A Self-Tuning EWMA Controller Utilizing Artificial Neural Network Function Approximation Techniques

Taber H. Smith, Member, IEEE, and Duane S. Boning, Member, IEEE

Abstract—Recent works have shown that an exponentially weighted moving average (EWMA) controller can be used on semiconductor processes to maintain process targets over extended periods for improved product quality and decreased machine downtime. Proper choice of controller parameters (EWMA weights) is critical to the performance of this system. This work examines how different process factors affect the optimal controller parameters. We show that a function mapping from the disturbance state (magnitude of linear drift and random noise) of a given process to the corresponding optimal EWMA weights can be generated, and an artificial neural network (ANN) trained to learn the mapping. A self-tuning EWMA controller is proposed which dynamically updates its controller parameters by estimating the disturbance state and using the ANN function mapping to provide updates to the controller parameters. The result is an adaptive controller which eliminates the need for an experienced engineer to tune the controller, thereby allowing it to be more easily applied to semiconductor processes.

Index Terms—Adaptive control, artificial neural network, EWMA, process control.

I. INTRODUCTION

RUN BY RUN control methods are receiving attention as a means to improve and maintain the performance of modern semiconductor manufacturing processes, particularly in the areas of chemical-mechanical polishing and plasma etching [1]–[12]. The exponentially weighted moving average (EWMA) controller [5] is one method for controlling processes in the presence of noise. The performance of this system is highly dependent on the choice of the EWMA controller parameters (EWMA weights). The ability to dynamically update the EWMA weights is important for systems which exhibit process dynamics that are unaccounted for in their representative process model. For example, noise and drift estimates may change from previous estimates upon machine start-up. In these situations, the optimal EWMA weight depends on the state of the system, and dynamic weight selection is essential for maximum controller performance. This work contributes a self-tuning EWMA controller which utilizes artificial neural network (ANN) function approximation to dynamically adjust the EWMA weight on-line.

Chemical-mechanical polishing (CMP) has been shown to exhibit characteristics such as rate drift (which changes with different sets of consumables), pad “rebound,” and varying amounts of process noise. These characteristics make the CMP process an appropriate example application for this controller.

Section II briefly reviews the CMP process, while Section III reviews the EWMA controller, highlights the effect of the EWMA weight value, and describes a method for choosing an optimal value for the EWMA controller. Section IV discusses the impact of the disturbance state on the optimal EWMA weight and describes a method for creating a function mapping from the disturbance state to the optimal EWMA weighting scheme. An outline of the self-tuning EWMA controller and simulations to demonstrate its effectiveness are presented in Sections V and VI, respectively. Finally, conclusions and future work are discussed in Section VII.

II. THE CMP PROCESS

Chemical mechanical polishing is recognized to be of critical importance to high performance interconnect technology [15]. In the CMP process, the wafer is affixed to a wafer carrier and pressed face-down on a rotating platen holding a polishing pad. A slurry with abrasive material held in an alkaline or acidic suspension is dripped onto the rotating platen during polish. The carrier and platen rotate at variable speeds, so that the process removes material at the surface of the wafer through a combination of mechanical and chemical action. A typical process goal is to achieve “global” planarization (across several mm) by preferential removal of “high” material on the wafer. The planarization of dielectric dioxide layers between multilevel metallization steps is one common application.

Due to poor process understanding, degradation (wear-out) of polishing pads, inconsistency of the slurry, and the lack of in-situ sensors, the control of CMP is difficult. In addition to difficulties achieving a reliable film thickness because of changing removal rates over time, the within-wafer uniformity of the polish is difficult to achieve and maintain. In other work, we have presented control experiments [1]–[4] as well as integrated control systems for CMP [7]–[9]. Here we focus on run by run process control using the EWMA controller and the proper choice of its weighting parameter to address the needs of poor process understanding in CMP. The product characteristics of concern are the removal rate (corresponding

Manuscript received January 7, 1997; revised April 18, 1997. This work has been supported in part by SEMATECH, by ARPA under Contract N00174-93-C-0035, and by the NSF/SRC Engineering Research Center for Environmentally Benign Semiconductor Manufacturing.

The authors are with the Microsystems Technology Laboratories, Massachusetts Institute of Technology, Cambridge, MA 02139 USA. 

Publisher Item Identifier S 1083-4400(97)05278-9.
Fig. 1. Baseline CMP experiment illustrating drift in removal rate and nonuniformity.

to a controlled amount of oxide polished during the step) and the within-wafer uniformity of that removal rate across the wafer. The change in removal rate and nonuniformity for a typical uncontrolled or baseline oxide polish process (with a fixed recipe) is shown in Fig. 1. Off-line experimental designs were used to generate a linearized multivariate model for the run by run controller of the form

\[ \hat{y}_t = \hat{A} \hat{x}_t + \hat{c}_t \]

where \( \hat{y}_t \) is a vector consisting of removal rate and nonuniformity, \( \hat{A} \) is a constant gain matrix, \( \hat{x}_t \) is a vector consisting of speed, pressure, force, and profile, and \( \hat{c}_t \) is an offset vector. Additional descriptions of baseline and control experiments are presented in [1]–[4].

III. THE EWMA CONTROLLER

The control scheme uses a dynamic model (on a lot by lot basis) as shown in Fig. 2. In this controller, the offset term \( \hat{c}_t \) is updated recursively by an exponentially weighted moving average (EWMA) based on the error between model prediction and measurement

\[ \hat{c}_t = \alpha (y_t - \hat{A} \hat{x}_t) + (1 - \alpha) \hat{c}_{t-1} \]

where \( \alpha \) is a diagonal EWMA weight vector, \( y_t \) is the measured output from the current run, and \( \hat{c}_{t-1} \) is the offset term used on the previous run. The EWMA controller has been shown to provide good control of a process while being relatively insensitive to noise.

A. Responsiveness as a Function of the EWMA Weight

The following simulation results demonstrate the effect of different values of the EWMA weight. In these simulations, both the process plant and the control model are assumed to be linear, with gain coefficients equal to 300 and offsets equal to zero. The process plant has normally distributed white noise with a mean of zero and a variance of 15. A shift disturbance is introduced at run 20, with a magnitude of 100. Fig. 3 shows the two controlled outputs against the uncontrolled process, one with a large value of the EWMA weight, \( \alpha = 0.9 \), and the other with a small value, \( \alpha = 0.2 \). We see that after the shift, the \( \alpha = 0.9 \) response returns to target much faster than the \( \alpha = 0.2 \) response. Higher values of the EWMA weight increase the noise in the controlled output, as can be seen by comparing the two responses before the shift was introduced. In order
to compare the effectiveness of the controller we utilize the mean of the squared error (MSE) from the process target. The MSE for the response with $\alpha = 0.2$ is 953 and the MSE with $\alpha = 0.9$ is 497. An optimum value for the EWMA weight parameter appears to exist since the MSE increases again to 509 with $\alpha = 1$.

**B. The Optimal Weight for a Fixed Disturbance State**

It is not readily apparent how to theoretically calculate the optimal value of the EWMA parameter vector, $\alpha$, under closed-loop control for a multiple input multiple output (MIMO) system. This is because different targets, different amounts of model mismatch, process noise, drifts, and shifts (expected number and size) all change the optimal value of the EWMA weight for closed-loop control. The combination of these different quantities will be referred to as the disturbance state. While the theoretical solution has not been found, it is possible to empirically determine an optimal weight. Given any known disturbance state, the optimal EWMA weight can be found by simulating the process for several EWMA weights and extracting the optimal weight vector. We now simplify this procedure to determine the optimal EWMA weight vector as a function of a reduced disturbance state. In order to do this, we need to determine how each disturbance affects the optimal EWMA weight. In this work, we focus on a linear drifting process buried in white noise (similar to that of a CMP process). Therefore, the effects of large shifts are not considered in our space of allowable disturbances. Next, we examine the effect of process drift, process noise, model mismatch and different targets on the optimal EWMA weights.

![Fig. 3. The effect of (a) small and (b) large EWMA weights on the control of a process shift.](image)

![Fig. 4. Mean squared error (MSE) of the controlled output as a function of the EWMA weight.](image)
A. The Impact of Drift and Process Noise on the Optimal Weights

Many processes, including CMP, exhibit a linear drift disturbance (each output increases or decreases at a constant rate) in their baseline process. As we saw earlier, the size of the EWMA weight determines the responsiveness of the EWMA controller to a shift disturbance. The response to a drift disturbance is similar, but a process drift causes continual change in the process and thus the controller is constantly trying to catch up. Fig. 5 shows two responses from the process above with the shift replaced by a drift and the noise removed. The upper plot shows the response with an EWMA weight of 0.2 whereas the lower plot provides the response with an EWMA weight of 0.9. This confirms that the higher the EWMA weight, the smaller the resulting steady-state offset [12]. This might lead us to choose a high EWMA weight, but when process noise exists, this may not be the best approach. Fig. 6 shows the controlled response of the EWMA controller to a noisy process with the drift removed. The upper plot is for an EWMA weight of 0.2 and the lower for a weight of 0.9. Here we see that the noise in the output is significantly increased by choosing a high EWMA weight. In fact, the mean squared error (MSE) for the process is composed of the sum of the mean error (offset from target) squared and the output variance (MSE from the process mean). With our assumption that the process is a linear drifting process buried in white noise, the only source of mean offset error is due to the process drift. The increase in output variance would be due to the controller's response to the underlying process noise. This concept is demonstrated in Fig. 7. We see that when the EWMA weight is large, the combined squared error is largely due to the controller overreacting to the process noise.

B. Model Error Impact on Optimal Weights

The actual model mismatch, taken here to be the controller model gain minus the true gain of the process divided by the true gain, may be different than originally estimated. The effect of model error is two-fold: first, it causes an initial offset which must be compensated for by the EWMA controller, and second, it increases the noise in the controlled output.

The initial transient caused by model mismatch can be seen by comparing two simulations with a fixed drift, EWMA weight, and a fixed amount of additive noise. Fig. 8(a) shows the controlled run with no model error and Fig. 8(b) shows the controlled run with a coefficient error of 10%. Notice that the run with zero model error is initially on target whereas the 10% model error case has a large initial offset. The rate at which this offset is compensated for has a large impact on the MSE. Therefore, the optimal EWMA weight with model error should be higher than that with no model error in order to compensate for the initial offset more quickly.

If we consider only the second half of each simulated run, we observe only the steady-state effect of model error on the optimal EWMA weight. Simulations were performed on the drifting process described above over a range of model error and EWMA weights. The weight which provides the minimum MSE in the second half of the simulation is selected as optimal. The optimal steady-state weights are plotted as a function of the amount of model error in Fig. 9. When the model error is negative, meaning a smaller controller gain than the true process gain, the controller makes larger than necessary changes in the recipe to compensate for errors in the output.
This increases the noise in the controlled output, which in turn requires more filtering of the updates to the control model in order to be optimal. Therefore, more and more negative model error requires lower and lower optimal EWMA weights. If the model error is positive, meaning the actual gain of the process is lower than the controller gain, then the controller recipe changes are too small to compensate for the drift disturbance. Therefore, recent measurements need more weight to increase the controller response. These two properties are demonstrated in Fig. 9: negative model error requires a lower weight than with no model error and positive model error requires a higher weight than with no model error.
Fig. 8. The effect of model error on the control of a process drift. (a) shows the response with no model error and (b) shows the response with 10% model error.

Fig. 9. Steady-state optimal EWMA weight for a fixed drift (5 units/run) and fixed process noise (mean zero, variance 15) as a function of model error (with a plant coefficient of 300 and offset of zero).

From our discussion at the beginning of the section, we know that model error (positive or negative) causes an initial offset which must be compensated for and this increases the optimal EWMA weight. We now complete the picture by combining these effects and repeating the simulations above with the initial transients included in the calculation of the MSE. The results are shown in Fig. 10, where they are plotted against the steady-state case above. We see here that the initial offset proportionally increases the necessary optimal weights over the steady-state case, except in the area where the model error, and thus the initial offset, is small. The slight difference in the optimal weight near zero is due to the slight differences in the noise used for the two cases. The first case uses the noise from the entire run and the second uses the noise from only the second half of the run. We will see later that we can eliminate this problem by averaging several simulations.

In an industrial setting, a significant offset would often trigger a statistical process control (SPC) alarm and the model would be reoptimized before any long-term operation of the machine were to take place. We therefore concentrate on the situation where the process model error has increased over time, while the tool continued to operate without reoptimiza-
tion, and the EWMA controller has gradually corrected for model changes. Throughout the remainder of our discussion, we will focus on the steady-state effects of model error, after any initial transients have died away.

Consider now the steady-state effects of model error on the mean offset squared, variance, and total MSE as a function of the EWMA weight for two cases of model error. These are shown in Fig. 11. The upper plot provides the zero model error case and the lower shows the positive model error case (7%). Here we see that the added model error in the lower plot has effectively increased the squared offset error for only the lowest weights and decreased the process variance over much of the larger weights. This has resulted in the optimal weight shifting from 0.7 in the zero model error case to 0.8 in the 7% model error case. Whether the drift term or the noise term dominates determines whether the optimal weight increases or decreases.

An important observation allows us to simplify our optimal weight selection procedure. We find that the effect of model error is to alter the amount of drift (or offset) and noise measured in the output. When addressing the issue of the optimal EWMA weight, the effect of model error may be combined with the drift and noise disturbances to form a reduced set of disturbances; the process drift and noise measured at the output are sufficient to determine an optimal weight.

C. Target Effect on Optimal EWMA Weights

In order to understand the effect of target variation on the optimal EWMA weight, we consider the example above and determine the optimal EWMA weights based on the entire run (including initial transients). We begin with the case of zero model error. Simulations were performed for several sets of targets and it was found that the optimal EWMA weights are unaffected by different targets when there is no model error. When model error is present, however, the changes in target translate into the size of the initial offset through the controller’s solution of the next recipe (using an incorrect model). Therefore, different targets affect the size of the initial offset due to model error.

The effect of the target is generally small but whether it increases or decreases the optimal weight depends on the specifics of the process, target, and control model. In particular, when there are more inputs than outputs it is possible for an inaccurate control model to predict the correct desired output for a given set of inputs. In such a case, the system appears to have a small amount of model error which translates into a small initial offset. The result is a smaller optimal weight; a high weight is not needed to shift the process back to the steady-state point. More commonly, the solution of the control model may be far from the new desired target. In this case there will be a large initial offset, which will require a higher weight to bring the process back to the steady-state. In the long run, the EWMA will shift the process model such that the error induced by model inaccuracy will reach a steady state.

Aside from the transient effect of target changes on the optimal weight vector, a question remains regarding the long term effects caused by different targets. This question can be answered by determining the optimal EWMA weights based on the MSE of the latter part of long simulation runs (when the target and model error transients have died out). By performing several simulations for different targets we find that different targets have no long-term effect on the optimal EWMA weights. That is to say, different targets do not change
the optimal EWMA weight. Since different targets translate into different initial offsets due to model error, which we know from our previous discussion have no effect on the steady-state optimal weights, this result should not be surprising.

D. An Optimal EWMA Weight Map

The previous section demonstrated that the disturbances (drift, noise, model error, and different targets) in a drifting processes can be reduced to a disturbance state of two disturbances (noise and drift measured in the process output), which determine the optimal EWMA weight. In order to create a general map from this reduced disturbance state to the optimal EWMA weights, we simulate the process over a grid of possible disturbance state combinations and extract the optimal EWMA weights. These simulations are then repeated and averaged to removed the specifics of any particular noise sample. Thus, the simulations performed for a given process (as in Section III-B) are repeated at every point on a grid in the reduced disturbance state. The response surface of the optimal EWMA weight for removal rate is shown in Fig. 12 for measured drift ranging from $-200 \text{ A/min per run}$ to $+200 \text{ A/min per run}$ and for measured process noise with standard deviation ranging from $0 \text{ A/min}$ to $600 \text{ A/min}$. We see that the main valley (low EWMA weight) is centered around the zero drift axis, and increasing drift magnitude (positive and negative) causes an increase in the optimal EWMA weight. In addition, increasing noise causes a corresponding decrease in the optimal EWMA weight. The response surface for the nonuniformity output is similar.

V. THE EWMA CONTROLLER WITH AN ANN OPTIMAL WEIGHT ESTIMATOR

Recently, updating of the EWMA weight for use in open-loop tracking of a process using adaptive Kalman filtering has been shown in [16]. Our goal with the following framework is to provide a similar ability for closed-loop control. Due to the complex disturbance state in the closed-loop case, the work presented here is more empirical in nature. The strategy for the artificial neural network (ANN) EWMA weight estimator is to utilize an ANN model of the mapping from the disturbance state to the optimal EWMA weight vector to dynamically update the EWMA controller during the control run. This idea is outlined in Fig. 13.

An ANN is trained to learn the mapping from the disturbance state to the corresponding optimal EWMA weight vector using the empirically determined optimal EWMA weights generated in Section IV-D. The disturbance state is estimated on-line and fed into the neural network. The ANN provides an excellent approximation tool in this case. As the number of outputs increases, the number of points in this disturbance state increases at a rapid rate; the ANN can often learn the structure with a small sample of the points over the grid. Once trained, the ANN provides the optimal value of the EWMA weight given the estimated disturbance state. Updates to the EWMA weighting parameters can be made continuously or periodically. It does, however, take several runs to build up good estimates of the process noise and standard deviation. To solve this dilemma, the updating is started after a delay time, usually ten to 15 runs.

A. Training the Neural Network

A multilayer perceptron (MLP) ANN was trained, using the Levenburg–Marquardt algorithm, to learn the mapping from the disturbance state to the optimal EWMA weight. The neural net was trained with an MSE $< 0.02$ for each optimal weight in the training set. Training the ANN using data over a large region of noise and drift values where limits on the EWMA weights were reached allowed the ANN to learn the leveling off at the top. This is crucial for proper performance of the system if the drift or noise fall outside the trained region. For actual implementation of the architecture shown in Fig. 13, bounds were added to the output of the ANN to ensure the zero and one limits (although these were found to be unnecessary). These may be lowered if it is felt that
the system is prone to instability. Overtraining of the ANN is a serious problem for EWMA weight estimation because fluctuations in the EWMA weights could cause unnecessary additive noise, improper tracking of process drift, or unstable control action. Therefore, care must be taken to ensure a smooth response surface. The optimal weight surface is shown in Fig. 14 for removal rate. The nonuniformity surface is similar [3].

B. Estimating the Disturbance State

The ability to properly estimate the process noise and the underlying drift size is imperative to the successful implementation of the self-tuning EWMA controller. Measuring the amount of process noise is fairly simple, as long as the process remains under control. Assuming the process is properly controlled and enough sample points are available, the process noise is estimated using a sample standard deviation.

Noise in the estimate of drift is a problem. If the size of the drift is small (relative to the noise), and is mistaken to be large (due to noise), the neural net would provide a large value of the EWMA weight. However, when there is a large amount of noise, a low value of the EWMA weight is desired since it decreases the noise due to over-control. To solve this problem we use a separate EWMA of the change in the process, with a very small weight, since we are looking for small changes in the disturbance state and we desire a measure of drift which is insensitive to noise. By choosing a very small EWMA weight for our estimate of the process drift, we make the assumption that the drift will change relatively slowly during the process; certainly, an EWMA with a small weight would be unable to track a rapidly changing drift.

VI. SIMULATIONS OF THE SELF-TUNING EWMA-ANN CONTROLLER

In the following simulations, the initial values for the EWMA weights were optimized for the baseline process shown in Fig. 1. Here we show that the EWMA weights optimized for the baseline process will adapt with a varying amount of noise and underlying drift. These simulations are compared based on their MSE. Results are summarized in Table I for all the simulations performed. In all cases, the EWMA weight adaptation was started on run 10, or wafer 100 (each run is taken to be ten wafers), to allow the estimates of process drift and noise to settle out before adaptation began. This, not a process shift, caused the weights to abruptly change at run 10.

The control of the baseline process with and without adaptation of the EWMA weights result in nearly identical performance (minimal difference in the MSE’s). This is expected since the initial weights used in both cases were optimized for this baseline process. Therefore, the dynamic system performs little adaptation, relative to the (0,1] range of possible EWMA weights, as shown in Fig. 15.

The improved performance of the self-tuning EWMA controller is demonstrated by the 9% decrease in MSE in the second example, where the process has a much smaller drift and increased noise in the removal rate output. The dynamic values of the EWMA weights are shown in Fig. 16, where we see that the removal rate weight decreases from its predetermined value as soon as the system estimates that there is no underlying drift. The decrease in the EWMA weight for removal rate reduces the noise added to the controlled response by the controller.

The next simulation shows the control of a process with a large drift and low noise in removal rate. The dynamic weights are provided in Fig. 17. In this case, the EWMA weight for removal rate increases to compensate for the large drift. The MSE for the adaptive controller is 30.7% better than in the fixed EWMA weight case.

As a final demonstration, we show an equivalent experiment with the nonuniformity (which has remained relatively steady for the previous experiments) by decreasing the noise from its usual high value and increasing the drift. The dynamic weights, provided in Fig. 18, show how the EWMA weight for nonuniformity nearly doubles. In addition, the MSE is improved by 38.7%.

VII. CONCLUSION

We have demonstrated that it is possible to quantify and distinguish between good and bad values for the EWMA weight. We have shown there exists a reasonable technique for determining optimal EWMA weights, and an ANN pro-
vides a sufficient method for mapping the disturbance state to a correct choice for the optimal EWMA weights. Finally, a self-tuning EWMA controller using an ANN approximation of the empirically determined optimal EWMA weights provides the ability to dynamically update the weights. This removes the need for operators to tune the weights, simplifies the implementation process, and improves performance.
Fig. 17. Dynamic EWMA weights for the high drift/low noise removal rate process.

Fig. 18. Dynamic EWMA weights for the high drift/low noise nonuniformity process.
Future work is needed to demonstrate this controller on an actual process. The use of this technique to properly choose the weights of double EWMA controllers, such as predictor corrector control [10], could possibly improve performance of these controllers as well.

REFERENCES


Taber H. Smith (M’97) received the B.S. degree from the Rochester Institute of Technology, Rochester, NY, in 1994 and the M.S. degree from the Massachusetts Institute of Technology, Cambridge, in 1996, where he is currently pursuing the Ph.D. degree.

His principle research interests focus on distributed systems for design and manufacturing technologies as well as statistical and intelligent control of semiconductor processes.

Mr. Smith is a member of Tau Beta Pi and Phi Kappa Phi.

Duane S. Boning (S’83–M’91) received the S.B. degrees in electrical engineering and in computer science in 1984, and the S.M. and Ph.D. degrees in electrical engineering in 1986 and 1991, respectively, all from the Massachusetts Institute of Technology (MIT), Cambridge.

From 1991 to 1993 he was a Member of Technical Staff, Texas Instruments Semiconductor Process and Design Center, Dallas, TX, where he worked on semiconductor process representation, process/device simulation tool integration, and statistical modeling and optimization. He is currently an Associate Professor in the Electrical Engineering and Computer Science Department at MIT. His research focuses on variation modeling and control in semiconductor processes, with special emphasis on chemical-mechanical polishing and plasma etch. Additional interests include tools and frameworks for process and device design, network technology for distributed design and fabrication, and computer integrated manufacturing. He is an Associate Editor for the IEEE TRANSACTIONS ON SEMICONDUCTOR MANUFACTURING.

Dr. Boning is a member ofEta Kappa Nu, Tau Beta Pi, Sigma Xi, and the Association of Computing Machinery. He was an NSF Fellow from 1984 to 1989, and an Intel Graduate Fellow in 1990.