Intelligent Quality Controllers for On-Line Parameter Design

Ratna Babu Chinnam, Jie Ding, and Gary S. May, *Senior Member, IEEE*

***Abstract*—Parameter design methods, in general, do not take into account the common occurrence that some of the uncontrollable factors are observable for products and processes, during operation and production, respectively. This paper introduces a methodology that facilitates on-line parameter design for products and processes utilizing the extra information available about observable uncontrollable factors. Implementation of the proposed methodology leads to a quality controller that operates in two distinct modes: identification mode and on-line parameter design mode. Identification mode involves establishing a model that relates quality response characteristics with significant controllable and uncontrollable variables. On-line parameter design mode involves optimization of the controllable variables with respect to desired levels of output quality parameters, with consideration to levels of the observable uncontrollable variables. A plasma etching semiconductor manufacturing process is used as a testbed for the proposed intelligent quality controllers. Results reveal that the proposed quality controllers can be used for on-line parameter design of manufacturing processes. Results also reveal that significant improvements in quality (measured in terms of average deviation of process outputs from target) over off-line parameter design approaches are to be expected in production processes with some level of control on uncontrollable variables. Even in the absence of any control on uncontrollable variables, the proposed controllers always perform better than traditional off-line robust parameter design techniques; however, the improvements may not be significant.**

***Index Terms*—Iterative inversion, neural networks, parameter design, quality.**

I. Introduction

CURRENT approaches to product/process design, improvement, and optimization have borrowed considerably from the principles of Taguchi (for an account of his techniques, see [1]–[3]. Taguchi and others working on these issues highlight an approach that puts emphasis on product/process variability, in contrast to traditional approaches that focused primarily on product/process location. They address the notion that products and processes lack quality because of performance inconsistency, often produced by factors that are uncontrollable in the design of the product or process (i.e., environmental factors, or factors that are a function of usage by the customers). Consequently, in recent years, attention has been placed on the choice of a product/process design that is said to be resistant (robust) to these environmental or noise variables. For a panel discussion on the topic, see Nair [4].

In general, parameter design methods do not take into account the common occurrence that some of the uncontrollable variables are observable during production [5] and part usage. This extra information regarding the levels of uncontrollable factors enhances our choice of values for the controllable factors and, in some cases, determines the viability of the production process and or the product. Given the rapid decline in instrumentation costs over the last decade, the development of methods that utilize this additional information will facilitate optimal utilization of the capability of manufacturing processes.

Pledger [5] described an approach that explicitly introduces uncontrollable factors into a designed experiment. The method involves splitting uncontrollable factors into two sets, observable and unobservable. In the first set there may be factors like temperature and humidity, while in the second there may be factors like chemical purity and material homogeneity that may be unmeasurable due to time, physical, and economic constraints. The aim is to find a relationship between the controllable factors and the observable uncontrollable factors while simultaneously minimizing the variance of the response and keeping the mean response on target. Given the levels of the observable uncontrollable variables, appropriate values for the controllable factors are generated on-line that meet the stated objectives. Pledger [5] derived a closed-form expression, using Lagrangian minimization, that facilitates minimization of product or process variance while keeping the mean on target, when the model that relates the quality response variable to the controllable and uncontrollable variables is linear in parameters and involves no higher order terms. However, as pointed out by Pledger [5], if the model involves quadratic terms or other higher order interactions, there can be no closed-form solution.

In this paper, we develop some general ideas that facilitate on-line parameter design. The specific objective is to not impose any constraint on the nature of the relationship between the different controllable and uncontrollable variables and the quality response characteristics, and allow multiple quality response characteristics. In particular, we recommend feed-forward neural networks (FFNs) for modeling the quality response characteristics. The reason for making this recommendation involves its universal approximation characteristics. It has been proved that FFNs can approximate any continuous function (R^N, R^M) over a compact subset of R^N to arbitrary precision [6]. Previous research has also shown that neural networks offer

Manuscript received March 9, 1999; revised May 10, 2000.

R. B. Chinnam is with the Industrial and Manufacturing Engineering Department, Wayne State University, Detroit, MI 48202 USA (e-mail: Chinnam@mie.eng.wayne.edu).

J. Ding is with Evolve, San Francisco, CA 94111 USA.

G. S. May is with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: gary.may@ece.gatech.edu).

Publisher Item Identifier S 0894-6507(00)09501-4.

0894–6507/00$10.00 © 2000 IEEE

advantages in both accuracy and robustness over statistical methods for modeling processes (for example, Nadi *et al.* [7], Himmel and May [8], and Kim and May [9]).

Besides proposing nonparametric neural network models for "modeling" quality response characteristics of manufacturing processes, we recommend a gradient descent search technique for "optimizing" the levels of the controllable variables on-line. In particular, this paper considers a neural network iterative inversion scheme for optimization of controllable variables. However, one can use other nonlinear optimization methods (in particular, stochastic search methods such as genetic algorithms). The overall framework that facilitates these two on-line tasks, i.e., modeling and optimization, constitutes a quality controller. The paper focuses on development of quality controllers for manufacturing processes whose quality response characteristics are static and time-invariant. Future research can concentrate on extending the proposed controllers to deal with dynamic and time-variant systems.

This paper is organized as follows. Section II provides a brief overview of feed-forward neural networks utilized in this paper for process modeling and optimization. Section III describes an approach to design intelligent quality controllers and discusses the relevant issues. Section IV presents some results from the application of the proposed methods to a plasma etching semiconductor manufacturing process. Section V provides a summary and gives directions for future work.

II. AN OVERVIEW OF FEED-FORWARD NEURAL NETWORKS

In general, FFNs are composed of many nonlinear computational elements, called nodes, operating in parallel, and arranged in patterns reminiscent of biological neural nets. These processing elements are connected by weight values, responsible for modifying signals propagating along connections and used for the training process. The number of nodes plus the connectivity define the topology of the network and range from totally connected to a topology where each node is just connected to its neighbors. The following sections briefly discuss the characteristics of a class of feed-forward neural networks.

A. Multilayer Perceptron Networks

A typical multilayer perceptron network (MLP) neural network with an input layer, an output layer, and two hidden layers is shown in Fig. 1 (referred to as a three-layer network; normally, input layer is not counted). For convenience, the same network is denoted in block diagram form as shown in Fig. 2 with three weight matrices $\mathbf{W}^{(1)}$, $\mathbf{W}^{(2)}$, and $\mathbf{W}^{(3)}$ and a diagonal nonlinear operator Γ with identical nodal function γ following each of the weight matrices. The most popular nonlinear nodal function for MLP networks is the sigmoid [*unipolar* $\rightarrow \gamma(x) = 1/(1+e^{-x})$, where $0 \leq \gamma(x) \leq 1$ for $-\infty < x < \infty$ and *bipolar* $\rightarrow \gamma(x) = (1-e^{-x})/(1+e^{-x})$, where $-1 \leq \gamma(x) \leq 1$ for $-\infty < x < \infty$]. Each layer of the network can then be represented by the operator

$$\mathbf{N}_l[\mathbf{x}] = \mathbf{\Gamma}[\mathbf{W}^{(l)}\mathbf{x}] \tag{1}$$

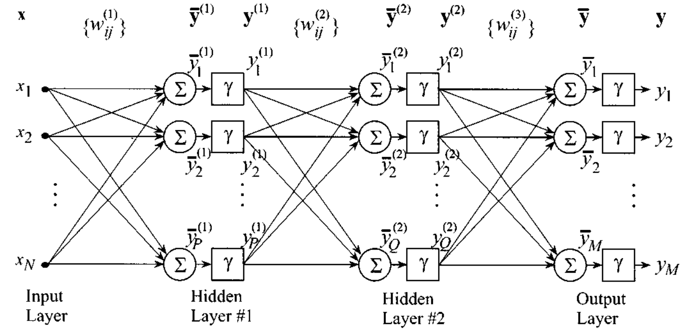


Fig. 1. A three-layer neural network.

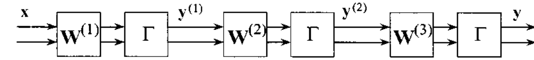


Fig. 2. A block diagram representation of a three-layer network.

and the input–output mapping of the MLP network can be represented by

$$\mathbf{y} = \mathbf{N}[\mathbf{x}] = \mathbf{\Gamma}\left[\mathbf{W}^{(3)}\mathbf{\Gamma}\left[\mathbf{W}^{(2)}\mathbf{\Gamma}\left[\mathbf{W}^{(1)}\mathbf{x}\right]\right]\right] = \mathbf{N}_3\mathbf{N}_2\mathbf{N}_1[\mathbf{x}]. \tag{2}$$

The weights of the network $\mathbf{W}^{(1)}$, $\mathbf{W}^{(2)}$, and $\mathbf{W}^{(3)}$ are adjusted (as described in Section II-B) to minimize a suitable function of error e between the predicted output $\mathbf{y}$ of the network and a desired output $\mathbf{y}_d$ (error-correction learning), resulting in a mapping function $\mathbf{N}[\mathbf{x}]$.

It has been shown in Hornik *et al.* [6], using the Stone–Weierstrass theorem, that even an MLP network with just one hidden layer and arbitrarily large number of nodes can approximate any continuous function $f \in C(R^N, R^M)$ over a compact subset of R^N to arbitrary precision (universal approximation). This provides the motivation to use MLP networks in modeling/identification of any manufacturing process' response characteristics.

B. Training MLP Networks Using Back-Propagation Algorithm

If MLP networks are used to solve the identification problems treated in this paper, the objective is to determine an adaptive algorithm or rule that adjusts the weights of the network based on a given set of input–output pairs. An error-correction learning algorithm will be discussed here, and readers can see Haykin [10] for information regarding other training algorithms. If the weights of the networks are considered as elements of a parameter vector θ, the error-correction learning process involves the determination of the vector θ^*, which optimizes a performance function J based on the output error. In error-correction learning, the gradient of the performance function with respect to θ is computed and adjusted along the negative gradient as follows:

$$\theta(s_{\text{le}}+1) = \theta(s_{\text{le}}) - \eta\frac{\partial J(s_{\text{le}})}{\partial\theta(s_{\text{le}})} \tag{3}$$

where η is a positive constant that determines the rate of learning (step size) and s_{le} denotes the network "learning" (or training) iteration step. If $\mathbf{y}_d = (y_{d1}, \ldots, y_{dM})^T$ is the desired output vector, the output error of a given input pattern $\mathbf{x}$ is defined as

$\mathbf{e} = \mathbf{y} - \mathbf{y}_d$. Typically, the performance function J is defined as a function of the mean-square error or mean absolute error.

In the literature, a well-known method for determining this gradient in (3) for MLP networks is the back-propagation method. The analytical method of deriving the gradient is well known in the literature and will not be repeated here. It can be shown that the back-propagation method leads to the following gradients for any MLP network with L layers

$$\frac{\partial J(s_{\text{le}})}{\partial w_{ij}^{(l)}(s_{\text{le}})} = -\delta_i^{(l)}(s_{\text{le}}) y_j^{(l-1)}(s_{\text{le}}) \tag{4}$$

$$\delta_i^{(L)}(s_{\text{le}}) = e_i \gamma_i' \left(\overline{y}_i^{(L)}(s_{\text{le}})\right) \quad \text{for neuron } i \text{ in output layer } L \tag{4a}$$

$$\delta_i^{(l)}(s_{\text{le}}) = \gamma_i' \left(\overline{y}_i^{(l)}(s_{\text{le}})\right) \sum_k \delta_k^{(l+1)}(s_{\text{le}}) w_{\text{ki}}^{(l+1)}(s_{\text{le}}) \quad \text{for neuron } i \text{ in hidden layer } l. \tag{4b}$$

Here, $\delta_i^{(l)}(s_{\text{le}})$ denotes the local gradient defined for neuron i in layer l, and the use of prime in $\gamma_i'(\overline{y}_i^{(l)}(s_{\text{le}}))$ signifies differentiation with respect to the argument. One starts with local gradient calculations for the outmost layer and proceeds backward until one reaches the first hidden layer (hence the name back-propagation). For more information on MLP networks, see Haykin [10].

C. Iterative Inversion of Neural Networks

In error back-propagation training of neural networks, the output error is "propagated backward" through the network. Linden and Kindermann [11] have shown that the same mechanism of weight learning can be used to iteratively invert a neural network model. This approach is used in this paper for on-line parameter design and hence the discussion. In this approach, errors in the network output are ascribed to errors in the network input signal, rather than to errors in the weights. Thus, iterative inversion of neural networks proceeds by a gradient descent search of the network input space, while error back-propagation training proceeds through a search in the synaptic weight space.

Through iterative inversion of the network, one can generate the input vector $\mathbf{x}$ that gives an output as close as possible to the desired output $\mathbf{y}_d$. The iterative gradient descent algorithm can be applied to obtain the desired input vector as follows:

$$\mathbf{x}(s_{ii}+1) = \mathbf{x}(s_{ii}) - \eta \cdot \frac{\partial J(s_{ii})}{\partial \mathbf{x}(s_{ii})} \tag{5}$$

where η is a positive constant that determines the rate of iterative inversion and s_{ii} refers to the "iterative inversion" iteration step. For further information, see Linden and Kindermann [11].

III. Design of Quality Controllers for On-Line Parameter Design

The proposed framework for performing on-line parameter design is illustrated in Fig. 3. In contrast to the classical control theory approaches, this structure includes two distinct control loops. The process control loop "maintains" the controllable variables at the optimal levels and will involve schemes such as feedback control, feed-forward control, and adaptive control. It is the quality controller in the quality control loop that "determines" these optimal levels, i.e., performs parameter design. The quality controller includes both a model of the product/process-quality response characteristics and an optimization routine to find the optimal levels of the controllable variables. Please note that there can be a distinct difference between product quality and process quality. For example, in metal cutting processes, process quality might be represented by surface geometry factors (such as surface roughness, finish, and waviness) and surface integrity factors (such as the nature of the heat affected zone and its influence on chemical/mechanical properties). However, in the metal cutting industry, product quality is predominantly judged in terms of geometry (which normally involves comparing the produced unit with the design specifications and the related tolerances). The methods discussed in this paper can be applied to address both process-quality issues and product-quality issues, as long as the stated assumptions are met. However, as mentioned earlier, the primary focus of this paper is to facilitate parameter design by accounting for changes in uncontrollable variables, and hence, one might say that the methods are currently more valuable for addressing process-quality issues.

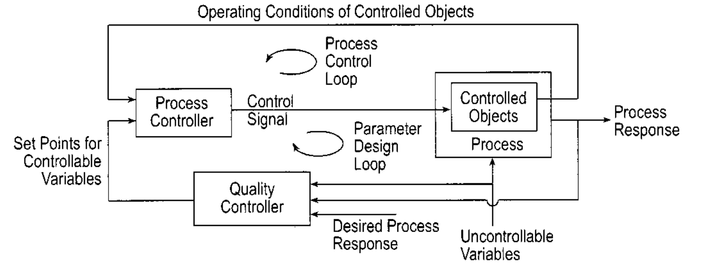


Fig. 3. Proposed framework for performing on-line parameter design.

As stated earlier, the focus of this paper is on time-invariant products and processes, and hence, the model-building process can be carried out off-line. In solving this on-line parameter design problem, the following assumptions are made.

1) The process time constant is relatively large in comparison with the rate of change of uncontrollable variables.
2) Uncontrollable variables are observable during production and unit operation.
3) Scales for the controllable variables and response variables are continuous.

A. Identification Mode

Let $\mathbf{x} = (x_1, \ldots, x_K, x_{K+1}, \ldots, x_N)^T$ be a column vector of K controllable variables and $N-K$ uncontrollable variables, where $K \leq N$. Let $\mathbf{y} = (y_1, \ldots, y_M)^T$ be a vector of M quality response characteristics of interest. For time-invariant systems, the quality vector y can be written as

$$\mathbf{y} = \mathbf{f}(\mathbf{x}). \tag{6}$$

In most cases, due to economic, time, and knowledge constraints, there exists no accurate mechanistic model for $\mathbf{f}$, and it has to be estimated in an empirical fashion. We recommend MLPs for modeling $\mathbf{f}$, given their universal approximation properties [6] and extreme success discussed in the literature

with regard to accurate approximation of complex nonlinear functions [10].

In contrast to some pattern-recognition problems and other function approximation problems, in general, off-line planning, design, and execution of experiments for modeling product/process response characteristics can be very time consuming and expensive. At the initial stage, it is not uncommon to see fractional factorial designs' being utilized for screening significant controllable and uncontrollable variables. Even second-phase experiments tend to use some form of a central composite design. The point here is that the size of the data set normally available for product/process identification is very limited. This makes division of the data set between training and testing more difficult, but does not prevent it. As the name implies, the training data set will be used for training the MLP network to approximate $\mathbf{f}$ from (6) as follows:

$$\tilde{\mathbf{f}}(\mathbf{x}) \approx \mathbf{f}(\mathbf{x}) \tag{7}$$

such that

$$J_I = \left\|\tilde{\mathbf{f}}(\mathbf{x}) - \mathbf{f}(\mathbf{x})\right\|_{\text{Identification}} \le \varepsilon \tag{8}$$

for some specified constant $\varepsilon \ge 0$ and a suitably defined norm (denoted by $||.||_{\text{Identification}}$).The testing data set will facilitate evaluation of the generalization characteristics of the network, i.e., the ability of the network to perform interpolations and make unbiased predictions. We use a multifold cross-validation method [10] for designing (i.e., determining the architecture in terms of number of hidden layers, nodes per different hidden layers, and connectivity) and building MLP models for approximating quality response characteristics. The method primarily involves dividing the available data set into S segments, $S > 1$, and using $S - 1$ segments at a time to train the network and testing it on the left-out segment. The process is repeated S times, each time using a different segment for validation. Networks with different configurations are compared with respect to their cumulative (or average) testing error from all S segments. Once the best configuration is identified, the network is trained using the complete data set (i.e., all S segments at once). Several other guidelines regarding selection of potential network configurations and their training are discussed in the literature and are not repeated here [10], [13]–[15].

B. On-Line Parameter Design Mode

Once the product/process identification is completed, parameter design can be performed on-line using the MLP model $\tilde{\mathbf{f}}(\mathbf{x})$. Let $\mathbf{y}_d = (y_{d1}, \ldots, y_{dM})^T$ denote the vector of M desired/target quality response characteristics of interest. The objective is to determine the optimal levels for the K controllable variables, x_1 through x_K, to

$$\text{minimize } J_{\text{PD}} = ||\mathbf{y} - \mathbf{y}_d||_{\text{ParameterDesign}} = ||\mathbf{f}(\mathbf{x}) - \mathbf{y}_d||_{\text{ParameterDesign}}, \tag{9}$$

for a suitably defined norm (denoted by $||.||_{\text{ParameterDesign}}$) on the output space. In the absence of any knowledge about $\mathbf{f}(\mathbf{x})$, the objective is to minimize the performance criterion

$$J_{\text{PD}} = ||\mathbf{y} - \mathbf{y}_d||_{\text{ParameterDesign}} \cong \left\|\tilde{\mathbf{f}}(\mathbf{x}) - \mathbf{y}_d\right\|_{\text{ParameterDesign}}. \tag{10}$$

The constraints would be those restricting the levels of the controllable variables to an acceptable domain.

1) Iterative Inversion Method: As discussed in Section II-C, through iterative inversion of the network, one can generate the optimal controllable variable input vector $[x_1, \ldots, x_K]$ that gives an output as close as possible to $\mathbf{y}_d$. In minimizing the performance criterion J_{PD}, the iterative gradient descent algorithm can be applied to obtain the desired input vector

$$x_j(s_{ii}{+}1) = x_j(s_{ii}) - \eta \cdot \frac{\partial J_{\text{PD}}(s_{ii})}{\partial x_j(s_{ii})} + \alpha\left(x_j(s_{ii}) - x_j(s_{ii}{-}1)\right) \tag{11}$$

for $1 \le j \le .K$. Here s denotes iteration step and η and α are the rates for inversion and momentum, respectively, in the gradient descent approach. If the least means square criterion was used as the performance criterion, for any MLP network, a derivation that parallels back-propagation algorithm will lead to the following gradient:

$$\frac{\partial J_{\text{PD}}(s_{ii})}{\partial x_j(s_{ii})} = -\sum_{i}^{\text{all}} \delta_i^{(1)}(s_{ii}) w_{ij}^{(1)}(s_{ii}). \tag{12}$$

The iterative inversion is performed until the controllable variables converge

$$\frac{\partial J_{\text{PD}}(s_{ii})}{\partial x_j(s_{ii})} \approx 0; \qquad x_j(s_{ii}+1) \approx x_j(s_{ii}). \tag{13}$$

If J_{PD} meets all the criteria for a strictly convex function in the input domain, gradient descent techniques lead to a global optimal solution. However, if it is not a convex function, iterative inversion method, being a gradient descent technique by definition, leads to a local optimal solution. Hence, it is necessary that the quality controller search through all the basins (multiple basins might exist in the case of nonconvex energy functions) to locate the global optimal levels for the controllable variables. Under the assumption that the step sizes taken along the negative gradient are not large enough to move into a different basin, the quality controller converges to the local minimum in the basin holding the starting point. However, one can incorporate a simulated annealing module (or other enhanced optimization techniques) to increase the likelihood of converging toward a global optimal solution.

IV. Case Study: Plasma Etching Process Modeling and On-Line Parameter Design

To facilitate evaluation of the proposed on-line intelligent quality controllers (IQCs), we chose to work with a semiconductor fabrication process, in particular, an ion-assisted plasma

etching process. As is pointed out by Himmel and May [8], plasma modeling from a fundamental physical standpoint has had limited success (see also [16]). Current physics-based models attempt to derive self-consisted solutions to first-principle equations involving continuity, momentum balance, and energy balance inside a high-frequency, high-intensity electric field. This is normally accomplished through expensive numerical simulation methods that are subject to many simplifying assumptions and are unacceptably slow. Since the complexity of practical plasma processes at the equipment level is presently ahead of theoretical comprehension, other efforts have focused on empirical approaches to plasma modeling involving response surface models [17]–[19] and neural network models [8], [9], and [20]. For a detailed discussion of ion-assisted plasma etching process, see Manos and Flamm [21] and Chapman [22].

As is pointed out by Card [23], in recent years, several groups conducting semiconductor manufacturing research have been developing empirical models for predicting process quality for plasma etch processes in terms of etch rate, uniformity, and oxide and photoresist selectivities. May *et al.* [17] and Himmel and May [8] focused on empirical modeling of the process using MLP networks and response surface models. Besides neural network modeling of plasma etching processes, Card [23] also proposed a framework for dynamic control of the process. However, Card's control (and modeling) framework does not account for the presence of any uncontrollable variables, and hence, their controller does not perform robust parameter design (in the Taguchi sense). On the contrary, the controllers proposed in this paper are primarily designed to optimize the controllable variables in light of the variation of the uncontrollable variables. In this sense, Card's controller is a traditional controller and the proposed controller is a robust parameter design controller (in the Taguchi sense). It is the presence of uncontrollable variables and the possibility of a lack of process inverse that calls for the utilization of the iterative inversion method for control in this paper. This paper utilizes the experimental data from Himmel and May [8] to demonstrate the performance of the proposed quality controller in improving process etch quality.

A. Experimental Technique

The study by May *et al.* [17] and Himmel and May [8] focused on the etch characteristics of n^+-doped polysilicon. In that study, etching was performed on a test structure designed to facilitate the simultaneous measurement of etch rates of polysilicon SiO_2 and photoresist. Test patterns were fabricated on 4-in-diameter silicon wafers. Approximately 1.2 μm of phosphorous-doped polysilicon was deposited over 0.5 μm of thermal SiO_2 by low-pressure chemical vapor deposition. A thick layer of oxide was grown to prevent etching through the oxide by the less selective experimental recipes. Poly resistivity was measured at 86.0 Ω·cm. Oxide was grown in a steam ambient at 1000 °C. One micrometer of Kodak 820 photoresist was spun-on and baked for 60 s at 120 °C.

The etching apparatus consisted of a lam Research Corporation Autotech 490 single-wafer parallel-plate system operating at 13.56 MHz. Film thickness measurements were performed on five points per wafer using a Nanometrics Nanospec AFT system and an Alphastep 200 Automatic Step Profiler. Vertical etch rates were calculated by dividing the difference between the pre- and postetch thickness by the etch time. Expressions for the selectivity of etching poly with respect to oxide ($S_{\rm ox}$) and selectivity of etching poly with respect to resist ($S_{\rm ph}$) are percent nonuniformity (U), and are given below

TABLE I
RANGES OF INPUT FACTORS

Parameter	Range	Units
Pressure (P)	200-300	watts
RF Power (Rf)	300-400	mtorr
Electrode Gap (G)	1.2-1.8	cm
CCl4 Flow	100-150	sccm
He Flow	50-200	sccm
O2 Flow	10-20	sccm

$$S_{\rm ox} = \frac{R_p}{R_{\rm ox}} \tag{14}$$

$$S_{\rm ph} = \frac{R_p}{R_{\rm ph}} \tag{15}$$

$$U = \frac{|R_{\rm pc} - R_{\rm pe}|}{R_{\rm pc}} * 100 \tag{16}$$

where
- R_p mean vertical poly etch rate over the five points;
- $R_{\rm ox}$ mean oxide etch rate;
- $R_{\rm ph}$ mean resist etch rate;
- $R_{\rm pc}$ poly etch rate at the center of the wafer;
- $R_{\rm pe}$ mean poly etch rate of the four points located about 1 in from the edge.

The overall objectives are to achieve high vertical poly etch rate, high selectivities, and low nonuniformity. For a detailed discussion of the study or the process, see May *et al.* [17] and Himmel and May [8].

B. Experimental Design

Of the nearly dozen different factors that have been shown to influence plasma etch behavior in the literature, the study focused on the following parameters regarded as most critical: chamber pressure (P), RF power (Rf), electrode spacing (G), and the gas flow rate of CCl_4. The primary etchant gas is CCl_4, but He and O_2 are added to the mixture to enhance uniformity and reduce polymer deposition in the process chamber, respectively. The six input factors and their respective ranges of variation are shown in Table I.

The experiments were conducted by May *et al.*[17] in two phases at the Berkeley Microfabrication Laboratory. In the first phase (screening experiment), a 2^{6-1} fractional factorial design requiring 32 runs was performed to reduce the experimental budget. Experimental runs were performed in two blocks of 16 trials each in such a way that no main effects or first-order interactions were confounded. Three center points were also added to check the model for nonlinearity. Analysis of the first stage of the experiment revealed significant nonlinearity and showed that all six factors are significant [17]. In order to obtain higher order models, the original experiment is augmented with a second experiment, which employed a central composite

TABLE II
MLP NEURAL NETWORK CONFIGURATION

Layer	Nodes Per Layer	Nodal Function	Data Scaling†
Input	6	Not Relevant	Yes (-1 to +1)
Hidden	12	Bipolar Sigmoid	Not Relevant
Output	4	Linear	Yes (-1 to +1)

† Before presenting the data as inputs and desired outputs to the neural network, it is scaled to a level easily managed by the network. In general, this facilitates rapid learning, and more importantly, gives equal importance to all the outputs in the network during the learning process (eliminating the undue influence of differing amplitudes and ranges of the outputs on the training process).

TABLE III
MLP NEURAL NETWORK TRAINING SCHEME

Training Algorithm	Back-propagation
S for Cross-Validation Scheme	9
Starting Learning Rate (η)	0.1
Learning Adaptation Rate	-10% (Reduction)
Minimum Learning Rate	0.00001
Starting Momentum (α)	0.001
Momentum Adaptation Rate	+15% (Growth)
Maximum Momentum	0.95
Parameter Adaptation Frequency	250 Epochs
Maximum Training Epochs†	20,000

† The training phase is also terminated if the percentage change of error with respect to training time/epochs is too small (<0.01% over 500 epochs) or if the training error is consistently increasing (>1% over 50 epochs).

circumscribed (CCC) Box–Wilson design [24]. In this design, the two-level factorial box was enhanced by further replicated experiments at the center as well as symmetrically located star points. In order to reduce the size of the experiment and combine it with results from the screening phase, a half-replicate design was again employed. The entire second phase required 18 additional runs. In total, there were 53 data points.

C. Process Quality Modeling Using MLP Networks

The task here is to design and train a neural network to recognize the interrelationships between the process input variables and outputs, using the 53 input–output data pairs. Experimental investigation has revealed that an MLP network with a single hidden layer can adequately model the input–output relationships of the process. An MLP network with six input nodes (matching the six process input factors), 12 nodes in the hidden layer (using a bipolar sigmoid nodal function in the hidden layer), and four nodes in the output layer (matching the four process outputs and carrying a linear nodal function), trained using the standard back-propagation algorithm, proved to be optimal (based on a full-factorial design considering multiple hidden nodes per layer, multiple learning rates, and multiple nodal functions) and yielded very good prediction accuracy. More information regarding the neural network configuration and training scheme is available in Tables II and III, respectively.

Table IV compares the performance of the neural network model with quadratic response surface method (RSM) models reported by May *et al.* [17] built using the same 53 data points, in terms of square root of the residual mean square error (mse) for each response. The mse is calculated as follows for any given response y_i:

TABLE IV
PERFORMANCE COMPARISON OF NEURAL NETWORK MODEL VERSUS RSM MODEL

Output	Sqrt(MSE_{RSM})	Sqrt(MSE_{NN})	Improvement
R_p	306.47 Å/min	114.76 Å/min	62.6%
U	6.60 [%]	2.63 [%]	60.2%
S_{ox}	0.90	0.50	44.4%
S_{ph}	0.26	0.09	65.4%

$$\text{MSE}_i = \frac{1}{D}\sum_{d=1}^{D}\left(y_i(d) - \hat{y}_i(d)\right)^2 \tag{17}$$

where

D number of experiments (data points);

$y_i(d)$ measured value;

$\hat{y}_i(d)$ corresponding model prediction for data point d.

Fig. 4 provides "goodness-of-fit" plots, which depict the neural network predictions versus actual measurements. In these plots, perfect model predictions lie on the diagonal line, whereas scatter in the data is indicative of experimental error and bias in the model.

D. On-Line Process Parameter Design Using MLP Model

To illustrate and evaluate the performance of the proposed IQCs, we simulate the manufacturing process, using the established MLP neural network model, with varying degrees of fluctuation in the uncontrollable variables. Here the three process gas flow rates (that of CCl_4, He, and O_2) are treated as uncontrollable variables, with different degrees of uncontrollability during different simulations. The strategy involves evaluating the performance of the IQC in the form of average deviation from desired target process outputs in standard deviations. Here, the standard deviations for the four different process outputs are calculated from the 53-point data set.

To facilitate a fair comparison of the performance of the IQC with traditional off-line parameter design approaches, we work with a modified Taguchi's inner/outer array parameter design method (for an account of his techniques, see [1]–[3]). Most traditional Taguchi parameter design methods evaluate the robustness of a given combination of controllable variable levels by conducting some sort of a factorial design in the uncontrollable variable space (at the given controllable variable level combination) and summarizing the results from the experimental design in the form of signal-to-noise ratios. In essence, these methods assume that the uncontrollable variables follow a uniform distribution during product/process operation. Since this assumption is not strictly true in most manufacturing processes, and hence not fair to results produced from any traditional parameter design approach, we chose to evaluate the robustness of the different combinations of the controllable variable levels using the actual time-series data of the uncontrollable variables (Section IV-D2 provides more discussion regarding this time-series data). In essence, we allow the off-line parameter design method to perform its best (for no assumptions are being made regarding the nature of the uncontrollable variables). We label this method

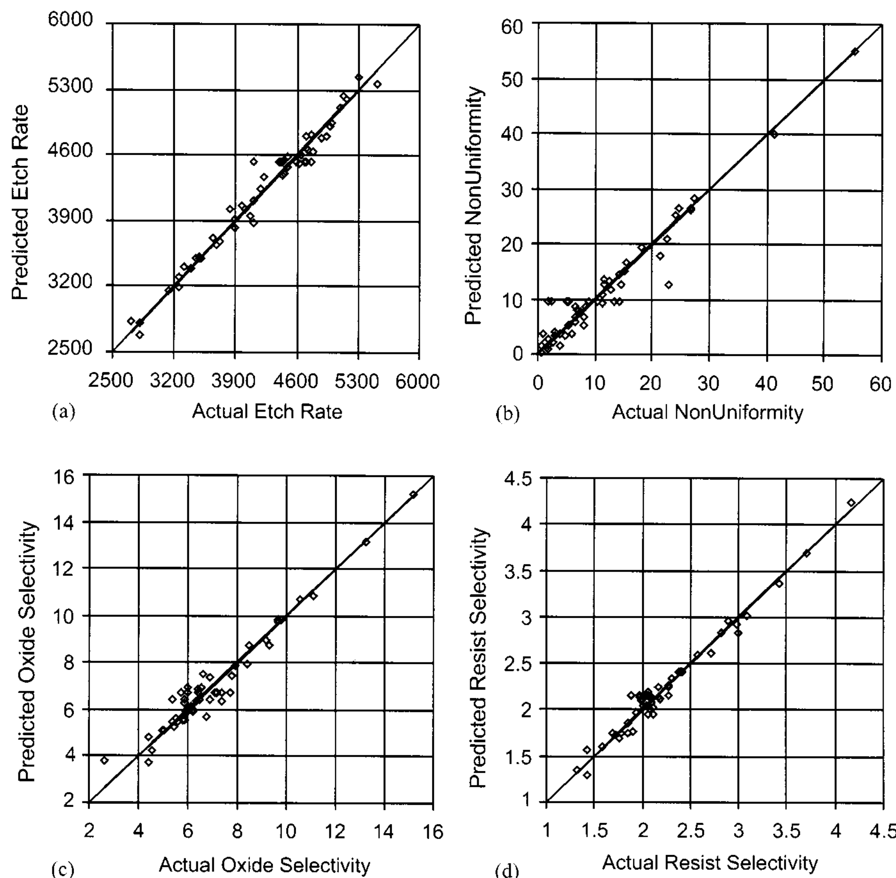


Fig. 4. Plots depicting neural model predictions versus actual measurements. (a) Predicted versus measured etch rate. (b) Predicted versus measured nonuniformity. (c) Predicted versus measured oxide selectivity. (d) Predicted versus measured resist selectivity.

as the pseudoparameter design (SPD) method. Thus, the SPD approach determines the optimal combination of controllable variable settings that lead to the least expected deviation from desired target process outputs, in light of the variation seen in the provided uncontrollable variable time-series data set. As is the case with most traditional parameter design approaches, the combinations of controllable variable levels considered represent the design points from a full factorial experimental design in the controllable variable space, using different resolution designs. It is normally common to see no more than three or four levels' being considered per controllable variable in off-line parameter design methods (so as to keep a tab on the required experimental resources). However, to enhance the fairness to SPD performance, we chose to consider five levels and eight levels per controllable variable, and we worked with full factorial designs.

1) Establishing Target Process Outputs: For the plasma etching process at hand, as was stated earlier, the overall objectives are to achieve high vertical poly etch rate (R_p), low nonuniformity (U), high oxide selectivity ($S_{\rm ox}$), and high resist selectivity ($S_{\rm ph}$). The optimum etch recipe that will lead to best etch responses was determined using the iterative inversion scheme and allowing all the six process input factors to be controllable. A comparison between the standard recipe (normally used for plasma etching) and the optimized recipe appears in Table V. Estimated etch responses for the standard and optimal recipes were determined using the neural network process model. Notably, significant improvement was to be obtained in all the four process responses. After the optimum recipe was determined, an experiment was undertaken to confirm the improvement of the etch responses. In this experiment, six wafers were identically prepared and divided into two equal groups and were approximately subjected to standard and optimized treatments. The results were consistent with the estimations. Hence, during the evaluation of the performance of IQC in comparison with SPD, it would be best to attempt to constantly achieve the optimized response (i.e., the optimized response values shown in Table VI will be used as targets) in spite of variation in the uncontrollable variables (i.e., the three gas flow rates).

TABLE V
STANDARD AND OPTIMIZED ETCH RECIPES

Parameter	Standard Recipe	Optimized Recipe
Pressure (mtorr)	300	300
RF Power (W)	280	300
Electrode Spacing (cm)	1.5	1.43
CCl_4 Flow (sccm)	130	150
He Flow (sccm)	130	50
O_2 Flow (sccm)	15	10

2) Simulation of Uncontrollable Variables: In general, the process gas flow rates tend to exhibit strong autocorrelation with

TABLE VI
ESTIMATED STANDARD AND OPTIMIZED RESPONSES

Response	Std. Recipe	Opt. Recipe	% Change
R_p	4100.00 Å/min	4663.33 Å/min	13.74
U	12.17 [%]	9.11 [%]	-25.14
S_{ox}	9.26	15.38	66.09
S_{ph}	3.10	4.90	58.06

respect to time. In this paper, we simulate the gas flow rates as an AR(1) autocorrelated process that carries the following model:[1]

$$x(t) = \phi x(t-1) + \varepsilon_t \tag{18}$$

where t denotes time and the ε_t's are independently identically distributed normal with zero mean and variance of σ_ε^2. The value of ϕ has to be restricted within the open interval of $(-1, 1)$ for the AR(1) process to be stationary.

In this paper, the simulations were conducted by setting ϕ at 0.9. As was stated earlier, for evaluation of the performance of the proposed IQC, we need to simulate the manufacturing process with varying degrees of fluctuation in the uncontrollable variables. The strategy here is to generate the AR(1) process data by setting σ_ε equal to one and then linearly scaling the generated data to the desired range of variation. The degree of variation (DOV) is allowed to be between zero and one, where zero denotes no change in the level of the uncontrollable variable and one denotes the case where the range of the generated data spans the tolerated range for the particular variable, shown in Table I. It is important to note that when the DOV is set to zero, the final levels generated should coincide with the optimal recipe levels, and any increase in the DOV will oscillate the levels of the uncontrollable variables in and around the optimal recipe levels. Fig. 5 illustrates the data generated for CCl_4 gas flow rate at different degrees of variation. Note that when the DOV is set to zero, the values match with the optimized recipe level (150 sccm) for the variable (CCl_4). All the simulations discussed in this paper involve 10 000 discrete time instants, and the starting random seeds are different for the three AR(1) processes for the uncontrollable variables. This ensures that the patterns are not the same for all three uncontrollable variables even if the DOV is set the same for any given simulation. The DOVs chosen for evaluation of the proposed IQC are as follows: 0.001, 0.005, 0.01, 0.02, 0.03, 0.04, 0.05, 0.1, 0.25, 0.5, 0.75, and 1.0.

3) Comparison of Performance of IQC and SPD: For the different simulations, the iterative inversion for the IQC is performed using an iterative inversion rate of 0.05 allowing a maximum of 5000 iterative inversions at any discrete time instant. Additional information regarding the iterative inversion scheme used by the IQC is available in Table VII. For SPD, the potential combinations for the levels of the controllable inputs were generated using a full factorial design with two different resolutions per variable, five and eight, resulting in 5^3 and 8^3 combinations. All the combinations are evaluated with the 10 000-point data set generated for each DOV, and the combination that leads to optimal overall performance is picked to represent the performance of SPD. What constitutes optimal is measured in terms

[1]In fact, additional simulations treating the gas flow rates as a random walk and other ARMA models led to results similar to those reported in this paper.

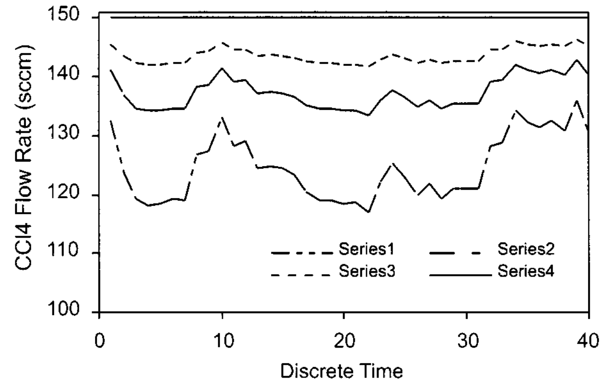


Fig. 5. Simulation of CCl_4 flow rate with different degrees of variation.

TABLE VII
IQC ITERATIVE INVERSION SCHEME

Starting Iterative Inversion Rate	0.05
Inversion Adaptation Rate	-10% (Reduction)
Minimum Inversion Rate	0.00001
Iterative Inversion Momentum	0.0
Parameter Adaptation Frequency	25 Iterations
Maximum Inversion Iterations†	5,000
Number of Iterative Inversion Starting Points	9‡

† The iterative inversion is also terminated if the percentage change of error with respect to time is too small (<0.1% over 10 iterations) or if the error is consistently increasing (>1% over 50 iterations).
‡ Two levels per controllable variable (dividing the range into three equal parts), leading to 2^3 combinations for the three controllable variables. An additional starting point is the center of the overall search space.

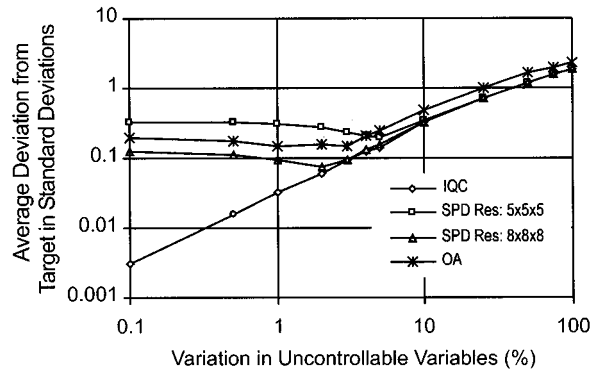


Fig. 6. Performance comparison of IQC, SPD, and OA.

of average deviation from target metric (AD), the smaller the better, and defined as follows:

$$\text{AD} = \frac{1}{M}\sum_{i=1}^{M}\sum_{t=0}^{T} w[i]\cdot|y_{\text{di}}(t) - \hat{y}_{\text{oi}}(t)| \tag{19}$$

where

$y_{\text{di}}(t)$ desired target output for process output i at time instant t;

$\hat{y}_{\text{oi}}(t)$ corresponding optimal output level achieved using iterative inversion scheme;

$w[i]$ weight assigned to process output i;

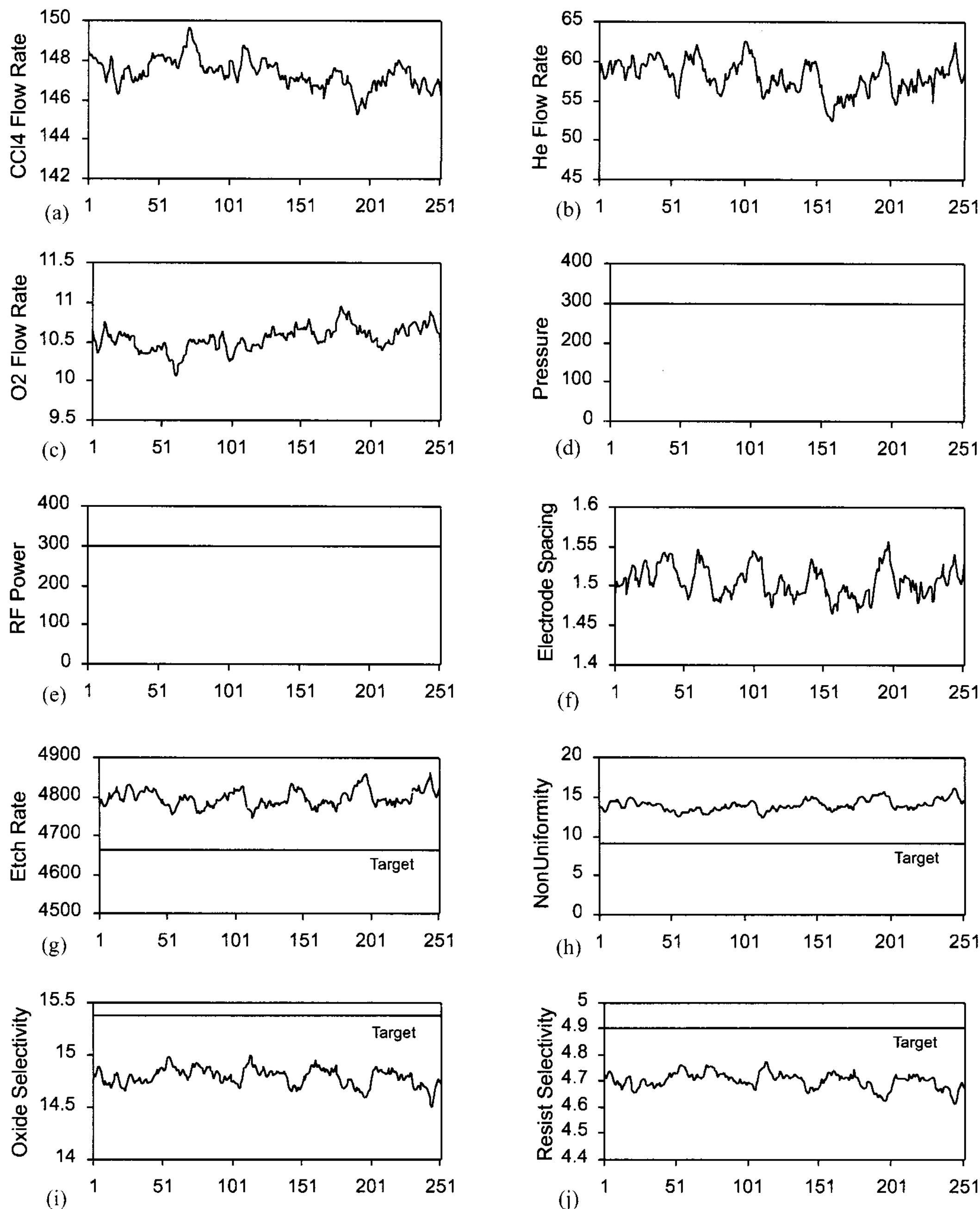


Fig. 7. Illustration of performance of IQC when the degree if variation set at 0.1. (a) Uncontrollable but observable CCl_4 flow rate (sccm). (b) Uncontrollable but observable He flow rate (sccm). (c) Uncontrollable but observable O_2 flow rate (sccm). (d) On-line optimized pressure (mtorr). (e) On-line optimized RF power (watts). (f) On-line optimized electrode gap (cm). (g) Process etch rate (Å/min). (h) Etch nonuniformity (%). (i) Oxide selectivity. (j) Photoresist selectivity.

T length of the simulation run in terms of discrete time instants;

M number of process outputs.

The weights for calculating the average deviation are defined as follows:

$$w[i] = \frac{1}{\sigma_{y_i}} \tag{20}$$

where σ_{y_i} denotes the standard deviation of the ith process output determined from the complete experimental data set. Such a weight definition facilitates a relatively fair calculation of the combined average deviation in standard deviations.

The performance comparisons of the IQC and SPD schemes in terms of average deviation from target for different degrees of variation in the uncontrollable variables, shown in Fig. 6, clearly illustrates the ability of IQC to significantly reduce the average deviation from target, when feasible. In addition, as expected, the IQC always outperforms the SPD approach, given that the IQC approach optimizes the levels of the controllable variables at all instants of time. Obviously, the improvement in general will depend on the particular process at hand and its sensitivity

to deviations in uncontrollable variables. Fig. 6 also shows the results of an approach (referred to as OA approach) that would set the controllable variables as constants at the optimal recipe levels shown in Table V. In this case, it appears that the OA approach fares somewhere in between the two SPD strategies (i.e., resolution of five levels per variable versus eight levels per variable) when the DOV is low and performs the worst when the DOV is relatively high.

Fig. 7 illustrates the performance of the IQC when the DOV in the uncontrollable variables is set at 0.1. Note that RF power and pressure remained the same throughout the first 251 discrete time instants shown in the figure (of the 10 000 time instant simulation). This is attributed to the fact that the iterative inversion procedure was suggesting that RF power be increased beyond 300 W and pressure beyond 300 mtorr; however, these levels already represent the boundary of the acceptable range for these variables (see Table I). Note that these levels do coincide with the optimal recipe levels shown in Table V. Also, note that the AD at any instant can be zero if and only if the levels of the uncontrollable variables at that instant during the simulation match optimal recipe levels from Table V. Obviously, during all the simulations, DOV is set to be greater than zero, and hence, it is impossible to meet all four targets for the four outputs at any instant. It is possible for IQC to deliver process performance that outperforms some of the targets; however, the IQC is not concentrating just on any one output but on all four outputs simultaneously (it is striving to minimize AD at all instants of time).

With respect to computational complexity, for all the simulations discussed above, on the average, the processing time for iterative inversion at any time instant was on the order of few microseconds on a Pentium II 300-MHz processor (this includes the ten to 100 iterations necessary on the average to converge toward the optimized controllable variables for each of the iterative inversion starting points). The far superior performance of IQC over SPD when the DOV is set to small values can be explained. When DOV is small, the levels of the uncontrollable variables are relatively close to optimal values, and hence the ability of IQC to more or less reduce the AD to zero. If DOV is set to be high, the levels of the uncontrollable variables are often very different from the optimal levels, and hence the inability of IQC to significantly reduce AD. The overall trajectory of IQC performance remained more or less the same even when the uncontrollable variables are simulated using other ARMA models, substantiating the above argument. Obviously, in the case of SPD, the focus is on average performance (maximizing robustness) while the controllable variables are held constant throughout the simulation, and hence will never be able to outperform the IQC approach. For most production processes, there will be some level of controllability on uncontrollable variables (meaning that DOV will be relatively small). For these processes, IQC should lead to significant improvements over off-line parameter design approaches. However, note that the results shown in this paper are only representative of the particular process involved. As mentioned earlier, the improvement in general will depend on the particular process at hand and its sensitivity to deviations in levels of uncontrollable variables from optimal values.

V. Conclusion

An on-line parameter design method that accounts for extra information available about observable uncontrollable factors in products and processes is introduced. Feed-forward neural networks are recommended for modeling the quality response characteristics due to their nonparametric nature, strong universal approximation properties, and compatibility with adaptive systems. An iterative inversion scheme is proposed for on-line optimization of controllable variables.

Deployment of the proposed on-line parameter design method on an reactive ion plasma etching semiconductor manufacturing process has revealed the ability of the method to significantly improve product/process quality beyond contemporary off-line parameter design approaches. In particular, the authors strongly believe that the proposed methods might be of great value in dealing with products/processes with low capability and numerous uncontrollable variables that have an impact on product/process output.

Acknowledgment

The authors would like to thank the two referees and the editor for their invaluable comments and suggestions that led to an improvement in the quality of the overall paper.

References

[1] G. Taguchi, *Introduction to Quality Engineering*. White Plains, NY: UNIPUB/Kraus , 1986.

[2] ——, *System of Experimental Design: Engineering Methods to Optimize Quality and Minimize Cost*. White Plains, NY: UNIPUB/Kraus, 1987.

[3] G. Taguchi and Y. Wu, *Introduction to Off-Line Quality Control*. Tokyo, Japan: Central Japan Quality Control Assoc., 1980.

[4] V. N. Nair, "Taguchi's parameter design: A panel discussion," *Technometrics*, vol. 34, pp. 127–161, 1992.

[5] M. Pledger, "Observable uncontrollable factors in parameter design," *J. Quality Technol.*, vol. 28, pp. 153–162, 1996.

[6] K. Hornik, M. Stinchcombe, and H. White, "Multi-layer feed-forward networks are universal approximators," *Neural Networks*, vol. 2, 1989.

[7] F. Nadi, A. Agogino, and D. Hodges, "Use of influence diagrams and neural networks in modeling semiconductor manufacturing processes," *IEEE Trans. Semiconduct. Manufact.*, vol. 4, pp. 52–58, 1991.

[8] C. D. Himmel and G. S. May, "Advantages of plasma etch modeling using neural networks over statistical techniques," *IEEE Trans. Semiconduct. Manufact.*, vol. 6, pp. 103–111, 1993.

[9] B. Kim and G. S. May, "An optimal neural network process model for plasma etching," *IEEE Trans. Semiconduct. Manufact.*, vol. 7, pp. 12–21, 1994.

[10] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed. Upper Saddle River, NJ: Prentice-Hall, 1999.

[11] A. Linden and J. Kindermann, "Inversion of multi-layer nets," in *Proc. Int. Joint Conf. Neural Networks (IJCNN)*, June 1989, pp. 425–430.

[12] S. M. Weiss and C. A. Kulikowski, *Computer Systems that Learn*. San Mateo, CA: Morgan Kaufmann , 1991.

[13] A. S. Weigand, D. E. Rumelhart, and B. A. Huberman, *Advances in Neural Information Processing Systems*. San Mateo, CA: Morgan Kaufmann , 1992.

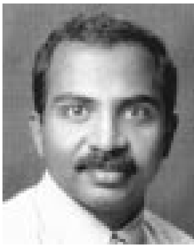
[14] S. A. Solla, "Learning and generalization in layered neural networks: The contiguity problem," in *Neural Networks from Models to Applications*, G. Dreyfus and L. Personnaz, Eds. Paris, France: I.D.S.E.T., 1989.

[15] E. B. Baum and D. Haussler, "What size net gives valid generalization?," *Neural Computation*, vol. 1, pp. 151–160, 1989.

[16] M. A. Lieberman and A. J. Lichtenberg, *Principles of Plasma Discharges and Materials Processing*. New York: Wiley, 1994.

[17] G. S. May, J. Huang, and C. J. Spanos, "Statistical experimental design in plasma etch modeling," *IEEE Trans. Semiconduct. Manufact.*, vol. 4, pp. 83–98, 1991.

[18] P. E. Riley and D. A. Hanson, “Study of etch rate characteristics of SF_6/He plasmas by response surface methodology: Effects of inter-electrode spacing,” *IEEE Trans. Semiconduct. Manufact.*, vol. 2, Nov. 1989.
[19] M. W. Jenkins, M. T. Mocella, K. D. Allen, and H. H. Sawin, “The modeling of plasma etching processes using response surface methodology,” *Solid State Tech.*, Apr. 1986.
[20] E. A. Rietman and E. R. Lory, “Use of neural networks in modeling semiconductor manufacturing processes: An example for plasma etch modeling,” *IEEE Trans. Semiconduct. Manufact.*, vol. 6, pp. 343–347, Nov. 1993.
[21] D. M. Manos and D. L. Flamm, Eds., *Plasma Etching: An Introduction*. Boston, MA: Academic , 1989.
[22] B. Chapman, *Glow Discharge Processes, Sputtering and Plasma Etching*. New York: Wiley, 1980.
[23] J. P. Card, “Dynamic neural control for a plasma etch process,” *IEEE Trans. Neural Networks*, vol. 8, no. 4, pp. 883–901, Jul. 1997.
[24] G. E. P. Box, W. Hunter, and J. Hunter, *Statistics for Experimenters*. New York: Wiley, 1978.



Ratna Babu Chinnam received the B.S. degree (with honors) in mechanical engineering from Mangalore University, India, in 1988 and the M.S. and Ph.D. degrees in industrial engineering from Texas Tech University, Lubbock, in 1990 and 1994, respectively.

He is currently an Associate Professor in the Industrial and Manufacturing Engineering Department, Wayne State University, Detroit, MI. Prior to that, he was an Assistant Professor with the Industrial and Manufacturing Engineering Department, North Dakota State University, Fargo, from 1994 to 2000. He is the author of several technical publications in the areas of intelligent process-quality control and process monitoring in manufacturing systems. His research interests include intelligent manufacturing, process monitoring, manufacturing process control, and quality and reliability engineering.

Dr. Chinnam is a member of Alpha Pi Mu, the American Society for Quality, the Institute of Industrial Engineers, the Society of Manufacturing Engineers, and the North American Manufacturing Research Institute.

Jie Ding received the B.S. and M.S. degrees in mechanical engineering from Jiaotong University, China, in 1993 and 1996, respectively, and the M.S. degree from the Industrial and Manufacturing Engineering Department, North Dakota State University, Fargo, in 1999.

She is currently with Evolve, San Francisco, CA. From 1998 to 2000, she was with Dassault Systems, Los Angeles, CA, as an Application Engineer. Her research interests are in software design, intelligent quality control, and computer-aided manufacturing.

Ms. Ding is a member of the Institute of Industrial Engineers.

Gary S. May (S'85–M'90–SM'97) received the B.S. degree in electrical engineering from Georgia Institute of Technology, Atlanta, in 1985 and the M.S. and Ph.D. degrees in electrical engineering and computer science from the University of California at Berkeley in 1987 and 1991, respectively.

He is currently an Associate Professor in the School of Electrical and Computer Engineering and Microelectronics Research Center, Georgia Institute of Technology. He was a Member of Technical Staff at AT&T Bell Laboratories, Murray Hill, NJ. His research is in the field of computer-aided manufacturing of integrated circuits. His interests include semiconductor process and equipment modeling, process simulation and control, automated process and equipment diagnosis, and yield management.

Dr. May is a National Science Foundation “National Young Investigator,” and is Editor-in-Chief of IEEE TRANSACTIONS ON SEMICONDUCTOR MANUFACTURING. He was a National Science Foundation and an AT&T Bell Laboratories graduate fellow. He is Chairperson of the National Advisory Board of the National Society of Black Engineers.