AN EFFICIENT RUN-TO-RUN CONTROLLER: ESET

Chang Zhang / Advisor: Prof. John S. Baras
zchang@isr.umd.edu
(301) 405-6578
Institute for Systems Research and
Department of Electrical and Computer Engineering
University of Maryland, College Park, MD, 20742

Abstract

The EWMA run-to-run (RtR) controller is easy to deploy and can effectively deal with small disturbances. But it can not compensate for shifts (step disturbances) and large model errors well. The common solution to such a problem is to use the SPC to detect the occurrence of a shift, and then remove the cause of the shift or apply a rapid mode. In this paper, a novel approach to handle shifts and large model errors is proposed: the SVR-MOVE controller is used to compensate for a shift when it is detected. The SVR-MOVE controller is very efficient to handle shifts and large model errors. On the other hand, a well-tuned EWMA controller may outperform a coarsely tuned SVR-MOVE controller when the disturbances are smooth drifts. It is much easier to tune the parameters of the EWMA controller than that of the SVR-MOVE controller. Our new controller, named ESET, has the advantage of both controllers. Simulation shows that the ESET outperforms the EWMA controller and the SVR-MOVE controller in most situations.

Keywords

Run-to-run control, EWMA, set-valued method, ellipsoid approximation, SVR-MOVE controller.

I. Introduction

The Exponentially Weighted Moving Average (EWMA) method is widely used in run-to-run (RtR) control of semiconductor processes for its simplicity and efficiency to compensate for small disturbances such as smooth drifts [1]. But when the initial process model is coarse and large step disturbances (shifts) are involved in a process, the EWMA controller can not handle them well without proper modifications. Model prediction methods such as the (Prediction Correction Controller) PCC [3] can not compensate for such kind of noises either, since the shifts are usually irregular and hard to predict. The common method to deal with a step disturbance is to use statistical methods to decide whether the step disturbance happens. Once a shift is detected, the cause of the shift is investigated and removed, or the magnitude of the step disturbance is estimated and a rapid mode is applied to the controller. Searching for the cause of the shift may be time-consuming and expensive. In the rapid mode approach, the constant term of the linear EWMA process model or even the whole model is modified. Modifying the constant term may be inefficient and may cause overshoots [1]. Modifying the whole model needs a lot attention since an inappropriate sensitivity parameter may significantly worsen the process. Sometimes, a local experiment is employed at the same time and a weighted least-squares regression is used to find the new model [2].

In this paper, we propose a novel scheme to handle shifts and model errors: Using the set-valued RtR controller with ellipsoid approximation [4, 5] to compensate for the large deviation when it is detected. The ellipsoid algorithm based RtR controllers are robust to model errors and large disturbances [4, 5]. Ellipsoids are used to approximate the feasible parameter sets. The process parameters that are robust to various disturbances are identified within these bounding ellipsoids. One important ellipsoid algorithm based RtR controller is the SVR-MOVE (Set-Valued RtR controller with Modified Optimal Volume Ellipsoid approximation) [4, 5]. The SVR-MOVE controller is very effective to handle large model errors and large disturbances [4, 5]. When a shift happens in a process, the controller can return the process output on target quickly in a few runs.

Design of the SVR-MOVE controller is complex, since we have multiple parameters to tune and have to solve a constrained optimization problem. Though more tunable parameters give us more freedom than less tunable parameters, it may be more time-consuming for us to find proper parameters of the SVR-MOVE controller. A well-tuned EWMA controller may outperform a coarsely tuned SVR-MOVE controller when the disturbances are small. Moreover, it is much easier to design an EWMA controller than the SVR-MOVE controller. Therefore, it leads us to develop a controller with the advantage of both controllers:

- The complex task of tuning the parameters of the SVR-MOVE controller should be avoided for small drifts. The SVR-MOVE controller is concentrated on handling irregular shifts and model errors.
- The EWMA controller should be fully tuned to deal with drifts.

We call the controller *ESET.*

II. Design of the ESET

Due to the space limitation, we will not introduce the design of the SVR-MOVE controller and the EWMA controller in this paper. We recommend readers to material in [1, 4, 5]. The structure of the ESET controller is shown in Figure 1.

At the beginning, the ESET works in the EWMA mode. The SVR-MOVE module is dormant. We choose a threshold parameter (we can also use the SPC) to detect the occurrence of a large deviation. The threshold parameter can be equal to 3σ or other values, depending on the property of the process and noises. Once the threshold is exceeded, the ESET is switched into the SVR-MOVE mode. After the large deviation is compensated, the ESET will switch back to the EWMA mode. One problem in the design of the ESET is how to make smooth transition between two different modes. It is easy to switch from the EWMA mode to the SVR-MOVE mode. We just let the constant term of the model in the SVR-MOVE module be equal to that of the EWMA module. Because the SVR-MOVE module is robust to model errors and step disturbances, it will return the process output on target quickly. When the deviation is less than the threshold, the ESET will switch back to the EWMA mode. Assume that the latest recipe by the SVR-MOVE module is X_M, and the EWMA model is given by $y_a = b^T X_a + a$, where b is the sensitivity parameter and a is the constant term. Then after the transition, the EWMA model is updated by:

$$a := (Y - b^T X_M) \tag{1}$$

where Y is the process target. Because the constant term of the model used in the EWMA module of the ESET is updated by the new recipe X_M that can stabilize the process after the shift, the transition from the SVR-MOVE controller to the EWMA controller is smooth. The adjustment of the recipe from the SVR-MOVE module to the EWMA module is consistent too.

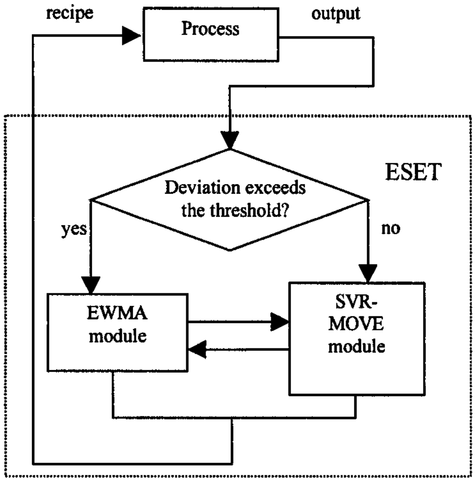

Figure 1. Structure of the ESET

The basic algorithm is as follows.

```
In the beginning, set token=0 and employ the EWMA module to control the process;
If deviation>threshold then
    If token==0 then
        Token:=1;
        Let the constant term in the model of the SVR-MOVE module be equal to that of the EWMA controller.
    End
    The SVR-MOVE module is employed.
Else if token==1 then
        Update the constant term of the EWMA module by equation (1).
        Token:=0
    End
    The EWMA module is employed
End if
```

In the algorithm, the *token* is used to avoid duplicate update of the process models after the transition between the SVR-MOVE mode and the EWMA mode.

III. Simulations

The new controller is tested in a Photoresist process. The process model is as the following [6].

$$T = -13814 + \frac{2.54\cdot 10^6}{\sqrt{SPS}} + \frac{1.95\cdot 10^7}{BTE\sqrt{SPS}} - 3.78BTI - 0.28SPT - \frac{6.16\cdot 10^7}{SPS} \tag{2}$$

where T is the resist thickness in *Angstroms*. SPS is the spin speed in *RPM*, BTE the baking temperature in *degrees Celsius*, BTI the baking time in *seconds*, and SPT

the spin time in *seconds*. They are the inputs to the process, which are confined to the following bounds:

$4500 < SPS < 4700$
$105 < BTE < 135$
$20 < BTI < 100$
$15 < SPT < 90$

The target Y is fixed at Y=12373.621 *Angstroms*.

After changing process variables and sampling at time instant (run) k, we can simplify equation (2) to a second order nonlinear process.

$$\begin{aligned} T_k = & -13814 + 2.54\cdot 10^6 u_{1,k} + 1.95\cdot 10^7 u_{1,k}u_{2,k} \\ & -3.78u_{3,k} - 0.28u_{4,k} - 6.16\cdot 10^7 {u_{1,k}}^2 \end{aligned} \qquad (3)$$

where $u_{1,k} = 1/\sqrt{SPS}$, $u_{2,k}$=1/BTE, $u_{3,k}$=BTI, and $u_{4,k}$=SPT. Equation (3) is assumed to be the real underlying process model. The observed process output y_k is given by:

$$y_k = T_k + d\cdot k + v_k$$

where $d=3$ is the drift value in each run and v_k stands for other noise. The disturbances d and v_k are assumed unknown and the controllers will try to maintain the output of the process on target by post-measurements.

The ESET will be compared with the EWMA controller and the SVR-MOVE controller. We define the Mean Square Deviation (*MSD*) as $MSD = \sum_{k=1}^{K} (y_k - Y)^2 / K$, where Y is the fixed process target and K is the total number of runs in a simulation. We perform thirty independent simulations with the same distributed random variables for each simulation. The average $\overline{MSD} = MSD/30$ will be used as the performance metric. We make the comparison in the following three scenarios.

Scenario 1. Drifts and Gaussian noises.

Assume that we have the perfect knowledge of the process model. In this scenario, only drifts and Gaussian noises exist. The Gaussian noises have zero means and variances 9. One of the simulations is shown in Figure 2. In the figure, the weight of the EWMA controller is equal to 0.5. The weight of the EWMA module in the ESET is also equal to 0.5.

The $\overline{MSD}$ of the uncontrolled process is 324.53; the $\overline{MSD}$ of the SVR-MOVE controller is 18.04, and the $\overline{MSD}$s of the EWMA controller and the ESET for different weights are listed in Table 1.

Table 1. $\overline{MSD}$ *s of the EWMA and ESET in scenario 1*

Weight	0.1	0.3	0.5	0.7	0.9
$\overline{MSD}$ of EWMA	33.31	13.32	11.71	12.92	14.04
$\overline{MSD}$ of ESET	33.07	13.32	11.71	12.92	14.04

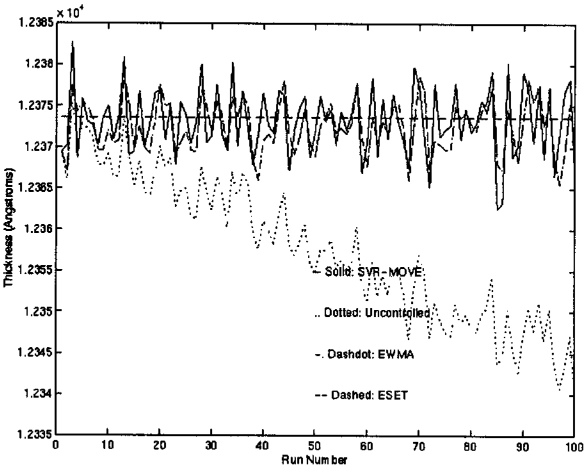

Figure 2. Drifts and Gaussian noises.

In this case, both the EWMA controller and the ESET outperform the SVR-MOVE controller with weights in the range [0.3, 0.9]. In this range, the $\overline{MSD}$s of the ESET are equal to that of the EWMA controller, since only the EWMA module is used in the ESET. When the weight is 0.1, the $\overline{MSD}$ of the ESET is slightly smaller than that of the EWMA controller, since the SVR-MOVE module is employed by the ESET for a deviation larger than the threshold.

Scenario 2. A step disturbance, drifts and Gaussian noises.

It is also assumed that we have the perfect knowledge of the process model. A step disturbance will happen at run 30 by changing the underlying model parameters. The change is assumed unknown to all the controllers. One simulation result is shown in Figure 3. The weights of the EWMA controller and the ESET are equal to 0.5 in the figure.

The $\overline{MSD}$ of the SVR-MOVE controller is 39.77 and the $\overline{MSD}$ of the uncontrolled process is 3844. The $\overline{MSD}$s of the EWMA controller and the ESET for different weights are shown in Table 2.

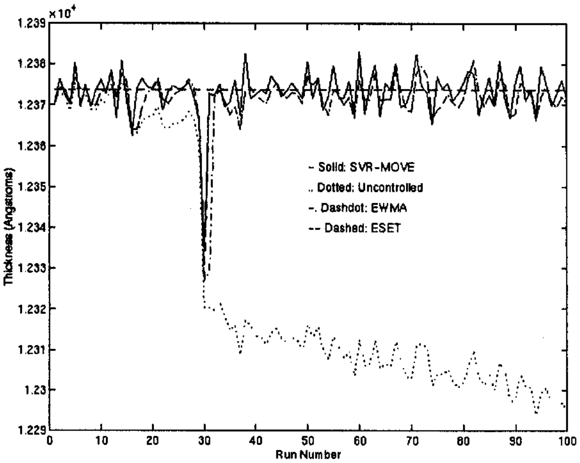

Figure 3. Step disturbance, drifts and Gaussian noises.

Table 2. $\overline{MSD}$ *s of the EWMA and ESET in scenario 2*

Weight	0.1	0.3	0.5	0.7	0.9
$\overline{MSD}$ of EWMA	307.3	105.9	74.60	65.22	66.55
$\overline{MSD}$ of ESET	56.35	35.50	33.69	34.15	37.03

In this case, the ESET with the weight 0.5 performs best. The ESET with the weight in the range [0.3, 0.9] beats the SVR-MOVE controller. At the same time, the SVR-MOVE controller produces much less $\overline{MSD}$ than the EWMA controller with all possible weights.

Scenario 3. A large model error, a step disturbance, drifts, and irregular noises.

In practice, the underlying process model is usually unknown. Suppose that the initial process model used by the controllers is:

$$\hat{T}_k = -13825 + 2.55\cdot 10^6 u_{1,k} + 1.93\cdot 10^7 u_{1,k}u_{2,k} - 2.69u_{3,k} - 0.3u_{4,k} - 6.15\cdot 10^7 u^2_{1,k}$$

and the observed process output is given by:

$$y_k = T_k + d\cdot k + v_{1,k} + v_{2,k} + v_{3,k} + v_{4,k}$$

where $v_{1,k}$ is a Gaussian variable with zero mean and variance 9, $v_{2,k}$ is a random variable that is the product of two Gaussian variables, $v_{3,k}$ is a uniformly distributed random variable in the range [-1,1] and $v_{4,k}$ is a random variable that is the product of a Gaussian variable and a uniformly distributed variable. Moreover, a step disturbance will happen at run 30 by changing the underlying model parameters. One simulation result is shown in Figure 4. The weights of the EWMA controller and the ESET are 0.5.

The $\overline{MSD}$ of the SVR-MOVE controller is 475.57, the $\overline{MSD}$ of the uncontrolled process is 66480. The $\overline{MSD}$ s of the EWMA controller and the ESET for different weights are listed in Table 3.

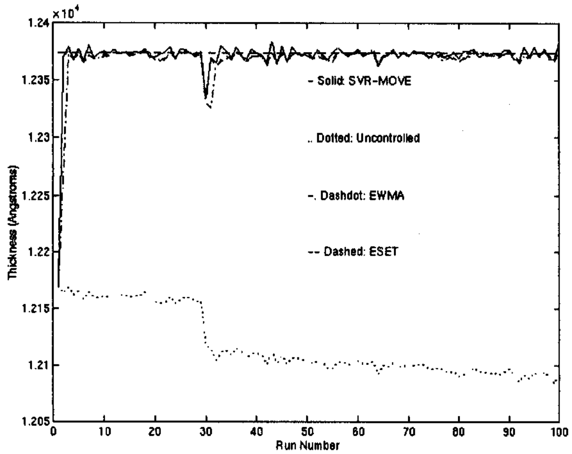

Figure 4. Large model error and irregular noises

Table 3. $\overline{MSD}$ *s of the EWMA and ESET in scenario 3*

Weight	0.1	0.3	0.5	0.7	0.9
$\overline{MSD}$ of EWMA	2089	862.0	664.4	635.5	648.0
$\overline{MSD}$ of ESET	490	474.2	473.1	474.3	471.5

In this case, the ESET controller with the weight in the range [0.3, 0.9] performs better than the SVR-MOVE controller. The $\overline{MSD}$ of the SVR-MOVE controller is much less than that of the EWMA controller with all possible weights.

IV. Conclusions

In this paper, a novel RtR controller named ESET is proposed to effectively compensate for drifts and shifts in a semiconductor process by combining the EWMA controller with the SVR-MOVE controller. The ESET overcomes the shortcomings of the EWMA controller and the SVR-MOVE controller, and fully embodies their advantages. When the disturbances are smooth drifts, the EWMA module of the ESET is employed and can be tuned easily to deal with the disturbances; when a large deviation happens, the SVR-MOVE module is called into the action to return the output on target quickly. From preliminary comparison results, one can see that the ESET with a proper weight (0.5) outperforms both the EWMA controller and the SVR-MOVE controller impressively in almost each situation. Because shifts may rarely happen in a process, the EWMA module is employed by the ESET most of the time. Hence, the average complexity of the ESET may be even less than that of the SVR-MOVE controller. When there is a constant linear drift, the ESET can also be combined with the PCC controller that may completely removes the drift. Another possible improvement is to apply the learning technique [7] to dynamically adjust the weight of the EWMA module in the ESET online.

References

[1] E. Sachs, A. Hu, and A. Ingolfsson, "Run by run process control: combining SPC and feedback control", IEEE Trans. Semiconductor Manufacturing, vol. 8, pp. 26-43, Feb. 1995.

[2] J. A. Stefani, S. Poarch, S. Saxena and P. K. Mozumder, "Advanced process control of a CVD Tungsten reactor", IEEE Trans. Semiconductor Manufacturing, vol. 9, no. 3, pp. 366-383, 1996.

[3] S. W. Butler and J. A. Stefani, "Supervisory run-to-run control of Polysilicon gate etch using in situ Ellipsometry", IEEE Trans. Semiconductor Manufacturing, vol. 7, no. 2, pp. 193-201, 1994.

[4] C. Zhang, H. Deng, J. S. Baras, "The set-valued run-to-run controller with ellipsoid approximation", AEC/APC 2000, vol. 1, pp. 167-178, 2000.

[5] C. Zhang, H. Deng, J. S. Baras, "Comparison of run-to-run control methods in semiconductor manufacturing", AEC/APC 2000, vol. 3, pp. 889-896, 2000.

[6] S. Leang, C. J. Spanos, "Statistically based feedback control of photoresist application", IEEE/SEMI advanced semiconductor manufacturing conference, pp. 185-190, 1991.

[7] R. S. Sutton, "Learning to predict by the methods of temporal differences", Machine Learning, vol. 3, pp. 9-44, 1988.