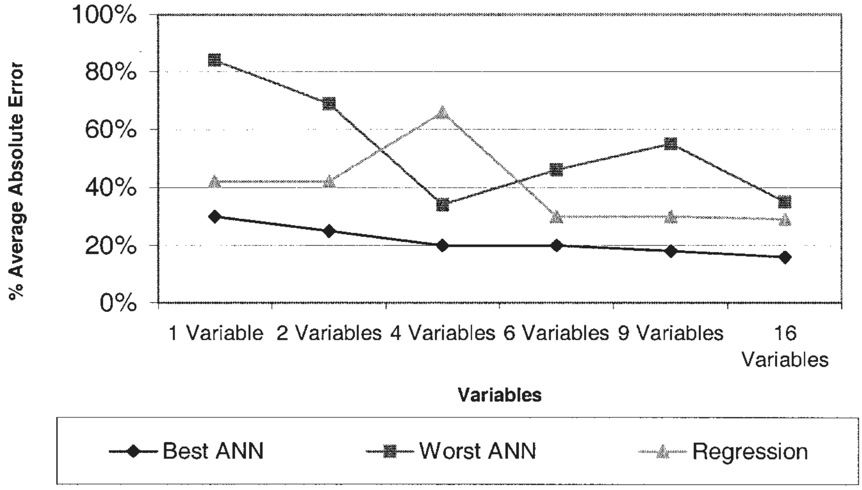

Fig. 5 Total of 150 data points with percentage average absolute error

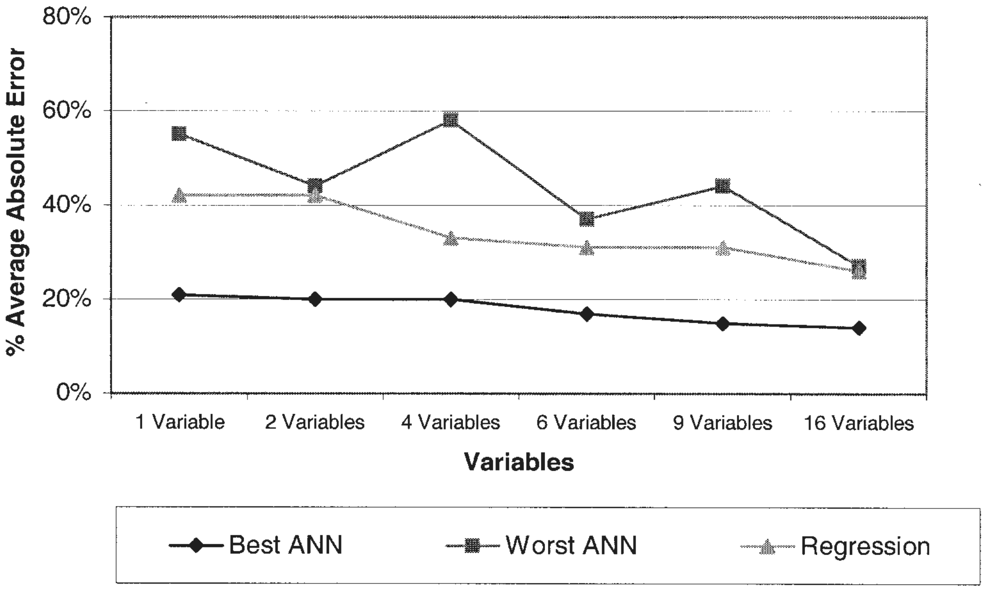

Fig. 6 Total of 450 data points with percentage average absolute error

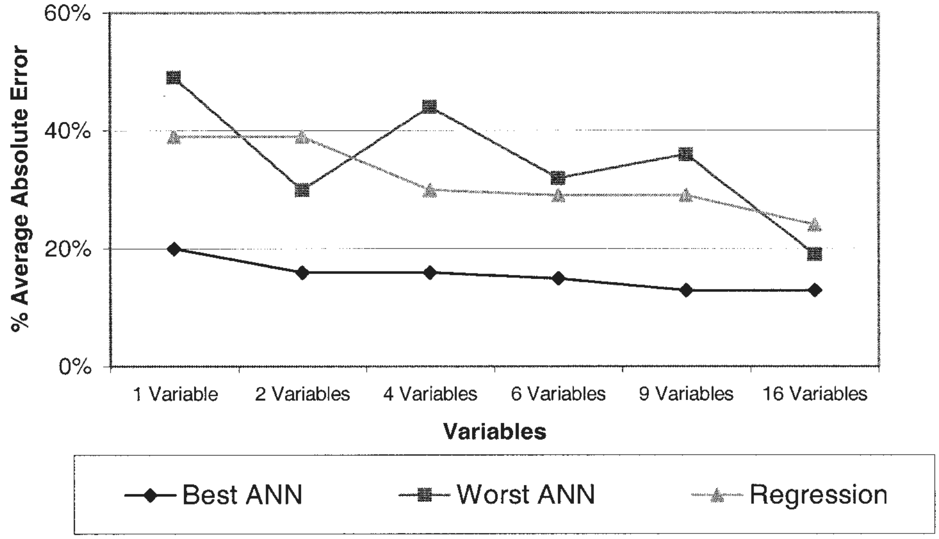

Fig. 7 Total of 750 data points with percentage average absolute error

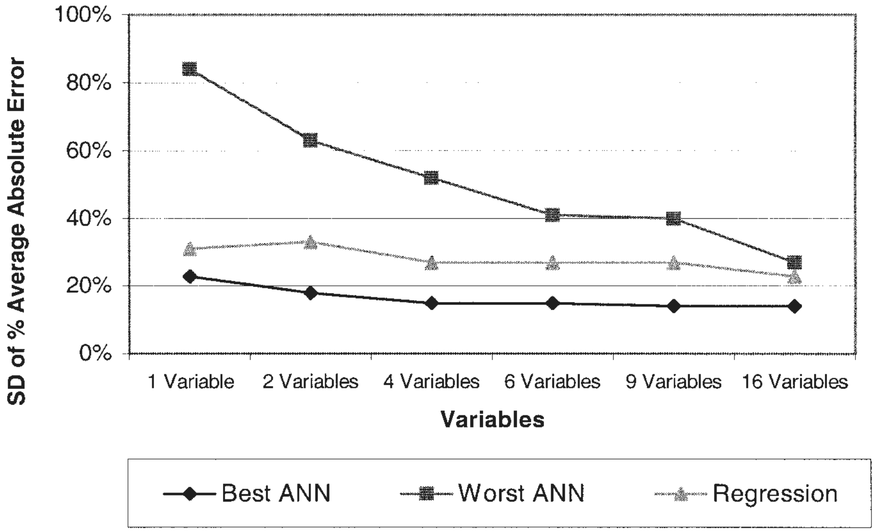

Fig. 8 Total of 150 data points with standard deviation of percentage average absolute error

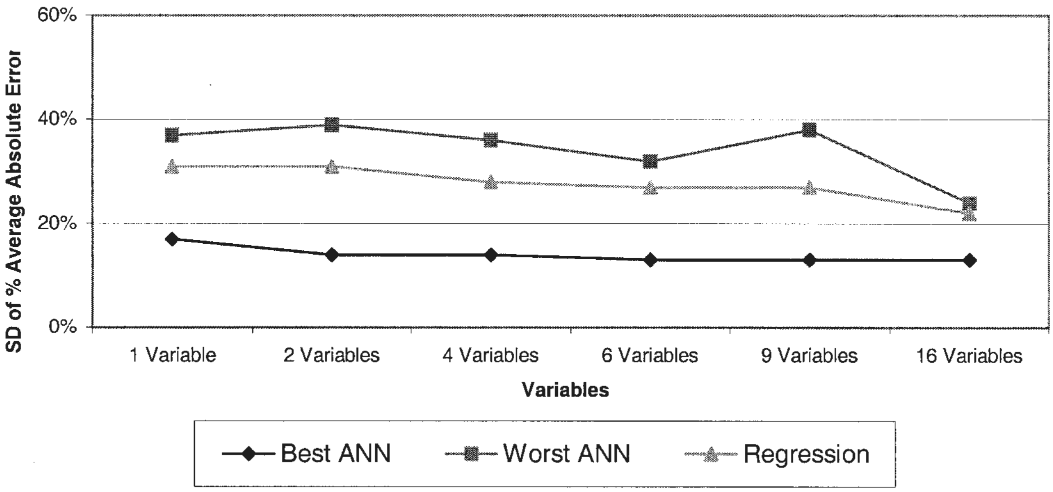

Fig. 9 Total of 450 data points with standard deviation of percentage average absolute error

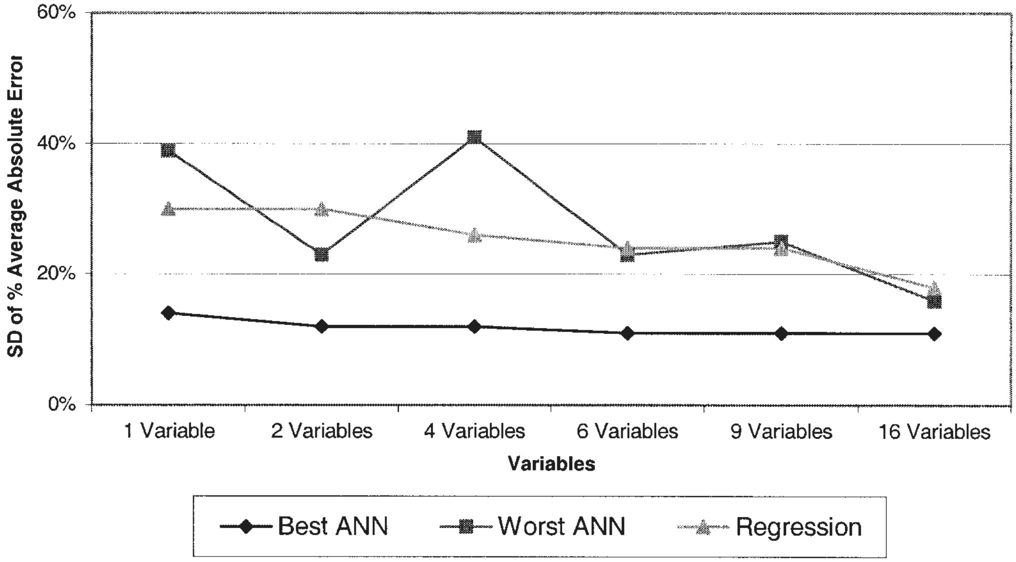

Fig. 10 Total of 750 data points with standard deviation of percentage average absolute error

accurate costs is erratic; i.e. in many cases there is no relationship between estimating accuracy, number of variables and number of data points.

4. The greater the number of variables used, the less will be the effect on estimating accuracy of increasing the number of data points.

8 CONCLUSIONS

Experiments have been undertaken to identify the influence, on the estimating accuracy of the resulting models, of varying the amount of data used to develop the model and varying the number of variables within the model. As would be expected, increasing the amount of data used to develop the model and increasing the number of variables within the model in all cases leads to an overall increase in the estimating accuracy of the resulting models. These experiments did, however, reveal that fewer variables and lower amounts of cost data are required to achieve a specific level of estimating accuracy than when using regression analysis. The results present in this paper also show that the consequences of choosing an appropriate network structure can be identified. Little evidence is available to suggest that specific ANN structures could be used over a range of applications. Hence, the Taguchi approach adopted in this research represents an efficient method for determining appropriate ANN structures since it allows the number of experiments required for identification to be minimized together with the costs of developing the resulting models.

ACKNOWLEDGEMENT

The authors wish to acknowledge the funding received from the Engineering and Physical Sciences Research Council (EPSRC) through the COSTMOD project (Ref. GR/M 58818).

REFERENCES

1 **Ostwald, P.** *Cost Estimating*, 4th edition, 1988 (Prentice-Hall, Englewood Cliffs, New Jersey).
2 **Cunningham, J. J.** and **Dixon, J. R.** Designing with features: the origin of features. *Trans. ASME*, *Computers in Engng*, 1988, 237–243.
3 **Chang, T.-C.** *Expert Process Planning for Manufacturing*, 1990 (Addison-Wesley, Reading, Massachusetts).
4 **Wild, R.** *Essentials of Production and Operations Management*, 2nd edition, 1985 (Holt, Rinehart and Winston, London).
5 **Venkatachalam, A. R.** A knowledge-based approach to design for manufacturability. PhD dissertation, Alabama University, Alabama, 1990.
6 **Boothroyd, G.** and **Reynolds, C.** Approximate cost estimates for typical turned parts. *J. Mfg Systems*, 1989, **8**(3), 185–193.
7 **Smith, A. E.** and **Mason, A. K.** Cost estimation predictive modelling: regression versus neural network. *J. Engng Econ.*, 1997, **42**(2), 137–161.
8 **McCulloch, W. S.** and **Pitts, W. A.** A logical calculus of the ideas imminent in nervous activity. *Bull. Math. Biophys.*, 1943, **5**, 115–133.
9 **Bode, J.** Neural networks for cost estimation. *Cost Engng*, 1998, **40**(1), 25–31.
10 **De la Garza, J. M.** and **Rouhana, K. G.** Neural networks versus parameter-based applications in cost engineering. *J. Cost Engng*, February 1995, **37**(2), 14–18.
11 **Shtub, A.** and **Zimerman, Y.** Neural-network-based approach for estimating the cost of assembly systems. *Int. J. Prod. Econ.*, September 1993, **32**(2), 189–207.
12 **Bode, J., Ren, S. J.** and **Shi, Z. Z.** Application of 3-layer perceptrons to cost estimation. In Proceedings of the IEEE International Conference on *Neural Networks*, 1995, **4**(4), 1749–1754.

13 **Wang, Q.** and **Stockton, D. J.** Process time estimating using neural networks. In Proceedings of the 15th International Conference on *Computer-Aided Production Engineering*, Durham University, April 1999, pp. 201–206.

14 *Advanced Reference Guide for Professional II/PLUS*, 1996 (The NeuralWare Inc., Pittsburgh, Pennsylvania).

15 **Wang, Q., Stockton, D. J.** and **Baguley, P.** Using neural networks in cost model development process. In Proceedings of the Sixteenth National Conference on *Manufacturing Research*, University of East London, September 2000, pp. 59–63.

16 **Bendell, A.** Introduction to Taguchi methodology. In Proceedings of the European Conference on *Taguchi Methods*, 1988, pp. 1–14.

17 **Kackar, R. N.** Taguchi's quality philosophy: analysis and commentary. *Qual. Prog.*, December 1986, 21–29.

18 **Logothetis, N.** and **Salmon, J. P.** Tolerance design and analysis of audio-circuits. In Proceedings of the European Conference on *Taguchi Methods*, 1988, pp. 161–175.

19 **Meisl, C. J.** Parametric cost analysis in the TQM environment. In 12th Annual Conference of International Society of Parametric Analysts, San Diego, California, 1990.

20 **Phadke, S. M.** *Quality Engineering Using Robust Design*, 1989 (Prentice-Hall, Englewood Cliffs, New Jersey).

21 **Gunter, B.** A perspective on the Taguchi methods. *Qual. Prog.*, June 1987, 44–51.

22 **Wille, R.** Landing gear weight optimisation using Taguchi analysis. In the 49th Annual International Conference of Society of Allied Weight Engineers, Chandler, Arkansas, 1990.

23 **Ross, P. J.** *Taguchi Techniques for Quality Engineering: Loss Function, Orthogonal Experiments, Parameter and Tolerance Design*, 1988 (McGraw-Hill, New York).

24 *Taguchi Methods: Implementation Manual*, 1989 (American Supplier Institute, Dearborn, Michigan).

25 **Halberg, B., Kinkoph, S.** and **Ray, B.** *Special Edition Using Microsoft Excel97*, 1997 (Que Corporation, Carmel, Indiana).