# Global and Local Virtual Metrology Models for a Plasma Etch Process

Shane A. Lynn, John Ringwood, *Senior Member, IEEE,* and Niall MacGearailt

***Abstract*—Virtual metrology (VM) is the estimation of metrology variables that may be expensive or difficult to measure using readily available process information. This paper investigates the application of global and local VM schemes to a data set recorded from an industrial plasma etch chamber. Windowed VM models are shown to be the most accurate local VM scheme, capable of producing useful estimates of plasma etch rates over multiple chamber maintenance events and many thousands of wafers. Partial least-squares regression, artificial neural networks, and Gaussian process regression are investigated as candidate modeling techniques, with windowed Gaussian process regression models providing the most accurate results for the data set investigated.**

***Index Terms*—Gaussian process regression, local modeling, neural network applications, plasma etch, virtual metrology (VM).**

## I. Introduction

VIRTUAL metrology (VM) is an active area of research in semiconductor manufacture [1]. Traditionally, manufacturing processes such as plasma etch are monitored using statistical process control methodologies with few measurements and large metrology delays. Such monitoring systems can result in wafer scraps due to slow response times and they do not provide the immediate feedback capability required for advanced process control (APC) of processing tools. APC is seen as a core enabling technology required for the continued advancement of the semiconductor industry [2], and fab-wide APC and VM schemes capable of increasing factory throughput, reducing wafer scraps, cutting production costs, and facilitating automated wafer-2-wafer control have been investigated [2]–[4].

Fab-wide APC systems cannot be implemented without the development of accurate VM models for each process in the manufacturing cycle. Plasma etch remains one of the most challenging modeling exercises [5]. While the etch process itself is quite complex, modeling of the process is further complicated by multistep recipes with changing chemistries, chamber conditioning effects, shifts in process characteristics due to preventative maintenance (PM) operations, and limited downstream metrology with which to validate modeling results. A complete review of the semiconductor etch VM literature is found in [6].

VM relies on process data recorded from etch processing tools to generate estimates of process outputs. Chamber-related process data such as temperature, pressure, gas flows, and power are typically collected from etch chambers using in-built sensors. Such data has been used by several authors to create empirical input-output models relating chamber inputs to etch rates, etch bias, and uniformity measures [7]. Additional data can be collected by installing more sensors on the etch chamber. Optical emission spectroscopy is one of the most commonly used noninvasive tools [8] providing information on the chemical species active in plasmas. Plasma impedance monitors (PIMs) analyze the electrical system of the chamber, noninvasively providing information on the current, voltage, and phase of the radio frequency energy applied to the chamber electrodes. PIM signals have been shown to relate to output variables such as etch rate [9] and etch end point [10].

With vast quantities of data available in fabrication plants, the difficulty faced by manufacturers is the effective extraction of useful information from the recorded variables. Variable selection and data reduction techniques are essential to identify key variables and to find useful correlations between recorded variables and process outputs [8], [11].

The treatment of large data sets is a subject requiring consideration by practitioners of VM in semiconductor manufacturing. The two paradigms investigated in this paper are *global models* and *local models*. As defined here, global models use all available training points to learn the behavior of a system. Training is carried out once at initialization, ideally using a training data that covers the full operational range of the system; all further activity is assumed to operate in the same regime. Local models, however, are models that are trained using subsets of the available data. The subsets can be determined from the full data set based on wafer context information, time, or any other criteria. In this manner, local models can provide more accurate estimates than global models over certain operation regimes, while global models may provide more general estimates but across the entire system operating space. Multiple local models are typically required to perform VM over a complete operating space, thus incurring a small complexity overhead over their global

Manuscript received November 18, 2010; revised June 21, 2011; accepted September 10, 2011. Date of publication November 18, 2011; current version February 3, 2012. This work was supported by the Irish Research Council for Science, Engineering, and Technology.

S. A. Lynn and J. Ringwood are with the Department of Electrical Engineering, National University of Ireland, Maynooth, Kildare, Ireland (e-mail: shane.a.lynn@eeng.nuim.ie; john.ringwood@eeng.nuim.ie).

N. MacGearailt is with Intel Ireland, Leixlip, Co. Kildare, Ireland (e-mail: niall.macgearailt@intel.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSM.2011.2176759

counterparts. In this paper, partial least squares (PLS), artificial neural networks (ANNs), and Gaussian process regression (GPR) are compared as candidate VM modeling techniques for global and local modeling of plasma etch rate.

The use of Gaussian processes (GPs) for regression and classification is a relatively new concept. In 1996, Williams and Rasmussen [12] successfully extended the use of GPs to high dimension problems that have been traditionally tackled using other modeling techniques such as neural networks and decision trees [13]. GPR does not impose a specific parametric structure on the underlying function being modeled [14]; rather, the training data are used to discover the model properties in a supervised manner.

GPR has several advantages over other modeling techniques. Using GPR, useful models can be created from training data sets with a relatively small number of training points, and the analyst's prior beliefs about the data can be encapsulated in the choice of a *covariance function*. Because the model form is not specified explicitly, both linear and nonlinear functions can be approximated. Confidence intervals on predictions can be easily evaluated since each prediction is given in the form of a distribution. However, during the training procedure, GP models require the inversion of covariance matrices, the size of which is determined by the number of training data points. For very large data sets, the computational demands of such inversions may become an issue. To the best of the authors' knowledge, this paper reports the first application of GPR to semiconductor etch data, apart from preliminary explorations previously reported in [15].

This paper is organized as follows. Section II describes the modeling techniques used. Because it is rarely seen in the semiconductor literature, particular focus is given to an explanation of GPR. Section III describes the data available for modeling while Sections IV and V discuss global and local modeling results, respectively. Finally, conclusions are given in Section VI.

## II. Modeling Techniques

The estimated etch rate for wafer $k$, $\hat{r}(k)$, is given by

$$\hat{r}(k) = f(u_1(k), u_2(k), \ldots, u_m(k)) \tag{1}$$

where $u_1(k), u_2(k), \ldots, u_m(k)$ are the measurements taken from the chamber sensors during the processing of wafer $k$. Static models are employed because etch rate measurements are not performed on a uniform basis, precluding the use of a time series model. It is assumed that the relationship between the measurements and the plasma etch rate is time-invariant during VM modeling.

### A. PLS Regression

PLS is a statistical technique originally applied to the area of chemometrics for statistical process modeling, and now regularly employed in the area of semiconductor manufacturing. Unlike simpler linear regression techniques, PLS can construct predictive models in the presence of collinear input variables.

PLS is related to another latent variable technique, principal component analysis (PCA). Suppose that we begin with a data matrix $X \in \mathbb{R}^{n\times p}$ made up of $n$ samples of $p$ variables. PCA [16] performs an eigenvalue decomposition of the covariance matrix of the data matrix $X^TX$ to decompose $X$ as the sum of the outer product of the column vectors $\mathbf{t}_i \in \mathbb{R}^{n\times 1}$ and $\mathbf{p}_i \in \mathbb{R}^{p\times 1}$, plus a residual matrix $E$ [16]

$$X = \mathbf{t_1}\mathbf{p_1}^T + \mathbf{t_2}\mathbf{p_2}^T + \cdots + \mathbf{t_l}\mathbf{p_l}^T + E \tag{2}$$

$$= TP^T + E \tag{3}$$

where

$$T = [\mathbf{t}_1\ \mathbf{t}_2 \cdots \mathbf{t}_l],\ P = [\mathbf{p}_1\ \mathbf{p}_2 \cdots \mathbf{p}_l] \tag{4}$$

$l$ is the number of principal components, and $E \in \mathbb{R}^{n\times p}$ is a matrix of residuals. The vectors $\mathbf{t}_i \in \mathbb{R}^{n\times 1}$ are the *scores* or *principal components*, with $T \in \mathbb{R}^{n\times l}$ the principal component matrix, and the vectors $\mathbf{p}_i \in \mathbb{R}^{p\times 1}$ are the *loadings*, where $P \in \mathbb{R}^{p\times l}$ is the loadings matrix. The principal components are arranged in descending order, consistent with the amount of variance explained in the original data set by each one.

In PLS analysis, a similar decomposition to PCA is simultaneously carried out on the output matrix Y such that

$$Y = UQ^T + F \tag{5}$$

where $Y = [\mathbf{y}_1, \mathbf{y}_2 \cdots \mathbf{y}_m]$, $Y \in \mathbb{R}^{n\times m}$, is the output matrix comprising $n$ samples of $m$ output variables, $U \in \mathbb{R}^{n\times h}$ and $Q \in \mathbb{R}^{m\times h}$ are the $Y$-components and $Y$-loadings, respectively, $F \in \mathbb{R}^{n\times m}$ is the $Y$-residual matrix, and finally $h$ is the number of principal components used in the output matrix decomposition. Although PLS is similar to PCA in that components describing the data set are extracted using eigenvalue decompositions, it has the advantage of being a supervised technique that uses information in the output to create a model.

The $X$-components and $Y$-components are chosen so that the relationship between successive pairs of principal components is as strong as possible by manipulating the *inner relation*, $U = TB$, where $B$ is a diagonal matrix of weights optimized to maximize the covariance between the components in $U$ and $T$. An adjusted version of the noniterative partial least squares algorithm, described in [17], can be used to calculate PLS models. Predictions from PLS models are obtained using the multivariate regression formula $\hat{Y} = TBQ^T$ [18].

### B. Artificial Neural Networks (ANNs)

ANNs have been applied extensively to the area of plasma etch for fault detection [19], modeling [20], and control [21], and have been shown to yield superior estimation accuracy over statistical techniques for some data sets [22].

Here, multilayer perceptron (MLP) neural networks are used where neurons are arranged in an input layer, a single hidden layer, and an output layer. The neurons in each layer receive weighted inputs from all neurons in the preceding layer, calculate an output value using tan-sigmoid activation functions and a preset bias value, and pass their outputs to the next layer. This a *feed-forward neural network*. Through experimentation, it is found that no significant improvement in model accuracy is achieved through the use of multiple hidden layers for the etch data set. To avoid limiting the output range, linear output neurons are used.

MLPs are trained by finding the optimal set of network weights and biases that minimizes the sum squared error ($SSE$), defined [23] as

$$SSE = \sum_{i=1}^{n}\sum_{j=1}^{m}(y_{ij} - \hat{y}_{ij})^2 \tag{6}$$

where $n$ is the number of samples in the training set, $\mathbf{y}_{ij}$ is the desired value for output $j$ at sample $i$, and $\hat{\mathbf{y}}_{ij}$ is the estimated value for output $j$ at sample $i$.

Training is carried out by first initializing the network weights randomly and then optimizing the weight values using a training algorithm. Back propagation, for example, gives an effective error measure for each neuron layer and allows a gradient descent optimization routine to adjust the weights and biases to achieve a minimum $SSE$. Higher order optimization techniques, such as the second-order gradient Broyden—Fletcher—Goldfarb—Shannon method [24] or the Levenberg—Marquardt (LM) algorithm [25], are also employed. The models described in this paper were trained using the LM method due to its fast convergence properties [26].

In all of the ANN models created in this paper, ANNs were optimized in a number of ways. First, the number of hidden neurons was varied from 5 to 25 neurons. Second, for each network structure, ten random weight initializations are tested in an attempt to ensure that the optimization routine finds the global minimum error solution. The network structure with the lowest validation error, which is a good measure of generalization capability, across both variations in network topology and training step number, is retained.

### C. GP Regression

A GP can be viewed as a collection of random variables $f(x_i)$ with joint multivariate Gaussian distribution $f(x_1), f(x_2), \cdots, f(x_n) \sim N(0, \Sigma)$, where $\Sigma_{ij}$ gives the value of the covariance between $f(x_i)$ and $f(x_j)$, and is a function of the inputs $x_i$ and $x_j$, $\Sigma_{ij} = k(x_i, x_j)$ [27]. For the purposes of this discussion, a 1-D input–output process is assumed.

Gaussian process models fit naturally into the Bayesian modeling framework where, instead of parameterizing the model function $f(x)$, a Gaussian prior is placed on the range of possible functions that could represent the mapping of inputs $x$ to outputs $y$. The Gaussian prior incorporates the analyst's knowledge about the underlying function in the data, and is specified using the GP covariance function.

The covariance function $k(x_i, x_j)$ can be any function, provided that it generates a positive definite covariance matrix $\Sigma$. A common covariance function is the *squared exponential* (SE) covariance function

$$k(x_i, x_j) = \nu^2 \exp\left(-\frac{(x_i - x_j)^2}{2l^2}\right) \tag{7}$$

where $\nu$ and $l$ are *hyperparameters* that vary the properties of the covariance function to best suit the training data set. The SE covariance function assumes that input points that are numerically close in the input space correspond to outputs that are more correlated in the output space than outputs corresponding to input points which are far apart. Variations in $l$ and $\nu$ control the smoothness of the covariance function. The parameter $\nu$ controls the scale of the variations between points $f(x_i)$ and $f(x_j)$ in the output space, while $l$, known as the length scale, determines the degree of variation in the input dimension. The squared exponential covariance function can be extended to many dimensions by introducing individual length scales for each input dimension.

For example, let the underlying function of the data be $y = f(x) + \epsilon$, where $\epsilon$ is a Gaussian white noise term with variance $\sigma_n^2$ such that $\epsilon \sim N(0, \sigma_n^2)$. A Gaussian process prior is put on the range of possible underlying functions $f(x)$ with covariance function as exemplified in 7 with unknown hyperparameters. Hence

$$y_1, y_2, \ldots, y_n \sim N(0, K) \tag{8}$$

$$K = \Sigma + \sigma_n^2 I \tag{9}$$

where $\sigma_n^2 I$ represents the covariance between outputs due to white noise. $I$ is the $n \times n$ identity matrix.

The aim now is to use the set of training data points $\{x_i, y_i\}_{i=1}^n$ to find the posterior distribution of $y_*$, given input $x_*$, that is $p(y_*|x_*, \mathbf{x}_{tr}, \mathbf{y}_{tr})$, where $\{x_*, y_*\}$ denotes an unseen test data point and $\mathbf{x}_{tr}$ and $\mathbf{y}_{tr}$ denote the complete set of input and output training data. Before the posterior distribution of $y_*$ is found, the unknown hyperparameters of the covariance function 7, $l$, $\nu$, and $\sigma_n^2$, must be optimized. This can be performed via a Monte Carlo method or, more typically, via maximization of the log marginal likelihood

$$\log(p(\mathbf{y}_{tr}|\mathbf{x}_{tr})) = -\frac{1}{2}\mathbf{y}_{tr}^T K^{-1}\mathbf{y}_{tr} - \frac{1}{2}\log(|K|) - \frac{n}{2}\log(2\pi). \tag{10}$$

Equation 10 is made up of a combination of a *data fit* term, $\frac{1}{2}\mathbf{y}_{tr}^T K^{-1}\mathbf{y}_{tr}$, that determines the success of the model in fitting the output training data, along with a *model complexity* term $\frac{1}{2}\log(|K|)$. Maximization of 10 requires the computation of the derivative of $\log(p(\mathbf{y}_{tr}|\mathbf{x}_{tr}))$ with respect to each of the hyperparameters in the covariance function 7. To initialize the gradient descent optimization in the current application, the initial values for the hyperparameters are initialized to the values suggested by Rasmussen [28] and also randomly initialized several times in an attempt to find a global minimum solution for the likelihood function. During optimization for multidimensional covariance functions, dimensions that do not influence the process being modeled are automatically assigned longer length scales than variables of influence. This process is a form of automatic relevance determination.

With the hyperparameters optimized, the GP model is used to predict the distribution of $y_*$ for input $x_*$. The predictive distribution of $y_*$, $p(y_*|x_*, \mathbf{x}_{tr}, \mathbf{y}_{tr})$, can be shown to be Gaussian [29], with mean and variance

$$\begin{aligned} \mu(x_*) &= \mathbf{k}_* K^{-1}\mathbf{y}_{tr} \\ \sigma^2(x_*) &= k_{**} - \mathbf{k}_* K^{-1}\mathbf{k}_*^T + \sigma_n^2 \end{aligned} \tag{11}$$

respectively, where $k_{**} = k(x_*, x_*)$ is the autocovariance of the test input and $\mathbf{k}_* = [k(x_*, x_1), k(x_*, x_2), \cdots k(x_*, x_n)]$ is a vector of covariances between the test and training data points. The vector $\mathbf{k}_* K^{-1}$ can be seen as a vector of weights that form a linear combination of the observed outputs $\mathbf{y}_{tr}$ to form the prediction at $x_*$. The variance on the predicted values, $\sigma^2(x_*)$,

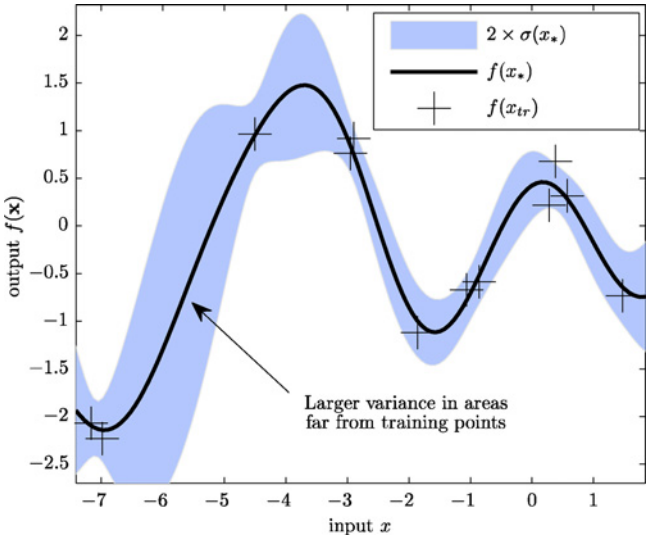


Fig. 1. Example prediction and 95% confidence intervals (2 × standard deviation) for a 1-D GP. The variance on the prediction grows with the distance from observed training points.

is given by the prior variance $k_{**}$, which is a positive term, minus the posterior variance $\mathbf{k}_* K^{-1} \mathbf{k}_*^T$ which is also positive. The posterior variance will be inversely proportional to the distance between the test point and the training points in the input space, as it depends on $\mathbf{k}_*$, resulting in large variances for test points that are far from training points, as shown in Fig. 1.

The covariance function can be chosen to include a number of different components, depending on the prior knowledge of the physical system being modeled, for example, periodic functions, linear components, or rational quadratic functions [27], [29]. The etch rate models in this paper use a covariance function with one linear component, a noise component, and a squared exponential component for each dimension. The squared exponential function is a somewhat intuitive choice for GPR applications since one might expect covariance between training outputs to decrease with distance in the input space. Previous work has shown that the GPR estimation performance for the plasma etch data is relatively insensitive to the covariance function choice [15]. However, the GPR training procedure may not be robust to the inappropriate covariance function choice [30].

## III. Data Set Description

The models examined in this report are constructed and tested on a data set collected from a multistep industrial trench etch process over a period of six months. The data consists of measurements collected from three sources.

1) Etch process (EP) data consist of 131 variables such as temperature, pressure, and gas flow rates for each process step collected directly from the processing tool. These EP data are reduced to a set of 28 variables by discarding variables unrelated to the main etch step and variables with constant values.
2) A PIM records an additional 159 variables for every wafer, comprising 53 harmonics of electrode current, voltage, and phase.

TABLE I
Data Sets Available for Modeling

| | Data Set A | Data Set B |
|---|---|---|
| Number of wafers | 12 133 | 18 513 |
| Etch rate measurements | 529 | 793 |
| PM cycles | 12 | 18 |
| Measurement frequency | 4.4% | 4.3% |
| Inputs available | EP, PIM, XR, EP$^+$ | EP |

3) Etch depth measurements, taken downstream from the etch process, are available for a small number of wafers.

Summary statistics such as mean and standard deviation are derived from the time series traces for each variable, and wafers recorded with erroneous data are detected using a $T^2$ statistic and removed.

Values for plasma reactance (X) and resistance (R) at the 53 harmonic frequencies are calculated from the PIM variables. These reactance and resistance values are henceforth referred to as "XR" data. To investigate whether VM results are improved by combining information from multiple sensor sources, plasma power and impedance values are calculated from the PIM data for each process step and combined with the EP variables for modeling. This set of variables is labeled "EP$^+$" data.

For each virtual metrology scheme, the data points used to build the models form the *training* data and the data points used to check model performance form the *test* data. *Validation* data can be extracted from the training data to enhance the training procedure for some modeling techniques, e.g., early stopping during ANN training and selection of the optimum number of components for PLS models.

The data set is collected from a single etch chamber in the fabrication plant and consists of correctly recorded EP data for 18 513 wafers. After removal of wafers with incorrectly recorded PIM data, only 12 133 wafers remain. For the purposes of this paper, it is useful to form two separate data sets from these data: Data Set A and Data Set B as described in Table I. Due to operational constraints, these are the only data sets available for this VM exercise.

## IV. Global Modeling

As described in Section I, global models use all available training points to learn the behavior of a system. Although data from designed experiments are often used to train global models [3], the high value nature of semiconductor processing means that such experiments can be prohibitively expensive in terms of wafer scrap and tool down-time. In this paper, only past production data are used for model training.

To explore global model performance, 30% of the wafers in Data Set A are put aside as test wafers. Since the input variables originate from measurements with different scales, all input and output variables are normalized to have zero mean and unit variance before modeling. Error metrics are reported on the estimates using the original scale of the variables.

In an online system, test wafer data chronologically follow training wafer data. Hence, chamber drift and PM events can result in models estimating etch rate for operational

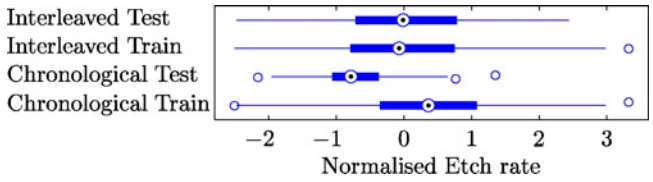


Fig. 2. Etch rate distributions in training and test data sets. The boxes show the 25th to 75th quartile ranges, whiskers extend to 2.7 $\sigma$, and outliers are marked with circles. Note that the chronological test data is restricted to a small range of etch rate values.

TABLE II
GLOBAL MODELING RESULTS FOR CHRONOLOGICAL DATA SET

| Data Source | PLS | | ANN | | GPR | |
|---|---|---|---|---|---|---|
| | MAPE | R2 | MAPE | R2 | MAPE | R2 |
| EP data | 1.23 | 0.25 | 1.63 | 0.00 | 1.29 | 0.28 |
| **EP-SS** | **1.17** | **0.28** | **1.37** | **0.11** | **1.19** | **0.33** |
| PIM | 1.21 | 0.22 | 2.07 | 0.15 | 1.21 | 0.24 |
| PIM-PCA | 1.19 | 0.25 | 1.91 | 0.21 | 1.25 | 0.24 |
| PIM-SS | 1.35 | 0.14 | 1.58 | 0.13 | 1.50 | 0.14 |
| X and R | 1.24 | 0.18 | 1.41 | 0.05 | 1.28 | 0.17 |
| X and R-PCA | 1.38 | 0.13 | 1.59 | 0.03 | 1.34 | 0.15 |
| X and R-SS | 1.29 | 0.16 | 1.45 | 0.03 | 1.44 | 0.16 |
| EP$^+$ | 1.22 | 0.25 | 1.65 | 0.21 | 1.34 | 0.25 |

spaces not represented in the training data, with unpredictable results. To investigate whether model accuracy is improved when information from the same operational region as that of the test wafers is included in the model training data sets, an *interleaved* data set is used, where the training and test wafers are interleaved throughout the complete data set. The interleaved scheme gives some measure of the merit of a more comprehensive data logging/metrology philosophy.

EP, PIM, XR, and EP$^+$ data are investigated as candidate input variable combinations to the global models. Stepwise selection (SS) [31] and PCA [16] are investigated as variable selection and data reduction techniques for the input variables. The global modeling results for both chronologically ordered and interleaved data sets are provided in Tables II and III, respectively. Results are compared using the mean absolute percentage error (MAPE) on test data

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|\hat{y}_i - y_i|}{|y_i|} \times 100 \qquad (12)$$

where $y$ is the real etch rate, $\hat{y}$ is the predicted etch rate, and $n$ is the number of samples. The coefficient of determination, $R^2$, representing the square of the correlation between $\mathbf{y}$ and $\hat{\mathbf{y}}$, is also given for the etch rate estimates. Because the VM models can follow low frequency fluctuations in etch rate, but fail at accurately estimating smaller high frequency fluctuations, the larger range of etch rate values contained in the interleaved test data (see Fig. 2) increases the reported $R^2$ values in Table III. As a result, unlike MAPE values, $R^2$ values are not directly comparable between Tables II and III.

Tables II and III do not suggest a particular selection of inputs that demonstrates significantly more accurate performance than other selections across all data sets. Models using EP data with SS of input variables produce consistently good results for all of the modeling techniques when the data

TABLE III
GLOBAL MODELING RESULTS FOR INTERLEAVED DATA SET

| Data Source | PLS | | ANN | | GPR | |
|---|---|---|---|---|---|---|
| | MAPE | R2 | MAPE | R2 | MAPE | R2 |
| EP data | 1.25 | 0.66 | 1.26 | 0.64 | 1.16 | 0.68 |
| EP-SS | 1.19 | 0.68 | 1.41 | 0.37 | 1.16 | 0.68 |
| PIM | 1.16 | 0.70 | 1.35 | 0.63 | 1.11 | 0.72 |
| PIM-PCA | 1.16 | 0.70 | 1.27 | 0.63 | 1.11 | 0.73 |
| **PIM-SS** | **1.13** | **0.71** | **1.21** | **0.68** | **1.10** | **0.72** |
| X and R | 1.19 | 0.69 | 1.28 | 0.65 | 1.14 | 0.71 |
| X and R-PCA | 1.18 | 0.68 | 1.35 | 0.62 | 1.18 | 0.69 |
| X and R-SS | 1.20 | 0.68 | 1.24 | 0.68 | 1.21 | 0.68 |
| EP$^+$ | 1.28 | 0.64 | 1.21 | 0.65 | 1.17 | 0.68 |

is in chronological order but for interleaved data sets, PIM inputs produce superior performance. ANN models yield the worst model performance for all input combinations, with PLS models performing best for chronological data, and GPR models performing best for interleaved data. GPR modeling is expected to be advantageous during modeling of the interleaved data, since estimation is mainly an exercise in interpolation rather than extrapolation. For chronologically ordered data, the test data points can arise from very different operating regions of the input data space, requiring extrapolation of the training data information. GPR models tend to sit down gracefully when offered an extrapolation task (a positive feature, one might argue), whereas linear techniques, such as PLS, will always give a best linear guess to an extrapolation problem.

It is important to have a measure of the degree of confidence in the VM estimates, in addition to the estimates themselves [32]. GPR models naturally permit confidence intervals to be established. As per 12, if a test point is distant from the training data points, the output estimate variance is large. Assuming that previously unseen tool shifts and drifts are reflected in the VM input variables, when they occur, high variance values can be used to alert practitioners of unreliable estimates. Fig. 3 shows 95% confidence intervals for a set of etch rate estimates using GPR models.

The addition of PIM sensor data to the etch models does not yield a substantial increase in the accuracy of the global models, making it difficult to justify the additional sensor cost for this data set.

On average, models built using the interleaved data sets result in better MAPE values than the models based on the chronologically organized data because the training data set contains information from the same operational space as the test data. However, such a situation is typically not realizable in a production environment due to constraints on the frequency of metrology. Hence, to minimize extrapolation across a PM event while not requiring unobtainable data from different operational spaces for model training, a number of local modeling methods are considered for etch-rate estimation.

## V. LOCAL MODELING

### A. Regional PM Cycle Models

The first of the local modeling schemes, *regional PM cycle modeling*, is a division of data such that wafers are partitioned into separate bins depending on their position within each PM

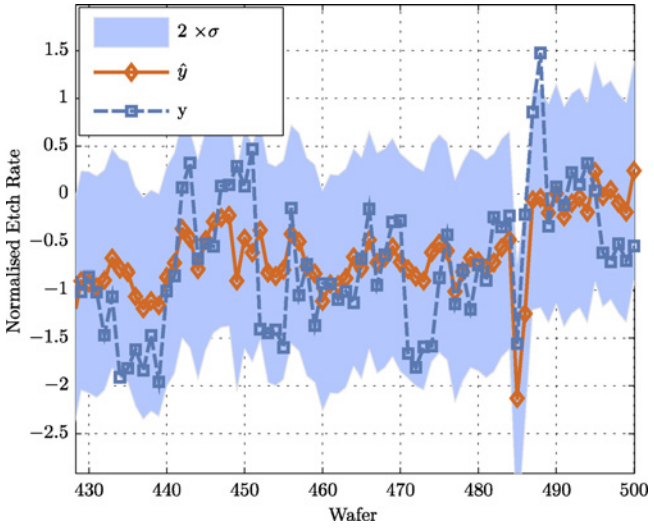


Fig. 3. Test wafer etch rate estimates, with 95% confidence intervals, from global GPR model based on stepwise selected EP data for chronologically ordered data. The high variance of the measured etch rate arises from unmodeled process variation and this variance is not reflected in the etch rate estimates because it is not captured by the EP variables. The etch rate variance remains constant in this figure because the data for the wafers shown are taken from the same operational space as those used to train the GPR model.

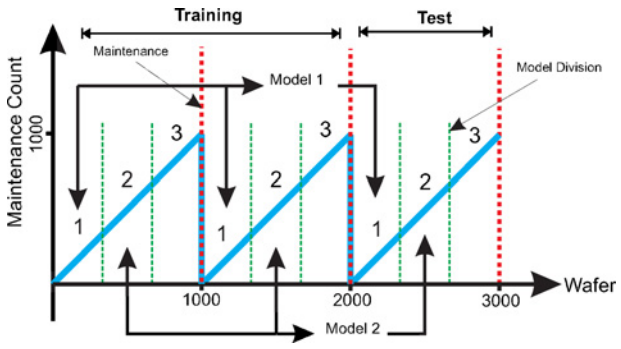


Fig. 4. Regional PM modeling scheme.

cycle. As depicted in Fig. 4, a different VM model is then constructed for each region of the PM cycle, and the models are switched for each unseen wafer depending on its PM cycle position.

The purpose of the regional PM cycle modeling scheme is to investigate whether similarities exist between the plasma etch data at different stages in PM cycles. It is conjectured that the beginning, middle, and end sections of individual maintenance cycles may be more similar to the corresponding sections in *other* PM cycles than to the other sections of the same cycle. Supporting this hypothesis are the facts that several of the measurements recorded from the etch chamber exhibit repeatable patterns over the course of each PM cycle, and it is known that chambers undergo a conditioning process as wafers are processed throughout each PM cycle.

The performance of the regional PM cycle models is tested using Data Set A to allow performance comparisons between models built using different input combinations. The test data set (30% of data points) occurs chronologically later than the training (50%) and validation (20%) data.

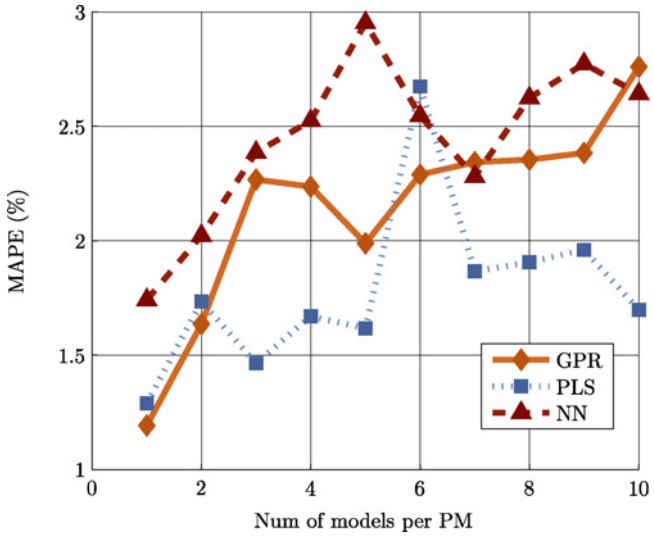


Fig. 5. MAPE for the regional PM cycle modeling scheme using EP data with varying numbers of regional models per PM cycle.

The number of models per PM cycle is varied from one to ten in order to determine an appropriate level of granularity and to highlight the effect of the PM cycle partitioning technique on estimation accuracy. Fig. 5 shows the MAPE performance for PLS, ANN, and GPR-based models. The first point for each model type is equivalent to the global modeling scheme seen in Section IV as it uses one model to cover all PM stages.

Fig. 5 illustrates that regional PM cycle models do not increase the estimation accuracy of the virtual metrology models using EP data as input variables. Rather, accuracy worsens with increasing numbers of models per PM cycle. Similar results are found for different input variable selections [33], [34]. This degradation in performance is attributed to a lack of exploitable commonality between similar sections of different PM cycles in the etch rate data set. Furthermore, there is a reduction in the number of training points available for each model as the number of models per PM cycle increases.

### *B. PM Cycle Clustering*

This section investigates whether similarities between different PM cycles can be found and harnessed to increase etch rate estimation accuracy. Analysis of Data Set A, where both EP and PIM data are available, reveals four distinct clusters of self-similar data points. The clusters are visible in Fig. 6 by performing a PCA on the EP and XR data separately and plotting the first two principal components of the XR data against the first principal component of the EP data.

Each cluster contains a number of different PM cycles with similar EP and XR data. The existence of these clusters in the data set suggests that the etch process moves between a finite number of operating points over the course of the complete data set. Similar modal behavior is seen in a stack etch process by He *et al* [35], and Zeng and Spanos [36] also reported on clustered behavior in an etch process where the clusters were associated with different etch chambers. In our work, changes in cluster, indicating changes of operating space, are

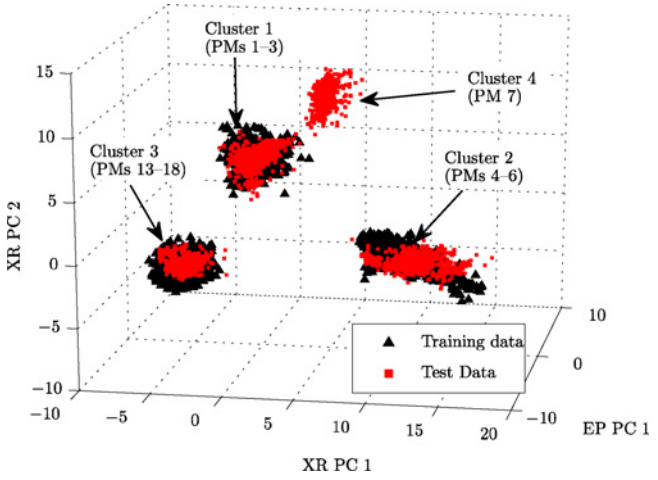


Fig. 6. First principal component of EP data plotted against the first two PCs of XR data. Data from one PM cycle from each cluster is chosen as test data.

brought about by PM events on the *same* chamber. An analysis of variance of the etch rates in each cluster rejects the null hypothesis that the mean etch rates in each cluster are equal at the $\alpha = 0.01$ significance level.

Specialized models for each cluster are tested to examine whether they can estimate etch rate more accurately than global models trained using information from all clusters. To test the performance of such cluster-based modeling, one PM cycle of wafers is extracted from each cluster to be used as unseen test data during model tests as shown in Fig. 6. To compare the cluster performance to a global-type scenario, one model is trained using the training data from all clusters, and the same unseen test data is used to measure its performance. Because Cluster 4 comprises a single PM cycle, no separate training and test data exists for Cluster 4 model creation. The data from Cluster 4 provide an opportunity to explore VM strategies during modes of chamber operation not previously captured by other cluster models.

PLS, ANN, and GPR models are examined as candidate modeling techniques for cluster models. EP, PIM, XR and EP$^+$ data are investigated as input variable options. Stepwise selection is applied to the input variables before GPR and ANN modeling to reduce the number of input variables and improve performance. Because PLS first projects the incoming data onto its latent variable space as described in Section II-A, it is capable of modeling the cluster data sets that have more input variables than training samples. Results are presented in Table IV.

In tests completed for Cluster 4, for which there is only one PM cycle, no cluster model is capable of accurately estimating etch rate; the global models yield the best estimates. Hence, when the etch tool operates in a previously unseen operational space, measurements of etch depth can be taken with greater frequency than before to allow new cluster model identification and to ensure the process is operating within specifications.

By way of an example, Fig. 7 shows the estimates from the cluster and global GPR models on the test data points. Improvements in accuracy can be seen for Cluster 2 in Fig. 7 for the cluster models. Cluster models are useful only in the cluster over which they are trained, while the global model can provide meaningful estimates over the full test data set.

TABLE IV
GLOBAL AND CLUSTER MODEL FOR ALL DATA TYPES

| Data | Model | Global Model | | Cluster Model | |
|---|---|---|---|---|---|
| | | MAPE | R2 | MAPE | R2 |
| EP | PLS | 1.79 | 0.53 | 1.60 | 0.52 |
| | ANN | 2.08 | 0.48 | 1.74 | 0.42 |
| | GPR | 1.77 | 0.52 | 1.62 | 0.52 |
| PIM | PLS | 1.57 | 0.50 | 1.59 | 0.49 |
| | ANN | 1.62 | 0.50 | 1.78 | 0.41 |
| | GPR | 1.63 | 0.47 | 1.77 | 0.44 |
| XR | PLS | 1.65 | 0.52 | 1.59 | 0.49 |
| | ANN | 1.76 | 0.42 | 1.96 | 0.32 |
| | GPR | 1.69 | 0.46 | 1.58 | 0.52 |
| EP+ | PLS | 1.97 | 0.51 | 1.71 | 0.51 |
| | ANN | 1.65 | 0.49 | 1.86 | 0.45 |
| | GPR | 1.77 | 0.52 | 1.70 | 0.50 |

The global model results differ from Tables II and III because the global model here is trained using data from every cluster in the data set and then tested using the same data as the cluster models.

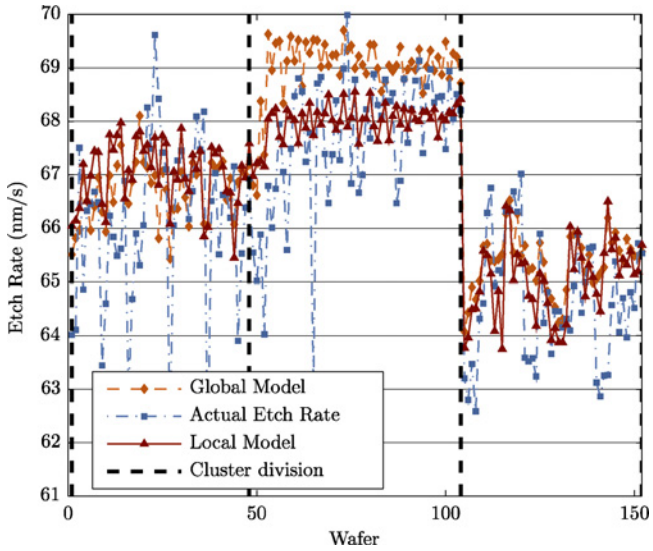


Fig. 7. Example etch rate estimates from global and clustered models using EP data as input.

In Table IV, cluster models yield better estimation accuracy for half of the model types and input selections, with superior global models being those trained using PIM data, and those using neural networks. The best *overall* MAPE is reported for the global PLS model using PIM data. Due to the lack of consistent results and the considerable extra complexity of cluster model implementation, we cannot definitively conclude that cluster modeling is superior to global modeling for this data set. A larger set of historical data is required to fully assess the capability of the clustering technique, but the potential benefits are demonstrated here. In the case of more data being available, new clusters are expected to be generated, or the wafer data is expected to return to an existing cluster, allowing model reuse.

### C. Windowed Models

Time-windowed models can be used to maintain model accuracy in time-varying systems. For example, Qin [37]

TABLE V
COMPARISON OF MODEL TRAINING TIMES (IN S)

| Window Length | PLS | ANN | GPR |
|---|---|---|---|
| 70 | 0.03 | 1.90 | 26.27 |
| 150 | 0.08 | 4.24 | 121.85 |
| 250 | 0.13 | 6.97 | 320.90 |

applied a moving window PLS scheme to a catalytic reformer. Khan *et al.* [3] described a virtual metrology and run-to-run control strategy that uses a continually updating windowed PLS model in their simulations of a semiconductor manufacturing fabrication environment.

PLS, ANN, and GPR-based models are compared as candidate modeling techniques for windowed modeling of the plasma etch data. Data Set B is used for windowed model analysis since the data set is almost fully contiguous, with only one substantial gap where wafer records are not available. The data set is kept in a chronological order throughout the experiments, providing a realistic representation of data produced by an etch tool during processing. The use of Data Set B restricts the analysis to using EP data only during modeling (see Table I). The models are applied to the data set using window lengths between 30 and 300 samples. The window length describes the number of past wafers used to train VM models that are used to estimate the etch rates of wafers subsequently processed. When a new etch rate measurement becomes available, the window is advanced to include the new measurement and a new model is created. Global models using similar numbers of wafers fail to produce accurate etch rate estimates over the complete data set because they fail to maintain their validity in new operating spaces [38].

Training times for PLS models are faster than those of ANN and GPR models because the latter two techniques require the use of optimization techniques and multiple initializations during training. Table V shows model training times (in seconds) for each technique, using a computer with a 2.6 GHz dual-core processor and 2 GB RAM. Although a second-order optimization technique is used during ANN model training, first-order gradient descent is used to optimize the GPR hyperparameters. Hence, although GPR training arguably involves a more complex optimization task than ANN training, the GPR training times may be improved through the use of more complex optimization techniques. However, the estimation times for all model types are typically less than 1 s. The etch processing time for each wafer is approximately 5 min, and there is a metrology delay of several hours for etch rate measurements. Hence, according to Table V, which indicates that models can be completely retrained in the order of seconds, any of the three modeling techniques investigated are suitable for real-time implementation of a window-based VM system.

To increase windowed modeling accuracy, the most recently measured value of etch rate is included as an input variable to the models. PLS model accuracy is enhanced via a maintenance-dependent sample weighting scheme as described in [38]. Stepwise selection of input variables is performed on each window before modeling for both ANN and GPR models.

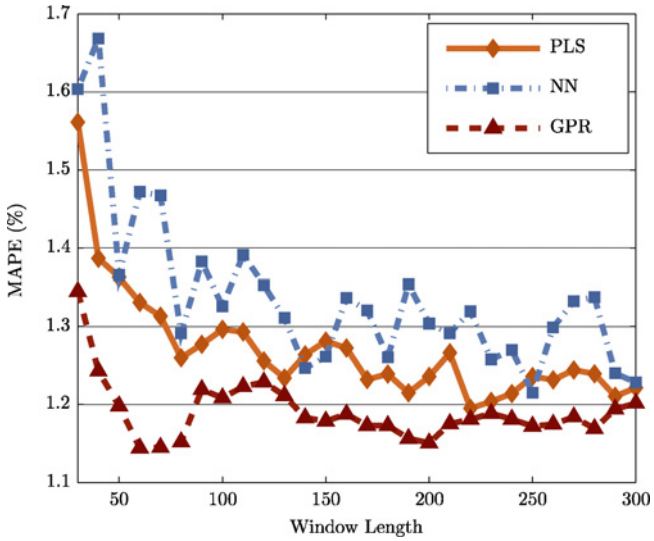


Fig. 8. Windowed model MAPE performance for varying window length.

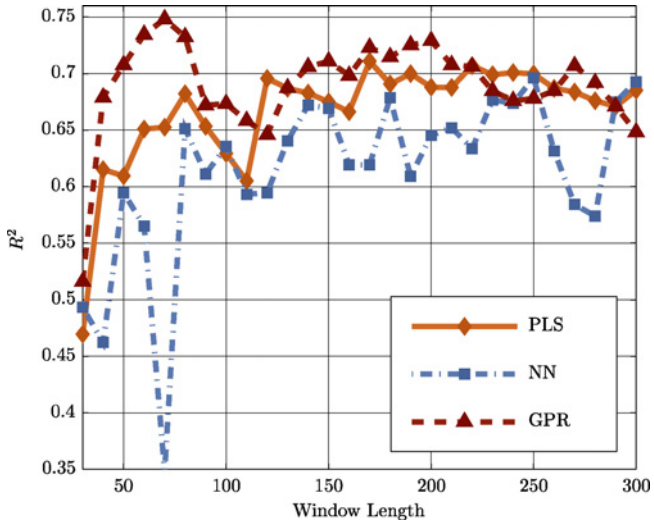


Fig. 9. Windowed model $R^2$ values for varying window length.

The error performance for each modeling technique is shown in Figs. 8 and 9 for the range of window lengths investigated.

The $R^2$ and MAPE values for the windowed models are significantly better than the global models values. Considering that a much smaller amount of training data is used during windowed model training, and that the models can perform accurately over numerous PM events in the test set, windowed models are preferable. The GPR-based windowed models outperform windowed models based on PLS and ANNs, especially for smaller window sizes. For small window sizes, ANN models perform poorly due to a lack of training data. Increasing the window size improves ANN model performance, but the GPR models still follow the etch rate more successfully.

The best results for the windowed GPR models are recorded for a window length of 70 wafers. This model estimated the actual etch rate for the 493 unseen test wafers (that span over 11 000 processed but unmeasured wafers) with a MAPE of 1.15% and $R^2$ of 0.75. The etch rate estimates, confidence

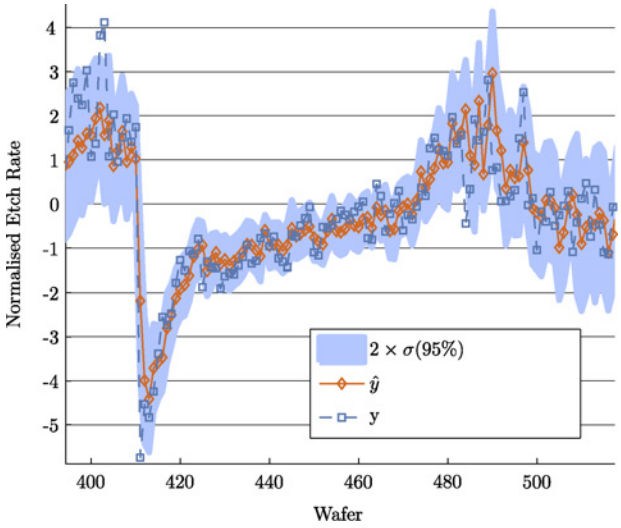


Fig. 10. Best etch rate estimates using windowed GPR model with window length 70.

TABLE VI
COMPARISON OF MAPE ACHIEVED BY ALL MODELING TECHNIQUES AND APPROACHES FOR EP DATA

| | PLS | ANN | GPR | |
|---|---|---|---|---|
| Global modeling | Chronological | 1.23 | 1.63 | 1.29 |
| | Interleaved | 1.25 | 1.26 | 1.16 |
| Local modeling | Regional | 1.43 | 2.01 | 1.57 |
| | Clustering | 1.60 | 1.74 | 1.62 |
| | Windowed | **1.2** | **1.23** | **1.14** |

limits, and actual etch rates for a section of the test data set are shown in Fig. 10.

While the windowed GPR model follows the overall trend of the etch rate variations, it struggles to accurately model high frequency fluctuations in the data. These high frequency fluctuations do not appear to be reflected in the recorded input variables, and may arise from other processes in the manufacturing line, or unmeasured disturbances in incoming material. However, the 95% confidence limits produced using the GPR models, which vary over the data set, encapsulate a large enough range to allow for the vast majority of these variations as shown in Fig. 10.

## VI. CONCLUSION

The results in this paper reflect the reality of using production data, with a limited number of measured wafers, to develop VM models. Particular difficulties attach to the utilization of VM models across PM boundaries, and a variety of local modeling approaches are explored to minimize this problem. However, disaggregation of production data for local model development results in small local data sets and this creates problems for a number of modeling paradigms, particularly ANNs. In contrast, GPR models work well with small data sets, and produce an accompanying variance value for each etch rate estimate. Table VI compares all of the modeling techniques and data disaggregation approaches explored in this paper for the EP data set.

Concerning the performance of various local modeling approaches, the division of PM cycles into separate sections for modeling is not beneficial, while the clustering of PM cycles with similar characteristics can improve marginally on the accuracy of models with global scope for some input variable selections. The use of a wafer window scheme (with GPR modeling) produces the best estimation accuracy of etch rate on the data set investigated.

## ACKNOWLEDGMENT

The authors would like to thank Intel Ireland, Leixlip, Co. Kildare, Ireland, for their help with this research.

## REFERENCES

[1] P. Chen, S. Wu, J. Lin, F. Ko, H. Lo, J. Wang, C. Yu, and M. Liang, "Virtual metrology: A solution for wafer to wafer advanced process control," in *Proc. IEEE Int. Symp. Semicond. Manuf.*, Sep. 2005, pp. 155–157.

[2] A. Khan, J. Moyne, and D. Tilbury, "An approach for factory-wide control utilizing virtual metrology," *IEEE Trans. Semicond. Manuf.*, vol. 20, no. 4, pp. 364–375, Nov. 2007.

[3] A. A. Khan, J. Moyne, and D. Tilbury, "Virtual metrology and feedback control for semiconductor manufacturing processes using recursive partial least squares," *J. Process Control*, vol. 18, no. 10, pp. 961–974, 2008.

[4] A. Ferreira, A. Roussy, and L. Conde, "Virtual metrology models for predicting physical measurement in semiconductor manufacturing," in *Proc. IEEE/SEMI Adv. Semicond. Manuf. Conf.*, May 2009, pp. 149–154.

[5] Y. Yang, M. Wang, and M. J. Kushner, "Progress, opportunities and challenges in modeling of plasma etching," in *Proc. Int. Interconnect Technol. Conf.*, Jun. 2008, pp. 90–92.

[6] J. V. Ringwood, S. Lynn, G. Bacelli, B. Ma, E. Ragnoli, and S. McLoone, "Estimation and control in semiconductor etch: Practice and possibilities," *IEEE Trans. Semicond. Manuf.*, vol. 23, no. 1, pp. 87–98, Feb. 2010.

[7] B. Kim and K. Kim, "Prediction of profile surface roughness in CHF3/CF4 plasma using neural network," *Appl. Surf. Sci.*, vol. 222, nos. 1–4, pp. 17–22, Jan. 2004.

[8] E. Ragnoli, S. McLoone, S. Lynn, J. Ringwood, and N. Macgearailt, "Identifying key process characteristics and predicting etch rate from high-dimension datasets," in *Proc. IEEE/SEMI Adv. Semicond. Manuf. Conf.*, May 2009, pp. 106–111.

[9] M. N. A. Dewan, P. J. McNally, T. Perova, and P. A. F. Herbert, "Use of plasma impedance monitoring for determination of SF6 reactive ion etching end point of the SiO2/Si system," *Microelectron. Eng.*, vol. 65, nos. 1–2, pp. 25–46, Jan. 2003.

[10] M. Kanoh, M. Yamage, and H. Takada, "End-point detection of reactive ion etching by plasma impedance monitoring," *Japan. J. Appl. Phys.*, vol. 40, no. 3A, pp. 1457–1462, Mar. 2001.

[11] D. Zeng, Y. Tan, and C. J. Spanos, "Dimensionality reduction methods in virtual metrology," *Proc. SPIE*, vol. 6922, no. 1, p. 692238, Feb. 2008.

[12] C. K. I. Williams and C. E. Rasmussen, *Gaussian Processes for Regression*. Cambridge, MA: MIT Press, 1996, ch. 8, pp. 514–520.

[13] C. K. I. Williams, "Prediction with Gaussian processes: From linear regression to linear prediction and beyond," Neural Comput. Res. Group, Aston Univ., Birmingham, U.K., Tech. Rep. NCRG/97/012, Oct. 1997.

[14] M. Ebden. (2008, Aug.). *Gaussian Processes for Regression: A Quick Introduction* [Online]. Available: http://www.robots.ox.ac.uk/mebden/reports/GPtutorial.pdf

[15] S. Lynn, J. Ringwood, and N. MacGearailt, "Gaussian process regression for virtual metrology of plasma etch," in *Proc. IET Irish Signals Syst. Conf.*, vol. 2010, no. CP566. 2010, pp. 42–47.

[16] J. E. Jackson, *A User's Guide to Principal Components*. New York: Wiley-Interscience, 1991.

[17] P. Geladi and B. R. Kowalski, "Partial least-squares regression: A tutorial," *Anal. Chim. Acta*, vol. 185, no. 1, pp. 1–17, 1986.

[18] H. Abdi. (2003). *Partial Least Squares (PLS) Regression*. Thousand Oaks, CA: Sage [Online]. Available: http://www.utdallas.edu/herve/Abdi-PLS-pretty.pdf
[19] Y.-J. Chang, "Fault detection for plasma etching processes using RBF neural networks," in *Proc. Int. Symp. Neural Netw.*, 2005, pp. 538–543.
[20] B. Kim, W. Choi, and H. Kim, "Using neural networks with a linear output neuron to model plasma etch processes," in *Proc. IEEE Int. Symp. Ind. Electron.*, vol. 1. Jun. 2001, pp. 441–445.
[21] D. Stokes and G. May, "Real-time control of reactive ion etching using neural networks," *IEEE Trans. Semicond. Manuf.*, vol. 13, no. 4, pp. 469–480, Nov. 2000.
[22] C. Himmel and G. May, "Advantages of plasma etch modeling using neural networks over statistical techniques," *IEEE Trans. Semicond. Manuf.*, vol. 6, no. 2, pp. 103–111, May 1993.
[23] B. Kim and S. Kim, "Partial diagnostic data to plasma etch modeling using neural network," *Microelectron. Eng.*, vol. 75, no. 4, pp. 397–404, Jul. 2004.
[24] R. Battiti and F. Masulli, "BFGS optimization for faster and automated supervised learning," in *Proc. Int. Neural Netw. Conf.*, vol. 2. 1990, pp. 757–760.
[25] D. Marquardt, "An algorithm for least squares estimation of non-linear parameters," *SIAM J. Appl. Math.*, vol. 11, no. 2, pp. 431–441, Jun. 1963.
[26] M. Hagan and M. Menhaj, "Training feedforward networks with the Marquardt algorithm," *IEEE Trans. Neural Netw.*, vol. 5, no. 6, pp. 989–993, Nov. 1994.
[27] J. Kocijan. (2008). "Gaussian process models for systems identification," in *Proc. 9th Int. Ph.D. Workshop Syst. Control Young Gener. Viewpoint*, pp. 8–15 [Online]. Available: http://dsc.ijs.si/jus.kocijan/GPdyn/
[28] C. E. Rasmussen. (1996). "Evaluation of Guassian processes and other methods for non-linear regression," Ph.D. dissertation, Dept. Comput. Sci., Univ. Toronto, Toronto, ON, Canada [Online]. Available: http://www.gaussianprocess.org/
[29] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press, 2006.
[30] P. Sollich, "Gaussian process regression with mismatched models," in *Proc. Neural Inform. Process. Syst.*, 2001, pp. 519–526.
[31] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to Linear Regression Analysis*. New York: Wiley, 2001.
[32] F.-T. Cheng, Y.-T. Chen, Y.-C. Su, and D.-L. Zeng, "Evaluating reliance level of a virtual metrology system," *IEEE Trans. Semicond. Manuf.*, vol. 21, no. 1, pp. 92–103, Feb. 2008.
[33] S. Lynn, J. Ringwood, E. Ragnoli, S. McLoone, and N. MacGearailt, "Virtual metrology for plasma etch using tool variables," in *Proc. IEEE/SEMI Adv. Semicond. Manuf. Conf.*, May 2009, pp. 143–148.
[34] S. Lynn, "Local modeling of a plasma etch data set," Dept. Electr. Eng., Natl. Univ. Ireland, Maynooth, Kildare, Ireland, Tech. Rep. EE/JVR/1/2010, Feb. 2010.
[35] Q. He and J. Wang, "Fault detection using the k-nearest neighbor rule for semiconductor manufacturing processes," *IEEE Trans. Semicond. Manuf.*, vol. 20, no. 4, pp. 345–354, Nov. 2007.
[36] D. Zeng and C. J. Spanos, "Virtual metrology modeling for plasma etch operations," *IEEE Trans. Semicond. Manuf.*, vol. 22, no. 4, pp. 419–431, Nov. 2009.
[37] S. J. Qin, "Recursive PLS algorithms for adaptive data modeling," *Comput. Chem. Eng.*, vol. 22, nos. 4–5, pp. 503–514, 1998.
[38] S. Lynn, J. V. Ringwood, and N. MacGearailt, "Weighted windowed PLS models for virtual metrology of an industrial plasma etch process," in *Proc. IEEE Int. Conf. Ind. Technol.*, Mar. 2010, pp. 271–276.

**Shane A. Lynn** received the B.Eng. degree in computer engineering and the Ph.D. degree in engineering from the National University of Ireland, Maynooth, Kildare, Ireland, in 2006 and 2011, respectively. His Ph.D. research focused on the virtual metrology and real-time control of a plasma etch process used in semiconductor manufacturing.

He has completed an internship with Ad Astra Rocket Company, Guanacaste, Costa Rica. He has worked with Intel Ireland, Leixlip, Co. Kildare, Ireland, on the development of virtual metrology and data analysis systems. He is currently with the Department of Electrical Engineering, National University of Ireland, Maynooth. His current research interests include empirical modeling of complex processes, advanced process control applications, and data analysis and data reduction of large data sets.

Dr. Lynn has received the Endeavour Award from the Australian Government to complete research on autonomous vehicles with the University of Technology, Sydney, Australia in 2012.

**John Ringwood** (M'87–SM'97) received the Diploma degree in electrical engineering from the Dublin Institute of Technology, Dublin, Ireland, and the Ph.D. degree in control systems from Strathclyde University, Glasgow, Scotland, in 1981 and 1985, respectively.

He is currently a Professor of electronic engineering with the Department of Electrical Engineering, National University of Ireland (NUI), Maynooth, Kildare, Ireland. He was the Founding Head of the Electronic Engineering Department, NUI, from 2000 to 2005, and also served as the Dean of the Faculty of Engineering. His current research interests include semiconductor manufacturing, ocean energy, and biomedical engineering.

Dr. Ringwood is a Chartered Engineer and a Fellow of Engineers Ireland.

**Niall Macgearailt** received the Bachelors degree in mechanical engineering from University College Dublin, Dublin, Ireland, in 1991. He is currently pursuing the Ph.D. degree with Dublin City University, Dublin.

He joined Intel Ireland, Leixlip, Co. Kildare, Ireland, in 2002 as a Process Engineer and worked on developing fault detection systems using advanced sensors before moving to a research role in 2005. His research with Intel focuses on the areas of metrology reduction and equipment performance improvement. He is also responsible for establishing and leading collaboration programs between Intel and universities to address key technology problems. Before working with Intel, he was with Lam Research, Fremont, CA, for 8 years, where he worked as a Research and Development Engineer developing next-generation plasma chambers and processes. He spent a number of years at various wafer fabrication facilities in San Jose, CA.