ENGINEERING APPLICATION OF CORRELATION ON ANN ESTMATED MASS

H. B. Bahar

Faculty of Engineering, Tabriz University
PO Code 5166614776, Tabriz, Iran

(Received: December 27, 2001 – Accepted in Revised Form: Mar 9, 2002)

Abstract A functional relationship between two variables, applied mass to a weighing platform and estimated mass using Multi-Layer Perceptron Artificial Neural Networks is approximated by a linear function. Linear relationships and correlation rates are obtained which quantitatively verify that the Artificial Neural Network model is functioning satisfactorily. Estimated mass is achieved through recalling the trained Artificial Neural Network model on a set of waveforms resulting from applied masses over the operating range of the weighing platform. In this work the Least-Squares Fit (LSF) method for straight line and correlation rate R between the applied and estimated masses are used to investigate the accuracy of the estimated masses. The slope of the linear functions together with correlation rates R are computed for both simulation and experimental data. The numerical results confirm the correctness of neural network technique in estimating the applied mass m(t).

Key Words Correlation Rate, Least-Squares Fit, Artificial Neural Networks, Dynamic Weight Estimation, Platform Model, Time Series Data, System Identification

چکیده رابطه بین دو کمیت متغیر، جرم اعمال شده به ترازو و جرم تخمینی بوسیله مدل شبکه عصبی MLP با تابع خطی تقریبن شده اند. روابط خطی و میزان همبستگی R بطور کمی نشانگر عملکرد رضایت بخش مدل شبکه عصبی MLP در تخمین جرم می باشد. شبکه عصبی آموزش دیده ، بر اساس پاسخ حاصله از اعمال جرم بر روی سکوی ترازو در محدوده ظرفیت دینامیکی آن و در حالت گذرای پاسخ، مقدار جرم تخمینی را مشخص می کند. در این مقاله تابع خطی ناشی از نظریه حداقل مربعات (LSF) و میزان همبستگی R، دقت مقادیر وزن های تخمین زده شده را بررسی می نماید. در بخش های شبیه سازی و تجربی شیب توابع خطی و مقادیر همبستگی محاسبه شده اند. نتایج حاصله از شیب توابع خطی و مقادیر R ها درستی استفاده از تکنیک شبکه عصبی را در تخمین جرم تایید می کنند.

1. INTRODUCTION

The application of an object to weighing platform results in a transient output waveform that can take a considerable time to settle sufficiently to accurately indicate the weight of the object. Accurate and fast weighing is a widespread requirement in industrial and other applications [1-4].

Various approaches have been proposed to improve the speed and accuracy of weighing platforms. These include adaptive filtering [5], dynamic weight under non-zero initial conditions [6], non-linear regression method [7,8], and dynamic weight estimation using an Artificial Neural Network [9]. The Multi-Layer Perceptron (MLP) Artificial Neural Network (ANN) is used with back propagation training for the analysis of time series data to estimate the applied mass [10–13].

In this paper use of the Least-Squares Fit (LSF) method for a straight line is used for verification purposes between a pair of applied and estimated ANN output masses. For justification of the straight line fitting approach, a linear correlation rate is presented [14,15].

The Least-Squares Fit (LSF) and the correlation Rate (R) of simulated results show the potential low sensitivity of the ANN method to simulated measurement noise. Results for LSF and correlation rate R on experimental data that are obtained from an actual industrial weighing platform, confirm the achievement of accurate mass prediction even when several modes of vibration are present.

2. WEIGHING SYSTEM MODEL

An ideal weighing platform can be modelled by a mass-spring-damping structure shown in Figure 1. It has a typical underdamped ideal step response as illustrated in Figure 2. It is governed by the solution of the following second order differential equation;

$$\left(m(t) + mp\right)y''(t) + Cy'(t) + Ky(t) = gm(t) \qquad (1)$$

where y(t), is the deflection signal obtained from the strain gauge on the weighing machine; m(t) and mp are the applied mass and the platform mass respectively; C is the damping factor; K is the spring constant and g is the gravitational constant.

For a general applied mass function m(t), this is a non-linear differential equation. However, for commonly encountered situations m(t) is a step function, which is assumed here. In this case the differential Equation 1 is linear, for which the explicit solution is modelled by a constant term plus a transient term which can be underdamped (u), critically damped (c), or over damped (o).Thus,

$$y(t) = q_0 + \{F_u(q_u,t), F_c(q_c,t) \text{ or } F_0(q_0,t)\} \qquad (2)$$

and the transient terms for underdamped, critically damped and overdamped cases are,

$F_u(q_u,t) = e^{-q_{u1}t} q_{u2} \sin(q_{u3}t + q_{u4})$,

$F_c(q_c,t) = e^{-q_{c1}t}(q_{c2} + q_{c3}t)$, and

$F_o(q_o,t) = e^{-q_{o1}t} q_{o2} + e^{-q_{o3}t} q_{o4}$, respectively, where the various q parameters are related to the initial platform displacement, b_0, initial velocity b_1, the platform parameters K, C, mp and the applied mass m(t) by the expressions given in the Appendix I, see [7,8]. These expressions have been used to generate data in the simulation study that is described below.

Sampled data signals are assumed for which t=nT, where T is the sample interval. Thus, y(t) is written as y(n).

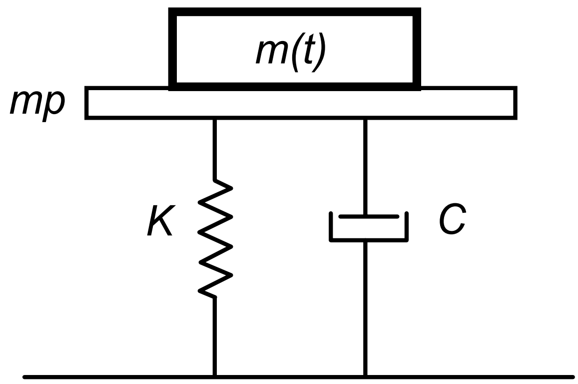

Figure 1. A model of weighing paltform.

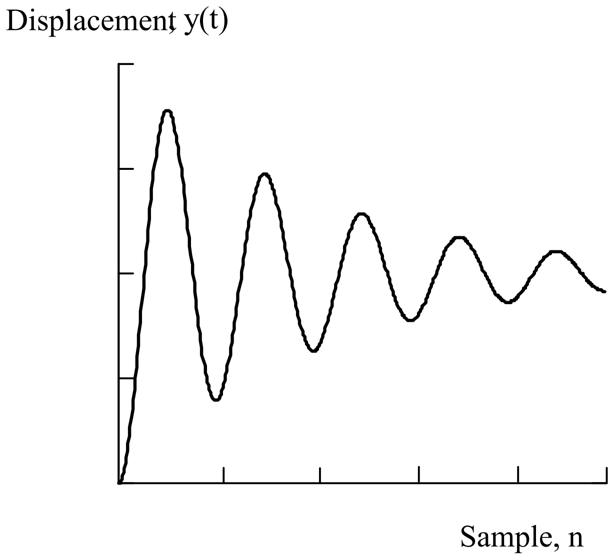

Figure 2. A typical step response of weighing platform.

3. ARTIFICIAL NEURAL NETWORK

An input-output model describes a dynamic system based on input and output data. An input-output model assumes that the new system output can be predicted by the past inputs of the system. Further, if the system is supposed to be deterministic, time invariant, multiple-input-single-output, the input-output model becomes:

$$w(n) = f(y(n),\ y(n-1), \cdots,\ y(n-N+1)) \quad (3)$$

where y(n), w(n) represent the input-output pair of the system at time n, positive integer N-1 is the number of past input samples (called the order of the system) and f is a static non-linear function which maps the past inputs to a new output as shown in Figure 3. The diagram illustrated in Figure 3, where the Artificial Neural Network model is defined by function f and symbol z-1 denotes the time delay between two successive samples, can implement Equation 3 [9-12].

Figure 4 indicates a Multi-Layer Perceptron (MLP) Artificial Neural Network (ANN) block diagram with back propagation training or learning method. In back propagation, training is performed by forward and backward operations; in the forward operation the network produces its actual outputs for a certain input pattern using the current connection coefficients. Subsequently, the backward operation is carried out to alter the coefficients to decrease the mean square error between the actual and desired outputs see [10-13] for example. Steps of the algorithm that are involved in constructing and training a MLP network are given in Appendix II.

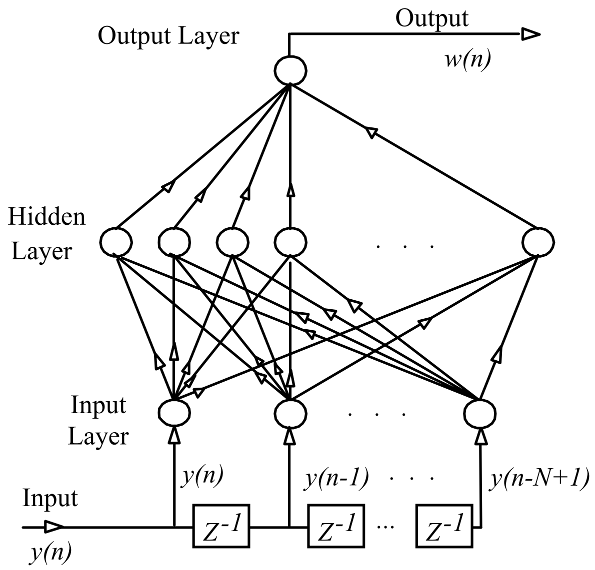

Figure 3. Input-output Artificial Neural Network Architecture.

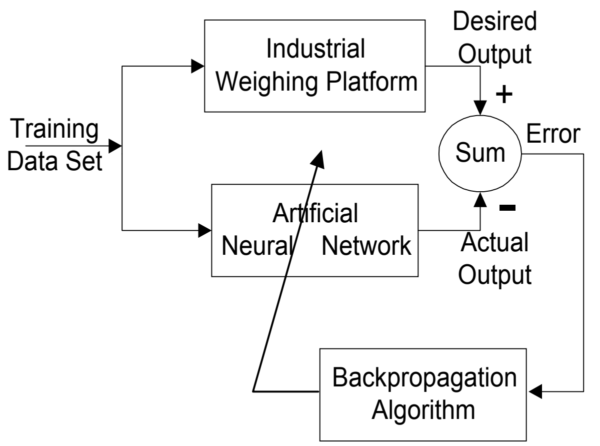

Figure 4. Artificial Neural Network Back propagation training model.

4. LEAST-SQUARES FIT TO A STRAIGHT LINE

In this work the Least-Squares Fit (LSF) method for straight line is used for verification purposes between a pair of applied mass m(t) and estimated output mass w(n). The functional relationship between two variables, desired applied mass m(t) and actual estimated mass w(n) can be approximated by the linear function :

$$w(n) = a + bm(t) \quad (4)$$

There is no unique method for optimising the coefficients in Equation 4, a and b which is valid for all cases. The method of maximum likelihood consists of making the assumption that the observed set of measurements is more likely to have come from the actual parent distribution of Equation 4 than from any other similar distribution with different coefficients [14]. For any pair of applied and estimated masses the mathematical expressions to the coefficients a and b are given in Appendix III.

5. LINEAR CORRELATION RATE

For justification of the straight line fitting procedure described in the previous section a linear correlation rate R is defined [14], which indicates the rate of correlation between two variable quantities; i.e. object weight m(t) and estimated mass w(n). The analytical solution developed for the coefficient b, which represents the slope of the fitted line given in Appendix III, is,

$$b = \frac{1}{\Delta}\left(M \sum x_i y_i - \sum x_i \sum y_i\right) \qquad (5)$$

Since the linear interrelation between the variables m(t), w(n) is being discussed, the data corresponds to a straight line, it has the form:

$$m(t) = a' + b' w(n) \qquad (6)$$

The analytical solution for the slope b' is similar to that for b in Appendix III, hence,

$$b' = \frac{M \sum x_i y_i - \sum x_i \sum y_i}{M \sum y_i^2 - \left(\sum y_i\right)^2} \qquad (7)$$

If there is complete correlation between m(t) and w(n), then the existence relationship between the coefficients a, b and a', b' is:

$$w(n) = -\frac{a'}{b'} + \frac{1}{b'} m(t) \qquad (8)$$

$$w(n) = a + bm(t) \qquad (9)$$

and equate slopes coefficients in Equation 8 and 9 then,

$$b = \frac{1}{b'} \qquad (10)$$

Here the experimental linear correlation coefficient is defined as:

$$R = \sqrt{b\,b'} \qquad (11)$$

Referring to Equations 10 and 11, the absolute magnitude of *R* ranges from 1 to 0 when there is no complete correlation between the desired and actual values. The application of the Least-Squares Fit (LSF) to a straight line and linear correlation rate *R* are found to be suitable criteria in this work for justification purposes between the desired and actual output values.

6. SIMULATION PERFORMANCE

The Artificial Neural Network architecture shown in Figure 3 is used for simulation purposes. ANN was trained to emulate the dynamic behaviour of the weighing system, so that output w(n) is an estimate of the applied mass m(t). Suitable specifications for the ANN model as illustrated in Figure 3 were found to be,

- Number of input samples y(n), ... y(n-N+1), is taken as N=200.
- Number of output samples w(n), is 1.
- Number of layers is 3, an input layer, a hidden layer and an output layer.
- Total number of neurones is 301: 200 neurones in the input layer, 100 neurones in the hidden layer, and single neurones in the output layer.
- Momentum rate is, 0.95.
- Learning rate is, 0.50.

A set of 100 patterns is used for ANN training and recalling. Each input pattern is composed of the first 200 samples, y(n), ..., y(n-199), following the application of the mass to the platform. The input patterns for training and recalling were generated by C++ programming language to simulate the Equation 2. The weighing platform parameters in all simulations are K=1000 N/mm, C=50 Ns/mm, mp=0 kg, g=10 m/s^2, sampling interval t_s=0.02 ms, initial platform displacement b_0=0 mm, and initial velocity b_1=0 mm/s. Applied masses were uniformly chosen to cover the range m(t)=1,2, ..., 100 kg. For the platform parameters given, only the underdamped expression $F_u(q_u,t)$ in Equation 2 was required.

TABLE 1. Simulated Training and Recalling Performances.

Patterns	RMS Error (kg)	Relative Average Error	Correlation Rate, R
Training	0.02327	0.040%	1.0000
Recalling (seen with 2% noise)	0.02327	0.040%	0.99989
Recalling (unseen with 2% noise)	0.36673	0.66%	0.99989

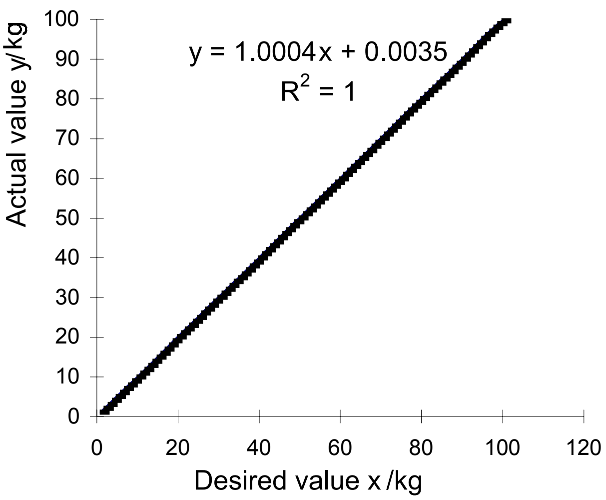

Figure 5. Simulated recalling results of noise free seen patterns.

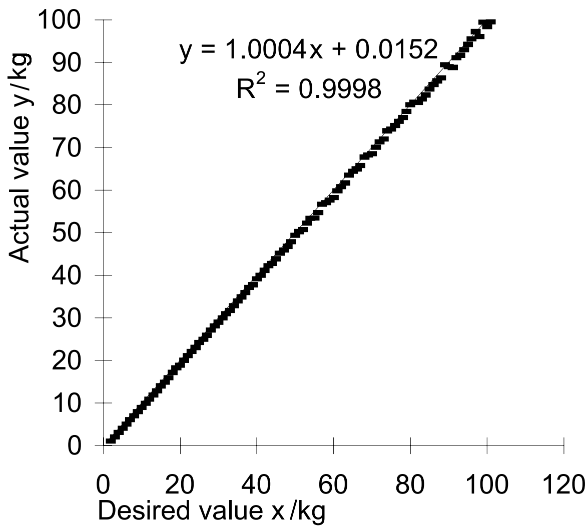

Figure 6. Simulated recalling results of seen patterns with 2% random noise.

The Artificial Neural Network model in Figure 3 is trained by applying the noise free patterns, where the training procedure is as explained in the previous section. The simulation performance of the trained ANN is illustrated in Figure 5. Where in Figure 5 the square root of R^2 assigns correlation rate of R=1 and linear relationship between applied mass m(t) and estimated output mass w(n) denotes a slope of 1.0004. For conclusion aims the numerical results together with rms error and relative average error are given in Table 1.

For recalled Artificial Neural Network purposes, the seen and unseen input patterns were contaminated by uniformly distributed random simulated measurement noise with an amplitude of 2% of the steady state mass. Then the patterns are applied to the aforementioned trained ANN to obtain the estimated output mass, w(n). The simulated recalling results for the seen and unseen input patterns ranging from 5 -100 in a step of 5 kg are shown in Figures 6 and 7 respectively. Table 1 provides the overall rms and relative average errors for training and recalling performances. The effect of the contaminated noise with amplitude of 2%, results

TABLE 2. Experimental Training and Recalling Performances.

Patterns	RMS Error (kg)	Relative Average Error	Correlation Rate, R
Training	0.04456	0.11%	0.99994
Recalling	0.72436	1.86%	0.99894

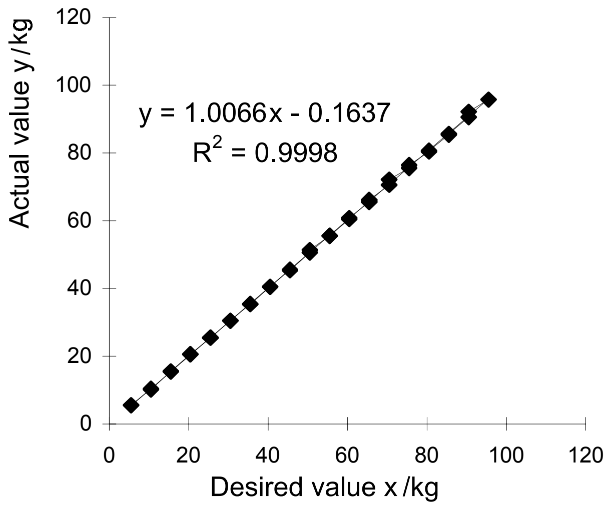

Figure 7. Simulated recalling results of unseen patterns with 2% random noise.

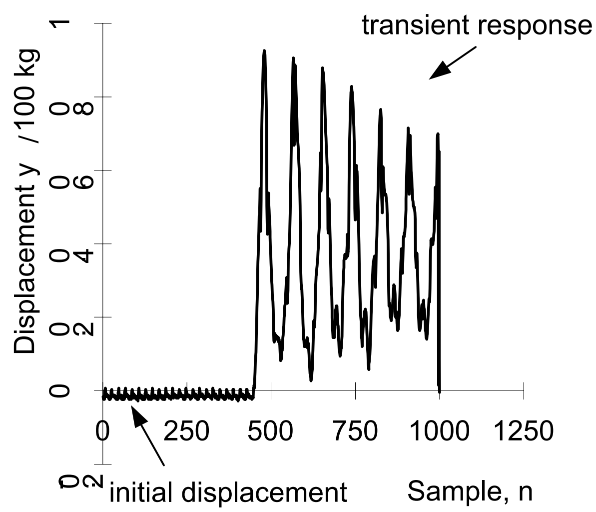

Figure 8. Experimental weighing platform displacement for applied mass, m(t)=40 kg.

estimated seen masses, see Table 1. Manipulations Figures 5, 6 and 7 give the magnitudes for the rms error, relative average error and correlation rate R, as tabulated in Table 1.

7. EXPERIMENTAL PERFORMANCE

In the present section training and recalling processes of the Artificial Neural Network are reported using experimental data, which were obtained from an industrial weighing platform. The weighing platform has dimensions of 55.50 cm, 50.50 cm, 16.50 cm for length, width and height, respectively with a nominal full scale taken to be about 100 kg.

An Artificial Neural Network was trained with experimental time series patterns. A typical weighing platform response for an applied mass, m(t)=40 kg is shown in Figure (8), where the response is composed of two distinctly different regions, initial displacement and transient response region. For each of sequences of applied masses, 200 samples were taken in the transient region immediately following the application of the mass to the platform with uniform sampling intervals of 2 ms. Thus, the overall time to produce an estimated output for each applied mass is 400 ms. Here the sequence of applied masses was taken to be 5-80 kg in steps of 5 kg, except the masses used for recall purposes. The resulting performance of

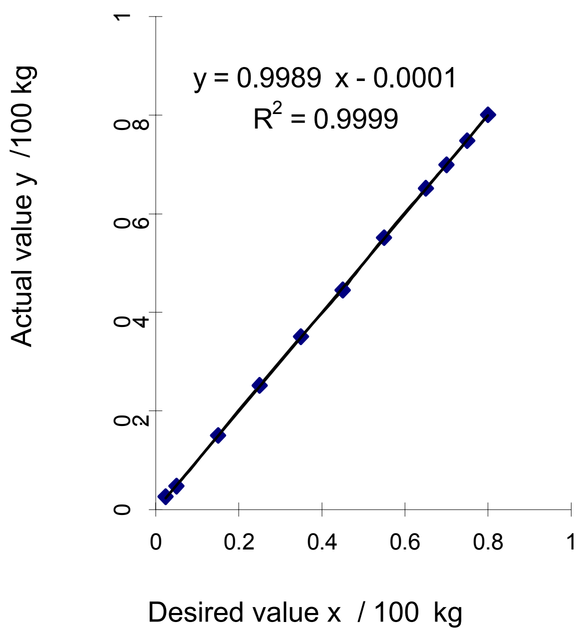

Figure 9. Experimental ANN trained performance to environmental noisy seen patterns.

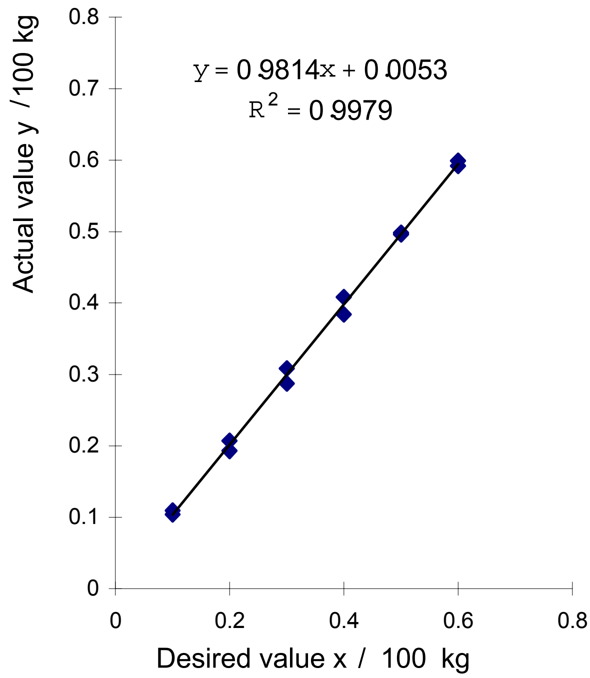

Figure 10. Experimental recalling performance to practical noisy unseen patterns.

the trained ANN in a laboratory environmental for various masses is illustrated in Figure 9. Sources of noise may include signal interference and any higher modes of platform vibration.

At recalling process, the trained Artificial Neural Network is examined using unseen time series experimental patterns, which are not used at the training of the ANN. To conform to a more real use of a weighing platform the unseen masses, 10-60 in step of 10 kg were applied less carefully than for the training data. That is the masses were dropped onto the platform from a height of typically 2 cm. The results of the recalling performance of the trained ANN for unseen masses is shown in Figure 10. Table 2 shows the overall errors for both the experimental training data and recalling data. However, the recall data errors as illustrated in Table 2, are higher because the masses were applied in a more realistic fashion. Even so, the errors are relatively small considering that the weights are estimated dynamically, long before the waveforms have settled to steady state.

8. DISCUSSIONS AND FURTHER WORK

Illustrations in Figures 5, 6, 7, 9 and 10, indicate the existence of linear relationships between actual and desired masses with slopes of 1.0004, 1.0004, 1.0066, 0.9989 and 0.9814 respectively. The square roots of R^2 in Figures 5, 6, 7, 9 and 10 allocate correlation rates of 1.00000, 0.99989, 0.99989, 0.99994 and 0.99894 for each performances respectively. These numerical results which indicate roughly unity magnitudes for both, slops and correlation rates, confirm the correctness of using neural network technique in estimating the applied mass m(t), with in the transient region as shown in Figure 8.

The constant terms -0.1637, 0.0035, 0.0152, -0.0001 and 0.0053 in the linear equations in Figures 5, 6, 7, 9 and 10, are considered as a mean initial displacement of the weighing platform. To acquire a knowledge from the nature of the responses, a typical experimental fluctuation and initial displacement of the weighing machine to applied mass m(t)=40kg is shown in Figure 8. Obviously the waveform in Figure 8 will be settled down to the steady state value, m(t)=40 kg. The constant terms -0.0001 and +0.0053 in Figures 9 and 10 could be considered as initial system displacements respectively. This implies that if there is any variation during the training and recalling processes, this would cause an error in the overall recalling performance. However, in the cases of a systematic noise, constant environmental noise and multiple displacements, the constant term in the linear equation may have a significant benefit, for example to identify initial system displacement, which it would be considered as a future work.

Further more, pointing out to one of the famous features of the Artificial Neural Network, where during training, ANN is learning the characteristics of a defined system. Hence, any initial displacement caused by systematic or constant environment noise will not have any serious effect on the accuracy of the output response. This is because the measure of the error criteria is a deviation of the slopes from unity; see linear equations shown in Figures 5, 6, 7, 9 and 10. Thus, the precision at the output response depends upon the accuracy of the training process rather than the initial system displacements.

9. CONCLUSION

The aim of the current work is solely to employ a Least-Squares Fit to a straight line together with correlation rate to investigate the accuracy of the application of an Artificial Neural Network technique to estimate an applied mass. This is achieved in a noisy environment while the weighing platform is still in the transient mode. Application of correlation between Artificial Neural Network based on dynamic weight estimation and applied mass, has been successfully carried out for both simulation and experimental time series data. The numerical results which roughly indicate unity magnitudes for both, slops and correlation rates, confirm the correctness of using Neural Network technique in estimating the applied mass m(t).

By concentrating on initial displacement in Figure 8, an interesting research topic rises for the discussion and further work that is to identify the initial system displacement, which is suggested to be considered as a future work.

10. ACKNOWLEDGEMENTS

The co-operation of W.&T. AVERY Ltd in providing access to a weighing platform is gratefully acknowledged. Thanks are due to Dr. M. Danaci and Dr. B. G. Cetiner for useful discussions while they were at The School of Engineering, Cardiff University of Wales, UK.

11. APPENDICES

I - Here the model parameters are defined for the weighing system platform constants K, C; the applied mass m(t), the platform mass mp, the initial platform displacement b_0 and initial velocity b_1. This is done for the case of underdamped (u), critically damped (c) and overdamped (o) respectively.

Underdamped (u) Case

$$q_0 = (m(t) + mp)g/K,\ q_1 = 0.5C/(m(t) + mp),$$

$$q_2 = \sqrt{B_1^{\,2} + B_2^{\,2}},$$

$$q_3 = \omega_d = \sqrt{K(m(t) + mp)^{-1} - q_1^{\,2}},$$

$$q_4 = \tan^{-1}(B_1/B_2),$$

where,

$$B_1 = q_0 - b_0,\ B_2 = b_1 + B_1 q_1/q_3,$$

and ω_d is the natural damped frequency.

Critically Damped (c) Case

$$q_0 = (m(t) + mp)g/K,\ q_1 = 0.5C/(m(t) + mp),$$

$$q_2 = q_0 - b_0,\ q_3 = q_1 q_2 - b_1.$$

Overdamped (*o*) Case

$$q_0 = (m(t) + mp)g/K,$$

$$q_1 = -0.5C/(m(t) + mp) + \omega_d,$$

$$q_2 = -\frac{(q_0 - b_0)q_3 + b_1}{2\omega_d},$$

$$q_3 = -0.5C/(m(t)+mp) - \omega_d,$$

$$q_4 = -\frac{(q_0 - b_0)q_1 + b_1}{2\omega_d}.$$

II - The following steps are involved in constructing and training a MLP network.

(a) Defining the structure of the network (the number of layers and the number of neurones in each layer);
(b) Selecting the learning parameters (learning rate and momentum rate);
(c) Initialising the connection coefficients;
(d) Selecting an input-output pair from the training examples set and presenting it to the network;
(e) Calculating the output values of the neurones in the hidden and output layers;
(f) Comparing the output values of the network with the desired output values and calculating the output errors;
(g) Adjusting the connection coefficients of the network in order to decrease the output errors;
(h) Repeating steps (d) to (g) until the error is acceptable or a predefined number of iterations is completed.

III - For the clarification of the mathematical expressions, we assume that estimated mass, w(n) and applied mass, m(t) are x_i and $y(x_i)$ respectively. So, for any given desired value x_i, the probability P_i for making the observed value y_i, assuming a Gaussian distribution with a standard deviation σ_i for the observations of the actual value $y(x_i)$ is:

$$P_i = \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left\{-\frac{1}{2}\left[\frac{y_i - y(x_i)}{\sigma_i}\right]^2\right\}$$

The probability for making the set of M values of y_i is the product of these probabilities. Similarly, for any estimated values of the coefficients a and b, the probability that we should make the observed set of measurements is:

$$p(a,b) = \Pi\, p_i$$

$$p(a,b) = \prod\left(\frac{1}{\sigma_{i_i}\sqrt{2\pi}}\right)\exp\left[-\frac{1}{2}\sum\left(\frac{\Delta y_i}{\sigma_i}\right)^2\right]$$

Applying the method of maximum likelihood, the best estimation for a and b are those values that maximise the probability, $p(a,b)$. Hence, this can be achieved by maximising the sum in the exponential. Let us define χ^2 to indicate the sum:

$$\chi^2 \equiv \sum\left[\frac{1}{\sigma_i^2}(y_i - a - bx_i)^2\right]$$

The minimum value of the function χ^2 of Equation 8 is one, which yields a value of zero for both the partial derivatives with respect to each of the coefficients,

$$\frac{\partial}{\partial a}\chi^2 = 0 \qquad \frac{\partial}{\partial b}\chi^2 = 0$$

The solution of the concluding equations with respect to the coefficient a and b is,

$$a = \frac{1}{\Delta}\left(\sum x_i \sum y_i - \sum x_i \sum x_i y_i\right)$$

$$b = \frac{1}{\Delta}\left(M\sum x_i y_i - \sum x_i \sum y_i\right)$$

$$\Delta = M\sum x_i^2 - \left(\sum x_i\right)^2$$

12. REFERENCES

1. Ferguson, C., “From Beam to Load Cell”, ***Engineering***, (1974), 202-207.
2. Lolley, R. A., “A Review of Industrial Weighing Systems Part 1”, ***Meas. and Cont.***, (1976), Vol. 9, 411-416.
3. Lolley, R. A., “A Review of Industrial Weighing Systems Part 2”, ***Meas. and Cont.***, Dec. 1976, Vol. 9, 435-439.
4. Norden, K. E., “Electronic Weighing in Industrial Processes”, Granada Publishing, Technical Books, (1984).
5. Shi, W. J., White, N. M. and Brignell, J. E., “Adaptive

Filters in Load Cell Response Correction", ***Sensors and Actuators***, Vol. 37, **(**1993), 280-285.

6. Shu, W. Q., "Dynamic Weighing Under Non-Zero Initial Conditions", ***IEEE Trans. on Inst. and Meas.***, Vol. 42, No. 4, (1993), 806-811.
7. Danaci, M. and. Horrocks, D. H., "A Non-linear Regression Technique For Improved Dynamic Weighing", ***Proc. Eur. Conf. on Circuit Theory and Design***, Istanbul, Turkey, (August 1995), 507-510.
8. Danaci, M., "Signal Processing Techniques Applied to Dynamic Weighing Systems", Ph.D. Thesis, School of Engineering, Cardiff University of Wales UK, (December 1996).
9. Bahar, H. B. and Horrocks, D. H., "Dynamic Weight Estimation Using Artificial Neural Network", ***Artificial Intelligence In Engineering***, Vol. 12, (1998), 135-139.
10. Pandya, A. S. and Macy, R. B., "Pattern Recognition with Neural Networks in C++", CRC Press Inc., (1996).
11. Haykin, S., "Neural Networks, A Comprehensive Foundation", Macmillan College Publishing Company, (1994).
12. Pham, D. T. and Liu, X., "Neural Networks for Identification, Prediction and Control", Springer-Verlag, 2nd Edition, (1995).
13. Masters, T., "Practical Neural Network Recipes in C++", Academic Press, Boston, (1993).
14. Bevington, P. R., "Data Reduction on Error Analysis for the Physical Sciences", McGraw-Hill, 2nd Edition, (1994).
15. Chatfield, C., "The Analysis of Time Series, An Introduction", Chapman & Hall, Fifth Edition, (1996).