PRODUCT COST MANAGEMENT STRUCTURES: A REVIEW AND NEURAL NETWORK MODELLING

Jha P. P., Glassey, J., Montague, G.A. and Mohan, P[1]
School of Chemical Engineering and Advanced Materials,
Merz Court, University of Newcastle upon Tyne, NE1 7RU, U.K.
phone: (+44) 191 2227275
email: pushkarjha@hotmail.com
[1]Eli Lilly and Co., Drop Code TL 32, Tippecanoe Labs, Post Box 685, Lafayette, Indiana -47902, U.S.A
Address for Correspondence:
Pushkar. P. Jha
4045A
Cass Business School
106 Bunhill Row
London EC1Y 8TZ
Tel: +44(0)20 7040 8727
email: pushkarjha@hotmail.com

ABSTRACT

This paper reviews the growth of approaches in product costing and draws synergies with information management and resource planning systems, to investigate potential application of state of the art modelling techniques of neural networks. Increasing demands on costing systems to serve multiple decision-making objectives, have made it essential to use better techniques for analysis of available data. This need is highlighted in the paper. The approach of neural networks, which have several analogous facets to complement and aid the information demands of modern product costing, Enterprise Resource Planning (ERP) structures and the dominant-computing environment (for information management in the object oriented paradigm) form the domain for investigation. Simulated data is used in neural network applications across activities that consume resources and deliver products, to generate information for monitoring and control decisions. The results in application for feature extraction and variation detection and their implications are presented in the paper.

Keywords: Activity Based Costing (ABC)**,** Enterprise Resource Planning (ERP), Neural Networks, Self-Organising Maps, Hopfield Networks and Object Orientation.

INTRODUCTION

Product costing information has undergone numerous changes in the last few decades towards an increasingly strategic nature. In the data and process standardisation in enterprises this role is augmented by the appreciation of the aspect of process and inter-linkages, which has synergies with the developments in costing approaches. Product costing structures are looking at activities as driving the cost and as the vital interface between resources (inputs) and products (outputs) (Cooper & Kaplan 1991). This issue is reviewed in the present contribution covering strategic approaches in product costing, including the object oriented paradigm that strives to improve product costing functionality and neural networks. Neural networks are discussed as a technique that can prove useful in data modelling for information extraction. The advantages over conventional techniques, especially those of a non-parameterised approach and the comprehensiveness of this approach due to the wide range of properties and structures that can be applied on data, are also highlighted in this paper.

The linkages across sub-systems within enterprises are appreciative of the influence each sub-domain has on the other. The issue of modelling to accurately estimate and capture this influence however, is subject to constant improvement. The latest developments in modelling techniques were in the area of the multidimensional facets of systems. The growth in modelling techniques towards more comprehensive nature and the increasing use of intelligent systems to escape the limitations imposed by conventional statistics are indicators of this development.

The basic issue in product costing is that of 'estimation' and 'detection' essentially for performance evaluation and control. These together with other multiple decision objectives are fed into strategy and planning. However, these areas have competing information demands that draw from various variable sets in product cost data. The accuracy, comprehensiveness and scope of techniques in analysis are thus critical.

The applications for feature extraction and detection in this paper were carried out on simulated data using MATLAB. MATLAB handles multidimensionality found in the increasingly wider perspectives in costing approaches in a straightforward manner. Expressions of external factors can be attached to approximate closely the real problem domain/ the real system within matrices that represent a 'natural' description of certain complex relationships. These relationships are always of concern in strategic planning, a key objective for which enterprises continuously seek better modelling solutions (Sherwood 1983).

COST INFORMATION STRUCTURES

The costing systems are expected to provide information for numerous and sometimes competing decision objectives. Advances in product costing to improve its characteristics as a strategic tool have made costing more open to implementation of uncertainty (influence of factors outside the firm's control), non-linear influences and relevance of a wider temporal approach (costing is no longer limited to only the short run). "Marketing, administrative, sales, distribution and deployment 'events' are considered to be cost drivers in the modern financial system" (Dekker 2000) showing a growing complexity of processes and need for accuracy and comprehensive approach in associating costs to processes. Also the modern day product cycle is characterised by products maturing over an increasingly shorter time frame and advantages due to innovations diminish quickly. The strategy implication of these characteristics is that of having a responsive strategy which responds quickly to changes in the dynamic market environment. The emphasis is on the process for both seeking competitive advantage through product and production strategies, and developing a response to a change in the factors that make a successful product. The ERP systems have provided for process and data standardisation to aid this but generation of information out of these systems is subject to modelling techniques. Thus the scope of neural networks in this regard is discussed in this paper after reviewing the growth in dimensions for product costing.

It is essential to identify what a product is actually costing if business decisions are to be well informed. The cost allocation process is essentially a two-stage process as in the Activity Based Costing approach. In this two-stage process separate cost centres or pools are created for each major activity. This is further traced using suitable measures or 'cost driver rates' to individual products. It is a manifestation of the fact that activities consume resources and product consumes activities (Cooper & Kaplan 1991).

Activities consume a variety of resources that are not always in relative or perfect proportion and the same applies for products consuming these activities. The trade-off problem between complexity (too many cost drivers) and loss of accuracy (too few cost drivers) is addressed in the comprehensive data and process standardisation in ERP packages such as SAP. Pragmatic product costing, an extension of ABC, tries to arrive at resource consumption by the processing time consumed for each activity. It works on the unit price of standard time. It is said to allow faster cost estimation as compared to process flows in standard ABC (Jiao & Tseng (1999). However data and process standardisation, as in SAP production planning modules, seem to address the needs of this approach and also any benefits it provides compared to the basic approach of ABC.

Influences that are not covered in the 'certainty' of assumptions of traditional wisdom in costing have become more and more relevant over the years. With a growth in multiproduct nature of businesses, strategic approaches like 'Target costing' complement the developments recognising the activity emphasis. Here, the recognition of external market factors to plan costs also induces a 'learning effect' over time taking costing out of temporal bounds and giving it more of a strategic nature.

In target costing one works from the selling price downwards to feasible cost levels. Target costing is an approach made popular by the 'Japanese business strategy thinking'. Until very recently the focus was limited to standard costing. The purpose of standard costing is on deviations between actual and expected results for a given manufacturing process. On the other hand, in target costing the target costs are set during the design phase and the reduction in standard costs is strived for to achieve the wider goal of target cost levels. Additionally companies realise the best sources of cost savings before the product reaches the production stage (Lagagne & Discenza (1993). Target costing is also said to improve motivation for performance, a manifestation of the modern view of integrating influences of management segments.

The extension of costing during the customer revolution of the 1980s saw the reconstruction of customer as an investment, the customer being 'chosen to invest in' for returns. Value to customer became more important in contrast to only the cost to customer in acquiring the product offered (Boyce 2000). This was actually always there in any business strategy but the involvement of accounting practise in customer valuation presents another strategy dimension costing is supposed to provide information for. These objectives are ever increasing, as the business environment becomes more and more competitive. A competitive product must address factors such as cost, performance, aesthetics, schedule or time-to-market and quality as value to the customer. Today, parametric estimating is usually applied to large systems that describe these factors (Crow 1997).

PRODUCT COST MODELLING SOFTWARE

Most of the integrated software for modelling of product costs is subject to a very specific framework, which is developed to suit the needs of a particular industry. The basic data interface is always a step structure 'or' hierarchy of levels that have costs at the base level. These are then linked to the activity structure and cost pools to identify the cost drivers and subsequently allocation is performed. Most models first begin by assuming the 'nature of these levels' based on an 'industry type' and then provide a cost estimation and forecasting technique. The data in these software packages is grouped across spreadsheets like for example in the four structures of Resource, Pool, Activity and Report (Reed & Goswami 1998).

An example of this, but also the recognition of the need to extend modelling techniques, is the ESA cost modelling software (ECOM) developed for cost estimation in the area for space projects. ECOM is able to escalate cost data to any given economic conditions (Fatelnig 1997). To prepare reliable price estimates, the cost analyst ideally needs to know the Product Tree, the Work breakdown Structure and the Work-Package descriptions. The cost estimation uses a mix of approaches including resource consumption in work packages, analogy, using historical estimates, standard cost estimation functions and to some extent rules of thumb, essentially similar to use of expert knowledge or 'the good guess' (Fatelnig 1997).

Very little application is reported in the field of product cost modelling using artificial intelligence and knowledge-based approaches. An example of this approach is seen in the building industry. Combine2 within the JOULE programme of the European Commission developed a prototype of a computer based integrated building design system. Within this, there exists a costing tool that presents a model of life cycle costing of building design in different phases. The prototype tool for one of the phases of design has been programmed with the neural network application development software. All the cost effects have been cased by means of real building information or knowledge collected and calculated from real cases (Lassila et al 1994).

ERP AND THE EMPHASIS

ERP is popular in businesses of today due to increasing process efficiencies. "ERPs primarily automate the transaction processing requirements of financial, distribution and manufacturing business processes" (Luettegen 2000). Thus enterprise Information and management packages like SAP have become widely applied. All parts of the business (company, business areas, controlling area, operating concern) are linked together with common data access and real time information flow across management divisions for a better accumulation of and return on information. The pre-requisite to SAP is business process re-engineering, which implies a thorough analysis and coding of the Sales, Production, Purchasing and Finance and accounting events in a business process and how each action influences processes in the other segments (Enterprise management Technology 2000). This is essentially a well-defined networking to link business nodes by appropriate flows. The emphasis in enterprise resource management paradigms is analogous to the emphasis in costing i.e., 'process' and 'tailored to enterprise need'. Analogies also exist between these paradigms and the network based technique discussed later in the paper. These include networking, real time flows in information systems and also the growing need to eliminate the limiting assumptions in response to an increasing learning interface with environment.

SAP AG is the major ERP package vendor for the Fortune 500 companies. SAP's first two products operated on mainframe hardware; R/1 was batch oriented, then in 1981 it was replaced by an online system called R/2. SAP has dominated the market with the help of its improved R/3 system in the last decade. The role of object oriented structure in enhancing functionality can be seen in the fact that R/3 currently contains over 1000 business processes that 'have to be selected from' for applications

The object-oriented approach can reduce the complexity since in such design approach there is some sort of classification, that associates the methods operating on data structures and the system draws relationships across objects. Functionality of different classes can be combined and objects stimulate each other to guide the system's functionality as a whole. Modifications are object based and not system based, allowing easy adaptability in adding new facets to the system. One can associate data and processes that operate on it to each object. Systems, based on an object-oriented approach, are easier to understand because they are more closely related to reality. They have a small semantic gap between the model and reality. Such systems are more flexible because they can be broken up into many small and independent entities (objects) that can be refined and evolved independently. Because of this independence they can be used later in other similar designs, improving 'code reuse' (Moreira & Clark 1996). This relates to the efforts to make SAP more adaptable/generic than it currently is and to improve its functionality.

To illustrate this point, a cost object can hold information such as the name of the cost centre, the resource consumed, consumption rate, activity type, activity type prices e.t.c. (depending on how the overall classification is set up). It will have a set of operations to modify this data through operations of monitoring the activity and/or intimating uncalled for fluctuations in resource consumption with activity level. Of course, it will also be connected to other cost objects, work in progress and inventory objects in the system. Production order can be taken as the cost object, in effect it could be a class if different types of production orders were the reference point, (an object can be viewed in different ways depending upon the underlying relationships that constitute the structure).

The data used to value manufacturing activity relies on master data in both the logistics and accounting modules. The Product costing sub module lies in the SAP set up for the Production Planning Module where planning and control is carried out for manufacturing activities. An illustration of this is shown in Figure 1. The figure gives an outline of the Product costing module to illustrate the flows in a micro section of SAP. Clearly an object-oriented structure would resolve complexity by placing emphasis as much on 'objects' as on the flows between them. Besides, the classification structure and associated properties allow for better automation and functionality in processing information

THE INTELLIGENT AGENT CONCEPT AND DATA MODELLING SOLUTIONS

Accurate information, accounting for anomalies in data structures and processing flexibility are always desirable. The intelligent agent concept extends the object oriented paradigm in to address there. The two distinct advantages in this extension may be noted as:

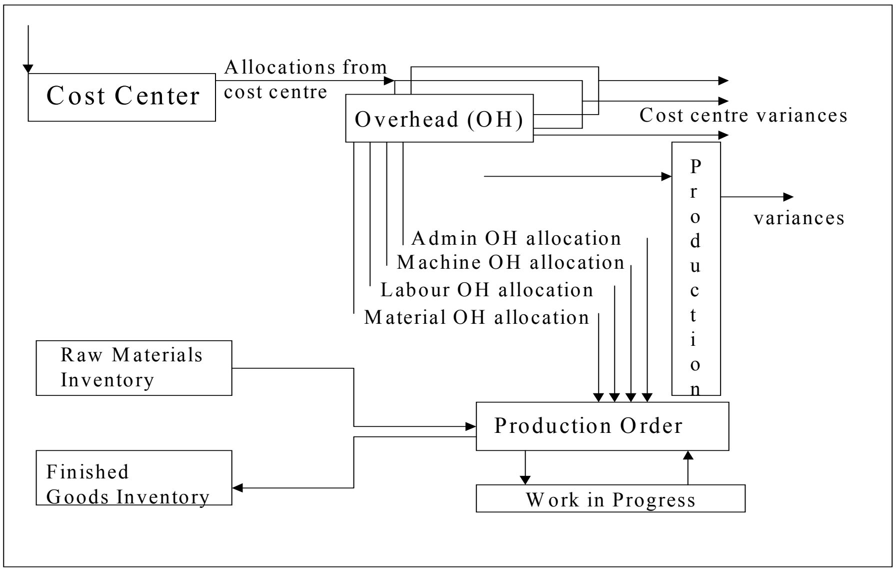
❑ Creation of smart business solutions, enabled but not necessarily driven by object technology. In these convergence to solutions despite inconsistency and anomalies in data or in case of incomplete knowledge is possible.

❑ Ability to extend the concept of information generation to knowledge and learning.

(Farhoodi & Fingar 1997).

The synergies can be summarised in Figure2

Figure 1: SAP Product Costing Sub-Module[2]



Adapted (SAP 1999)

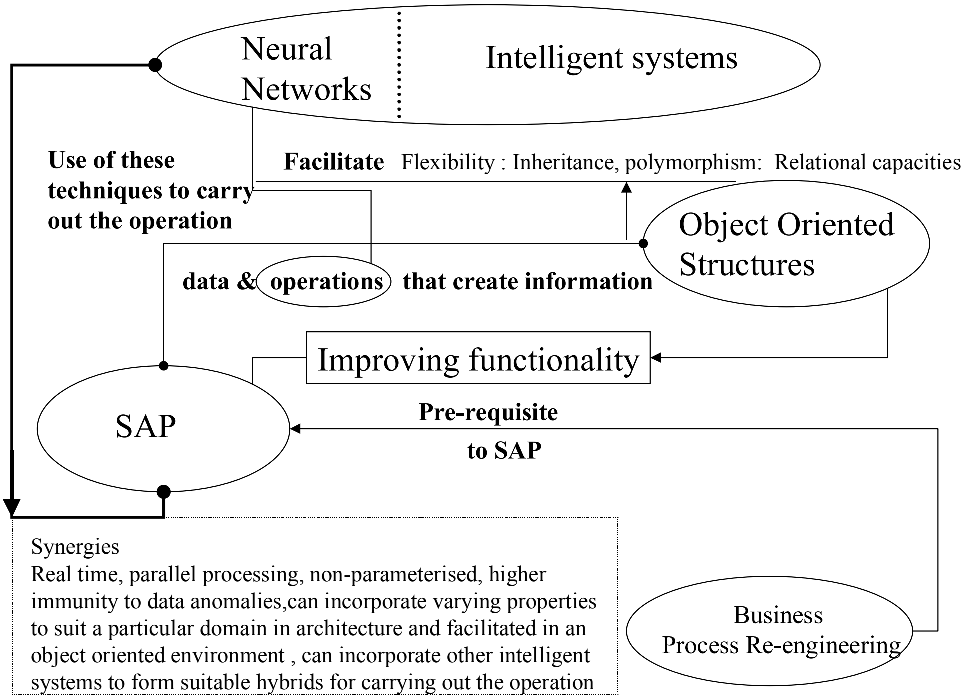
NEURAL NETWORKS AN OVERVIEW

In management science and operations research, neural networks have been applied to marketing, investment, banking, stock price prediction, property pricing, project management and exchange rate prediction besides other areas (Sarle et al 1997). Comparisons of neural network performance particularly with linear regression have been performed during the inception of interest in this domain for a wider investigation of product designs, process routings and cost data (Zhang et al 1996) with results that are in favour of neural networks.

In finance neural networks have been applied to the area of firm failure or bankruptcy prediction, where the emphasis is essentially on feature extraction. In solving classification problems for prediction of bankruptcy there have been results to suggest that though neural networks perform better than statistical models, their dependence on the nature of training data makes for an argument for their application in conjunction with statistical models (Jain & Nag 1997). This has already been tested in applications of non-linear partial least square regression using a neural network structure for other areas. This approach is being investigated in this research with applications on product cost data. Other comparison studies have also been performed where these techniques were compared with the results from statistical techniques (Zhang et al 1996). Depending upon the data the results were mixed but neural networks performed as well or better than the statistical techniques in comparison. The emphasis is being placed on the ability of networks to identify patterns data (non-parameterised) which allows a better appreciation of external market factors. When these are taken

[2] In following figure profitability analysis is not included which would then link into the other modules and further attachments would flow across the entire system.

into perspective neural networks have been shown to have very good modelling potential. These results are for non-product cost financial data like market index forecasting (Walczak 1999). P*r*oduct costs also involve this facet given the strategic approach of target costing (working from the selling price downwards) where market forces and expectations is a vital ingredient in decisions. This 'external' influence is also expressed in the 'valuation of the customer' in product costing decisions.
Examples of work in cost system design that make use of concepts similar to intelligent systems are given in the discussion on cost modelling software in this paper (Fatelnig 1997). Work on exploring the use of neural networks in product costing has been largely limited to research apart from some industry specific applications that have been mentioned earlier. Feature extraction in the manufacturing process for costs has been explored in research using neural networks. This is for estimation during the design phase in keeping with the key idea pursued in Life cycle costs, that of 'design to cost' so as to reduce unnecessary costs later on in the product life cycle (Ou-Yang & Lin 1997).
There are many kinds of neural networks. The developments in this modelling technique are marked by perpetual evolution. [24]. All the types vary in terms of the training methods they may use and further in terms of the learning law used, which comprises of a variety of characteristics (type of activation and learning rates).


Neural Networks
Intelligent systems
Use of these techniques to carry out the operation
Facilitate
Flexibility : Inheritance, polymorphism: Relational capacities
Object Oriented Structures
data & operations that create information
Improving functionality
SAP
Pre-requisite to SAP
Business Process Re-engineering

Synergies
Real time, parallel processing, non-parameterised, higher immunity to data anomalies,can incorporate varying properties to suit a particular domain in architecture and facilitated in an object oriented environment , can incorporate other intelligent systems to form suitable hybrids for carrying out the operation

Figure 2: Synergies

For any application of NNs the following properties need to be defined:

- The topology (connectivity options)
- The learning rate [learning made possible through attaching a local memory to the processing element so that results of previous runs can be used to modify weights. Weights are adaptive coefficients that determine the intensity of the input signal *(Nelson, Illingworth,1993)]*
- The approach to learning i.e. training
- Choices available in neuron functions of combination and then further using this as an activation function to allow outputs to vary with time before passing them on to the threshold or transfer function (a non-linear function, typically a sigmoid).

These properties are not separate compartments of characteristics but particular sets of these are 'complementary in selection of a model'. But what influences the selection? Essentially it is the nature of the objective function, the nature of the variable set. Contemporary research looks at how the use of neural networks can better capture the variability in costs i.e., feature extraction and detection. In their research, Rohde and Otzen of the Copenhagen Business School (1999), have discussed the ABC approach and neural networks in conjunction. They have used terms of 'ABC networks' and 'Full cost networks' essentially based on the kind of input data used. The importance of the company's cost allocation policy has been articulated and results indicated that such full cost networks are better at generalising and predicting than the ABC networks. These results seem to be focussed on 'calculatory precision' and on 'data issues' in application of networks.

In order to estimate of costs it is essential to accumulate activity information over the process. However, the information needs for decision-making needs may be very different to those for control. Qualitative information viewed from processing the available data in a certain fashion may be required (Borjesson 1994). For control purposes the variability in activity measures over time in a relative space and detection of unqualified deviations is important. It is this issue that is illustrated in the applications later on the present contribution. The better handling of feature extraction in neural networks compared to other techniques is established in numerous studies. The processing structures in neural networks throw some light on the reasons behind this besides the object oriented synergies that attach them well with the computational environment for information management and the outlook in SAP.

A very basic topology and learning algorithm-based classification of neural networks is as provided in table I.

Table I: Basic Classification of neural networks

Learning algorithm:	Topology:
Supervised: The target values are given to the NN during training so that it can adjust weights (synaptic weights are 'knowledge' of the NN) to try and match outputs to these (performance parameterised by a tolerance level within which the output may be considered as good). There are two forms to this: Autoassociative, where the target values are the same as inputs Heteroassociative where the targets are different from inputs. Neural networks can accept either categorical or quantitative data, supervised learning with quantitative data is called 'regression' (as estimating the best fit). A variant of supervised learning is 'reinforcement learning'. Here the total generalised error in the output is spread out over each incoming signal and the eligibility of update of a synapse is based on a probability relationship (of the output being correct). On the other hand pure supervised learning does not depend upon the updates in the last levels activation, it simply takes into account a specific synapse and does not take into account the global estimate of error. Unsupervised: The NN is not provided with target values to match. These essentially perform data compression to reduce dimensionality.	Feedforward: Here the connections between nodes are not cyclical. Due to this they provide a result faster relative to the feedback topology. Feedback: Here there are cycles in connections i.e. the network iterates for a relatively long time as compared to the feedforward set up before providing a response (past output included as current input).
	A form of learning that can be either supervised or unsupervised (implementable either through feedforward or feedback structures) is competitive learning. Here there is a random specialisation, in the order of which output neuron produces the 'closest signal' to the target.

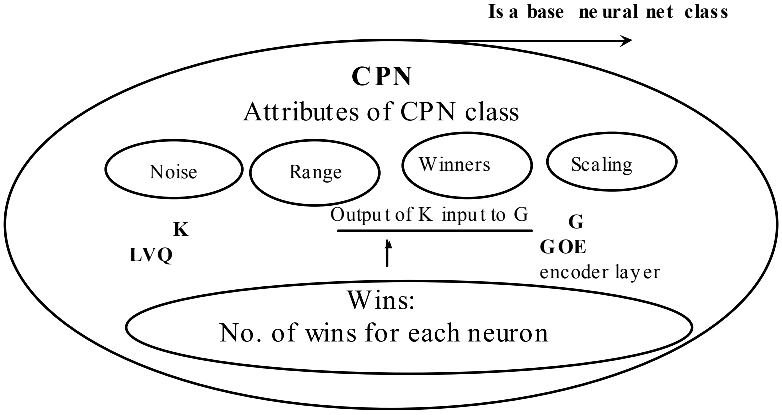
To hypothesise the applicability, potential and synergy of neural networks one may look at the Hebbian learning rule for neural networks. This postulates that if two neurons were very active at the same time the strength of connection between them would increase. This would imply that activities consuming resources could be mapped in a relative space. The 'instar rule' is a modification of the Hebbian rule(to recognise an input) and it may be used in competitive networks of the SOM (self-organising maps: 'feature extraction') type. Here weights of the winning neurons and neighbours both are updated (Demuth & Beale 1996). In other words activities consuming the resource would be 'weighted' in relative intensities. Given more elaborate data, as in SAP systems, comprehensive mapping of activities consuming resources could be performed for various decision objectives.

The 'data centered design' of neural networks is in line with the Object Oriented approach (OO) that is popular in computing and automation today. The abstractness in classes and optimisation of methods that operate on the data in terms of their frequency of use has an analogy with generalisation and self-pruning respectively, in neural networks. The importance of the class interface and inheritance in such systems has synergies with the classification emphasis in SAP, provides designing simplifications and also easier update as inherent in the object oriented approach.

The extension of classes or their specialisation using the concept of inheritance and the property of polymorphism[3] are very analogous to the learning attributes and parallel structure of the neural networks. OOs relate to the synergies of association, hierarchy, organisation and search across different intelligent systems like neural networks, expert systems and genetic algorithms.

For example, an OO structure of the Counterpropagation Neural Network (CPN, Figure 3) indicates how a neural network system can be looked at under the object oriented system. Here a particular property/ attribute to constitute the neural network structure can be called a data centered design where integration does not close options but allows for an 'associative classification' across objects and methods that operate on them.

Figure 3: Hierarchy and class in OO systems for CPNs



*LVQ and GOE mentioned in the figure refer to different training methods.

(Blum 1992)

'The object-oriented paradigm has given a new dimension to integration technologies. It is leading to a faster and more reliable componentisation of SAP (Runge 2000). Clearly having techniques of analysis that relate well with the generic nature of such systems shall ensure better efficiency.

FEATURE EXTRACTION USING SELF-ORGANISING MAPS

Self-organising maps (SOM) are auto-associative, topological vector maps, which have possibly the largest applications among the basic neural network model types (Backpropagation, Hopfield, Boltzmann machine, etc). These can be used in applications of classification and feature extraction, image processing and optimisation. It is the feature extraction property of SOM, which will be our area of interest in this section for use in analysis for product costing. SOM converts non-linear statistical relationships between high-dimensional data items into simple relationships expressed geometrically in a 'relative mapping'. Information is compressed preserving the most important topological relationships of the data items, in kind of abstract classes.

Given more elaborate data, as in SAP systems, a comprehensive mapping of activities consuming resources for temporal comparisons and visualisation of the relative spread of activity intensities can be made available through use of SOMs. The application of feature extraction to product cost can be much wider than the activity analysis covered by this contribution. Feature extraction based on the illustration here can be easily hypothesised to hold potential for the 'design to cost' concept and 'value analysis', both attributes of the discussed strategic role of costing in the businesses of today.

Self-Organising Networks are different from a classifier system where boundaries between known categories are adjusted (like in Learning Vector Quantizer Networks) by not relating to classes but unknown clusters (unsupervised). It is a mix of competitive learning and neighbourhood influences that constitute the SOM.

Before interpreting the application results, there is a need to discuss further the variety and the property selection issues in SOM and the underlying process. Among other issues the number of neurons used for feature extraction and also the neighbourhood distance function to be used have to be selected. The SOM grid could theoretically cater for any number of samples and be as large as required. Setting the number of nodes to approximately the number of input samples is generally acceptable when data sets are relatively small. The issues of accuracy for SOM show that neural networks are influenced by distributional assumptions, as when the map is to be used for new data, accuracy is assured only when the new data approximates the same distribution as the data set used in learning. Distance functions used can also vary; layer distance functions are used to find distances between neurons in a layer. A combination of this with Euclidean distance weight function (applying weights to input) can also be used.

During the training phase, the SOM strives to approximate the density of the data. The reference vectors in the network weights move to the areas where the density of the input data is high. Eventually, only few weight vectors lie in areas where the input data is sparse. In the learning process, sample input vectors selected at random are presented to the SOM. The weight vectors are updated based on the similarity between the two using the distance function. As the learning converges the neighbourhood radius (space) also reduces. The precision of the mapping is derived from the ‘average quantization error’, a measure of the accuracy of neuron responses to the given data set. The second measure of precision is that of topology preservation. This deals with the map structure and essentially how much the process has to twist the data set for the mapping (Hodju & Halme 1999).

Case study

An activity record set was used in terms of activity measures. Seven such activities driving the cost over a large set of multivariate columns were simulated within the MATLAB environment.

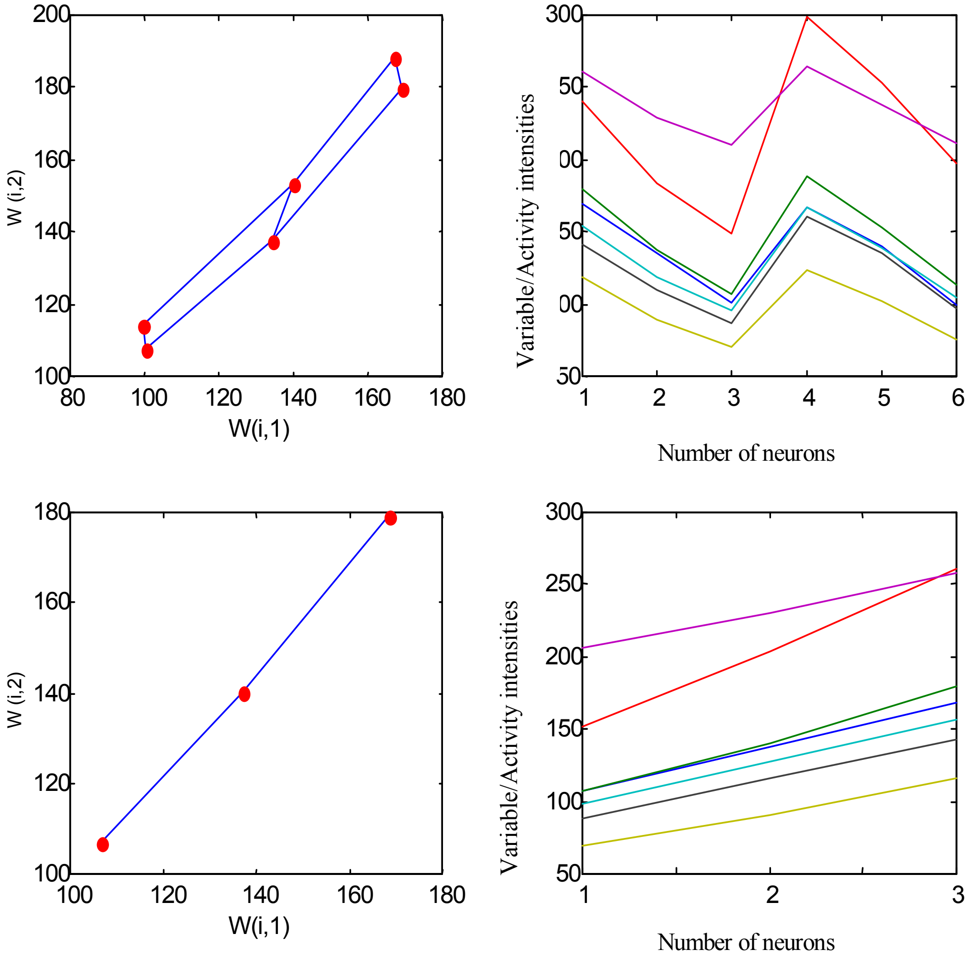
The rows/drivers (input vectors) were as follows:

a: Set up hours; **b:** Number of set ups; **c:** Material handling hours; **d:** Number of purchase orders; **e:** Number of hours in delivery runs; **f:** Maintenance hours **g:** Inspection hours. These were taken as indicative of a production process for a bulk production process.

The objective here was to use this in feature extraction to investigate the spread in intensity of the activities in a relative space and over time and to hypothesise the utility of this information.

The weight graphs (Figures 4, 5) show the intensity of each activity in terms of the update that the neurons have undergone. For the data set described above the best network topology was found to use six neurons. Choosing a lower number of neurons essentially makes the activation less visible as can be seen in the weight graphs corresponding to the graphs showing the number of neurons chosen to constitute the SOM grid (Figures 6,7).

Clock wise from top left: fig 4. Six neuron feature space; fig 5. Activation of the seven drivers; fig (s) 7 and 6 for three neuron feature space and activation, respectively



However, since in SAP there is already data and process standardisation, there will be records for all activities. Thus there is no need to reduce to a suitable, smaller number of pools. So feature extractions for this purpose 'may be obsolete' given that SAP is already in place. The argument to use this for 'optimising SAP' rather than use it in the business process re-engineering has again a serious drawback. If some activity records were to be excluded, which now show a high degree of overlap with others, it would be impossible to map changes in intensity of activities overtime. However, for this purpose feature extraction could be of great value. If the data set was broken into four parts a change in pattern of activity intensities over time can be observed.. This particular approach in using feature extraction can be very useful in identifying rise or fall in intensity of activities in a relative space.

Use of the 'outstar function' (here weights are transpose of instar), could give insightful information. The analyst can investigate the impact of changing one of the activities. The outstar produces a modified prediction vector of remaining activities, accounting for changes in the investigated activity.

HOPFIELD EQUILIBRIUM POINTS AND DETECTION OF DEVIATIONS

Hopfield networks are auto-associative structures with single layer lateral connections. Here encoding is performed simply by adding the product of activation at both ends of the synapse to the weight stored. Training is performed by presenting a pattern to the network and then letting it cascade back and forth till it stabilises to a steady retrieved pattern. An expel application of Hopfield networks is the optimisation of the famous Travelling Salesman like problems. Alternatively, the Hopfield algorithm can be applied for linear programming problems. However, in the case of highly erratic non-linear mapping they are outperformed by other network models like backpropagation [17]. In some ways it can be argued that Hopfield networks do not really learn. However they can be used for classification and detection in deviation from equilibrium in their basic form by the storing of equilibrium points. This is what is explored in the following application notes for product cost systems.

The objective here is to detect process anomalies, taking activity vectors as inputs over time. This is performed using the Hopfield networks to store a stable multivariate vector and then allowing the recall of the network (when presented with similar patterns) to identify unqualified relative variations in a particular activity in a given time period. Subsequently similar sized sets are fed in to see whether the pattern was stored. It can be seen in the Figures: 8 to 11 that the output corresponding to each of these sets marks out clearly the point of non-conformity in the activity consumption pattern. This can be very useful in spotting the exact point of non-conformity (unqualified variation).

A =

$$\begin{matrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & \underline{\mathbf{-1}} & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & \underline{\mathbf{-1}} & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{matrix}$$

B =

$$\begin{matrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & \underline{\mathbf{-1}} \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{matrix}$$

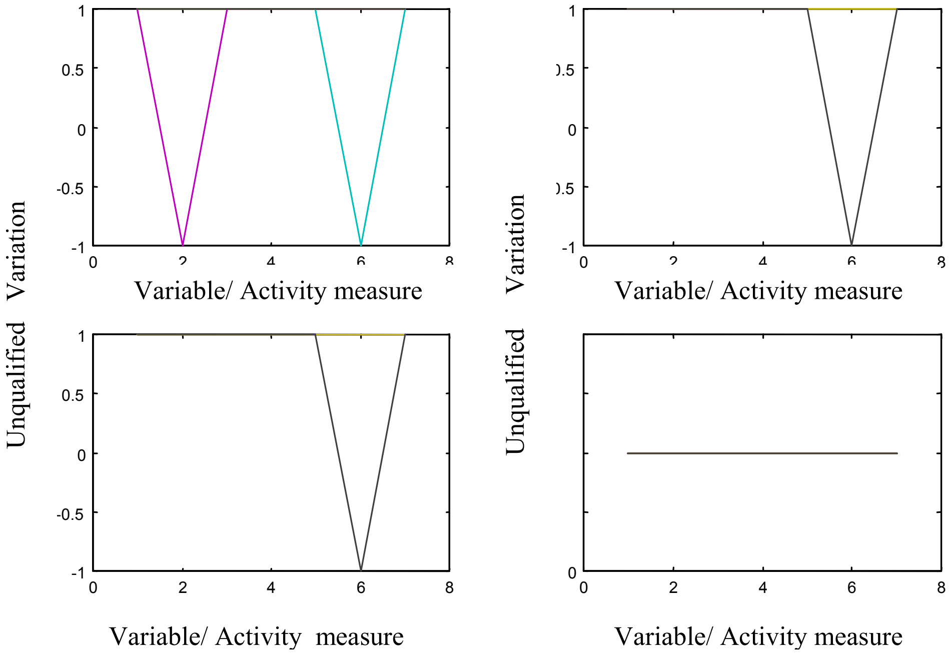
C =

$$\begin{matrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & \underline{\mathbf{-1}} \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{matrix}$$

D =

$$\begin{matrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{matrix}$$

Figure(s) 8,9,10,11



Unqualified Variation Detection. Clockwise from top left: Set A, Set B, Set C, Set D

Here since the data set of activities is being used, it can be seen that for the set 'B', an unqualified variation occurred for activity measure (**a:** Set up hours; **b:** Number of set ups; **c:** Material handling hours; **d:** Number of purchase orders; **e:** Number of hours in delivery runs; **f:** Maintenance hours **g:** Inspection hours). 'f' (row 6) in the time period 7 (column 7). In set 'A', there are two such events and one in set 'C'. For set D there are no unqualified variations implying that the consumption of activities in production is within the learned patterns. The same detection approach can be used for a different variable set where identification of unqualified variations is needed.

CONCLUSIONS

Product costing has taken a more process based outlook and a strategic role as is inevitable in the existing business environment where products mature very quickly and advantages of innovations are washed away very quickly. The need to monitor, control and optimise for product costs has thus found expression in fertile approaches that allow better organising and planning. Specifically these include life cycle costing, target costing, activity based costing and industry specific software. In the integrative outlook to management in ERP application software like SAP, there exists a potential for 'more expressive' and non-parameterised data processing, to improve the quality of information and utilisation of increased data interfaces available. Data modelling capacities of neural network, as non-parameterised systems, have a better reach in terms of capturing and expressing information by identifying patterns in data.

The growth in concepts and systems in product costing within the integrated information structures have been very much parallel to the growth in modelling techniques of neural networks. The integrated information structures implied are two fold. First relates to the enterprise effort at achieving the best decisions through generating exhaustive information in real time and monitoring processes for efficiency. The second relates to the computational issues that are dominated by the object-oriented paradigm that promises solving the complexity and improving the functionality of ERP structures like SAP. The approach of neural networks is analogous and thus hypothesised to be adaptable for increasing functionality in the process emphasis and computational environment. The information needs to serve the multiple strategic decision objectives can be effectively provided by neural networks as discussed in application illustrations and also in discussion on numerous studies that have established their equivalence if not superiority to conventional statistical techniques. This is in light of both the synergetic and quality of output advantages that they have in relation to the conventional techniques. The future work will concentrate on a comparative analysis of conventional statistical estimation with neural networks and will include a discussion on and application of Neuro-fuzzy hybrid systems.

REFERENCES

Blum, A. (1992) Neural networks in C++: **An object-oriented framework for buildingConnectionist systems**, New York: Wiley,, pp 6-55.

Börjesson, S. (1994) What kind of activity-based information does your purpose require? Two case studies. **Int Jnl Operations and Production Management**14(12), pp 79-99

Boyce, G. (2000) Valuing customers and loyalty: The rhetoric of customer focus versus the reality of alienation and exclusion of (devalued) customers. **Critical Perspectives on Accounting**, Vol. 11. No. 6, 649-689.

Cooper, R. and Kaplan R., S. (1991) **The Design of Cost Management Systems, Text Cases and Readings,** pp.3-7, 147-155.

Crow, K., DRM associates. (1997) Value analysis and function analysis system technique. http://members.aol.com/drmassoc/QFD.html

Dekker, S. Management accounting for the year 2000. http://www.dtrakker.com/articles/frame_y2kMgmtAcct.htm.

Demuth, H., and Beale, M. (1996) **Neural network toolbox**. Mathworks Inc, pp. 8-10 8-19.

Enterprise Management Technology.(2000) Kania School of Management. http://academic.uofs.edu/faculty/gramborw/sap/saptutorial.htm.

Farhoodi, F., and Fingar, F.(1997) Distributed Object Computing, "DOC" Magazine. http://home1.gte.net/pfingar/docmag_part2.htm (Oct, Nov 1997).

Fatelnig, K. P. (1997) ESA Cost Modelling, A General Description. http://www.estec.esa.nl/eawww/ecom/article/ecom.htm.

Hodju, P., and Halme, J. (1999) The SOM algorithm. http://koti.mbnet.fi/~phodju/nenet/SelfOrganizingMap/Theory.html .

Jain, B., A., and Nag, B., N. (1997) **Journal of Management Information Systems**, Vol. 14 No2, (1997), pp201-217.

Jiao, J., and Tseng, M. M. (1999) A pragmatic approach to product costing based on standard time estimation. **International Journal of Operations and Production Management**, (UK), Vol. 9 No. 7, pp 738-756.

Lagagne, M., and Discenza, R. (1993) New Product costing Japanese style-CPA in industry. The CPA Journal http://www.nysscpa.org/cpajournal/old/14345315.htm.

Lassila, K., Pallas, J., Pulakka, S., and Lee, S. (1994) A Comparison of Neural Networks, non-linear Biased Regression and a Genetic Algorithm for Dynamic Model Identification. International Chemometrics Internet Conference (InCINC'94). (1994).

Luettegen, R. (2000).ERP/ Package Implementation: Methods and considerations. Enterprise Software Group, Technology and Business Integrators, Inc., Woodcliff Lake, NJ.

McAuley, D. (1997).The Back Propagation Network: Learning by Example http://www2.psy.uq.edu.au/~brainwav/Manual/BackProp.html

Moreira, A.M.D., Clark R.G. (1996). Adding Rigour to Object-Oriented Analysis, **Software Engineering Journal**, Vol. 11, pp. 270—280

Otzen, D., and Rohde, C. (1999) (Copenhagen Business School) Product design and cost estimation using Neural Networks - A management accounting perspective- **Manufacturing Accounting Research conference** , University of Southern Denmark.

Ou-Yang, C., and Lin, T. S. (1997) Developing an integrated framework for feature-based early manufacturing cost estimation. **Int Jnl Advanced Manufacturing Technology** 13(9), 618-629.

Reed, J., and Goswami, S. (1998) Object-Oriented Integration in SAP - the BAPI. http://www.softwarejobs.com/sap/sap88.html .

Runge Pty Ltd., Xerax Cost Modelling Software. http://www.runge.com/xeras/HowWork.htm (1991-2000)

SAP labs, Inc. SAP AG . (1999). Product Costing Scenarios Made Easy 2-1 to 2-5. http://wwwtech.saplabs.com/other/product_costing/02%20Chapter%202.pdf

Sarle, W.S., Wise, B. M., Holt, B. R., and Gallagher, N.B., ed., (1997)- Neural Network FAQ, part 1-3 of 7 (introduction, learning, generalisation and applications): periodic posting to the Usenet newsgroup comp.ai.neural-nets, URL: ftp://ftp.sas.com/pub/neural/FAQ.html

Sharda, R., and Rampal, R. (1998) Neural Networks and Management Science/Operations Research: A Bibliographic Essay**. Encyclopaedia of Library and Information Science**, Vol. 61, Supp. 24, , pp. 247-259.

Sherwood, D. (1983) **Financial Modelling, A practical Guide**, Gee and Co. (Publishers) Ltd, pp 371-376.

Walczak, S. (1999) **Journal of Management Information Systems.** Vol. 16, No2, pp 177-195.

Zhang Y.F., Fuh J.Y.H., and Chan, W.T. (1996) Feature-based cost estimation using neural networks for packaging products. **Computers in Industry**, Vol. 32, pp. 95-113.