NSE TECHNICAL TRANSACTION VOL. 43, NO. 2, April – June, 2008 Fashanu, T. A. and Bakre, O. B.

A NEURAL MONITORING SCHEME FOR ON-LINE QUALITY ASSESSMENT OF PRODUCTS IN MANUFACTURING INDUSTRY

By

Fashanu, T. A. and Bakre, O. B.

Department of Systems Engineering, University of Lagos,
Akoka Yaba, Lagos State, Nigeria

ABSTRACT

The viability of most manufacturing systems is often benchmarked on low cost of operation, availability, low defective output ratio, and low life-cycle cost. Output quality depends largely on system's efficiency. Efficiency is predicated on overall machinery reliability, via the orchestration of effective subsystems and minimization of disturbances. The task of tailoring product quality therefore consists of functionality assurance of the production facility at the micro and macro levels. Consequently, this work is motivated by the need to assure output quality and systems reliability by continuous monitoring of operational, structural and environmental disturbances. Thus, we present herein a composite but continuous neural monitoring scheme for output quality assurance and reliability of manufacturing systems. Considering the diversity of inputs into a typical manufacturing set up, our scheme starts with the reduction of large systems using the significant parameters identification approach as incorporated into Artificial Neural Network (ANN) architecture. Subsequently, the feature extraction attributes of the Kolmogorov's ANN is employed to monitor the health status of products and machinery. Instrumentation data for training our ANN monitoring scheme was obtained from a local food processing plant.

1.0 INTRODUCTION

Many manufacturing setups consist of large and highly complex systems. As a result, it is often an enormous challenge to consistently achieve quality targets. Topmost on the list of challenges is process variability, this can be traced to endless sources such as; raw materials, equipment, sensitivity of sensors, operator shifts, environmental conditions, e.t.c. Each source can add subtle or significant changes that may lead to process drift and off-specification products. Non linearity of process dynamics may also contribute to the complication. In modern production environment, great dexterity and ingenuity is necessary to handle large and complex work piece spectrum and also to determine process latitude smartly. According to Muammer et al (2004), the efficient factory is one that uses real-time manufacturing intelligence to achieve best performance with minimal downtime. By implementing modern technology strategies that guarantee real-time manufacturing intelligence, manufacturers can achieve high level optimization and overall equipment effectiveness. Intelligent manufacturing ensures that machines operate within defined expectations, while monitoring efficiency and reliability. It also optimizes predictive and preventive maintenance. In addition, it assists manufacturers to identify subsystems that need more attention during turn around maintenance.

In many real world optimization problems, such as resource management and manufacturing, optimization has to be done under uncertainty. Uncertainty also makes the task nonlinear, and standard methods such as Linear Programming are known to perform sub optimally in such circumstances. In this work, our aim is to employ ANN to monitor and subsequently optimize manufacturing efficiency and reliability. The objective is to compensate for uncertainty, and extract hidden features in instrumentation data by taking advantage of the feature extraction capability of ANN as illustrated by Kotani et al (1993). Through learning from experience, the knowledge of relationship between previous failure patterns and causes can assist the ANN to minimize uncertainties associated with the optimization of manufacturing processes.

In view of the aforementioned, the emergence of computer-integrated manufacturing has made planning, optimization and control less tasking. Nevertheless, the need to find more efficient and less expensive ways to produce quality products has necessitated inclusion of ANN in the task of process optimization. From engineering perspective, ANN is a parallel dynamical system that can perform transformation by means of their state response to input information. The transformation is performed based on the ANN model (i.e. architecture, topology and choice of transfer function).

In architecture, ANN consists of processing units (neurons that actually execute instructions) and direct connections between each unit. Each of these connections has an associated real-value weight through which the network processes the impact of each variable and determines the real-time, nonlinear interactions of those variables on the final output. 'The ANN architecture is therefore identical to what obtains in most manufacturing setups. Thus the advantage of ANN in this regard is the ability to adapt its architecture to that of a specific manufacturing outfit. Consequently, basic features of subsystems can be extracted in the weight functions of corresponding ANN linkages. This provides real-time, remote and distributed monitoring of system to predict system's performance status and enable capacity for high quality products. In addition, it enables a scalable and reconfigurable platform for system evaluation, prognostics and optimization.

2.0 MODEL DEVELOPMENT: CASE BASED REASONING APPROACH

Consider an hypothetical manufacturing unit that consists of N machines labeled by numbers 1,2,3..., N. if there are M raw-materials denoted by r_1, r_2, r_3,......, r_M; T semi finished product numbered from s_1, s_2, s_3,......s_T and one finished product say Q. The product corresponds to a specific buffer. Under this assumption, there are $4_{(N+1)}$ buffers, the raw materials buffer are assumed to be of infinite capacity, thus their dynamics is ignored, and no dynamic equation is written here for them.

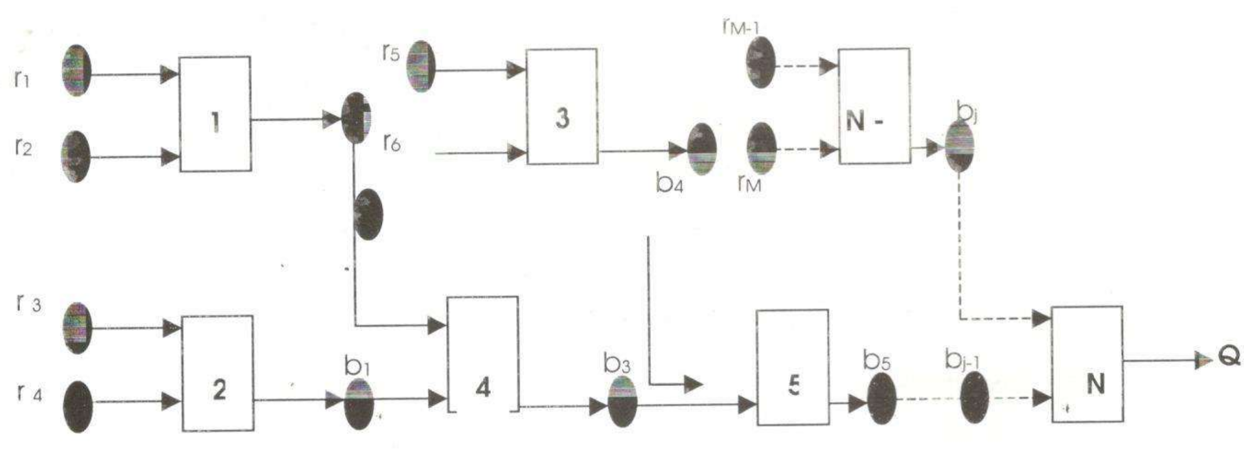

Figure 1: Hypothetical Manufacturing System Flowchart

Given that product quality and state of a manufacturing facility are obtained by instrumentation for successive points within a time window say $t_0 \leq t_n \leq t_0 + N\Delta t$; with t_n =

$t_0 + n\Delta t$, $n = 1,2,\ldots N$. This time window description allows a long string of measured variables to be isolated for analysis. To develop our model, we take cognizance of the general outline of process/product interactions as shown in Figure 2. To be consistent with this framework, we propose two distinct monitoring functions namely;

i. Manufacturing Facility Reliability Function:

ii. State of Product Quality Function

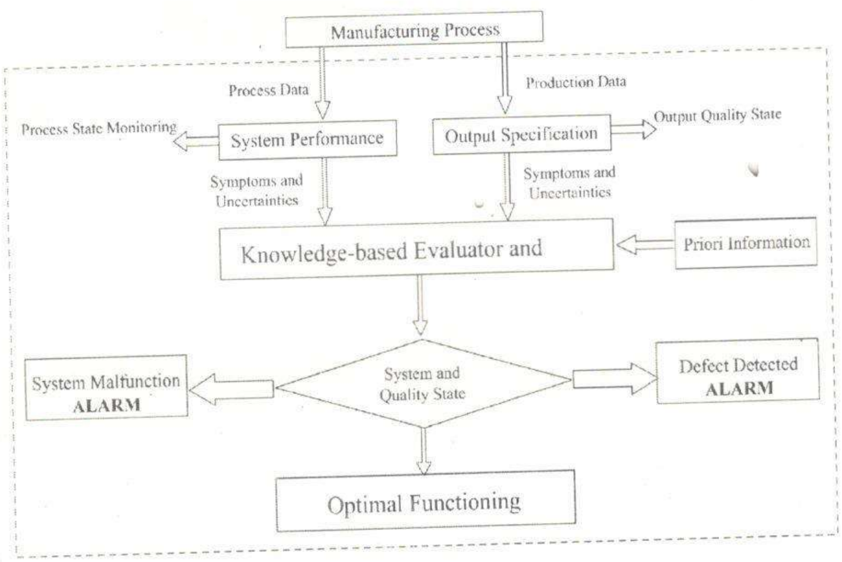

Figure 2: ANN Framework for Quality Assessment

2.1. Manufacturing Facility Reliability Function

Within two discrete time intervals t + Δt, we define the production reliability function

$$R(t) = [r_i^{t+\Delta t}]$$

Where r_i^t is the reliability measure defined for the subsystem r_i at time t, so that

$$R_j(t) = [r_i^{t+j\Delta t}];\ i,j = 1, 2,3,\ldots n; \qquad (1)$$

The reliability measure for each subsystem at the time t_j, r_i^t is estimated based on the performance index, using the following rule;

$$r_i^t = \begin{cases} 1 - \sum_{i=1}^{n} z_i^j \left(\sum_{i=1}^{n} \delta_i z_i^{max} \right)^{-1} & \sum_{i=1}^{n} z_i^{min} \leq \sum_{i=1}^{n} z_i^j \leq \sum_{i=1}^{n} z_i^{max} \\ 1 & \sum_{i=1}^{n} z_i^j \geq \sum_{i=1}^{n} z_i^{max} \\ 0 & \sum_{i=1}^{n} z_i^j \leq \sum_{i=1}^{n} z_i^{min} \end{cases} \qquad (2)$$

Where z_i^j = Performance of subsystem i at time j, z_i^{max} = Optimal performance of subsystem i, z_i^{min} = Minimum expected performance of subsystem i, and δ_i = Reliability index for subsystem $i z_i^j$, z_i^{min} and

z_i^{max} are indicated in the instrumentation data or the subsystems/system. Similarly, δ_i depends on the range $| z_i^{max} - z_i^{\min} |$.

Thus if we assume independent and incipient mode of failure in subsystem i at time t we can write

$$R(t) = \prod_{i=1}^{n} r_i^t \qquad (3)$$

The primary concern is to ascertain the overall system effectiveness by explicitly accounting for the level of interactions between different subsystems in the presence of uncertainty. Hence, applying the concept of process flexibility which is the ability of system continuously accommodate uncertainty, we define the flexibility index F_i for each subsystem. F_i measures the capability of the subsystem i to simultaneously accommodate inherent uncertainty in equipment availability and process parameters. Thus the flexibility of the system at time t can be expressed as

$$F^t = \prod_{i=1}^{n} r_i^t \; F_i \qquad (4)$$

and the flexibility index is then given as

$$\overline{F}^t = \frac{\eta(t)}{t} F^t \qquad (5)$$

$\eta(t)$ is the margin of safety at time t defined as the difference between capacity and demand. *In view of* equations (4) and (5), equation (2) reduces to

$$R^t = \begin{cases} 1; & \overline{F}^t \geq 0.5 F_S \\ 0; & otherwise \end{cases} \qquad (6)$$

Qualitatively, equation (5) represents the average flexibility index of the system state at a given time t. Infact it assesses its performance under uncertainty and random equipment failure. The closer this index is to the flexibility index of the initial state i.e. unity, the more reliable the system.

2.2 State of Product Quality Function

The overall quality performance of product q is derivable from quality of the constituent of the semi finished product q_i ; Thus Q can be expressed as

$$Q[q_i]; \quad i=1,2,\ldots n \qquad (7)$$

Product quality assessments are mainly based on quantitative measurement of constituents and an analysis of work piece features. Thus the limits of quantitative measurement of each constituent are set. In this model q_0 is quality the finished product. But since the quality is assessed over all the buffer points in the manufacturing set up, we will assign quality criteria for each buffer as

$$q_i = \begin{cases} 1; & if \quad \Omega \geq 0.5 \\ 0; & otherwise \end{cases} \qquad (8)$$

In an earlier work, Fashanu T. A., and Bakre O. B. (2005), equations (6) and (9) are utilized

Consequently Q is defined as

$$Q = \begin{cases} 1; & \sum_{i=1}^{n} \Omega_i \geq 0.5n \\ 0; & otherwise \end{cases} \quad (9)$$

Ω = is the parameter that qualifies quality heuristically; $0 \leq \Omega \leq 1$

In an earlier work, Fashanu T.A., and Bakre. O. B. (2005), equations (6) and (9) are utilized in the development of probability boundary function for ANN feature extractor. However, this work employs the K-Nearest Neighbour (KNN) algorithm coupled with the Gaussian kernel of Radial Basis Function (RBF) feature extraction and classification. The special attraction of the KNN in this regard is its ability to ascribe weights according to relative importance of subnets. This is of advantage in this case to prioritize our production subsystems.

To quantify Ω for a given product Q, we invoke the general form of similarity measure function for two samples X and E, based on the Euclidean distance between their attributes:

$$SIM(X,E) = 1 - \sqrt{\frac{1}{B\sum_{b=1}^{B}\lambda_b}\sum_{b=1}^{B}\lambda_b w_b^2 dist(x_b, e_b)}$$

(10)

X = Ideal product with no error pattern

E = Real product with error pattern

B is the total number of potential error patterns in the manufacturing process. λ_b is the number of attributes in case *b*. Normalized weight w_b characterizes the importance of b^{th} attribute corresponding to the normalized metric dist (x_b, e_b): $dist(x_b, e_b) = |x_b - e_b| / |\max_b - \min_b|$

(11)

x_b, e_b are the b^{th} attribute values in the two cases, while numerical attributes "max_b" and "min_b" denote the maximum and minimum values of the b^{th} attribute. Also defining $v = (v_1\ v_2\ ...v_n)$ as the vector of control strategies designed to transform measurable parameters of the system and product to their peak values. The functionality objective function:

$$\sqrt{n}\|R(t)+Q(t)\| = \langle v(t), SIM(X,E), R' \rangle \quad (12)$$

must be satisfied. $\|\bullet\|$ denotes the Euclidean norm of $\bullet$. Note that, $\Phi(t) = QF^t$ is defined as the Output Specificity. Furthermore given $v(t)$, the transformation of $H(t,v)$ is presumed to proceed via:

$$SIM(X,E)v(t)Q(t)R^{T} = H(t,v) \qquad (13)$$

Equations (12) and (13) unify product quality and systems reliability.

3.0 ARTIFICIAL NEURAL NETWORK ARCHITECTURE

In view of the objectives highlighted above, Hierarchical Neural Network (HNN) model i.e. global networks consisting of hierarchy of single neural networks as shown in Figure 3 is adopted in this work. In the review of Wan Hussain and Wan Ishak (2004), the HNN is known to be efficient for distributed learning. Furthermore, it demonstrates considerable reduction in size of training data, large spatial extensions and sharp contrast in classification. The HNN architecture is also adaptable to that of the distributed system that it is designed to learn. Thus, our HNN is considered to have three subnet levels. These include the parametric network, specialized network and the integrated network. As illustrated in the key to Figure 3, each subnet is expected to model and summarize in its output the important characteristics of its ultimate set of input variables. In this arrangement, no two subnets can have identical set of variables; however a variable may participate in many subnets of each layer.

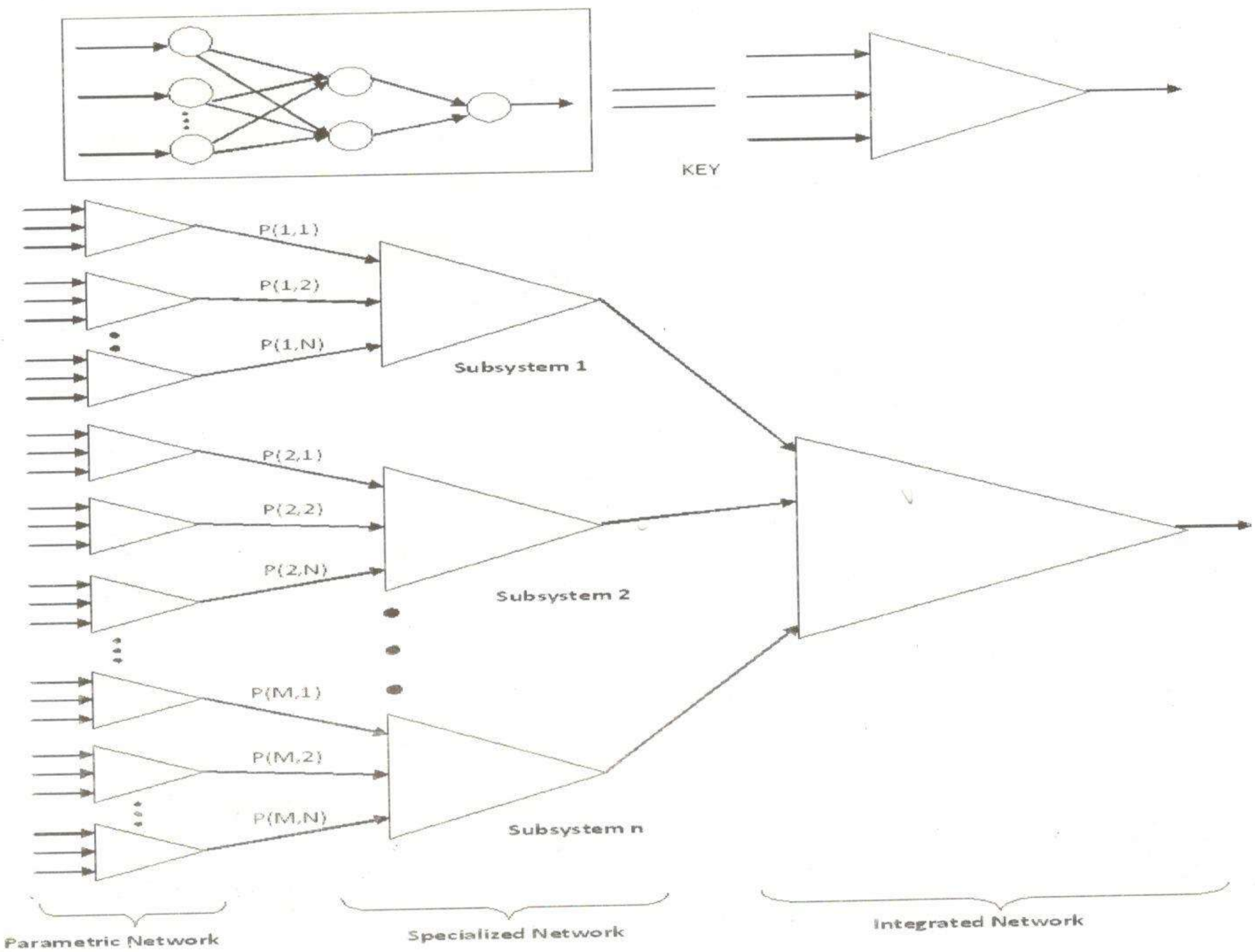

Figure 3: Hierarchical Neural Network Architecture

3.1 Pattern Classification

For any given operational variable, measurements at different time windows form an important cluster that contains the features and dynamics of the variable. Similarly, for a process stream within any time window, measurements of all the variables of the stream/operation are somehow related. Also, for any specific kind of variable at a fixed time point, measurements for all streams form a cluster. In our proposed architecture, the outputs of lower nets are screened and fed as inputs to higher level subnets. Since each subnet and their input variables can be used by several subnets at the next layer. The complexity of individual subnets depends on the attributes of the semi finished product and features of the subsystem that the subnet models.

3.2 KNN & Gaussian Radial Basis Function Classifier

According to Meng Joo et al (2002), an appropriate training protocol for the HNN architecture and KNN classifier is the Gaussian Quadrature of the Radial Basis Function (RBF), this choice is informed by the need to avoid moving target problems that may evolve during network training which may be more pronounced with increasing number of inputs. This combination also imparts computational ability into the neural network. As a result, the health status $h(P_p)$ of a product/component due to input vector:

$$P = [P_1, P_2, \ldots, P_N] \tag{14}$$

That satisfies the non-linear state equation

$$x(t+1)=f(w(t);P(t)) \tag{15}$$

Can be mapped by the Gaussian Kernel of the RBF as

$$h(P_p)= w_0 + \sum_{i=1}^{M} w_i \exp\left(-1/2\sigma_i{}^2 \sum_{j=1}^{J} (P_{pj} - c_{ij})^2\right) \tag{16}$$

Where c = center, σ = variance, w_i = weights.

From equation (16) suppose we set

$$a_{p,i} = \Theta\left\|\bar{P}_p - \bar{c}_i\right\| = \exp\left\{-\frac{1}{2\sigma_i^2}\sum_{j=1}^{J}(P_{p,j} - c_{i,j})\right\} \tag{17}$$

Hence, the compact form of equation 16 can be written as

$$h(P_p) = \sum_{i=0}^{M} w_i a_{p,i} \quad p = 1,2,\ldots N \tag{18}$$

For a perfect product or a newly commissioned plant we set $a_{p0} = 1$. Thus in matrix format, equation 18 becomes

$$\begin{pmatrix} h(P_1) \\ h(P_2) \\ \bullet \\ \bullet \\ \bullet \\ h(P_N) \end{pmatrix} = \begin{pmatrix} 1 & a_{11} & \bullet & \bullet & \bullet & a_{1M} \\ 1 & a_{21} & \bullet & \bullet & \bullet & a_{2M} \\ \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ 1 & a_{N1} & \bullet & \bullet & \bullet & a_{NM} \end{pmatrix} \begin{pmatrix} w_0 \\ w_1 \\ \bullet \\ \bullet \\ \bullet \\ w_M \end{pmatrix} \tag{19}$$

Considering the rectangular nature of the system in equation (19), the weight vectors W for each subsystem are obtained directly by pseudo inverse matrix techniques based on the least square method. Also, the center c; and the width σ in the Gaussian RBF kernel are obtained by the clustering algorithm reported by Musavi et al (1992).

Thus,

$$c^l = \frac{1}{L^l}\sum_{k=0}^{n} Y_i^l \tag{20}$$

Where Y_i^l is the i^{th} sample class and L^l is the total number of pattern in class l

Similarly, the width σ^l is given by the equation

$$\sigma^l = \sqrt{\frac{1}{k|\ln \beta|}\sum_{k=1}^{K}\left\|c^k - c^l\right\|^2}$$

(21)

Where $0.5 \leq \beta < 1$; is called confidence, and c^k is the k-th nearest neighbor of c^l. The choice of β is determined by the distribution of sample pattern.

3.3 Computational Algorithm

As mentioned earlier, the KNN classifier and the HNN network are combined here for computational rather than learning procedure. The updating equations for the Gaussian RBF kernel can be expressed as

$$w_i(t+1) = \rho w_i + \begin{Bmatrix} \alpha_w \sum_{p=1}^{N} \psi(P_p)\,\theta(P_p) \text{ when } i = 1,2 \ldots, M \\ \alpha_w \sum_{p=1}^{N} \psi(P_p), \text{ when } i = 0 \end{Bmatrix} \quad (22)$$

$$\sigma_i(t+1) = \sigma_{ij}(t) + \alpha_\sigma \sum_{p=1}^{n} \psi(P_p) w_i \theta_i(P_p)\left(P_{pj} - (c_{ij})\right)^2 [\sigma_{ij}(t)]^{-3}$$

(23)

$$c_{ij}(t+1) = c_{ij}(t) + \alpha_c \sum_{k=1}^{n} \psi(P_p) w_i \theta_i(P_p)(P_{pj} - c_{ij})[\sigma_{ij}(t)]^{-2}$$

(24)

Given that $\psi(P_p) = d(P_p) - h(P_p)$ (25)

4.0 MODEL APPLICATION

To validate our model, we consider a food seasoning plant within a local food processing industry. The plant's layout is as shown in Figure 4 below.

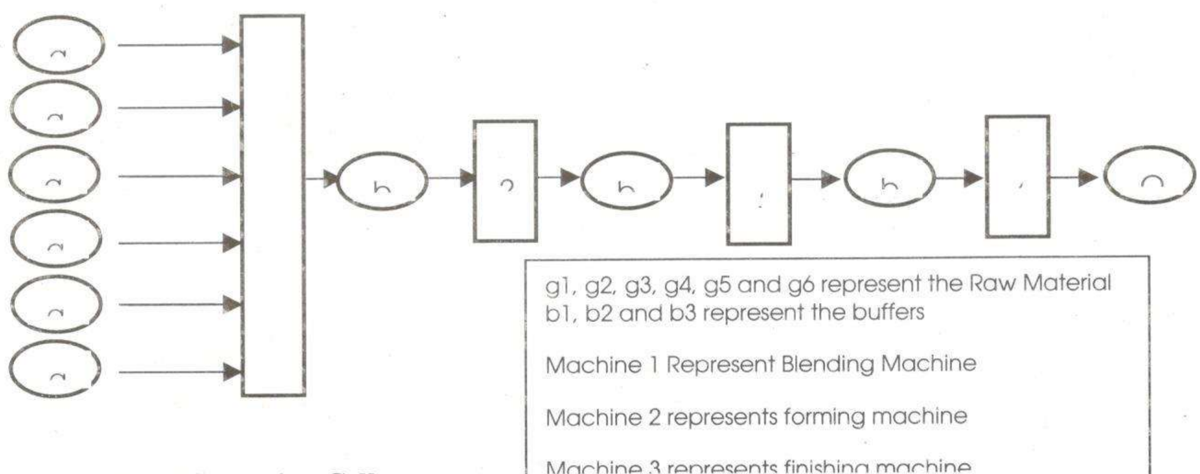

Figure 4: Food Seasoning Cell

In this setup, raw materials are mixed in the blender in the proportion 0.01g1, 0.48g2, 0.30g3; 0.12g4 and 0.09g5, at buffer 1 i.e. b1 there are 2 quality assessment criteria namely; quantitative test and homogeneity test. The quantitative test is a laboratory test that ensures appropriate composition. Homogeneity test is done heuristically by rating between 0.1 and 1.0 by an expert through visual inspection. Compression to shape takes place at machine 2. The strength of the formed shape is subsequently obtained using the Brinnel Hardness Test. This test is necessary to ensure that the seasoned product is as crunchy as required. Machine 3 is finishing machine, it wraps the product. The quality of the wrapping is checked at buffer 3, to ensure good prese vation. Bagging is done by machine 4 to obtain Q.

The learning procedure is done by forward propagation of the data in a given time window, the input data given with the functional signal is given to calculate output of the j^{th} RBF unit, the weight W as stated in equation (19). Having identified the weight, the functional signal is propagated forward until the error is calculated as given in equation (25). During the backward pass, the error will propagate from the output end towards the input end while modification to the weight, center and width are done as given by equation (22), (23) and (24). Table 1 shows the learning rate of the RBF parameters in these equations. Similarly, the weights are initialized by $\alpha_{i,j}(0) = 0.04\alpha_w$ while $C_{i,j}(0)$ is obtained through equation (18).

Table 1: Learning Rates of Radial Basis Function Parameters

Parameter	Weight (α_w)	Width (α_σ)	Centre (α_c)
Learning Rate	0.07	0.05	0.03

5.0 RESULTS AND DISCUSSION

A hundred clusters of instrumentation data were obtained over continuous but uniform time window of five seconds each. These clusters were labeled odd and even according to their serial occurrence. The fifty odd numbered clusters were used for training while the even numbered clusters were used for validation and neural computation of product/facility health status. This arrangement is to ensure that no significant shift occurred between training and testing data. Simulation was conducted in Matlab 6.1 environment with the application of Neural Network and Optimization Tool Boxes. Convergence condition for training all parameters is $|\tau(t+1) - \tau(t)| < 0.0001$, where τ represents the parameter under training. The average duration for training was 29seconds, in 3,197epochs on an Intel 1.83GHz processor.

The shortlisted input variables by the parametric network include; feed composition, hardness, and homogeneity.

Others are blender speed, mould rigidity, forming pressure, wrapper size, wrapping speed, and wrapping integrity. Wrapping integrity is also obtained by rule of thumb using visual inspection. The network acts as a discriminant function that passes or rejects a product. In addition, through the health status of the subsystems, the network classifies the overall health of the food seasoning cell into three, namely; good, tolerable and emergency performance. The subsystem with the poorest health condition determines the overall health of the cell. The criteria for classification are as shown in Table 2.

Table 2: Classification Criteria for Seasoning Plant Health Condition

Machine	Good Performance	Tolerable Performance	Emergency Condition
Blending	$0.915 \leq h(P_p) \leq 1.000$	$0.825 \leq h(P_p) \leq 0.915$	$h(P_p)$ □ 0.825
Forming	$0.940 \leq h(P_p) \leq 1.000$	$0.885 \leq h(P_p) \leq 0.940$	$h(P_p)$ □ 0.880
Wrapping	$0.875 \leq h(P_p) \leq 1.000$	$0.850 \leq h(P_p) \leq 0.875$	$h(P_p)$ □ 0.850
Bagging	$0.90 \leq h(P_p) \leq 1.000$	$0.850 \leq h(P_p) \leq 0.900$	$h(P_p)$ □ 0.850

Table 3: Distribution of Performance Status and Specificity for 9 Minute Monitoring

Performance Status	Percentage Distribution	Normalized Specificity
Good	32.36	0.62
Tolerable	67.45	0.81
Emergency	0.19	0.47

The seasoning plant was monitored for duration of nine minutes. During this period, the percentage classification of machine health and corresponding average specificity within the specified time windows is as shown in Table 3.

Good performance indicates good condition of health of the system. Tolerable performance implies that the system is operating in a sub optimal but acceptable condition. Average specificity is an index of reliability on the present state of health. To be precise, it specifies the degree of belongingness of the system to the prevailing state of health. Emergency condition with relatively high normalized specificity (>0.6) demands that system undergoes preventive maintenance. Also, there is a time delay of eleven seconds between acquisition of instrumentation data and the display of systems health status. However, the time lapsed can be significantly reduced by improving the processor's speed, reducing the data size for analysis or adapting the ANN scheme for predictive diagnosis.

6.0 CONCLUSION

Information relating the quality of product and process performance is necessary in order to be able to improve the output of a manufacturing system. The neural monitoring scheme which has been developed in this work has the potential to assist manufacturers and other system managers in gaining real time information on the performance of their system and providing feedback on subsystems that need prompt attention. By combining the learning, feature extraction and computing abilities of the artificial neural network, a significant improvement was made on the tools available for real time analysis of instrumentation data. It is hoped that further efforts will refine this scheme, minimize the time lag between data acquisition and status reporting, as well as incorporate stability analysis of the monitored system.

REFERENCES

1. Fashanu A. T. and Bakre O.B; 2005 " A Neural diagnostics model for continuous Estimation of Reliability and Availability of Engineering Systems" Proceeding of SMRssl '05, Ben Gurion University, Ber Sheeba Isreal.

2. Kotani M., Uno K., Masumoto and T. Knagawa 1993 "Acoustic diagnosis with neural network" Robotics Mechatronics and Manufacturing systems: T. Takamori and K. Tsuchiya (Editors); Elsevier Science Publisher; pp.835-840.

3. Meng Joo Er et al; (2002) "Face recognition with Radial basis Function (RBF) Neural Networks" IEEE Transactions on Neural Editors, Vol 13 No.3 pp697-710.