

This article was downloaded by: [National Chiao Tung University 國立交通大學]
On: 27 April 2014, At: 17:12
Publisher: Taylor & Francis
Informa Ltd Registered in England and Wales Registered Number: 1072954
Registered office: Mortimer House, 37-41 Mortimer Street, London W1T 3JH, UK



International Journal of Production Research

Publication details, including instructions for authors and subscription information:

<http://www.tandfonline.com/loi/tprs20>

On-line tuning of a single EWMA controller based on the neural technique

C.-T. Su^a & C.-C. Hsu^a

^a Department of Industrial Engineering and Management ,
National Chiao Tung University , 1001 Ta Hsueh Road,
Hsinchu, Taiwan, ROC

Published online: 21 Feb 2007.

To cite this article: C.-T. Su & C.-C. Hsu (2004) On-line tuning of a single EWMA controller based on the neural technique, International Journal of Production Research, 42:11, 2163-2178, DOI: [10.1080/00207540410001661409](https://doi.org/10.1080/00207540410001661409)

To link to this article: <http://dx.doi.org/10.1080/00207540410001661409>

PLEASE SCROLL DOWN FOR ARTICLE

Taylor & Francis makes every effort to ensure the accuracy of all the information (the "Content") contained in the publications on our platform. However, Taylor & Francis, our agents, and our licensors make no representations or warranties whatsoever as to the accuracy, completeness, or suitability for any purpose of the Content. Any opinions and views expressed in this publication are the opinions and views of the authors, and are not the views of or endorsed by Taylor & Francis. The accuracy of the Content should not be relied upon and should be independently verified with primary sources of information. Taylor and Francis shall not be liable for any losses, actions, claims, proceedings, demands, costs, expenses, damages, and other liabilities whatsoever or howsoever caused arising directly or indirectly in connection with, in relation to or arising out of the use of the Content.

This article may be used for research, teaching, and private study purposes. Any substantial or systematic reproduction, redistribution, reselling, loan, sub-licensing, systematic supply, or distribution in any form to anyone is expressly forbidden. Terms & Conditions of access and use can be found at <http://www.tandfonline.com/page/terms-and-conditions>

On-line tuning of a single EWMA controller based on the neural technique

C.-T. SU* and C.-C. HSU

The exponentially weighted moving average (EWMA) controller has been proven to be an effective algorithm in the control the modern manufacturing system. The performance of the EWMA controlled process is based on choosing the correct EWMA gain. Most related research has focused on analysing the optimal EWMA gain in the static condition. The objective was to propose an approach based on the neural technique for on-line tuning of the single EWMA gain. The underlying approach indicated that the network learns very quickly when taking auto-correlation function and sample partial autocorrelation function patterns as the input features. It is shown that the sequence of the EWMA gains, generated by the proposed adaptive approach, converges close to the optimal controller value under several disturbance models, including IMA(1,1), and step and small ramp disturbances. In addition, the approach possesses a superior controlled output performance compared with the previous adaptive system.

1. Introduction

Statistical process control (SPC) is a well-recognized technique for process monitoring. The main assumption of traditional SPC is that successive quality characteristic values should have no correlation with each other. However, modern manufacturing processes exhibit quality data that are serially correlated over time. These traditional control charts will give misleading results in the form of too many false alarms if the data are correlated. To address this problem, an approach called engineering process control (EPC) or automatic process control (APC) is widely used in the chemical and processing industries for variation reduction. The EPC scheme refers to an algorithm that describes how the manipulating variable of a process needs to be adjusted from observation to observation.

Recently, a run-by-run feedback control algorithm called the exponentially weighted moving average (EWMA) controller was popular in semiconductor manufacturing, particularly in the chemical mechanical polishing (CMP) process. The performance of the controlled process output under the EWMA controller depends on setting its gains, which means that incorrectly selecting the weight will have the opposite effect on the controlled process output. When the process environment is static and the system parameters are determined, the perfect controller parameters can be obtained by mathematical modelling technology. Unfortunately, a process environment is usually dynamic in a real manufacturing world. To achieve a better performance in the dynamic system, developing a method of on-line tuning of the EWMA controller parameter is an important issue.

Revision received October 2003.

Department of Industrial Engineering and Management, National Chiao Tung University, 1001 Ta Hsueh Road, Hsinchu, Taiwan, ROC.

*To whom correspondence should be addressed. Fax: 00886 35 722 392

As mentioned above, some previous studies have used statistical techniques such as the least-squares estimate (LSE) (Sastri 1988) or the maximum likelihood estimate (MLE) (Luceno 1995) to estimate the parameters recursively. However, these techniques assumed that the probability distribution is known in advance. Patel and Jenkins (2000) provided another statistical adaptive algorithm to update the EWMA gain by taking the signal-to-noise (SN) ratio. This involves estimating the mean of the output and the mean square of the output, which implies that additional parameters should be chosen at first to estimate them. The present will introduce their adaptive system in more detail below.

In this study, a simple and efficient method based on the neural technique is suggested to tune adaptively the EWMA gain. This proposed approach takes advantage of the fact that no assumption is needed; it constructs input–output relations by learning the historic patterns. The autocorrelation function (ACF) and partial autocorrelation function (PACF) patterns are taken as input features of the training network. The ACF is a well-known tool for identifying the order of the moving average (MA) stochastic process, and the PACF is used to determine the order of the autoregressive (AR) model. Taking another view of both statistics: they are functions of autoregressive moving average (ARMA) parameters, which implies that different combinations of AR and MA parameters will produce distinct ACF/PACF patterns. Based on this idea, a neural network will be trained to estimate on-line the EWMA gain at the next run via ACF/PACF pattern families.

The paper is organized as follows. Section 2 reviews the single EWMA controller, and its effects of incorrectly setting the EWMA gain will be shown. Section 3 introduces a recent adaptive algorithm for the EWMA controller. Section 4 introduces neural network techniques. Section 5 presents the structure of the proposed approach. The idea of using the SACF and PACF patterns to estimate the EWMA gain will be demonstrated in more detail. Section 6 specifies the performance of the off-line-trained network and, by using the Simulink toolbox, implements the proposed structure on-line through three examples. Finally, conclusions are drawn, and future research is suggested in section 7.

2. Single EWMA controller

2.1. Brief review of the single EWMA controller

The EWMA statistic was first suggested by Roberts (1959) for process monitoring, but Roberts referred to it as a geometric moving average (GMA). The use of the EWMA statistic has two distinct purposes (Fatin *et al.* 1990): as control charts (Box and Kramer 1992, Montgomery 1996, Box and Luceno 1997, Chen and Elsayed 2002) and as forecasts (Box and Jenkins 1976, Box *et al.* 1994, Brockwell and Davis, 1996). Recently, the statistic has been used widely for process adjustment purposes (Lucas and Saccucci 1992, Ingolfsson and Sachs 1993, Del Castillo and Hurwitz 1997, Del Castillo 2001, 2002a, Pan and Del Castillo 2001, Fan *et al.* 2002, O'Shaughnessy and Haugh 2002).

In semiconductor manufacturing, EWMA controllers are sometimes called bias tuning controllers (Butler and Stefani 1994). The purpose of EWMA-based controllers is for compensating against disturbances that affect the run-to-run variability in quality characteristics (Del Castillo 2002b). Assume that the relation between the input and output of a manufacturing process can be expressed as follows:

$$e_t = \alpha + \beta u_{t-1} + \varepsilon_t, \quad (1)$$

where e_t is the observed output deviation from target, ε_t is a white noise stochastic process and u_t is the manipulated variable; the parameter α represents the process offset, β is the process gain, and both parameters need to be estimated. Equation (1) implies that all the effects of a change in the compensating variable will be realized at the output, in one time interval. Such a system is called a responsive system (Box and Luceno 1997) and is commonly seen in the discrete part manufacturing. Let b represent an estimate of the gain (β) that can be estimated off-line by fitting the regression model. The single EWMA scheme can be expressed as follows:

$$u_t = -\frac{a_t}{b}, \tag{2}$$

where

$$\begin{aligned} a_t &= \lambda(e_t - bu_{t-1}) + (1 - \lambda)a_{t-1} \\ &= \lambda[e_t - bu_{t-1} + (1 - \lambda)(e_{t-1} - bu_{t-2}) + (1 - \lambda)^2(e_{t-2} - bu_{t-3}) + \dots] \end{aligned} \tag{3}$$

is an estimate of the offset and is computed recursively based on the EWMA statistic with the last measurement data. The previous estimate a_{t-1} and λ are the controller parameters, which can be adjusted to achieve a desired output. Substituting equation (3) into (2) leads to:

$$u_t = -\frac{\lambda}{b} \sum_{j=-\infty}^t e_j. \tag{4}$$

Therefore, the single EWMA controller is a pure integral (I) controller with integral constant $K_I = -\lambda/b$, which is a particular case of the well-known PID controller. A PID control scheme can be expressed as $u_t = -k_p e_t - k_I \sum_{j=0}^{\infty} e_{t-j} - k_D(e_t - e_{t-1})$, where k_p , K_I and K_D are constants. The integral action in the EWMA controller can eliminate offsets or shifts, and provides robustness in the controlled process.

2.2. Effects of incorrectly setting the controller parameter

As mentioned by Smith and Boning (1997), the controlled process output under a higher EWMA weight would return to the target much faster than a lower weight, but it would create more oscillations. Therefore, it is important to select the EWMA parameter carefully. This section will describe in more detail the effect of incorrectly choosing the EWMA gain. Consider a process that can be modelled by:

$$e_t = \alpha + \beta u_{t-1} + N_t, \tag{5}$$

where N_t is the disturbance model one wants to compensate for. Assume it follows an IMA(1,1) stochastic process as follows:

$$(1 - B)N_t = (1 - \theta B)\varepsilon_t, \tag{6}$$

where B is the backshift operator ($B^k N_t = N_{t-k}$) and θ is the moving average parameter. An IMA(1,1) model is widely used for modelling the drift in the discrete manufacturing (Lucas and Saccucci 1992, Box *et al.* 1994). If the single EWMA controller is used to compensate for the disturbance, then the controlled process is as follows:

$$(1 - (1 - \lambda\xi)B)e_t = (1 - \theta B)\varepsilon_t, \tag{7}$$

where $\xi = \beta/b$ is a bias of the gain estimate. It can be seen that the controlled process exhibits an ARMA(1,1) process, and that the stable condition is $|1 - \lambda\xi| \leq 1$.

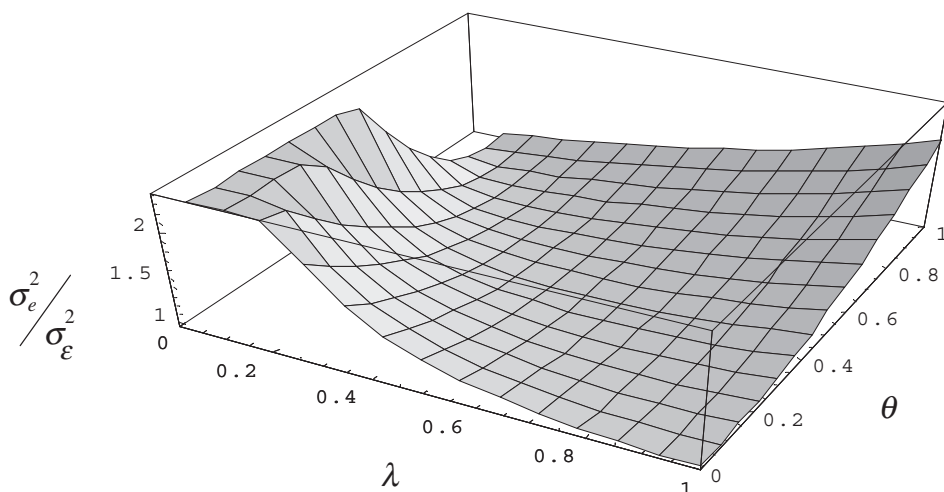


Figure 1. Inflation factor versus λ and θ .

Therefore, the inflation factor of the controlled process will be:

$$\frac{\sigma_e^2}{\sigma_\varepsilon^2} = 1 + \frac{(1 - \lambda\xi - \theta)^2}{1 - (1 - \lambda\xi)^2}. \quad (8)$$

Figure 1 shows the inflation factor ($\sigma_e^2/\sigma_\varepsilon^2$) versus λ and θ given the process gain is known ($\xi = 1$). Consider that the disturbance model follows a white noise process, which implies $\theta = 1$ in equation (6) and a full adjustment to the process ($\lambda = 1$) is used, then the controlled output variance will be inflated twice as much than if there was no adjustment. This is what Deming (1986) meant by ‘tampering with the process’. To achieve the minimum mean square error (MMSE) controlled process output, one should set the controller parameter to be as follows:

$$\lambda^* = 1 - \theta. \quad (9)$$

Although the actual optimal controller parameter in the above equation was already known, it was only an optimal value in the static sense. In practice, the parameter of the disturbance model changes with time, so it is necessary to develop an approach to adjust the controller parameter dynamically in order to obtain a better performance of the controlled process output. Section 3 introduces an adaptive algorithm recently suggested by Patel and Jenkins (2000).

3. Adaptive algorithm

In the sense of an adaptive system, Sastri (1988) used the theory of LSE for sequential parameter detection and revision of the moving average parameter in the IMA(1,1) time series model. Luceño (1995) presented a computer program to choose the EWMA controller parameter in the EPC. Luceño’s algorithm was based on the MLE theory. These above-mentioned algorithms all have a common constraint in that the probability distribution must be known beforehand. Therefore, Smith and Boning (1997) used a neural network as an approximation function to map from the disturbance state (magnitude of linear drift and random noise) of a given process to the corresponding optimal EWMA weights. However, it was a

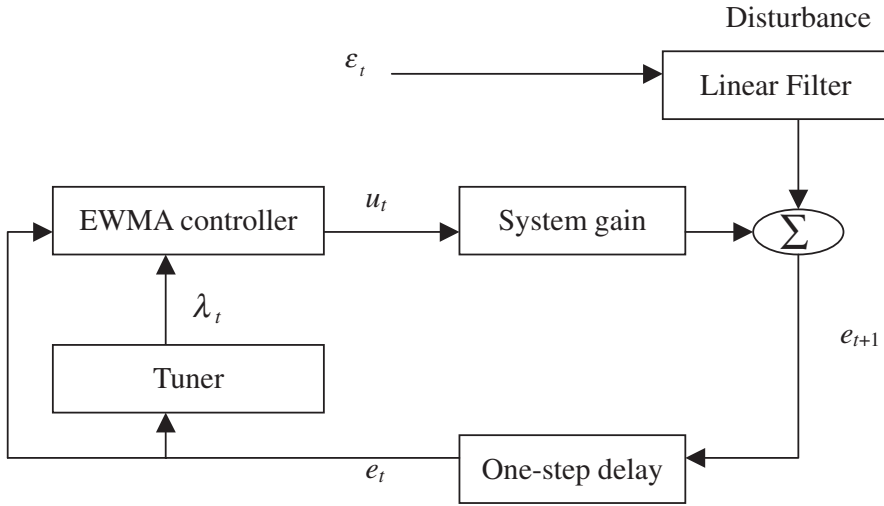


Figure 2. Structure of the adaptive tuning controller.

problem to estimate the slope of the controlled process. Del Castillo and Yeh (1998) presented an adaptive run-to-run multiple-input multiple-output controller for linear and non-linear semiconductor processes. Recently, an adaptive algorithm to estimate the EWMA gain was suggested by Patel and Jenkins (2000). Their objective was to design an automated scheme for optimizing the numerical parameter of the EWMA controller. Figure 2 shows the adaptive EWMA controller block diagram they proposed.

The Patel–Jenkins adaptive algorithm can simply be described as the following system:

$$\begin{aligned}
 \mu_{t+1} &= \mu_t + \tau_t(e_{t+1} - \mu_t) \\
 \zeta_{t+1} &= \zeta_t + \tau_t(e_{t+1}^2 - \zeta_t) \\
 \lambda_t &= \frac{\delta^2 + 4\mu_t^2}{\delta + \mu_t^2 + \zeta_t},
 \end{aligned}
 \tag{10}$$

where $\{\mu_t\}$ is the estimate of the mean of the output and $\{\zeta_t\}$ is the estimate of the mean square of the output. Initial conditions of (μ_0, ζ_0) satisfy $0 \leq \mu_0^2 \leq \zeta_0$. δ is a constant with a very small value that satisfies $0 < \delta < 1$, and $\{\tau_t\}$ is a sequence such that $0 \leq \tau_t < 1$ and satisfies (I) $\lim_{t \rightarrow \infty} \tau_t = 0$ (II) $\sum_{t=0}^{\infty} \tau_t = \infty$ and (III) $\sum_{t=0}^{\infty} \tau_t^2 < \infty$. The form of λ_t in equation (10) intuitively provides a measure of the SN ratio that satisfies $0 \leq \lambda_t \leq 2$. According to the above-mentioned adaptive system, the EWMA control equation can be updated dynamically as follows:

$$u_t = -\frac{\lambda_t}{b} \sum_{j=-\infty}^t e_j,
 \tag{11}$$

where the adaptive parameter λ_t follows the system in equation (10).

4. Neural network techniques

Neural networks are of particular interest because they offer a means of modeling large and complex problems efficiently in which there might be hundreds of predictor variables with many interactions. Neural nets have been used widely in pattern recognition (Su *et al.* 2002), function approximation (Smith and Boning 1997), optimization (Sjoberg and Agarwal 2002) and data clustering (Andrews and Geva 2002). In general, neural networks can be classified into two different categories: feed-forward and feedback (Cheng and Titterington 1994). The present study used the feed-forward network because it is an effective system for learning distinguishing patterns from a body of examples.

The back-propagation learning algorithm is the most commonly used algorithm to train multilayer feed-forward networks by implementing a local gradient-search to minimize the square error between realized and desired outputs. A typical back-propagation neural network always has an input layer, an output layer and at least one hidden layer. There is no theoretical limit on the number of hidden layers, but typically, there will be one or two. Figure 3 shows a three-layer network. Each layer is fully connected to the succeeding layer. The back-propagation algorithm involves forward and backward passes. The purpose of the forward pass is to obtain the activation value; the purpose of the backward pass is to adjust weights according to the difference between the desired and actual network outputs. The above statement can be explained by the following equations.

4.1. Forward pass

The net input to node i for pattern p is:

$$\text{net}_{pi} = \sum_j w_{ij} a_{pj} + b_i \quad (12)$$

$$a_{pj} = \frac{1}{1 + e^{-\text{net}_{pj}}}, \quad (13)$$

where w_{ij} is the weight from unit j to unit i , b_i is a bias associated with unit i , and a_{pj} is the activation value of unit j with sigmoid function for pattern p .

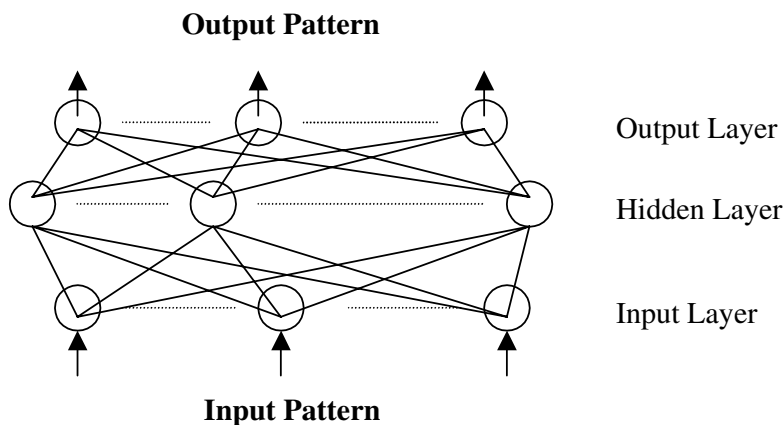


Figure 3. Backpropagation neural network structure.

4.2. Backward pass

The sum of the squares error function is as follows:

$$E_p = \frac{1}{2} \|t_p - o_p\|_2^2, \tag{14}$$

where t_p is the target output for the p th pattern and o_p is the actual output for the p th pattern. By minimizing the errors E_p using the gradient decent method, the weights can be updated using the following equation:

$$\Delta_p w_{ij} = \eta \delta_{pi} a_{pj}, \tag{15}$$

where

$$\delta_{pi} = \begin{cases} (t_{pi} - o_{pi}) o_{pi} (1 - o_{pi}) & \text{if unit } i \text{ is an output unit} \\ o_{pi} (1 - o_{pi}) \left(\sum_k \delta_{pk} w_{ki} \right) & \text{if unit } i \text{ is a hidden unit} \end{cases} \tag{16}$$

and η is the learning rate. In general, a larger learning rate will increase the training speed. However, it may oscillate widely. One way to increase the learning rate without oscillating is to modify equation (15) to the following:

$$\Delta_p w_{ij} = \eta \delta_{pi} a_{pj} + m \Delta_{p-1} w_{ij}, \tag{17}$$

where m is the momentum coefficient ($m \in [0, 1]$) that determines the effect of past weight changes on the current direction of movement in weight space. There is no principle to determine the parameters of η and m ; they are chosen by the neural network trainer via the trial-and-error approach. Concerning the model selection, one of the most useful methods in selecting problems is the cross-validation (CV) method. Breiman and Spector (1992) found 10- and fivefold CV to work better than the leave-one-out method for choosing subsets of inputs in linear regression. Zhang (1993) showed that the delete- d multifold CV (MVC) criterion is asymptotically equivalent to the well-known FPE criterion under a regression model. For a detailed discussion about the CV method, see Witten and Frank (2001).

5. Proposed approach

A methodology for tuning the EWMA controller on-line based on a neural technique is developed in this section. The input features of the neural structure are sample autocorrelation function (SACF) and sample partial autocorrelation function (SPACF). The output unit is an estimate of the EWMA controller parameter at run t . The theoretical autocorrelation function (ACF) at lag h is defined as follows:

$$\rho(h) = \frac{\text{Cov}(e_t, e_{t+h})}{\sigma(e_t)\sigma(e_{t+h})}. \tag{18}$$

Equation (18) is estimated by the SACF as follows:

$$\hat{\rho}(h) = \frac{\sum_{i=1}^{n-h} (e_i - \bar{e})(e_{i+h} - \bar{e})}{\sum_{i=1}^n (e_i - \bar{e})^2}, \tag{19}$$

where n is the sample size and \bar{e} is the sample mean. Also, the theoretical partial autocorrelation function (PACF) is defined as follows:

$$\rho_{hh} = \frac{\text{Cov}(e_t, e_{t+h} | e_{t+1}, e_{t+2}, \dots, e_{t+h-1})}{\sigma(e_t)\sigma(e_{t+h})}, \tag{20}$$

which can be estimated by the SPACF as follows:

$$\hat{\rho}_{hh} = \frac{\hat{\rho}_h - \sum_{j=1}^{h-1} \hat{\rho}_{h-1,j} \hat{\rho}_{h-j}}{1 - \sum_{j=1}^{h-1} \hat{\rho}_{h-1,j} \hat{\rho}_j}, \quad h = 3, \dots \tag{21}$$

where

$$\hat{\rho}_{hj} = \hat{\rho}_{h-1,j} - \hat{\rho}_{hh} \hat{\rho}_{h-1,h-j} \quad h = 2, \dots, \quad j = 1, 2, \dots, h-1 \tag{22}$$

and $\hat{\rho}_{11} = \hat{\rho}_1$, $\hat{\rho}_{22} = (\hat{\rho}_2 - \hat{\rho}_1^2)/(1 - \hat{\rho}_1^2)$.

Opinions of selecting SACF and SPACF statistics to be input features can be described simply from figure 4. Figure 4(a) shows the family of SACF/SPACF (denoted ρ) patterns under the condition of a perfect controlled process up to run t ,

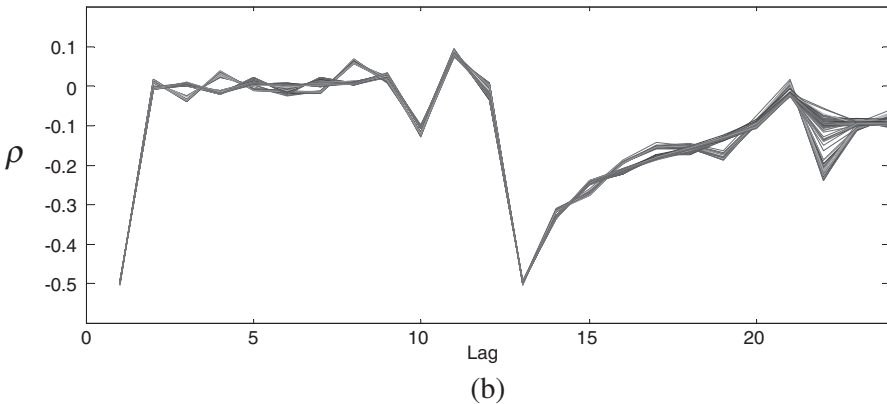
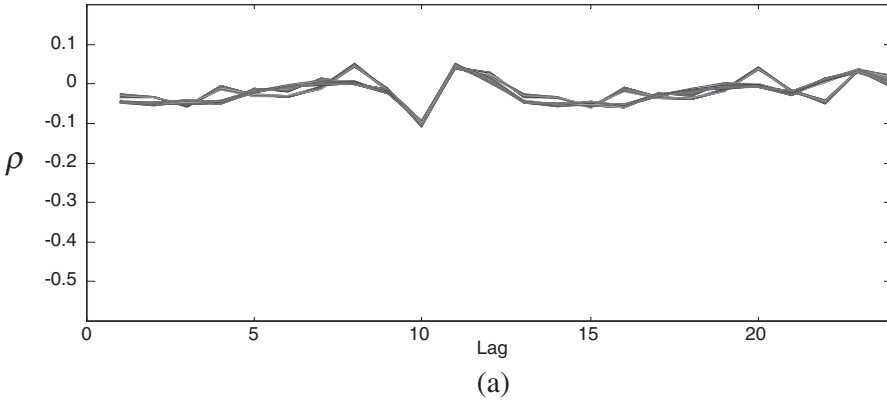


Figure 4. Family of SACF/SPACF patterns: (a) perfect controlled and (b) $\theta = 1; \lambda = 1$.

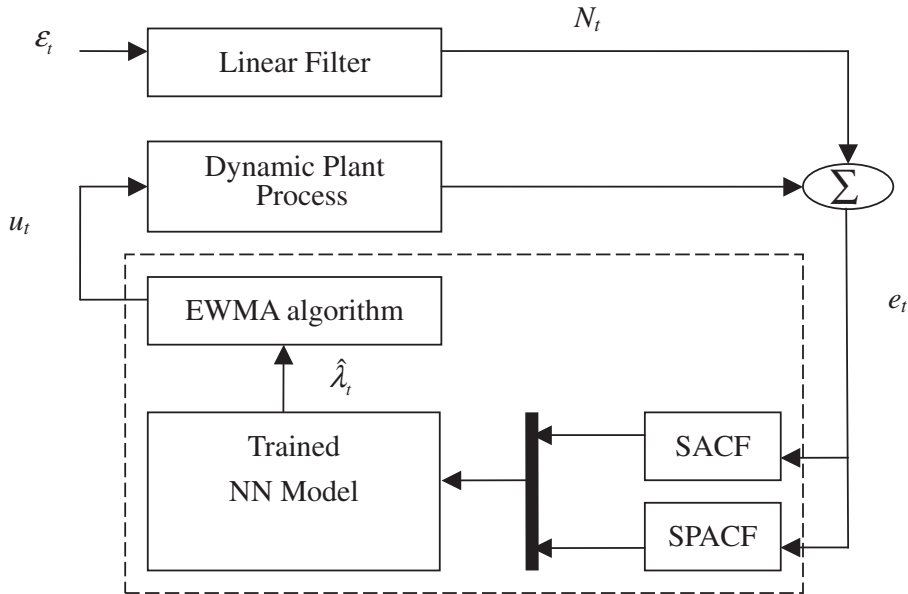


Figure 5. NN-based adaptive EWMA controller.

given the disturbance model obeys equation (6) and the controller parameter follows equation (9). If this type of SACF/SPACF patterns is received, then the controller parameter at next run ($t + 1$) will be set to be the same as the previous run. Figure 4(b) shows the simulated family of SACF/SPACF patterns with parameter $\theta = 1$ (white noise process) in equation (6) and the full adjustment ($\lambda = 1$) in equation (9) up to run t . Obviously, figure 4(b) behaves in a more non-stationary manner than figure 4(a) because of the incorrect way of choosing the controller parameter. If the neural network receives the types of patterns such as shown in figure 4(b) at run t , then it will respond by setting the controller parameter to be zero at the next run to meet the optimal condition. As per the above, the objective is to estimate the controller parameter adaptively through pattern recognition on the SACF and SPACF patterns.

The structure of the proposed adaptive neural network (NN)-based single EWMA controller is shown in figure 5. At first, the controlled quality characteristic was sent to SACF and SPACF blocks to calculate the statistic individually. The combined (denoted as the black bar) SACF/SPACF pattern was then sent to the trained NN model block to estimate the controller parameter for the next run. After estimating the parameter, one should update the single EWMA controller parameter dynamically to provide a better control performance. The proposed methodology was implemented in Section 7 and compared with the method of Patel and Jenkins (2000) introduced in Section 3.

6. Implementation

6.1. Off-line training the neural network

The training data sets were generated by simulating different combinations of the disturbance and controller parameters. There were 121 data sets (i.e. $\theta \in [0, 1]$, $\lambda \in [0, 1]$), which implied that there were 121 SACF/SPACF patterns. Thirty data sets were used as the testing data; the remainder was to be training data. A useful

Run	Learning rate	Momentum constant	Epoch
1	0.35	0.85	54
2	0.15	0.85	28
3	0.35	0.95	51
4	0.15	0.95	37

Table 1. Experimental design results.

guide was provided by Box and Jenkins (1976: 33), who suggested that the size of time series (t) be at least 50 and lags (h) to analyse the series at most $t/4$. Thus, 50 runs were simulated at each simulation, and 12 lags in each SACF and SPCAF pattern were taken.

The considered case included 24 nodes at the input layer and one node at the output layer. The problem in a full-connected neural network was to determine the number of neurons in the hidden layer. A trial-and-error approach was used to determine that a single hidden layer with 22 neurons formed the required structure for the considered problem. To improve the network performance, a 2^2 factorial design was used to find the learning rate and momentum constants (table 1). The factors involved in this design were the learning rate and momentum constants, and the response variable was the number of epochs used to achieve the desired level (0.01) of the root-mean-square (RMS) error. Note that the best network performance (with the smallest number of epochs) was achieved when the learning rate was 0.15 and the momentum constant was 0.85. Figure 6 shows the learning behaviour versus iterations of the selected network structure; it indicates that the network learns very quickly and only required 3393 iterations (about 28 epochs). Thus, the 24-22-1 network structure will be used to implement the NN-based EWMA controller on-line in the following examples.

6.2. Examples of on-line implementation

The toolbox of the Matlab/Simulink version 4.1.1 was used to implement the Patel–Jenkins method and the proposed NN-based adaptive EWMA controller, and then to make a comparison between them. Three examples under different disturbance models, which are commonly encountered in practice, will also be shown, including step IMA(1,1) and the trend disturbance models.

6.2.1. Example 1. Step disturbance model

The step disturbance model can be expressed as follows:

$$N_t = \begin{cases} L & t \geq t_s \\ 0 & t < t_s \end{cases} \quad (23)$$

where L is the level of the step change disturbance and t_s is the time of the disturbance introduced into the process. The tuner parameters in the Patel–Jenkins system (equation 10) were set to be the same as their simulation example: $\sigma_\varepsilon^2 = 1$, $\mu_0 = 0.1$, $\zeta_0 = 1$, $\delta = 10^{-4}$, $\tau = 0.005$, and the step disturbance was introduced at run 50 with $L = 10$. Figure 7(a) shows the controlled process output under the Patel–Jenkins approach; figure 7(b) plots the EWMA gain λ_t through 800 runs.

The off-line trained network was implemented on-line to tune the EWMA controller gain under the step disturbance, which was also introduced at run 50

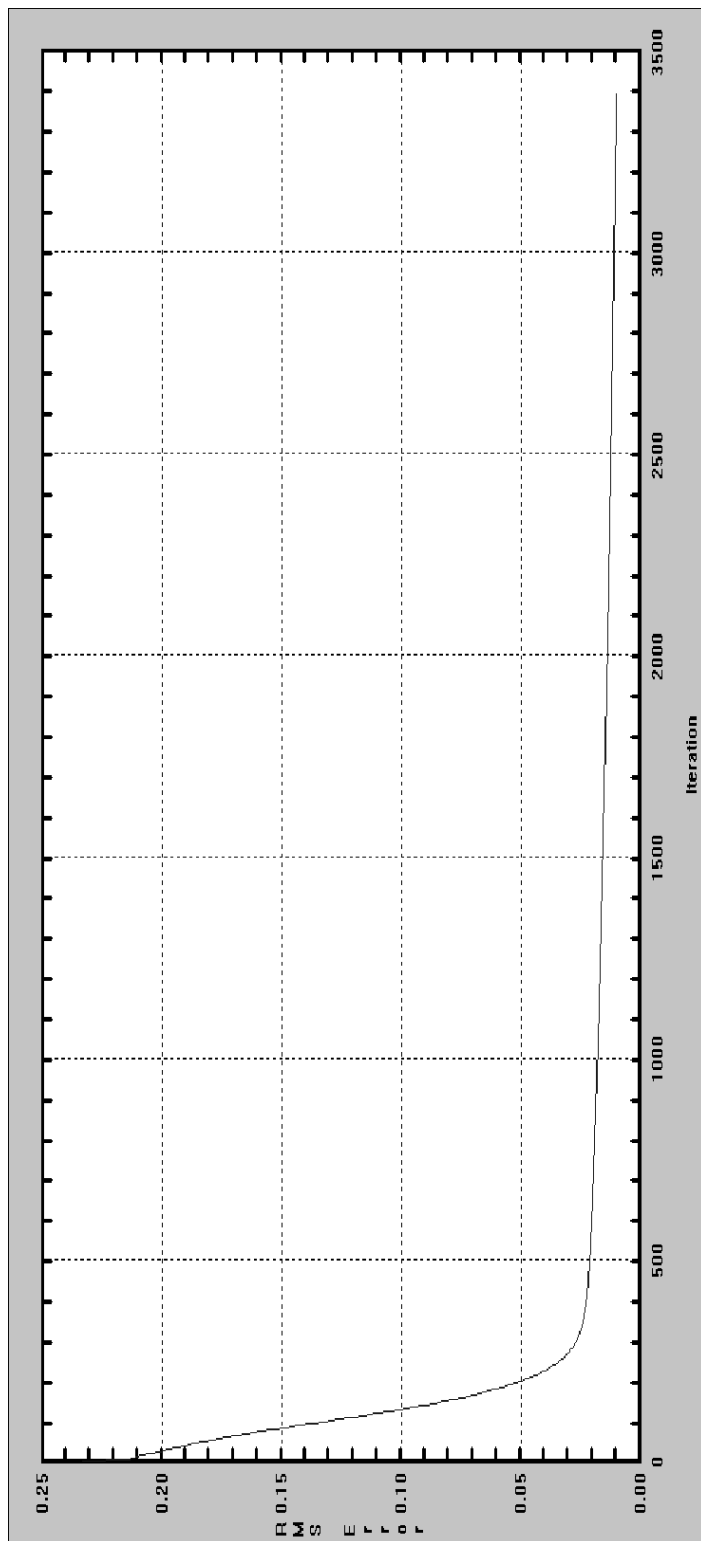


Figure 6. Neural network learning behaviour.

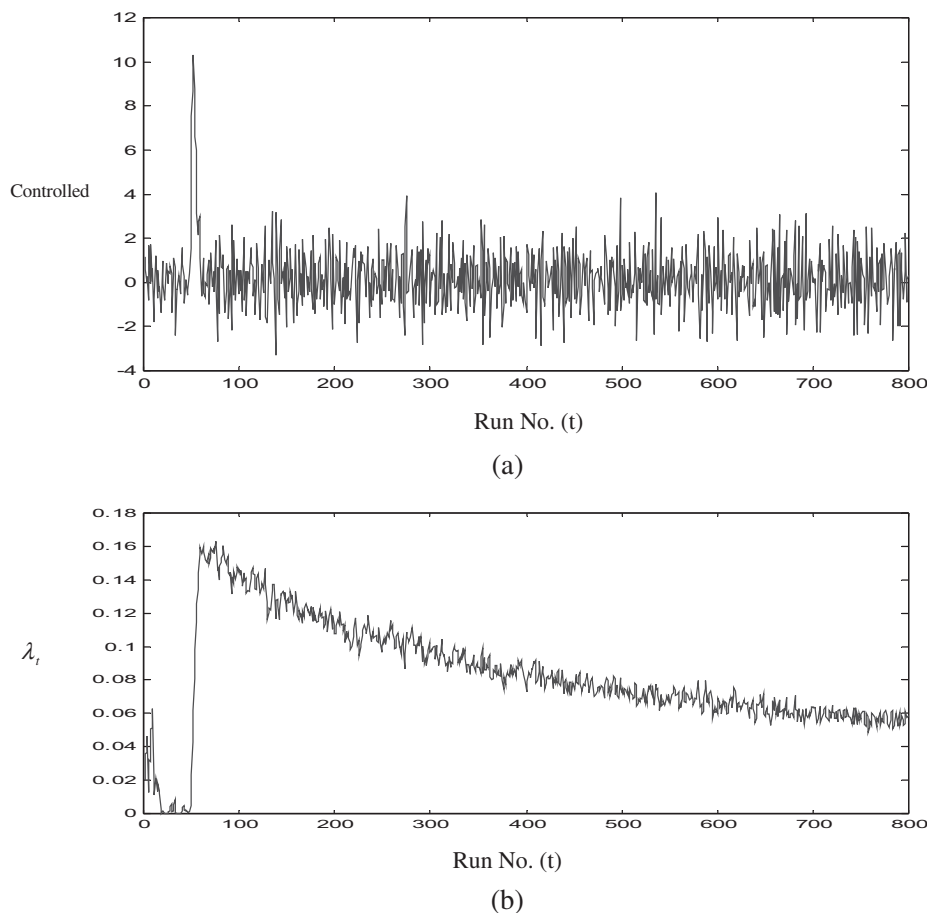


Figure 7. Patel-Jenkins adaptive method: (a) controlled output and (b) EWMA gain.

with magnitude 10. Figure 8(a) shows the NN-based controlled process output; figure 8(b) plots the NN-based EWMA gain λ_t through 800 runs. As expected, λ_t increased on a shift and decreased to a small number with time. The performance of the uncontrolled process measured in the inflation factor ($\hat{\sigma}_e^2/\hat{\sigma}_e^2$) was 5.2795; the controlled inflation factor under the Patel-Jenkins method was 1.8521, and 1.3321 in the NN-based EWMA controller. Therefore, the NN-based adaptive EWMA controller had a superior performance.

6.2.2. Example 2. IMA(1,1) disturbance model

The IMA(1,1) disturbance model is now considered. The moving average parameter $\theta = 0.2$ and $\sigma_e^2 = 1$ was used to simulate the problem. From equation (9), it is known that the optimal controller parameter was $\lambda^* = 1 - \theta$. Assume that the disturbance was introduced at run 50 over 800 runs. Figure 9 shows the EWMA gain under the NN-based adaptive controller. The value of λ_t oscillated along with the optimal controller value (say 0.8). Taking the sample mean of the last 200 runs, it tended to 0.8078, which was close to the optimal controller parameter. The inflation factor of the controlled process output approximated to 1, which implied that the

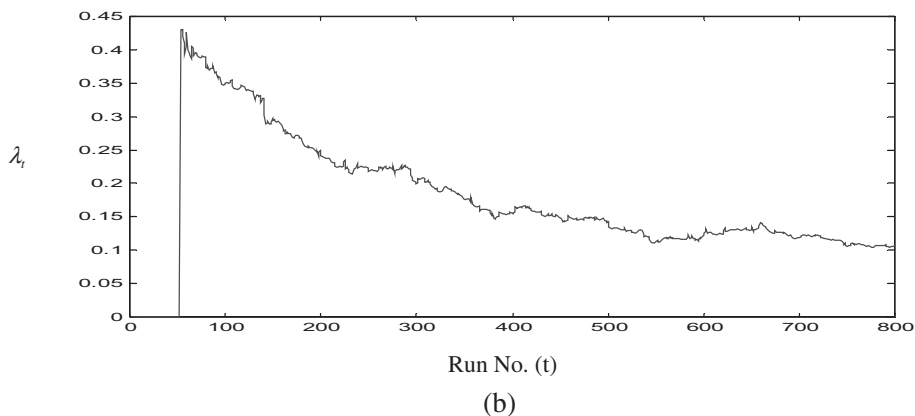
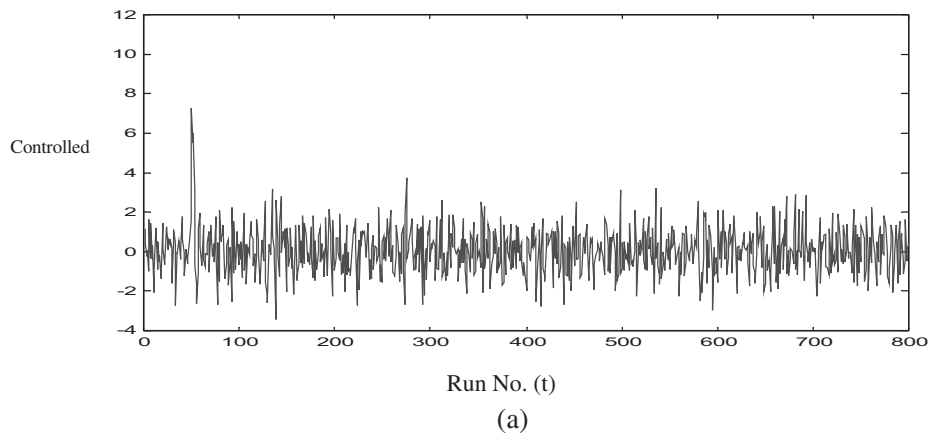


Figure 8. NN-based adaptive method: (a) controlled output and (b) EWMA gain.

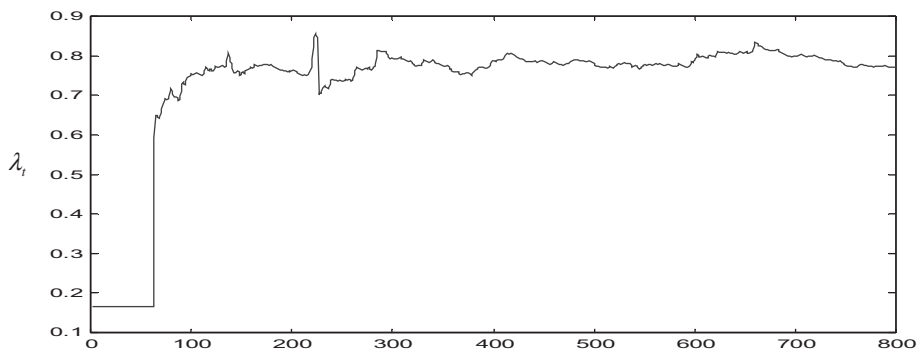


Figure 9. NN-based adaptive EWMA gain under IMA (1,1).

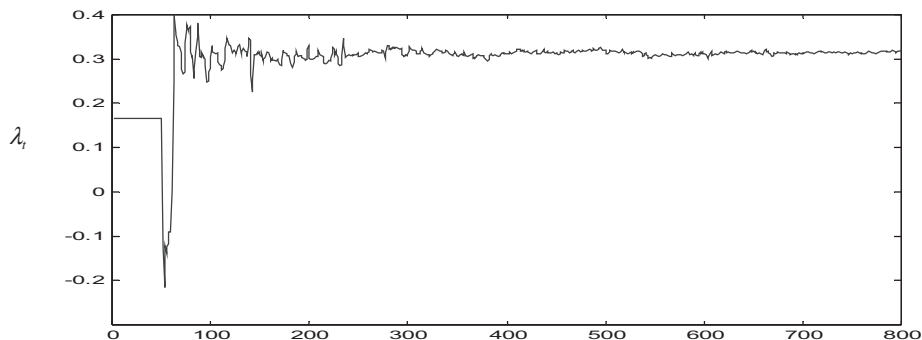


Figure 10. NN-based adaptive EWMA gain under trend disturbance.

proposed adaptive controller had the ability to produce the minimum mean-square-error (MMSE) process output.

6.2.3. Example 3. Ramp disturbance model with small slope

This example considered the ramp disturbance model, which can be expressed as:

$$N_t = \begin{cases} S(t - t_s) & t \geq t_s \\ 0 & t < t_s \end{cases}, \quad (24)$$

where S is the trend rate (slope). The optimal EWMA controller parameter under the trend disturbance can be obtained by solving the following equation:

$$\sigma_\epsilon^2 \lambda^3 - S^2 \lambda^2 + 4S^2 \lambda - 4S^2 = 0 \quad (25)$$

Following the example of Patel–Jenkins, the trend disturbance was introduced, with $S = 0.1$, $\sigma_\epsilon^2 = 1$ at run 50. Figure 10 shows the EWMA gain under the NN-based adaptive method. The value of λ_t oscillated along with the optimal controller value (say 0.3061). The inflation factor under Patel–Jenkins was 2.0914, but only 1.3825 under the proposed method.

7. Conclusion

A method for dynamically tuning the single EWMA controller was proposed. The proposed methodology was based on the pattern recognition of the SACF/SPACF patterns by training the neural network. The behaviour of the off-line-trained network showed that the network learns quickly with input features being SACF/SPACF patterns. It has been shown that the proposed approach has a superior performance over the Patel–Jenkins adaptive algorithm through three implementations. From Example 1, as expected, the NN-based EWMA gain increased on a transient period and then decreased to a small number on a long run. Example 2 showed that the NN-based adaptive EWMA controller approximated to the MMSE controlled performance under the IMA(1,1) stochastic process. From Example 3, it was observed that the EWMA gain behaved close to the optimal controller parameter when the ramp disturbance with a lower slope existed in the process. Although the proposed methodology was implemented via simulation, it is nevertheless anticipated to improve the performance of the EWMA controller in an actual process.

Further research can use the proposed techniques properly to pick the controller parameters dynamically for the double or triple EWMA controller, which can compensate for the server slopes of the trend disturbance model. Also, similar research can be extended to the multiple-input multiple-output system.

References

- ANDREWS, R. and GEVA, S., 2002, Rule extraction from local cluster neural nets. *Neurocomputing*, **47**, 1–20.
- BOX, G. E. P. and JENKINS, G. M., 1976, *Time Series Analysis, Forecasting and Control*, 2nd edn (San Francisco: Holden-Day).
- BOX, G. E. P., JENKINS, G. M. and REINSEL, G., 1994, *Time Series Analysis, Forecasting and Control*, 3rd edn (Englewood Cliffs, NJ: Prentice-Hall).
- BOX, G. E. P. and KRAMER, T., 1992, Statistical process monitoring and feedback adjustment—a discussion. *Technometrics*, **34**, 251–285.
- BOX, G. E. P. and LUCEÑO, A., 1997, *Statistical Control—By Monitoring and Feedback Adjustment* (New York: Wiley).
- BREIMAN, L. and SPECTOR, P., 1992, Submodel selection and evaluation in regression: the X-random case. *International Statistics Review*, **60**, 291–319.
- BROCKWELL, P. J. and DAVIS, R. A., 1996, *Introduction to Time Series and Forecasting* (New York: Springer).
- BUTLER, S. W. and STEFANI, J. A., 1994, Supervisory run-to-run control of a polysilicon gate etch using *in situ* ellipsometry. *IEEE Transactions on Semiconductor Manufacturing*, **7**, 193–201.
- CHEN, A. and ELSAYED, E. A., 2002, Design and performance analysis of the exponentially weighted moving average mean estimate for processes subject to random step changes. *Technometrics*, **44**, 379–389.
- CHENG, B. and TITTERINGTON, D. M., 1994, Neural networks: a review from a statistical perspective. *Statistical Science*, **9**, 2–54.
- DEL CASTILLO, E., 2001, Some properties of EWMA feedback quality adjustment schemes for drifting disturbance. *Journal of Quality Technology*, **33**, 153–166.
- DEL CASTILLO, E., 2002a, Closed-loop disturbance identification and controller tuning for discrete manufacturing processes. *Technometrics*, **44**, 134–141.
- DEL CASTILLO, E., 2002b, *Statistical Process Adjustment for Quality Control* (New York: Wiley).
- DEL CASTILLO, E. and HURWITZ, A., 1997, Run-to-run process control: literature review and extensions. *Journal of Quality Technology*, **29**, 184–196.
- DEL CASTILLO, E. and YEY, J. Y., 1998, An adaptive run-to-run optimizing controller for linear and nonlinear semiconductor processes. *IEEE Transactions on Semiconductor Manufacturing*, **11**, 285–295.
- DEMING, W. E., 1986, *Out of Crisis* (Cambridge, MA: Massachusetts Institute of Technology, Center for Advanced Engineering Studies).
- FAN, S.-K. S., JIANG, B. C., JEN, C. H. and WANG, C. C., 2002, SISO run-to-run feedback controller using triple EWMA smoothing for semiconductor manufacturing processes. *International Journal of Production Research*, **40**, 3093–3120.
- FATIN, W. F., HAHN, G. J. and TUCKER, W. T., 1990, Discussion. *Technometrics*, **32**, 1–29.
- INGOLFSSON, A. and SACHS, E., 1993, Stability and sensitivity of an EWMA controller. *Journal of Quality Technology*, **25**, 271–287.
- LUCAS, J. M. and SACCUCCI, M. S., 1992, Exponentially weighted moving average control schemes: properties and enhancements. *Technometrics*, **32**, 1–29.
- LUCEÑO, A., 1995, Choosing the EWMA parameter in engineering process control. *Journal of Quality Technology*, **27**, 162–168.
- MONTGOMERY, D. C., 1996, *Introduction to Statistical Quality Control*, 3rd edn (New York: Wiley).
- O'SHAUGHNESSY, P. and HAUGH, L., 2002, EWMA-based bounded adjustment scheme with adaptive noise variance estimation. *Journal of Quality Technology*, **34**, 327–339.

- PAN, R. and DEL CASTILLO, E., 2001, Identification and fine tuning of closed-loop processes under discrete EWMA and PI adjustments. *Quality and Reliability Engineering International*, **17**, 419–427.
- PATEL, N. and JENKINS, S., 2000, Adaptive optimization of run-to-run controllers: the EWMA example. *IEEE Transactions on Semiconductor Manufacturing*, **13**, 97–107.
- ROBERTS, S. W., 1959, Control chart tests based on geometric moving averages. *Technometrics*, **1**, 239–250.
- SASTRI, T., 1988, An adaptive estimation. *IIE Transactions*, **20**, 176–185.
- SJOBERG, J. and AGARWAL, M., 2002, Trajectory tracking in batch processes using neural controllers. *Engineering Applications of Artificial Intelligence*, **15**, 41–51.
- SMITH, T. and BONING, D., 1997, A self-tuning EWMA controller utilizing artificial neural network function approximation techniques. *IEEE Transactions on Components, Packaging, and Manufacturing Technology—Part C*, **20**, 121–132.
- SU, C.-T., YANG, T. and KE, C.-M., 2002, A neural-network approach for semiconductor wafer post-sawing inspection. *IEEE Transactions on Semiconductor Manufacturing*, **15**, 260–266.
- WITTEN, I. H. and FRANK, E., 2001, *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations* (San Francisco: Morgan Kaufmann).
- ZHANG, P., 1993, Model selection via multifold cross-validation. *Annals of Statistics*, **21**, 299–311.