Next-generation Virtual Metrology for Semiconductor Manufacturing: A Feature-based Framework

Kerul Suthar[a], Devarshi Shah[a], Jin Wang[a], Q. Peter He[a*]

[a]*Department of Chemical Engineering, Auburn University, Auburn AL 36849, USA*
[*]corresponding author, e-mail: *qhe@auburn.edu*

Abstract

In semiconductor manufacturing, virtual metrology (VM), also known as soft sensor, is the prediction of wafer properties using process variables and other information available for the process and/or the product without physically conducting property measurement. VM has been utilized in semiconductor manufacturing for process monitoring and control for the last decades. In this work, we demonstrate the shortcomings of some of the commonly used VM methods and propose a feature-based VM (FVM) framework. Unlike existing VM approaches where the original process variables are correlated to metrology measurements, FVM correlates batch features to metrology measurements. We argue that batch features can better capture semiconductor batch process characteristics and dynamic behaviors. As a result, they can be used to build better predictive models for predicting metrology measurements. FVM naturally addresses some common challenges that cannot be readily handled by existing VM approaches, such as unequal batch lengths and/or unsynchronized batch trajectories. A simulated and an industrial case studies are used to demonstrate the effectiveness of the proposed FVM method. We discuss how to generate and select features systematically, and demonstrate how feature selection affects FVM performance using a case study. Finally, the capabilities of FVM in addressing process nonlinearity is investigated in great details for the first time, which helps establish the theoretical foundations of the proposed framework for the semiconductor industry.

Keywords: semiconductor manufacturing, virtual metrology, process monitoring, statistics pattern analysis, batch feature, feature space

1 Introduction

In semiconductor manufacturing, a wafer undergoes hundreds of different steps to yield the final product. After a processing step, typically a few (1–3) wafers within a lot are measured at the metrology station, and this sampled metrology data represent the whole lot. However, this methodology using the traditional off-line metrology tools (*e.g.*, ellipsometer, atomic force microscope (AFM)) becomes insufficient when the device dimensions continue to decrease and the lot-to-lot process control is being increasingly replaced with the wafer-to-wafer (W2W) control. W2W control requires metrology measurements of every wafer and it has been proposed to use the integrated metrology (IM) sensors at the processing tool to provide such measurements (Lensing and Stirton, 2006). However, issues such as impact on throughput, increase in cycle time, and higher cost make IM less attractive in many process environments. On the other hand, virtual metrology (VM) technology (also known as soft sensor in process industry) has been suggested as an alternative to 100% wafer measurement to support W2W control (Gill et al., 2010; He et al., 2012; Hung et al., 2007). Because machine data are usually sampled much more frequently compared to metrology data, and machine data are instantly available compared to delays often associated with metrology tools, an accurate VM can significantly improve process monitoring and control performance by providing real-time predicted metrology data.

One of the most important factors that need to be considered when implementing any VM for industrial applications is the level of data pre-processing required. Data pre-processing has direct and significant impact on the deployment and maintenance of the VM. Generally speaking, fewer data pre-processing steps and/or more automated data pre-processing steps lead to more sustainable method in a production environment. To address this challenge, in this work, we propose a feature-based VM (FVM) framework based on the statistics pattern analysis (SPA) process modeling and monitoring framework we proposed previously (He and Wang, 2011; Wang and He, 2010). The most significant difference between the proposed VM approach and other existing approaches is that instead of extracting correlations between process variables and metrology measurements, the proposed method extracts the correlations between process features and metrology measurements to build VM models. By doing so, the proposed method can not only eliminate most data pre-processing steps, but also provide superior prediction performance. It is worth noting that we have tested SPA as a VM technology on a plasma etch process previously (He et al., 2012). However, in that work, the features were limited to process variable statistics and the mechanisms behind SPA were not explored. One major contribution of present work

is to extend and generalize the method to include any features, not just statics, but also non-statistical process features, such as process knowledge based landmark features (Wold et al., 2009); profile-driven features (Rendall et al., 2017); geometry based features (Wang et al., 2015). In addition, this is the first time we provide a detailed discussion on how features are generated and selected systematically, and we demonstrate how feature selection affects FVM performance using a case study. Finally, the capabilities of FVM in addressing process nonlinearity is investigated in great details for the first time, which helps establish the theoretical foundations of the proposed framework for the semiconductor industry.

The rest of the paper is organized as follows: Section 2 provides a brief review on existing VM approaches. Section 3 presents the FVM framework. Section 4 discuss the performance metrics for comparing different VM methods. Section 5 uses a simulated case study to demonstrate the performance of the proposed FVM framework, which is compared with several other VM approaches. In this section we also investigate how FVM addresses process nonlinearity to achieve superior performance compared to other existing VM approaches. Section 6 demonstrate the performance of the FVM framework using a real industrial case study. Section 7 draws the conclusion and discusses future directions.

2 A Brief Review of Existing VM Approaches

As discussed previously, VM is not unique to the semiconductor industry, which essentially serves the same purposes as the soft sensor, a term often used in the process industry. VM or soft sensor makes use of secondary variables that are measured online frequently to predict the product quality variables that are not measured online or measured infrequently. VM can be developed using either model-based approaches or data-driven approaches. For industrial processes, data driven approaches are usually easier to develop and to implement online, therefore they are potentially more attractive. Due to the limited space, only some of the data-driven VM approaches applied to semiconductor manufacturing processes are reviewed in this work.

Among data-drive approaches, the commonly used ones are time series analysis (TSA), Kalman filter (KF), multiple linear regression (MLR), principal component regression (PCR), partial least squares (PLS), and other nonlinear methods such as those based on artificial neural networks (ANNs).

2.1 Time series analysis (TSA)

Because the metrology data are generally sequential in time, autoregressive moving average (ARMA) or autoregressive integrated moving average (ARIMA) models can be identified, *e.g.*, following the procedure proposed by Box and Jenkins (Box et al., 2015). Once the model structure is determined and parameters are estimated using the historical metrology data, the model can be used to predict the future values of the metrology data. Non-seasonal ARMA models are usually denoted by ARMA(p,q) in the following form that combines AR and MA models.

$$y_t - \alpha_1 y_{t-1} - \cdots - \alpha_p y_{t-p} = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} \quad (1)$$

where $y_t, y_{t-1} \cdots y_{t-p}$ are present (at time t) and past metrology data. parameters p and q are non-negative integers, p is the order (number of time lags) of the autoregressive model, and q is the order of the moving-average model. ARMA model assumes that the time series is stationary and it is recommended to difference non-stationary series one or more times to achieve stationary, which results in a more general ARIMA(p, d, q) model where d is the degree/time of differentiation.

2.2 Kalman filter (KF)

Kalman filter was proposed in the early 1960s and has been extensively used for the state estimation of dynamic systems (Haugen, 2010; Kalman, 1960). It has also been formulated for VM (Gill et al., 2010):

$$\mathbf{K} = \mathbf{P}_{old}\mathbf{C}^T(\mathbf{C}\mathbf{P}_{old}\mathbf{C}^T + \mathbf{R})^{-1} \quad (2)$$

$$\mathbf{x}_{new} = \mathbf{x}_{old} + \mathbf{K}(\mathbf{y} - \mathbf{C}\mathbf{x}_{old}) \quad (3)$$

$$\mathbf{P}_{new} = \mathbf{P}_{old} - \mathbf{K}\mathbf{C}\mathbf{P}_{old} \quad (4)$$

$$\mathbf{y}_{est} = \mathbf{C}\mathbf{x}_{new} \quad (5)$$

where $\mathbf{K}$ is the Kalman gain, $\mathbf{P}$ the state error covariance matrix, $\mathbf{R}$ the measurement noise covariance matrix, $\mathbf{x}$ the independent or process variables, $\mathbf{y}$ the dependent or metrology variable.

2.3 Multiple linear regression (MLR)

Multiple linear regression (MLR) aims to model the relationship between multiple explanatory or independent variables from machine data and a response or dependent variable of metrology data by fitting a linear equation to the historical data, which takes the following form:

$$\mathbf{y} = \mathbf{Xb} \tag{6}$$

where $\mathbf{X} \in \mathfrak{R}^{N\times K}$ is the independent variable matrix; $\mathbf{y} \in \mathfrak{R}^{N\times 1}$ is a vector of metrology measurements. The coefficient vector $\mathbf{b}$ is estimated by minimizing the sum of squares of the differences between the actual and modeled metrology measurements, and the obtained model is used to predict metrology measurement when a new set of process variables are available. The potential issue with MLR for VM is that the process variables are quite often (highly) correlated and the collinearities among $\mathbf{x}_i$ can cause severe problems for MLR – the estimated coefficients $\hat{\mathbf{b}}$ can be very unstable, which makes predictions by the regression model unstable or poor.

2.4 Principal component regression (PCR)

Principal component regression (PCR) is an alternative to MLR, which addresses independent variable collinearities. PCR is a regression analysis technique based on principal component analysis (PCA). In PCR, the matrix of raw data $\mathbf{X} \in \mathfrak{R}^{N\times K}$ is decomposed as follows

$$\mathbf{X} = \mathbf{TP}^{\mathrm{T}} + \tilde{\mathbf{X}} \tag{7}$$

where $\mathbf{T} \in \mathfrak{R}^{N\times L}$ and $\mathbf{P} \in \mathfrak{R}^{K\times L}$ are the score and loading matrices, respectively. $\tilde{\mathbf{X}}$ is the residual matrix containing mainly the noise. Then $\mathbf{y}$ is related to $\mathbf{T}$:

$$\mathbf{y} = \mathbf{Tb} \tag{8}$$

which can be solved as

$$\mathbf{b} = (\mathbf{T}^{\mathrm{T}}\mathbf{T})^{-1}\mathbf{T}^{\mathrm{T}}\mathbf{y} \tag{9}$$

In short, instead of regressing the dependent variable (*i.e.*, the metrology measurements) on the explanatory or independent variables (*i.e.*, the process variables) directly as in MLR, the principal components (PCs) or scores of the explanatory variables are used as regressors in PCR. Compared to MLR, PCR has the advantage of addressing the multicollinearity problem. In addition, PCR handles noisy process variables better as usually only a subset of all the PCs are used to build the model. However, the PCs are derived without any reference to the dependent variables. In other words, PC's explain the most variation in $\mathbf{X}$, which may not be (highly) related to the variation in $\mathbf{y}$. This point is clearly demonstrated in the simulated case study in Sec. 5. Due to this reason, the performance of PCR for VM is not guaranteed.

2.5 Partial least squares (PLS)

Partial least squares (PLS) has all the benefits of PCR while also taking the variation of dependent variables into account. Mathematically,

$$\mathbf{X} = \mathbf{TP}^{\mathbf{T}} + \tilde{\mathbf{X}} \tag{10}$$

$$\mathbf{y} = \mathbf{Ub}^{\mathrm{T}} + \tilde{\mathbf{y}} \tag{11}$$

where $\mathbf{U} \in \mathfrak{R}^{N\times \mathrm{L}}$ and the decompositions of $\mathbf{X}$ and $\mathbf{y}$ are made so as to maximize the covariance between $\mathbf{T}$ and $\mathbf{U}$. In other words, PLS models the inner relation that correlates the scores of independent variables with the scores of dependent variables. Therefore, PLS usually has better prediction performance than PCR, which explains why PLS and its variants are the most commonly used VM methods in industrial applications.

2.6 Other methods

Beside the classical VM methods discussed above, driven by the rapid development of machine learning and artificial intelligence in the past a few years, other methods have been proposed. For example, a radial basis function neural network (RBFN) has been proposed as a VM to predict film thickness of a chemical vapor deposition (CVD) process (Hung et al., 2007). Support vector regression (SVR) has been applied for VM as well (Kang et al., 2016). However, these methods have seen few applications because of the complexity involved in implementation and maintenance. In

addition, it has been shown that these kernel-based or NN-based nonlinear methods may not necessarily outperform linear methods in soft sensor. Due to limited space, these methods are not reviewed or compared in this work.

2.7 Recursive or adaptive VM methods

For all the VM methods discussed in the previous subsections, some of them are intrinsically recursive or adaptive methods such as TSA and KF, while the others can be straightforwardly extended to recursive or adaptive variants such as recursive PLS (RPLS). For PCR or PLS based methods, there are various adaptation schemes. In this work, adaptation is achieved by a first-in-first-out (FIFO) window-based approach where in each step the latest sample is included into model training while dropping the oldest sample. This is for the sake of simplicity, not computation efficiency or adaptation performance as neither is the focus of this work.

It is worth noting that due to the high dimensionality of the process variables, in this work TSA only utilizes the metrology data for model building and prediction, while the process data are completely ignored. For KF based VM, to reduce the model size, $\boldsymbol{x}$ are the batch-mean of process variables. Also, because KF is developed for dynamics systems, its good performance relies on continuous updates of the model parameters. Therefore, TSA and KF are included only as recursive VM methods.

2.8 Batch data unfolding

For MLR, PCR and PLS, the traditional batch-wise unfolding is employed to convert 3-D matrix into 2-D matrix of $\mathbf{X}$. In other words, the data matrix $\mathbf{X} \in \mathfrak{R}^{N \times K}$ contains N batches with K variables where $K = V \times M$ with V denoting number of variables being measured, and M denoting number of measurements taken during a batch. For the simulated CMP process, M is the same for all the batches. Therefore, the unfolding process is straightforward. For the industrial plasma etch case study, different batches have different durations and hence different M. In this work, instead of using more complicated dynamic time warping (DTW) or derivative DTW (DDTW) (Zhang et al., 2013), we use simple cut based on the duration of the shortest batch to remove the last few measurements for longer batches. After that, the batches are unfolded into 2-D matrix $\mathbf{X}$.

From the above discussion, we see that all existing VM methods discussed previously make predictions by extracting linear or nonlinear correlations between process variables and metrology measurements. One drawback of utilizing process variables is that some data preprocessing steps are usually required. This is due to the characteristics associated with batch processes, such as unequal batch and/or step length, and unsynchronized or misaligned batch trajectories. These preprocessing steps add complexities to VM implementation and maintenance. In addition, studies have suggested that there could be information loss or distortion caused by data manipulation during preprocessing, which could lead to performance deterioration (He and Wang, 2011). To address this limitation, in the following section, we present the proposed feature-based VM framework, where batch statistics and other features are used as the regressors to predict metrology measurements, which naturally handles unequal batch/step lengths and/or unsynchronized batch/step trajectories. In addition, we show that the feature-based VM framework provides superior prediction performance compared to the traditional VM methods in Sec. 4 and 5 using simulated and industrial case studies.

3 Feature-based Virtual Metrology (FVM)

The feature-based VM (FVM) framework is developed based on statistics pattern analysis (SPA), a process monitoring framework we proposed previously. In SPA, various statistics are used to quantify process characteristics, and these statistics, instead of process variables themselves, are modeled for process monitoring. SPA has been applied for fault detection (He and Wang, 2011; Wang and He, 2010), fault diagnosis (Galiciaa et al., 2012) and virtual metrology (He et al., 2012). In this work, we extend the features to not just statistics, but also other features such as process knowledge based landmark features (Wold et al., 2009); profile-driven features (Rendall et al., 2017); geometry based features (Wang et al., 2015). In the FVM framework, we hypothesize that the batch behavior can be better characterized by the *process features* than by the *process variables*. Therefore, in the FVM framework, process features instead of process variables are used as input variables to build the VM model. Figure 1 provides a schematic diagram of the FVM framework which consists of two steps.

In the first step, various features are extracted from batch trajectories:

$$\mathcal{P}: \mathbf{X} \mapsto \mathbf{F} \tag{12}$$

where $\mathcal{P}$ denotes the operator that maps the process or machine data matrix $\mathbf{X} \in \mathfrak{R}^{N \times K}$ containing N batches with K variables into a feature matrix $\mathbf{F} \in \mathfrak{R}^{N \times S}$ containing N batches with each batch now characterized by S features. The S features can be anything that characterizes the process behavior, such as various statistics that characterize individual variables (such as the mean, variance, autocorrelation), the interactions among different variables (such as the cross-correlations), as well as other features that characterize the process (such as batch and step durations, the time integrals of power input). The features can also be extracted from each step or phase of the batch instead of lumping all steps or phases together.

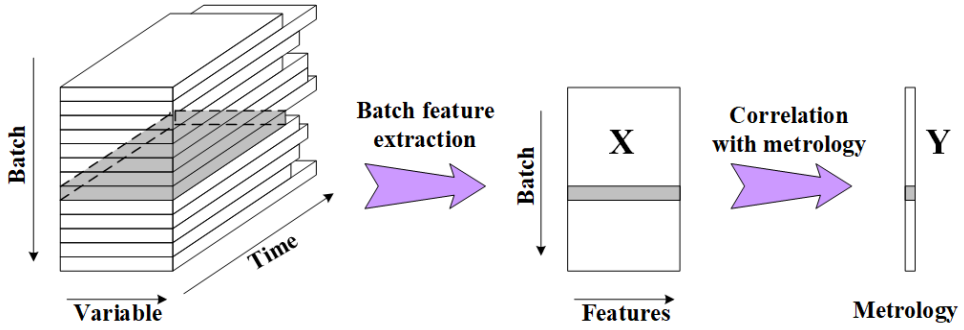


Figure 1. The schematic diagram of the feature-based virtual metrology

In the second step, a regression method, such as PLS used in this work, is utilized to extract the relationships between the features and the metrology measurements.

$$\mathbf{F} = \mathbf{T}\mathbf{P}^{\mathrm{T}} + \tilde{\mathbf{F}} \tag{13}$$

$$\mathbf{y} = \mathbf{U}\mathbf{b}^{\mathrm{T}} + \tilde{\mathbf{y}} \tag{14}$$

where $\mathbf{U} \in \mathfrak{R}^{N \times \mathrm{L}}$ and the decompositions of $\mathbf{F}$ and $\mathbf{y}$ are made so as to maximize the covariance between $\boldsymbol{T}$ and $\boldsymbol{U}$, similar to the regular PLS. As can be seen from Figure 1, unequal batch (or batch step) length and unsynchronized batch (or batch step) trajectories will have no effect on the FVM framework. In other words, the data preprocessing steps that are required by most existing methods, including trajectory alignment/warping and data unfolding, are eliminated by FVM.

[Remarks] The inclusion of features (*i.e.*, what features to be retained in the mapping of Eqn. 12) depends on the process. FVM is a flexible framework and feature inclusion is carried out through cross validation to optimize the VM based on the performance measures to be introduced in the next section. Based on our experiences, the following are some general guidelines that can help with the feature inclusion process: (1) In general, the means and standard deviations of all variables are included due to their general importance to characterize a process; (2) Features such as correlations, auto/cross-correlations are added based on the significance of the correlations and dynamics that exhibit between variables in the process; (3) Higher order statistics (HOS) such as skewness and kurtosis measure the extent of process nonlinearity and process data non-Gaussianity. Their inclusion would enhance VM performance if such characteristics are present in the process. (4) Other non-statistical features, such as process profile, or knowledge, or geometry based features can also be included. One example is given in Sec. where it shows that the more features included, the better performance of the FVM model. It is worth noting that the regression methods such as PCR and PLS can naturally handle collinearity among features. Therefore, feature redundancy is usually not an issue for FVM. Although feature selection is out of the scope of this work, it has been shown that variable or feature selection can sometimes improve the performance of the regression methods. Therefore, any feature selection methods can be used as a preprocessing step for FVM if further performance improvement is desired.

As discussed in (He et al., 2019), the ever-increasing prevalence of big data with 4V challenges, i.e., Volume, Velocity, Variety and Veracity (Zikopoulos et al., 2012), has necessitated the transition from the original variable space monitoring paradigm to the feature space monitoring paradigm. Therefore, we argue that the next generation VM will shift from the original variable space to the feature space as well.

4 Performance measures for comparing different VM methods

In this work, we compare the proposed FVM with the following static VM approaches: MLR, PCR, and PLS. Because FVM utilizes PLS to correlate features with metrology, it can be straightforwardly extended to recursive VM by deploying recursive PLS (RPLS), which is termed recursive FVM or RFVM. We compare RFVM with some existing recursive VM approaches including TSA, KF and RPLS.

The prediction performance of the VM methods are quantified by root-mean-square error (RMSE), the coefficient of determination (R^2) and the mean absolute percentage error (MAPE) s defined below.

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\mathrm{y_i} - \hat{\mathrm{y}}_\mathrm{i})^2} \tag{15}$$

$$\mathrm{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\% \tag{16}$$

where n is the total number of samples, y_i the actual metrology value of the output, and $\hat{y}_i$ the VM predicted value of the output.

$$\mathrm{R}^2 = 1 - \frac{SS_{err}}{SS_{tot}} \tag{17}$$

where $SS_{err} = \sum_{i=1}^{N}(y_i - \hat{y}_i)^2$, $SS_{tot} = \sum_{i=1}^{N}(y_i - \bar{y})^2$, and $\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$.

All methods are optimized using cross-validation by minimizing RMSE whenever applicable.

PLS and RPLS were used as the regression methods for static and recursive FVM methods.

5 Application to a simulated chemical mechanical planarization process

5.1 Chemical mechanical planarization (CMP) simulation

Chemical mechanical planarization (CMP) is a widely used semiconductor manufacturing process to planarize and smooth semiconductor wafers. In CMP, a wafer is held by a rotating wafer carrier and a down force (a.k.a. back-pressure) is applied on the wafer carrier to press the wafer face-down against a rotating polishing pad. Slightly corrosive colloidal slurry containing fine abrasive particles is released onto the pad surface (Baisie et al., 2013). The polishing pad, which is made of porous material that can hold the abrasive particles in the slurry, plays a key role by distributing slurry under the wafer so the chemical and mechanical processes can occur. The material removal occurs as a result of a combination of chemical reaction (between the slurry chemicals and the wafer surface) and the repeated mechanical interaction (between the wafer surface and the polishing pad) under an applied down force (Baisie et al., 2013). Polishing pads can last from about twenty to forty hours and can complete hundreds to even thousands of wafers depending on the particular process (Baisie et al., 2013; Hooper et al., 2002).

In this work, the product characteristics of concern are material removal rate and within-wafer nonuniformity. The material removal rate (MRR) is determined by measuring film thickness before and after polish at each of nine sites on the wafer, then the difference is divided by the polish time. The removal rate is the average of the nine sites on a wafer. The within wafer nonuniformity (WWNU) is computed for each wafer as the standard deviation of the amount removed over the nine sites on the wafer, divided by the average amount removed over the nine sites, times 100 (Baisie et al., 2013; Boning et al., 1996). It is well recognized that MRR and WWNU are difficult to predict and control due to several reasons, including poor understanding of the process, degradation or wear-out of polishing pads, inconsistency of the slurry, variation in pad physical properties, and the lack of in-situ sensors (Boning et al., 1996).

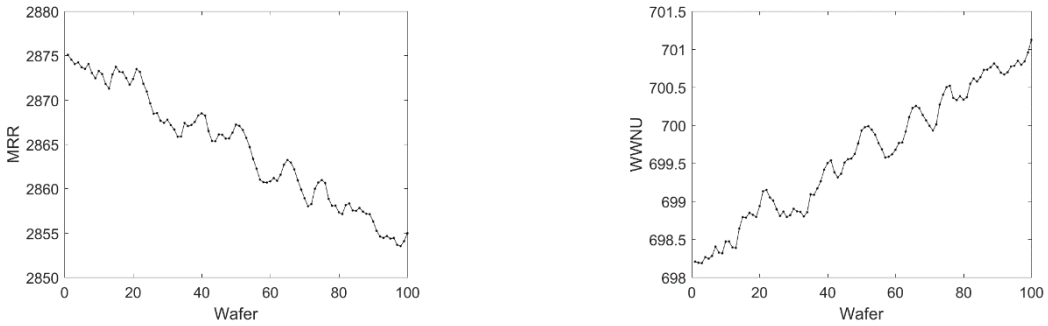
In this work, we adopt an industrial three-input, two-output quadratic CMP process model with linear drift (Khuri, 1996; Moyne et al., 2000) as below.

$$\begin{aligned} y_1 = {} & 2756.5 + 547.6u_1 + 616.3u_2 - 126.7u_3 - 1109.5u_1^2 - 286.1u_2^2 \\ & + 989.1u_3^2 - 52.9u_1u_2 - 156.9u_1u_3 - 550.3u_2u_3 - 10t + \mathcal{E}_{1t} \end{aligned} \tag{18}$$

$$\begin{aligned} y_2 = {} & 746.3 + 62.3u_1 + 128.6u_2 - 152.1u_3 - 289.7u_1^2 - 32.1u_2^2 \\ & + 237.7u_3^2 - 28.9u_1u_2 - 122.1u_1u_3 - 140.6u_2u_3 + 1.5t + \mathcal{E}_{2t} \end{aligned} \tag{19}$$

where the two outputs y_1 and y_2 are MRR and WWNU, respectively. The three inputs u_1, u_2, and u_3 are wafer carrier

down force applied on the wafer, platen speed, and slurry concentration, respectively. u_1, u_2, and u_3 are normalized to the (-1, 1) range. t is time, which is also normalized to (-1, 1) based on the lifetime of the polishing pad, which is set as 100 wafers in this work. $\mathcal{E}_{1t} \sim N(0, 60^2)$ and $\mathcal{E}_{2t} \sim N(0, 30^2)$ are white noises. To illustrate the linear drifts of the CMP process, we perform baseline simulations by fixing all the inputs. Fig. 2 shows that over the life span of a polishing pad, MRR decreases over time while WWNU increases over time (after filtering out the measurement noises), which are consistent with experimental observations (Boning et al., 1996).



(a) (b)

Fig. 2 Baseline simulations with fixed u_1, u_2, and u_3 indicate a decreasing trend in MRR (a), and an increasing trend in WWNU (b), over a polishing pad life span

To test various VM approaches, in this work we simulate open loop runs without process control. To simulate the fluctuations of the inputs u_1, u_2, and u_3, integrated moving average (IMA) models were used. The sampling interval is one second and the processing time for each wafer is 1 minute. *i.e.*, 60 seconds. To mimic production data, it is assumed that only the end of processing value of y_1 and y_2 are available, *i.e.*, one MRR and one WWNU per wafer. The data is generated for 10,000 wafers (*i.e.*, 100 batches with 100 wafers per batch).

5.2 Static VM approach comparison

Data from 25 batches (*i.e.*, 2,500 wafers) are used for building VMs. 25 batches are used for validation and the rest 50 batches are used for testing. For fair comparison, 20 Monte Carlo (MC) runs are carried out to select random batches for training, validation and testing. Since there is no clear correlation between MRR and WWNU, separate MRR and WWNU models based on different approaches are trained, validated and tested. The unfolded original process variables (*i.e.*, $u_1 \sim u_3$ and t) are used as $\boldsymbol{X}$ for MLR, PCR and PLS while MRR or WWNU is the metrology data. For FVM of both MRR and WWNU, the following eight types of features are included: mean (*mn*), standard deviation (*st*), skewness (*sk*), kurtosis (*ku*), auto- and cross-correlations with zero to one lag (*xc*), and time integral (*it*) of $u_1 \sim u_3$, the mean of pair-wise products among $u_1 \sim u_3$ (*mn2*), the wafer index in the batch (*id*). Table 1 compares the average R^2, MAPE and RMSE over 20 MC runs for the two models developed based on different approaches. For PCR, PLS and FVM, the optimal number of PCs used for prediction are also listed in Table 1, which are obtained through validation during each MC run. The optimal number of PCs may vary from run to run due to the change of training, validation and testing samples.

Table 1. Performance comparison of various static VM approaches in predicting MRR and WWNU

Approach	MRR				WWNU			
	# of PC	R^2	MAPE (%)	RMSE	# of PC	R^2	MAPE (%)	RMSE
MLR	-	0.413	5.59	205.9	-	0.087	5.32	52.34
PCR	4-5	0.592	4.58	176.2	3-5	0.462	3.97	42.59
PLS	1-3	0.584	4.64	177.9	1-3	0.453	3.99	42.90
FVM	7	0.980	1.16	38.5	7	0.977	0.98	8.70

As can be seen from Table 1, MLR based VM performs the worst among all approaches. PCR and PLS perform similarly with reasonably high R^2 and MAPE for both MRR and WWNU. RMSE is harder to judge as it is unit or scale dependent. FVM significantly outperforms MLR, PCR and PLS in this case study with ~0.98 R^2 and ~1% MAPE for both MRR and WWNU. These results are visualized in Fig. 3 where the predicted and measured MRR and WWNU

are plotted for MLR, PLS and FVM. Fig. 3 (c) and (f) demonstrates the superior performance of FVM where the predicted MRR and WWNU values agree with the measurements very well.

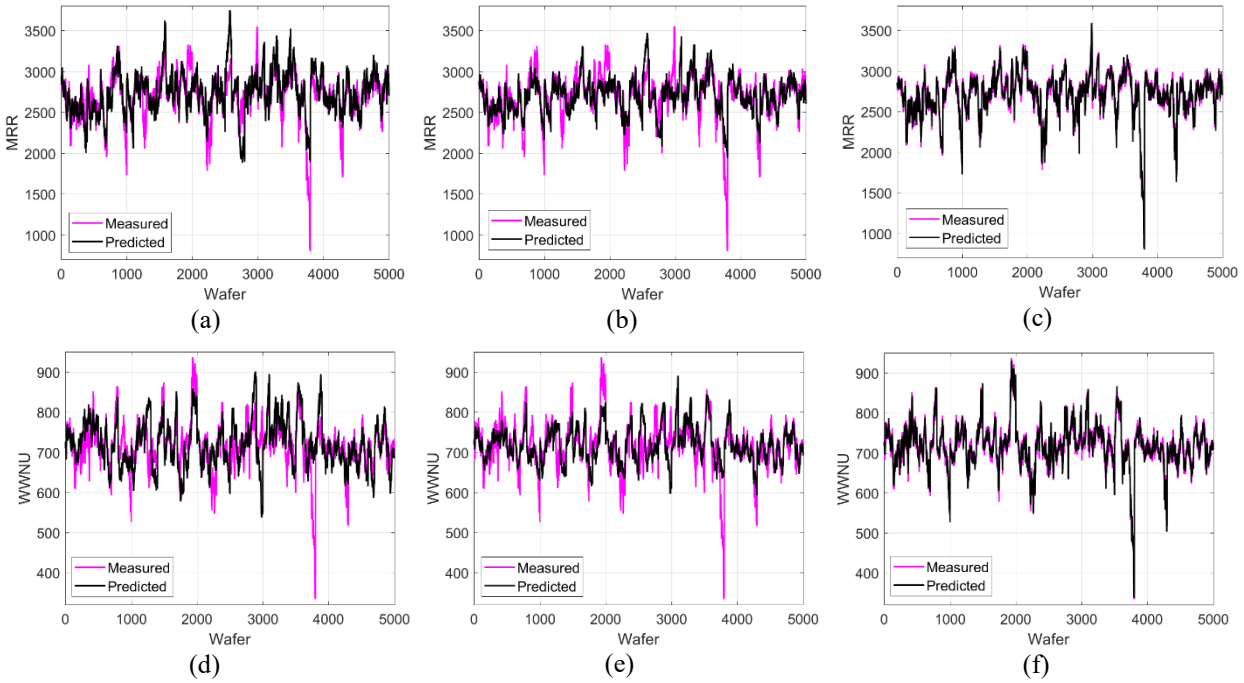


Fig. 3 VM predicted vs. measured MRR (top row) and WWNU (bottom row) based on MLR ((a) and (d)), PLS ((b) and (e)), and FVM ((c) and (f))

To investigate what factors contribute to the superior performance of FVM, the variances of **X** and **y** captured by the first three principal components (PCs) are examined. Since MRR and WWNU models behave similarly as indicated by the consistent trends in Table 1, only MRR models are examined. As shown in Table 2, among the three methods, PCR captures the most variance in ***X*** with 3 PCs, which makes sense as PCs in PCR are determined solely based on **X** without considering **y** (*i.e.*, MRR). On the other hand, although PLS based VM captures slightly less variance in **X**, it captures more variance in MRR, which is consistent with its compromising mechanism that maximizes covariance between **X** and MRR. However, this higher variance of y captured by PLS does not translate into better VM performance in this case study. Compared to PCR and PLS, FVM captures significantly less variance in **X**. Since for FVM **X** consists of features instead of the original variables, the captured variance in **X** cannot be directly compared to those of PCR and PLS. However, the variance captured in MRR can be directly compared and it shows that with 3 PCs, FVM captures 96.0% of total variance in MRR, which is significantly higher than PCR and PLS. This might explain the significantly better performance of FVM in predicting MRR than PCR and PLS, which also means that many included features are probably not relevant to MRR. This suggests that feature selection may further improve the performance of FVM.

Table 2. Variances captured by the first three PCs of different VM approaches (averages over 20 MC runs)

Approach	**Variance captured in X by first 3 PCs (%)**	**Variance captured in y_1(*i.e.*, MRR) by first 3 PCs (%)**
PCR	78.4	58.6
PLS	75.0	67.3
FVM	39.2	96.0

Another way to compare different VM approaches is to check the linearity between PCs and MRR. Here linear regressions are performed to fit MRR to each individual PC, then R^2 of the linear regression and p-value of the F-test on the significance of the coefficient are examined. Generally speaking, R^2 measures how well the model explains the data. In this case, because the models are linear, R^2 quantifies the fraction of the variance in MRR explained by the model. The p-value measures if there is a statistically significant (linear) relationship between MRR and a particular PC. These results are listed in Table 3.

Table 3. R^2 and p-value of linear regression between MRR and individual PC for a particular MC run

Approach	R^2			p-value of F-test		
	PC 1	PC 2	PC 3	PC 1	PC 2	PC 3
PCR	0.058	0.211	0.392	<0.001	<0.001	<0.001
PLS	0.699	0.004	<0.001	<0.001	<0.001	0.707
FVM	0.806	0.126	0.037	<0.001	<0.001	<0.001

Table 3 indicates that for PCR, the PC directions may not be related to the variability in MRR at all. For example, although the first PC captures the most variance in **X**, it only captures 5.8% of the variance in MRR. On the other hand, PC 3 captures the most variance in MRR among the first 3 PCs. Once again, this is attributed to the fact that the PCs are determined solely based on **X**, and their relationship to MRR is established afterwards. For PLS, because of its mechanism of considering the covariance between **X** and MRR, its first PC naturally captures the most variance in MRR and this amount decreases monotonically with PC order. In this case, only the first PC is useful while the other two PCs do not contribute much in capturing variance in MRR. For FVM, since it is PLS applied on features, it follows the decreasing trend of R^2 with PC order. Here it shows that the first PC of the features can capture over 80% of the variance in MRR while the second PC also contribute to 12.6% of the total variance in MRR. Since PCs are orthogonal to each other, these R^2 values add up to the total variances in MRR captured by the first 3 PCs. Table 3 indicates that the features extracted from the original process variables have significantly improved linear relationship with MRR. This is validated by the scatter plots of PC 1 and MRR for different approaches as shown in Fig. 4. Fig. 4 (a) indicates that PC 1 of PCR has the weakest linear relationship with MRR (normalized). PC 1 of PLS has much improved linear relationship with MRR as shown in Fig. 4 (b). However, a clear curvature of the scatter plot indicates the noticeable nonlinearity between PLS PC 1 and MRR. In comparison, PC1 of FVM shows the strongest linear relationship with MRR as shown in Fig. 4 (c).

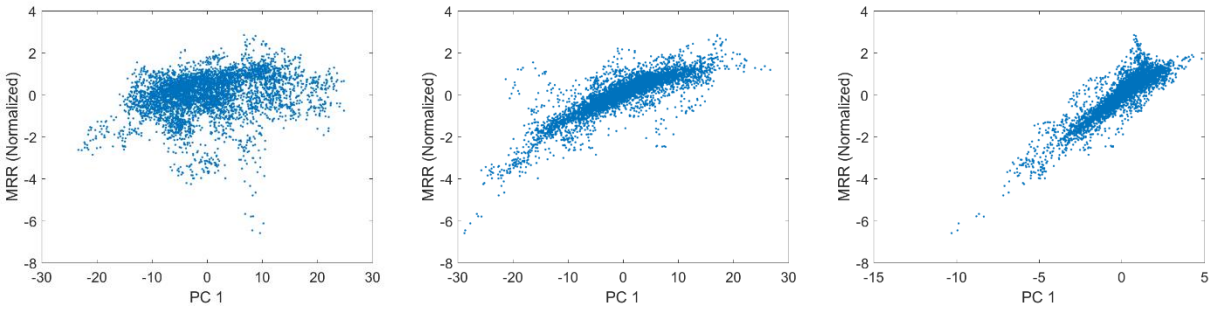


Fig. 4 Scatter plots of the normalized MRR vs. the first PC of PCR (a), PLS (b) and FVM (c).

In summary, despite the clear nonlinearity between **X** and MRR (also WWNU) as indicated by the process models (*i.e.*, Eqns. (18) and (19)) and illustrated in Fig. 4 (b), the features extracted show much improved linearity, which in our view, contributes the most to the much improved performance of FVM compared to other existing approaches.

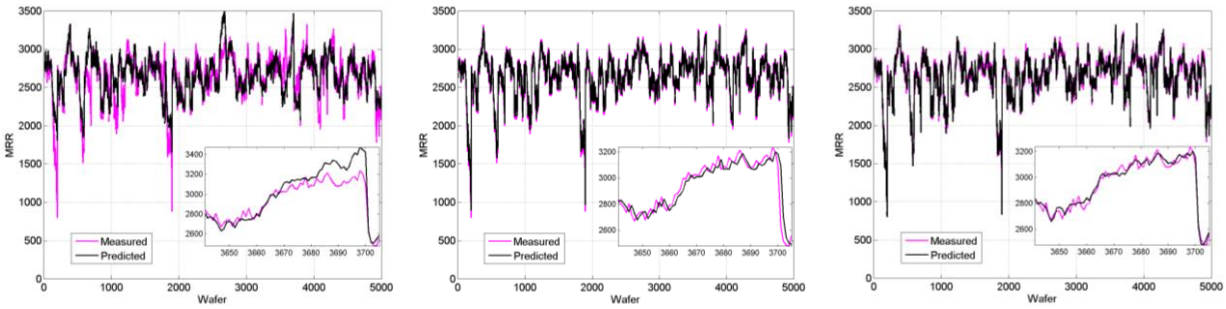
5.3 Recursive VM approach comparison

In this section we compare recursive VM approaches. Because MLR performs the worst in static VM comparison, it is not included in the comparison of recursive methods. In addition, since PCR and PLS perform similarly in the static case, only RPLS is included. As discussed in Sec. 2.7, TSA and KF are recursive in nature, they are included in comparison to recursive FVM (RFVM). The same features used in the static FVM are used for RFVM. For every method, parameter tuning/optimization is done through validation similar to the static case, where the first 25 batches are used for training, the next 25 batches for validation and the remaining 50 batches for testing. One difference is that 20 MC runs are used in the static case to get the average performance of difference static approaches, which is not implemented for the comparison of recursive approaches due to the online nature (*i.e.*, the time sequence must be followed) of recursive approaches.

The comparison results are listed in Table 4 in terms of R^2, MAPE and RMSE for two separate models of MRR and WWNU. As discussed above, because Table 4 is obtained based on a particular composition of training, validation and test samples (*i.e.*, they are divided sequentially in time), the results in Table 4 cannot be directly compared to those in Table 3, which are the average of 20 MC runs with randomly selected training, validation and test samples. Table 4 shows that RPLS and KF perform similarly, which is consistent with our previously established theoretical equivalency between RPLS and KF in state estimation (Wang et al., 2009). TSA performs significantly better than KF and RPLS while RFVM performs the best in both MRR and WWNU predictions. To further investigate the performance metrics in Table 4, we plot the measured vs. predicted MRR for RPLS, TSA and RFVM in Fig. 5. Fig. 5 (b) shows that TSA predicted MRRs follow measurements closely. However, the zoomed-in view of a small segment in the insert of Fig. 5 (b) shows that there is a clear one-step delay in prediction, indicating that the prediction is predominantly determined by the last measurement, which makes sense given the nature of the ARIMA models without input(s). Fig. 5 (a) shows that RPLS does not have such one-step delay in prediction, but the discrepancies between predictions and measurements are significant at places. It is worth noting that the simple FIFO window-based scheme is used to implement RPLS in this work, which means that the latest measurement weighs as much as the oldest measurement in the training data. If a weighting mechanism (*e.g.*, exponentially weighted moving average or EWMA) is employed, we expect much improved performance from RPLS. Similar to RPLS, RFVM does not have one-step delay in prediction since it makes use of the current process variable (*i.e.*, x_t) in the model. In addition, the model predictions agree with the measurements very well. Since RFVM is implemented using RPLS, the performance of RFVM could be further improved if, for example, EWMA is implemented instead of FIFO.

Table 4. Performance comparison of various recursive VM approaches in predicting MRR and WWNU

Approach	MRR				WWNU			
	# of PC	R^2	MAPE (%)	RMSE	# of PC	R^2	MAPE (%)	RMSE
RPLS	3	0.607	5.35	191.0	3	0.409	4.59	46.4
KF	-	0.608	5.25	190.8	-	0.413	4.45	46.2
TSA	-	0.934	2.05	78.2	-	0.923	1.65	16.7
RFVM	11	0.984	1.16	38.5	13	0.979	0.99	8.7



(a) (b) (c)

Fig. 5 Predicted vs. measured MRR based on RPLS (a), TSA (b), and RFVM (c)

6 Application to an Industrial Case Study

In this section, a dataset collected from a plasma etch system at one of Texas Instruments' wafer fabs (Gill et al., 2010) is used to compare the proposed feature-based VM and other VM methods. The dataset contains the recorded values of 18 Optical Emission Spectroscopy (OES) signals collected every 0.1 second for 1121 wafers. The dataset also contains the metrology measurement values of the sheet resistance, which is one of the most important electrical-test parameters used in the semiconductor manufacturing industry to assess the electrical quality of a product. The goal of VM is to predict the end-of-batch sheet resistance using the OES signals.

One OES signal of several wafers is plotted in Fig. 6, which shows the typical characteristics of a semiconductor machine data: unequal batch length or process duration; large variations between wafers and unsynchronized trajectories. To apply traditional VM methods such as PLS on this type of data, several data pre-processing steps have

to be taken, including trajectory alignment or time warping to make trajectories equal length, and trajectory unfolding to flatten the 3-D structure into 2-D matrix. As discussed in Sec. 2.8, for simplicity, we use simple cut based on the duration of the shortest batch to remove the last few measurements for longer batches. After that, the batches are unfolded into 2-D matrix $\boldsymbol{X}$ following the traditional batch-wise unfolding as described in Sec. 2.8.

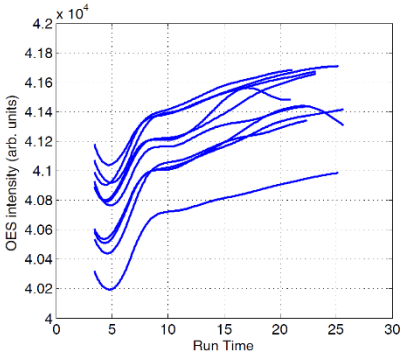


Fig. 6. A sample OES signal of several wafers

6.1 Static VM approach comparison

In this subsection, the static FVM is applied to the dataset discussed previously to predict the sheet resistance using the OES data. The features used in FVM include: time integral of the OES signals (*it*), univariate statistics including mean (*mn*), standard deviation (*st*), skewness (*sk*) and kurtosis (*ku*), as well as the means of pair-wise products (*mn2*) of all 18 variables. The performance is compared with other VM methods. For all VM methods, 70% of the data (784 wafers) are utilized for model building and the rest 30% of the data (337 wafers) are used for testing. Table 1 compares R^2, MAPE and RMSE of FVM to those of MLR, PCR, and PLS. Similar to the simulated case study, MLR performs poorly. This is again due to the high dimensionality of the independent variables after unfolding and the multilinearity among them. Unlike the simulated case study where PCR and PLS perform similarly, in this industrial case study, PLS performs slightly better than PCR. FVM significantly outperforms all other methods in terms of R^2, MAPE and RMSE.

Table 5. Comparison of different static VM methods

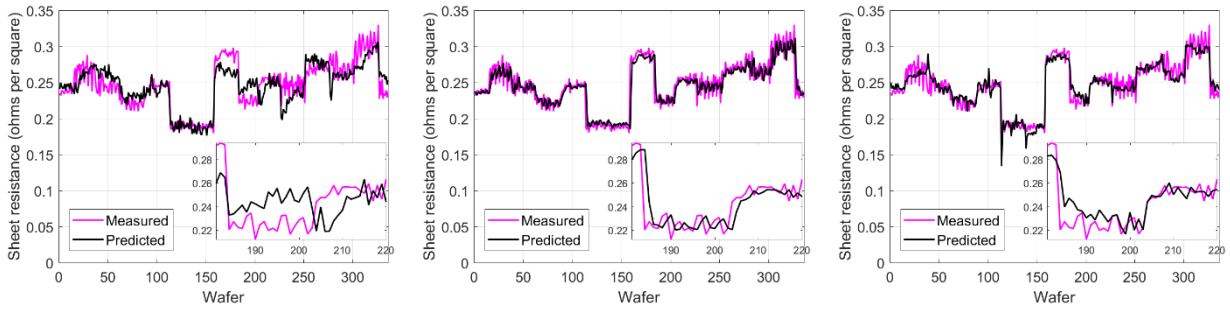
Model	# of PC	R^2	MAPE (%)	RMSE
MLR	-	0.049	10.27	0.0313
PCR	21	0.396	8.55	0.0253
PLS	50	0.437	8.12	0.0245
FVM	18	0.718	5.94	0.0173

6.2 Recursive VM approach comparison

In this subsection, RFVM is applied to the dataset and its performance is compared with those of other recursive VM methods. The initial VM model is built based on the training data of 784 wafers, and is updated when new data becomes available. The same features used in the static FVM are used for RFVM. The comparison results are summarized in Table 6. RPLS and KF perform similarly, which resembles the simulated case study. TSA performs better than RPLS and KF without the use of the inputs (*i.e.*, the OES measurements). Fig. 7 shows the comparison of measured vs. predicted sheet resistances of RPLS, TSA and RFVM. Fig. 7 (b) reveals persistent one-step delay in prediction of TSA. This is again similar to the simulated case study, indicating that TSA prediction is predominantly determined by the last measurement. RPLS and RFVM do not have this phenomenon as shown in Fig. 7 (a) and (c). Both Table 6 and Fig. 7 demonstrate the superior performance of FVM compared to RPLS, KF and TSA.

Table 6. Comparison of different recursive VM methods

Model	# of PC	R^2	MAPE (%)	RMSE
RPLS	15	0.689	5.73	0.0182
KF	-	0.697	5.31	0.0179
TSA	-	0.776	4.20	0.0154
FVM	72	0.855	3.77	0.0124



(a) (b) (c)

Fig. 7 Predicted vs. measured sheet resistance based on RPLS (a), TSA (b), and RFVM (c)

To investigate the effect of feature inclusion on the performance of FVM, we performed RFVM by including different sets of features in RFVM. Table 7 lists the features included, number of principal components determined or optimized through validation, and the resulted performance measures of RFVM. By comparing Tables 7 and 6, it can be seen that by including *mn* alone, RFVM achieves good performance similar to RPLS. By including *mn*, *st* and HOS (*i.e.*, *sk* and *ku*), RFVM outperforms RPLS and KF, which demonstrates the importance of including HOS as features in FVM. By including *mn* and *mn2*, RFVM outperforms all other VM methods listed in Table 6, which indicates that the process nonlinearity is significant. In addition, although the nature of the nonlinearity is unknown due to the complexity of the plasma etch process, the means of pair-wise products (*mn2*) provide a good capture of its nonlinearity. Finally, by using all features, including *mn*, *st*, *sk*, *ku*, *mn2*, as well as *it* (the time integral of the OES signals as a measure of total power input at different frequencies) of all 18 variables, RFVM provides the best performance among all cases listed in Table 7. Table 7 indicates that the more features included, the better performance of RFVM. This is generally true based on our experiences and can be explained by the fact that PLS can naturally handle collinearities among features. In other words, including more features can add process information to the model while feature redundancy poses no issue for FVM. It is worth noting that variable or feature selection can sometimes improve the performance of the regression methods. Therefore, any feature selection methods can be used as a preprocessing step for FVM if further performance improvement is desired. This subject is outside the scope of this work. Further investigation is worth pursuing and feature selection can be integrated as part of the FVM framework.

Table 7. Effect of feature inclusion on the performance of RFVM

Features in RFVM	# of PC's	R^2	MAPE (%)	RMSE
mn	11	0.607	6.57	0.0204
mn*, *st	16	0.627	6.22	0.0199
mn*, *st*, *sk	13	0.685	5.88	0.0183
mn*, *st*, *sk*, *ku	19	0.725	5.45	0.0171
mn*, *st*, *sk*, *ku*, *it	55	0.797	4.65	0.0147
mn*, *mn2	63	0.802	4.55	0.0145
mn*, *st*, *sk*, *ku*, *it*, *mn2	72	0.855	3.77	0.0124

7 Discussions and Conclusions

A feature-based VM (FVM) framework and its recursive/adaptive variant RFVM are proposed in this work to address the challenges presented in semiconductor VM applications, such as unequal batch/step duration and/or unsynchronized trajectories; and large number of variables caused by data unfolding. Because FVM does not require any data preprocessing steps, it is uniquely suited for automatic online applications. The performances of FVM and RFVM are compared with several commonly used VM approaches using a simulated and an industrial case studies.

Among static or off-line VM approaches, both simulated and industrial case studies demonstrate that MLR is not a good VM approach, especially where there are many independent variables (*e.g.*, partly due to batch unfolding) and there exists multilinearity among them. We have also demonstrated that PCR could be problematic for VM as the selected PCs are based on their capabilities in capturing variance among the independent variables, which may not be relevant to the variance of the metrology data. In the simulated case study, the first PC only captures 5.8% of the variance in MRR while PC 3 captures 39.2%, the most among the first 3 PCs. PLS models the inner relation that correlates the scores of independent variables with the scores of dependent variables, which theoretically to enable PLS to have better performance than PCR. Although this point is not shown in the simulated case study, PLS does perform better than PCR in the industrial case study. The proposed FVM approach performs the best in both simulated and industrial cases studies. The analyses reveal that when there exists nonlinearity between independent and dependent variables, the extracted features show much improved linearity, which enables FVM to capture significantly larger amount of variance in the dependent variable(s) with only the first few PCs. This point, in our view contributes the most to the much improved performance of FVM compared to other existing VM approaches.

Among recursive or online VM approaches, KF performs similarly to RPLS. This is expected as the theoretical equivalency between RPLS and KF in state estimation has been established. TSA performs surprisingly well, even without any consideration of any input, in both simulated and industrial case studies. Analyses reveal that this is due to the fact that the metrology data in both cases are highly autoregressive time series and the TSA prediction is predominantly determined by the last measurement. The performance of TSA without input is not guaranteed if metrology time series are not significantly autoregressive. RFVM outperforms all existing recursive VM approaches in both simulated and industrial case studies and its performance can be potentially further improved if a weighting mechanism such as EWMA is implemented instead of FIFO for highly autoregressive metrology measurements such as the ones in this study.

It is worth noting that there are other nonlinear VM approaches as discussed in this work such as kernel-based or ANN-based approaches. Although FVM uses features, they are different from features extracted using kernel-based methods because they have clear physical and/or statistical meanings. Similarly, although ANN-based methods can extract nonlinear relationship between independent and dependent variables, the interpretation of the relationship is challenging. In other words, it is difficult if not impossible to find which variable/feature contribute how much to the output.

There are areas of FVM and RFVM that worth further investigation, such as feature selection and different recursive or adaptive schemes to further improve their performances.

8 Acknowledgement

Financial support from National Science Foundation (NSF-CBET #1805950) is greatly appreciated.

References:

Baisie, E.A., Li, Z., Zhang, X., 2013. Pad conditioning in chemical mechanical polishing: a conditioning density distribution model to predict pad surface shape. Int. J. Manuf. Res. 8, 103–119.

Boning, D.S., Moyne, W.P., Smith, T.H., Moyne, J., Telfeyan, R., Hurwitz, A., Shellman, S., Taylor, J., 1996. Run by run control of chemical-mechanical polishing. IEEE Trans. components, Packag. Manuf. Technol. Part C. Manuf. 8, 103–119. https://doi.org/10.1109/3476.558560

Box, G.E.P., Jenkins, G.M., Reinsel, G.C., Ljung, G.M., 2015. Time Series Analysis: Forecasting & Control, Prentice Hall New Jersey 1994. https://doi.org/10.1016/j.ijforecast.2004.02.001

Galiciaa, H., Heb, Q., Wanga, J., 2012. Statistics Pattern Analysis based fault detection and diagnosis, in: CPC VIII Conference. Paper#55.

Gill, B., Edgar, T.F., Stuber, J., 2010. A novel approach to virtual metrology using Kalman Filtering. Futur. Fab Int. 35, 86–91.

Haugen, F., 2010. Advanced Dynamics and Control. TechTeach.

He, Q.P., Wang, J., 2011. Statistics pattern analysis: A new process monitoring framework and its application to semiconductor batch processes. AIChE J. 57, 107–121. https://doi.org/10.1002/aic.12247

He, Q.P., Wang, J., Gilicia, H.E., Stuber, J.D., Gill, B.S., 2012. Statistics pattern analysis based virtual metrology for plasma etch processes, in: American Control Conference (ACC), 2012. pp. 4897–4902.

He, Q.P., Wang, J., Shah, D., 2019. Feature Space Monitoring for Smart Manufacturing via Statistics Pattern Analysis. Comput. Chem. Eng. 126, 321–331.

Hooper, B.J., Byrne, G., Galligan, S., 2002. Pad conditioning in chemical mechanical polishing. J. Mater. Process. Technol. 123, 107–113. https://doi.org/10.1016/S0924-0136(01)01137-2

Hung, M.-H., Lin, T.-H., Cheng, F.-T., Lin, R.-C., 2007. A Novel Virtual Metrology Scheme for Predicting CVD Thickness in Semiconductor Manufacturing. IEEE/ASME Trans. Mechatronics 12, 308–316. https://doi.org/10.1109/TMECH.2007.897275

Kalman, R.E., 1960. A new approach to linear filtering and prediction problems. J. basic Eng. 82, 35–45.

Kang, P., Kim, D., Cho, S., 2016. Semi-supervised support vector regression based on self-training with label uncertainty: An application to virtual metrology in semiconductor manufacturing. Expert Syst. Appl. 51, 85–106.

Khuri, A.I., 1996. Response surface methods for multiresponse experiments, in: 13th SEMATECH Statistical Methods Symposium.

Lensing, K., Stirton, B., 2006. Integrated metrology and wafer-level contro. Semicond. Int. 29.

Moyne, J., Del Castillo, E., Hurwitz, A.M., 2000. Run-to-run control in semiconductor manufacturing. CRC press.

Rendall, R., Lu, B., Castillo, I., Chin, S.-T., Chiang, L.H., Reis, M.S., 2017. A Unifying and Integrated Framework for Feature Oriented Analysis of Batch Processes. Ind. Eng. Chem. Res. 56, 8590–8605.

Wang, J., He, Q.P., 2010. Multivariate Statistical Process Monitoring Based on Statistics Pattern Analysis. Ind. Eng. Chem. Res. 49, 7858–7869. https://doi.org/10.1021/ie901911p

Wang, J., He, Q.P., Edgar, T.F., 2009. State estimation in high-mix semiconductor manufacturing. J. Process. Control. 19, 443–456.

Wang, R.C., Edgar, T.F., Baldea, M., Nixon, M., Wojsznis, W., Dunia, R., 2015. Process fault detection using time-explicit Kiviat diagrams. AIChE J. 61, 4277–4293.

Wold, S., Kettaneh-Wold, N., MacGregor, J.F., Dunn, K.G., 2009. Batch process modeling and MSPC, in: Brown, S., Tauler, R., Walczak, B. (Eds.), Comprehensive Chemometrics: Chemical and Biochemical Data Analysis. Elsevier, pp. 163–197.

Zhang, Y., Lu, B., Edgar, T.F., 2013. Batch Trajectory Synchronization with Robust Derivative Dynamic Time Warping. Ind. Eng. Chem. Res. 52, 12319–12328. https://doi.org/10.1021/ie303310c

Zikopoulos, P., Parasuraman, K., Deutsch, T., Giles, J., Corrigan, D., others, 2012. Harness the Power of Big Data The IBM Big Data Platform. McGraw Hill Professional.