Application of Multivariable and Intelligent Control Strategies for Improving Plasma Characteristics in Reactive Ion Etching

Nicolae Tudoroiu

A thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfilment of the Requirements

for the Degree of Doctor of Philosophy at

Concordia University

Montreal, Quebec, Canada

April 2001

ABSTRACT

Application of Multivariable and Intelligent Control Strategies for Improving Plasma Characteristics in Reactive Ion Etching

Nicolae Tudoroiu

Concordia University, 2001

Reactive Ion Etching (RIE) is a critical technology for modern VLSI circuit fabrication and is used at many stages of the manufacturing process. Several real-time control strategies such as *Proportional-Integral* (PI) self-tuning, *Linear Quadratic Gaussian* (LQG), stochastic adaptive control, neurocontrol, robust and hierarchical control based on both linear and nonlinear models of the Plasma Generation Subsystem (PGS) are developed to improve plasma characteristics in the Reactive Ion Etching process. The proposed approaches result in superior accuracy and performance when compared to results that are available in the literature. The identification process (prediction error approach) to determine linear Auto Regressive Moving Average ($ARMA$) models of the PGS is based on the computationally efficient recursive least squared (RLS) procedure. This is an alternative to the use of Kalman filter that is based on state estimation. The massively parallel processing, nonlinear mapping, and self-learning abilities of neural networks are exploited in the development of intelligent control systems. Neurocontrollers enhance RIE manufacturability and may be used for process optimization, control, and diagnosis. A hierarchical real-time control strategy is developed that automatically selects during each specific operating interval the best real-time control strategy for tracking the *dc self bias voltage* and *fluorine concentration* set points. It is shown that the proposed methodology results in higher performance and is computationally more efficient than that using a single control strategy that is dependent on a range of operating conditions.

DEDICATION

This dissertation is dedicated, with love, to my mother, to my wife, to my daughters, and to my brother.

ACKNOWLEDGEMENTS

I would like to express my sincere appreciation to Drs. K. Khorasani and R.V. Patel for their guidance, support, and encouragement throughout the course of the research described in this thesis. I would also like to thank Dr. R.V. Patel for proposing the topic of this research and to acknowledge his help to acquire from the University of Michigan the experimental input-output data sets for the RIE process used in this thesis. Finally, I would like to express my sincere thanks to Dr. P. P. Khargonekar and his group of the University of Michigan for providing us with extensive and complete sets of experimental input-output data for the RIE process.

List of Symbols and Abbreviations

Δ—*first difference operator*

$E\{*\}$— *expectation operator*

$\%OS$— *overshoot*

t_i, t_r— *rise time, settling time*

ϵ— *steady-state error*

σ_i— *singular values of the matrix*

e—*standard deviation error*

γ_{opt}—*optimal value of the iteration parameter*

$\hat{\Phi}(k)$— *predictive value of the vector* $\Phi(k)$

$\hat{\alpha}_{ij}$, $\hat{\beta}_{ij}$— *estimated values of the coefficients* α_{ij}, β_{ij}

$\epsilon(k)$— *prediction error*

$\Psi^T(k)$— *transposed regressor vector*

$\Theta(k)$—*parameter vector*

$\hat{\Theta}(k)$— *estimated value of the parameter vector*

K— *steady-state Kalman gain filter*

$L(k)$— *identification gain for the stochastic adaptive control*

$P(k)$— *covariance estimation matrix*

Λ— *forgetting factor in the RLS procedure*

x, q—*system and accumulator vector states*

$\hat{x}(k)$— *estimated value of the vector x(k)*

$r(k)$— *reference input vector*

ω— *frequency [Hz]*

$f(.)$, $g(.)$—*function operators*

$\zeta(k)$—*nonstationary disturbances*

$u(k)$— *input vector*

$y(k)$, $z(k)$— *output vectors*

$u_{sp,\, y_{sp}}$—*set point vectors for the input and for the output, respectively*

$\hat{y}$—*predictive value of the output*

y^*, y_p , y_d—*target values of the output*

J—*cost function*

r— *control weight in the cost function*

Q, R— *weighting matrices in the cost function*

$e(k)$—*disturbance vector*

ϵ—*standard deviation error*

z—*complex variable in the discrete domain*

w—*process noise*

v—*measurement noise*

R_v, $Q_w(R_v)$, R_{x_0}— *covariance matrices*

R_{wv}—*cross-correlation matrix of the vectors w, v*

$Mean_w$—*mean value of the vector w*

$R_n(k)$, $R_\epsilon(k)$— *covariance matrices for the stochastic adaptive control*

$n(k)$— *zero-mean Gaussian modeling error*

$H(z)$, $G(z)$—*open-loop transfer matrices of the Plasma Generation Subsystem*

$F(z)$—*robust controller transfer matrix*

$F(k)$— *gain control matrix for the stochastic adaptive control*

$H^{-1}(z)$— *inverse transfer matrix*

$\{h_{ij}\}$—*steady-state open-loop gain*

$\{\mu_{ij}\}$—*components of the Relative Gain Array matrix (RGA)*

$\{\gamma_{ij}\}$—*steady-state gain (Γ—matrix components)*

Y, Σ, U^T, Y_d, Σ_d, U_d^T—*matrix components for the singular value decomposition (SVD) of the transfer matrices H(1) (for the models without delay) and* $H_d(1)$ *(for the models with delay)*

U^T, U_d^T— *transposed matrices* U, U_d

w^T— *transposed weight vector*

E—*network error*

$\nabla_w E$—*gradient matrix of* E *with respect to the vector* w

$H(w)$, $\hat{H}(w)$—*Hessian matrices with respect to the vector* w

μ—*relaxation parameter for the Levenberg-Marquardt algorithm*

I_2— *two-dimensional identity matrix*

$O_{m\times n}$— *matrix with all the elements equal to zero,* $m\times n$*-dimensional*

$U_{m\times n}$— *matrix with all the elements equal to one,* $m\times n$*-dimensional*

H^∞—*H-infinity techniques used in robust control*

$|z|$ — *module of the complex variable* z

$\left\|T_{Y_{sp}U_{sp}}(z)\right\|_\infty$—*H-infinity norm of the closed-loop transfer matrix* $T_{Y_{sp}U_{sp}}$

RIE— *Reactive Ion Etching*

PGS— *Plasma Generation Subsystem*

WES— *Wafer Etch Subsystem*

$MIMO$— *Multi-Input Multi-Output systems*

RLS— *Recursive Least Squared*

$ARMA$— *Auto Regressive Moving Average*

$ARIMA$— *Auto Regressive Integrated Moving Average*

PI self – tuning control– Proportional Integral self-tuning control

PID– Proportional Integral Derivative

LQG control– Linear Quadratic Gaussian control

SMVAC– stochastic minimum variance adaptive control

CVD– Chemical Vapour Deposition

LPCVD– Low plasma Chemical Vapour Deposition

PECVD - Plasma Enhanced Chemical Vapour Deposition

TMAH– Tetramethyl ammonium hydroxide

[F]– Fluorine concentration

V_{bias}*– self bias voltage*

Si– silicon

SiO_2*– silicon dioxide*

N– Nitrogen

O– Oxygen

H– Hydrogen

Ar– Argon

CF_4*– tetra fluorocarbon gas*

SiF_2, SiF_4*–fluorosilicon gases*

CF_3Cl*–chlorofluorocarbon gas*

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This thesis is motivated by the fact that presently most semiconductor manufacturing equipments are designed to be operated in an open-loop mode. Consequently, the manufacturing performance of these equipments are not as good as desired.

In order for the Canadian semiconductor industry improve its competitiveness, it is critical that its fabrication facilities produce highly advanced products at low cost. To achieve these goals, these fabrication facilities must be equipped with processing systems which can perform their functions with very high accuracy and throughput but with low overall costs. Reactive Ion Etching (RIE) is a critical technology for modern *VLSI* circuit fabrication which is used in many stages of the manufacturing process [1]. Silicon dioxide films are of significant interest as an interlayer dielectric material for integrated circuits and multichip modules (*MCM's*). The patterning of these films is of crucial importance in semiconductor manufacturing. RIE in radio-frequency (*RF*) glow discharges is among the most commonly used methods for forming patterns via holes in between metal layers of an *MCM* and for achieving the level of detail necessary to define small features in film [2].

Etching is a process through which a desired pattern is transferred to a silicon

wafer by removing material, such as silicon (Si) or silicon dioxide (SiO_2) via the interaction of fluorine [F] or other chemicals such as chlorine or bromine with the wafer and exhausting the reaction products [1]. Some of the most important variables used for determining the success of the etching process are *selectivity, uniformity, anisotropy*, and *etch depth*:

(*i*) *selectivity* refers to the ability to etch, for example, Si without etching either SiO_2 or photoresist.

(*ii*) *uniformity* is the requirement that etching be spatially uniform across the wafer.

(*iii*) *anisotropy* is the ability to etch vertically while minimizing horizontal etching, thus creating vertical walls, and

(*iv*) *etch depth* is the ability to remove exactly a desired amount of material.

The relative importance of these characteristics depends upon the function of the layer being etched. For instance, during fabrication of a transistor gate region, a polysilicon layer is etched down to the gate oxide. Selectivity to the oxide is of primary importance since only a small variation in the oxide layer is tolerable.

The other key requirement is critical dimension control (*etch depth*). Experiments can be designed to locate an operating point which offers good performance in the characteristics of most importance while maintaining adequate performance in the characteristics of secondary interest. Models generated from experimental data are used to achieve this goal. Better control of these variables translates directly into improved yield, finer line widths and hence higher device density and higher throughput.

The physical and chemical mechanisms involved in RIE process are very complex and are not fully understood at the present time. Since the focus of this research is on the demonstration of the potential benefits of using feedback control systems, an

extremely simple plasma process is used: namely a CF_4 gas system for the etching of unmasked wafers with a material stack of polysilicon substrate. This simple system has been studied extensively with many known material parameters and also a reasonably advanced understanding of the etch mechanisms and pathways is available [1]. The plasma process could increase in complexity by adding hydrogen, chlorine or oxygen to enhance the selectivity and etch rate, and A_r/N_2 for actinometry. Also, if the number of wafers in reactor chamber increases the performance of the etching process decreases, especially the etch rate or etch depth.

RIE is a *low pressure, low temperature* plasma system. The plasma is generated by capacitively coupling a RF (13.56 MHz) power source to one electrode, which has a smaller surface area than the ground electrode. This leads to dissociation and ionization of the feed gas. Consequently, it generates a chemically active mixture of electrons, ions and free radicals. Due to the fact that the electrons are more mobile than ions, a *dc self bias voltage* is developed across the electrodes to achieve current continuity. This self bias voltage accelerates ions toward the surface of the wafer. The free radicals diffuse to the surface of the wafer where they react with the exposed silicon surface. The surface reactions are quite complicated and are not completely understood. However, in simple terms, the fluorine atoms react with silicon atoms and produce various volatile components such as SiF_4 and SiF_2, desorbed possibly with the aide of the impinging ions. In addition, the impinging ions further enhance these etching processes. On the other hand, various polymers are formed as a result of chemical reactions between the radicals and the surface material. These inhibit the etching process. The polymers formed on the side walls are largely unaffected by the ion bombardment and thus facilitate *highly anisotropic etching* [1]. Polymers on the horizontal surface are removed by the *impinging ions*, provided the polymer film is not too thick and the ions have sufficient energy, and this allows the etching

process to continue. The *impinging ions* can also physically sputter Si atoms, thereby etching the wafer surface, or cause surface activation, thereby speeding up the etch process. For this reason it is useful to conceptualize RIE as consisting of two distinct but interacting mechanisms:

(i) chemical etching caused by radicals, and

(ii) physical etching caused by ion bombardment [1].

Therefore, etch characteristics can be adjusted by carefully controlling the plasma species composition and ion energy.

The lack of feedback control in these systems is generally considered as one of the main challenging problems facing the semiconductor manufacturing industry. This in particular is a major impediment to reliable operation of low pressure reactive plasma systems [1].

The principal motivation for introducing advanced control techniques in these systems is that by controlling appropriate plasma parameters (the concentrations of the reactive radicals and ions and ions energy), it is possible to improve the etch performance of the reactive ion etchers, namely their selectivity, uniformity, anisotropy and etch depth. The current state of knowledge in RIE does not yet allow for a definitive choice of the key plasma parameters to be controlled. For example in [3] four measured variables (namely $[F]$, $[CF_2]$, *radicals* $[CO_x]$, and V_{bias}), four manipulated variables (namely % O_2, *pressure*, *power*, and *flow rate*) and seven performance variables (namely *Si etch rate*, SiO_2 *etch rate*, *Si* / SiO_2 *selectivity*, SiO_2 *anisotropy*, *Si uniformity*, SiO_2 *uniformity*, and *Si anisotropy*) were considered for the RIE of silicon and silicon dioxide in CF_4/O_2 and CF_4/H_2 plasma. Furthermore, in [4], [5]-[8] only two manipulated variables (namely *power* and *throttle valve position*), two measured variables of the key plasma parameters to be controlled (namely V_{bias} and $[F]$), and four performance variables for RIE (namely *etch depth*, *selectivity*,

uniformity, and *anisotropy*) were considered. There has been a major interdisciplinary effort at many major US universities and institutes, such as the University of Michigan, the University of California at Berkeley, the University of Texas at Austin, Georgia Institute of Technology, Carnegie Mellon University, the University of Colorado and Massachusetts Institute of Technology, to name a few, for improving the manufacturing characteristics of semiconductor processing equipment [1], [9].

The interest in use of plasma processes is increasing, particularly in thin film etching for *VLSI* applications. Plasma processing allows a more effective control of the composition and the profile for these components and processes. Vital requirements such as achieving robustness and high quality of the electrical connections and minimum feature sizes for the *MOS* devices are obtained by controlling the main etching characteristics of the process, such as the *etching rate, anisotropy, selectivity* and *uniformity.* For instance, high etching rates for acceptable throughput in single wafer tools as wafer sizes increase, anisotropic profiles, selectivity over thin gate oxides, uniformity and directionality of plasma processes can all be optimized for a particular application through appropriate selection of the chemical reactants and the operating conditions.

Better understanding of etching processes will allow improved control of the etching characteristics vital to the present and future productivity of the *VLSI* fabrication process. A few studies of polysilicon etching in CF_3Cl (Freon 13) have been published and interesting and encouraging results have been reported [10], [11], [12].

In [11] the plasma etching of heavily phosphorus doped polysilicon (an important gate and interconnect material for MOS devices) with the chlorofluorocarbon gas, CF_3Cl, is examined. The authors are interested in identifying and investigating the important physical processes responsible for polysilicon etching in CF_3Cl/Ar discharges. A method is presented by which polysilicon etching in a CF_3Cl plasma is

described successfully by a statistical model. The observed characteristics in terms of physical plasma properties inferred from an electrical impedance model are also interpreted. With this model physical properties of the plasma, including time-averaged bias voltage can be predicted. Increasing demands on etching processes for improved throughput, uniformity profiles and selectivity require better control of these etching characteristics. To place etching process design and control on a more rational basis, models for the etching process need to be formulated. Thus for polysilicon etching in CF_4 discharges predictive kinetic models have been formulated. In such models, a set of chemical reactions is proposed to simulate radical and ion formation and loss, transport and surface reaction. The steady-state solutions of the resulting ordinary differential equations are obtained. However, rate constants for most of the reactions in the model are not known and must be estimated. Consequently, these models necessarily contain at least 10 fitted or estimated parameters. The present inability of kinetic models to predict trends in plasma properties or etching characteristics is not surprising, given the present lack of knowledge on fundamental chemical and physical processes involved in plasma etching. In [11] the plasma processes modelling problem was divided into a series of more manageable tasks. First, each etching process "response" (etching rate or total ion flux) is characterized experimentally in terms of externally controlled variables, or "factors" (pressure or power input). Response surface methodology, an experimental design and data analysis strategy, is used to examine efficiently the etching response over wide factor ranges. The resulting data set is used with statistical analysis techniques to generate polynomial representations of etching process responses. The polynomial response models are valuable for process design in cases where predictive kinetic models are unavailable. Through an electrical impedance model for the plasma, the external variables are related to physically significant plasma properties (e.g., electron density). The etching responses are

then examined in light of the physical properties predicted by the impedance model. Response surface methodology is helpful in this endeavour, as polynomial response models are easily converted into contour plots of etching responses but not very accurately. In [11] expressions are derived for bulk plasma resistance, sheath capacitance, potential drop across the sheaths, bulk electric field, and electron density, using a simple electrical model of a radio-frequency (RF) discharge. Each of these quantities varies with temporal position in the RF cycle.

In [10] the authors present an experimental technique for measuring total positive ion bombardment fluxes and energy distribution. An empirical description using the mean ion energy provides a good representation of the ion energy distribution over a wide range of conditions. These results allow estimation of ion energy distribution in other parallel plate reactors. In [10] the ion bombardment of the surface during etching-ion energy is predicted using the plasma impedance results and a simple elastic collision model for ion transport through the plasma sheath is examined. The results obtained are crucial to the formulation of kinetic models of etching rate and directionality studied in [12]. In [12] the authors develop three kinetic models, based on experimentally and theoretically estimated plasma properties. These models include both chemical and ion-assisted processes, and therefore, predict etching directionality, as well as etching rate. The results indicate the importance of relatively low energy (<30eV) ion bombardment, and suggest that the dominant loss process for the chemical etchant is diffusion-limited recombination at the electrode surfaces.

The present understanding of plasma chemistry, and in particular, the surface processes involved in etching is limited. It is well known that energetic ion bombardment can have profound effects on thin film etching, but the details of the physical processes responsible for ion production and transport are not clear. Consequently, formulating a complex kinetic model for the etching process may not be fruitful at

present. A simple model, incorporating variables of known physical importance allows evaluation of the etching process at a level commensurate with the basic data available. In [3] process control strategies for reactive ion etching of silicon and silicon dioxide in CF_4/O_2 and CF_4/H_2 plasma are developed. The authors take into consideration four measured variables, four manipulated variables, and seven performance variables, and develop a MIMO model (four inputs, four outputs), which is very complex, difficult to manipulate and does not take into account the dynamics of the wafer characteristics with respect to the plasma parameters.

Relative Gain Array (RGA) analysis and *Singular Value Decomposition* (SVD) methods are used to select manipulated/process variable control loop pairings for feedback control and to evaluate potential difficulties in control system performance. However, these methods do not guarantee consistent results and deal only with steady-state representation of the system.

The development of real-time control techniques for improving the manufacturing characteristics of reactive ion etching process is well documented in [1]. The overall goal is to redesign the RIE machine for enhanced controllability and improved performance. To achieve this, the research is directed towards:

(i) control-oriented modelling and identification of the physico-chemical processes involved in RIE. A control-oriented model must capture the significant dynamics of the physical system and be suitable for control design.

(ii) analysis and improvement of the controllability and observability properties of RIE.

(iii) design and implementation of a hierarchical controller for the RIE.

The objective in [1] is to develop sufficiently general methods and results that allow implementation of real-time feedback control systems to a large class of RIE machines with a minimal amount of tuning. Based on a novel decomposition of the process, the

authors present a general strategy for the control of RIE. The principal idea is that by controlling appropriate plasma key parameters, it is possible to improve the etch performance of these machines. In [1] and [4]-[8], the *bias voltage*, V_{bias}, and *fluorine concentration* $[F]$ are used as the key plasma parameters to be controlled and *power* of RF generator and *throttle valve position* are selected as input variables. Based on the measured output data, and using standard identification algorithms, the authors in [1] have constructed a two-input, two-output model mapping small perturbations in *power* (Watts) and *throttle valve* (% opening) to the V_{bias} (Volts) and $[F]$ signals. The idea is very interesting and although the model is very simple and easy to manipulate, it may not be able to capture all the dynamics of the plasma. Specifically, the problem of representing the dynamics of the wafer still remains unanswered.

In [8] experimental results are presented on nonlinear models of the Hammerstein type for a reactive ion etcher, and a nonlinear tracking controller is implemented. This is motivated by the observation that the RIE exhibits significant nonlinear behaviour. In [8] a simple nonlinear model structure is used that is an input static nonlinear block (polynomial of second degree) in series with a linear time-invariant system. The model is improved compared to linear models since it takes into consideration the nonlinearity of the throttle valve actuator. However, the problem of capturing the dynamics of the wafer, and whether the model is capable of capturing all the dynamics of the plasma still needs to be investigated.

The experimental results demonstrate that the closed-loop predictions are very close to the observed data, and that the controller provides the capability of command following large excursions in the operating space. In [7] a dynamic model describing the relationship between the various physical variables of interest such as etch depth, etch rate and reflected light intensity is presented, and an extended Kalman filter is used to estimate the etch depth by processing reflectometry data for fast in-situ

etch rate measurements. This is motivated by the fact that in etching and deposition processes, the capability of measuring thin film thickness is very useful. Single or multiple wavelength reflectometers are commonly used for both *in-situ* and *ex-situ* measurements of film thickness. Techniques for in-situ real-time measurement of etch (or deposition) rate can be of great benefit in process development, on-line process diagnostic, and real-time feedback control. The fundamental idea here is to use nonlinear estimation theory, namely the Extended Kalman filter method for a simple dynamic model that combines the etching process with an optical model for reflection from a stack of parallel thin films. Problems that remain still open are *selectivity, uniformity and anisotropy* characterizations. A similar idea is developed in [6] where *etch rate* and *etch depth* are estimated from dual-wavelength reflectometry data. Using the estimated *etch rate*, an *etch rate* stabilization is achieved by using feedback control by actuating forward power in real-time. Using the estimated *etch depth* to trigger an end point, the authors demonstrate that it is possible to get an 83% final thickness variation over timed etches. The results depend very much on whether the model of the Plasma Generation Subsystem is accurate or not. To reduce the loading effect in RIE, [8] develops a real-time closed-loop control system. Recently, empirical RIE models based on neural networks [13], [14], [15] have been shown to exhibit superior performance in accuracy, predictive capability, and robustness over more traditional statistical approaches (regression methods). Neuromodel predictions of the RIE process outputs are used in conjunction with genetic algorithms and other optimization techniques to optimize the etching process performance as well as for recipe synthesize. More recently, adaptive learning techniques which utilise neural networks combined with statistical experimental design methods have been applied to semiconductor manufacturing [16], [14]. The neuromodels have shown good accuracy because these models fit well to experimental input-output data, and the nonlinearity

of the process is well reflected in the network. Algorithms based on adaptive learning techniques were developed that enable these networks to adapt through a *trial-and-error* procedure. In this adaptive process, the connection strengths between the active elements of the network are gradually modified until the network exhibits a desired behaviour. Several other neural network based plasma etch models have also appeared in the literature [13], [17], [18], [15]. Some recent research is focused on developing more specific and advanced structures and algorithms, such as exponentially weighted moving average (*EWMA*) neurocontrollers [19]. These networks are integrated in the overall intelligent control structures of the plasma etching processes to maintain process targets over extended periods for improved product quality and decreased machine downtime [19], [14]. Proper choice of neurocontroller parameters (weights) is critical to the performance of these systems.

At the present time RIE's are typically supplied with a *PID controller* for regulating the chamber pressure. In addition, several of the actuators, such as the mass flowmeters and the RF power generation unit, have internal controllers in order to make them less sensitive to variations in operating environment. These actuators influence the actual etch process by affecting the plasma characteristics. Currently, the only plasma property which is stabilized by using feedback control is the pressure. It is commonly acknowledged that the RIE process is not very robust and requires frequent tuning to achieve acceptable yields. The research described in this thesis is directed toward the application of a real-time control systems theory to RIE. The main thrusts of this research are as follows:

(*i*) control-oriented modelling and identification of the physico-chemical processes involved in the RIE process (linear ARIMA models and neuromodels), and

(*ii*) controller design for the RIE using real-time multivariable control strategies (minimum variance stochastic adaptive control, robust control, self-tuning control

and neurocontrol).

From a control engineering viewpoint, the RIE process represents an interesting challenge in several different ways. The key issue is the fact that many of the crucial etch parameters that need to be controlled cannot, at present time, be measured in real-time. This necessitates indirect control strategies wherein plasma parameters are used for feedback to achieve tight control of the etch characteristics.

1.1 Summary

The objective of this thesis is to develop MIMO real-time control strategies for improving plasma characteristics in the reactive ion etching process.

Vital requirements for achieving robustness and high quality of the electrical connections, uniformity profiles and minimum feature sizes for MOS devices require better control of the main etching characteristics of the process, namely *etching rate, anisotropy, selectivity and uniformity.* The priority in our research is focused more upon practical aspects in order to implement the most appropriate conventional and nonconventional control strategies that are needed for addressing some of the difficulties that the plasma system community currently faces.

This goal is motivated by the following main factors:

(*i*) Presently most semiconductor manufacturing equipment is designed to be operated in an open-loop mode and consequently the manufacturing performance of this equipment is not as good as desired.

(*ii*) The reactive ion etching process is a critical technology for modern *VLSI* circuit fabrication at various manufacturing process stages.

(*iii*) High complexities such as nonlinearities, a MIMO process, and operating at low

pressure, have made the reactive plasma systems as interesting research topic in plasma community. In the last decade this has captured a lot of attention of the specialists from systems and control field to solve practical aspects of the control problems of these systems in closed-loop.

(*iv*) This field of research offers an excellent opportunity to researchers and to specialists from the control area to apply dynamic systems analysis, synthesis, and control methods to study reactive ion etching systems.

(*v*) By this research we could partially fill the void that exists due to the lack of feedback control in these systems. This is presently considered as one of the main problems facing the semiconductor manufacturing industry.

(*vi*) Our results could also be extended and applied to other industrial applications such as chemical vapour deposition (CVD), anisotropic wet etching silicon ($TMAH$), plasma enhanced chemical vapour deposition ($PECVD$), low plasma chemical vapour deposition ($LPCVD$), etc.

The research is based on the main idea that by controlling appropriate plasma parameters, namely the concentration of the fluorine reactive radicals and ions $[F]$, and theirs energy reflected in V_{bias}, one could ensure satisfactory tracking for the etching rate or equivalently etch depth (being one of the most important etching characteristics).

Specifically, in this thesis we will present some results concerning the models of the PGS, namely the *coupled* and *decoupled cases* with or without *delay* (Chapter 3), the design of real-time multivariable feedback conventional and nonconventional control strategies, namely *PI self-tuning* adaptive controller based on minimization of a quadratic cost function, a *Linear Quadratic Gaussian scheme* (LQG), minimum variance stochastic adaptive control, robust control, and neurocontrol (Chapters 4-7).

In the last chapter (Chapter 7) we will compare the performances of all the algorithms developed in this thesis and we will select the best structure for a real-time feedback control strategy using a hierarchical control approach. The hierarchical control strategy that is capable of determining at each sampling interval which controller is the most suitable choice to be employed will be of great benefit. In order to have a criterion for comparing the performances of different control strategies, we will translate the results obtained in Chapters 4-6 for the PGS into performances of the wafer subsystem, namely *etch rate* or *etch depth*. The main objective of our proposed control strategies is to keep the etch depth at a desired target value in the face of variations in the process disturbances, namely CF_4 flow-rate disturbances, loading disturbances, and oxygen disturbances. The results obtained confirm the utility of our proposed strategies to control the etching process despite the drift during an etch. However, these algorithms can also be used as a research tool for evaluating complicated gas kinetics. The controllers can drive the film properties, namely *etch depth*, *anisotropy*, *selectivity* and *uniformity* to desired regimes based on important gas species and dc induced voltage in an optimal fashion.

Chapter 2

Overview of RIE: Equipment and System

In this chapter a brief description of the RIE equipment is provided. This description is given from a control system perspective. The emphasis is on the overall system behaviour rather than on an individual physical/chemical processes. It is well known in the plasma literature that the RIE process is *highly nonlinear* and *multivariable* [1]. Existing plasma systems attempt to control the important wafer etch characteristics with the input variables *pressure*, applied *RF power*, and *gas flow rates*. However, there is no standard and known way to use these inputs to predict the etch performances in different machines or in identical machines, or even in the same machine on two different runs [1]. This is due to the variations in plasma properties and disturbances and the fact that there is a significant amount of uncertainty in the open-loop system. This is the main reason why we believe that the five real-time feedback control strategies developed in this thesis will be of great potential benefit to the control of RIE process.

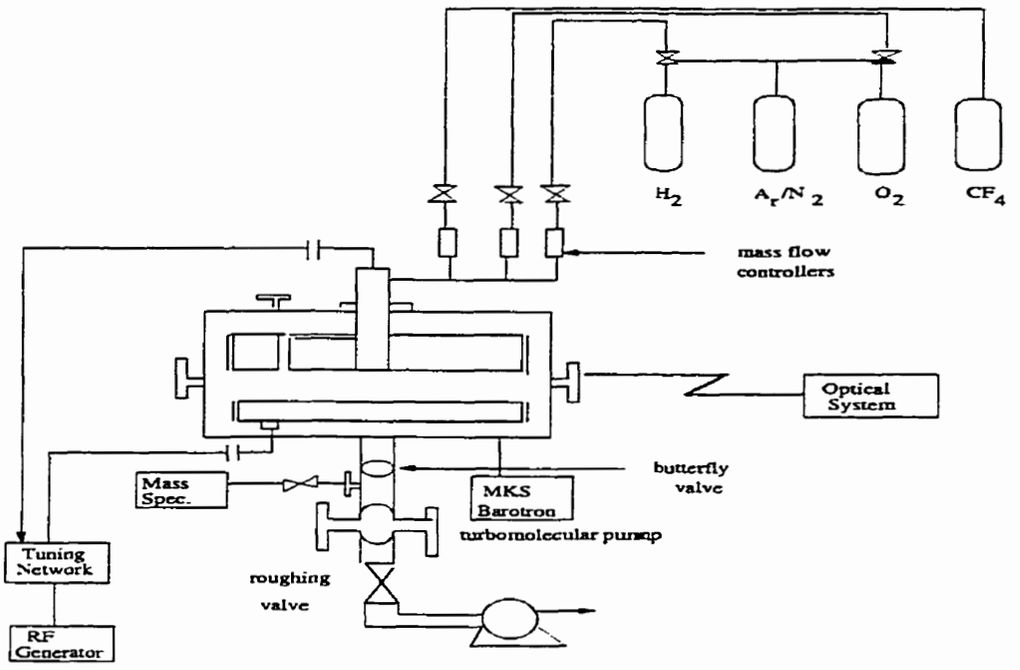


Figure 2.1: Plasma reactor etching system

2.1 The Reactor

The plasma reactor that is used in this thesis is an Applied Materials 8300 Hexode Reactive Ion Etcher used at the Control Systems Laboratory of the University of Michigan, from where experimental input-output data set for different operating points was obtained. This reactor is equipped with a data acquisition system, actuators and sensors appropriate for real-time feedback control. The configuration of parallel plate Plasma Reactor Etching System type 1000TP is presented in Figure 2.1. A simplified block representation of this configuration is shown in Figure 2.2.

The parallel plate system has a bottom electrode with a diameter of 29.2 cm, 34.3 cm top electrode, and variable electrode spacing set by the user at 3.81 cm. The power source is a 600 W, 13.56 MHz *RF power* generator capacitively coupled to each electrode through an automatic tuning network impedance matching system. Each electrode is equipped with an *RF* plasma dark space shield to prevent parasitic

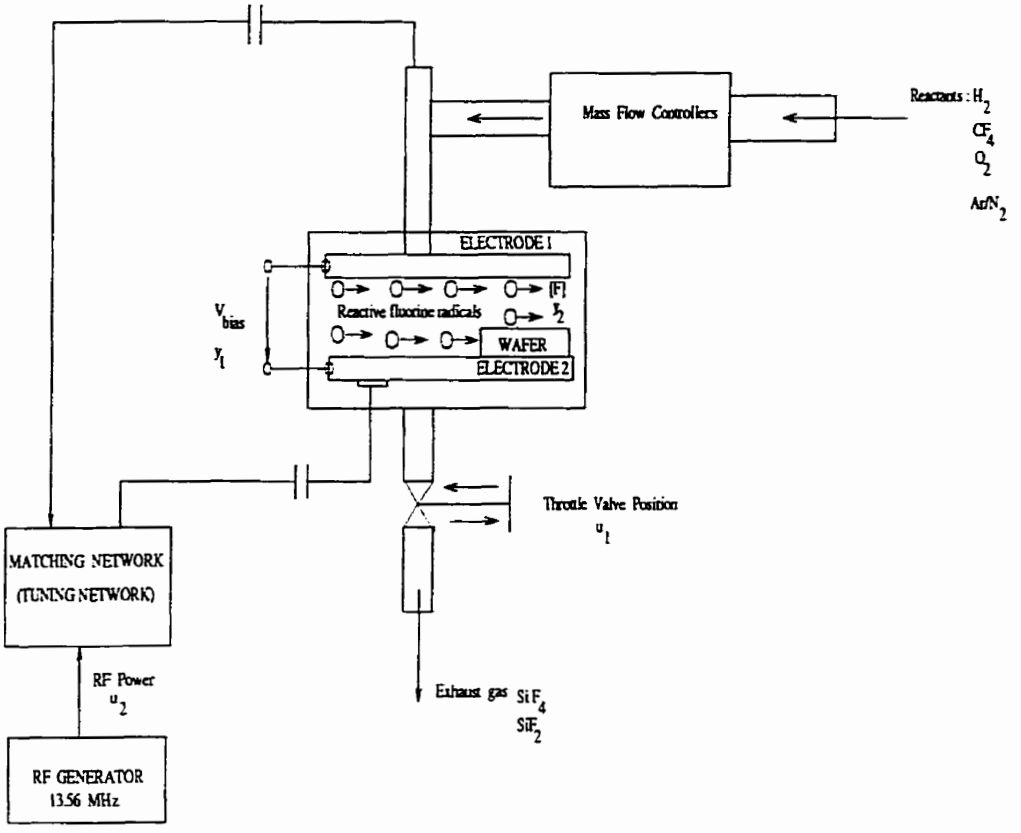


Figure 2.2: Plasma reactor etching system - simplified block diagram.

plasma from forming outside of the shields.

2.2 Actuators and Sensors

To regulate the exhaust of reactant gases from the chamber, the reactor is equipped with a throttle valve. For improved control capability this valve was sized to be small, thus moving its operating region away from the saturation with a low leakage conductance when fully closed for a rapid response time for the measurement of the actual valve position. We will consider in the thesis two cases: the first case where the valve is a linear element and the second case, which is closer to reality, where the valve is a nonlinear element with the nonlinearity given by a second order polynomial

[8] or of a hysteresis type [1].

The throttle valve controller allows remote control of the valve. The *RF power* actuator includes a 13.56 MHz generation unit and a matching network. In order to increase the range of flows that can be achieved, an additional mass flow controller has been added. The bias voltage is measured through an inductive tap into powered electrode. *Pressure* in the chamber is monitored by a Baraton Capacitance Manometer which is sensitive to pressures between 1 mTorr and 100 mTorr, where 20 mTorr is a typical operating point for the RIE process in the above system. The *fluorine concentration* is measured via optical emission spectroscopy using actinometry, with argon as the calibration species. Light from the plasma glow is modulated to approximately 2KHz by a mechanical chopper, then passed through a pair of monochromators which select specific wavelengths in the fluorine and argon spectra [1]. The light signals are then converted to electrical signals by photomultiplier tubes and demodulated via lock-in amplifier.

2.3 RIE System: Decomposition and Control

Motivated by the ideas developed in [1], a real-time control system for improving the manufacturing characteristics of reactive ion etchers is developed in [4]. In this work, the RIE system is decomposed into two functional blocks:

(i) the plasma generation subsystem (PGS), and

(ii) the wafer etch subsystem (WES).

The above sequential processes separate the generation of the important chemical and physical species from the action of etching the surface of the wafer. The inputs to the plasma generation subsystem are:

(i) the *throttle position*, and

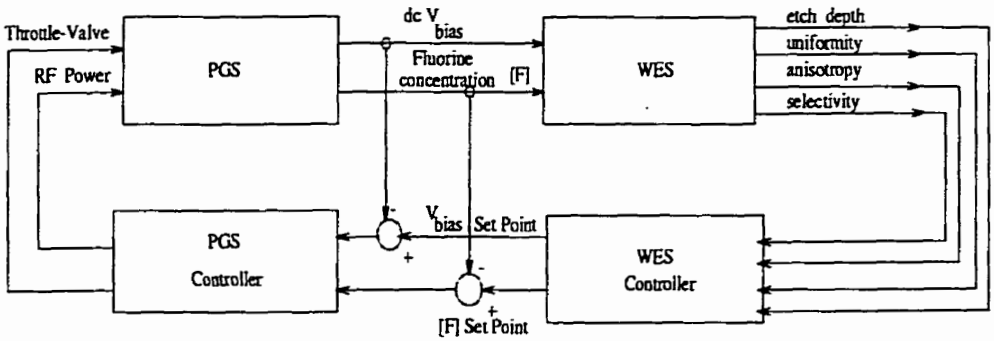


Figure 2.3: The decomposed structure of the **RIE** system

(*ii*) the *applied RF power*

The outputs of the plasma generation subsystem are the key plasma parameters responsible for etching, namely

(*i*) the V_{bias}, and

(*ii*) the *fluorine concentration* $[F]$,

where V_{bias} represents an estimate of the mechanical energy of the impinging ions and $[F]$ represents the concentration of the chemical compounds (*fluorine*) involved in etching.

The WES is driven by the key plasma parameters and its outputs represent quantities crucial to the etch performance. The above decomposition actually represents a physical separation, that is

(*i*) the PGS represents the bulk plasma, and

(*ii*) the WES represents the wafer surface phenomena, and the interface is the boundary layer.

The decomposition suggests a suitable control structure for the RIE system as shown in Figure 2.3. While this decomposition is based on sound physical principles, it is not completely accurate as there is a certain amount of feedback coupling from the wafer surface reactions to the plasma.

The key idea is to regulate the inputs to the WES by precisely controlling the outputs of the PGS. The RIE system is susceptible to disturbances which increase the variance of the output characteristics regardless of the operating point selected. Disturbances that affect the RIE process include:

(*i*) the *load*

(*ii*) the chamber seasoning

(*iii*) the variation in the *RF power* generator and matching network

(*iv*) the variation in the mass flow controllers, and

(*v*) the water vapour and other contaminants.

Here, *load* refers to the total surface area of material to be etched, which is a function of the number of wafers and the pattern on each wafer. In that work a multivariable feedback control is designed which is capable of compensating these disturbances and thereby reducing the process variance. By applying this form of multivariable feedback control to RIE, the effect of the variations in the load on the rate is dramatically reduced. Typically, when the load is increased the *etch rate* decreases. This phenomenon is commonly known as the *loading effect*. The *loading effect* is known to be significant in certain etch chemistries such as fluorine etching of silicon. The *loading effect* is undesirable because it is a source of variance in *etch rate*. Because of the variance in the *etch rate*, selection of the appropriate etch time is difficult. Selection of the proper etch time becomes even more difficult when considering that the *etch rate* also changes during an etch due to *the loading effect*. Because of non uniformity, wafers are typically overetched to ensure that all material is removed. As the end of the etch approaches, less material is left and the *etch rate* increases. The side walls become the focus of the etchant and any additional etch time contributes at a greater rate to the undercut, one of the wafer qualities of interest. Consequently, the process yield may be reduced and costly trial runs are

necessary to establish the proper etch time for new loads. The key idea is to regulate the inputs to the WES by precisely controlling the outputs of the PGS. What remains to be investigated is the problem of determining a supervisory controller to perform the high-level cell control function that includes on line monitoring and diagnostics as well as postprocessing analysis of the data for the purposes of sequential optimization, quality control, etc. To solve this problem it is necessary to develop a controller around WES which will translate desired etch characteristics into set points for the PGS control system.

With the existing sensor technology, it is very difficult to measure the key wafer etch parameters, namely *selectivity*, *anisotropy*, etc., in real time during the etch process. Therefore, for real time feedback control, an indirect strategy is necessary. The decomposition of the etching process is very important because the modeling task for the WES would involve relating the effects of the key plasma parameters to the etch performance. This is much more direct than trying to build a single model from the equipment inputs to the etch characteristics. The switch from specifying the process parameters in terms of (*power*, *throttle valve*) to (V_{bias}, $[F]$) is a significant change of viewpoint. The new set points are in many ways more directly related to the overall etch performance. In other words tightly regulating them should eliminate much of the variances seen in the plasma systems. As stated above, it will be necessary to relate desired *etch rate*, *selectivity*, *anisotropy*, *uniformity* to the key plasma parameters. This is the purpose of the WES controller. This controller would also operate in real-time, but perhaps would operate on a slower time scale than the plasma controller. The sampling rate for the Plasma Generation Subsystem is 20 Hz. The reactor is also instrumented with an in-situ spectral reflectometry system, which samples 798 wavelengths between 400 and 800 nm every 0.5 s for 2 Hz sampling rate, resulting in the PGS being 10 times faster than WES. Presently, the details of this second

level WES controller are not quite clear, because real-time sensors for the wafer etch parameters are not readily available and the models for relating plasma parameters to the wafer etch parameters are not well understood. As a matter of fact, it is not clear whether constructing empirical small perturbation models for the wafer etch subsystem is appropriate. In [1] a primitive supervisory platform is developed and is tested on the RIE system. This high level system emphasizes a generic central control mechanism, a sequential statistical process optimization algorithm, standardized communications, and extensive *in-situ* monitoring and diagnostics capabilities.

2.4 Summary

In this chapter, we have presented a comprehensive approach to the real-time control of an RIE process. This is based on the different conceptual decompositions of the RIE system into plasma and etch subsystems. Also, we have presented the main results from the literature concerning modelling and multivariable real-time feedback control strategies of RIE system. The experimental results on system identification and real-time control offer encouraging support that controlling certain plasma parameters results in improved etch performance, namely *etch depth*, *anisotropy*, *uniformity* and *selectivity*. The results demonstrate conclusively that the introduction of real-time feedback control leads to a much more stable *etch rate* as compared to the standard industrial practice of setting *pressure*, *flow*, and *RF power*. The significant gains may be obtained by using techniques from control and systems theory in semiconductor manufacturing. At this time, we are able to perform a more complete investigation to understand the appropriate plasma parameters whose control will lead to improved etch performance.

Chapter 3

Modeling and Validation

3.1 Linear Plasma Generation Subsystem Models

3.1.1 System Identification

In order to build models for the Plasma Generation Subsystem (PGS), an experimental identification approach is needed. Generally, a region of operating points is delineated in the space of *pressure*, $[CF_4]$ *flow rate*, and power corresponding to the RIE region of the plasma parameters space. Specifically power ranges from 900 to 1300 Watts, pressure ranges from 20-30 mTorr, and flow rate ranges from 20-40 standard conventional cubic meters (sccm) [1]. Typical values of 1000 Watts, 20 mTorr, and 30 sccm are selected as a nominal operating point. The identification experiments involve application of step increments to the power and throttle inputs from their nominal values. The experiments are done without a wafer in the chamber, because the modifications to the system identification with a wafer being etched are quite straightforward.

In the PGS, the control inputs are *RF power*, *throttle position*, and CF_4/Ar *flow*. The disturbances are *load* and *water vapours*. The state of the plasma system are

the *fluorine concentration* $[F]$, and *dc bias voltage*, V_{bias}. *Fluorine* is the dominant etchant species and V_{bias} is used as a measure of the physical energy of the *impinging ions*. Multivariable feedback control strategies reported in the literature [10], [11], [12] use $\%H_2/\%O_2$, *pressure, RF power, flow rate* to control V_{bias}, $[CF_2]$, $[F]$, $[H]/[CO_x]$, but the models are more complex and very difficult to control, due to poorly conditioned systems with strong interactions between the input and the output variables. The models in [1], [20], [4]-[8], [21] are simpler since they use only two independent input variables, namely *RF power* and *throttle valve* to control two independent output variables, namely V_{bias} and $[F]$. We also choose these variables since the plasma variables and $[F]$ are more directly related to the *etch rate* and other output characteristics when compared to the *RF power* and *pressure*, which are held constant conventionally. The disturbances mostly affect the PGS and not WES, so by controlling the plasma variables, the effects of these disturbances can be mitigated. To design a real-time control strategy first a linear model of the plant is obtained about an operating point.

3.1.1.1 Linear ARMA Models

Using the available experimental input-output data set a multivariable Auto Regressive Moving Average ($ARMA$) model of second-degree is considered

$$A(q)y(k) = B(q)u(k) + C(q)e(k) \tag{3.1}$$

where

$$y(k) = \begin{bmatrix} y_1(k) \\ y_2(k) \end{bmatrix} = \begin{bmatrix} V_{bias} \\ F \end{bmatrix}, \; u(k) = \begin{bmatrix} u_1(k) \\ u_2(k) \end{bmatrix} = \begin{bmatrix} throttle\ position \\ RF\ power \end{bmatrix}$$

$$e(k) = \begin{bmatrix} e_1(k) \\ e_2(k) \end{bmatrix}, \; C(q)e(k) = N(k) = \begin{bmatrix} N_1(k) \\ N_2(k) \end{bmatrix}$$ represent the white noise and

the output of the coloured noise filter, respectively, and

$A(q) = I_2q^2 + A_1q + A_2$, $B(q) = B_0q^2 + B_1q + B_2$, $C(q) = C_0q + C_1$,

I_2 is an identity matrix, $A_1 = \begin{pmatrix} a_{11} & 0 \\ 0 & a_{21} \end{pmatrix} = diag(a_{11}, a_{21})$, $A_2 = diag(a_{12}, a_{22})$

$B_0 = \begin{pmatrix} b_{01} & b_{02} \\ b_{03} & b_{04} \end{pmatrix}$, $B_1 = \begin{pmatrix} b_{11} & b_{12} \\ b_{13} & b_{14} \end{pmatrix}$, $B_2 = \begin{pmatrix} b_{21} & b_{22} \\ b_{23} & b_{24} \end{pmatrix}$

$C_0 = \begin{pmatrix} c_{01} & c_{02} \\ c_{03} & c_{04} \end{pmatrix}$, $C_1 = \begin{pmatrix} c_{11} & c_{12} \\ c_{13} & c_{14} \end{pmatrix}$.

The model (3.1) can be expressed in one of the following four forms:

(*i*) *Coupled case with pure delay*:

This is obtained from the general structure (3.1) by setting

$B_0 = O_{2x2}$ (defined as a 2x2 dimensional matrix with all its elements equal to zero)

(***ii***) *Coupled case without delay*:

This is obtained from the general structure (3.1) by setting

$B_2 = O_{2x2}$

(***iii***) *Decoupled case with pure delay*:

This is obtained from the general structure (3.1) by setting

$B_0 = O_{2x2}$, $B_1 = diag(b_{11}, b_{14})$, $B_2 = (b_{21}, b_{24})$, $C_0 = diag(c_{01}, c_{04})$

$C_1 = diag(c_{11}, c_{14})$, where B_1, B_2, C_0, C_1 represent diagonal 2x2 matrices.

(***iv***) *Decoupled case without delay*:

This is obtained from the general structure (3.1) by setting

$B_0 = diag(b_{01}, b_{04})$, $B_1 = diag(b_{11}, b_{14})$, $B_2 = O_{2x2}$, $C_0 = diag(c_{01}, c_{04})$

$C_1 = diag(c_{11}, c_{14})$

In the above two decoupled cases we can consider the multivariable system as two independent *SISO* loops which may be expressed in the following two forms:

(i) *Decoupled case with pure delay*:

$$Y_1(z) = z^{-1}\frac{\tilde{B}_1(z)}{\tilde{A}_1(z)}U_1(z) + \frac{\tilde{C}_1(z)}{\tilde{A}_1(z)}E_1(z) = z^{-1}\frac{\tilde{B}_1(z)}{\tilde{A}_1(z)}U_1(z) + N_1(z) \qquad (3.2)$$

$$Y_2(z) = z^{-1}\frac{\tilde{B}_2(z)}{\tilde{A}_2(z)}U_2(z) + \frac{\tilde{C}_2(z)}{\tilde{A}_2(z)}E_2(z) = z^{-1}\frac{\tilde{B}_2(z)}{\tilde{A}_2(z)}U_2(z) + N_2(z) \qquad (3.3)$$

Based on the available experimental input-output data set for the particular case of uncoloured noise we get:

$\tilde{B}_1(z)$ = 4.1996z -3.9567, $\tilde{B}_2(z)$ = 0.0001z -0.0001,

$\tilde{A}_1(z) = z^2$ -0.5016z -0.4906, $\tilde{A}_2(z) = z^2$ -0.9482z -0.0433.

(ii) *Decoupled case without delay*:

$$Y_1(z) = \frac{\tilde{B}_1(z)}{\tilde{A}_1(z)}U_1(z) + \frac{\tilde{C}_1(z)}{\tilde{A}_1(z)}E_1(z) = \frac{\tilde{B}_1(z)}{\tilde{A}_1(z)}U_1(z) + N_1(z) \qquad (3.4)$$

$$Y_2(z) = \frac{\tilde{B}_2(z)}{\tilde{A}_2(z)}U_2(z) + \frac{\tilde{C}_2(z)}{\tilde{A}_2(z)}E_2(z) = \frac{\tilde{B}_2(z)}{\tilde{A}_2(z)}U_2(z) + N_2(z) \qquad (3.5)$$

Based on the same experimental input-output data set for the particular case of uncoloured noise we get:

$\tilde{B}_1(z)$ = -2.6879z +2.939, $\tilde{B}_2(z)$ = -0.00001z +0.00002,

$\tilde{A}_1(z) = z^2$ -0.4978z -0.4941, $\tilde{A}_2(z) = z^2$ -0.9455z -0.0449.

Similarly, the coupled PGS models of (3.1) may be expressed in the following two forms:

(i) *Coupled case with pure delay*:

$$Y_1(z) = z^{-1}\frac{\tilde{B}_1(z)}{\tilde{A}_1(z)}U_1(z) + z^{-1}\frac{\tilde{D}_1(z)}{\tilde{A}_1(z)}U_2(z) + \frac{\tilde{C}_1(z)}{\tilde{A}_1(z)}E_1(z) \qquad (3.6)$$

$$Y_2(z) = z^{-1}\frac{\tilde{B}_2(z)}{\tilde{A}_2(z)}U_2(z) + z^{-1}\frac{\tilde{D}_2(z)}{\tilde{A}_2(z)}U_2(z) + \frac{\tilde{C}_2(z)}{\tilde{A}_2(z)}E_2(z) \qquad (3.7)$$

Using the same experimental input-output data set for the particular case of uncoloured noise we get:

$\tilde{B}_1(z) = 4.1996z$ -3.9568, $\tilde{B}_2(z) = 0.0008z$ -0.005, $\tilde{D}_1(z) = 0.0837z$-0.0569,

$\tilde{D}_2(z) = 0.0001z$, $\tilde{A}_1(z) = z^2$ -0.3714z -0.4942, $\tilde{A}_2(z) = z^2$ -0.853z -0.0922.

(ii) *Coupled case without delay*:

$$Y_1(z) = \frac{\tilde{B}_1(z)}{\tilde{A}_1(z)}U_1(z) + \frac{\tilde{D}_1(z)}{\tilde{A}_1(z)}U_2(z) + \frac{\tilde{C}_1(z)}{\tilde{A}_1(z)}E_1(z) \tag{3.8}$$

$$Y_2(z) = \frac{\tilde{B}_2(z)}{\tilde{A}_2(z)}U_1(z) + \frac{\tilde{D}_2(z)}{\tilde{A}_2(z)}U_2(z) + \frac{\tilde{C}_2(z)}{\tilde{A}_2(z)}E_2(z) \tag{3.9}$$

From the experimental input-output data setup for the particular case of uncoloured noise we get:

$\tilde{B}_1(z) =$ -1.0683z +3.1912, $\tilde{B}_2(z) =$ -0.0059z +0.0015, $\tilde{D}_1(z) = 0.1703z$-0.1382,

$\tilde{D}_2(z) = 0.0001z$. $\tilde{A}_1(z) = z^2$ -0.4571z -0.3846, $\tilde{A}_2(z) = z^2$ -0.8511z -0.0922.

For the general case these polynomials could be expressed as following

$$\tilde{A}_1(z) = z^2 + \tilde{a}_{11}z + \tilde{a}_{12}$$

$$\tilde{A}_2(z) = z^2 + \tilde{a}_{21}z + \tilde{a}_{22}$$

$$\tilde{B}_1(z) = \begin{cases} \tilde{b}_{01}z + \tilde{b}_{11} & \textit{without delay} \\ \tilde{b}_{11}z + \tilde{b}_{21} & \textit{with pure delay} \end{cases}$$

$$\tilde{B}_2(z) = \begin{cases} \tilde{b}_{03}z + \tilde{b}_{13} & \textit{without delay} \\ \tilde{b}_{13}z + \tilde{b}_{23} & \textit{with pure delay} \end{cases}$$

$$\tilde{C}_1(z) = \tilde{c}_{01}z + \tilde{c}_{11}$$

$$\tilde{C}_2(z) = \tilde{c}_{04}z + \tilde{c}_{14}$$

$$\tilde{D}_1(z) = \begin{cases} \tilde{d}_{01}z + \tilde{d}_{11} & \textit{without delay} \\ \tilde{d}_{11}z + \tilde{d}_{21} & \textit{with pure delay} \end{cases}$$

$$\tilde{D}_2(z) = \begin{cases} \tilde{d}_{03}z + \tilde{d}_{13} & \textit{without delay} \\ \tilde{d}_{13}z + \tilde{d}_{23} & \textit{with pure delay} \end{cases}$$

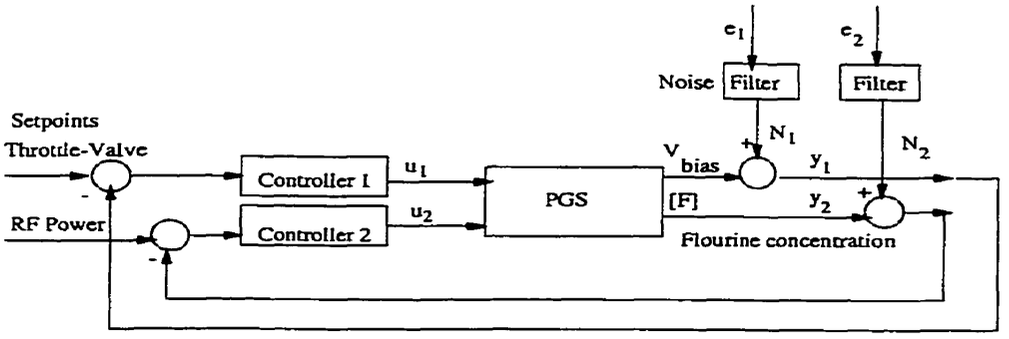


Figure 3.1: System representation in *ARIMA* (Auto Regressive Integrated Moving Average) model with stochastic disturbances

The schematic of the overall system considered can be represented as shown in Figure 3.1.

An adaptive control law based on this model will not have an integral action because of the stationary character of the disturbances. To design a control law with integral action that is capable of dealing with nonstationary disturbances, a disturbance process with stationary increments is postulated [22]:

$$\zeta_1(z) = \frac{\tilde{C}_1(z)}{(z-1)\tilde{A}_1(z)} E_1(z) \tag{3.10}$$

$$\zeta_2(z) = \frac{\tilde{C}_2(z)}{(z-1)\tilde{A}_2(z)} E_2(z) \tag{3.11}$$

or equivalently:

$$\Delta\zeta_1(z) = N_1(z)$$

$$\Delta\zeta_2(z) = N_2(z)$$

where $\Delta = z - 1$ is the first-order difference operator. In this case we get the general representation of the process by the *ARIMA* (Auto Regressive Integrated Moving Average) model integrated in our proposed *MIMO* real-time control strategies that are developed later in Chapters 4-7:

(i) Coupled ARIMA model without delay

$$\tilde{A}_1(z)\Delta Y_1(z) = \tilde{B}_1(z)\Delta U_1(z) + \tilde{D}_1(z)\Delta U_2(z) + \tilde{C}_1(z)E_1(z) \tag{3.12}$$

$$\tilde{A}_2(z)\Delta Y_2(z) = \tilde{B}_2(z)\Delta U_1(z) + \tilde{D}_2(z)\Delta U_2(z) + \tilde{C}_2(z)E_2(z) \tag{3.13}$$

(ii) Coupled ARIMA model with pure delay

$$\tilde{A}_1(z)\Delta Y_1(z) = z^{-1}\tilde{B}_1(z)\Delta U_1(z) + z^{-1}\tilde{D}_1(z)\Delta U_2(z) + \tilde{C}_1(z)E_1(z) \tag{3.14}$$

$$\tilde{A}_2(z)\Delta Y_2(z) = z^{-1}\tilde{B}_2(z)\Delta U_1(z) + z^{-1}\tilde{D}_2(z)\Delta U_2(z) + \tilde{C}_2(z)E_2(z) \tag{3.15}$$

(iii) Decoupled ARIMA model without delay

$$\tilde{A}_1(z)\Delta Y_1(z) = \tilde{B}_1(z)\Delta U_1(z) + \tilde{C}_1(z)E_1(z) \tag{3.16}$$

$$\tilde{A}_2(z)\Delta Y_2(z) = \tilde{B}_2(z)\Delta U_2(z) + \tilde{C}_2(z)E_2(z) \tag{3.17}$$

(iv) Decoupled ARIMA model with pure delay

$$\tilde{A}_1(z)\Delta Y_1(z) = z^{-1}\tilde{B}_1(z)\Delta U_1(z) + \tilde{C}_1(z)E_1(z) \tag{3.18}$$

$$\tilde{A}_2(z)\Delta Y_2(z) = z^{-1}\tilde{B}_2(z)\Delta U_2(z) + \tilde{C}_2(z)E_2(z) \tag{3.19}$$

where $\Delta Y(z) = (z-1)Y(z)$, and $\Delta U(z) = (z-1)U(z)$

Although the form of the above models are linear, the model coefficients are time varying and are to be estimated in real-time. These models are preferred for adaptive control because they lead to algorithms that can be easily implemented on a digital computer. The above models are used in Chapter 3 to implement multivariable adaptive control strategies based on the minimization of a quadratic cost function,

namely *PI self-tuning* adaptive control.

3.1.1.2 Transfer Matrix Representation

To measure the interactions between the input and output system variables to determine suitable pairs among them, *Relative Gain Array* (*RGA*) and *Singular Value Decomposition* (*SVD*) techniques may be used. However, they would require a transfer matrix representation of the *ARMA* models. Based on the measured input-output data, and using the standard identification for our system, i.e., the PGS system under investigation, we obtain the following transfer matrices for the *coupled case with pure delay*:

$$H(z) = \begin{bmatrix} \frac{4.1996z-3.9568}{z^2-0.3714z-0.4942} & \frac{0.0837z-0.0569}{z^2-0.3714z-0.4942} \\ \frac{0.0008z-0.005}{z^2-0.853z-0.0922} & \frac{0.0001z}{z^2-0.853z-0.0922} \end{bmatrix}. \tag{3.20}$$

and for the *coupled case without delay*:

$$H(z) = \begin{bmatrix} \frac{-1.0683z^2+3.1912z}{z^2-0.4571z-0.3846} & \frac{0.1703z^2-0.1382z}{z^2-0.4571z-0.3846} \\ \frac{-0.0059z^2+0.0015z}{z^2-0.8511z-0.0922} & \frac{0.0001z^2}{z^2-0.8511z-0.0922} \end{bmatrix} \tag{3.21}$$

where these matrices are calculated directly from the equations (3.6)-(3.9):

$$H(z) = \begin{bmatrix} \frac{\tilde{B}_1(z)}{\tilde{A}_1(z)} & \frac{\tilde{D}_1(z)}{\tilde{A}_1(z)} \\ \frac{\tilde{B}_2(z)}{\tilde{A}_2(z)} & \frac{\tilde{D}_2(z)}{\tilde{A}_2(z)} \end{bmatrix}$$

for the coupled $ARMA$ models without delay, and

$$H(z) = \begin{bmatrix} \frac{\tilde{B}_1(z)}{z\tilde{A}_1(z)} & \frac{\tilde{D}_1(z)}{z\tilde{A}_1(z)} \\ \frac{\tilde{B}_2(z)}{z\tilde{A}_2(z)} & \frac{\tilde{D}_2(z)}{z\tilde{A}_2(z)} \end{bmatrix}$$

for the coupled $ARMA$ models with pure delay.

The coefficients of the polynomials *ARMA* models $\tilde{A}_1$, $\tilde{A}_2$, $\tilde{B}_1$, $\tilde{B}_2$, $\tilde{D}_1$, *and* $\tilde{D}_2$ are computed by the algorithm during the identification process based on the available experimental input-output data set. The above transfer matrix representations of the *ARMA* models will be used in Chapter 6 to implement a *MIMO* real-time robust control strategy using standard H^∞ control techniques.

3.1.1.3 Linear State-Space Representation

Alternatively, a discrete-time state-space representation of the PGS may be expressed as

$$x(k+1) = A_G x(k) + B_G u(k) + w(k) \tag{3.22}$$

$$y(k) = C_G x(k) + D_G u(k) + v(k) \tag{3.23}$$

where
$x(k) = \begin{bmatrix} x_1(k) & x_2(k) & x_3(k) & x_4(k) \end{bmatrix}^T$, $u(k) = \begin{bmatrix} u_1(k) & u_2(k) \end{bmatrix}^T$, and
$y(k) = \begin{bmatrix} y_1(k) & y_2(k) \end{bmatrix}^T$ represent the state vector, input vector, and the output vector of the PGS system, respectively. For this representation we set $x_2 = y_1$ and $x_4 = y_2$, to give some physical significances to the x_i's, i=1,..., 4. Furthermore, $w(k)$, $v(k)$ are uncorrelated (for *Linear Quadratic Gaussian* control algorithm developed in Section 4.2) or correlated (for stochastic adaptive control algorithm developed in Section 4.3) zero-mean Gaussian random vectors representing the effects of process fluctuations and measurement noise, respectively, i.e., $E(w) = E(v)$=0, $E(ww^T) = Q_w$ (covariance matrix of $w(k)$, a symmetric positive definite matrix), $E(vv^T) = R_v$ (covariance matrix of $v(k)$, a symmetric positive definite matrix), and $E(wv^T) = R_{wv}$ (cross-correlation matrix, and R_{wv} =0 for w and v uncorrelated). Based on the available experimental input-output data set the

following values for the matrices are obtained:

$$A_G = \begin{bmatrix} 0 & 0.3846 & 0 & 0 \\ 1 & 0.4571 & 0 & 0 \\ 0 & 0 & 0 & 0.0927 \\ 0 & 0 & 1 & 0.8511 \end{bmatrix}, B_G = \begin{bmatrix} 0.4108 & 0.0654 \\ 2.7029 & -0.0604 \\ -0.0005 & 0 \\ -0.0035 & 0.0001 \end{bmatrix},$$

$$C_G = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, D_G = \begin{bmatrix} -1.0683 & 0.175 \\ -0.0059 & 0.00035 \end{bmatrix}.$$

The above representation is used in Chapters 4-7 to implement multivariable adaptive control strategies based on the minimization of a quadratic cost function, *Linear Quadratique Gaussian (LQG)*-scheme, stochastic minimum variance adaptive control (*SMVAC*), intelligent and robust control.

3.1.2 Simulations and Model Validation

In order to determine if the models represent a good approximation to the physical system, experimental tests are performed. The models were identified by exciting both system actuators together, i.e., varying *throttle valve* opening and *RF power* simultaneously. In principle, our linear models should predict the response to small simultaneous variations in the actuators (1%-5% of their set points) with the same fidelity as they predict the response to individual variations. For the operating point were selected the following values of the *RF power*, *pressure* and *flow rate*: 1000 Watts, 20 mTorr, and 30 sccm. In practice, however, the models may fail to accurately describe the system response due to neglected effects of nonlinearities of the plasma process and actuators. Physical limitations for the throttle valve created by the mechanical problems experienced during the opening represent one of the main source of this failure.

Using Matlab subroutines from the System Identification Toolbox open-loop sim-

Linear ARMA PGS models	V_{bias} e [Volts]	$[F]$ e [%]
Coupled models with delay	3.839	0.014
Coupled models without delay	4.09	0.014
Decoupled models with delay (cross-couplings)	11.996	0.143
Decoupled models with delay (direct-couplings)	9.465	0.1
Decoupled models without delay	12.0007	0.142

Table 3.1: Performance comparison for the linear $ARMA$ models.

ulations were performed for each model to see if the models match well with the experimental input-output data set (the second half of the data set was used to serve as validation data). The first half of the data set was used to estimate the coefficients of the models. During the identification process, after several experiments, it was possible to conclude that the lowest order $ARMA$ models yielding stable results that match well the experimental input-output data set was the second-order models. For each model two cases were studied:

(i) *with pure delay,* and

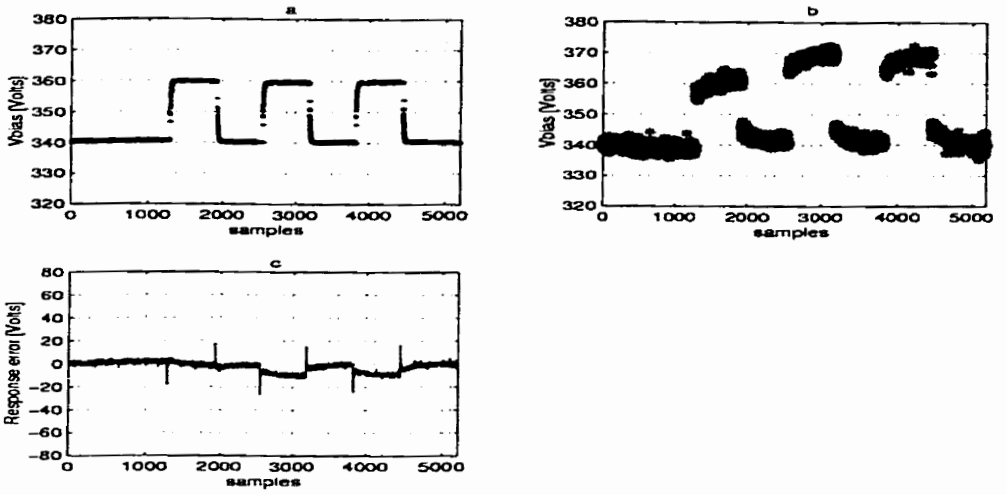
(ii) *without delay.*

The results of these simulations are synthesized in Table 3.1, and presented in Figures 3.2-3.21. In this table we present for each model the standard deviation errors between the model and the experimental data set for V_{bias} and fluorine concentration $[F]$ (e).

The coupled PGS models with pure delay represented in Figures 3.2-3.3 fit experimental input-output data set with the standard deviation errors $e = 3.839$ [Volts] for V_{bias}, and $e = 0.014$ [%] for $[F]$, and the coupled PGS models without delay represented in Figures 3.6-3.7 fit the same data with the standard deviation errors $e =$ 4.09 [Volts] for V_{bias}, and $e = 0.014$ [%] for $[F]$. The decoupled PGS models with pure delay represented in Figures 3.4-3.5, 3.8-3.9 fit the experimental input-output data set with the standard deviation errors e =11.996 [Volts] for V_{bias}, and $e = 0.0143$

[%] for $[F]$, respectively $e = 9.465$ [Volts] for V_{bias}, and $e = 0.1$ [%] for $[F]$. The decoupled PGS models without delay depicted in Figures 3.10-3.11 fit the same data with the standard deviation errors e =12.007 [Volts] for V_{bias}, and $e = 0.0142$ [%] for $[F]$. Good accuracy for these models is obtained if the maximum magnitude of the deviation error e doesn't exceed 10 Volts (3% of the V_{bias} setting point) for V_{bias} and 0.1% for $[F]$ (7% of $[F]$ setting point). These statistics reveal that the best match of the experimental input-output data set is performed by the coupled PGS models with pure delay closely followed by the PGS models without delay (almost the same standard deviation errors) and decoupled (direct-couplings) PGS models with pure delay which yield the smallest standard deviation errors with respect to their maximum magnitude. For this reason we can use one of the first two coupled PGS models to build our *MIMO* real-time control strategies presented in details in Chapters 4-7, and decoupled PGS models with delay to build *PI self-tuning* control strategy in Section 4.1.3.

To test the fidelity of our models in describing simultaneous actuator variations, two simultaneous pseudo-random binary signals (*PRBS*) were applied to the actuators as depicted in Figures 3.12-3.21. The mismatch in Figures 3.5-3.15 can be explained by a weak cross-couplings between the input and output variables of the PGS plant. The spikes and the small discontinuities in some of these figures are due to the spikes existing in the experimental data set. The PRBS applied to the *throttle position* for the coupled PGS models with and without delay represented in Figures 3.12-3.13, and Figures 3.16-3.17, respectively was given a slower switching rate because the dynamics associated with the throttle are slower than those associated with the power input.



(a) Coupled case with pure delay:

Figure 3.2: The performance of the coupled PGS model *with pure delay* on the V_{bias} for the open-loop system.
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

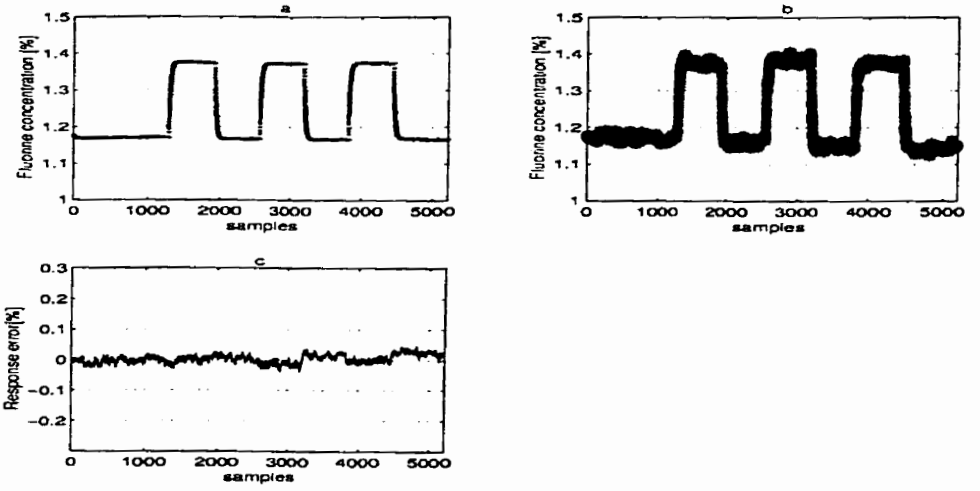


Figure 3.3: The performance of the coupled PGS model *with pure delay* on $[F]$ for the open-loop system.
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

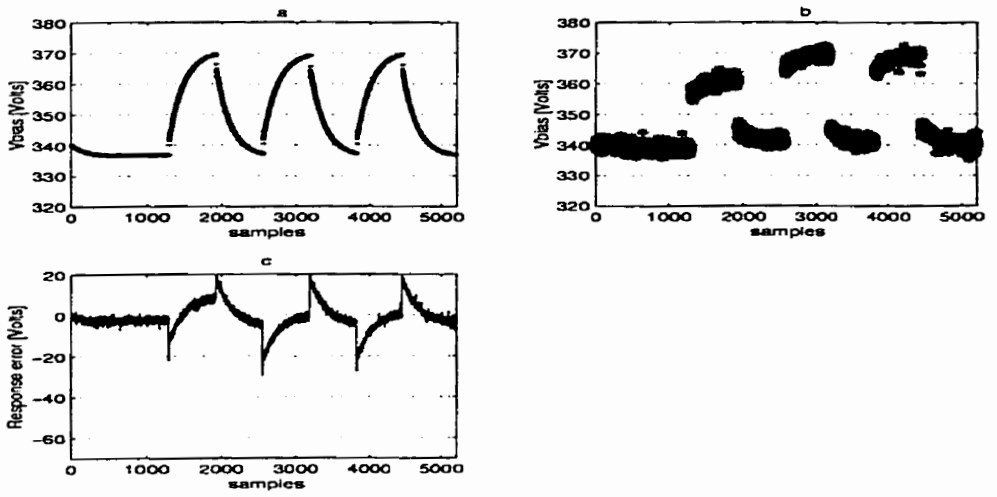


Figure 3.4: The performance of the decoupled PGS model *with pure delay* on the V_{bias} (cross-couplings: $V_{bias} - RF\ power$) for the open-loop system.
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

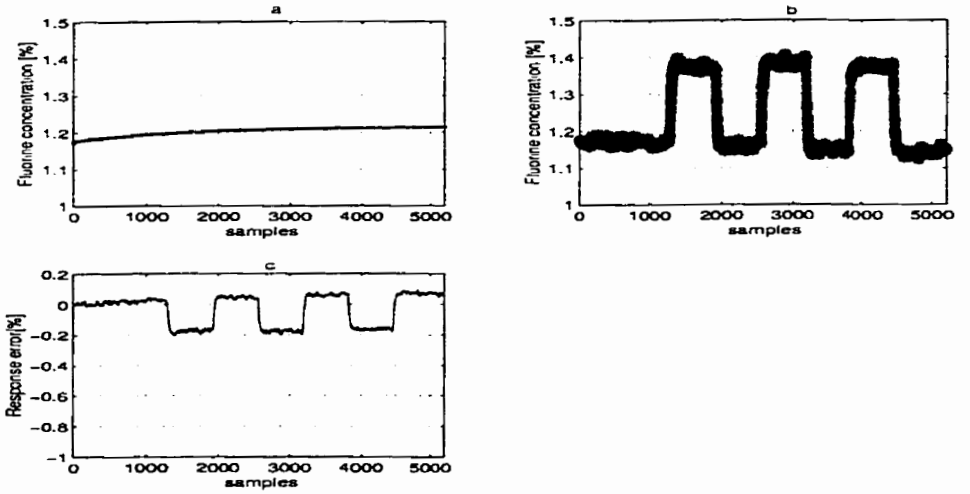


Figure 3.5: The performance of the decoupled PGS model *with pure delay* on $[F]$ (cross-couplings: $[F] - throttle\ position$) for the open-loop system.
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

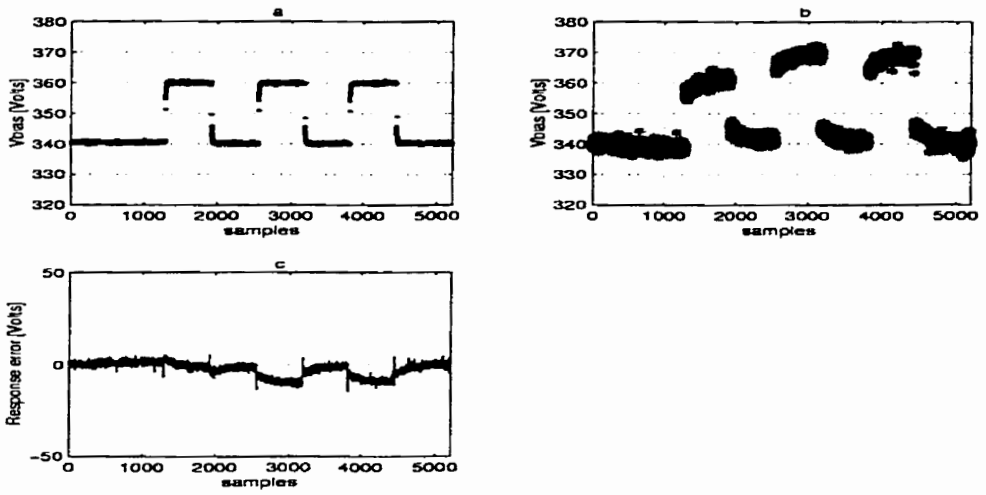


Figure 3.6: The performance of the coupled PGS model *without delay* on the V_{bias} for the open-loop system.
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

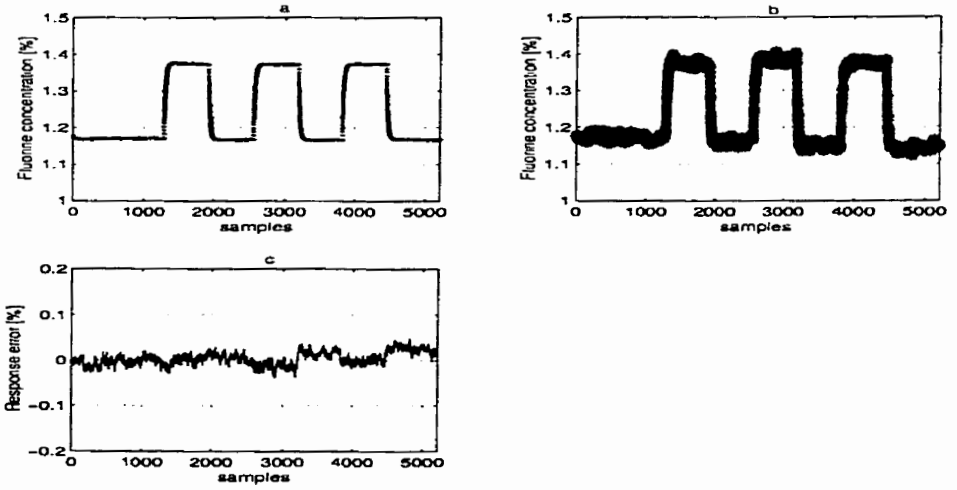


Figure 3.7: The performance of the coupled PGS model *without delay* on $[F]$ for the open-loop system.
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

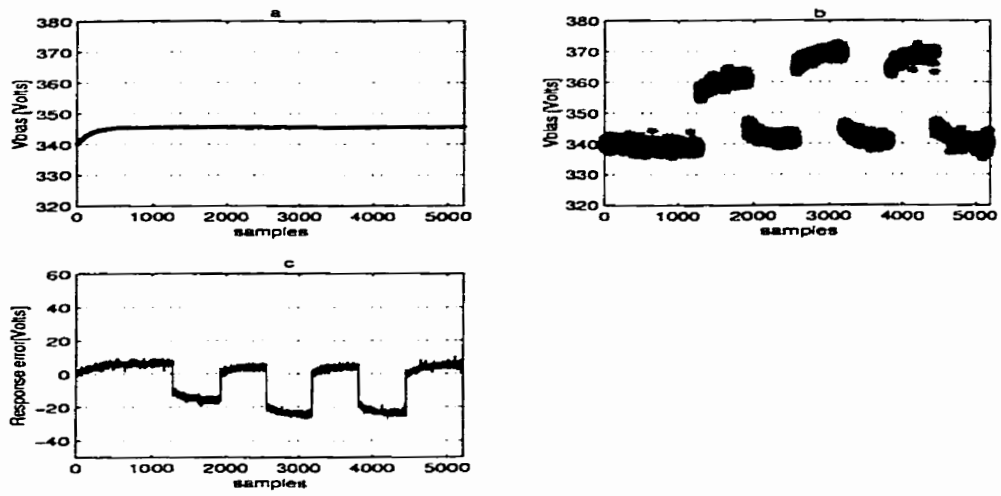


Figure 3.8: The performance of the decoupled PGS model *with pure delay* on the V_{bias} for open-loop system ($V_{bias} - throttle\,position$).
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

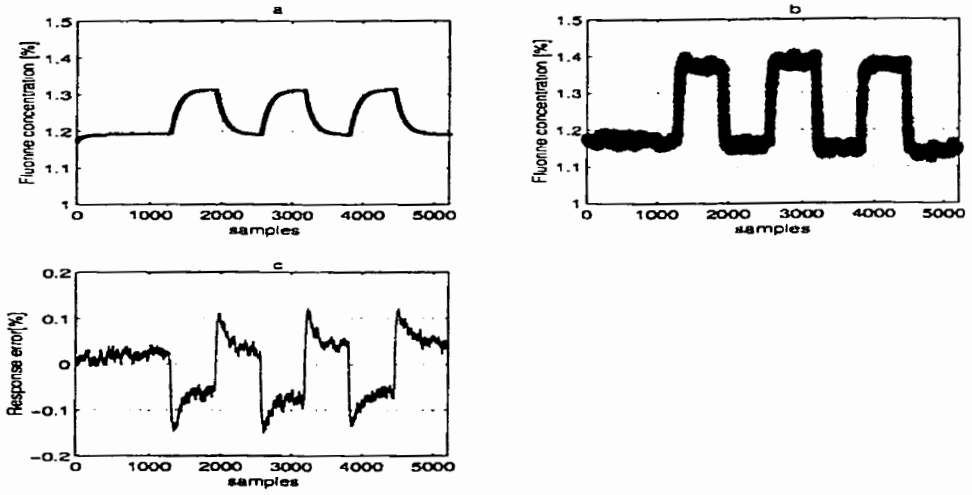


Figure 3.9: The performance of the decoupled PGS model *with pure delay* on $[F]$ for the open-loop system ($[F] - RF\,power$).
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

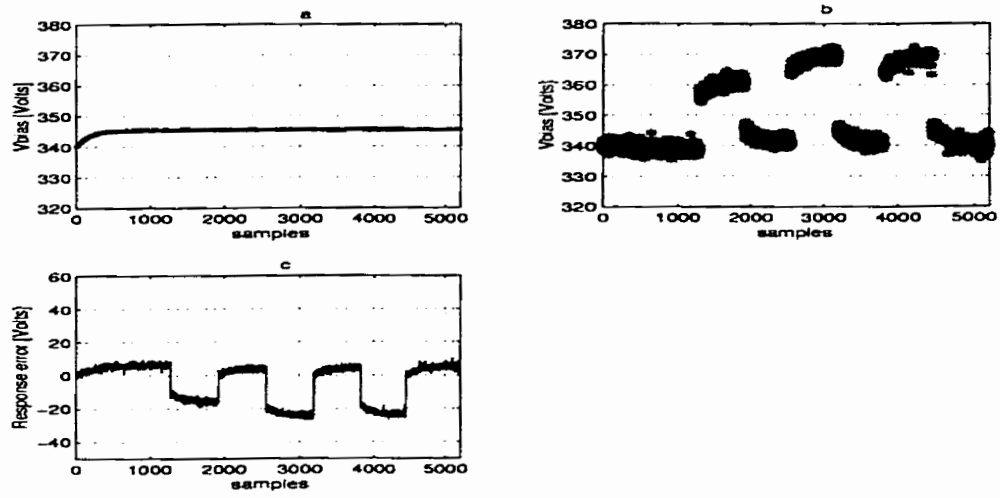


Figure 3.10: The performance of the decoupled PGS model *without delay* on the V_{bias} for the open-loop system.
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

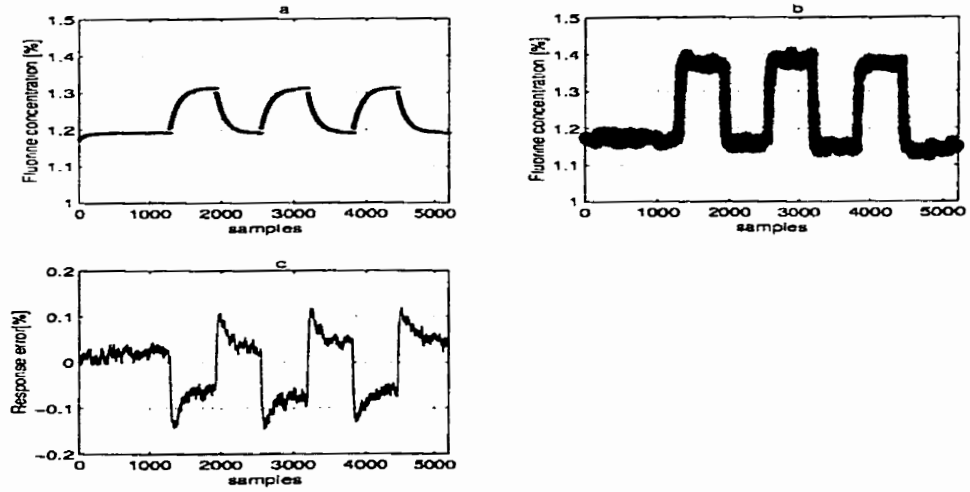


Figure 3.11: The performance of the decoupled PGS model *without delay* on $[F]$ for the open-loop system.
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

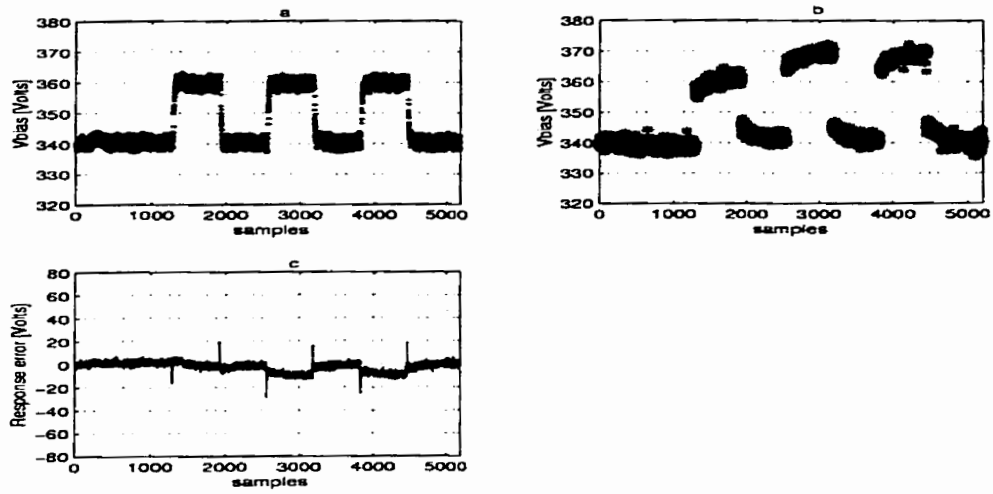


Figure 3.12: The open-loop system response to simultaneous *PRBS* inputs of the coupled PGS model *with pure delay*, V_{bias} .
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

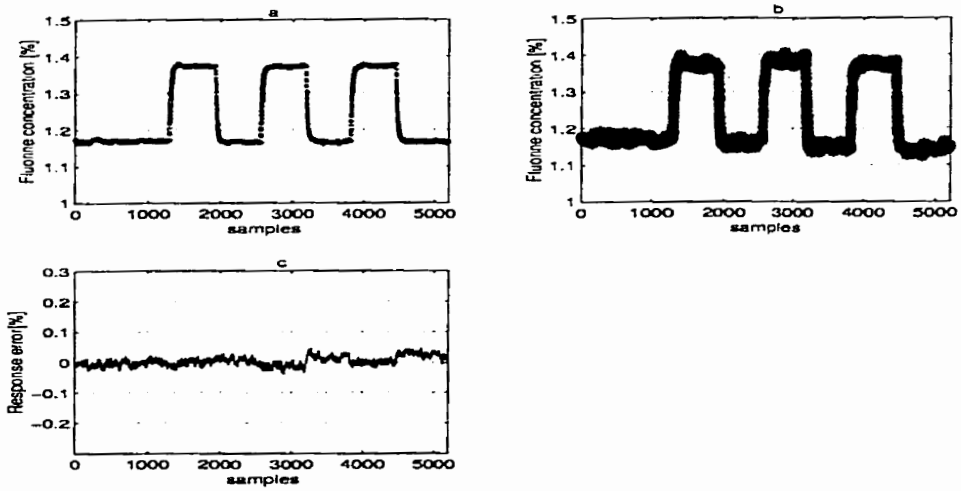


Figure 3.13: The open-loop system response to simultaneous *PRBS* inputs of the coupled PGS model *with pure delay*, $[F]$.
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

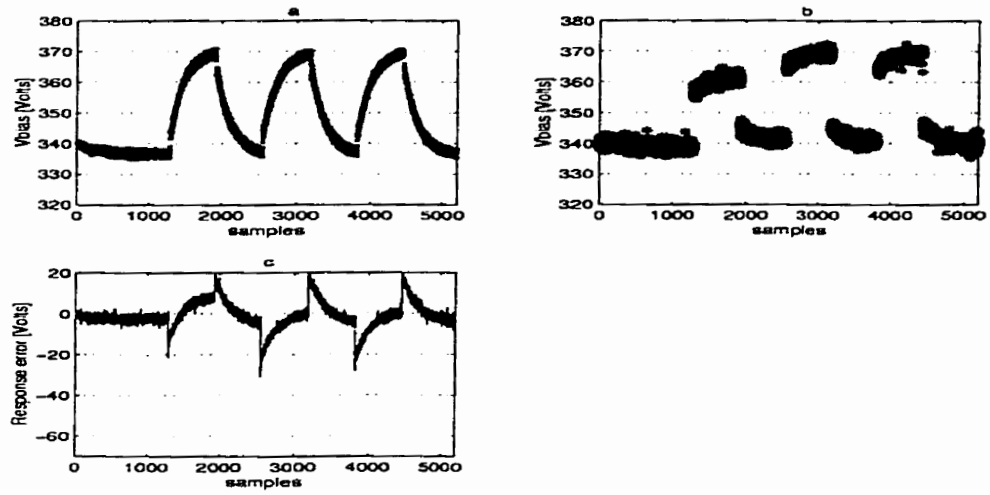


Figure 3.14: The open-loop system response to simultaneous *PRBS* inputs of the decoupled PGS model *with pure delay* (cross-couplings), V_{bias}.
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

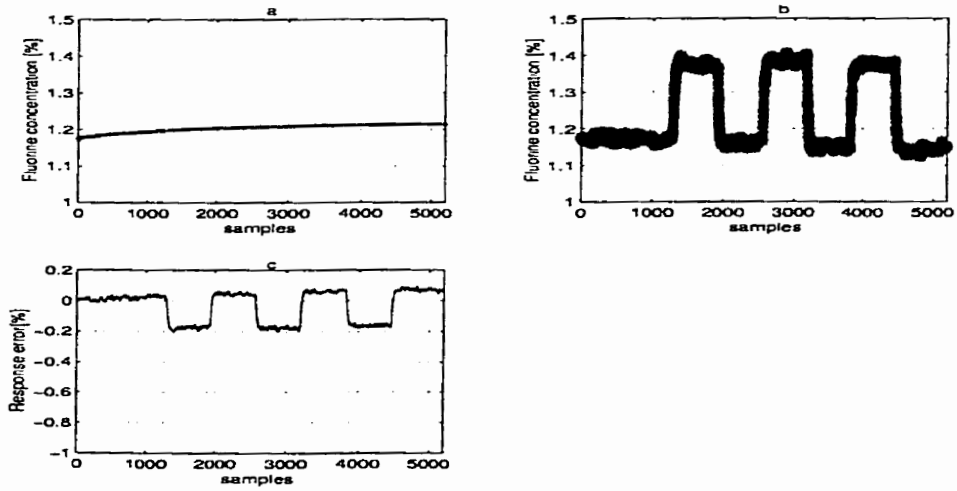


Figure 3.15: The open-loop system response to simultaneous *PRBS* inputs of the decoupled PGS model *with pure delay* (cross-couplings), $[F]$.
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

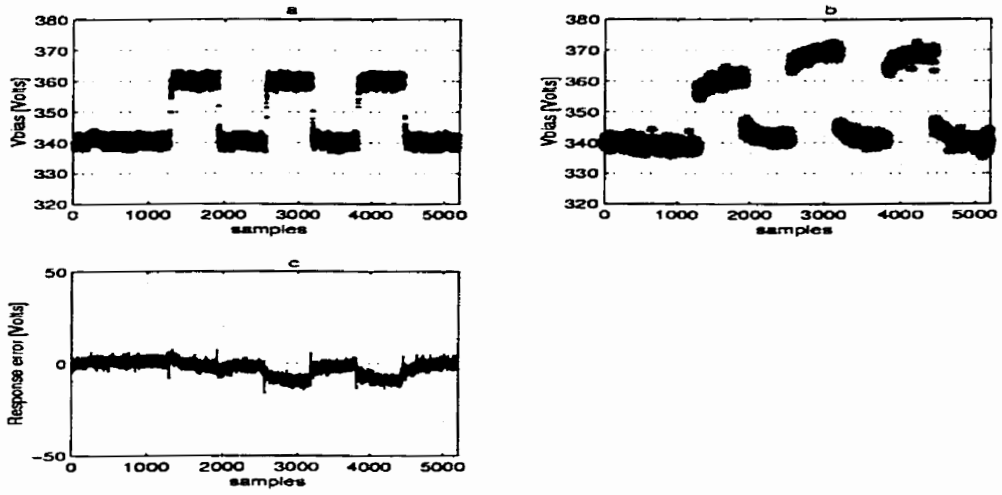


Figure 3.16: The open-loop system response to simultaneous *PRBS* inputs of the coupled PGS model *without delay*, V_{bias}.
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

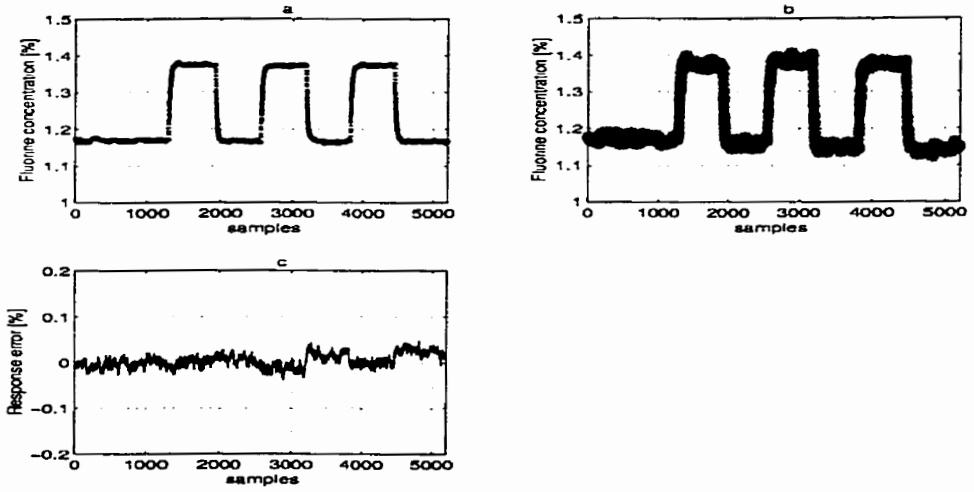


Figure 3.17: The open-loop system response to simultaneous *PRBS* inputs of the coupled PGS model *without delay*, $[F]$.
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

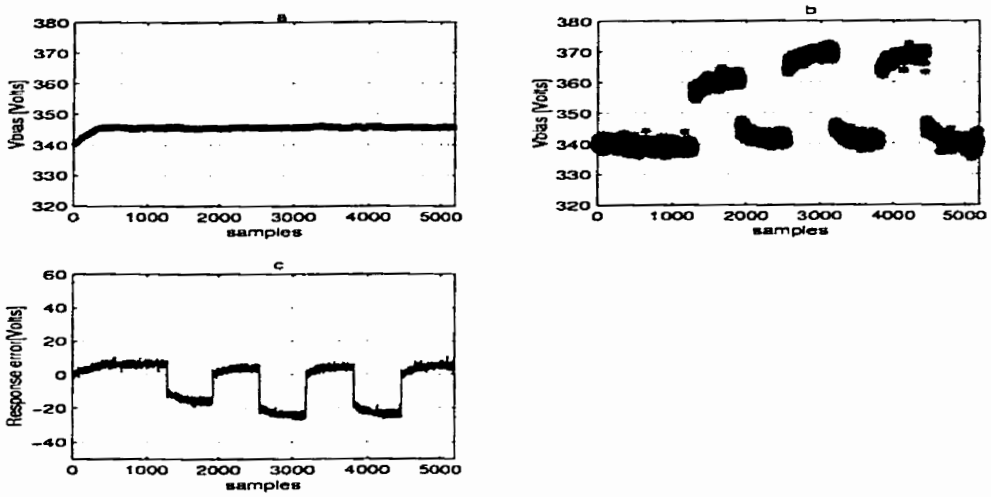


Figure 3.18: The open-loop system response to simultaneous *PRBS* inputs of the decoupled PGS model *with pure delay*, V_{bias}.
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

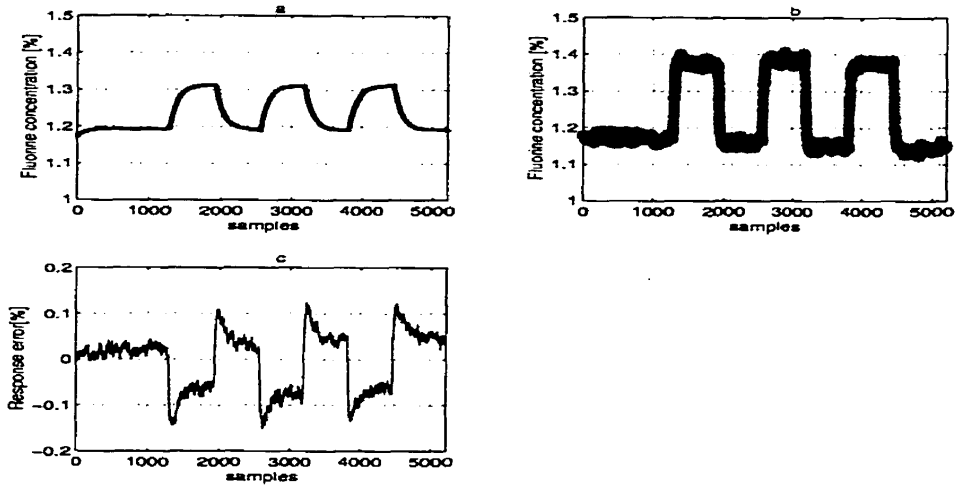


Figure 3.19: The open-loop system response to simultaneous *PRBS* inputs of the decoupled PGS model *with pure delay*, $[F]$.
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

3.2 Control System Analysis Techniques

In this section, we develop some techniques to measure the interactions between the input and output system variables to determine suitable pairs among them, namely we consider *Relative Gain Array* (*RGA*) and *Singular Value Decomposition* (*SVD*) techniques. By taking into account these interactions, we are able to build more appropriate *MIMO* models in coupled or decoupled forms as described in the Section 3.1.

3.2.1 Relative Gain Array (*RGA*)

This technique uses an interaction measure as a tool for the design of multivariable control systems to overcome theoretical and practical deficiencies of the system matrix representation. The term interaction is used here because in practice it is often desirable to control multivariable processes as if they were made up of isolated single variable processes. The resulting loops are in interaction with each other. An

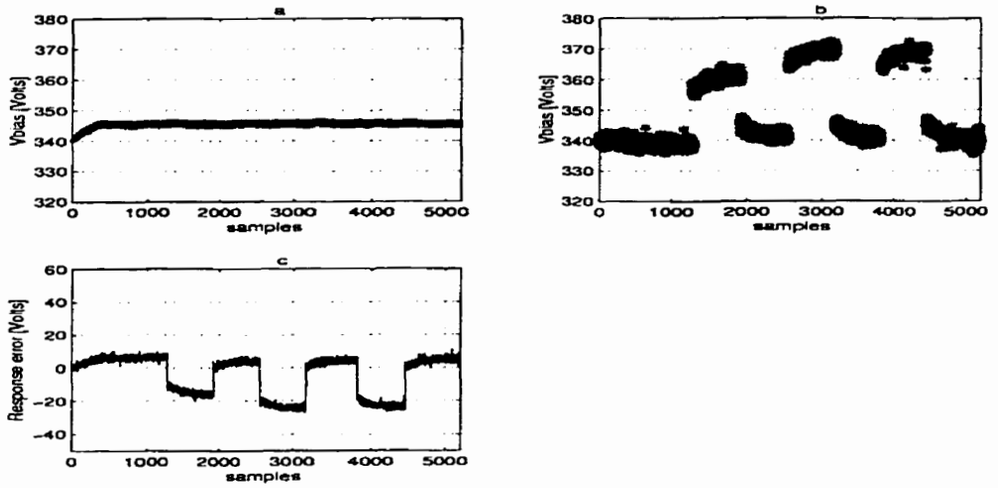


Figure 3.20: The open-loop system response to simultaneous *PRBS* inputs of the decoupled PGS model *without delay*, V_{bias}.
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

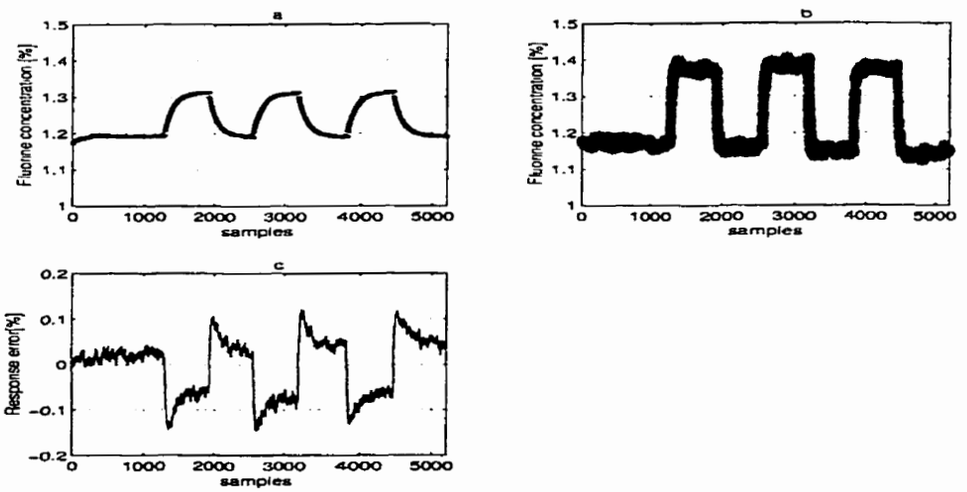


Figure 3.21: The open-loop system response to simultaneous *PRBS* inputs of the decoupled PGS model *without delay*, $[F]$.
Legend: (a) model output; (b) experimental data set; (c) representation error performance.

interaction measure attempts to answer the question: how is the measured transfer function between a given manipulated variables (*power* and *throttle position*) and a given controlled variables (V_{bias} and $[F]$) affected by the complete control of all other controlled variables. The measure taken to answer this question is the product of two gains representing the process gain in an isolated loop and the apparent process gain in the same loop when all other control loops are closed [3]. The first gain is the steady-state gain in open-loop and is given by

$$H(1) = \{h_{ij}\}_{i=1,\ldots,n\,,\,j=1,\ldots,n} = \lim_{z\to 1} H(z) \tag{3.24}$$

where $H(z)$ is the matrix transfer function of the PGS of dimension $n \times n$. Let us now set

$$\Gamma(1) = H^{-1}(1) = \{\gamma_{ij}\}_{i=1,\ldots,n\,,\,j=1,\ldots,n} \tag{3.25}$$

which represents the second steady-state gain matrix.

An input-output control loop is said to be noninteracting with other loops in a process if the steady-state gain of the loop is relatively insensitive to whether other loops are open or closed [3].

The "relative gain" was formulated by Bristol [22] as:

$$\mu_{ij} = h_{ij}\gamma_{ji} \tag{3.26}$$

The matrix of all relative gains for a process represents the RGA defined according to:

$$RGA = \{\mu_{ij}\}_{i=1,2\,,\,j=1,2} \tag{3.27}$$

This matrix is used to determine the pairings of manipulated and controlled variables

in single loops (*SISO*) such that the relative gains of the input-output pairings are positive and close to unity. Several properties of the *RGA* [3], [23] that are worth noting are as follows:

(*i*) The sum of any row or column is equal to unity.

(*ii*) Reordering the rows or columns of the steady-state gain matrix results in the same reordering of *RGA*.

(*iii*) The *RGA* is not affected by scaling of process inputs or outputs.

(*iv*) If the steady-state gain matrix is triangular or diagonal, then the corresponding *RGA* is the identity matrix.

(*v*) Measures much larger than one imply a " nearly " singular gain matrix.

(*vi*) The transfer function between y_i and u_j with all other loops closed will be non-minimum phase if μ_{ij} is negative .

(*vii*) A *SISO* closed-loop system controlled by a negative feedback controller must be stable if the controller is assigned to variable pairs with positive measure. The same system with negative measure will be stable only if the loop has positive feedback.

The above measure can serve as a design tool to select preferred processes and to specify the control structure once a process is selected. The control structure is specified by one to one pairing of the controlled and manipulated variables as a basis for control. The above properties suggest that the measure corresponding to the paired variables should be positive and as close as possible to one, negative numbers or much larger than one being particularly undesirable. The above design procedure is simple to use. Using this technique, we can simplify the structure of the model by

decoupling it into *SISO* closed loops. For these models we can perform many feedback control system strategies or we can adapt easily the existing control strategies from the literature. However, it is possible in some situations that these simplifications can lead to poorly conditioned systems, and for this reason one has to find alternative multivariable feedback control strategies for the *MIMO* systems.

By applying the above definitions to our PGS system under investigation, we obtain the following steady-state gains in open-loop for the *coupled case with delay*:

$$H_d(1) = \begin{bmatrix} 1.8 & 0.1994 \\ -0.0766 & 0.0018 \end{bmatrix} \tag{3.28}$$

and for the *coupled PGS model without delay*, we get :

$$H(1) = \begin{bmatrix} 13.4106 & 0.2027 \\ -0.0776 & 0.0017 \end{bmatrix} \tag{3.29}$$

where $H(1)$ and $H_d(1)$ are calculated by replacing $z = 1$ in the expressions of the transfer matrices given by (3.20)-(3.21).

The relative gain matrices RGA_d (*with delay*) and RGA (*without delay*) become, respectively:

$$RGA_d = \begin{bmatrix} 0.1750 & 0.8250 \\ 0.8250 & 0.1750 \end{bmatrix} \tag{3.30}$$

$$RGA = \begin{bmatrix} 0.5917 & 0.4083 \\ 0.4083 & 0.5917 \end{bmatrix} \tag{3.31}$$

From the RGA_d matrix we can see that the recommended pairings for PGS are found to occur on the off diagonal elements (positive and close to one), namely *throttle position* controls *fluorine concentration* $[F]$, *power* controls V_{bias} for the *coupled case*

with pure delay. For the *coupled case without delay,* we observe the difficulty in the interaction between input and output variables. In this last case *throttle position* controls V_{bias} and *power* controls *fluorine concentration* $[F]$. Therefore, in the first case we can use a decoupled control structure with two SISO loops acting as two independent loops. The control strategies clearly now become simpler compared to the MIMO coupled case.

3.2.2 Singular Value Decomposition Method (*SVD*)

In [3] SVD was used as a technique that requires only steady-state gain information, namely

$$G = H(1) = U \begin{bmatrix} \Sigma & 0 \\ 0 & 0 \end{bmatrix} V^T \tag{3.32}$$

where $H(1)$ is given in (3.24), U is an $m \times m$ orthogonal matrix, for which the columns are eigenvectors of GG^T, Σ is the singular value matrix of r singular values (σ_i) in descending order (σ_i's are positive square roots of the non-zero eigenvalues of G^TG ; non-zero eigenvalues of G^TG = non-zero eigenvalues of GG^T), and V is an $n \times n$ orthogonal matrix, for which the columns are eigenvectors of G^TG .

If G is the matrix of steady-state gains between the manipulated and controlled variables, then σ_i represents the ideal decoupled gain of the open-loop process between manipulated and controlled variables for the direction i [3]. This technique can be applied to either square or non-square systems, whereas the RGA can be applied only to square systems. The additional information that is obtained from SVD is the *condition number* of the steady-state gain matrix as the ratio of the *largest* and the *smallest* singular values. The *condition number* is used to identify the potential problems for controlling a process since it is a measure of the coupledness of the control

problem. A large *condition number* indicates the *high sensitivity* of the system. When the relative sensitivity in one multivariable direction is low, then a complete control objective cannot be met. Since *RGA* does not provide conclusive evidence of the nature of the interactions between the loops it is preferred to explore the use of *SVD* techniques in our problem. As an illustration, for our *with delay* system we have

$$H_d = Y_d \Sigma_d U_d^T \tag{3.33}$$

$$Y_d = \begin{bmatrix} 0.9991 & 0.0419 \\ -0.0419 & 0.9991 \end{bmatrix}, \Sigma_d = \begin{bmatrix} 1.8126 & 0 \\ 0 & 0.0102 \end{bmatrix} \tag{3.34}$$

$$U_d = \begin{bmatrix} 0.993 & -0.109 \\ 0.109 & 0.993 \end{bmatrix} \tag{3.35}$$

We conclude that the resulting *condition number* $(H_d) = 177.7058$, which is relatively large and indicates a poorly conditioned process. Also, for the *without delay case*, we have

$$H = Y \Sigma U^T \tag{3.36}$$

$$Y = \begin{bmatrix} 1 & 0.0058 \\ -0.0058 & 1 \end{bmatrix}, \Sigma = \begin{bmatrix} 13.4124 & 0 \\ 0 & 0.0029 \end{bmatrix} \tag{3.37}$$

$$U = \begin{bmatrix} 0.9999 & -0.0151 \\ 0.0151 & 0.9999 \end{bmatrix}, \tag{3.38}$$

and the resulting *condition number* $(H) = 4624.9655$, which is even larger than the previous case, indicating a more poorly conditioned process.

The source of the above problem is revealed by studying the right singular matrices (U_d^T, U^T), that is

$$u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \rightarrow U_d^T = \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} \rightarrow \Sigma_d \rightarrow Y_d = \begin{bmatrix} Y_1 & Y_2 \end{bmatrix} \rightarrow y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

$$u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \rightarrow U^T = \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} \rightarrow \Sigma \rightarrow Y = \begin{bmatrix} Y_1 & Y_2 \end{bmatrix} \rightarrow y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

This structure represents the transformation of the input vector $u \in U$ whose space U is defined by the set of vectors (U_1, U_2) (the rows of the matrix U^T) into a output vector $y \in Y$ whose space Y is generated by the set of orthonormal vectors (Y_1, Y_2) (the columns of the matrix Y), where

$U_1 = d_{11}u_1 + d_{12}u_2$, is the *most sensitive (best) control direction*,

$U_2 = d_{21}u_1 + d_{22}u_2$, is the *worst control direction, independent of* U_1.

$Y_1 = c_{11}y_1 + c_{12}y_2$, is the *most sensitive observed direction*, and

$Y_2 = c_{21}y_1 + c_{22}y_2$, is the *least observable direction*, independent of Y_1 [3].

The above formulation can be summarized as follows

Inputs: $\begin{bmatrix} throttle\ position & power \end{bmatrix} = \begin{bmatrix} u_1 & u_2 \end{bmatrix}$

Outputs: $\begin{bmatrix} V_{bias} \\ fluorine\ concentration \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$

$$U^T = \begin{bmatrix} 0.9999 & 0.0151 \\ -0.0151 & 0.9999 \end{bmatrix} = \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} \Rightarrow \text{Inputs}$$

$$Y = \begin{bmatrix} 1 & 0.0058 \\ -0.0058 & 1 \end{bmatrix} = \begin{bmatrix} Y_1 & Y_2 \end{bmatrix} \Rightarrow \text{Outputs}$$

SVD can also be used to determine suitable control loop pairings [12], [3], namely the so-called natural loop pairings. The best controllable direction is paired with

the best observable direction and the worst controllable direction is paired with the least observable direction. Therefore Y_1 is paired with U_1 and Y_2 is paired with U_2. In addition each principal component direction is assumed to be composed of only one space dimension, i.e, only one component of each vector Y_i or U_i is one, and all other components are equal to zero, namely

$Y_1 = [\ 1\ 0\ ]$, $Y_2 = [\ 0\ 1\ ]$, and $U_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, $U_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$. Since this situation is rarely met in reality, therefore, in practice, the space variable with the largest coefficient is taken to be the only component. If some or all of the coefficients have nearly equal values, then using a control structure with SISO loops will result in an unsatisfactory response. In our system, for both cases, *throttle position* would be paired with V_{bias}, and *RF power* with the $[F]$, and so for the first case *with delay* we get a result that is opposite to the one obtained using the *RGA* technique and for the second case i.e., *without delay*, we obtain the same result. The last principal component of U_d and U is composed entirely of *RF power* (one coefficient in U_d is close to unity (0.993) and the other is near zero (0.10099) and one coefficient in U is closed to unity (0.9999) and the other is near zero (0.00151), and has a singular value ($\sigma_2 = 0.0102$ for the first case and $\sigma_2 = 0.0029$ for the second case). Thus, the system is relatively unaffected by *RF power* and the control objectives (V_{bias} and *fluorine concentration* $[F]$) cannot be achieved. This method reveals that the first case with delay is better than the second one without delay. However, both *SISO ARIMA* models still remain incapable of capturing all the dynamics of the PGS and of reflecting accurately the physical phenomenom within the reactive ion etching plasma process.

3.3 Nonlinear Models: RIE Neuromodels

Process modelling permits an engineer to manipulate and optimize the process efficiency with a minimum amount of experimentation. An accurate process model offers a competitive edge in today's complex and competitive semiconductor manufacturing. As discussed in the previous sections, precise modelling of the RIE from a fundamental physical standpoint is difficult due to the extremely complex nature of particle dynamics within a plasma. Recently, empirical RIE models derived from neural networks [2], [16], [14] have been shown to exhibit superior performance in accuracy, predictive capability, and robustness over more traditional statistical approaches (regression methods). Neuromodel predictions of the RIE process outputs are used in conjunction with genetic algorithms and other optimization techniques to optimize the etching process performance and for recipe synthesize.

One of our goals in this thesis is to determine an appropriate neuromodel for the highly complex PGS that takes into account the nonlinearity of the actuator (*throttle valve*). The experimental input-output data set will be used to train feedforward neural networks using an error backpropagation algorithm. We will focus our attention on matching model predictions with measurements for network learning and generalization. For this purpose our proposed neural networks consist of three layers configured in different architectures, ranging from a 2-10-2 network to a 4-10-2 and 6-10-2 structures all being trained by the Levenberg-Marquardt backpropagation error algorithm [24].

Even though a simple steepest descent gradient algorithm can be efficient, there are situations when moving the weights within a simple learning step along the negative gradient vector by a fixed proportion will yield a minor reduction of error. For flat error surfaces for instance, too many steps may be required to compensate for

small gradient values. Furthermore, the error contours may not be circular and the gradient vector may not point toward the minimum. To avoid these situations one may replace the gradient descent method by the Gauss-Newton optimization method, which uses the second derivative of the error function E, namely its Hessian matrix $H(w) = \nabla_w^2 E = \nabla(\nabla_w E)$. To update the weights, a recursive Gauss-Newton optimization algorithm may be expressed in the following matrix form

$$W^{(i)} = W^{(i-1)} - \gamma[H(w^{(i-1)})]^{-1}(\nabla_w E) \tag{3.39}$$

Because the Hessian matrix may be singular, it can be made invertible by using the Levenberg-Marquardt relaxation as follows:

$$\hat{H}(w) = H(w) + \mu I \tag{3.40}$$

where μ is a relaxation parameter and I is an identity matrix. The Levenberg-Marquardt algorithm is preferred for a small number of weights because the computational speed of the inverse Hessian matrix decreases when the number of the weights increases. Otherwise, the steepest descent optimization algorithm is preferred.

The backpropagation algorithm attempts to minimize the error between the output of the network when compared to the target or the desired response. The number of hidden neurons and layers are varied to provide optimal network performance.

The development of an optimal neural network structure is complicated by the fact that backpropagation networks contain several adjustable parameters for which the optimal values are initially unknown. These include structural parameters (such as the number of hidden layer neurons, initial weights and biases) as well as learning parameters (such as the learning rate, momentum, and error goal). The learning rate determines the speed of convergence by regulating the step size. However, the

network may settle far away from the global minimum of the error surface if the learning rate is too large. On the other hand, smaller rates can ensure stability of the network by diminishing the gradient of noise in the weights, but result in longer training times [2], [14]. For this reason the algorithm is improved by introducing an adaptive learning scheme which decreases considerably the training time.

A smaller training tolerance usually increases learning accuracy, but can also result in less generalization capability as well as longer training time [2], [14]. Conversely, a larger tolerance enhances convergence speed at the expense of accuracy in learning. It is shown in the literature that a single hidden layer is sufficient for learning any function, but the number of hidden neurons can grow without a bound [16]. This of course, may result in a network with a large number of connections which defeats the main purpose of having an accurate prediction. By increasing the number of hidden layers, each consisting of sigmoidal nodes, the complexity of the network can increase more rapidly than the number of connections. The optimum network architecture should have a minimum number of connections and produces a low cross-validation error.

Development of neural network models typically consists of considerable training and testing. The objective is to find a network that will perform well on the test data. For the training set we will select the first half of the experimental input-output data set to update the weight matrices and use the other half as the test set. Network performance is measured by the root mean squared error, σ which is given by:

$$\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3.41}$$

where n is the size of the test set, y_i is the measured value of the output, and $\hat{y}_i$ is the response provided by the neural networks.

All the neural network architectures proposed in this section are trained on the experimental data set to learn the mapping from inputs to outputs of the process model. A small network with only ten hidden neurons was sufficient to generate this mapping. To avoid overfitting we limited the number of neurons to the fewest as possible as long as the network converges to the desired error level. The training was terminated once the error goal was achieved.

The input variables to the neural network models are u_1 =*throttle valve position* and u_2 = *RF power* and the outputs are y_1 =V_{bias} induced and y_2 =*fluorine concentration* $[F]$. Present values of the output variables or delayed output and input variables can also be used as inputs to the model, depending on the complexity of the representation of the PGS considered.

Our proposed four neuromodel structures are described in detail below:

(*i*) *nonlinear static model (2-10-2).* The neural network objective is to represent a static model of the PGS which is assumed to be expressed as a nonlinear function f where $u(k) = \begin{bmatrix} u_1(k) & u_2(k) \end{bmatrix}^T$, $y(k) = \begin{bmatrix} y_1(k) & y_2(k) \end{bmatrix}^T$, and

$$y(k) = f(u(k)) \tag{3.42}$$

(*ii*) *nonlinear first-order model (4-10-2).* In this case the delayed plant output $y(k-1)$ is used in addition to the present plant input $u(k)$ as input variables i.e., the dynamic input-output model is assumed to be expressed as a recursive equation of first-order

$$y(k) = f(y(k-1), u(k)) \tag{3.43}$$

(*iii*) *nonlinear second-order model (6-10-2).* In this case the delayed plant outputs $y(k-1)$, $y(k-2)$ are used in addition to the plant input $u(k)$ as input vectors,

i.e., the dynamic input-output model is assumed to be expressed as a recursive equation of second-order

$$y(k) = f(y(k-1), y(k-2), u(k)) \tag{3.44}$$

(*iv*) *nonlinear first-order model (6-10-2) with delayed input vector* $u(k-1)$. In this case the delayed output $y(k-1)$ and the delayed input $u(k-1)$ are used in addition to the plant input $u(k)$ as input vectors, i.e.,

$$y(k) = f(y(k-1), u(k), u(k-1)) \tag{3.45}$$

The results of the simulations presented in the next section show that the last two neuromodel representations yield the best performance due to the presence of their internal feedback and the delayed input signals as input vectors to the networks.

After constructing the neural network-based models we are now able to develop neurocontrollers that must meet the following performance objectives for the closed-loop system:

(*i*) tracking the reference target without delay,

(*ii*) preventing disturbances from influencing the output, and

(*iii*) rejecting noise, i.e., not responding to spurious fluctuations.

It is known that objectives (*i*) and (*iii*) are sometimes mutually exclusive. In other words, a neurocontroller that improves both the speed of the response and rejects noise is, in general, very challenging to design.

Neuromodel	V_{bias} e [Volts]	$[F]$ e [%]
Static representation	2.99289	0.0256
First-order	1.387	0.0159
Second-order	1.3516	0.0121
First-order with delayed input	1.387	0.0172

Table 3.2: Performance comparison for the neuromodels.

3.4 Neuromodel Simulation Results

In this section we design several neuromodels capable of learning and generalizing accurately the highly nonlinear dynamics of the PGS plant in open-loop. The results of the training phase and the capability of the neural network models to learn and represent the experimental input-output data set are presented below. In Table 3.2 we synthesize the performance of these neuromodels represented by the standard deviation errors between the models and the experimental data set (e). The nonlinear static model network fit the experimental input-output data set (not shown) in the testing phase with the standard deviation errors of $e = 2.99289$ [Volts] for V_{bias} and $e = 0.0256$ [%] for $[F]$. The nonlinear first-order model network fit the same data with the standard deviation errors of $e = 1.387$ [Volts] for V_{bias} and $e = 0.0159$ [%] for $[F]$. The best fit is performed by the nonlinear second-order model network depicted in Figures 3.26-3.27 which fit the experimental input-output data with the smallest standard deviation errors of $e = 1.3516$ [Volts] for V_{bias} and $e = 0.0121$ [%] for $[F]$. Finally, the nonlinear first-order model network with delayed control results in standard deviation errors of $e = 1.387$ [Volts] for V_{bias} and $e = 0.0172$ [%] for $[F]$. The maximum magnitude of these standard errors is the same as for linear $ARMA$ models developed in Section 3.111.

3.5 Summary

In this chapter, we have developed four linear and four nonlinear models for the PGS process. In the *decoupled PGS models,* we can consider the multivariable PGS plant as two independent SISO loops, for which we will develop in Chapter 4 a real-time feedback control strategy, namely a *PI self-tuning* adaptive controller. To determine suitable pairs between the input variables and the output variables, we used *RGA* and *SVD* techniques. Using the *coupled PGS models with pure delay* in state-space representation, we will develop later real-time *multi-input multi-output (MIMO)* control strategies such as *Linear Quadratic Gaussian (LQG)* control (Section 4.2), stochastic minimum variance adaptive control (Section 4.3) and robust control (Chapter 7). The neuromodels will be used in Chapter 5 to build four real-time neurocontrol strategies which will ensure the stability, good tracking error and robustness of the RIE system in closed-loop.

As a consequence of our model validation results, by computing the standard deviation errors between the model output and the measured output, we can conclude that the *ARIMA* models (i), (ii) provide accurate fit of the experimental input-output data sets and work well in a small operating range. The nonlinear models such as neural network models (i)-(iv) work well in larger operating range, and capture the nonlinearities of the actuators and the plant.

Among the important problems investigated here are the optimization of the architecture of the neural networks and the type of networks used to obtain a good fit with the experimental input-output data set. The training used for determining the neural network models of the RIE system is based on dynamic/static backpropagation using the Levenberg-Marquardt algorithm.

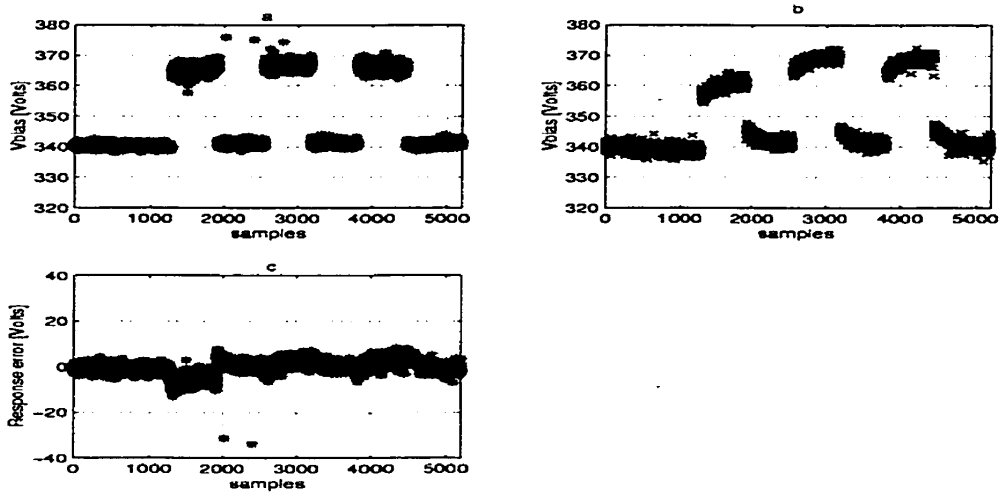


Figure 3.22: Generalization results for the nonlinear static representation for the *dc* V_{bias}.
Legend: (a) neuromodel output; (b) experimental data set; (c) representation error performance.

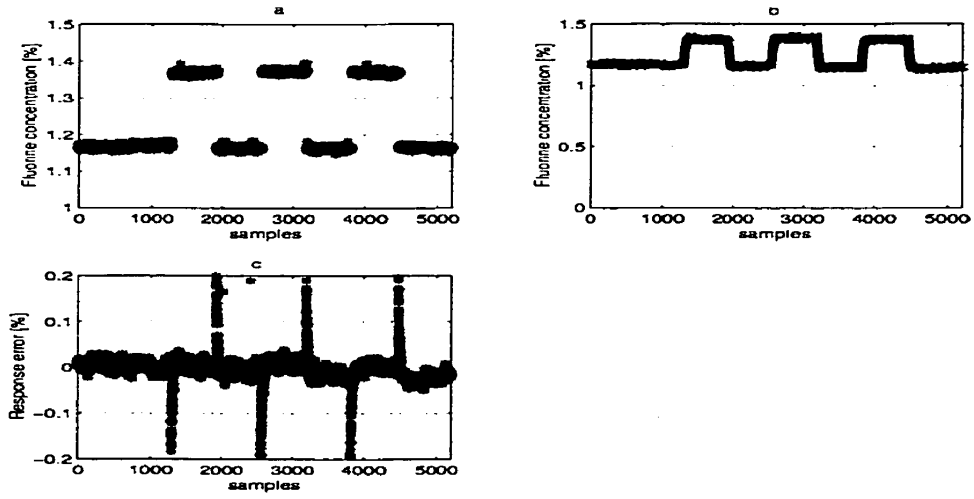


Figure 3.23: Generalization results for the nonlinear static representation for the *fluorine concentration* $[F]$.
Legend: (a) neuromodel output; (b) experimental data set; (c) representation error performance.

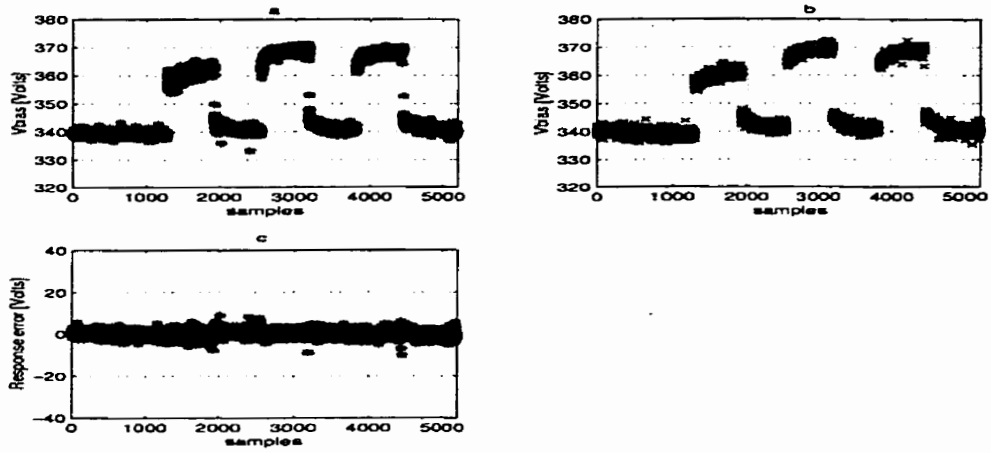


Figure 3.24: Generalization results for the nonlinear first-order representation for the *dc* V_{bias}.
Legend: (a) neuromodel output; (b) experimental data set; (c) representation error performance.

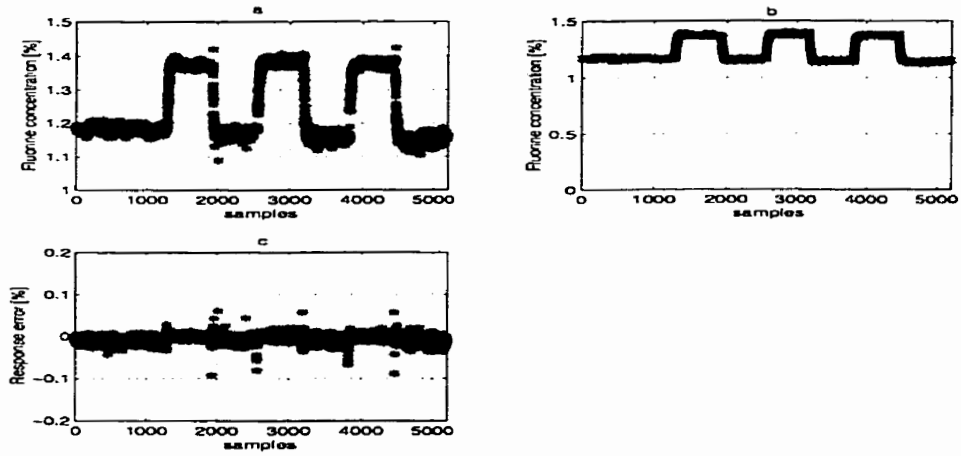


Figure 3.25: Generalization results for the nonlinear first-order representation for the *fluorine concentration* $[F]$.
Legend: (a) neuromodel output; (b) experimental data set; (c) representation error performance.

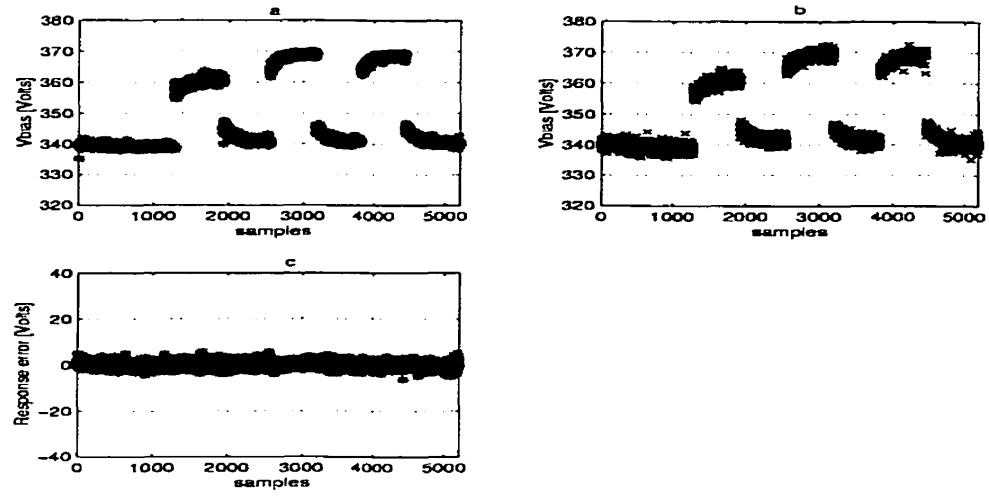


Figure 3.26: Generalization results for the nonlinear second-order representation for the *dc* V_{bias}.
Legend: (a) neuromodel output; (b) experimental data set; (c) representation error performance.

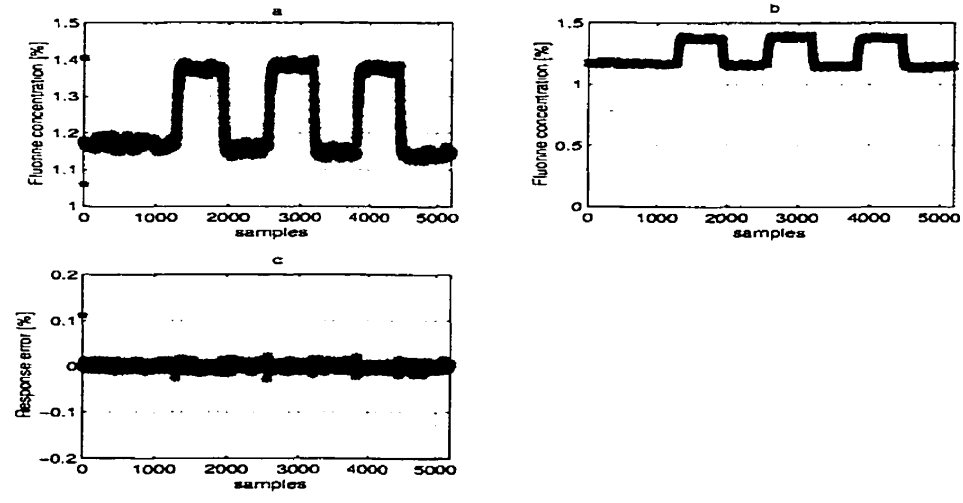


Figure 3.27: Generalization results for the nonlinear second-order representation for the *fluorine concentration* $[F]$.
Legend: (a) neuromodel output; (b) experimental data set; (c) representation error performance.

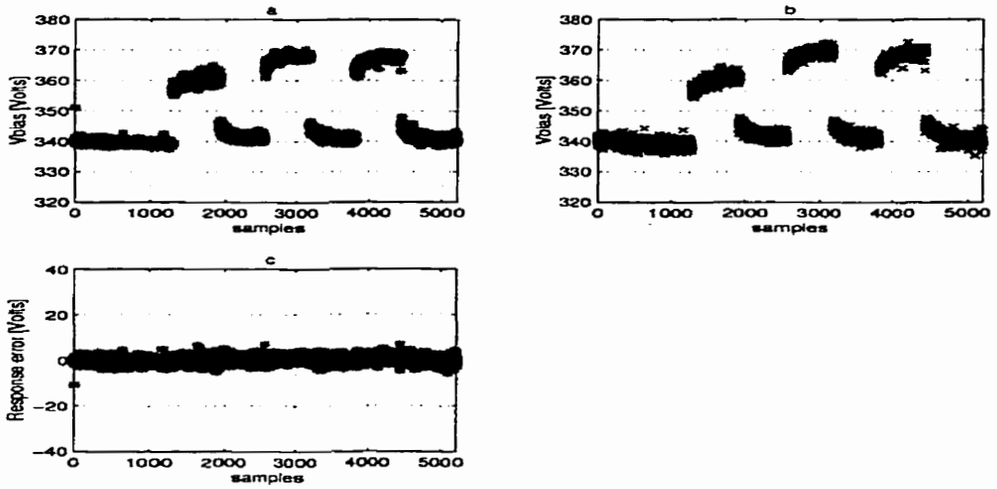


Figure 3.28: Generalization results for the nonlinear first-order representation with control delayed for the dc V_{bias}.
Legend: (a) neuromodel output; (b) experimental data set; (c) representation error performance.

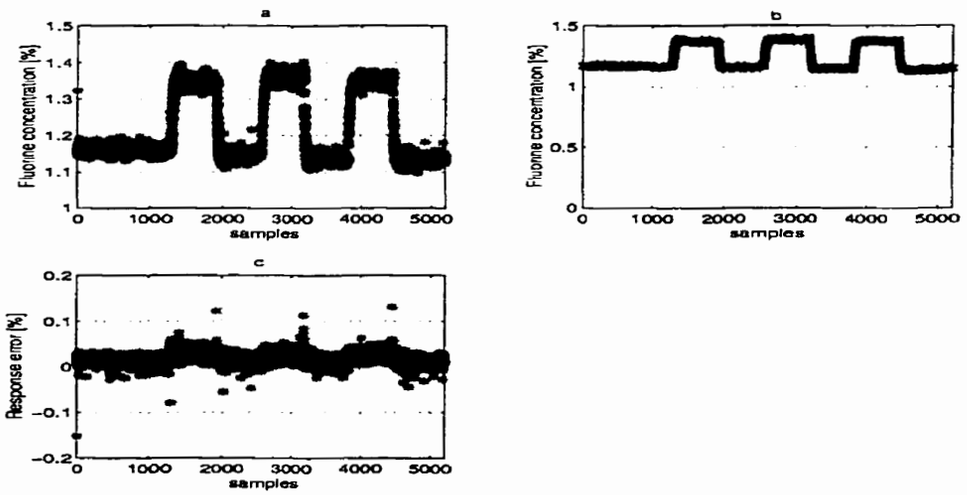


Figure 3.29: Generalization results for the nonlinear first-order representation with control delayed for the *fluorine concentration* $[F]$.
Legend: (a) neuromodel output; (b) experimental data set; (c) representation error performance.

Chapter 4

MIMO Feedback Control Strategies

4.1 Adaptive Controllers Based on Quadratic Cost Function Minimization

4.1.1 Introduction

Automatic tuning of controllers for industrial processes has received both theoretical and practical interest for many years. Self-tuning control is one approach to the automatic tuning problem. A self-tuning controller has three main elements:

(*i*) A standard feedback law in the form of a difference equation that acts upon a set of values such as the measured output and feedforward signals and the current set-point to provide the new control action,

(*ii*) A recursive parameter estimator that monitors the plant's inputs and outputs and computes an estimate of the plant dynamics in terms of a set of parameters in a prescribed structural model, and

(*iii*) The parameter estimates that are fed into a control design algorithm which then provides a new set of coefficients for the feedback law. The control design

algorithm simply accepts current estimates and ignores their uncertainties.

Appropriate modifications to the basic algorithms and their relative robustness are still open topics for current research. The application of self-tuning control strategies started with the development of self-adaptive systems in the aircraft industry for changing flight conditions. The usefulness of linear adaptive control techniques is still debated in the control system community. The principal reason is that all controlled processes for which linear adaptive control systems might be suitable are essentially nonlinear and stochastic, and therefore difficult to control and to analyse. If they were not nonlinear they could be optimally controlled by classical linear controllers; and if they were not in some way uncertain, or stochastic, there would be no need for learning in the form of self-adjustment or estimation of coefficients. Nonlinear, stochastic problems are difficult to study analytically because, there can be in general no analytic solutions to them. In particular no general design procedures are available for designing controllers for nonlinear stochastic processes.

Self-tuning control techniques can be classified into two main categories:

(*i*) explicit method, where the process model is used and the control is based on estimated model parameters which do not directly appear in the control law. At each sampling interval, the parameters in the process model are estimated recursively from input-output data of the PGS and the controller parameters are then updated as shown in Figure 4.1. This approach is the basis for the self-tuning control approach. The dynamic model is assumed to be either a linear input-output difference equation or a linear discrete-time state space representation with constant parameters. The parameter estimator is designed based on the RLS algorithm. The controller is designed in such a way that it minimizes a quadratic cost function with the form given by (4.3)-(4.4) [22].

(*ii*) implicit methods where the process model is converted into a predictive form

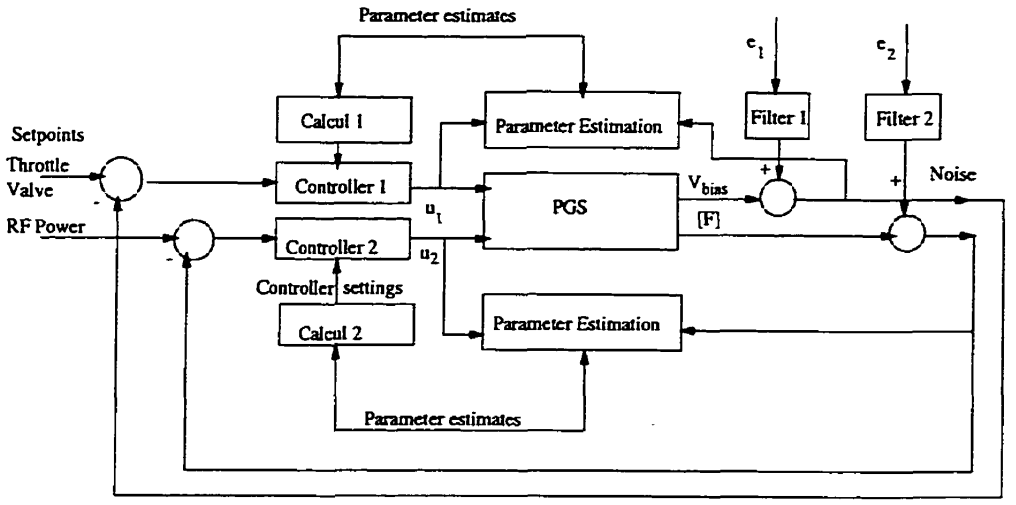


Figure 4.1: Self-tuning controller for the PGS

that allows the future process output to be predicted from the current and past values of the input and output variables by using a predictive model as shown in Figure 4.1. In this case, the control law parameters are directly updated from the input-output measurements.

In this figure, the blocks *Calcul 1* and *Calcul 2* represent the subroutines for the *RLS* procedure, and the *low pass Filter 1* and *Filter 2* blocks are used to filter the white noises e_1 and e_2. These filters have the same significance as N_1 and N_2 defined in the Chapter 3, equation (3.1).

4.1.2 Design of the Predictive Models of the PGS

In this section, we design an adaptive *self-tuning* controller based on minimization of a quadratic cost function described by equations (4.3), (4.4), known in the literature as minimum variance control, or minimum regulation [22].

Minimum variance control [25], [22] is applied here to our *decoupled PGS models with pure delay* (*direct – couplings*) represented in the Figures 3.8-3.9 to compare with other advanced control algorithms such as neurocontrollers and robust controllers

that are developed subsequently in the thesis. The process model can first be written in the predictive form [25]:

$$y_1(k+1) = \frac{B_1}{A_1}u_1(k) + \frac{C_1}{A_1\Delta}e_1(k+1) \tag{4.1}$$

$$y_2(k+1) = \frac{B_2}{A_2}u_2(k) + \frac{C_2}{A_2\Delta}e_2(k+1) \tag{4.2}$$

where

y_1, y_2 are defined as $y_1 - y_{1sp}$, $y_2 - y_{2sp}$

u_1, u_2 are defined as $u_1 - u_{1sp}$, $u_2 - u_{2sp}$

y_{1sp}, y_{2sp} , u_{1sp}, u_{2sp} denoting the steady-state values of y_1, , y_2, $and\, u_1$, u_2, respectively.

The objective is to design two controllers that minimize the following cost functions:

$$\min J_1 = E\left\{\hat{y}_1(k+1)^2 + r_1\Delta u_1(k)^2\right\} \tag{4.3}$$

$$\min J_2 = E\left\{\hat{y}_2(k+1)^2 + r_2\Delta u_2(k)^2\right\} \tag{4.4}$$

where $\hat{y}_1(k+1)$, $\hat{y}_2(k+1)$ represent the one step ahead (given by the pure delay) predicted values of the V_{bias} and *fluorine concentration* $[F]$ computed by the *RLS estimation* algorithm [35], [26], [27], [22], [25]. The model of the process (4.1)-(4.2) can be rewritten in the form indicated in [22], [25] as:

$$A_1\Delta y_1(k+1) = B_1\Delta u_1(k) + E_1A_1\Delta e_1(k) + F_1e_1(k) \tag{4.5}$$

$$A_2\Delta y_2(k+1) = B_2\Delta u_2(k) + E_2A_2\Delta e_2(k) + F_2e_2(k) \tag{4.6}$$

where

$\Delta y_i(k+1) = y_i(k+1) - y_i(k)$, $\Delta u_i(k+1) = u_i(k+1) - u_i(k)$, and $\Delta e_i(k+1) = e_i(k+1) - e_i(k)$, $i = 1, 2$, represent the first order difference of the output, input, and error, respectively, and the polynomials F_1, F_2, E_1, E_2 are of degrees $n_{f_1}, n_{f_2}, n_{e_1}, n_{e_2}$, and satisfy the Diophantine equations [22]:

$$C_1 = E_1 A_1 \Delta + F_1 z^{-1} \tag{4.7}$$

$$C_2 = E_2 A_2 \Delta + F_2 z^{-1} \tag{4.8}$$

where:

$$E_1(z^{-1}) = 1 + \sum_{n=1}^{n_{e_1}} e_{1n} z^{-n} \tag{4.9}$$

$$E_2(z^{-1}) = 1 + \sum_{n=1}^{n_{e_2}} e_{2n} z^{-n} \tag{4.10}$$

$$F_1(z^{-1}) = 1 + \sum_{n=0}^{n_{f_1}} f_{1n} z^{-n} \tag{4.11}$$

$$F_2(z^{-1}) = 1 + \sum_{n=0}^{n_{f_2}} f_{2n} z^{-n} \quad , \tag{4.12}$$

with n representing the time delay in the closed-loop, and

$n_{e_1} = n_{e_2} = n - 1 (= 0$ for *pure delay*$)$, $n_{f_1} = max(n_{a_1}, n_{e_1} - n)$,

$n_{f_2} = max(n_{a_2}, n_{e_2} - n)$.

Therefore, the model of the process reduces to

$$y_1(k+1) = \frac{B_1 E_1}{C_1} \Delta u_1(k) + \frac{F_1}{C_1} y_1(k) \tag{4.13}$$

$$y_2(k+1) = \frac{B_2 E_2}{C_2} \Delta u_2(k) + \frac{F_2}{C_2} y_2(k) \tag{4.14}$$

Minimizing the cost functions J_1 in (4.3) and J_2 in (4.4) yield the self-tuning control laws [22]:

$$\Delta u_1(k) = -\frac{F_1}{B_1 E_1 + R_1 C_1} y_1(k) \tag{4.15}$$

$$\Delta u_2(k) = -\frac{F_2}{B_2 E_2 + R_2 C_2} y_2(k) \tag{4.16}$$

where for the *pure delay case* $R_1 = \frac{r_1}{b_{11}}$ and $R_2 = \frac{r_2}{b_{21}}$, with r_1, r_2 representing the weights of the control in J_1 and J_2, respectively, and b_{11}, b_{21} representing the elements of the matrices B_1, B_2 introduced in equation (3.1) to describe the PGS dynamics. The minima of the cost functions J_1 and J_2 are reached for $R_1 = R_2 = 0$, and therefore we get the minimum variance control laws for the *self-tuning controller*.

Remarks:

- The design of the *self-tuner* includes the specification of the following model parameters:

  (*i*) model orders: n_{a_1} , n_{a_2} , n_{b_1} , n_{b_2} , n_{c_1} , n_{c_2}

  (*ii*) time delay parameter n ($n = 1$, for *pure delay*)

- Generally, larger values of the model parameters should not be chosen because the controller will take longer to tune.

- The control action, therefore, depends on the predicted values of the one step ahead model.

- The predicted values of $\hat{y}_1(k+1)$ and $\hat{y}_2(k+1)$ can be calculated from the process input and output values according to equations:

$$\hat{y}_1(k+1) = \frac{\hat{B}_1 \hat{E}_1}{\hat{C}_1} \Delta u_1(k) + \frac{\hat{F}_1}{\hat{C}_1} y_1(k) \tag{4.17}$$

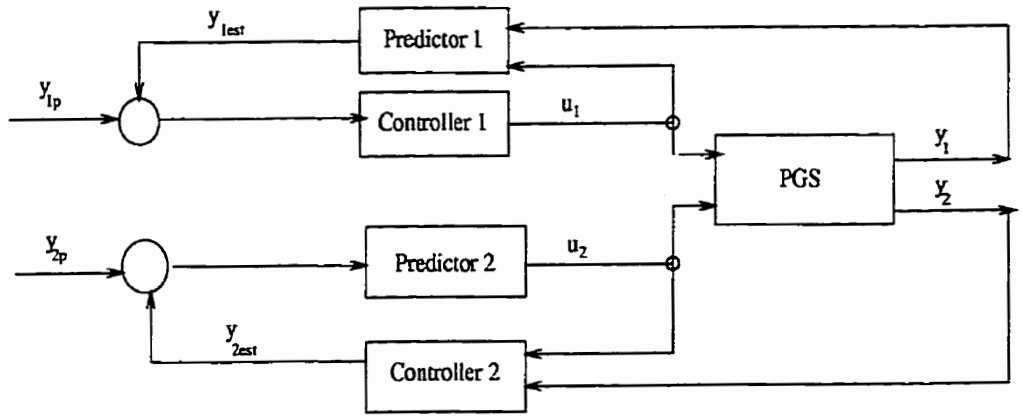


Figure 4.2: Block diagram for the predictive control scheme

$$\hat{y}_2(k+1) = \frac{\hat{B}_2\hat{E}_2}{\hat{C}_2}\Delta u_2(k) + \frac{\hat{F}_2}{\hat{C}_2}y_2(k) \tag{4.18}$$

where $\hat{B}_1$, $\hat{B}_2$, $\hat{C}_1$, $\hat{C}_2$, $\hat{E}_1$, $\hat{E}_2$, $\hat{F}_1$, $\hat{F}_2$ are estimated by the RLS algorithm.

- By solving the Diophantine equations (4.7)-(4.8) using the experimental input-output data set for this model ($(u_1$, $y_1)$, $(u_2$, $y_2)$ interactions). we get the control law for the self-tuners in each loop.

The algorithm is similar to the one step ahead control algorithm, where

$$y_1(k+1) = y_1^*(k+1)$$

$$y_2(k+1) = y_2^*(k+1)$$

are fixed at their target values. The explicit identification procedure has the disadvantage that the coefficients E_1 , E_2, F_1 , F_2 have to be calculated on line at each sampling interval from the Diophantine equations (4.7)-(4.8). Solution of the Diophantine equation can be avoided if the coefficients of the polynomials E_1 , E_2, F_1 , F_2 are identified directly. If we introduce the variables $\hat{\Phi}_1$ and $\hat{\Phi}_2$ defined as: $\hat{\Phi}_1(k+1) = \hat{y}_1(k+1) + R_1\Delta u_1(k)$, and $\hat{\Phi}_2(k+1) = \hat{y}_2(k+1) + R_2\Delta u_2(k)$, the re-

gression model for identifying the implicit parameters E_1 , E_2, F_1 , F_2 is obtained as follows

$$\hat{\Phi}_1(k+1) = \frac{F_1}{C_1} y_1(k) + (\frac{B_1 E_1}{C_1} + R_1)\Delta u_1(k) \tag{4.19}$$

$$\hat{\Phi}_2(k+1) = \frac{F_2}{C_2} y_2(k) + (\frac{B_2 E_2}{C_2} + R_2)\Delta u_2(k) \tag{4.20}$$

The above forms still cause difficulties to estimate these variables by solving the highly complex Diophantine equations at each sampling time. Building a *PI* self-tuning adaptive controller based on this procedure is not computationally efficient. In practice we can simplify this procedure considering the variables $\hat{\Phi}_i$, as polynomials of first-order in the following form:

$$\hat{\Phi}_i(k+1) = \hat{\alpha}_{0i} y_i(k) + \hat{\alpha}_{1i} y_i(k-1) + \hat{\beta}_{0i}\Delta u_i(k), \; i = 1, 2 \tag{4.21}$$

where $\alpha_i(z^{-1}) = F_i(z^{-1})$, and $\beta_i(z^{-1})$=$G_i(z^{-1}) + R_i$ are obtained by the parameter identification expression given in equation (3.1) or estimated directly using a *RLS* procedure presented in detail in Section 4.1.4. By adding supplementary terms (second-order or higher) to the equation (4.21) we can build a *Proportional Integral Derivative (PID) self-tuning controller* or more other sofisticated control laws.

4.1.3 PI Self-Tuning Controller

Our goal is to design a standard *PI self-tuning* controller for the *decoupled PGS model with pure delay* in the predictive form given by equations (4.1)-(4.2). The objective of this approach is to build a standard controller to be used easily in practice and to have a reference model so as to compare the results obtained by applying this algorithm with the other algorithms developed in this chapter and in Chapters 5-7.

The equations (4.19) and (4.20) for the *PI self-tuning* controller could be represented in the following predictive forms:

$$\hat{\Phi}_1(k+1) = \hat{\alpha}_{01}y_1(k) + \hat{\alpha}_{11}y_1(k-1) + \hat{\beta}_{01}\Delta u_1(k) \tag{4.22}$$

$$\hat{\Phi}_2(k+1) = \hat{\alpha}_{02}y_2(k) + \hat{\alpha}_{12}y_2(k-1) + \hat{\beta}_{02}\Delta u_2(k) \tag{4.23}$$

where $\hat{\alpha}_{ij}$, $\hat{\beta}_{ij}$ appear in equation (4.21), and are estimated directly using a *RLS* procedure (Section 4.1.4). The *PI control law* then takes the form [22]:

$$\Delta u_1(k) = -\frac{1}{\hat{\beta}_{01}}[\hat{\alpha}_{01}y_1(k) + \hat{\alpha}_{11}y_1(k-1)] \tag{4.24}$$

$$\Delta u_2(k) = -\frac{1}{\hat{\beta}_{02}}[\hat{\alpha}_{02}y_2(k) + \hat{\alpha}_{12}y_2(k-1)] \tag{4.25}$$

This approach is implicit because the predictive model parameters appear directly in the control law. As the number of parameters increases the initial estimates become increasingly important and poor estimates may result in an extended period of poor performance and even lead to instability [22].

4.1.4 The RLS Procedure

To estimate the parameters in the *PI self-tuning* controllers, one may use the *RLS algorithm.* This algorithm identifies the model parameters using the experimental input-output data set. In order to set up the regressor vectors the predictive equations (4.22)-(4.23) are rewritten at the sampling time k as:

$$\hat{\Phi}_1(k) = \hat{\alpha}_{01}y_1(k-1) + \hat{\alpha}_{11}y_1(k-2) + \hat{\beta}_{01}\Delta u_1(k-1) \tag{4.26}$$

$$\hat{\Phi}_2(k) = \hat{\alpha}_{02}y_2(k-1) + \hat{\alpha}_{12}y_2(k-2) + \hat{\beta}_{02}\Delta u_2(k-1) \tag{4.27}$$

Using the vector notation these equations may be written in the form:

$$\hat{\Phi}_1(k) = \Psi_1^T(k-1)\Theta_1(k-1) \tag{4.28}$$

$$\hat{\Phi}_2(k) = \Psi_2^T(k-1)\Theta_2(k-1) \tag{4.29}$$

where the regressor vectors Ψ_1, Ψ_2 are given by

$$\Psi_1^T(k-1) = [y_1(k-1)\ \ y_1(k-2)\ \ \Delta u_1(k-1)] \tag{4.30}$$

$$\Psi_2^T(k-1) = [y_2(k-1)\ \ y_2(k-2)\ \ \Delta u_2(k-1)] \tag{4.31}$$

and parameter vectors Θ_1, Θ_2 have the form

$$\hat{\Theta}_1^T(k-1) = [\hat{\alpha}_{01}\ \ \hat{\alpha}_{11}\ \ \hat{\beta}_{01}] \tag{4.32}$$

$$\hat{\Theta}_2^T(k-1) = [\hat{\alpha}_{02}\ \ \hat{\alpha}_{12}\ \ \hat{\beta}_{02}] \tag{4.33}$$

The updates for the parameter vectors Θ_1, Θ_2 are performed according to [22]

$$\hat{\Theta}_i(k) = \hat{\Theta}_i(k-1) + K_i(k)[\Delta y_i(k) - \Delta\hat{y}_i(k)] \tag{4.34}$$

$$K_i(k) = \frac{P_i(k-1)\Psi_i(k-1)}{\Lambda_i + \Psi_i^T(k-1)P_i(k-1)\Psi_i(k-1)},\ i = 1,\ 2 \tag{4.35}$$

$$P_i(k) = \frac{1}{\Lambda_i}[I - K_i(k)\Psi_i(k-1)]P_i(k-1) \tag{4.36}$$

where K_i and P_i, $i = 1, 2$ represent the Kalman filter gains and covariance matrices of the estimation errors, respectively. $P_i's$ are positive definite measure of the parameter estimate errors and tend to decrease as k increases. The above equation requires an initial estimate of the parameter vectors $\hat{\Theta}_i$ and P_i. Usually $P_i(0)$ is chosen as a diagonal matrix. Large diagonal values (e.g.10^4 or higher) indicate that

$V_{bias}/[F]$	t_r[s]	t_i[s]	%OS	ϵ	Convergence
V_{bias}	80	35	2 [Volts]	0.0015 [Volts]	very good
$[F]$	60	15	0.02 [%]	0.00002 [%]	very good

Table 4.1: Closed-loop system performance of the PI self-tuning control strategy.

the confidence in $\hat{\Theta}_i(0)$ is poor and will cause initially rapid changes in $\hat{\Theta}_i$. If these matrices are not well initialized it is possible a degradation of the system performance in the closed-loop. Small values indicate that $\hat{\Theta}_i(0)$ is a good estimate and will cause slow changes in $\hat{\Theta}_i$.

4.1.5 Simulation Results

We design in this section a standard *PI self-tuning* adaptive controller based on minimization of a quadratic cost function (4.3)-(4.4) (minimum variance regulation), capable of performing fast transient and good tracking error for the PGS in closed-loop. Also, the algorithm can take into account the physical constraints due to the rate of change in the input variables *throttle valve* and *RF power* actuators by setting minimum and maximum bounds for these inputs: $0 \leq u_1 \leq 100$ (% opening), and $0 \leq u_2 \leq 5000$ [Watts], such that each time when u_1 and u_2 reach the minimum or maximum values then $u_i = u_{imax}$ or $u_i = u_{imin}$, until u_i, i=1, 2, given by the algorithm take values between the input bounds. The algorithm is developed for the *decoupled PGS model with pure delay* depicted in Figures 3.8-3.9 to serve as comparison for the other advanced algorithms which are developed later in the thesis in neurocontrollers and robust controllers chapters. To estimate the parameters of the *PI self-tuning* controller we use the standard *RLS estimation algorithm* initialized by the following values of the parameters:

$$R_1 = R_2 = 2.5,\ P_1(0) = \begin{bmatrix} 7 & 0 \\ 0 & 7 \end{bmatrix},\ P_2(0) = \begin{bmatrix} 500 & 0 \\ 0 & 500 \end{bmatrix},$$

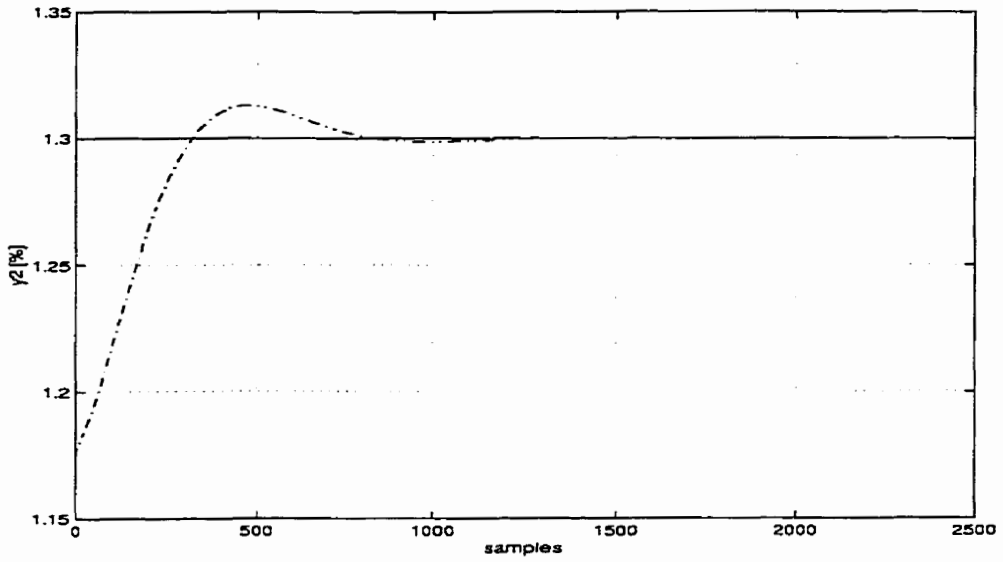


Figure 4.3: The performance of the *PI self-tuning* controller on the *fluorine concentration* $[F]$ for the closed-loop system.
Legend: The dot line designates the evolution of $[F]$ and the solid line designates the set point.

$\Theta_1(0) = \begin{bmatrix} 0.5 & 1 & 2500 \end{bmatrix}$, $\Theta_2(0) = \begin{bmatrix} 1 & 0.5 & -0.1 \end{bmatrix}$, and $\Lambda_{1,2} = 0.98$.

These values of the parameters were obtained by trials, monitoring for each trial the system performance in closed-loop (fast transient and good tracking error). Analyzing both the open-loop and the closed-loop simulations revealed that the algorithm works well and provides satisfactory results. The results of the simulations for the closed-loop PGS system without changes in the set point are shown in Figures 4.3-4.6, and with changes in the set points are given in Figures 4.7-4.10. These results reveal good tracking error and good convergence of the *self-tuning PI* algorithm for the both cases. The performance comparison of the results for the closed-loop and open-loop system are presented in Figures 4.11-4.12. Also, we synthesize the closed-loop system performance of this control strategy, namely the settling time (t_r), rise time (t_i), overshoot ($\%OS$), steady-state error (ϵ) and the convergence of the algorithm for V_{bias} and $[F]$ in Table 4.1.

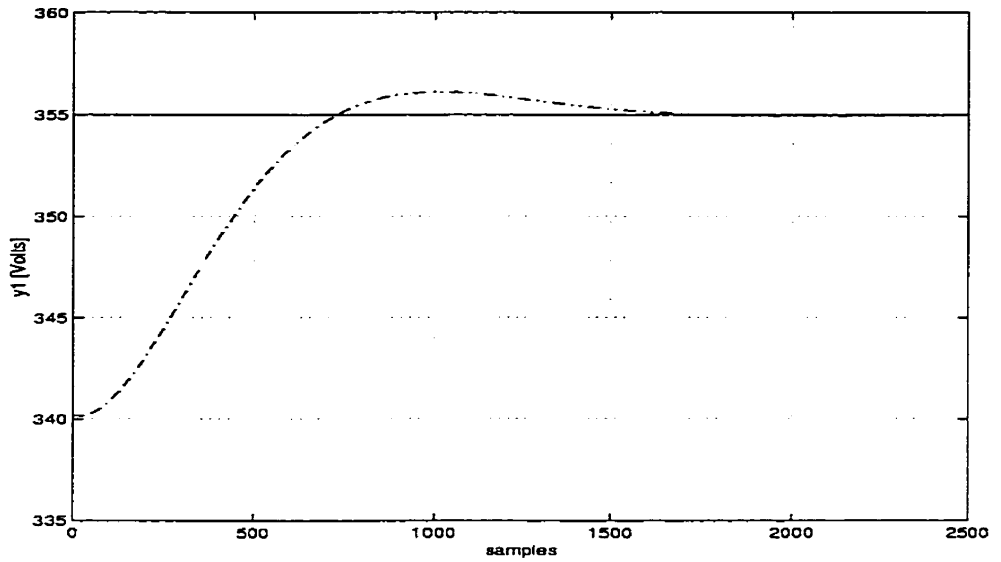


Figure 4.4: The performance of the *PI self-tuning* controller on the *dc* V_{bias} for the closed-loop system.
Legend: The dot line designates the evolution of the V_{bias} and the solid line designates the set point.

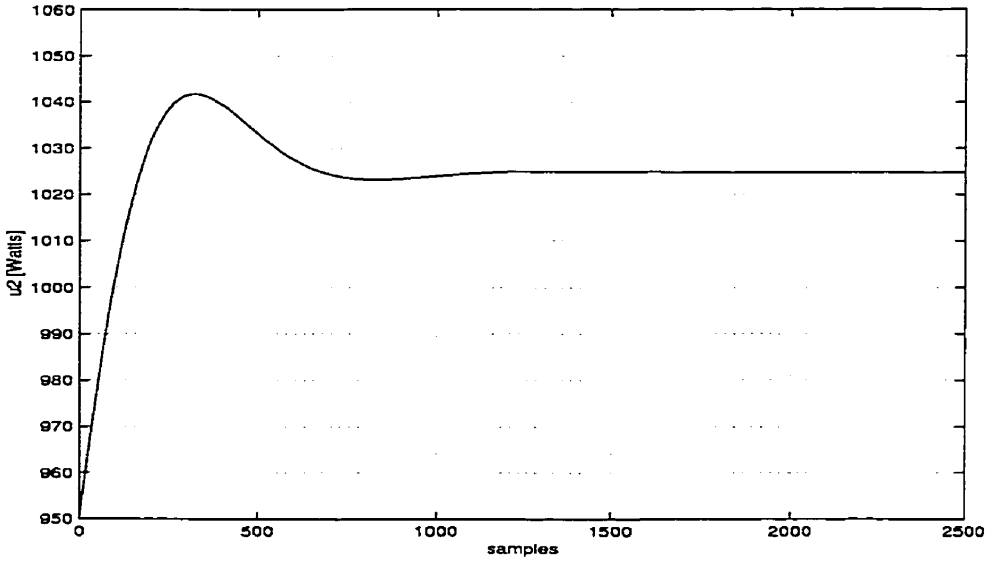


Figure 4.5: The performance of the *PI self-tuning controller* on the *RF power* for the closed-loop system.

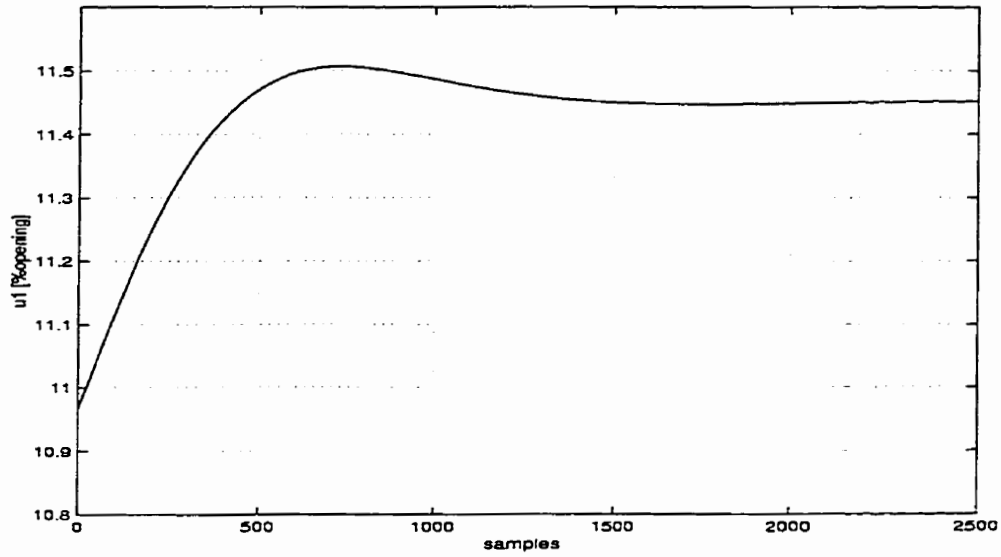


Figure 4.6: The performance of the *PI self-tuning* controller on the *throttle position* for the closed-loop system.

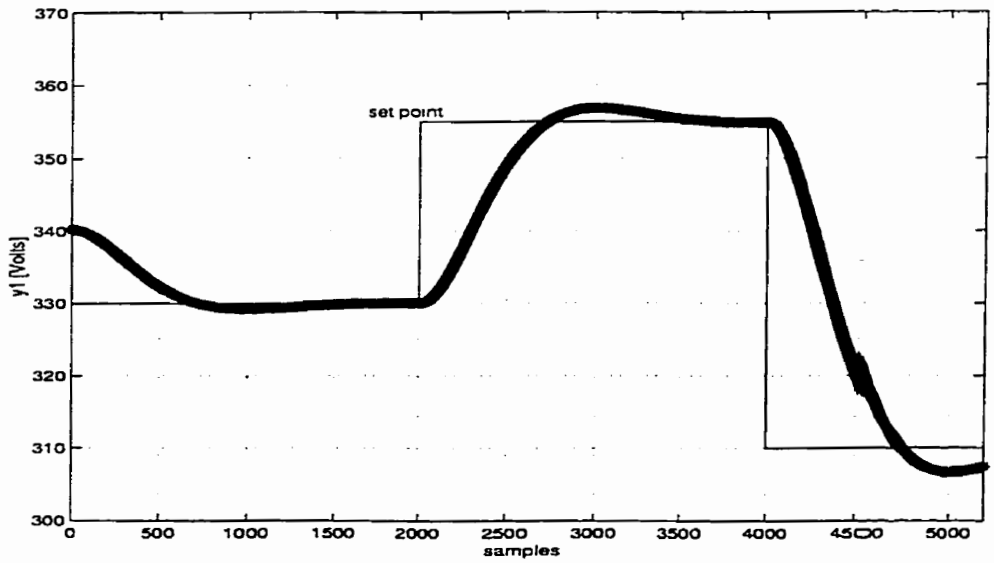


Figure 4.7: The performance of the *PI self-tuning* controller on the *dc* V_{bias} for the closed-loop system with changes in the set points.

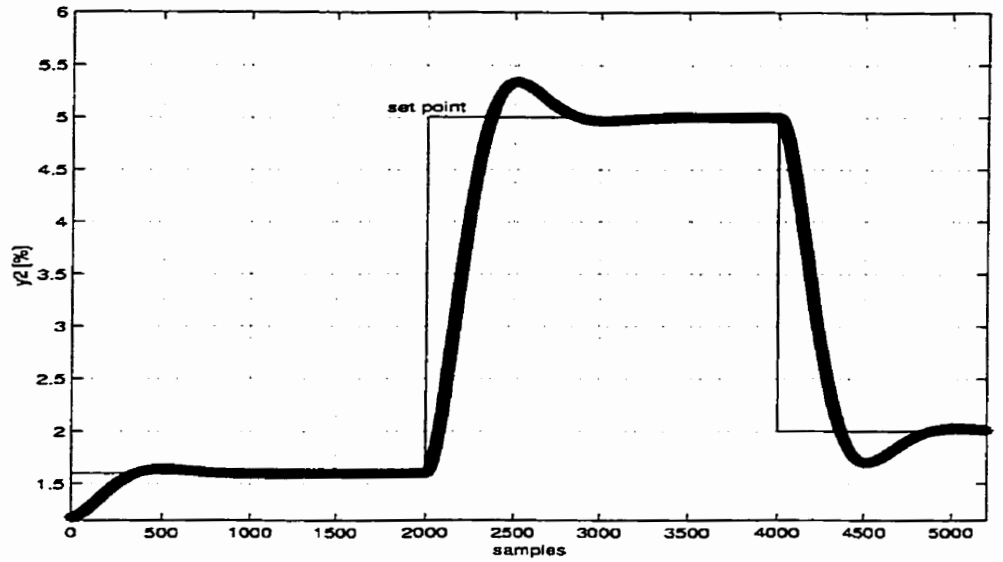


Figure 4.8: The performance of the *PI self-tuning* controller on the $[F]$ for the closed-loop system with changes in the set points.

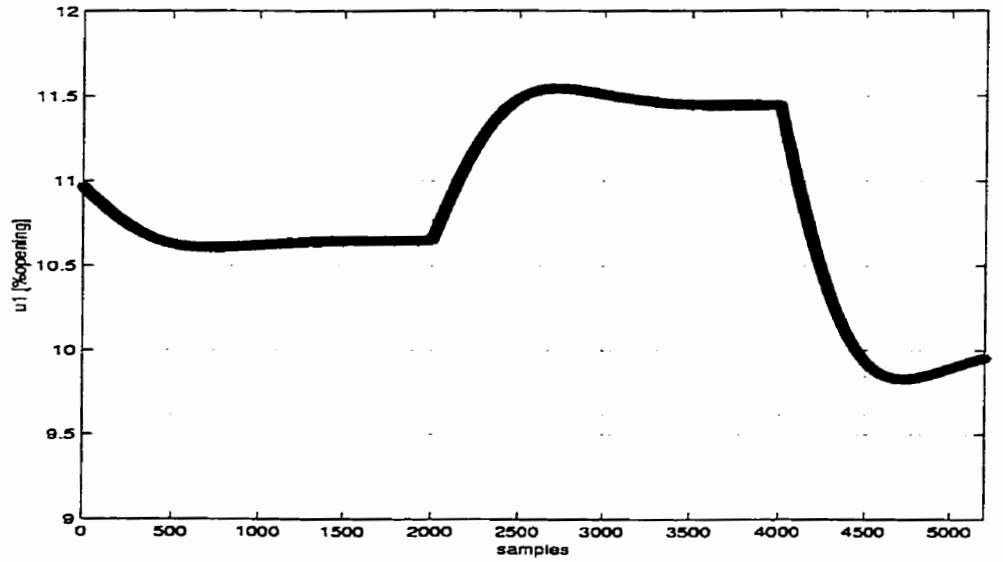


Figure 4.9: The performance of the *PI self-tuning* controller on the *throttle position* for the closed-loop system. with changes in the set points.

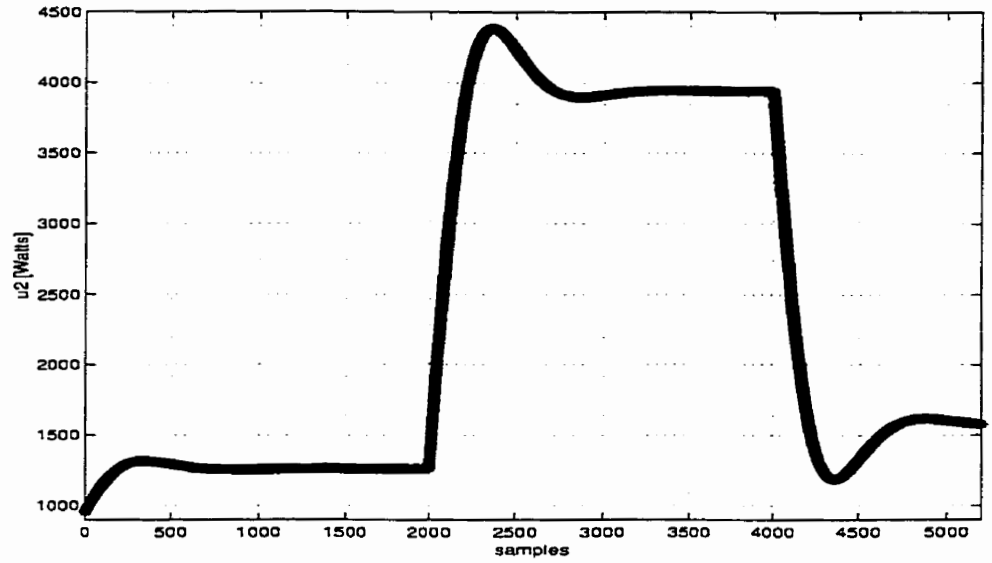


Figure 4.10: The performance of the *PI self-tuning* controller on the *RF power* for the closed-loop system with changes in the set points.

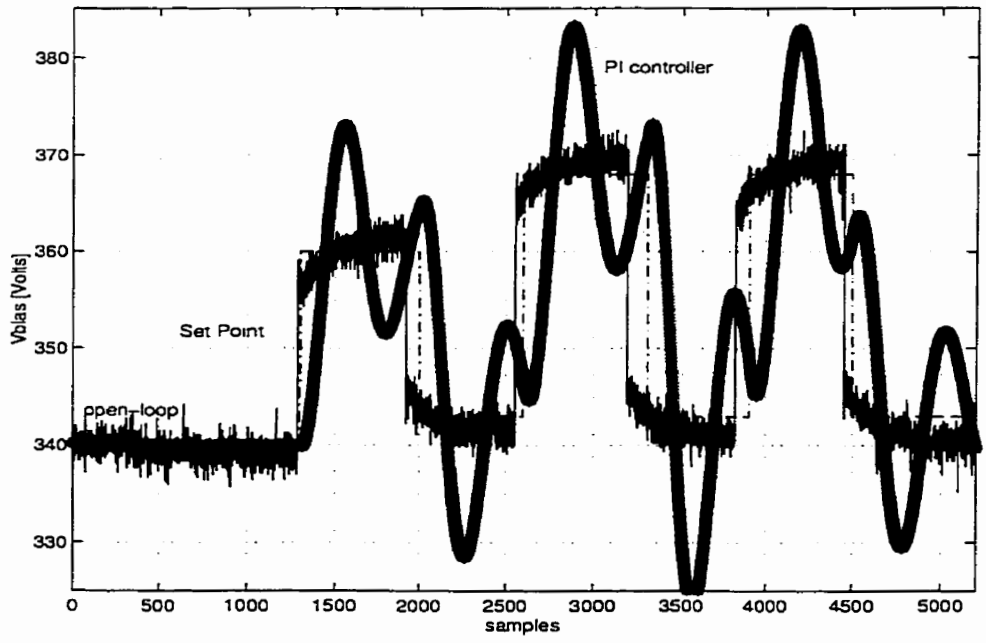


Figure 4.11: The performance comparison results for the V_{bias} for the open-loop and closed-loop system with the *PI self-tuning* controller.

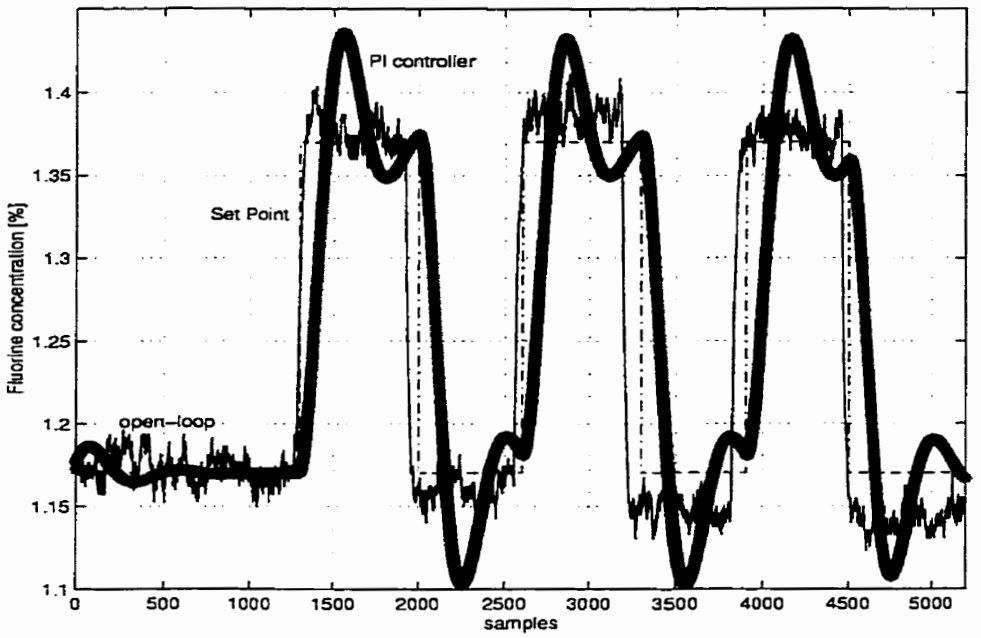


Figure 4.12: The performance comparison results for the *fluorine concentration* $[F]$ for the open-loop and closed-loop system with the *PI self-tuning* controller.

4.2 Real-Time Reactive Ion Etching Multi-Input Multi-Output Control based on the Linear Quadratic Gaussian Scheme

4.2.1 Linear Quadratic Gaussian Controller (LQG)

In this section, we propose to use *Linear Quadratic Gaussian (LQG)* strategies [29], due to their conceptual simplicity and their ability to handle systematically coupled dynamics. Given the model of the plasma generation subsystem in the state-space representation form, an *Linear Quadratic Gaussian (LQG)* controller is designed and tuned to meet certain desired performance specifications. We use in this approach the linear plant model represented in a state-space representation by the equations (3.22)-(3.23) where the process and the measurement noise ($w(k)$ and $v(k)$, respectively) is

assumed to be zero-mean Gaussian and uncorrelated. In order to obtain a unique solution to the design problem the Reactive Ion Etching system (RIE) must be both controllable and observable.

To eliminate steady-state errors when the set points are constant we add an integral control action represented by the accumulator block as shown in Figure 4.13. The accumulator dynamics is described in the state space form by the discrete equation

$$q(k+1) = q(k) + r(k) - y(k) = q(k) + e(k) \tag{4.37}$$

where $q(k) = \begin{bmatrix} q_1(k) & q_2(k) \end{bmatrix}^T$ represents the state vector of the accumulator. The input to the accumulator is the error signals $e(k)$ (defined as the difference between the reference signal $r(k)$ and the measured process variables $y(k)$). Once the feedback control is designed such that $q(k)$ converges to a steady state value then $y(k)$ must converge to the reference input $r(k)$ (since $e(k) \rightarrow 0$). Defining the extended or augmented state vector by

$$z(k) = \begin{bmatrix} x(k) \\ q(k) \end{bmatrix} \tag{4.38}$$

the full order augmented dynamics of the PGS in closed-loop can be written in state-space representation as

$$z(k+1) = Az(k) + Bu(k) + G_w w(k) + G_v v(k) + Er(k) \tag{4.39}$$

$$y(k) = Cz(k) + Du(k) + v(k) \tag{4.40}$$

with appropriate definitions for the matrices A, B, C, D, and E:

$$A = \begin{bmatrix} A_G \\ -C_G \end{bmatrix}, B = \begin{bmatrix} B_G \\ -D_G \end{bmatrix}, C = \begin{bmatrix} C_G & O_{2\times 2} \end{bmatrix}, D = D_G,$$

$$E = \begin{bmatrix} O_{4\times 2} \\ I_2 \end{bmatrix}, G_v = \begin{bmatrix} O_{4\times 2} \\ -I_2 \end{bmatrix}, G_w = \begin{bmatrix} I_4 \\ O_{2\times 2} \end{bmatrix}$$

where from our experimental data set used in identification process of the PGS the following values for the matrices $A_G, B_G, C_G, and\ D_G$ were obtained:

$$A_G = \begin{bmatrix} 0 & 0.3846 & 0 & 0 \\ 1 & 0.4571 & 0 & 0 \\ 0 & 0 & 0 & 0.0927 \\ 0 & 0 & 1 & 0.8511 \end{bmatrix}, B_G = \begin{bmatrix} 0.4108 & 0.0654 \\ 2.7029 & -0.0604 \\ -0.0005 & 0 \\ -0.0035 & 0.0001 \end{bmatrix}$$

$$C_G = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, D_G = \begin{bmatrix} -1.0683 & 0.175 \\ -0.0059 & 0.00035 \end{bmatrix}.$$

Using the separation principle [28], [29], [30], the LQG controller is now designed in two steps:

(i) *Linear Quadratic Estimator design (LQE) (Kalman filter design)*, and

(ii) *Linear Quadratic Regulator design (LQR) (Full-state feedback design)*

4.2.1.1 *Linear Quadratic Estimator design (LQE)*

In this step an estimator for the process state vector $x(k)$, denoted by $\hat{x}(k)$, is designed since only $y(k)$ is assumed to be measurable. The algorithm computes the estimate of the PGS state, $\hat{x}(k)$, by employing a steady-state Kalman filter [29] with

the dynamic state equation governed by

$$\hat{x}(k+1) = A_G\hat{x}(k) + B_G u(k) + K_e[y(k) - C_G A_G\hat{x}(k) - C_G B_G u(k)] \tag{4.41}$$

where K_e is the Kalman gain matrix that is given by

$$K_e = A_G[S - SC_G^T(C_G S C_G^T + R_v)^{-1} C_G S]C_G^T R_v^{-1} \tag{4.42}$$

where S is the unique positive definite solution of the algebraic Riccati equation [29]

$$S = A_G[S - SC_G(C_G S C_G^T + R_v)^{-1} C_G S]A_G^T + Q_w \tag{4.43}$$

The Kalman filter generates a state estimate that minimizes the variance of the estimation error where R_v and Q_w are the covariance matrices for the noise processes $v(k)$ and $w(k)$, respectively. To achieve a "good" dynamic performance, the covariance matrices R_v and Q_w are chosen as diagonal matrices such that the estimator bandwidth is approximately four times the bandwidth of the closed-loop system [28], [29], [30].

4.2.1.2 *Linear Quadratic Regulator design (LQR)*

In this step a feedback gain matrix K is computed such that when

$$u(k) = -Kz(k) \tag{4.44}$$

is applied to the full order system (4.39), the resulting closed-loop system is stable, i.e., the eigenvalues of the matrix $(A - BK)$ are all inside the unit circle. Since the state of the accumulator, $q(k)$, is available for use in the controller, the control may

now be implemented as follows

$$u(k) = -\begin{bmatrix} K_1 & K_2 \end{bmatrix} \begin{bmatrix} \hat{x}(k) \\ q(k) \end{bmatrix} = -K_1\hat{x}(k) - K_2 q(k) = -Kz(k) \tag{4.45}$$

To obtain a stabilizing state feedback gain matrix, we use the deterministic formulation of linear quadratic optimal control (LQR), in which the process noise $w(k)$ and reference $r(k)$ are assumed to be zero. The control is selected to minimize the following quadratic function

$$J = \sum_{k=0}^{\infty} [z(k)^T Q z(k) + u(k)^T R u(k)] \tag{4.46}$$

Consequently, the feedback gain matrix K is given by [28], [29], [31]

$$K = (R + B^T P B)^{-1} B^T P A \tag{4.47}$$

where P is a solution to the algebraic Riccati equation [29]

$$P = A^T [P - PB(B^T P B + R)^{-1} B^T P] A + C^T Q C \tag{4.48}$$

Under full controllability and observability of (A, B, C), a unique positive definite solution to P does exist.

The weighting matrices Q and R are chosen to be diagonal so that increasing a nonzero term in Q has the effect of reducing the rate of the response in the corresponding process variable, and increasing a nonzero term in R makes the corresponding control input less aggressive. For practical reasons we choose Q and R such that the closed-loop system bandwidth is approximately one tenth of the sampling frequency [28].

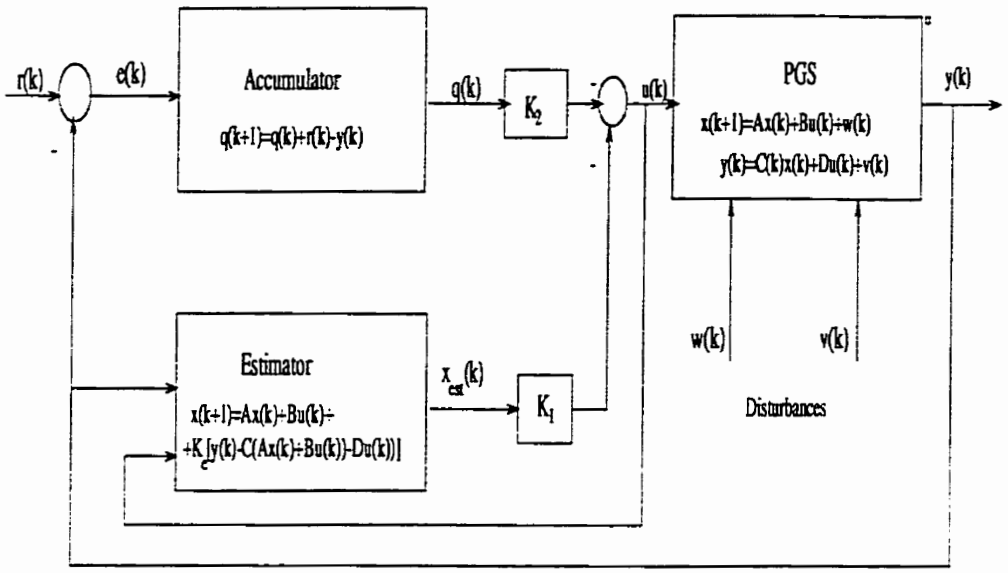


Figure 4.13: Block diagram of the LQG controller

4.2.2 Simulation Results

In this section we design a real-time *MIMO LQG* control strategy capable to perform in stochastic environment fast transient and good tracking error for the PGS in closed-loop. For the simulations we choose $Q_w = 0.001 I_4$ and $R_v = 3.5 I_2$ values obtained by trials and monitoring for each trial the closed-loop system performance (fast transient and good tracking error). The solution to the *Linear Quadratic Estimation* problem (LQE) has the following form:

$$K_e = \begin{bmatrix} 0.0118 & 0 \\ 0.444 & 0 \\ 0 & 0.1619 \\ 0 & 0.3965 \end{bmatrix}, \quad S = \begin{bmatrix} 0.1657 & 0.1235 & 0 & 0 \\ 0.1235 & 0.4650 & 0 & 0 \\ 0 & 0 & 1.2758 & 2.6822 \\ 0 & 0 & 2.6822 & 6.5690 \end{bmatrix} \tag{4.49}$$

It can be shown easily that the matrices (A, B, C) of the RIE extended system satisfy the full controllability and observability criteria. Therefore, the LQR problem has a unique positive definite solution P. For example by setting $Q = 12I_6$ and $R = 100I_2$ we get:

$$K = \begin{bmatrix} 0.463 & 0.521 & -2.075 & -2.08 & 0.082 & -0.04 \\ 0.19 & 0.479 & 16.17 & 16.21 & 0.117 & 0.314 \end{bmatrix} \quad (4.50)$$

$$P = 10^6 * \begin{bmatrix} 0.00011 & 0.00017 & 0.0188 & 0.01891 & 0.000035 & 0.000365 \\ 0.00017 & 0.00041 & 0.01808 & 0.01811 & 0.000095 & 0.00035 \\ 0.0188 & 0.01808 & 9.4785 & 9.4939 & 0.0007 & 0.18318 \\ 0.1891 & 0.01811 & 9.4939 & 9.51017 & 0.0007 & 0.18379 \\ 0.000035 & 0.000095 & 0.0007 & 0.00070 & 0.000036 & 0.000013 \\ 0.000365 & 0.00035 & 0.1831 & 0.18379 & 0.000013 & 0.003667 \end{bmatrix} \quad (4.51)$$

Regulation of the process variables in the RIE system requires a multivariable controller that can vary two inputs (manipulated variables) simultaneously to track setpoints for the two process variables. Also, the measurements of the plasma process variables, particularly *fluorine concentration* are contaminated by the noise measurement. The sensor noise is very close to a white, Gaussian one. For these reasons, we have selected the *Linear Quadratic Gaussian* (LQG) as a suitable multivariable control strategy to be applied to the RIE system. This control strategy facilitates the use of *in-situ* sensing to modify inputs in real-time, providing considerable information about the process and wafer state. The closed-loop system performance of this control strategy, namely the settling time (t_r), rise time (t_i), overshoot ($\%OS$), steady-state error (ϵ) and the convergence of the algorithm for V_{bias} and $[F]$ are presented in Table 4.2. The simulation results for the closed-loop system with changes in the set points are presented in Figures 4.14-4.17. For constant set points the evolution of the control

$V_{bias}/[F]$	t_r[s]	t_i[s]	%OS	ϵ	Convergence
V_{bias}	15	5	45 [Volts]	0.0014 [Volts]	very good
$[F]$	75	-	-	0.00001 [%]	very good

Table 4.2: The closed-loop system performance of the LQG control strategy.

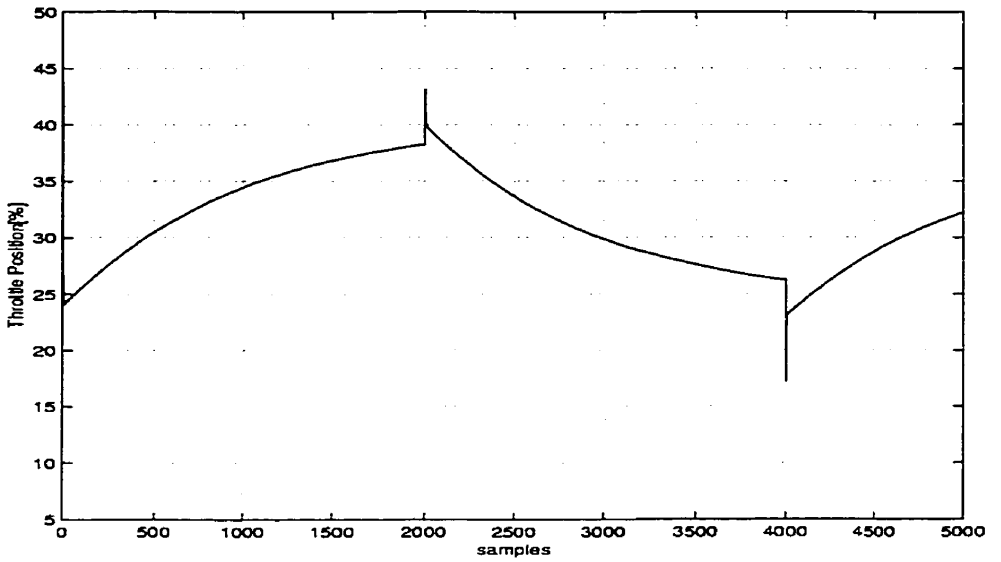


Figure 4.14: The performance of the LQG controller on the *throttle position* of the closed-loop system with changes in the set points.

and outputs of the closed-loop system are presented in Figures 4.18-4.21. The effects of the noise on the control and the outputs are shown in Figures 4.22-4.23, where the weighting matrices are tuned with the values: $Q = 12I_6$, $R = 100I_2$, $Q_w = 0.001I_4$, and $R_v = 3.5I_2$.

By analyzing the above simulations we can conclude that very good results are obtained in the deterministic case. Moreover, the control strategy has also performed satisfactorily in the stochastic environment. The figures reveal the significant impact of the noise in $[F]$ level and in the V_{bias} level during the transient and the steady-state evolution. The LQG algorithm has good convergence properties and can be used in real-time control since it is computationally efficient due its low complexity, and the small number of operations involved in the computation process.

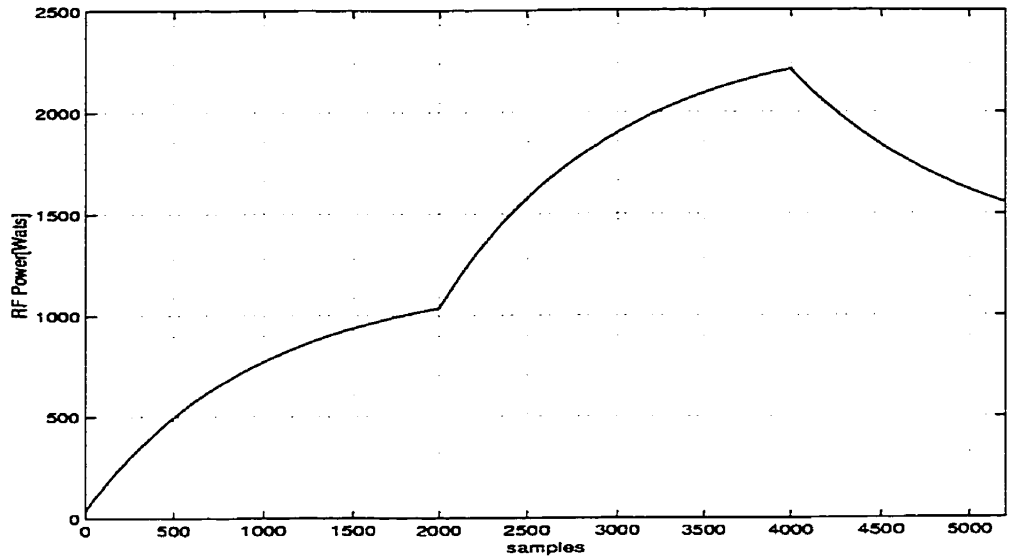


Figure 4.15: The performance of the LQG controller on the *power* of the closed-loop system with changes in the set points.

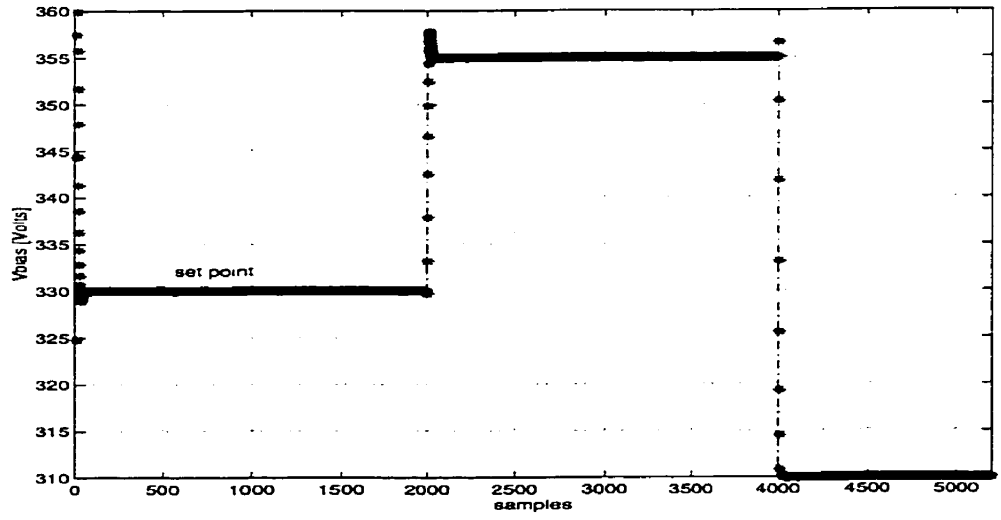


Figure 4.16: The performance of the LQG controller *on* V_{bias} of the closed-loop system with changes in the set points.
Legend: The star denotes the evolution of V_{bias} and the solid line is the V_{bias} set points.

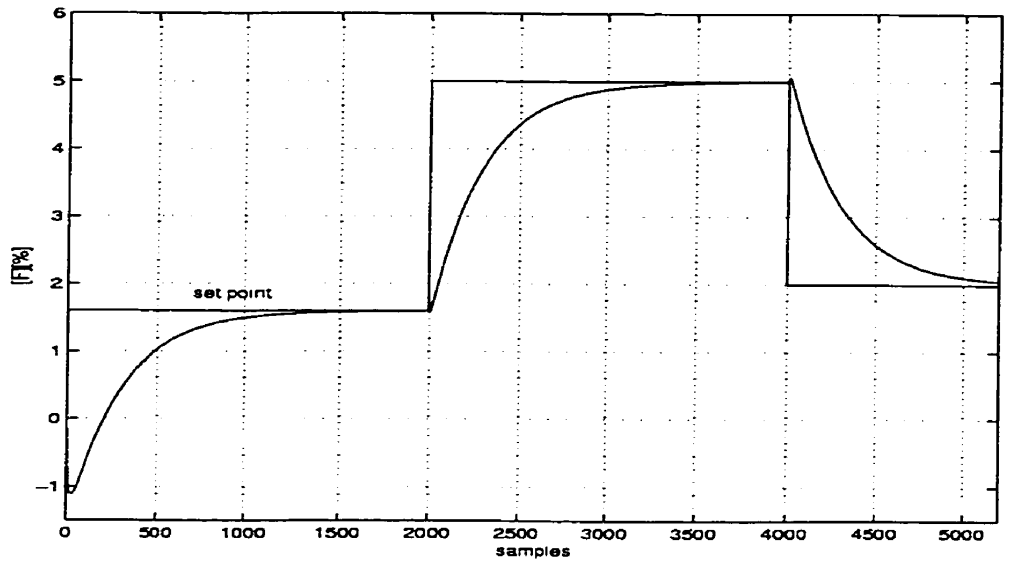


Figure 4.17: The performance of the LQG controller on $[F]$ of closed-loop system with changes in the set points.
Legend: The solid line curve designates the evolution of $[F]$

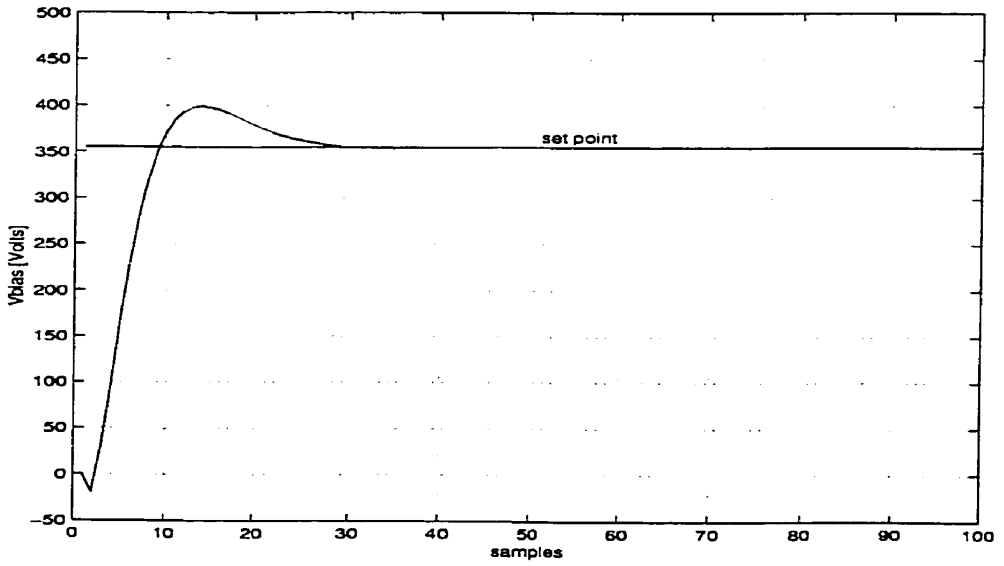


Figure 4.18: The performance of the LQG controller on V_{bias} of the closed-loop system.

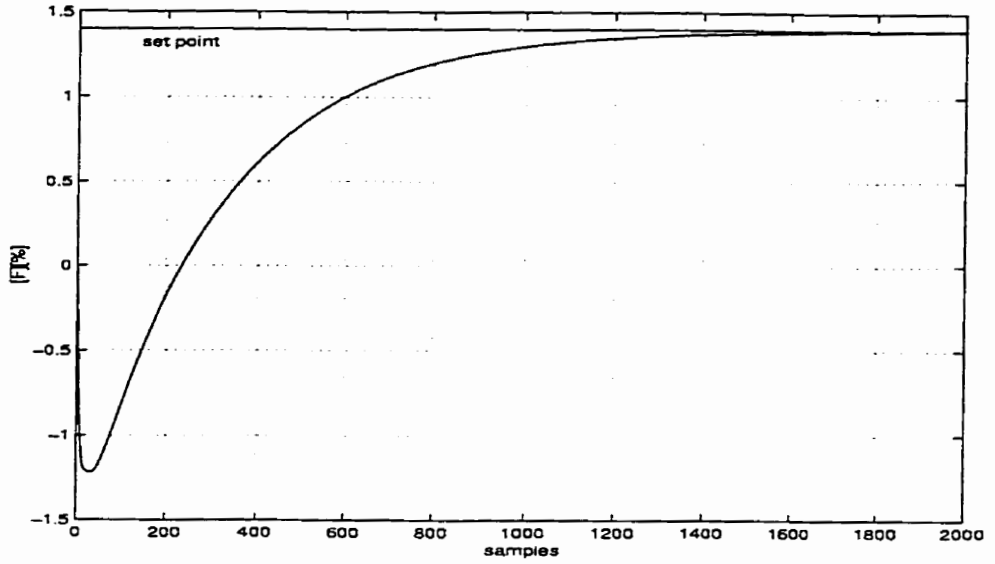


Figure 4.19: The performance of the LQG controller on $[F]$ of the closed-loop system.

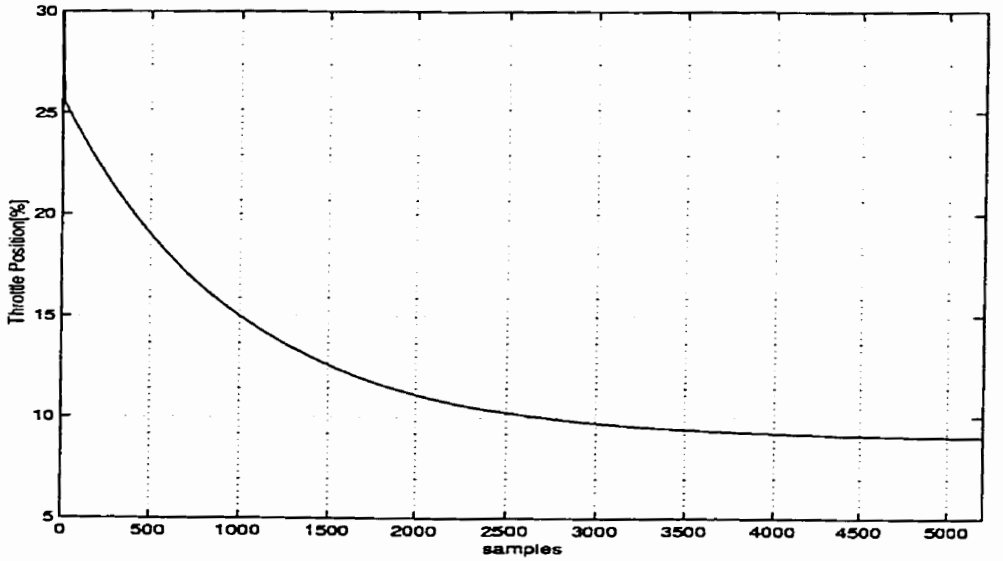


Figure 4.20: The performance of the LQG controller on the *throttle position* of the closed-loop system.

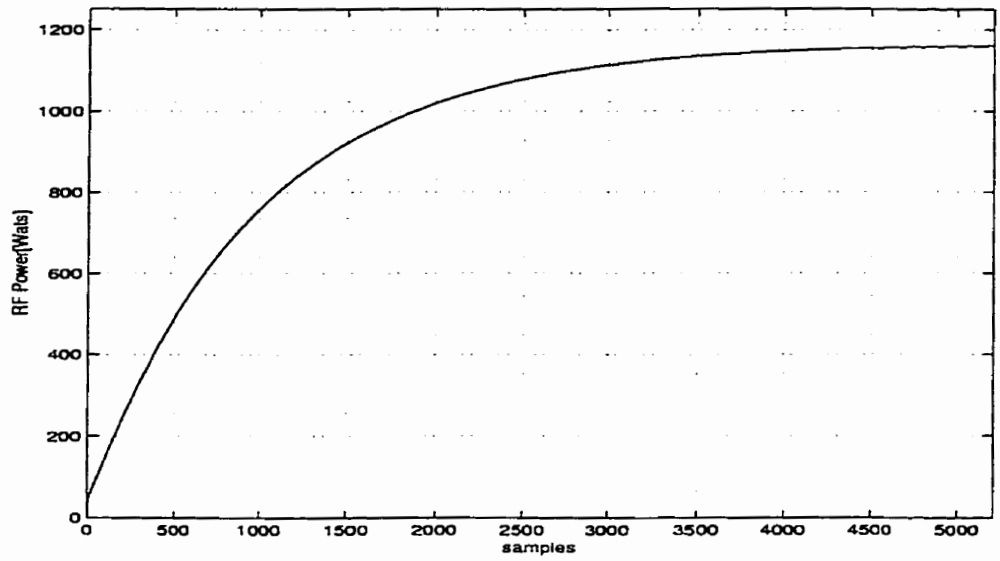


Figure 4.21: The performance of the LQG controller on the *power* of the closed-loop system.

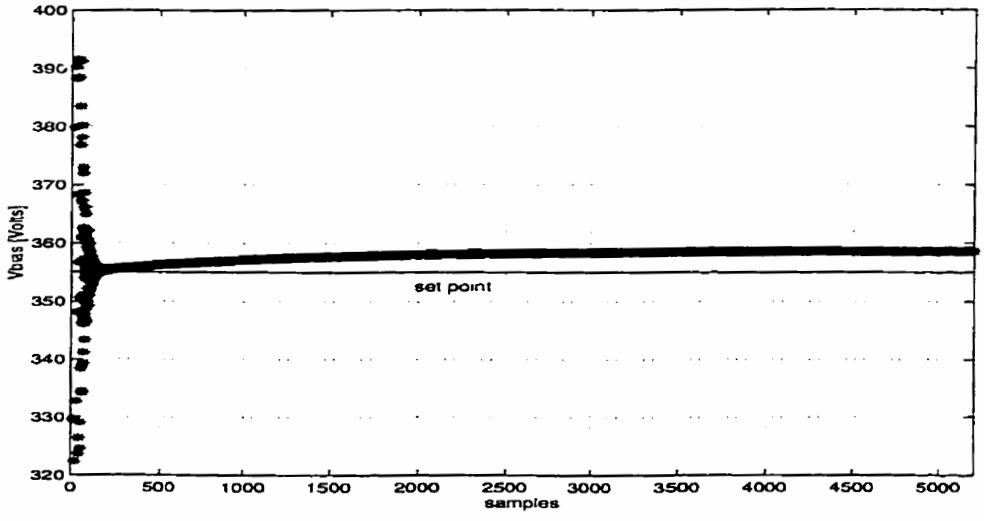


Figure 4.22: The performance of the LQG controller on the *dc* V_{bias} of the closed-loop system contaminated by noise.

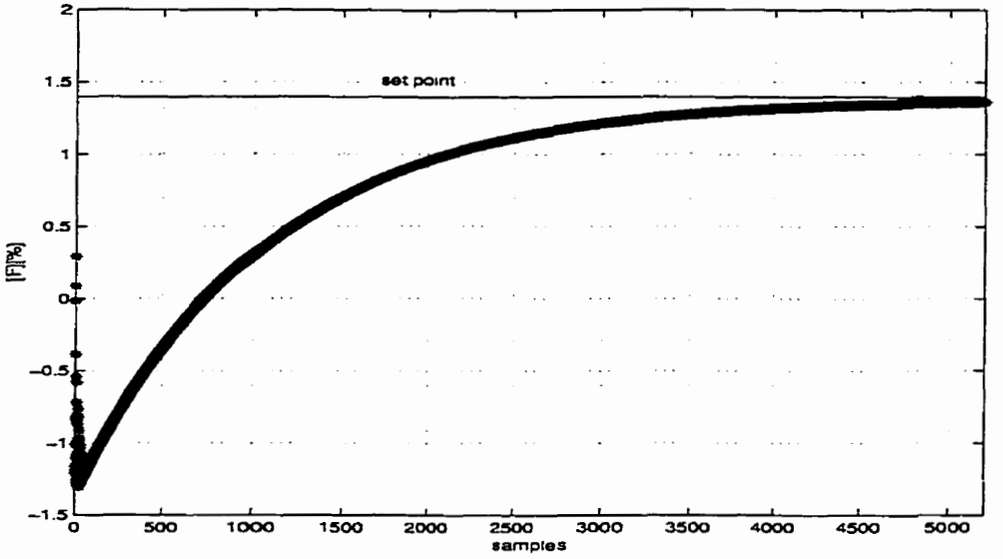


Figure 4.23: The performance of the LQG controller on $[F]$ of the closed-loop system contaminated by noise.

4.3 Coupled Real-Time Multi-Input Multi-Output Stochastic Adaptive Control Strategy of the PGS System

In this section we consider a real-time MIMO stochastic adaptive control strategy developed as a combination of the system identification scheme based on the minimum variance principle approach and the control law strategy derived from minimization of a per-interval performance index. The high efficiency and high performance of this control strategy is utilized for developing our real-time *MIMO* control scheme for the RIE system. This approach, as demonstrated below, results in superior accuracy and performance. We believe that this control strategy is suitable for implementation in our *MIMO* RIE system expressed in the state-space representation form. The potential advantages of this control strategy when applied to RIE system are as follows:

(*i*) process recipes are defined in terms of the desired physical and chemical processes occurring in the plasma generation subsystem and on the wafer.

(*ii*) the stochastic adaptive controller can compensate for progressive drifts in real-time, namely changes to components of the RIE system such as the mass flow controllers, throttle valve, pumps, and variations in loading effect.

(*iii*) the stochastic adaptive controller can also compensate for drift which occurs during the process, in principle reducing process variability. Drifts during the process can be caused by physical or chemical changes to the reactor during deposition due to wall heating or deposition.

(*iv*) failure detection and fault classification is enhanced because *in-situ* sensor-based stochastic adaptive controller provides considerable information about the process and wafer state.

(*v*) the identification process (prediction error approach) is based on the state-space innovation model, which eliminates the use of the Kalman filter algorithm for the state estimation, thus resulting in a computationally efficient scheme.

(*vi*) the tracking control law is generated using per-interval performance index optimized for a regulation problem (one step ahead prediction). This approach ensures higher performance and is computationally more efficient as compared to the dynamic programming approach.

It is assumed that the PGS system is an unknown strictly proper plant to be controlled which is characterized by a general stochastic state-space model, namely

$$x(k+1) = A(k)x(k) + B(k)u(k) + w(k) \tag{4.52}$$

$$y(k) = C(k)x(k) + v(k) \tag{4.53}$$

were the elements of the matrices $A(k)$, $B(k)$ and $C(k)$ are time dependent and

$x(k)$, $y(k)$ and $u(k)$ are the state, output and input vectors, respectively. $A(k)$, $B(k)$ and $C(k)$ are unknown system matrices, $w(k)$ and $v(k)$ represent uncorrelated zero-mean Gaussian modelling errors and disturbances with the following statistics: $w(k) \sim N\{0, R_w(k)\}$, $v(k) \sim N\{0, R_v(k)\}$, and x_0 is the initial state having a X_0 mean and a covariance matrix $R_{x_0}(k)$, i.e, $x_0 \sim N\{X_0, R_{x_0}\}$.

It is desired to adaptively control the plant in order to minimize the following performance index:

$$J(k) = \frac{1}{2}E\left\{[y(k+1) - y_d(k+1)]^T Q[y(k+1) - y_d(k+1)] + u^T(k)Ru(k)\right\} \quad (4.54)$$

subject to the constraints of equations (4.52) and (4.53), where $y_d(k)$ is the desired output tracking signal, Q and R are symmetric positive definite weighting matrices, and $E\{*\}$ denotes the expectation operator. To effectively control the unknown plant given by (4.52), (4.53), the plant dynamics must be identified and be known to the controller. Therefore, the control strategy for the PGS system consists of the following two main stages, namely

(i) identification of the plant using a stochastic identification algorithm, and

(ii) the control law design using a stochastic control scheme.

For the first stage, we consider $\theta(k)$, a vector of the unknown parameters of the process to be identified, which may be originating from the elements of the system matrices $A(k)$, $B(k)$ and $C(k)$ as well as the covariance matrices $R_w(k)$, R_v, R_{x_0}. In other words, the state-space model (4.52), (4.53) may be rewritten as

$$x(k+1) = A(\theta)x(k) + B(\theta)u(k) + w(k) \quad (4.55)$$

$$y(k) = C(\theta)x(k) + v(k) \quad (4.56)$$

In the identification process, it is useful to associate with the above general state space-model, the state-space innovations model of the following form [32]:

$$x(k+1,\theta) = A(\theta)x(k,\theta) + B(\theta)u(k) + K(\theta)\epsilon(k,\theta) \quad (4.57)$$

$$y(k,\theta) = C(\theta)x(k,\theta) + \epsilon(k,\theta) \quad (4.58)$$

where $\hat{y}(k,\theta) = C(\theta)x(k,\theta)$ is the predicted output, $\epsilon(k,\theta) = y(k) - \hat{y}(k,\theta)$ is the prediction error, $K(\theta) \in R^{dimX \times dimY}$ ($dimX$ and $dimY$ being the dimensions of the state space X and output space Y, respectively) is the steady-state Kalman filter matrix with unknown elements to be determined by the algorithm, and $\hat{\theta}(k)$ is the estimate of $\theta(k)$ given by

$$\hat{\theta}(k) = \hat{\theta}(k-1) + L(k)\epsilon(k,\hat{\theta}) \quad (4.59)$$

where $L(k)$ is the identification gain to be determined subsequently. As indicated below this gain is expressed in a recursive form in step (i) of the stochastic minimum variance adaptive control algorithm. In the identification process the Kalman filter gain $K(\theta)$ of the state-space innovations model is explicitly parametrized in terms of $\theta(k)$ which requires an indirect computation of $K(k)$ through the Kalman filter algorithm. The identification part of the control strategy is computationally more efficient than the modified extended Kalman filter algorithm, since the state-space innovations model eliminates the computation of the Kalman gradient $\nabla_\theta K(\theta)$ with respect to θ via the Kalman gain algorithm. Also, in this case the execution of the Kalman filter for the state estimation is not needed since the Kalman gain $K(\theta)$ is directly parametrized by $\theta(k)$ which is considered as a Markov-Gauss process [32]:

$$\theta(k) = \theta(k-1) + n(k) \quad (4.60)$$

where $n(k)$ and $\epsilon(k)$ are uncorrelated zero-mean Gaussian modeling and prediction errors, respectively, with the covariance matrices $R_n(k)$ and $R_\epsilon(k)$. The control part of the adaptive strategy uses the per-interval performance index (4.54), which is computationally more efficient than the dynamic programming algorithm, and which yields good tracking error and fast transient as shown later in Section 4.3.1.

The control law *u(k)* that minimizes the per-interval performance index (4.54) subject to the constraints of the state-space model (4.55)-(4.56) is given by

$$\begin{aligned} u(k,\theta) &= -F(k)[C(\hat{\theta})A(\hat{\theta})\hat{x}(k,\hat{\theta}) - y_d(k+1)] \\ F(k) &= [B^T(\hat{\theta})C^T(\hat{\theta})QC(\hat{\theta})B(\hat{\theta}) + R]^{-1}B^T(\hat{\theta})C^T(\hat{\theta})Q \end{aligned}$$

and where the variables required in the algorithm may be determined by using the following recursive expressions:

(*i*) Computation of the gain *L(k)*

$$\begin{aligned} R_\epsilon(k) &= R_\epsilon(k-1) + \frac{1}{k}[\epsilon(k,\hat{\theta})\epsilon^T(k,\hat{\theta}) - R_\epsilon(k-1)] \\ S(k) &= \Psi^T(k,\hat{\theta})P(k-1)\Psi(k,\hat{\theta}) + R_\epsilon(k) \\ L(k) &= P(k-1)\Psi(k,\hat{\theta})S^{-1}(k) \\ P(k) &= P(k-1) - L(k)\Psi^T(k,\hat{\theta})P(k-1) + R_\epsilon(k) \end{aligned}$$

(*ii*) Stochastic minimum variance adaptive identification

$$\begin{aligned} \epsilon(k,\hat{\theta}) &= y(k) - \hat{y}(k,\hat{\theta}) \\ \hat{\theta}(k) &= \hat{\theta}(k-1) + L(k)\epsilon(k,\hat{\theta}) \\ \hat{x}(k+1,\hat{\theta}) &= A(\hat{\theta})\hat{x}(k,\hat{\theta}) + B(\hat{\theta})u(k) + K(\hat{\theta})\epsilon(k,\hat{\theta}) \end{aligned}$$

$$
\begin{aligned}
\hat{y}(k+1,\hat{\theta}) &= C(\hat{\theta})\hat{x}(k+1,\hat{\theta}) + D(\hat{\theta})u(k+1) \\
W(k+1,\hat{\theta}) &= [A(\hat{\theta}) - K(\hat{\theta})C(\hat{\theta})]W(k,\hat{\theta}) + M(\hat{\theta},\hat{x},u,\epsilon) - K(\hat{\theta})D(\hat{\theta},\hat{x}) \\
\Psi(k+1,\hat{\theta}) &= W^T(k+1,\hat{\theta})C^T(\hat{\theta}) + D^T(\hat{\theta},\hat{x}(k+1))
\end{aligned}
$$

(*iii*) Control law for the closed-loop system

$$
\begin{aligned}
u(k,\theta) &= -F(k)[C(\hat{\theta})A(\hat{\theta})\hat{x}(k,\hat{\theta}) - y_d(k+1)] \\
F(k) &= [B^T(\hat{\theta})C^T(\hat{\theta})QC(\hat{\theta})B(\hat{\theta}) + R]^{-1}B^T(\hat{\theta})C^T(\hat{\theta})Q
\end{aligned}
$$

where

$$
\begin{aligned}
M(\hat{\theta},\hat{x},u,\epsilon) &= \frac{d}{d\theta}[A(\theta)\hat{x}(k) + B(\theta)u(k) + K(\theta)\epsilon(k)]_{\theta=\hat{\theta}} \\
D_1(\hat{\theta},x) &= \frac{d}{d\theta}[C(\theta)\hat{x}(k,\theta)]_{\theta=\hat{\theta}},
\end{aligned}
$$

and

$$K(\hat{\theta}(k)) = K(\hat{\theta}(k-1)) + L(k)\epsilon(k,\hat{\theta}).$$

4.3.1 Simulation Results

Our goal in this section is to design a real-time MIMO stochastic adaptive control strategy capable of operating effectively to work well in a stochastic environment and yielding fast transient and good tracking error. For simulation purposes and to compare our results, we consider the model already used in the LQG control problem. This model was developed in Section 4.2.1 and is given by (3.22)-(3.23). The plant is parametrized by $\theta(k)$ as a 20x1 vector of unknown parameters to be identified for the system matrices $A(k)$, $B(k)$, $C(k)$ and the Kalman gain matrix $K(\hat{\theta})$ as shown explicitly below:

$$A(k) = \begin{bmatrix} 0 & \theta_1(k) & 0 & 0 \\ 1 & \theta_2(k) & 1 & 0 \\ 0 & 0 & 0 & \theta_3(k) \\ 0 & 0 & 1 & \theta_4(k) \end{bmatrix}, B(k) = \begin{bmatrix} \theta_5(k) & \theta_6(k) \\ \theta_7(k) & \theta_8(k) \\ \theta_9(k) & \theta_{10}(k) \\ \theta_{11}(k) & \theta_{12}(k) \end{bmatrix}, K(\hat{\theta}) = \begin{bmatrix} \hat{\theta}_{13}(k) & \hat{\theta}_{14}(k) \\ \hat{\theta}_{15}(k) & \hat{\theta}_{16}(k) \\ \hat{\theta}_{17}(k) & \hat{\theta}_{18}(k) \\ \hat{\theta}_{19}(k) & \hat{\theta}_{20}(k) \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \text{ and } \hat{x}(k,\hat{\theta}) = \begin{bmatrix} \hat{x}_1(k,\hat{\theta}) \\ \hat{x}_2(k,\hat{\theta}) \\ \hat{x}_3(k,\hat{\theta}) \\ \hat{x}_4(k,\hat{\theta}) \end{bmatrix}.$$

The matrix $M(\hat{\theta}, x, u, \epsilon)_{4\times 20}$ has the following elements :

$M(1,1) = \hat{x}_2(k,\hat{\theta})$, $M(1,5) = u_{1opt}$, $M(1,6) = u_{2opt}$, $M(1,13) = e_1$, $M(1,14) = e_2$

$M(2,2) = \hat{x}_2(k,\hat{\theta})$, $M(2,7) = u_{1opt}$, $M(2,8) = u_{2opt}$, $M(2,15) = e_1$, $M(2,16) = e_2$

$M(3,3) = \hat{x}_4(k,\hat{\theta})$, $M(3,9) = u_{1opt}$, $M(3,10) = u_{2opt}$, $M(3,17) = e_1$, $M(3,18) = e_2$

$M(4,4) = \hat{x}_4(k,\hat{\theta})$, $M(4,11) = u_{1opt}$, $M(4,12) = u_{2opt}$, $M(4,19) = e_1$, $M(4,20) = e_2$.

where $M(i,j) = 0$ for all other i's and j's different than the above values, $u_{opt} = F(y_d - CA\hat{x}(k,\hat{\theta})) = \begin{bmatrix} u_{1opt} \\ u_{2opt} \end{bmatrix}$ is the optimal control for the closed-loop system, and $e(k) = y(k) - \hat{y}(k) = \begin{bmatrix} e_1 \\ e_2 \end{bmatrix}$ is the prediction error.

For the proposed algorithm the covariance matrix and the weighting matrices Q and R of the performance index $J(k)$ are assigned so as to increase the effect of the control or of the system output. The improved performance of the stochastic minimum variance adaptive control algorithm can be seen in Figures 4.24-4.27. These figures also show a fast adaptation speed and convergence. In the identification process, the highly oscillatory property of the adaptive control signal contributes largely to exciting the system dynamics for yielding good identification. For a deterministic

$V_{bias}/[F]$	t_r[s]	t_i[s]	%OS	ϵ	Convergence
V_{bias}	100	5	200 [Volts]	1.29 [Volts]	good
$[F]$	50	5	22 [%]	0.05 [%]	good

Table 4.3: The closed-loop system performance of the Stochastic minimum variance adaptive control strategy.

control environment, perfect tracking at steady-state can be obtained. The following parameters are selected for the controller in our simulation results:

$\hat{y}(0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$, $\hat{\theta}(0) = O_{20\times1}$, $R_n(0) = 50.2 * U_{20\times20}$ (matrix with all the elements equal to one), $\Psi(0)_{20\times2}$ =0.581 $* U_{20\times2}$, $P(0) = 0.81 * I_{20\times20}$ (covariance matrix), $M(0) = O_{4\times20}$, $K(0) = O_{4\times2}$, $W(0) = O_{4\times20}$, $R_w = 0.005$, $R_v = 0.005$, $Mean_w = 0$, $Mean_v = 0$,

$Q = \begin{bmatrix} 8000 & 0 \\ 0 & 8000 \end{bmatrix}$ (the output weighting matrix in the performance index $J(k)$), and $R = \begin{bmatrix} 0.001 & 0 \\ 0 & 0.001 \end{bmatrix}$ (the input weighting matrix in the performance index $J(k)$).

The closed-loop system performance of this control strategy, namely the settling time (t_r), rise time (t_i), overshoot ($\%OS$), steady-state error (ϵ) and the convergence of the algorithm for V_{bias} and $[F]$ are presented in Table 4.3.

The results of the simulations are presented in Figures 4.24-4.27 for the case without changes in the set point and in Figures 4.28-4.29 for the case with changes in the set points. These figures reveal good tracking performance in steady-state for V_{bias} and $[F]$ and a smooth control effort for both the actuators *throttle valve* and *RF power*, as depicted in Figures 4.24-4.25. Also, the simulations reveal a strong dependence of the computation speed of the algorithm with the number of samples and the poor estimation of the parameters in the first period of the transient with

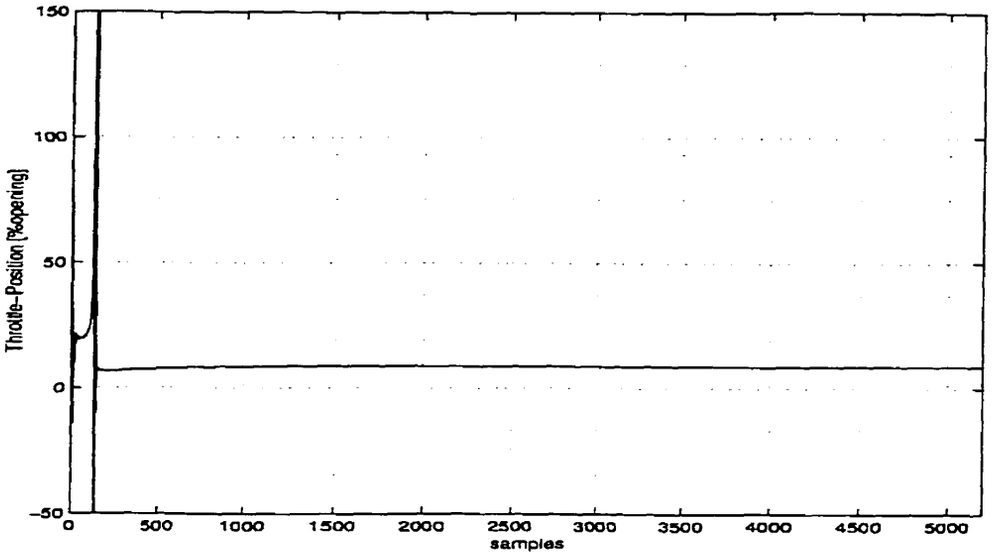


Figure 4.24: The performance of the minimum variance adaptive controller on the *throttle position* of the closed-loop system.

a degradation of the performance, but in the second period of the transient (after 2 seconds) the algorithm works properly having enough informations about the system, and estimates well these parameters. The algorithm is not quite robust to the variations in the tuning parameters, namely on the initial values of the covariance matrices $P(0)$, $R_n(0)$, and the weighting matrices Q and R. If the algorithm is not initialized properly its performance could be affected and get degraded. Unfortunately due to the nonlinear nature of the algorithm a formal procedure for tuning these parameters are not available, and consequently in practice one has to resort to some trial and error to determine the best tuning quantities.

4.4 Summary

In this chapter we have developed three real-time feedback control strategies to manipulate fluorine concentration ($[F]$) and induced dc voltage (V_{bias}) across the reactor electrodes in an RIE system. The control strategies are based on *PI self-tuning* adap-

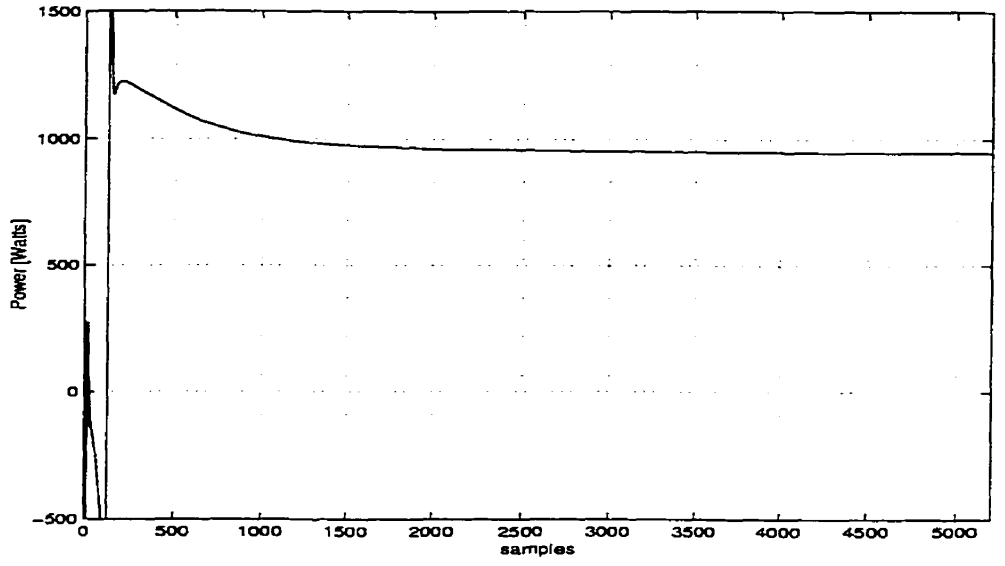


Figure 4.25: The performance of the minimum variance adaptive controller on the *RF power* of the closed-loop system.

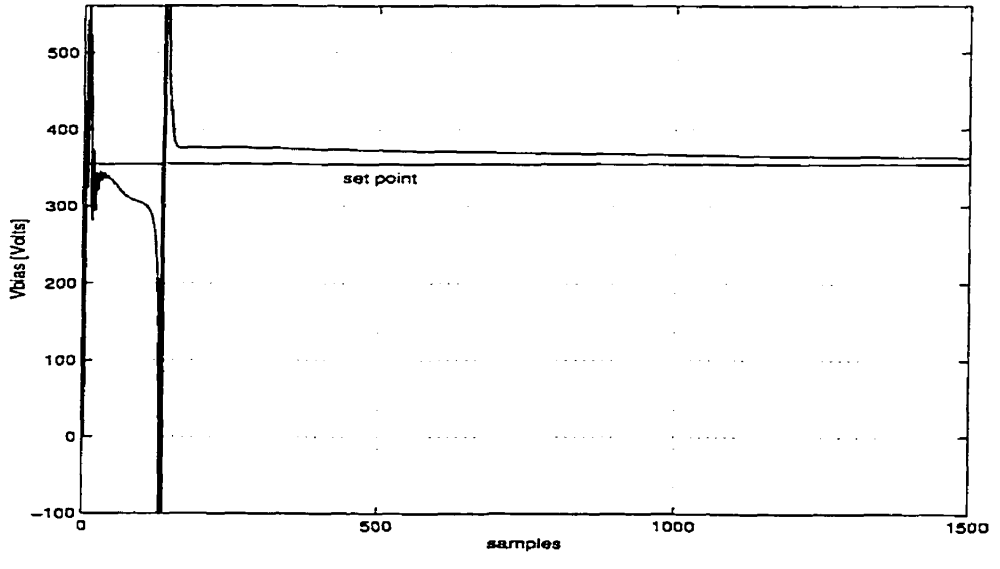


Figure 4.26: The performance of the minimum variance adaptive controller on the *dc* V_{bias} of the closed-loop system.

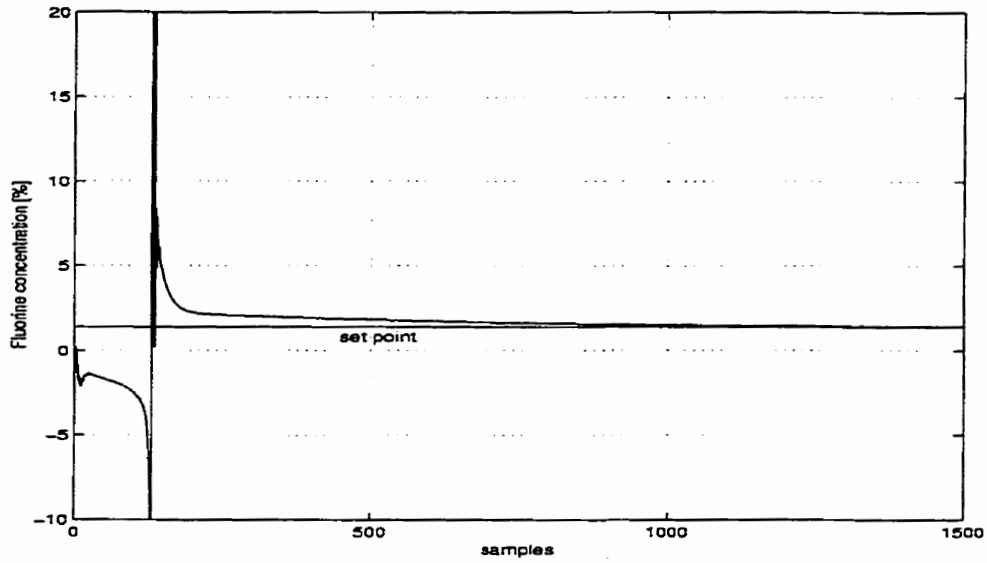


Figure 4.27: The performance of the minimum variance adaptive controller on the *fluorine concentration* $[F]$ of the closed-loop system.

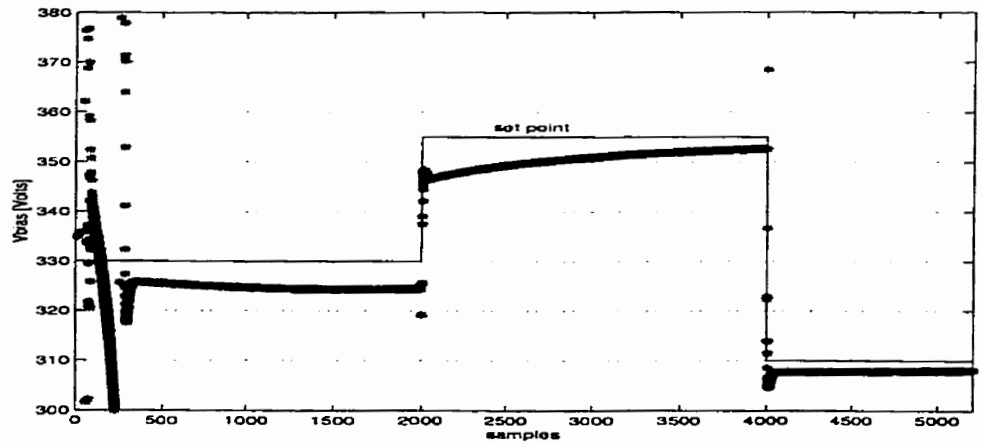


Figure 4.28: The performance of the minimum variance adaptive controller on the *dc* V_{bias} for the closed-loop system with changes in the set points.

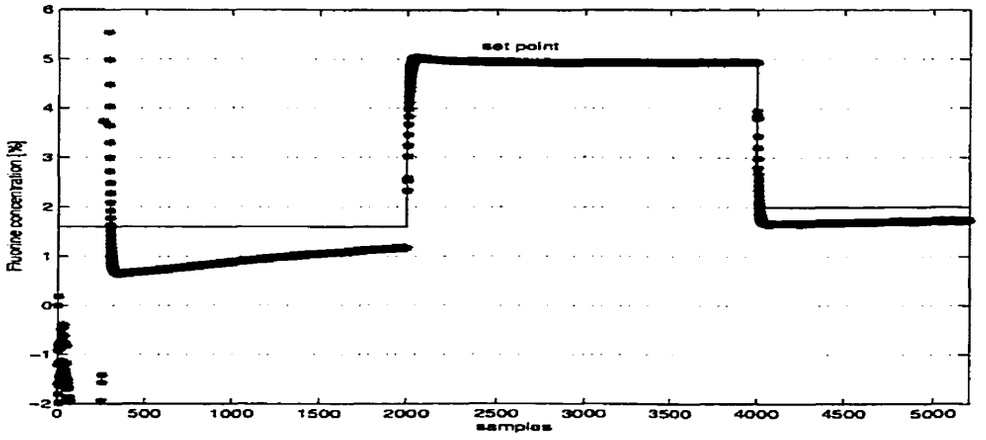


Figure 4.29: The performance of the minimum variance adaptive controller on $[F]$ for the closed-loop system with changes in the set points.

tive controller, *Linear Quadratic Gaussian* (LQG) scheme and minimum variance stochastic adaptive control. The controlled species were measured by quadrupole mass spectrometry with direct effect on the outcome of the performance variables. This demonstrates the feasibility of our control strategies in the sense that an *in-situ* sensor-based controller is capable of reducing significantly the variations of the process variables. The results obtained confirm the utility of our proposed strategies to control the etching process with drifts. Moreover, these algorithms can be used as a research tool for evaluating complicated gas kinetics. The controllers can drive the film properties, namely *etch depth, anisotropy, selectivity* and *uniformity* to desired regimes based on important gas species and dc in induced voltage in an optimal fashion. Also, the mass flow controller offset experiments [21], [37] demonstrate the utility of real-time sensing and control for disturbance rejection.

Chapter 5

Intelligent Control Strategies

In this chapter we develop neuromodels used to construct intelligent controllers real-time neurocontrol strategies. Application of neural networks to control systems have become increasingly important. The massive parallel processing, nonlinear mapping, and self-learning abilities of neural networks have been the motivating factors for development of "intelligent" control systems. Our objective in this thesis is to demonstrate that the RIE neuromodels integrated in intelligent control architectures offer advantages in both accuracy and robustness over traditional statistical models.

In [15] a method to develop plasma etching neuromodels is presented which outperforms the predictions of statistical regression models from limited experimental data. In fact, in nearly every one of these studies, the authors found that building accurate plasma etching neuromodels generally requires fewer training experiments than classical statistical methods. Also, in [13] neural networks are used to model etch rate, etch anisotropy, spatial variations in etch rate (i.e., etch uniformity), and etch selectivity. Our goal here is to built accurate and robust neurocontrollers, and to explore their potential benefits as measured by their ability to attenuate the effects of exogenous perturbations on the etch characteristics. It is not clear *a priori*

whether good control of V_{bias} and F will lead to stabilization and control of the etch characteristics. Etch rate data is collected in real-time using the interference of laser light reflected off of the laser surface.

5.1 Inverse Dynamics Neuromodels for the Plasma Generation Subsystem (PGS)

In recent years, there has been a number of neuro-control learning schemes proposed in the literature. Among these, the inverse model neurocontrol approach, developed by Widrow and Stearns and Psaltis et al. [33], [34] has been one of the most viable techniques for implementation of neural networks in control. One reason for its utility is its simplicity. Once the network has learned the inverse model of the plant, it is configured as a direct controller for the system. We chose to investigate the inverse dynamics control technique because of its ease of implementation. The objective of a nonlinear dynamic inversion is to invert the dynamic equations of the plant directly in order to find the control necessary to yield the given output. In this thesis we perform this operation using neuromodels capable of learning the highly nonlinear inverse dynamics of the plasma process. We describe several architectures for the inverse dynamics of the RIE process. To learn the inverse dynamics of the plant, we train the neurocontrollers off-line. By applying the desired range of inputs to the plant, its corresponding outputs can be obtained and a set of training patterns can be selected. Once trained, the networks could be used to produce the appropriate control input as a function of the desired plant output. The performance of the neural networks based on these input vectors are observed by configuring it directly to control the plasma etching process. Based on these observations, the neural network structures that give the best performances are then used in the neurocontrol structures of the

PGS system.

5.2 Neurocontroller Structures

In this section four nonlinear neurocontroller structures (inverse dynamics neuromodels) for the PGS system are developed, namely

(i) nonlinear static neurocontroller

(ii) nonlinear first-order neurocontroller

(iii) nonlinear second-order neurocontroller

(iv) nonlinear first-order controller with delayed control

To learn the inverse dynamics model of the PGS system, we use the same neural network architectures that were developed in Chapter 3, Section 3. Each inverse neuromodel is then configured as a direct controller for the PGS system. The results of the simulations presented in the next section demonstrate that the neurocontrollers perform very well and offer encouraging advantages as compared to the other conventional methods.

The structures proposed in this section are described in more details below:

(*i*) *Nonlinear static neurocontroller* (2-8-2): This architecture is equivalent to the neural network model developed in Section 3.3, and is repeated here for convenience:

$$u(k) = g(y(k)) \tag{5.1}$$

(*ii*) *Nonlinear first-order neurocontroller* (4-8-2): This architecture is equivalent to the first order nonlinear model introduced in Section 3.3, and is repeated here as follows:

$$u(k) = g(y(k), y(k-1)) \tag{5.2}$$

(*iii*) *Nonlinear second-order neurocontroller* (6-8-2): This architecture is equivalent to the second-order nonlinear model given in Section 3.3, and is repeated here as follows:

$$u(k) = g(y(k), y(k-1), y(k-2)) \tag{5.3}$$

(*iv*) *Nonlinear first-order neurocontroller with delayed control* (6-8-2): This architecture is equivalent to the first-order nonlinear model introduced in Section 3.3, and is repeated here for convenience:

$$u(k) = g(y(k), y(k-1), y(k-2)) \tag{5.4}$$

5.2.1 Simulation Results

Our objective in this section is to design several neurostructures capable of learning and generalizing accurately the inverse dynamics of the PGS, and representing effectively the experimental input-output data. The above neurostructures are trained off-line using backpropagation error algorithm with adaptive learning rate and momentum [24]. The simulation results for off-line training are presented in Figures 5.1-5.8. Before training the networks weights and biases are initialized with Nguyen-Widrow initial conditions (small random values) [24]. The training parameters are initialized to the following values: error goal = 0.01, learning rate = 0.02, and momentum = 0.95. The number of the epochs to reach the error goal varied depending on the initial conditions of the weights and biases and the number of hidden neurons. By trial and error we determined that 8 hidden neurons are sufficient to design an appropriate neurostructure.

In the Table 5.1 we synthesize the performance of these neurocontrol models represented by the standard deviation errors between the models and the experimental

Neurocontrol	Throttle Position e [%]	RF power e [Watts]
Static representation	2.3894	12.1474
First-order	2.3875	12.1374
Second-order	2.3785	11.9156
First-order with delayed input	2.3814	11.9207

Table 5.1: Performance comparison for the neuromodels.

data set (e). The maximum magnitude of these standard deviation errors is 2.5 [%] for the *throttle position* and 15 [Watts] for the *RF power*. The nonlinear static neurocontroller in testing phase fits the experimental data set, as shown in Figures 5.1-5.2, with standard deviation errors of $e = 12.1474$ [Watts] for the RF *power* actuator and $e = 2.3894$ [%] for the *throttle position* actuator. The nonlinear first-order neurocontroller in testing phase fits the experimental data set, as shown in Figures 5.3-5.4, with standard deviation errors of $e = 2.3875$ [%] for the *throttle position* actuator and $e = 12.137$ [Watts] for the *RF power*. The nonlinear second-order neurocontroller in the testing phase fits the experimental data set, as shown in Figures 5.5-5.6, with the smallest standard deviation errors of $e = 2.3785$ [%] for the *throttle position* actuator and $e = 11.9156$ [Watts] for the RF *power* actuator. The nonlinear first-order neurocontroller with delayed input fits the experimental data set, as shown in Figures 5.7-5.8, with the standard deviation errors very close to the third structure. The above neurocontrollers will be used for the closed-loop real-time strategies developed in the next section.

5.3 Real-Time Neurocontrol Strategies

After training the neuromodels to learn accurately the inverse dynamics of the process in all the four architectures, each neural network controller is now configured as a direct controller for the PGS system as shown in Figures 5.9-5.12. Although the

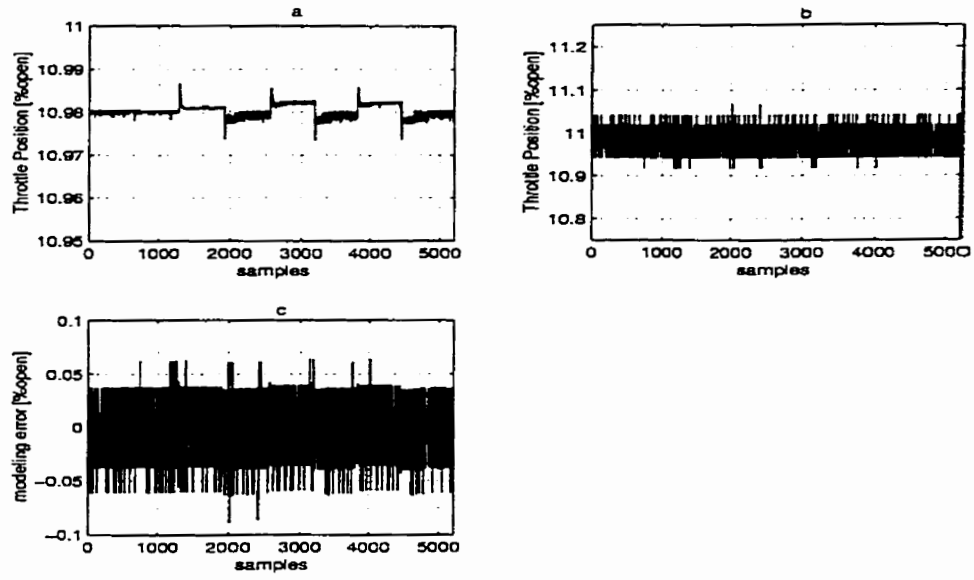


Figure 5.1: Generalization results for the nonlinear static neurocontroller for the *throttle position*.
Legend: (a) neuromodel output; (b) experimental data set; (c) representation error performance.

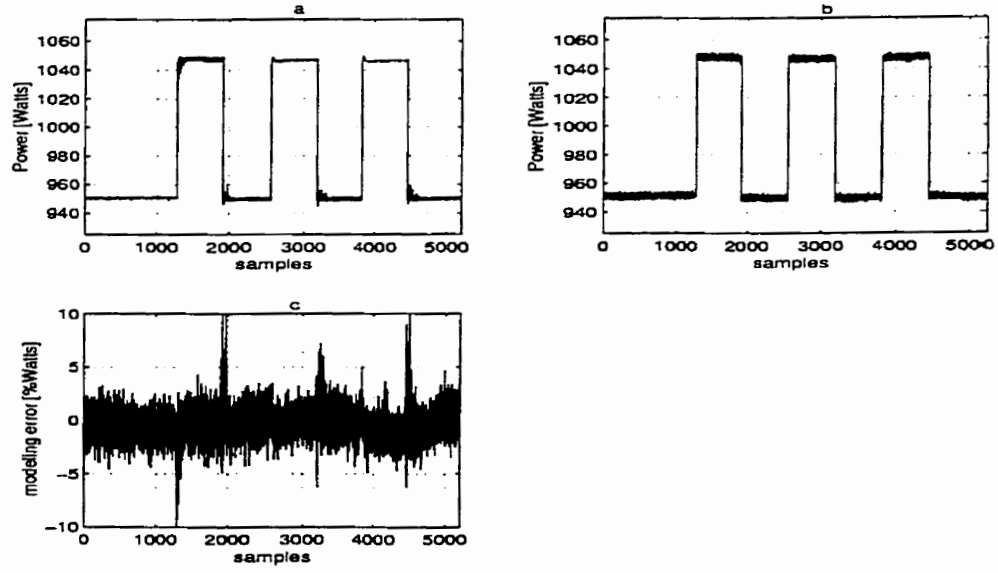


Figure 5.2: Generalization results for the nonlinear static neurocontroller for the *RF power*.
Legend: (a) neuromodel output; (b) experimental data set; (c) representation error performance.

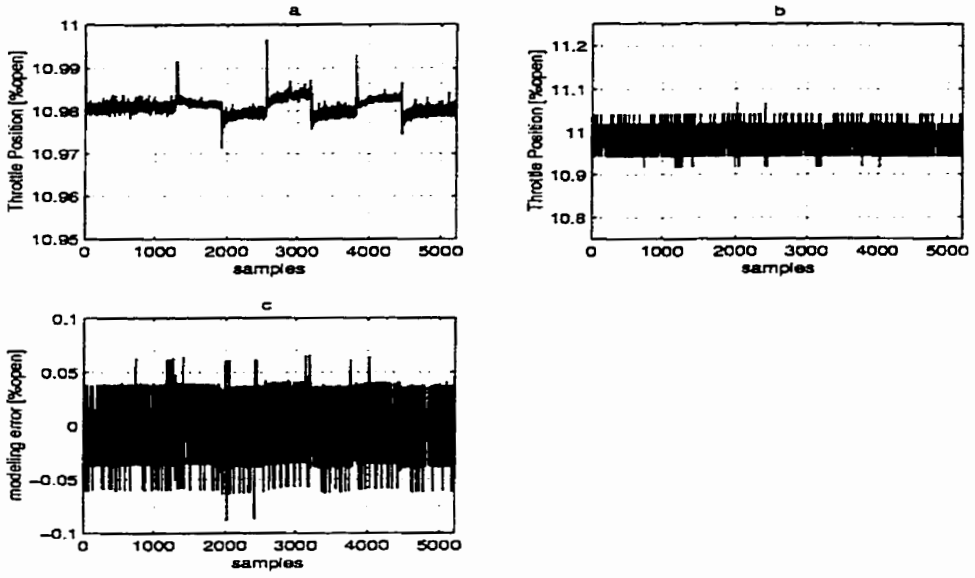


Figure 5.3: Generalization results for the nonlinear first-order neurocontroller for the *throttle position*.
Legend: (a) neuromodel output; (b) experimental data set; (c) representation error performance.

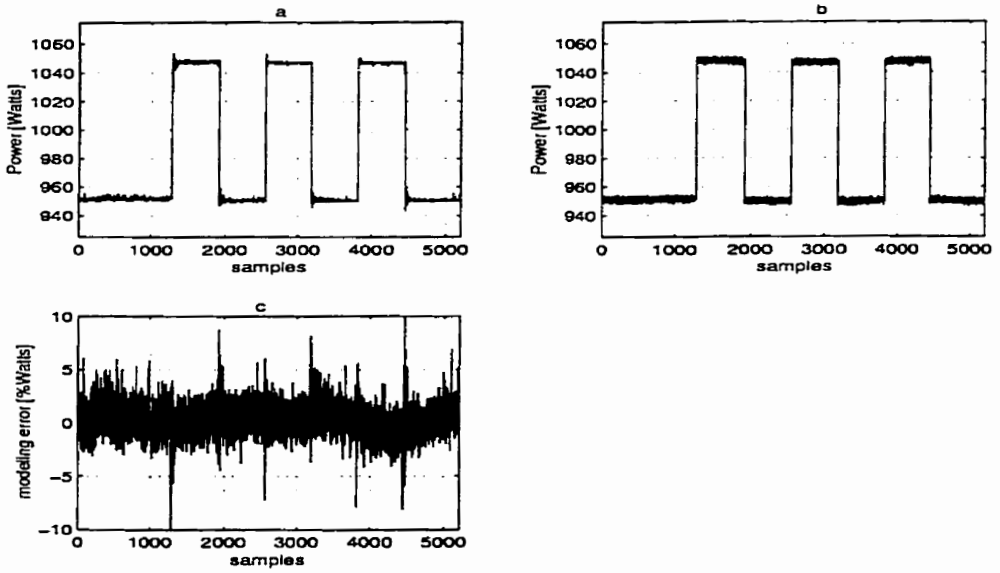


Figure 5.4: Generalization results for the nonlinear first-order neurocontroller for the *RF power*.
Legend: (a) neuromodel output; (b) experimental data set; (c) representation error performance.

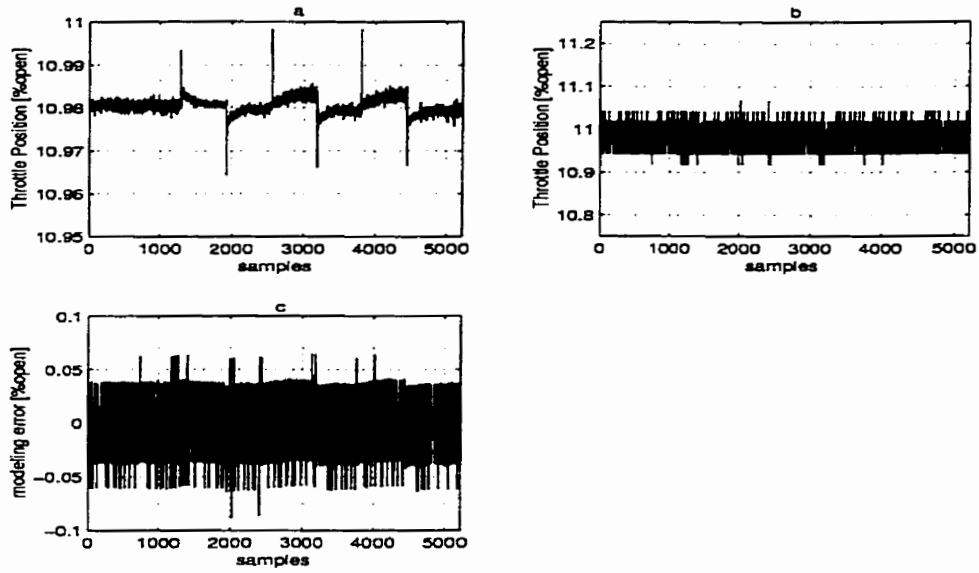


Figure 5.5: Generalization results for the nonlinear second-order neurocontroller for the *throttle position*.
Legend: (a) neuromodel output; (b) experimental data set; (c) representation error performance.

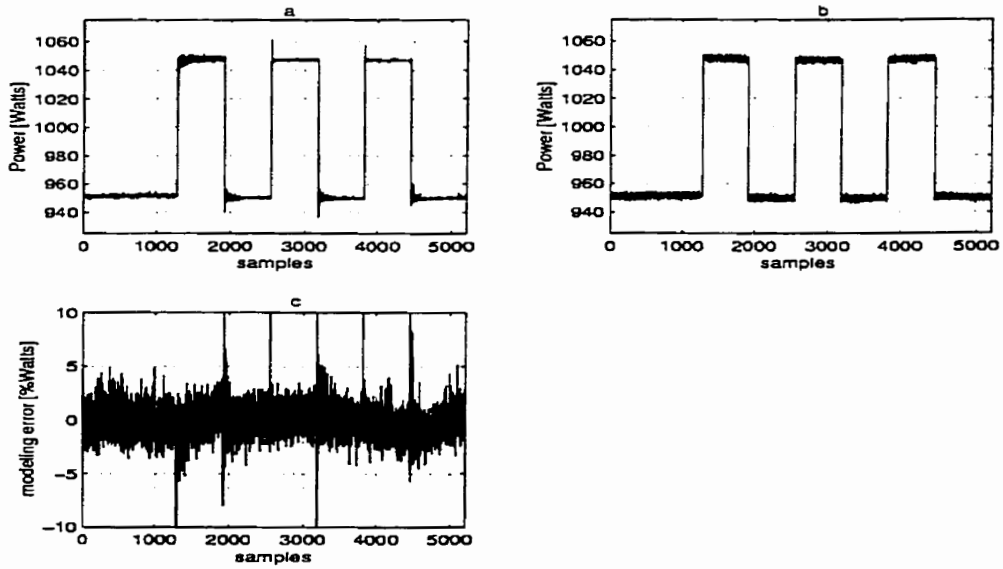


Figure 5.6: Generalization results for the nonlinear second-order neurocontroller for the *RF power*.
Legend: (a) neuromodel output; (b) experimental data set; (c) representation error performance.

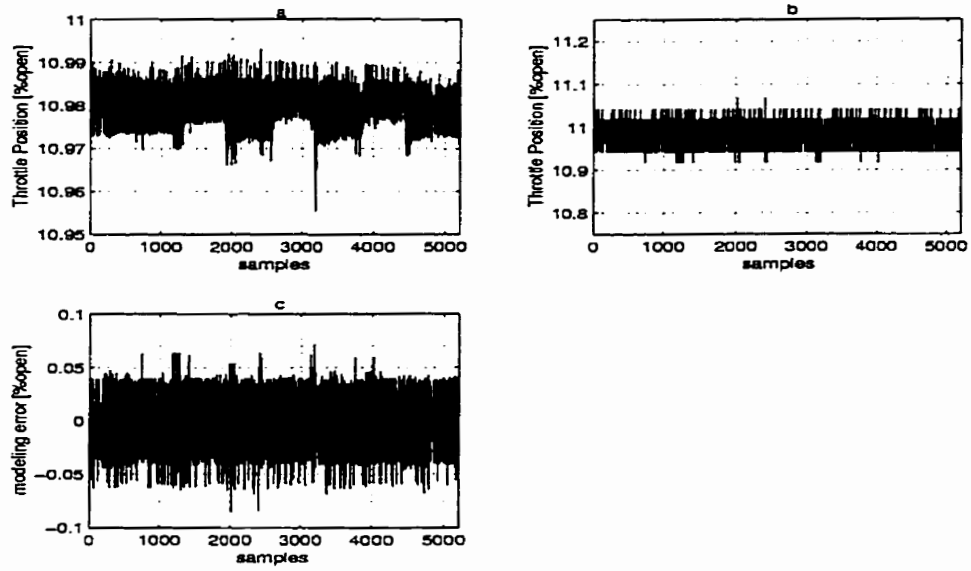


Figure 5.7: Generalization results for the nonlinear first-order neurocontroller with delayed control for the *throttle position*.
Legend: (a) neuromodel output; (b) experimental data set; (c) representation error performance.

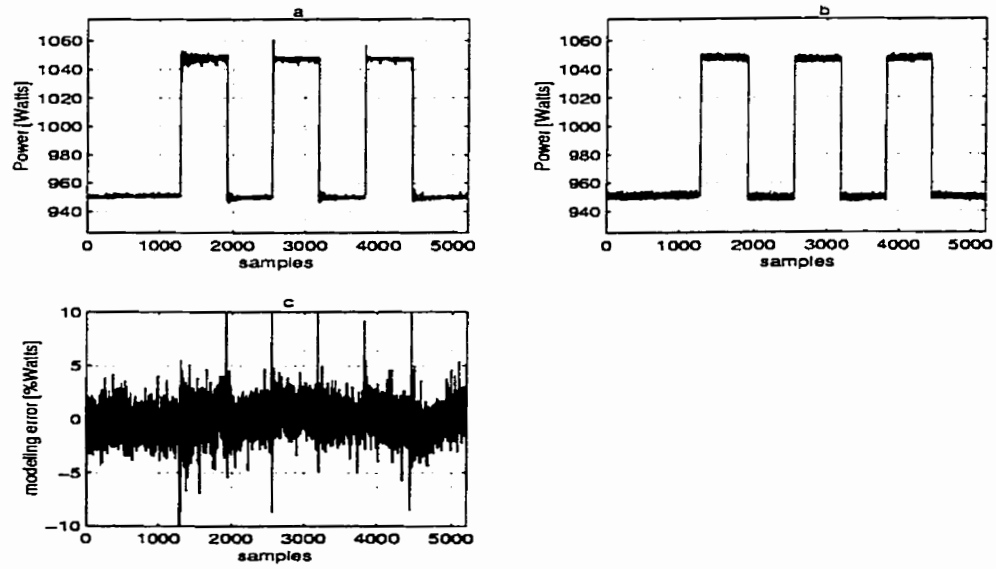


Figure 5.8: Generalization results for the nonlinear first-order neurocontroller with delayed control for the *RF power*.
Legend: (a) neuromodel output; (b) experimental data set; (c) representation error performance.

inverse neuromodel control scheme is similar in architecture to a conventional feedback control scheme, the neurocontroller property is similar to that of a traditional self-tuning adaptive controller. In the direct self-tuning control scheme, an *a priori* knowledge of the plant's mathematical model is required and the plant's parameters need to be estimated through an estimation scheme, such as the popular least-squares parameter estimation algorithm [22], [35]. From the estimates of the plant's parameters, the controller is then able to generate the correct control input through an inverse mathematical model of the plant. The inverse model neurocontrol approach is more robust and simpler in this respect since it is not necessary to derive the mathematical model of the plant for implementation. The ability of neural network models to learn and generalize based on the input-output behaviour of a process has a great advantage where many control problems can now be treated with less precision and advanced knowledge of the plant. The use of the nonlinear sigmoidal functions in the hidden layers of the neural networks has also made it possible to consider highly nonlinear control systems where many traditional adaptive and conventional control techniques may be insufficient. Moreover, it is applicable to complex and ill-defined plants when mathematical modeling is difficult. Another advantage of using neural networks is that it can also be trained on-line to further improve its performance. In the basic inverse model approach, the neurocontroller can be trained on-line by minimizing the system's performance error. A more suitable approach for training the neurocontroller on-line is to adapt the architecture of an indirect adaptive control scheme where the identification model is replaced by a neural network similar to one of the architectures presented in the previous section.

In order for the proposed neurocontrollers to operate as controllers, the input vector $y(k)$ is replaced by the desired plant output (i.e., the set point, $y_p(k)$) and the remaining input signals remain unchanged (as feedback signals of the delayed outputs

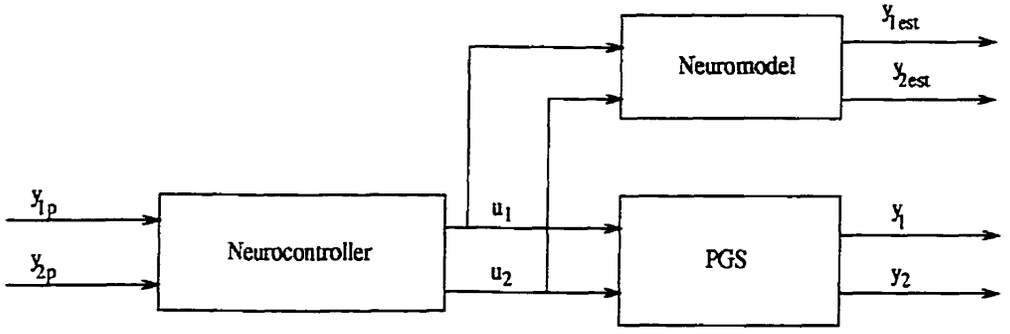


Figure 5.9: The first neurocontrol strategy.

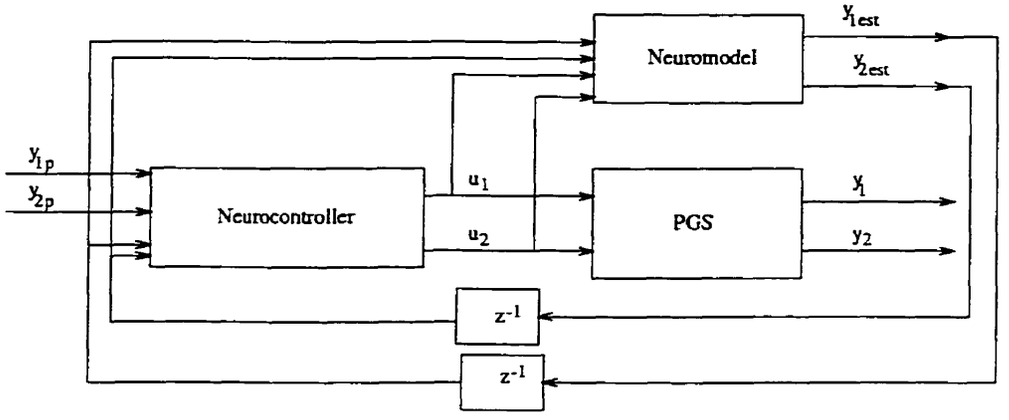


Figure 5.10: The second neurocontrol strategy

from the neuromodels) as shown in the Figures 5.9-5.12.

With respect to the above configuration the following four neurocontrol strategies are proposed:

(*i*) Nonlinear neurocontroller and neuromodel of the plant in static representation as depicted in Figure 5.9.

(*ii*) Nonlinear first-order neurocontroller and the first-order neuromodel of the plant as depicted in Figure 5.10.

(*iii*) Nonlinear second-order neurocontroller and the second-order neuromodel of the plant as depicted in Figure 5.11.

(*iv*) Nonlinear first-order neurocontroller and the first-order neuromodel of the plant with delayed control as depicted in Figure 5.12.

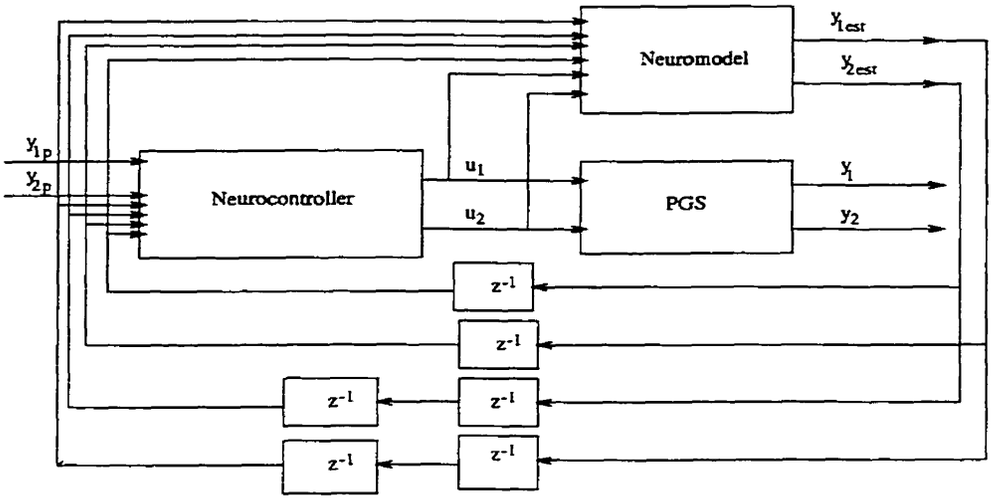


Figure 5.11: The third neurocontrol strategy.

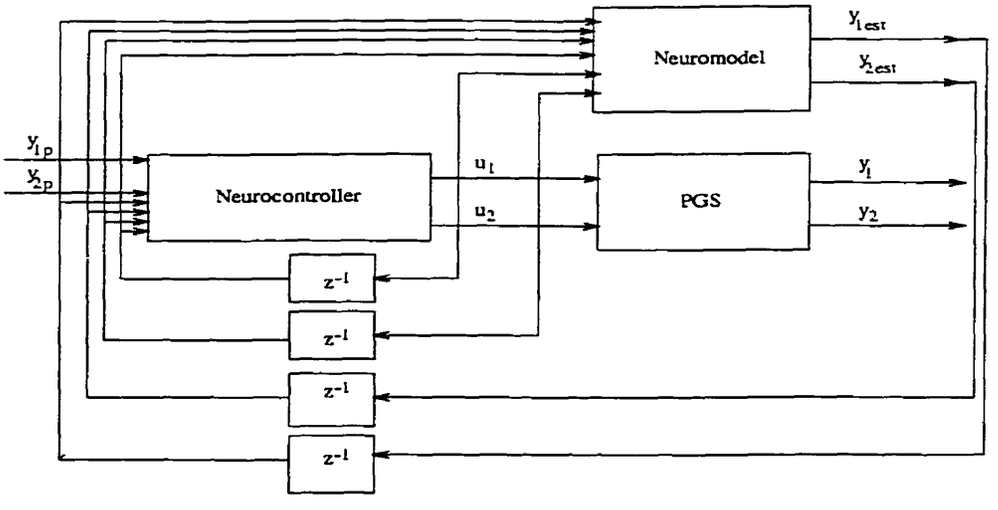


Figure 5.12: The fourth neurocontrol strategy.

The simulation results performed for the above architectures are discussed in the next section.

Each of the above neurocontrollers are trained to learn the inverse dynamics of each channel of the PGS process by using the respective channel's output data together with some corresponding delayed output values as the input patterns and the corresponding channel's input data as target patterns. The PGS neural network models are trained to learn the feedforward dynamics of each channel by using the input data together with some delayed output data as the input patterns and the corresponding output data as the target patterns. For training a neural network to learn the inverse or the feedforward dynamic model of the plant off-line training based on Levenberg-Marquardt backpropagation error algorithm [24] is used as described before. Since the neural networks are highly nonlinear, it is difficult to determine analytically which model has completely learned the true plant inverse and feedforward dynamics. A simple and a reliable method to establish the performance of these networks is to test each of these neuromodels on-line as configured in our proposed neurostructures shown in Figures 5.9-5.12.

5.3.1 Simulation Results

Our goal in this section is to design several real-time *MIMO* intelligent control strategies and to select amongst them the most suitable one which ensures among others the stability of the PGS system in closed-loop, good tracking error, robustness to changes in the set points and operate effectively in a large operating range.

The closed-loop system performance of these neurocontrol strategies, namely the settling time (t_r), rise time (t_i), overshoot ($\%OS$), steady-state error (ϵ) and the convergence of the algorithm for V_{bias} and $[F]$ are presented in Table 5.2. The simulations results are presented in this section. In these simulations, as shown in Figures

Neurostructure	t_r[s][	t_i[s]	%OS	ϵ	Conv.
First structure	1/-	0.5/-	22 [Volts]/-	22 [Volts]/-	bad
Second structure	2/0.5	-/-	-/-	10 [Volts]/5 [%]	good
Third structure	2/1	1.5/-	7 [Volts]/-	1.36 [Volts]/0.015 [%]	very good
Fourth structure	2.5/ 2	2/1	4 [Volts]/-	1.36 [Volts]/0.025 [%]	very good

Table 5.2: The closed-loop system performance of the neurocontrol strategies.

5.13-5.18, the set points are taken as: y_{1p}= 355 [Volts] and y_{2p} = 1.4 [%]. During the simulations with the nonlinear model of the PGS plant, changes in the set points are taken into consideration and so we are able to choose the neurocontroller that gives the best performance for these changes. The changes in the set points for the third neurocontroller occur at the same time as in the open-loop simulations as depicted in Figures 5.17-5.18. The first case (Figures 5.13-5.14) results in the largest errors and the convergence is relatively poor during training. The evolution of the actuators *throttle valve position* and *RF power* for this case (Figures 5.15-5.16) is very smooth and without significant variations around the operating point. This implies that the control effort to maintain the key plasma parameters V_{bias} and *fluorine concentration* $[F]$ at their set point values remain well behaved. By only one input vector $u(k)$, it is clear that the neurocontroller is unable to perform satisfactorily as a controller, as it has only the desired plant output to deal with. The third case (Figures 5.17-5.18) produces the best performance with the minimum absolute error. Also, the evolution of the actuators *throttle valve position* and *RF power* are very smooth with small changes around the operating point. Since two of the three input vectors are a delayed plant output, the neurocontroller can act very quickly to changes in the plant output, thus yielding rapid and accurate performance. For the second and fourth cases (Figures 5.13-5.14) the neurocontrollers results show some output overshoots, thus the controllers are not able to provide accurate tracking of the set points. With additional delayed plant outputs used as input vectors, a smoothing effect is achieved

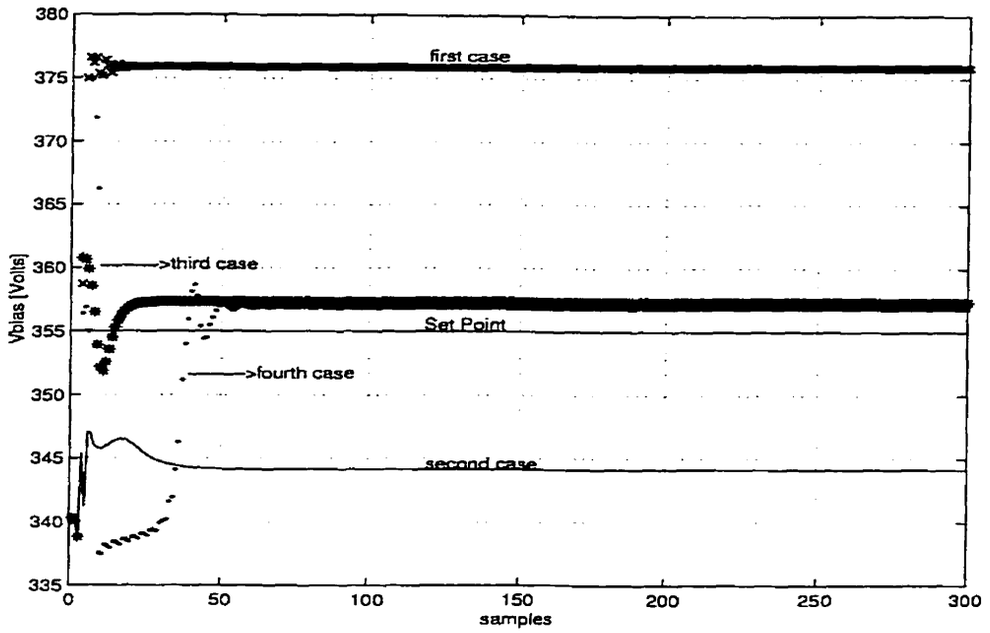


Figure 5.13: The evolution of the *dc* V_{bias} for the closed-loop system.

at the output of the neurocontroller, however a delay is also introduced in reacting to the plant behaviour. This could be due to the fact that the PGS plant is dominated as a second-order plant, and so the modeling errors generate a mismatch in its input vectors to follow the actual dynamics of the plant, which could lead to an imperfect inverse model learning. For the first case, the error converges during the training to a relatively smaller value compared to the other cases, however, its recall performance is very poor as the true inverse model cannot be obtained by using the particular input. The reason the fourth case converges quickly during the training stage is that the outputs of the neurocontroller follow the target values rapidly but it has, however, not generalized properly over the entire input range. The third neurocontroller structure seems to be the most suitable one for controlling the PGS system in closed-loop.

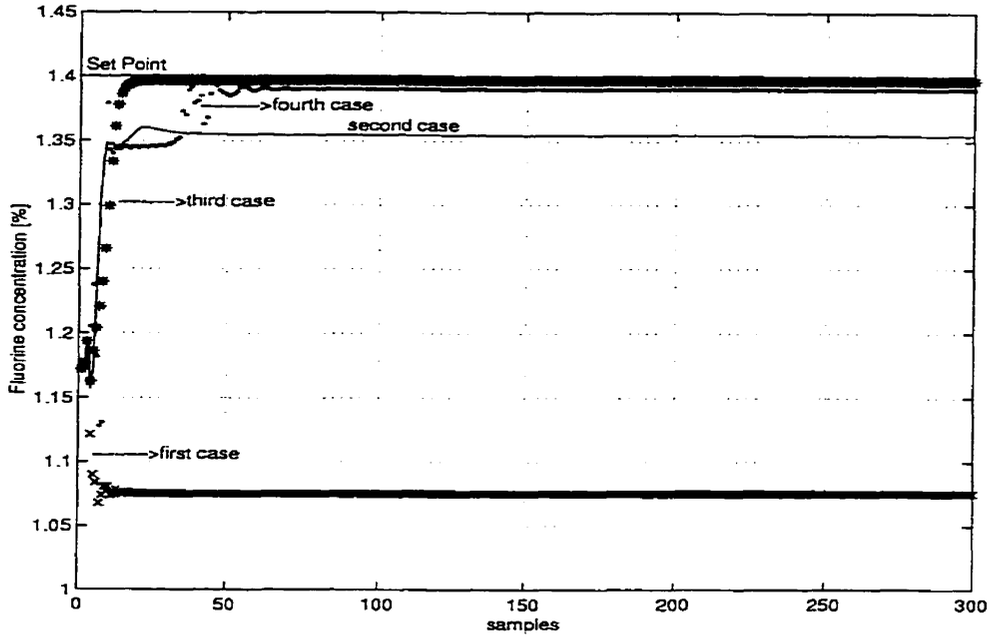


Figure 5.14: The evolution of the *fluorine concentration* $[F]$ for the closed-loop system

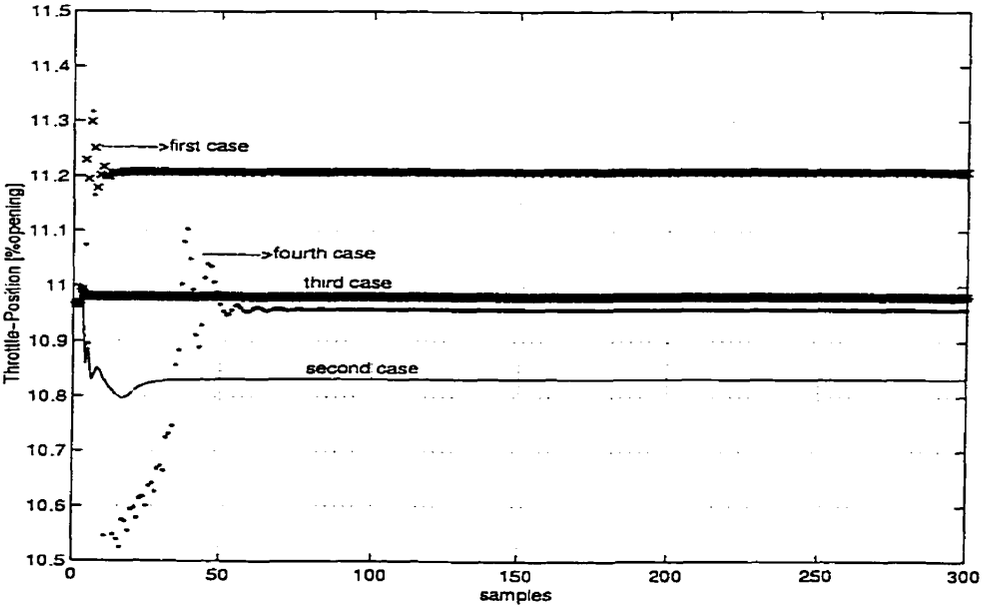


Figure 5.15: The evolution of the *throttle position* for the closed-loop system.

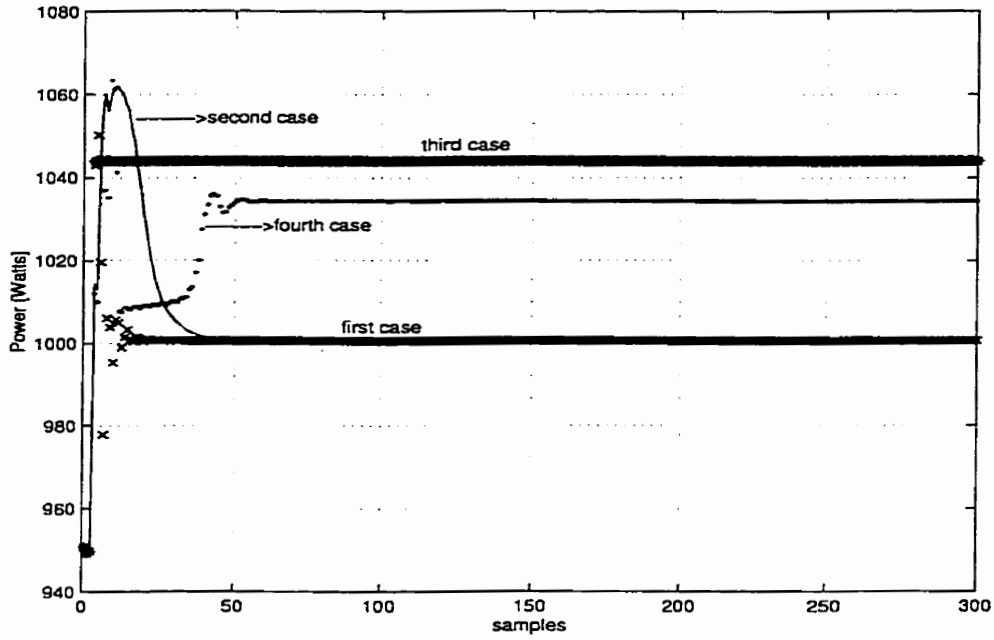


Figure 5.16: The evolution of the *RF power* for the closed-loop system.

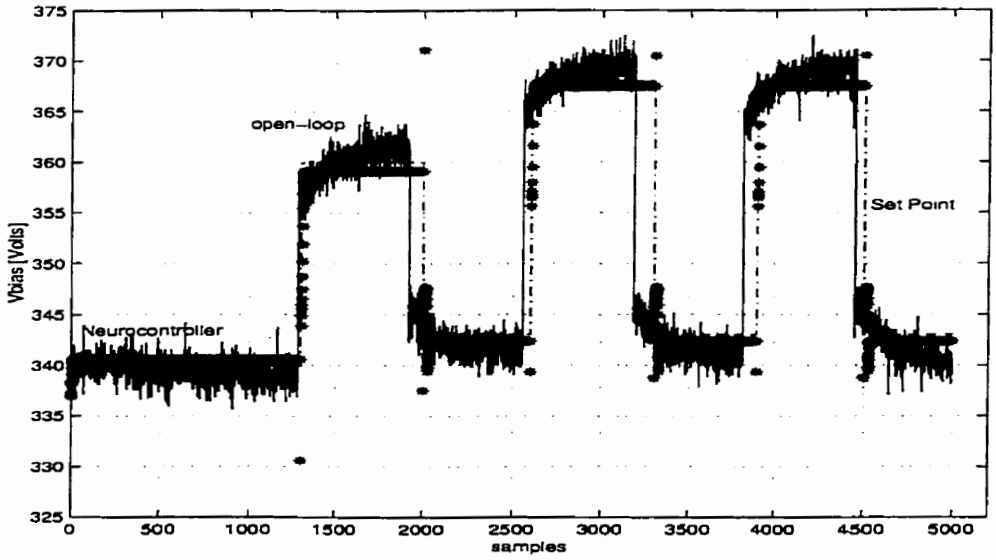


Figure 5.17: The performance comparison results for the V_{bias} for the open-loop and closed-loop system with the neurocontroller.

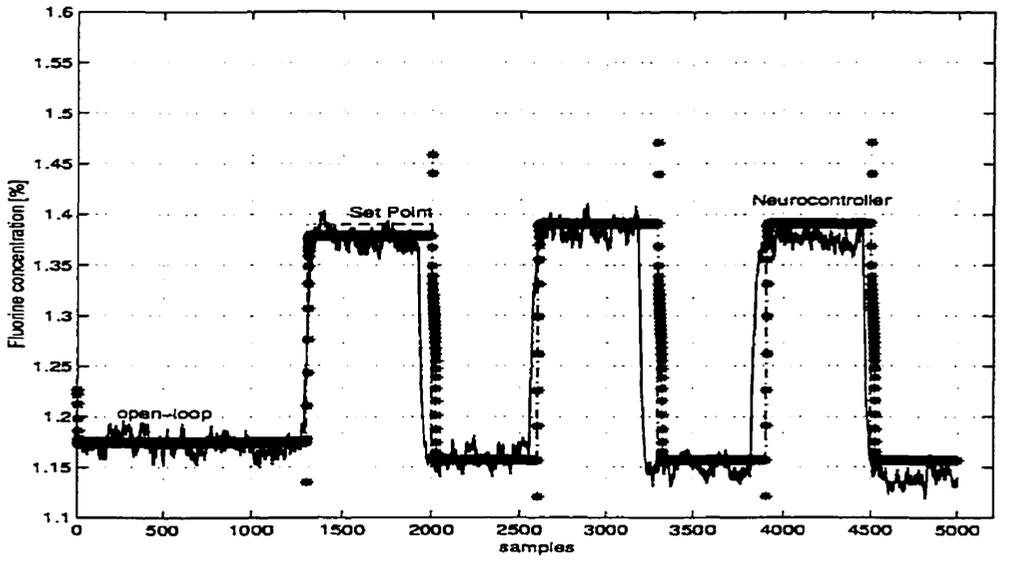


Figure 5.18: The performance comparison results for $[F]$ for the open-loop and closed-loop system with the neurocontroller.

5.4 Summary

In this chapter, we have developed four nonlinear neuromodels for the PGS process as well as four neurocontrollers to control the plasma etching process. These neural networks are used to construct four real-time neurocontrol strategies that are capable of ensuring the stability of the RIE system in closed-loop and to perform a good tracking performance such as 1.36 [Volts] and 0.015 [%] steady-state error for V_{bias} and $[F]$, respectively. Following our validation results obtained by computing the standard deviation errors between the model output and measured output, we can conclude that the neuromodels (iii) and (iv) are accurate and can fit the experimental input-output data set quite satisfactorily. Our results for identification of the nonlinear inverse dynamics of the process through neurocontrollers indicate that they are able to be valid in a large operating range and are capable of capturing the nonlinearities of the actuators and of the plant as compared to linear models which are valid in only a small operating region. Among the important contributions of this chapter are

the development of neural network architectures used for obtaining a good fit with the experimental input-output data set. The learning algorithm used for determining the neural network parameters (weights) is the dynamic/static backpropagation with off-line Levenberg-Marquardt algorithm. From the simulation results we observe that the third neurocontrol structure yields the best performances in both learning and generalization and is capable of reaching the steady-state error goal (1.36 [Volts] for V_{bias} and 0.015 [%] for $[F]$) relatively faster than the other cases (10 samples). The real-time feedback neurocontrollers manipulate important gas species (*fluorine concentration* $[F]$ and induced *dc Voltage* $[V_{bias}]$) across the reactor electrodes in the RIE system. The controlled species were measured by quadrupole mass spectrometry with direct effect on the outcome of the performance variables. This reveals the feasibility of our neurocontroller design, namely an *in-situ* sensor-based controller is capable of reducing the process variations. The simulation results provide good performance with the minimum absolute error and reveal that the third real-time neurocontroller strategy seems to be the most suitable to control the PGS in closed-loop. Since two of the three input vectors are delayed plant output, the neurocontroller can act very quickly to changes in the plant output, thus yielding precise and accurate response with good tracking performance. The neurocontrol strategies developed in this chapter can be used as a tool for evaluating complicated gas kinetics.

Chapter 6

Multi-Input Multi-Output Robust Control

The real problem in robust multivariable feedback control system design is to synthesize a control law which maintains system response and error signals to within prespecified tolerances despite the effects of uncertainty in the system. Uncertainty is a major issue in most control system design and may take many forms, but among the most important ones one could mention noise/disturbance signals, transfer function modelling errors, and unmodeled nonlinearities. This motivates one to seek a quantitative measure for the size of the uncertainty, using different tools like H^2 and H^∞ norms. The multivariable nature of a problem introduces another aspect that can be accounted for through the use of singular value gain measure. The singular values of the transfer matrix $G(z)$ of the PGS given by (3.21), and denoted by $\sigma_i(j\omega)$, are functions of frequency ω (sometimes referred to as the principal gains of $G(z)$), where the complex variable z is given by $z = exp(j\omega T_s)$, and T_s being the sampling period for the plasma RIE process. In MIMO feedback control system design several performances and robust stability requirements can be expressed in terms of specifi-

cations on the maximum singular values of a particular closed-loop transfer matrix. In fact, the gain of a multivariable system (measured in terms of the induced matrix norm) is between the smallest and largest principal gains, $\sigma_{min}(j\omega)$ and $\sigma_{max}(j\omega)$ [36]. Most of the conditions required on the singular values of a system are similar to those needed on the gain of a scalar system. For example, for a negative feedback system with plant $G(z)$ and controller $F(z)$, for "good" performance $\sigma_{max}((I+GF)^{-1})$ and $\sigma_{max}((I+GF)^{-1}G)$ should be small, particularly at low frequencies [36]. Similarly for good robust stability properties $\sigma_{max}(F(I+GF)^{-1})$ and $\sigma_{max}(GF(I+GF)^{-1})$ should be small, normally in the high-frequency range, where uncertainty is higher. The loop-shaping design rules are similar to those for a SISO system but take into account the mapping from matrix to scalar quantities through the singular values. For a SISO system, when the loop gain is greater than unity, the sensitivity function:

$$S = (I+GF)^{-1} \tag{6.1}$$

tends to zero and the closed-loop transfer function tends to unity. For MIMO control systems the tracking error signal e, the control u and the plant output y are weighted by transfer matrix specifications $W_1(z)$, $W_2(z)$, and $W_3(z)$, which play an important role in ensuring the desired disturbance attenuation, and the desired stability margins for the multivariable feedback design subject to additive and multiplicative plant perturbations [36]. In order to quantify the multivariable stability margins and performance of the system, the singular values of the following sensitivity matrices are utilized:

$$S(z) = (I+G(z)F(z))^{-1} \tag{6.2}$$

$$R(z) = F(z)(I+G(z)F(z))^{-1} \tag{6.3}$$

$$T(z) = G(z)F(z)(I + G(z)F(z))^{-1} \tag{6.4}$$

The two matrices $S(z)$ and $T(z)$ are known as the sensitivity and complementary sensitivity functions, respectively, and the matrix $R(z)$ as control sensitivity function. The singular value Bode plots of each of these transfer matrices play an important role in multivariable robust control design. The singular values of the matrix $S(z)$ determines disturbance attenuation, since $S(z)$ is in fact the closed-loop transfer matrix from the additive disturbance to the plant output y. In [36], it is shown that a disturbance attenuation performance specification may be expressed as

$$\sigma_{max}(S(exp(j\omega T_s))) \leq |W_1^{-1}(exp(j\omega T_s))| \tag{6.5}$$

where $|W_1^{-1}(exp(j\omega T_s))|$ is the desired disturbance attenuation factor, and T_s is the sampling period. Allowing $W_1(exp(j\omega T_s))$ to depend on frequency ω enables one to specify a different attenuation factor for a given frequency ω.

Furthermore, we can specify the stability margins of the system by the singular value inequalities

$$\sigma_{max}(R(exp(j\omega T_s))) \leq |W_2^{-1}(exp(j\omega T_s))| \tag{6.6}$$

$$\sigma_{max}(T(exp(j\omega T_s))) \leq |W_3^{-1}(exp(j\omega T_s))| \tag{6.7}$$

where $|W_2(exp(j\omega T_s))|$ and $|W_3(exp(j\omega T_s))|$ are the respective sizes of the largest anticipated additive and multiplicative plant perturbations. It is a common practice to lump the effects of all the plant uncertainties into a single fictitious multiplicative perturbation, so that the overall control design requirements may be alternatively expressed by the following singular value inequalities

$\frac{1}{\sigma_1(S(exp(j\omega T_s)))} \geq |W_1(exp(j\omega T_s))|$

$$\sigma_i(T(exp(j\omega T_s)))^{-1} \leq |W_3^{-1}(exp(j\omega T_s))|$$

To choose the design specifications represented by the weighting functions $W_1(z)$ (performance specifications such as disturbance attenuation) and $W_3(z)$ (robustness specifications such as roll-off frequency, stability margin), the 0 db crossover frequency in the Bode plot of $W_1(exp(j\omega T_s))$ must be sufficiently below the 0 db crossover frequency of $W_3^{-1}(exp(j\omega T_s))$. More precisely, we require

$$\sigma_{max}(W_1^{-1}(exp(j\omega T_s))) + \sigma_{max}(W_3^{-1}(exp(j\omega T_s))) < 1 \quad \forall\omega \tag{6.8}$$

The recently developed H^∞, frequency-weighted LQG, LQG loop transfer recovery (LQG/LTR), and μ-synthesis theories have made multivariable loop shaping a standard technique. The H^∞ theory provides a direct, reliable procedure for synthesizing a controller that optimally satisfies singular-value loop shaping specifications. The frequency-weighted LQG optimal synthesis theory known as H^2 theory and LQG/LTR lead to somewhat less direct, but nonetheless highly effective iterative procedures for manipulating the singular value Bode plots to satisfy singular value loop shaping specifications.

6.1 Robust Controller Architecture and State-Space Representation

The structure of the robust control system for the PGS in closed-loop is represented in Figure 6.1. In this structure we use a compensator after the controller in order to decrease the amplitude of the oscillations during the transient phase for which we

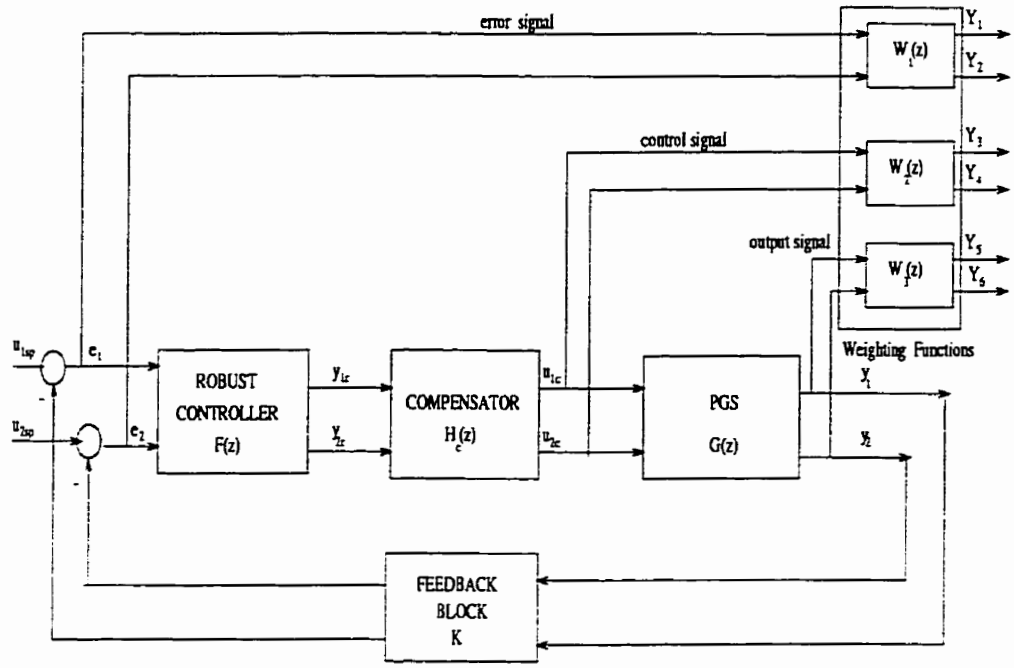


Figure 6.1: Robust controller architecture for the PGS process.

select the transfer function of the form:

$$H_c = \begin{bmatrix} k_{1c}\frac{z+a_1}{z+b_1} & 0 \\ 0 & k_{2c}\frac{z+a_2}{z+b_2} \end{bmatrix} \tag{6.9}$$

The proportional feedback block used to control independently each output has coefficients given by the matrix:

$$K = \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix} \tag{6.10}$$

The H^2 and H^∞ synthesis methods are especially powerful tools for designing robust multivariable feedback control systems to achieve singular-value shaping specifications.

By defining:

$U_{sp} = \begin{bmatrix} u_1 & u_2 \end{bmatrix}^T$, $y(k) = \begin{bmatrix} y_1(k) & y_2(k) \end{bmatrix}^T$, $Y_{sp} = \begin{bmatrix} Y_1 & Y_2 & Y_3 & Y_4 & Y_5 & Y_6 \end{bmatrix}^T$,

$e(k) = U_{sp} - y = \begin{bmatrix} e_1(k) & e_2(k) \end{bmatrix}^T$, $u_c(k) = \begin{bmatrix} u_{1c} & u_{2c} \end{bmatrix}^T$, and

$y_c(k) = \begin{bmatrix} y_{1c}(k) & y_{2c}(k) \end{bmatrix}^T$, as the reference input vector, output vector, weighted output vector, error vector, controller output vector and plant input vector, respectively, and letting $T_{Y_{sp}U_{sp}}(z)$ denote the transfer matrix from U_{sp} to Y_{sp}, the H^2 and H^∞ discrete norms (corresponding to Figure 6.1) are defined as follows:

H^2 norm:

$$\left\| T_{Y_{sp}U_{sp}} \right\|_2 = \sigma_{max}(T_{Y_{sp}U_{sp}}(z)) \tag{6.11}$$

H^∞ norm:

$$\left\| T_{Y_{sp}U_{sp}} \right\|_\infty = \sup_{|z=1|} \sigma_{max}(T_{Y_{sp}U_{sp}}(z)) \tag{6.12}$$

The PGS plant $G(z) = C_G(zI - A_G)^{-1}B_G + D_G$ is "augmented" with weighting transfer matrices $W_1(z) = C_{w_1}(zI - A_{w_1})^{-1} + D_{w_1}$, $W_2(z) = C_{w_2}(zI - A_{w_2})^{-1} + D_{w_2}$, and $W_3(z) = C_{w_3}(zI - A_{w_3})^{-1} + D_{w_3} + P_n z^n + ... + P_1 z + P_0$ (possibly improper) penalizing the error signal, control signal and output signal, respectively (as in the block diagram shown in Figure 6.2) such that the augmented system can be represented in the following state-space representation form:

$$x_A(k+1) = Ax_A(k) + B_1 \begin{bmatrix} u_1(k) \\ u_2(k) \end{bmatrix} + B_2 \begin{bmatrix} u_{1c}(k) \\ u_{2c}(k) \end{bmatrix} + w(k) \tag{6.13}$$

$$Y_{sp}(k) = C_1 x_A(k) + D_{11} \begin{bmatrix} u_1(k) \\ u_2(k) \end{bmatrix} + D_{12} \begin{bmatrix} u_{1c}(k) \\ u_{2c}(k) \end{bmatrix} \tag{6.14}$$

$$e(k) = C_2 x_A(k) + D_{21} \begin{bmatrix} u_1(k) \\ u_2(k) \end{bmatrix} + D_{22} \begin{bmatrix} u_{1c}(k) \\ u_{2c}(k) \end{bmatrix} + v(k) \tag{6.15}$$

with the closed-loop transfer matrix $P(z) = \begin{bmatrix} P_{11}(z) & P_{12}(z) \\ P_{21}(z) & P_{22}(z) \end{bmatrix}$, where $P_{11}(z) = \begin{bmatrix} W_1(z) \\ 0 \\ 0 \end{bmatrix}$, $P_{12}(z) = \begin{bmatrix} -W_1(z)G(z) \\ W_2(z) \\ W_3(z) \end{bmatrix}$, $P_{21}(z) = I$, $P_{22}(z) = -G(z)$, respectively, and $x_A \in R^{n_G+n_{W_1}+n_{W_2}+n_{W_3}}$ (n_G, n_{W_1}, n_{W_2}, n_{W_3} represents the minimal realization order of the PGS plant and of the weighting transfer matrices, respectively) [36]. Now, if we denote by

$$\tilde{C}_G = P_0 C_G + P_1 C_G A_G + ... + P_n C_G A_G^{n-1} \tag{6.16}$$

$$\tilde{D}_G = P_0 D_G + P_1 C_G B_G + ... + P_n C_G A_G^{n-2} B_G \tag{6.17}$$

the matrices A, B_1, B_2, C_1, C_2, D_{12}, D_{21}, D_{22} can be expressed by the following expressions [36]:

$$A = \begin{bmatrix} A_G & 0 & 0 & 0 \\ -B_{w_1} C_G & A_{w_1} & 0 & 0 \\ 0 & 0 & A_{w_2} & 0 \\ B_{w_3} C_G & 0 & 0 & A_{w_3} \end{bmatrix}, \; B_1 = \begin{bmatrix} 0 \\ B_{w_1} \\ 0 \\ 0 \end{bmatrix}, \; B_2 = \begin{bmatrix} B_G \\ -B_{w_1} D_G \\ B_{w_2} \\ B_{w_3} D_G \end{bmatrix},$$

$$C_1 = \begin{bmatrix} -D_{w_1} C_G & C_{w_1} & 0 & 0 \\ 0 & 0 & C_{w_2} & 0 \\ \tilde{C}_G + D_{w_3} C_G & 0 & 0 & C_{w_3} \end{bmatrix}, \; C_2 = \begin{bmatrix} -C_G & 0 & 0 & 0 \end{bmatrix}, \; D_{11} = \begin{bmatrix} D_{w_1} \\ 0 \\ 0 \end{bmatrix},$$

$$D_{12} = \begin{bmatrix} -D_{w_1} D_G \\ D_{w_2} \\ \tilde{D}_G + D_{w_3} D_G \end{bmatrix}, \; D_{21} = I, \; D_{22} = -D_G,$$

The closed-loop transfer matrix $T_{Y_{sp}U_{sp}}$, known as weighted mixed sensitivity, is given

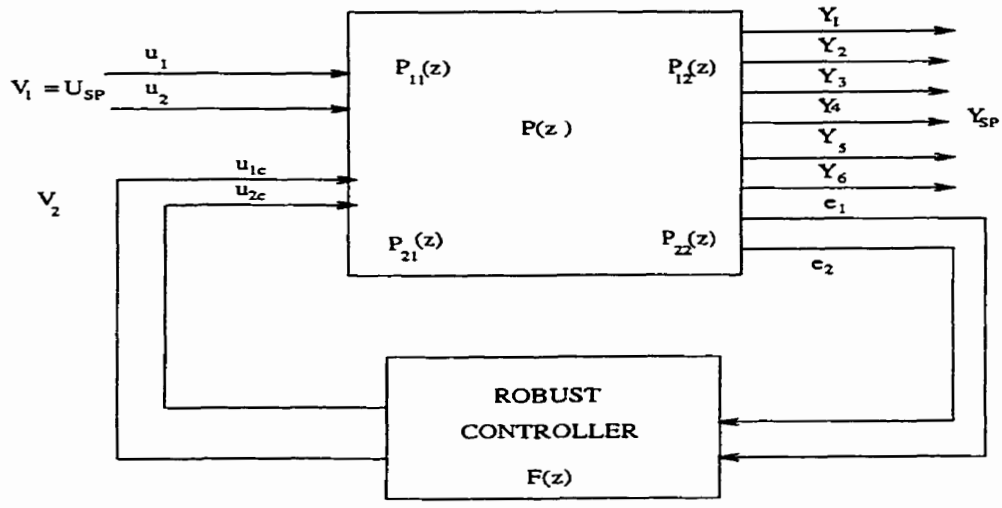


Figure 6.2: The simplified block scheme of the augmented plant.

by:

$$T_{Y_{sp}U_{sp}} = \begin{bmatrix} W_1(I+GF)^{-1} \\ W_2F(I+GF)^{-1} \\ W_3F(I+GF)^{-1} \end{bmatrix},$$

and which has a significant role in designing for a robust control for the closed-loop system.

6.2 Optimal Robust Control Problems

With respect to the definitions (6.11)-(6.12) and the state-space realization of the augmented plant (6.13)-(6.15) we are now in a position to formulate the H^2 and H^∞ optimal design problems as follow:

Given a state-space realization of the "augmented" PGS plant $P(z)$, find a stabilizing feedback control law:

$$V_2(z) = \begin{bmatrix} u_{1c}(z) \\ u_2c(z) \end{bmatrix} = F(z) \begin{bmatrix} e_1(z) \\ e_2(z) \end{bmatrix} \tag{6.18}$$

subject to additive and multiplicative plant perturbations such that the H^2 or H^∞ discrete norms of the closed-loop transfer matrix

$$T_{Y_{sp}U_{sp}}(z) = P_{11}(z) + P_{12}(z)(I - F(z)P_{22}(z))^{-1}F(z)P_{21}(z) \tag{6.19}$$

are sufficiently "small" (less than or equal to one), i.e.,

H^2 optimal control:

$$min_{F(z)} \left\| T_{Y_{sp}Usp}(z) \right\|_2 \leq \mu \leq 1$$

H^∞optimal control:

$$min_{F(z)} \left\| T_{Y_{sp}Usp}(z) \right\|_\infty \leq \mu \leq 1 \tag{6.20}$$

standard H^∞ control (*Small-Gain problem*):

$$\left\| T_{Y_{sp}U_{sp}}(z) \right\|_\infty \leq 1 \tag{6.21}$$

To solve the above problems, we use the design procedure presented in [36] and the weighting functions $W_1(z)$,$W_2(z)$, $W_3(z)$ (design specifications) to augment the plant $G(z)$. The $H^2 - norm\ optimal\ control\ problem$ is equivalent to a conventional *Linear Quadratic Gaussian optimal control problem* using the following steps:

(i) Linear Quadratic Estimator Design (LQE) (Kalman filter design)

This procedure computes the estimate of the augmented PGS state, $\hat{x}(k)$, using a steady-state Kalman filter [36] with the dynamic state equation

$$\hat{x}(k+1) = A\hat{x}(k) + B_1 v_1(k) + B_2 v_2(k) + K_e[e(k) - C_2\hat{x}(k) - D_{21}v_1 - D_{22}v_2(k)] \quad (6.22)$$

where K_e is the Kalman gain matrix and is given by

$$K_e = (SC_2^T + B_1 D_{21}^T + B_2 D_{22}^T)(D_{21}D_{21}^T)^{-1} \quad (6.23)$$

$$SA^T + AS - (SC_2^T + R_{wv})R_v^{-1}(C_2 S + R_{wv}^T) + Q_w = 0 \quad (6.24)$$

where S is the unique positive definite solution of the above algebraic Riccati equation, and $v_1(k) = \begin{bmatrix} u_1(k) & u_2(k) \end{bmatrix}^T$, $v_2(k) = \begin{bmatrix} u_{1c}(k) & u_{2c}(k) \end{bmatrix}^T$. The Kalman filter generates a state estimate that minimizes the variance of the estimation error where R_v and Q_w are the covariance matrices with correlated white plant noise $w(k)$ and white measurement noise $v(k)$, respectively, having a cross-correlation matrix R_{wv}.

(ii) Linear Quadratic Regulator Design (LQR) (Full-state feedback design)

This procedure computes a feedback gain matrix K such that when

$$v_2(k) = -K\hat{x}(k) \quad (6.25)$$

is applied to the augmented full-order system (6.13), the resulting closed-loop system is stable, i.e., all the poles of the closed-loop system are inside the unit circle. This

control is selected to minimize the following quadratic function:

$$J = \sum_{k=0}^{\infty} [x(k)^T Q x(k) + v_2(k)^T R v_2(k)] \tag{6.26}$$

The feedback gain matrix K [36] is then given by:

$$K = (D_{12}^T D_{12})^{-1}(B_2^T P + D_{12}^T C_1) \tag{6.27}$$

where P is the unique positive definite solution to the discrete time matrix algebraic Riccati equation:

$$A^T P + PA - PB_2 R^{-1} B_2^T P + Q = 0 \tag{6.28}$$

The H^2 optimal controller has a transfer matrix of the following form:

$$F(z) = K(zI - A + K_e C_2 + B_2 K - K_e D_{22} K)^{-1} K_e \tag{6.29}$$

The disturbance attenuation specifications and the stability margin specifications given by the inequalities (??)-(6.8), may be expressed into a single infinity norm specification of the form

$$\left\| T_{Y_{sp}U_{sp}} \right\|_\infty < 1 \tag{6.30}$$

where

$$T_{Y_{sp}U_{sp}} = \begin{bmatrix} W_1 S \\ W_3 T \end{bmatrix} \tag{6.31}$$

This function is known as the mixed-sensitivity cost matrix since it penalizes both the sensitivity matrix S and complementary sensitivity matrix T. The mixed-sensitivity approach for the robust control design is a direct and an effective way of

achieving multivariable loop shaping.

To find a stabilizing controller $F(z)$ such that $\left\|T_{Y_{sp}U_{sp}}\right\|_\infty \leq 1$, we will first start with H^2 synthesis and then apply H^∞ techniques to determine the actual design limitations. The $T_{Y_{sp}U_{sp}}$ singular value Bode plot associated with each design will indicate how close the design is to the specifications. For the design problem, in the structure of the weighting matrices (W_1, in general), one parameter appears explicitly, denoted by γ, with respect to which one has to iterate several times until a suitable design is obtained. In this case, the standard H^∞control problem is referred to as the H^∞*Small-Gain problem*. Both $H^\infty$and H^2 synthesis are often used together, with H^2 synthesis being used as a first try to get a sense for what level of performance is achievable. Then, an H^∞ performance criterion is selected and the H^∞ synthesis theory is used to do the final design work [36]. If one imposes overly demanding design requirements then the minimal achievable H^∞ norm may be greater than one, in which case there is no solution to the standard H^∞ control problem (that is to find a controller for which $\left\|T_{Y_{sp}U_{sp}}\right\|_\infty \leq 1$). In [36], four conditions for the existence of a solution to the standard H^∞ control problem are provided, namely

(*i*) D_{11} is sufficiently small, i.e., there must exist a constant feedback control law $F(z)$ such that the closed-loop D matrix satisfies $\sigma_{max}(D) < 1$

(*ii*) Riccati control matrix P given by (6.28) is positive semidefinite

(*iii*) Ricatti observer matrix S given by (6.24) is positive semidefinite

(*iv*) $\lambda_{max}(PS) < 1$

The optimal value for the parameter γ, denoted by γ_{opt}, is given by solving the *Small-Gain* H^∞ *problem*. To determine and observe the effects of the compensator,

the feedback block and the weighting functions, the following scenarios to the control structure given in Figure 6.1 are now investigated, specifically

(*i*) without the compensator

(*ii*) with the compensator

(*iii*) with the compensator and weighted control

(*iv*) with the compensator and weighted error, control and output.

6.3 Robust Control Structures

Case 1: The Robust Control Structure without Compensator

For this case the parameters k_1 and k_2 in (6.10) are tuned by trial and error to get the smallest possible steady-state tracking errors, resulting in $k_1 = 0.001$, $k_2 = 0.005$, $W_1 = \gamma * I_2$, and $W_3 = I_2$. Using the Matlab subroutine for H^∞, the optimal value for γ is obtained as γ_{opt}= 1.414. The results of the H^∞ synthesis are presented in Figures 6.3-6.6 without changes in the set points and in Figures 6.7-6.8 for changes in the set points. From these simulations, we can observe that very good results are obtained in stabilizing the system after a brief settling time, but at the cost of a high transitory oscillations. The controller structure is very robust to changes in the set points, and the system is stabilized with very small tracking errors.

Case 2: Robust Control Structure with Compensator

This structure is used to reduce the amplitude of the transient oscillations based on the integral effect of the compensator structure. The values for the parameters are determined by trial and error using the Matlab subroutine for H^∞ technique to get the smallest possible steady-state tracking errors for the PGS in closed-loop as follows:

$k_1 = 0.3$, $k_2 = 0.07$, $a_1 = 0.102$, a_2 = -0.2, $b_1 = 0.3$, and b_2 = -0.51 (see equation(6.9)), and

$$K_c = \begin{bmatrix} 2 & 0 \\ 0 & 0.75 \end{bmatrix} \tag{6.32}$$

for the case with changes in the set points and $k_1 = 0.001$, $k_2 = 0.005$, a_1 = -0.302, a_2 = -0.33, $b_1 = 0.3$, and b_2 = -0.51, and

$$K_c = \begin{bmatrix} 2 & 0 \\ 0 & 0.75 \end{bmatrix} \tag{6.33}$$

for the case without changes in the set points. The weighting functions have the same structure as in the first case. Using the Matlab subroutine for H^∞ technique we get γ_{opt}= 1.414, which is the same as in the first case since the weighting functions are not changed. The simulation results are presented in Figures 6.3-6.6 for the case with no changes in the set points and in Figures 6.7-6.8 for the case with changes in the set points. From these simulations one may observe that the amplitude of the oscillations are reduced considerably when compared to the previous case.

Case 3: Robust Control Structure with Compensator and Weighted Control

The weighted control case attempts to optimize the control effort as much as possible to perform good tracking error with small oscillations. In this case the control is weighted by the weighting transfer matrix $W_2(z)$ which is chosen to ensure that the D_{12} matrix of $P(z)$ is full rank as a necessary condition for having a solution to the H^∞ control problem. This is achieved by tuning the parameters using the Matlab subroutine for H^∞technique with the following values:

$k_1 = 0.12$, $k_2 = 0.001$, $a_1 = -0.302$, $a_2 = -0.33$, $b_1 = 0.3$, $b_2 = -0.51$ and

$$W_2(z) = \begin{bmatrix} \frac{0.001(z+1)}{z-1} & 0 \\ 0 & \frac{0.001(z+1)}{z-1} \end{bmatrix} \tag{6.34}$$

$$K_c = \begin{bmatrix} 2.05 & 0 \\ 0 & 0.85 \end{bmatrix} \tag{6.35}$$

It is our conclusion that this scenario is perhaps the best control structure for yielding small tracking errors, fast response and high robustness to the changes in the set points. The simulation results are presented in Figures 6.3-6.6 with $\gamma_{opt} = 1.375$.

Case 4: Robust Control Structure with the Compensator and Weighted Error, Control and Output

In this case all the possible outputs of the PGS system, namely error, control and plant output are weighted by the appropriate weighting transfer matrices $W_1(z)$, $W_2(z)$, and $W_3(z)$. By example, now we seek for the following singular value design specifications:

(*i*) *Robustness Specifications (stability margin)* expressed by $W_3(z)$, chosen with respect to the requirements imposed by the application, and here in our case, assumed to perform, e.g., -40db/decade roll-off and at least -20db at 100 rad/sec.

(*ii*) *Performance Specifications* (disturbance attenuation) expressed by $W_1(z)$ so as to minimize the sensitivity function. Note that we have

$$W_1(z) = \begin{bmatrix} 0.015\gamma\frac{2.33z+1}{z-1} & 0 \\ 0 & 0.015\gamma\frac{2.33z+1}{z-1} \end{bmatrix} \tag{6.36}$$

$$W_2(z) = \begin{bmatrix} \frac{210(1.4z+1)}{z-1} & 0 \\ 0 & \frac{0.075(z-1)}{z-1} \end{bmatrix} \tag{6.37}$$

$$W_3(z) = \begin{bmatrix} \frac{1.6(z-1)^2}{(z+1)^2} & 0 \\ 0 & \frac{1.28(z-1)^2(1.5z+1)}{(z+1)^3} \end{bmatrix} \tag{6.38}$$

where these weighting matrices are selected to meet the design specifications for the sensitivity function S and the complementary sensitivity function T=$(I - S)$ (disturbance attenuation, stability margin) as well as the four existence conditions for a feedback control law as required in the standard H^{∞}control problem (see Section 4.2). The weighting matrix $W_2(z)$ is chosen to assure that the D_{12} matrix of $P(z)$ is full rank to solve the H^{∞} control problem. The parameter γ is initially set to 1 and later decreased until γ_{opt}= 0.0099487 is reached by the γ-synthesis algorithm. The values of the tuned parameters for this structure are given by:

$$K = \begin{bmatrix} 0.15 & 0 \\ 0 & 13.85 \end{bmatrix} \tag{6.39}$$

$$K_c = \begin{bmatrix} 0.95 & 0 \\ 0 & 0.75 \end{bmatrix} \tag{6.40}$$

a_1 =-0.82, a_2= -0.29, b_1 = 0.3, b_2= -0.51.

The closed-loop system performance of these robust control strategies, namely the settling time (t_r), rise time (t_i), overshoot ($\%OS$), steady-state error (ϵ) and the convergence of the algorithm for V_{bias} and $[F]$ are presented in Table 6.1.

The results of the simulations are presented in Figures 6.3-6.8. In Figure 6.7 the V_{bias} doesn't reach the steady-state after 5200 samples, however this state will eventually reach a steady-state after a long time, and so for this scenarios the

Structure	t_r[s][	t_i[s]	%OS	ϵ	Conv.
First	3/5	0.4/-	850 [Volts]/55 [%]	0.35 [Volts]/0.001 [%]	good
Second	3/5	0.5/-	550 [Volts]/55 [%]	0.36 [Volts]/0.001 [%]	very good
Third	3/4.5	0.4/-	250 [Volts]/55 [%]	0.34 [Volts]/0.0009 [%]	very good
Fourth	0.5/ 5	-/-	- /55%	300 [Volts]/0.0009 [%]	bad

Table 6.1: The closed-loop performance system of the control robust strategies.

performance is not as good as desired. These results illustrate the robustness of this control strategy as the set points are varied. This algorithm is among the most effective control strategies for the highly nonlinear PGS control system.

6.4 Summary

In this chapter we have developed a *MIMO* real-time robust control strategy using standard H^∞ control techniques. The contributions of the compensator, the feedback block and the weighting matrices in the control structure depicted in Figure 6.1 are investigated, namely the closed-loop system

(i) without compensator

(ii) with compensator

(iii) with compensator and weighted control, and

(iv) with compensator and weighted error, control and *output.*

For each of the above cases we study the behaviour of the robust controller to changes in the set points. The results of the simulations indicate that the third structure yields the highest performance in terms of the tracking error in the steady-state, fast response and high robustness to changes in the set points in the presence of disturbances and parameter variations in the PGS system. This algorithm is among the most effective control strategies considered in this chapter and thesis for controlling a highly nonlinear PGS system.

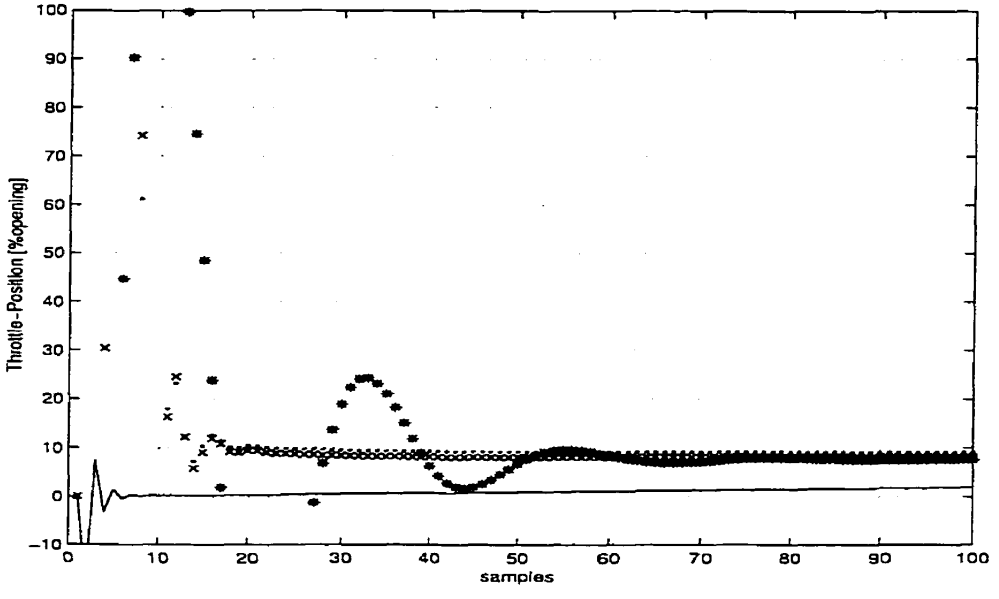


Figure 6.3: The performance of the robust controller on the *throttle position* for the closed-loop system.
Legend: The star designates the first case, the dot designates the second case, the "x" designates the third case, and the solid line designates the fourth case.

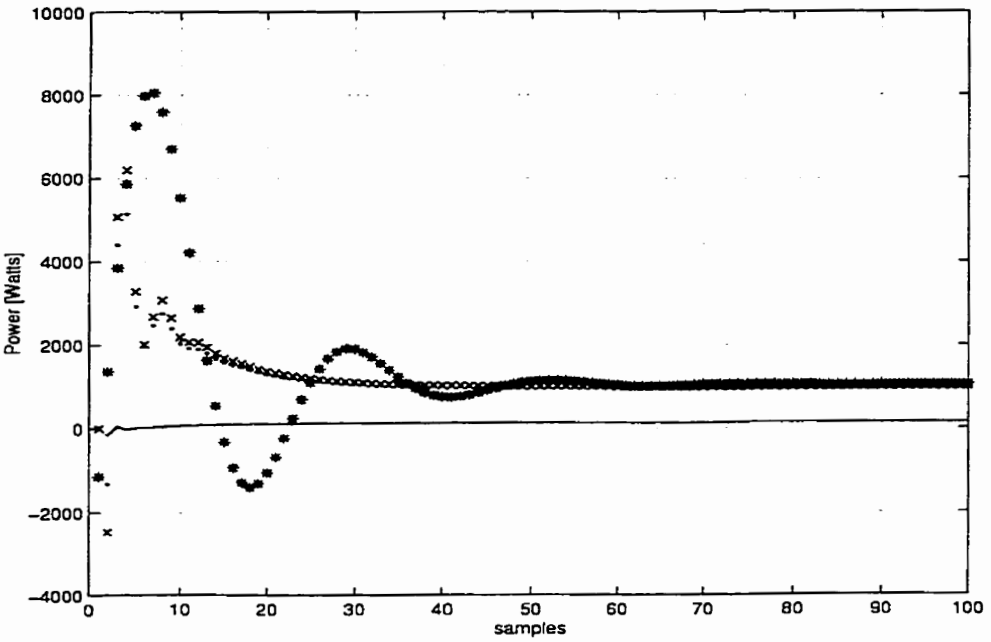


Figure 6.4: The performance of the robust controller on the *RF power* for the closed-loop system.
Legend: The star designates the first case, the dot designates the second case, the "x" designates the third case, and the solid line designates the fourth case.

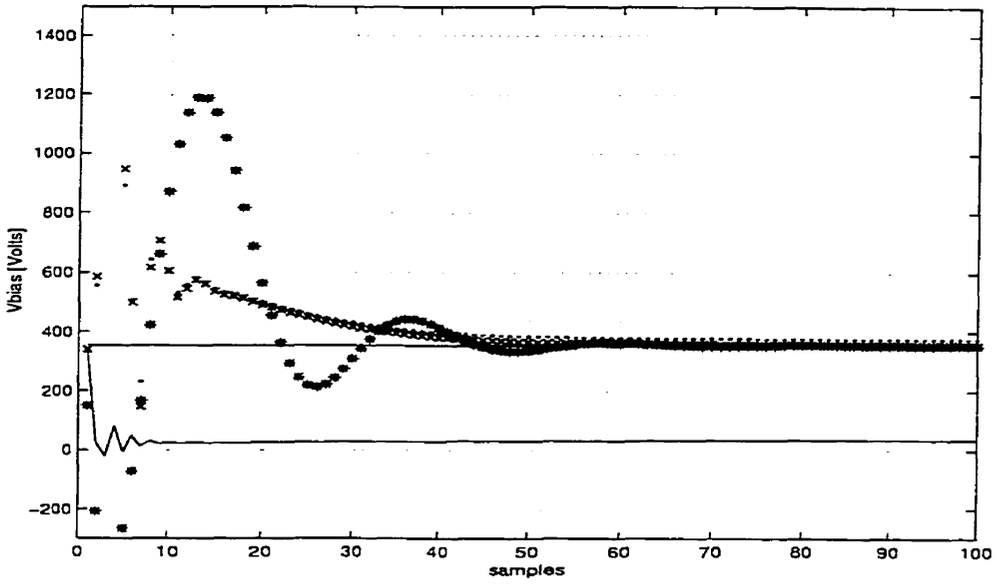


Figure 6.5: The performance of the robust controller on the V_{bias} for the closed-loop system.
Legend: The star designates the first case, the dot designates the second case, the "x" designates the third case, and the solid line designates the fourth case.

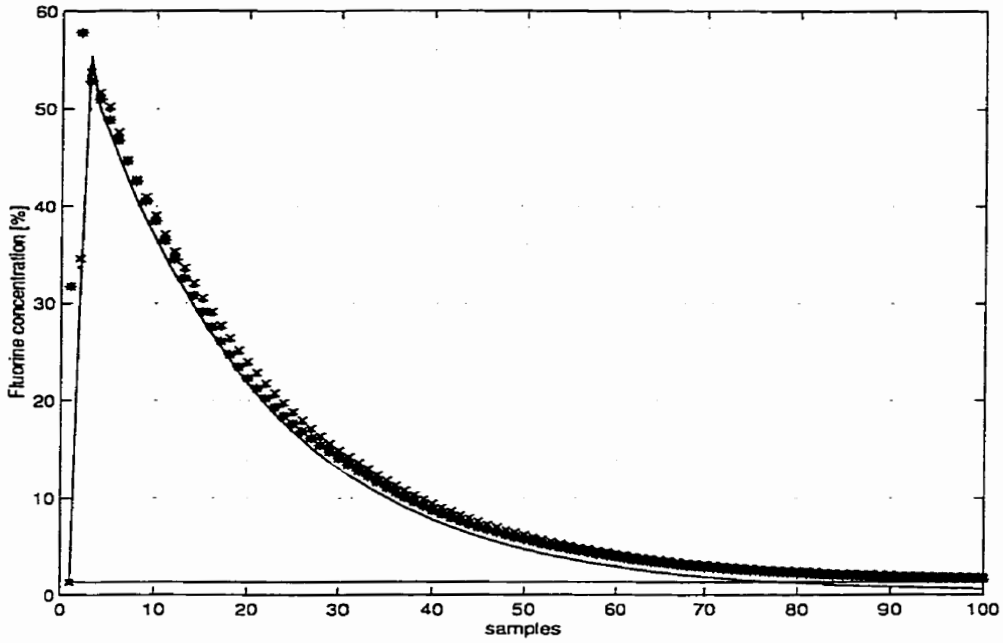


Figure 6.6: The performance of the robust controller on the *fluorine concentration* $[F]$ for the closed-loop system.
Legend: The star designates the first case, the dot designates the second case, the "x" designates the third case, and the solid line designates the fourth case.

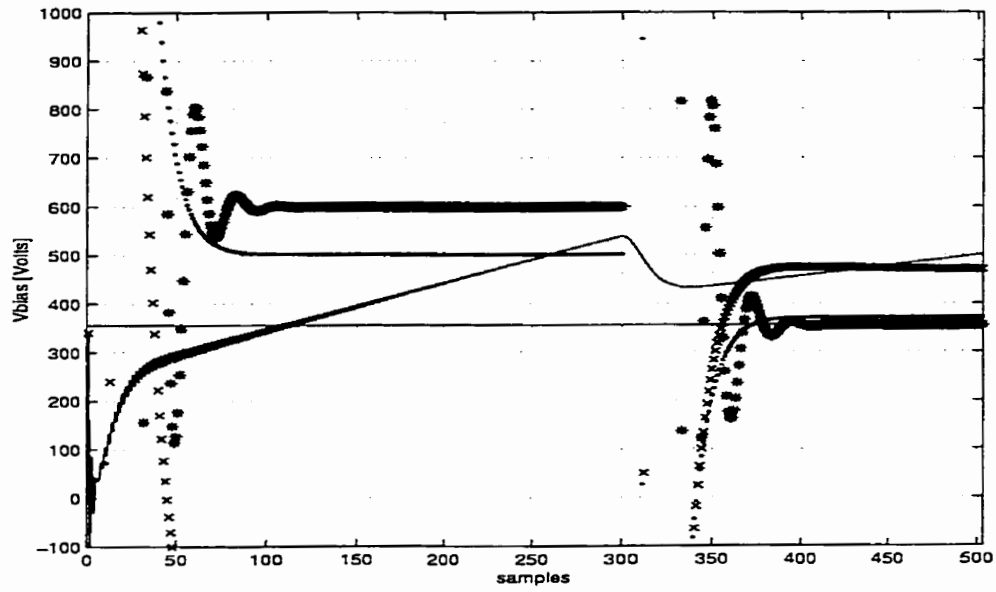


Figure 6.7: The performance of the robust controller on the V_{bias} for the closed-loop system.with changes in the set points.
Legend: The star designates the first case, the dot designates the second case, the "x" designates the third case, and the solid line designates the fourth case.

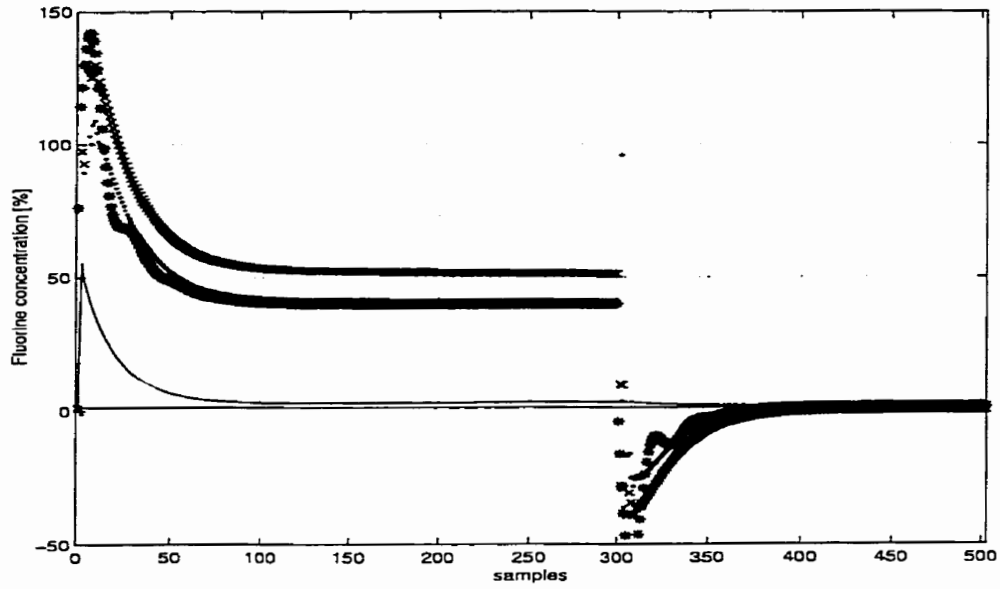


Figure 6.8: The performance of the robust controller on the *fluorine concentration* $[F]$ for the closed-loop system with changes in the set points.
Legend: The star designates the first case, the dot designates the second case, the "x" designates the third case, and the solid line designates the fourth case.

Chapter 7

Performance Comparison of the Proposed Real-Time Control Strategies

7.1 Wafer Performance

The real-time robust control strategies developed in this thesis use the process variables V_{bias} and $[F]$ with the attention being focused on modelling, control and analysis of the PGS system. In these control strategies the real-time controller maintains the desired V_{bias} and $[F]$ set points for the duration of an etch cycle. However, in order to have a criterion for comparing the performance of different control strategies, we have to translate the results obtained so far for the PGS system into performance of the wafer subsystem, namely etch rate or etch depth. The main objective here is to keep the etch depth at a desired target value in the face of variations in the process disturbances, namely CF_4 flow-rate disturbances, loading disturbances, and oxygen disturbances. In [21] the simplified Mogab-Flamm wafer model was used which as-

sumes that the etch rate is directly proportional to the *fluorine concentration* $[F]$, and the coefficient of the proportionality depends only on the operating point. Therefore, to study the etch rate performance of the wafer, or equivalently the etch depth, we can consider the performance of the PGS system in terms of $[F]$. To choose the "best" real-time control strategy we can select the controller that yields the best tracking error with respect to the fluorine concentration $[F]$ set points.

7.1.1 Experiments Results

The effect of the disturbances in [21] is studied at two operating points: operating point A (op A) at 1000 W, 20 mTorr, and 30 sccm, and operating point B (op B) at 1000 W, 40 mTorr, and 40 sccm. Operating point A, because of its lower pressure, uses much more ion bombardment for etching than operating point B. The ions collide less with other particles in the sheath and, therefore, are able to develop more kinetic energy. For each operating point, the load was varied from 1 to 5 wafers. The etch rate was measured on unpatterned wafers of 600 nm polysilicon on 32 nm silicon dioxide on silicon. This stack of materials provides a nice reflection from the incident reflectometry light sources, thereby enabling the acquisition of good reflectometry data [21]. Bare silicon wafers were used for additional loading. The etch rate for polysilicon is about the same as that for silicon. The change in etch depth as measured by the white light reflectometry data was used to calculate an average etch rate. The impact of the water vapour disturbance is also studied in [21]. The magnitude of the water vapour disturbance depend upon the humidity in the clean room, the polymer buildup on the chamber walls, and the time that the chamber was left open to load the wafers.

7.2 Performance Comparison and Feasibility of the Algorithms

Our efforts on real-time control have concentrated on feeding back plasma variables to attenuate the effects of process disturbances. Additional results on two-input, two-output controllers which regulate estimated fluorine concentration and bias voltage are reported in [1], [4]-[8], [37]. Also substantial contributions for modelling and control of RIE systems are found in [10], [12], [20], [2], and [18].

The performance achieved by our proposed real-time control strategies is compared by depicting the results in the same graph as shown in Figures 7.1-7.4. The legend used in these figures is as follows:

1. *PI self-tuning control* algorithm [A_1].
2. *LQG control* algorithm [A_2].
3. Intelligent control strategy [A_3].
4. Stochastic minimum variance adaptive control strategy [A_4].
5. Robust control strategy [A_5]

For each operating condition, we can see that the best performance for the V_{bias} as depicted in Figure 7.1 is obtained, in order, by the strategies A_3, A_5, A_1, A_2, and A_4, and for the fluorine concentration $[F]$, as depicted in Figure 7.3 by the strategies A_3, A_5, A_2, A_1, and A_4. For the A_2 and A_4 strategies, we have a small decrease in performance due to the stochastic environment (disturbance variations). By taking into account control effort, as depicted in Figures 7.5-7.6 for comparing the performances of these strategies, we can state that the best overall performance is achieved in order by the controllers A_5, A_3, A_1, A_2, and A_4. Note that by changing the operating condition this order may also change, therefore it is possible that one controller performs better than others in a given operating condition. As far as the evolution of the flu-

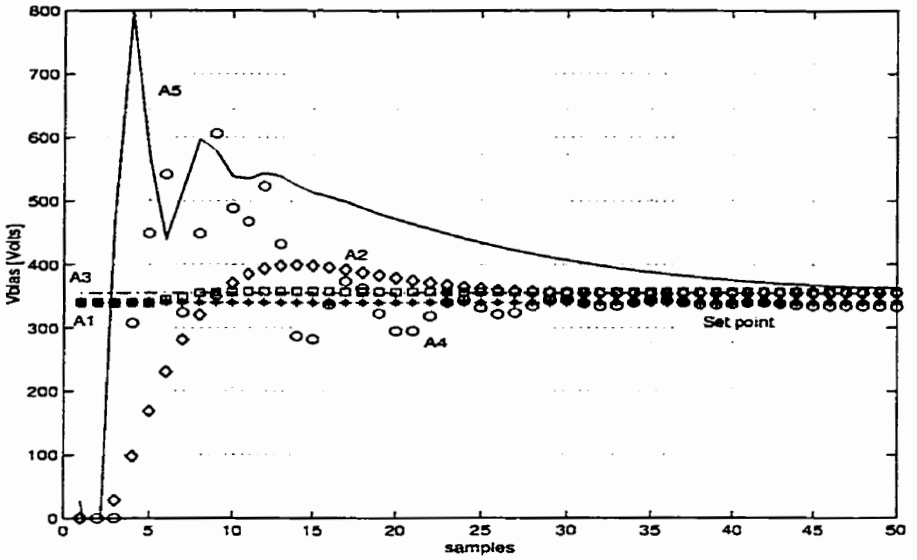


Figure 7.1: The evolution of the *dc* V_{bias} in closed-loop system for all the control strategies (the first period of the transient).

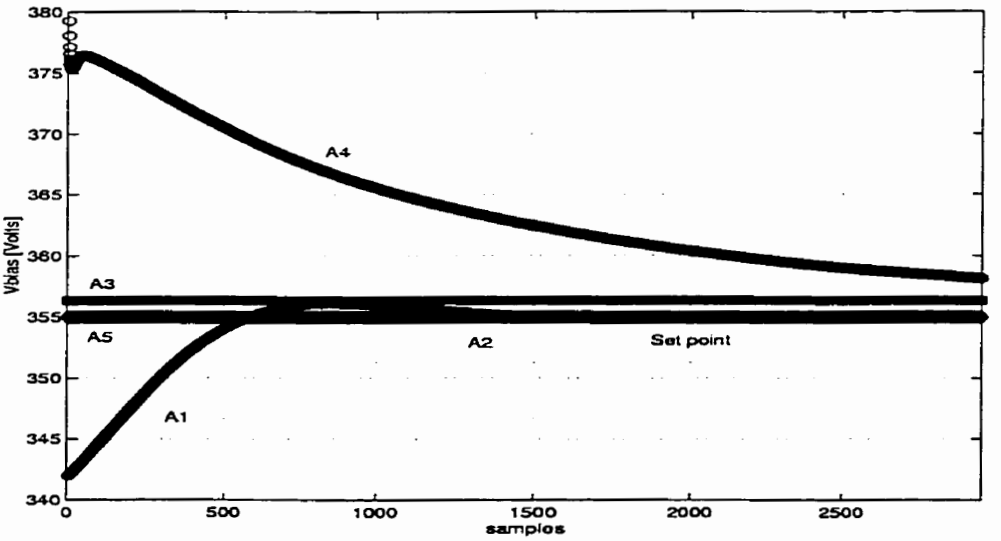


Figure 7.2: The evolution of the *dc* V_{bias} in closed-loop system for all the control strategies (the second period of the transient).

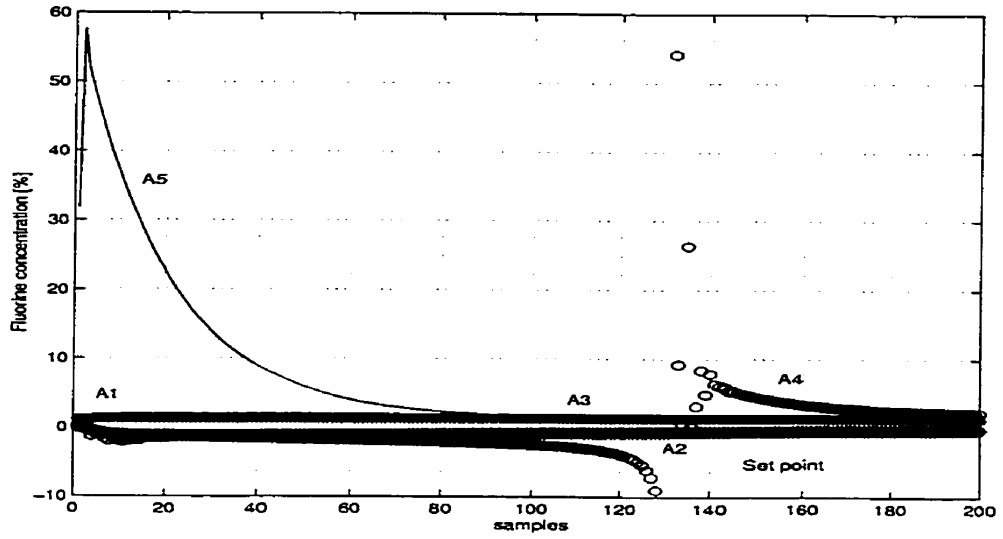


Figure 7.3: The evolution of the *fluorine concentration* $[F]$ in closed-loop for all the control strategies (the first period of the transient).

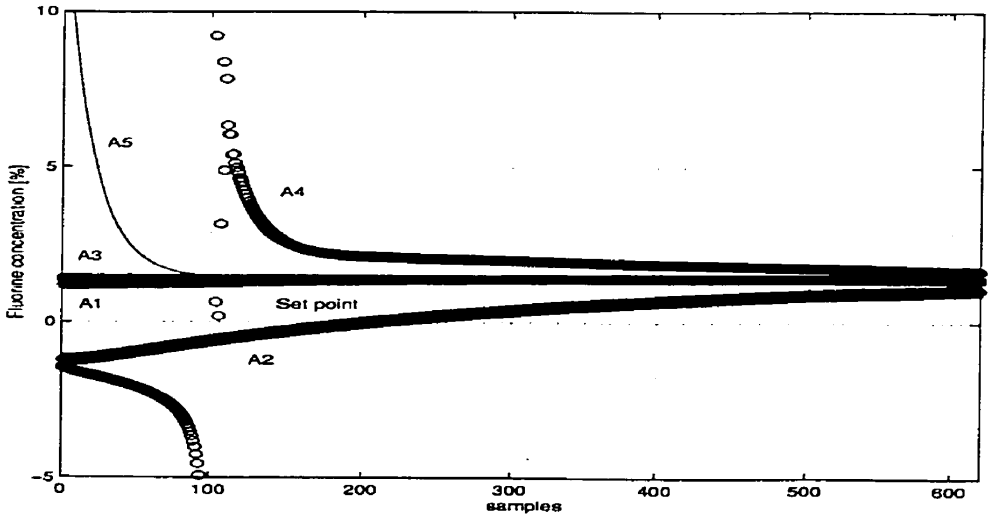


Figure 7.4: The evolution of the *fluorine concentration* $[F]$ in closed-loop for all the control strategies (the second period of the transient).

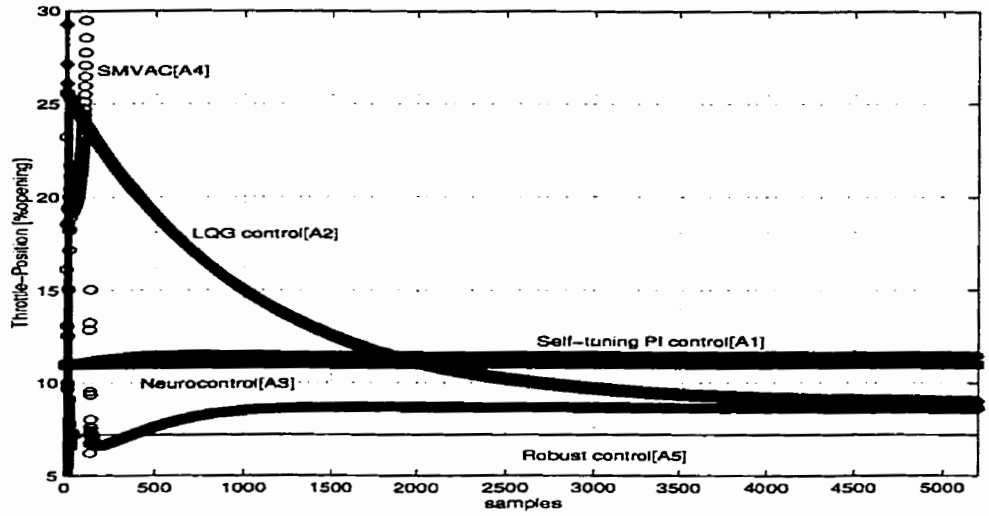


Figure 7.5: The evolution of the *throttle-position* in closed-loop for all the control strategies.

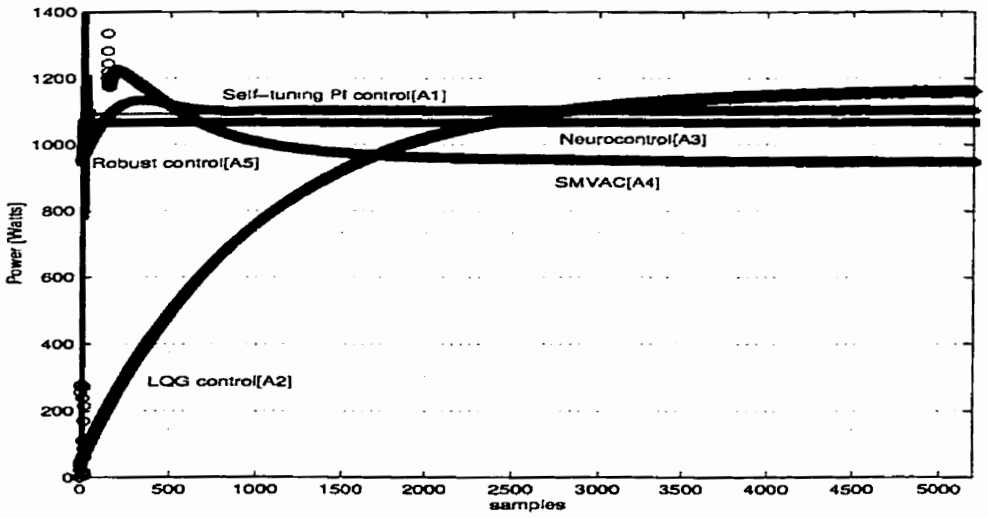


Figure 7.6: The evolution of the *power* in closed-loop for all the control strategies.

Algorithm	$V_{bias}/[F]$ t_r[sec]	V_{bias} e [Volts]	$[F]$ e [%]	Convergence
Self-tuning PI [A_1]	80/60	0.0015	0.00002	very good
LQG [A_2]	15/75	0.0014	0.00001	very good
Neurocontrol [A_3]	2/1	1.36	0.015	very good
SMVAC [A_4]	100/50	1.29	0.05	good
Robust control [A_5]	2.5/4.5	0.31	0.0009	very good

Table 7.1: Performance comparison for the algorithms A_1-A_5.

orine concentration for the A_1-A_5 cases is concerned, as shown in Figure 7.3, we can conclude that the best performance for the same operating condition is obtained in the following order: A_5, A_3, A_2, A_1, and A_4. Generally a hierarchical control strategy that is capable of determining at each sampling interval which controller is the most suitable choice to be employed will be of great use. Other performance indicators such as the settling time (t_r), steady-state error for V_{bias} and $[F]$ (e) and the convergence achieved by the algorithms $A_1 - A_5$ developed in this thesis are presented in Table 7.1 as a synthesis of the performance presented in details (settling time, rise time, overshoot, steady-state error and convergence) in the Tables 4.1-4.3, 5.2, 6.1. From this table and from the overall simulation results we could conclude that almost all the proposed algorithms have a good convergence, are computationally efficient due to their modular structure and recursive form with a relative small number of operations involved in the computation process, are feasible to be implemented in practice, and have very close steady-state errors. During the transient, the behaviour of the controllers are completely different as far as the oscillations, the settling time, the actuator efforts, the capability of working well in a large region around the operating point, are concerned.

The *self-tuning PI* control approach [A_1] is proposed as a standard controller easy to use in practice. The algorithm is developed for the *MIMO decoupled PGS model with pure delay* in comparison with [22], [25] which gave a general approach for

SISO systems, and [1] that studied the possibility of using a standard *PI controller* structure for MIMO coupled PGS model identification approach.

The *self-tuning PI* algorithm is simple, convergent, fast, computationally efficient, and is feasible to be implemented in practice, however it works well only in a small region around the operating point for which it was designed for. Generally, larger values of the model parameters will cause the controller to take longer to fine tune itself. The control action, therefore, depends on the predicted values of the one-step ahead model in time, and the algorithm works well when the number of parameters is small. As the number of parameters increases the initial estimates become increasingly important and poor estimates may result in an extended period of poor performance and this could even lead to instability.

Also, good performance is given by the stochastic minimum variance identification algorithm [A_4] when applied to the MIMO coupled RIE system having 12 unknown parameters. For comparison the approach in [32] is applied only to SISO systems with two unknown parameters. The simulation results depicted in Figures 4.24-4.27 show a fast adaptation and convergence. In the identification process, the highly fluctuating behaviour of the adaptive control signal contributes greatly to the excitation of the system dynamics required for a convergence. For a deterministic control environment, perfect tracking at steady-state can be obtained, since the control effort $u(k)$ is not being constrained by the disturbances. The identification part of the control strategy is computationally more efficient than the modified extended Kalman filter algorithm, because the state-space innovations model eliminates the computation of the Kalman gradient $\nabla_\theta K(\theta)$ with respect to θ via the Kalman filter gain algorithm. Also, in this case the execution of the Kalman filter for the state estimation is not needed since the Kalman gain $K(\theta)$ is directly parametrized by $\theta(k)$ considered as a Markov-Gauss process [32]. The control part of the proposed strategy uses the

per-interval performance index (4.54), which is computationally more efficient than the dynamic programming algorithm, and yields a higher performance compared to a general performance index. The algorithm is therefore feasible and can be easily implemented in practice in real-time, however it works well only in a small region of the operating point and is perhaps most sensitive to the deviations around the operating point.The LQG algorithm [A_2] has good convergence properties and can be implemented in real-time. It is computationally efficient and is similar in performance to the approach developed in [29], and was chosen in this thesis because we found it to be well adapted to the specifics of the plasma etching process (more disturbances, measurement and process noise contamination), and consequently operates properly in the stochastic environment.

All the three linear controllers A_1, A_2, and A_4 applied to the highly nonlinear MIMO plasma generation subsystem (PGS) can achieve their desired performance characteristics for only small regions around the operating points for which they were designed for. The transient behaviour at the beginning of the etch process may force the system into regions outside of which the linear controller can be applied safely.

Higher fidelity neuromodel architectures and real-time nonlinear neurocontrol strategies [A_3] for the PGS system in closed-loop are developed to improve substantially the plasma characteristics. In comparison with the structures developed in the plasma literature [2], [9], [19], [14] our approach is conceptually different, the algorithms are more efficient, faster, more accurate and more suitable to be implement in real-time for controlling the reactive ion etching processes. To implement these neurocontrol strategies in real-time, we need accurate real-time sensors and high speed microprocessors and parallel processing architectures to perform off-line training, on-line parameter adjustments, model processing, and feedback control processing.

Similar performance is obtained by using a real-time robust control strategy [A_5]

which responds more accurately to the variations in the plasma parameters, disturbances, and the changes in the set points. The proposed structure for this control strategy is introduced for the first time in the literature to control the performance of the reactive ion etching process. It seems that this algorithm is perhaps the best control structure yielding small tracking errors, fast response and high robustness to changes in the set points in presence of disturbances and parameter variations in the highly nonlinear PGS system.

7.3 Real-Time Control Integration in Multi-Input Multi-Output Reactive Ion Etching Systems

Our goal is to reduce the variance of the etch characteristics by integrating the real-time control of the plasma and process variables. The real-time controller A_5, developed in this thesis adjusts the reactor input variables to maintain the plasma parameters at constant set-point values. The idea is to use real-time plasma sensors to monitor key plasma parameters and design a real-time multivariable controller to control the PGS. The controlled species were measured by quadrupole mass spectrometry with direct effect on the outcome of the performance variables. This verifies the feasibility of our control strategy, in other words an *in-situ* sensor based controller is capable of reducing process variations. The results obtained confirm the utility of our proposed strategies to control the etching process despite the drift during an etch. However, these algorithms can also be used as a research tool for evaluating complicated gas kinetics. The controllers can drive the film properties, namely *etch depth, anisotropy, selectivity* and *uniformity* to desired regimes based on important gas species and dc induced voltage in an optimal fashion. Also, the mass flow controller offset experiments [21], [37] demonstrate the utility of real-time sensing and

control for disturbance rejection. At this time we are using the bias voltage, V_{bias}, signal from the RF matching network and an optical emission spectroscopy (with A_r as an actinometer) based measurement for the concentration of the main etchant species fluorine, $[F]$, as the key plasma variables. The real-time two-input, two-output controller A_5 is designed to provide set point tracking (on V_{bias} and $[F]$), and disturbance attenuation in a large regions of the RIE parameter space. The output of the wafer etch subsystem (WES) namely, *etch depth*, is assumed to be monitored by one or more reflectometers, or can be estimated by Kalman filters to provide set points for the plasma real-time control module. The use of plasma real-time robust controller has the following advantages:

(i) dynamic plasma disturbances rejection,

(ii) maintenance of plasma variables at desired values, and

(iii) diagnoses capabilities in the plasma subsystem.

As for as the disadvantages of using only the plasma real-time controller, without using any information on the output of the wafer etch subsystem, one can enumerate the following issues:

(i) plasma sensor drift may cause the etch process to drift

(ii) the etch disturbances are not controlled, and

(iii) the cost of real-time sensors.

The integrated approach combines the benefit of better disturbance rejection, more relevant control variables, and diagnosis on two different time scales.

7.4 Hierarchical Real-Time Control Strategies

In the previous section, we saw that if the operating conditions are changed, that could also change the best control strategy among the A_1, A_2, A_3, A_4, and A_5 configurations.

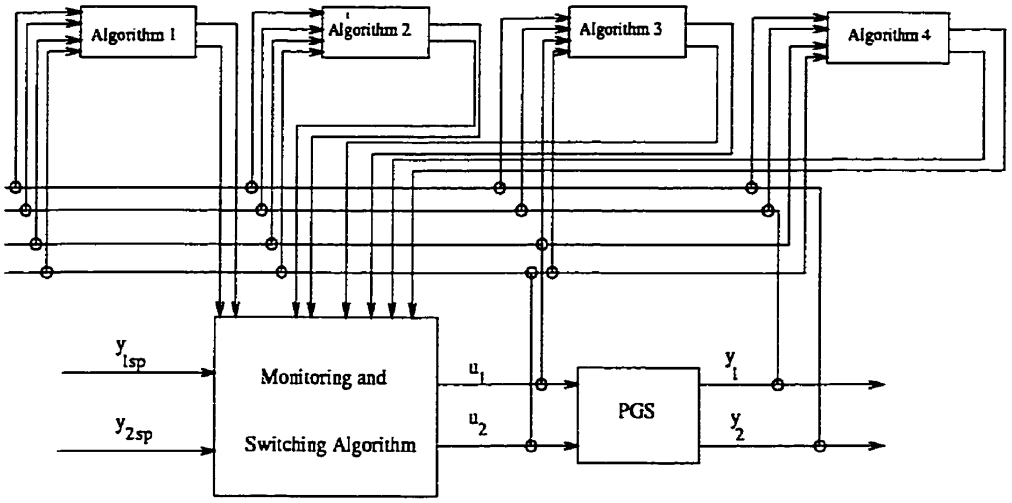


Figure 7.7: Hierarchical control strategy structure.

To select the best control strategy for each sample interval we have conceived a hierarchical control structure depicted in Figure 7.7. The main idea behind this control strategy consists of selecting for each sampling interval the best algorithm that is capable of yielding the closest value of the fluorine concentration with respect to the desired value (set point) or equivalently the best tracking error with respect to the fluorine concentration. The hierarchical control strategy compares for each sampling interval the tracking error resulting from each algorithm and then selects the best algorithm that results in the smallest tracking error.

The results of the selection are depicted in Figures 7.8-7.9 for four assumed sectors of the system trajectory for the sake of clarity. From these figures, we can observe that in the first sector, the best control strategies selected are A_3 and A_5, for the second sector we obtained A_5 and A_1, with high frequency A_1 due its the best tracking error achieved in the steady-state, for third sector we obtained A_1, and for the last sector of the trajectory, we obtained A_1, A_2 with high frequency A_2 due its the best tracking steady-state error compared with other strategies developed in this thesis. The above results demonstrate the fact that each control strategy yields

a different performance at various points of the system trajectory. The hierarchical control strategy performs the best selection of the control strategy taking into account the tracking error for each configuration with minimum control effort so to achieve continuously the best overall control performance. The overall performance obtained by this control strategy is represented in Figures 7.10-7.11. These figures reveal some peaks along the trajectory due to the switching from one algorithm to another, but the general trend is to reach the steady-state after a very short transient interval. The peaks in the *throttle position* and in the *RF power* depicted in Figures 7.12-7.13 appear due to the brief time switch and fast return of the *robust control strategy* [A_5] and *PI self-tuning control* strategy (A_1). Almost for the first three sectors of the trajectory, the control effort is piecewise constant and has a smooth evolution in the last sector due to the choice of configurations A_5, A_3, A_1 and A_2 which yield the best tracking performance. We can conceive other control structures by selecting different models instead of selecting a control algorithm, or alternatively using mixed structures that combine a control algorithm and the model simultaneously.

7.5 Summary

In this chapter, we attempted to select the best real-time control strategy by performance comparisons of the real-time control strategies developed in Chapters 4-6, namely *PI self-tuning* adaptive controller, *LQG* controller, stochastic minimum variance adaptive controller, intelligent controller, and robust controller. To realize this, we use the Mogab-Flamm model of the wafer [21] which assumes that the etch rate is directly proportional to the fluorine concentration, $[F]$.

In this chapter we have also developed a hierarchical control strategy to select the best real-time feedback control strategy for each sampling time interval. The hi-

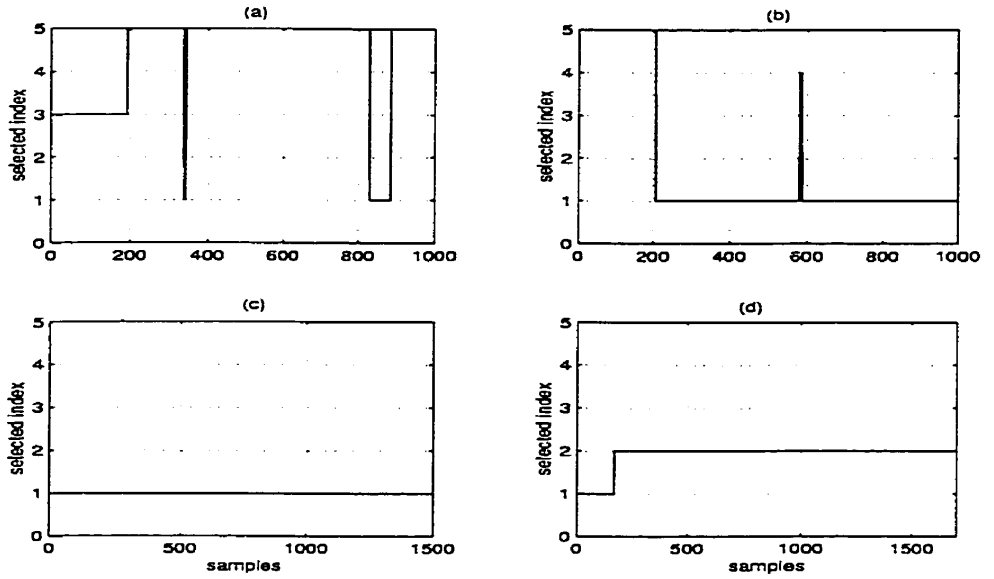


Figure 7.8: The selection index of the best per-interval control strategy divided into four sectors of the trajectory.
Legend: (a) The first sector of 1000 samples designates the interval [1, 1000];
(b) The second sector of 1000 samples designates the interval [1001, 2000];
(c) The third sector of 1500 samples designates the interval [2001, 3500];
(d) The fourth sector of 1703 samples designates the interval [3501, 5203].

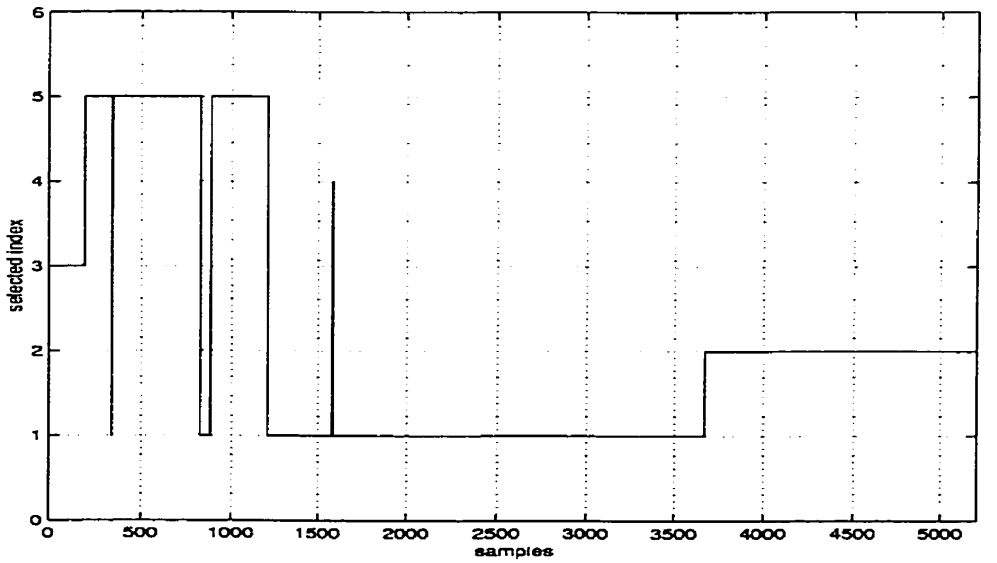


Figure 7.9: The overall selection index of the best per-interval control strategy.

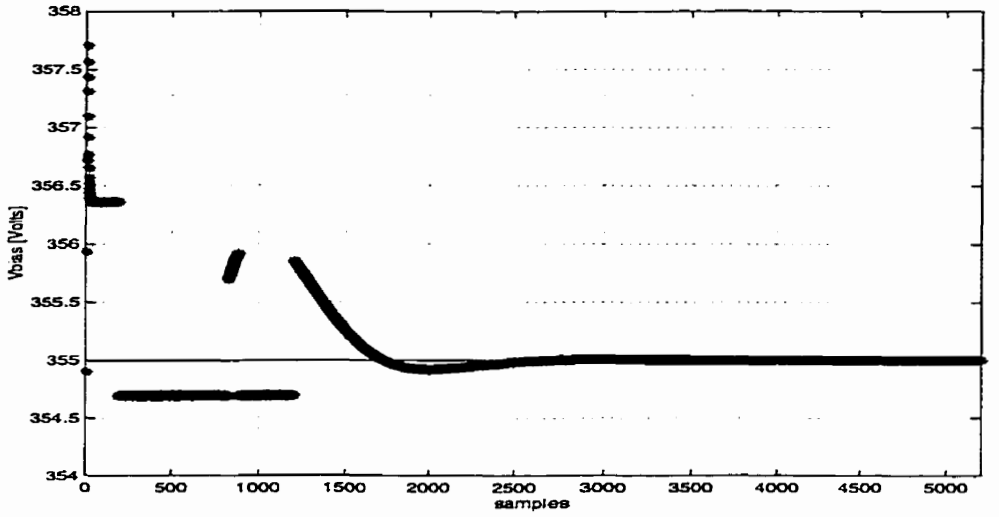


Figure 7.10: The evolution of the V_{bias} in closed-loop selected by the hierarchical control strategy without input constraints.
Legend: The star designates V_{bias} calculated by the algorithm, and the solid line designates the V_{bias} set point.

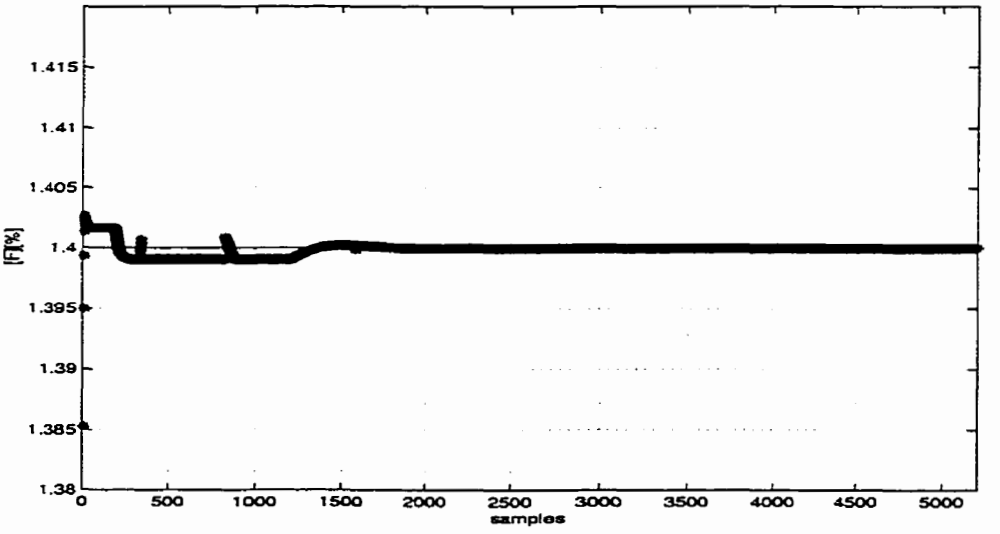


Figure 7.11: The evolution of the *fluorine concentration* $[F]$ in closed-loop selected by the hierarchical control strategy without input constraints.
Legend: The star designates $[F]$ calculated by the algorithm, and the solid line designates the $[F]$ set point.

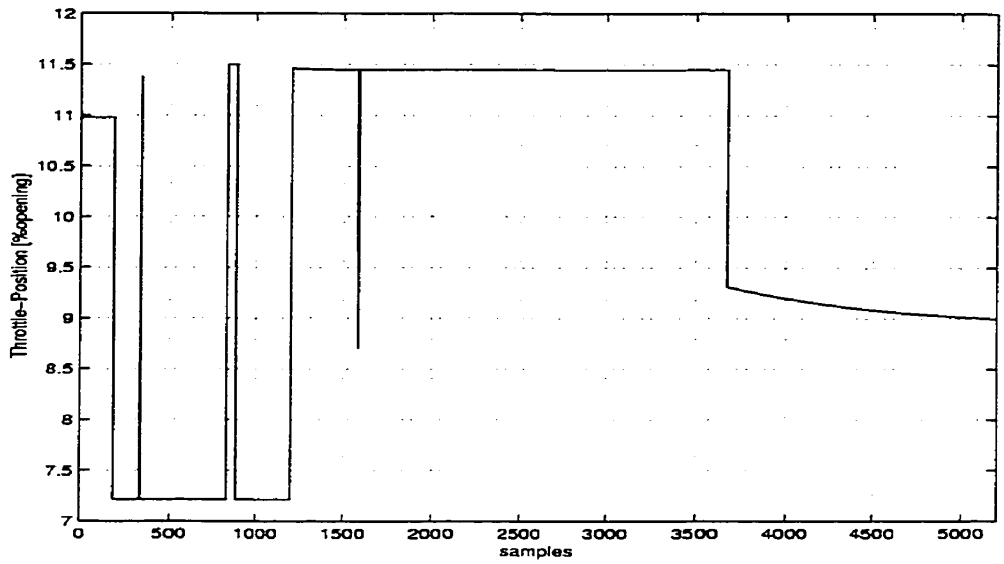


Figure 7.12: The evolution of the *throttle position* in closed-loop selected by the hierarchical control strategy without input constraints.

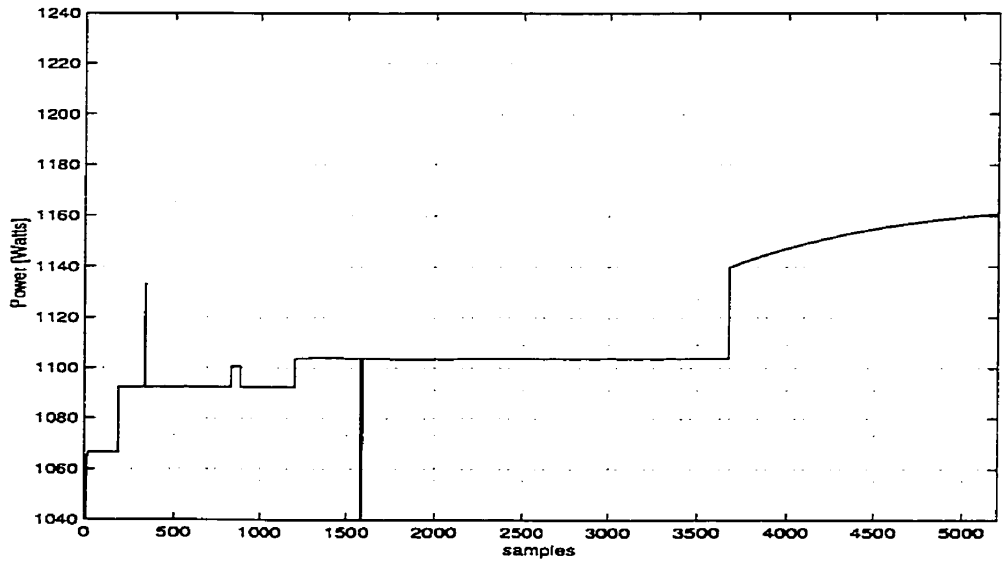


Figure 7.13: The evolution of the *power* in closed-loop selected by the hierarchical control strategy without input constraints.

erarchical control strategy performs the best selection of the control strategy taking into account the tracking error of each configuration with minimum control effort to achieve continuously the best overall control performance. This control strategy confirms that the choice of the best per-interval control action for the PGS system would achieve good performance for the etching process. We can justify this conclusion by considering the results depicted in Figures 7.10, 7.11 for the V_{bias}, and fluorine concentration $[F]$ in closed-loop PGS system. From these simulation results, we can observe that after a few samples (fast transient) both process variables reach the set points with good tracking performance.

Chapter 8

Conclusions and Future Work

8.1 Conclusions

To place etching process design and control on a more rational basis, we have developed in this research both linear and nonlinear models of the PGS system as well as five MIMO real-time control strategies yielding superior performance in a stochastic environment. From both theoretical and practical perspectives the following observations can be made:

- The algorithms use a second-order *ARIMA* model of the etching process based only on the experimental input-output data, and a linear model in state-space representation. We have also explored the possibility of using neural networks for modelling and control of the RIE system in real-time since precise modelling of the RIE is difficult due to the extremely complex nature of the particle dynamics within plasma. The development of the neuromodels requires considerable training and testing. For the training set, the first half of the experimental input-output data is selected to adjust the weight matrices, and the other half is used as the testing set.

- The above models represented our basic support for developing multivariable and intelligent control strategies, namely *PI self-tuning* controller, *LQG* controller, stochastic adaptive control using minimum variance identification, robust control, and hierarchical control strategies.

- All the controllers are developed to directly affect the transient behaviour of the PGS system in closed-loop and all the real-time control strategies are intended for on-line use.

- Past design experience and a *priori* information on system dynamics has been incorporated into the adaptive system design.

- Addition of sensors to the adaptive system such as plasma spectrometers and reflectometers, could improve the state estimation and parameter identification algorithms substantially.

- The algorithms developed could avoid excessive changes in the control input by taking into account the physical limitations of the *throttle valve* and *RF power* actuators.

- A self-tuning regulator was designed by assuming that the system to be controlled is not open-loop minimum-phase and each loop has the same time delay. The algorithm is relatively simple, reliable and robust.

- As the level of disturbances increase, the convergence rate of the performance index $J(k)$ for the *PI self-tuning* control, *LQG* control and stochastic adaptive control algorithms will decrease. The tracking performance may degrade as the noise levels increase.

- Perfect tracking can be obtained for the deterministic (i.e. $w(k)$=0 and $v(k)$=0) case.

- The neuromodels were able to capture the nonlinearities in the *throttle valve actuator* and to ensure an accurate representation of the dynamics of the RIE process.

- The development of an optimal neuromodel was complicated by the fact that back-propagation networks contain several adjustable parameters i.e., number of hidden layers and hidden neurons, learning rate, momentum, error goal, etc., for which the optimal values are initially unknown.

- The convergence of the neuromodels also depends on the experimental input-output data set used to train the neural networks.

- Even if the prescribed error goal is reached during the training phase by the neural networks this fact does not necessarily always guarantee good performance. This should be verified in the testing phase (generalization) where it is possible to discover that the neural networks were not capable of learning the experimental input-output data sets.

- The neurocontrollers eliminate the need for an experienced engineer to tune the parameters and can be more easily applied to reactive ion etching process. Furthermore, the real-time neurocontrol strategies for the reactive ion etching systems can support the efforts of process and equipment engineers by reducing the number of off-line experiments that must be performed, and by eliminating the need for building extra equipment prototypes.

- Proper choice of neurocontroller parameters (weights) is critical to the performance of RIE systems.

- The real-time neurocontrol strategy is very robust to a wide range of input command signals, as well as varying initial conditions and different noise levels.

- Robust control design ensures that closed-loop stability and performance specifications are insensitive to unknown components of the plasma dynamics [36]. The proposed real-time robust control strategy for RIE system is capable of producing the required transient and steady-state performances by suitable choice of feedback loop, compensator, and controller parameters.

- In this thesis, a robust control algorithm that ensures stability of the RIE system in closed-loop has been developed with reduced sensitivity to all the changes that can occur in the parameters of the etching process. Towards this end we have used generalized H^2 and H^∞strategies as described in [36].

- The performance of the robust control strategy depends also on the selection of the weighting functions $W_1(s)$, $W_2(s)$, and $W_3(s)$.

- In a performance comparison of the proposed control strategies, it was observed that neurocontrol, robust control, LQG, and stochastic minimum variance adaptive control yield the best performance.

- The performance of these control strategies differ with respect to the range of operating conditions. To choose the best strategy when the operating conditions change, we have proposed a hierarchical control strategy to generate the best per-interval control action for the PGS system.

- The correlation between the etch rate and fluorine concentration is the main reason for relating the wafer performance to the PGS performance. This has motivated us to only focus on developing control strategies to ensure tight control of the PGS outputs, i.e., V_{bias} and fluorine concentration, $[F]$.

8.2 Contributions

Among the main contributions in this thesis, we list the following:

- Development of linear models of the PGS, namely second-order coupled and decoupled MIMO models *with pure delay* and *without delay*, validated and simulated in Figures 3.2-3.21. These models were developed corresponding to the operating point provided by the experimental input-output data. The most accurate of these models seems to be the coupled MIMO PGS model *with pure delay*. This was used to build the stochastic adaptive control, LQG and robust control strategies.

- Development of four nonlinear neuromodels for the PGS system and controllers, and four real-time neurocontrol strategies with the simulated results shown in Figures 5.13-5.16, 3.22-3.29, and Figures 5.1-5.8 respectively. The most accurate neuromodel which fits the experimental input-output data seems to be the second-order nonlinear PGS neuromodel. This neuromodel is able to capture the nonlinearities of the *throttle valve actuator* and to operate in a wide range of operating conditions. The best suitable neurocontroller structure which more accurately learns the inverse dynamics of the plant uses the same neuromodel structure i.e., second-order nonlinear PGS inverse neuromodel. The structure of these neuromodels appears for the first time in the plasma literature.

- Development of a novel real-time robust control strategy structure which responds more accurately to the variations in plasma parameters, disturbances, and changes in the set points.

- Development of a novel hierarchical real-time control strategy capable of selecting the best per-interval control using the architecture depicted in Figure

7.7. This architecture is proposed for the first time in the plasma literature to control the PGS system in closed-loop.

- Development of practical aspects of real-time feedback control strategies to address a real application problem, namely design of real-time MIMO control strategies for the highly nonlinear RIE process to improve the manufacturing characteristics of the plasma.

8.3 Future Work

The problem of proof of stability for the neurocontrol strategies would be a challenging effort which is essential for dynamic systems analysis, synthesis, and control. This is one aspect of the future work to be carried out. Possible approaches would be to use the performance index $J(k)$ as a Lyapunov function, and show that it satisfies the Lyapunov stability criteria, or associating the control algorithm with a set of nonlinear time-varying deterministic differential equations [32], and then studying the stability of these equations that are related to the stability of the closed-loop adaptive system.

For MIMO coupled systems, a suboptimal alternative which is inherently less sensitive to time delay variations and non-minimum phase behaviour is the multivariable pole assignment method. Using the design methodology presented in [22], it will be interesting to extend our control design to an on-line stochastic regulator, namely a self-tuning algorithm based on the minimum variance principle.

A self-organizing fuzzy logic controller using auto regressive moving average model will be another interesting approach to consider in this research. This algorithm could be a design methodology for on-line self-organizing fuzzy logic controllers without using any plant model [38]. The control algorithm obtains the control rules for a system about which little knowledge is available. Compared with conventional fuzzy

logic control when knowledge about the system for developing control rules has to be supplied by an expert, the proposed fuzzy logic controller needs no expert in developing control rules. The proposed self-organizing fuzzy controller will be a rule-based type of controller which learns on line how to control the system, and should be possible to use for the reactive ion etching process. The algorithm will combine system identification and control knowledge that are obtained through learning and experience.

Future research could also be directed towards stochastic dual adaptive control and stochastic adaptive neurocontrol strategies. In this context, it will be useful to extend the stochastic adaptive control results discussed in Chapter 4 for developing a neural self-tuning adaptive control algorithm for nonlinear systems. Here the implicit identification may be performed by a neural network based on minimum variance, Newton and gradient optimizations with the advantage of increasing the computational capability due to the parallel processing structure [32]. The neural identification schemes and the control law should be robust and computationally efficient for real-time adaptive control design for the PGS.

Another future direction could be a robust neurocontrol design. In this approach the neurocontroller structure will be determined so as to be able to subsume the mapping classes necessary to satisfy the given robustness requirements. The neurocontroller will be trained so that these requirements are actually satisfied. The means to enable the training algorithm to find the solution satisfying these requirements is through the choice of the appropriate cost function. The most important part of the cost function that incorporates robustness requirements is the set of training examples. These two aspects, neurocontroller structure and training examples will form the basis for developing a theoretical foundation for implementing a robust neurocontroller for the PGS system, based on the methodology outlined in [39].

The similarities between the reactive ion etching process and the other chemical processes, such as chemical vapour deposition (CVD), anisotropic wet etching silicon ($TMAH$), plasma enhanced chemical vapour deposition ($PECVD$), low plasma chemical vapour deposition ($LPCVD$), etc., may give us the opportunity to further improve the theoretical approach (computational efficiency, convergence, stability, and robustness), and to extend the application field of the algorithms developed in this thesis to these other processes.

Finally, it is envisaged that the semiconductor manufacturing processes will still remain an open field and provide a great opportunity for much further applications of the systems and control techniques.

Bibliography

[1] M. Elta, H. Etemad, J. S. Freudenberg, M. D. Giles, J. W. Grizzle, P. T. Kabamba, P. P. Khargonekar, S. Lafortune, S. M. Meerkov, J. R. Moyne, B. A. Rashap, D. Teneketzis, F. L. Terry, *"Applications of control to Semiconductor Manufacturing: Reactive Ion Etching"*, Proceeding of the American Control Conference, San Francisco, California, June 1993, pp. 2990-2995.

[2] B. W. Kim, G. S. May, *"Reactive Ion Etch Modeling Using Neural Networks and Simulated Annealing"*, IEEE Transactions on Components, Packaging, and Manufacturing Technology-Part C, vol. 19, no. 1, January 1996, pp. 3-8.

[3] S. W. Butler, K. J. Mc Laughlin, T. F. Edgar, I. Trachtenberg, *"Development of Techniques for Real-Time Monitoring and Contrrol in Plasma Etching - Multivariable Control System Analysis of Manipulated, Measured, and Performance Variables"*, Journal of Electrochemical Society, vol. 138, no. 9, September 1991, pp. 2727-2735.

[4] B. A. Rashap, M. E. Elta, H. Etemad, J. P. Fournier , J. S. Freudenberg, M. D. Giles, J. W. Grizzle, P. T. Kabamba, P. P. Khargonekar, S. Lafortune, J. R. Moyne, D. Teneketzis, F. L. Terry, *"Control of Semiconductor Manufacturing Equipment : Real-Time Feedback Control of Reactive Ion Etcher"*, IEEE Transactions on Semiconductor Manufacturing, vol. 8, no. 3, August 1995, pp. 286-297.

[5] E. S. Hamby, A. T. Demos, P. T. Kabamba, P. P. Khargonekar, *"A Control Oriented Modeling Methodology for Plasma Enhanced Chemical Vapour Deposition Processes"*, Proceedings of the American Control Conference, Seattle, June 1995, pp. 220-224.

[6] T. L. Vincent, P. P. Khargonekar, F. L. Terry, *" End Point and Etch Rate Control using Dual-Wavelength Laser Reflectometry with a Nonlinear Estimator"*, Journal of Electrochemical Society, vol. 144, no. 7, July 1997, pp. 2467-2472.

[7] T. L. Vincent, P. P. Khargonekar, F. L. Terry, *"An Extended Kalman Filter Based Method of Processing Reflectometry Data for Fast In-Situ Etch Rate Measurements"*, IEEE Transactions on Semiconductor Manufacturing, vol. 10, no. 1, February 1997, pp. 42-51.

[8] T. L. Vincent, P. P. Khargonekar, B. A. Rashap, F. Terry, M. Elta, *"Nonlinear System Identification and Control of a Reactive Ion Etcher"*, Proceedings of the American Control Conference, Baltimore, Maryland, June 1994, pp. 902-906.

[9] S. H. Smith, D. S. Boning, *"A Self - Tuning EWMA Controller using Artificial Neural Network Function Approximation Techniques"*, IEEE Transactions on Components, Packaging, and Manufacturing Technology-Part C, vol. 20, no. 2, April 1997, pp. 121-131.

[10] K. D. Allen, H. H. Sawin, *"The Plasma Etching of Polysilicon with CF3Cl / Argon Discharges-Modeling of Ion Bombardement Energy Distributions"*, Journal of the Electrochemical Society - Solid State Science and Technology, vol. 133, no. 11, pp. 2326-2330.

[11] K. D. Allen, H. H. Sawin, M. T. Mocella, M. W. Jerkins, *"The Plasma Etching of Polysilicon with CF3Cl / Argon Discharges-Parameter Modeling and Impedance Analysis"*, Journal of the Electrochemical Society, Solid-State Science and Technology, vol. 133, no. 11, November, 1986, pp. 2315-2325.

[12] K. D. Allen, H. H. Sawin, A. Yokozeki, *"The Plasma Etching of Polysilicon with CF3Cl / Argon Discharges, Modeling of Etching Rate and Directionality"*, Journal of Electrochemical Society, Solid-State Science and Technology, vol. 133, no. 11, November 1986, pp. 2331-2338.

[13] B. Kim, G. S. May, *"Reactive Ion Etch Modeling Using Neural Networks and Simulated Annealing"*, IEEE Transactions on Components, Packaging, and Manufacturing Technology-Part C, vol. 19, no. 1, January 1996, pp. 3-8.

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

[15] Y. Huang, T. Edgar, D. Himmelblau, I. Trachtenberg, *"Constructing a reliable neural network model for a plasma etching process using limited experimental data"*, IEEE Trans. Semiconduct. Manufact. Vol. 7, no. 3, pp. 333-344, Aug. 1994.

[16] S. S. Han, L. Cai, G. S. May, A. Rohatgi, *"Modeling the Growth of PECVD Silicon Nitride Films for Solar Cell Applications using Neural Networks"*, IEEE Transactions on Semiconductor Manufacturing, vol. 9, no. 3, August 1996, pp. 303-310.

[17] E. Reitman, S. Patel, *"A production demonstration of wafer-to-wafer plasma gate etch control by adaptive real-time computation over-etch time from in situ process signals"*, IEEE Trans. Semiconduct. Manufact., vol. 8, pp. 304-308, Aug. 1995.

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

[19] T. H. Smith, D. S. Boning, *"A Self-Tuning EWMA Controller Utilizing Artificial Neural Network Function Approximation Techniques"*, IEEE Transactions on Components, Packaging, and Manufacturing Technology - Part C, vol. 20, no. 2, April, 1997, pp. 121-132.

[20] K. J. McLaughlin, S. W. Butler, T. F. Edgar, S. Trachtenberg, *"Development of Techniques for Real-Time Monitoring and Control in Plasma Etching"*, Journal of the Electrochemical Society, vol. 13, no. 3, March 1991, pp. 789-798.

[21] O. D. Patterson, P. P. Khargonekar, *"Reduction of Loading Effect in Reactive Ion Etching using Real-Time Closed-Loop Control"*, Journal of Electrochemical Society, vol. 144, no. 8, August 1997, pp. 2865-2871.

[22] E. H. Bristol, P. J. Vermeer, P. A. Chin, *" Simulation and Implementation of Self-Tuning Controllers"*, Prentice Hall, 1989.

[23] E. H. Bristol, *"On a New Measure of Interaction for Multivariable Process Control"*, IEEE Transactions On Automatic Control, pp. 133-134, January 1966.

[24] H. Demuth, M. Beale, *"Neural Network Toolbox For Use with Matlab-User's Guide"*, The Math Works Inc., June 1992.

[25] C. J. Harris, S. A. Bilings, *"Self-Tuning and Adaptive Control-Theory and Applications"*, The Institute of Electrical Engineering, London, 1981.

[26] K. S. Narendra, A. M. Annaswamy, *"Stable Adaptive Systems "*, Prentice Hall, Englewood, Cliffs, NJ., 1989.

[27] K. Astrom, B. Wittenmark, *"Adaptive Control"*, Addison-Wesley, 1990.

[28] F. Franklin, J. D. Powell, J. W. L. Michael, *"Digital Control of Dynamic Systems"*, Addison-Wesley, 1990.

[29] T. J. Knight, D. W. Greve, X. Cheng, B. H. Krogh, *"Real-Time Multivariable Control of PECVD Silicon Nitride Film Properties"*, IEEE Transactions on Semiconductor Manufacturing , vol. 10, no. 1, February 1997, pp. 137-146.

[30] J. M. Maciejowski, *"Multivariable feedback design"*, Addison-Wesley, 1989.

[31] L. Ljung, *"System Identification Toolbox-User's Guide"*, July 1991, Mathworks Inc.

[32] T. T. Ho, *"A generalized stochastic adaptive control algorithm"*, Ph.D Thesis, University of Colorado, 1990.

[33] M. Khalid, S. Omatu, *"A neural controller for a temperature control system"*, IEEE Contr. System. Mag., vol. 12, no. 3, pp. 58-64, 1990.

[34] M. Khalid, S. Omatu, R. Yusof, *"MIMO Furnace Control with Neural Networks"*, IEEE Transactions On Control Systems Technology, vol. 1, no. 4, pp. 238-245, 1993.

[35] K. Khorasani, *"Adaptive Control Systems - Notes"*, Concordia University, 1997.

[36] R. Y. Chiang, M. G. Safonov, *"Robust Control Toolbox For Use with Matlab-User's Guide"*, The MathWorks, Inc. , 1992.

[37] M. Hankinson, T. Vincent, K. B. Irani, P. P. Khargonekar, *"Integrated Real-Time and Run-to-Run Control of Etch Depth in Reactive Ion Etching"*, IEEE Transactions on Semiconductors Manufacturing, vol. 10 , no. 1, pp. 121-130, 1997.

[38] H. Ying, W. Siler, J. J. Buckley, *"Fuzzy control theory : A nonlinear case"*, Automatica, vol. 26, no. 3, pp. 513-520, 1990.

[39] T. Hrycej, *"Neurocontrol Towards an industrial Control Methodology"*, John Wiley & Sons, Inc., New-York, 1997.

[40] R. V. Patel, *"Multivariable System Theory and Design"*, Oxford, England, Pergamon Press, 1982.

[41] N. Tudoroiu, K. Khorasani, R. V. Patel, *"MIMO RIE Adaptive Control Strategies-Self Tuning PI Decoupled Controllers"*, Acta Universitatis Cibiniensis, "BEYOND 2000" International Conference, vol. XXXVIII, Technical Series, Computer Science and Automatic Control, Sibiu, Romania, 1999, pp. 115-120.

[42] N. Tudoroiu, K. Khorasani, R. V. Patel, *"MIMO RIE Control Strategies Using Linear Quadratic Gaussian Scheme"*, Acta Universitatis Cibiniensis, "BEYOND 2000" International Conference, vol. XXXVIII, Technical Series, Computer Science and Automatic Control, Sibiu, Romania, 1999, pp. 107-114.

[43] N. Tudoroiu, K. Khorasani, R. V. Patel, *"MIMO RIE Neurocontrol Strategies"*, IEE Proceedings - Computerized Machines and Systems, vol. 147, Issue 2, March 2000, pp. 142-146.

[44] N. Tudoroiu, K. Khorasani, R. V. Patel, *"Linear and Nonlinear MIMO Models for Plasma Reactive Ion Etching Process"*, International Scientific Symposium "Universitaria ROPET 2000", Petrosani, Romania, 19-20 October 2000, pp. 229-234.

[45] N. Tudoroiu, K. Khorasani, R. V. Patel, *"A Real-Time MIMO stochastic Adaptive Control Strategy for Plasma Reactive Ion Etching System"*, International Scientific Symposium "Universitaria ROPET 2000", Petrosani, Romania, 19-20 October 2000, pp. 235-240.

[46] N. Tudoroiu, O. Prostean, D. Curiac, *"Complex Automations"*, Timisoara, Romania, Mirton Publishing Company, 1993.

[47] N. Tudoroiu, O. Prostean, D. Curiac, I. Filip, *"Nonlinear, Digital and Optimal Control Systems Theory"*,Timisoara, Romania, Mirton Publishing Company, 1993.

[48] N. Tudoroiu, D. Curiac, *"Civil Plants Automation-Course for Students"*, Technical University Timisoara, Romania, 1994.

[49] N. Tudoroiu, D. Curiac, *"Continuous Control Systems Theory"*,Timisoara, Romania, Mirton Publishing Company, 1993.

[50] M. J. Grimble, *"Robust Industrial Control Optimal Design Approach for Polynomial Systems"*, Prentice Hall, Englewood, Cliffs, NJ., 1994.

[51] K. R. Williams, R. S. Muller , *"Etch Rates for Micromachining Processing"*, Journal of Microelectromechanical Systems, vol. 5, no. 4, December 1994, pp. 256-269.

[52] X. A. Wang and R. L. Mahajan, *"Artificial Neural Network Models Based Run-to-Run Process Controller"*, IEEE Transaction on Component, Packaging, and Manufacturing Technology - Part C, vol. 19, no. 1, January 1996, pp. 19-25.

[53] B. W. Kim, G. S. May, *"Reactive Ion Etch Modeling Using Neural Networks and Simulated Annealing"*, IEEE Transactions on Components, Packaging, and Manufacturing Technology -Part C, vol. 19, no. 1, January 1996, pp. 3-8.

[54] S. W. Butler, J. Stefani, *"Application of Predictor Corrector Control to Polysilicon Gate Etching"*, Proceedings of the American Control Conference, San Francisco, June 1993, pp. 3003-3007.

[55] Y. M. Park , U. C. Moon, K. Y.Lee," *A Self-Organizing Fuzzy Logic Controller for Dynamic Systems using a Fuzzy Auto-Regressive Moving Average (FARMA) Model"*, IEEE Transactions on Fuzzy Systems, no. 1, February, 1995.

[56] A. Wolovich, "Linear Multivariable Systems", Springer: New-York, 1974.

[57] H. Kwakernaak, R. Sivan, *"Linear optimal control systems"*, Wiley:New-York,1972.

[58] H. Kwakernaak, *"Minimax frequency domain performance and robustness optimization of linear feedback systems"*, IEEE Transactions on Automatic Control, vol. AC-30, no. 10, pp. 994-1004, 1985.

[59] H. Kwakernaak, *"A polynomial approach to minimax frequency domain optimization of multivariable feedback systems"*, Int. J. Control, vol. 44, no. 1, pp. 117-156, 1986.

[60] P. Dorato, *"Robust Control"*, IEEE Press, New-York,1 987.

[61] E. Mosca, L. Giarre, *"A polynomial approach to the MIMO, LQ servo and disturbance rejection problems"*, Automatica, vol. 28, no. 1, pp. 209-213, 1990.

[62] E. Mosca, A. Casavola, L. Giarre, *"Minimax LQG stochastic tracking and servo Problems"*, IEEE Transactions on Automatic Control, vol. AC-35, no. 1, pp. 95-97, 1990.

[63] E. Mosca, L. Giarre, A. Casavola, *"On the polynomial equations for the MIMO LQ stochastic regulator"*, IEEE Trans. Autom. Control, vol. 35, no. 3, pp. 320-322, 1990.

[64] M. J. Grimble,, M. A. Johnson, *"Optimal control and stochastic estimation: Theory and Applications"*, vols. I and II, Wiley, 1988.

[65] M. J. Grimble, *"Dual criterion stochastic optimal control problem for robustness improvement "*, IEEE Trans. Autom. Control, vol AC-31, no. 2, pp. 181-185, 1986.

[66] M. J. Grimble, *"Controllers for LQG self-tuning application with colored measurement noise and dynamic costing"*, IEE Proc., vol 133, Pt.D, no. 1, pp. 19-29, 1986.

[67] M. J. Grimble, *"Design of optimal machine-control systems"*, IEE Proc., vol. 124, no. 9, pp. 821-827, 1977.

[68] M. J. Grimble, *"Youla parametrized two and a half degrees of freedom LQG controller and robustness improvement cost weighting"*, IEE Proc., vol. 139, Pt.D, no. 2, pp. 147-160, 1992.

[69] M. J. Grimble, *"Multivariable controllers for LQG self-tuning applications with colored measurement noise and dynamic cost weighting"*, Int. J. Syst., Sci., vol. 17, no. 4, pp. 543-547.

[70] M. J. Grimble, *"H^∞ Robust controller for self-tuning applications"*, Int. J. Control, Part I, vol. 46, no. 4, pp. 1429-1444; Part II, vol. 46, no. 5, pp. 1819-1884, 1987.

[71] V. Kucera, *"Stochastic multivariable control-A polynomial equation approach"*, IEEE Transactions on Automatic Control, vol. AC-25, no. 5, pp. 913-919, 1980.

[72] J. M. Mendel, *"White noise estimators for seismic data processing in oil exploration"*, IEEE. Trans. Autom. Control, vol. AC-22, no. 5, pp. 694-707, 1977.

[73] T. J. Moir, M. J. Grimble, *"Optimal self-tuning filtering, prediction and smoothing for Discrete multivariable processes"*, IEEE Trans. Autom. Control, vol. AC-29, no. 2, pp. 128-137, 1984.

[74] K. J. Hunt, M. Sebek, M. J. Grimble, *"Optimal multivariable LQG control using a single diophantine equation"*, Int. J. Control, vol. 46, no. 4, pp. 1445-1453, 1987.

[75] K. Glover, D. McFarlane, *"Robust stabilization of normalized coprime factor plant descriptions with H^∞ bounded uncertainty"*, IEEE Trans. Autom. Control, AC-34, no. 8, pp. 821-830, 1989.

[76] D. C. Youla, H. A. Jabr, J. J. Bongiorno, *"Modern Wiener-Hopf design of optimal controllers Part II : The multivariable case"*, IEEE Trans. Autom. Control, vol. AC-21, no. 3, pp. 319-338, 1976.

[77] N. A. Fairbairn, M. J. Grimble, *"H^∞ Robust controller for self-tuning applications, Part III: Self-tuning controller implementation"*, Int. J. Control, vol. 5, no. 1, pp. 15-36, 1990.

[78] J. C. Doyle, K. Glover, P. P. Khargonekar, B. A. Francis, *"State-space solutions to standard H^2 and H^∞ control problems"*, IEEE Trans. Autom. Control, vol. AC-34, no. 8, pp. 831-846, 1989.

[79] J. C. Doyle, G. Stien, *"Multivariable feedback design: Concepts for a classical / modern synthesis"*, IEEE Trans. Autom. Control , vol. 26, pp. 4-16, 1981.

[80] D. P. Bertsekas, *"Dynamic Programming and Optimal Control"*, vol. I., Belmont, 1995.

[81] J. Ljung , *"System Identification: Theory for the user"*, Prentice Hall: Englewood Cliffs, NJ., 1987.

[82] T. E. Benson, L. I. Kamlet, P. Klimecky, F. L. Terry, *"In-situ spectroscopic reflectometry for polycrystalline silicon thin film etch rate determination during Reactive Ion Etching"*, Electronic Materials Conf., Charlottesville, VA, pp. 21-23, June 1995.

[83] Z. H. Zhou, S. Compton, S. Yang, R. Reif, *"In-situ semiconductor materials characterization by emission Fourier transform infrared spectroscopy "*, IEEE Trans. Semiconduct. Manufact., vol. 7, pp. 87-91, 1994.

[84] M. Wilcoxson, V. Manousiouthakis, *"Continuum fluid models for plasma etching Reactor control"*, Proc. Americain Control Conference, San Francisco, California, pp. 3013-3017, FP9, June 1993.

[85] M. Meyyappan, J. P. Kreskovski, *"Glow descharge simulation through solutions to the moments of the Boltzmann transport equation"*, J. Appl. Phys., vol. 68, no. 4, pp. 1506- 1512, 1990.

[86] A. R. Soper, A. D. Mellichamp , E. D. Seborg, *"An adaptive nonlinear control strategy for Photolithography "*, Proc. American Control Conference, San Francisco, California, pp. 2998- 3002, FP9, June 1993.

[87] T. F. Edgar, D. M. Himmelblau, *"Optimization of chemical processes"*, McGraw-Hill, New-York, 1988.

[88] M. A. Henson, D. E. Seborg, *"An internal model control strategy for nonlinear systems"*, AIChE J., vol 37, no.7, pp.1065, 1991.

[89] A. D. White, D. Boning, S. W. Butler, G. Barna, *"Spatial characterization of wafer state using principal component analysis of optical emission spectra in plasma etch"*, IEEE Trans. Semiconduct. Manufact., vol. 10, no. 1, pp. 52-60, February, 1997.

[90] S. Butler, K. McLaughlin, T. Edgar, I. Trachtenberg, *"Development of techniques for real-time monitoring and control in plasma etching"*, J. Electrochemistry Society, vol. 138, no. 9, Sept. 1991.

[91] S. Lee, C. Spanos, *"Prediction of wafer state after plasma processing using real-time tool data"*, IEEE Trans. Semiconduct. Manufact., vol. 8, pp. 252-261, Aug. 1995.

[92] D. Shadmehr, P. Chou, G. Oehrlein, R. Jaffe, *"Principal component analysis of optical emission spectroscopy and mass spectrometry: Application to Reactive Ion Etch process parameter estimation using neural networks"*, J. Electrochemistry Soc., vol. 139, no. 3, March, 1992.

[93] D. White, M. Jordan, *"Optimal control: A foundation for intelligent control", Handbook of Intelligent control : Neural Fuzzy and adaptive approaches"*, New-York, 1992.

[94] L. X. Wang, *" Stable adaptive fuzzy control of nonlinear systems "*, IEEE Trans. Fuzzy Systems , vol. 1, no. 2, pp. 146-155, 1993.

[95] L. X. Wang, *" A supervisory controller for fuzzy control systems that guarantees stability "*, IEEE Trans. Autom. Control, vol. 39, pp. 1845-1847, Sept. 1994.

[96] E. H. Mamdani, S. Assilian, *"An experiment in linguistic synthesis with a fuzzy logic controller"*, Int. J. Man. Mach. Studies, vol.7, no. 1, pp. 1-13, 1975.

[97] E. H. Mamdani, *"Application of fuzzy logic to approximate reasoning using linguistic Synthesis"*, IEEE Trans. Computer, vol. AC-26, no. 12, pp. 1182-1191, 1977.

[98] T. J. Pocyk, E. H. Mamdani, *" A linguistic self-organizing process controller"*, Automatica, vol. 15, pp. 15-30, 1979.

[99] R. Tanscheit, E. M. Scharf, *"Experiments with the use of a rule-based self organizing controller for robotics applications"*, Fuzzy Set and System, vol. 26, pp. 195-214, 1988.

[100] S. Shao, *"Fuzzy self-organizing controller and its applications for dynamic processes"*, Fuzzy Set Syst., vol. 26 , pp. 151-164, 1988.

[101] Z. Y. Zhao, M. Tomizuka, S. Sagara, *"A fuzzy tuner for fuzzy logic controller"*, Proc. American Control Conference, pp. 2268-2272, 1992.

[102] B. Kosco, *"Neural Networks and Fuzzy systems"*, Prentice Hall, Englewood Cliffs, NJ., 1992

[103] V. C. Klema, A. J. Laub, *"The singular value decomposition: Its computation and some Applications"*, IEEE Trans. Autom. Control, vol. AC-25, no. 2, pp. 164-133, April 1980.

[104] B. S. Morgan, *"Sensitivity analysis and synthesis of multivariable systems"*, IEEE Trans. Autom. Control., vol. AC-11, no. 3, pp. 506-512, July 1966.

[105] A. T. Demos, P. T. Kabamba, P. P. Khargonekar, *"A control oriented modeling methodology for plasma enhanced chemical vapor deposition processes"*, Proc. American Control Conference, Seattle, pp. 220- 224, June 1996.

[106] M. Mocella, J. Bondur, T. Turner, *"Etch process characterization using neural network methodology : A case study"*, SPIE Proc. Module Metrology, Control and Clustering, pp. 232-242, 1991.

[107] B. Kim, G. S. May, *" An optimal neural network process model for plasma etching"*, IEEE Trans. Semiconduct. Manufact., vol. 7, no. 1, pp. 12-21, Feb. 1993