Control Relevant Model Identification with Prior Knowledge

By

Ali Esmaili ,

B.A.Sc.

A Thesis

Submitted to the School of Graduate Studies

In Partial Fulfillment of the Requirements

for the Degree

Doctor of Philosophy

McMaster University

Control Relevant Model Identification with Prior Knowledge

DOCTOR OF PHILOSOPHY (2001) McMASTER UNIVERSITY

(Chemical Engineering) Hamilton, Ontario, Canada

TITLE: Control Relevant Model Identification with Prior Knowledge

AUTHOR: Ali Esmaili, B.A.Sc. (University of Waterloo)

SUPERVISORS: Professor J.F. MacGregor and Professor P.A. Taylor

NUMBER OF PAGES: xiv, 261

Abstract

The use of prior or accumulated knowledge for the identification of multivariable models that will generally assure the stability of multivariable model based controller designs is investigated. The systems that are considered in this study are by and large linear, time invariant, ill-conditioned, and multi-input multi-output (MIMO) in nature. The effect of different types of prior information on controller stability is studied. It is shown that use of some types of prior knowledge may improve the model quality in terms of the stability of the resulting closed-loop system, while use of other types of prior knowledge may degrade the model quality. Prior knowledge that provides information about the low-gain direction of the process has the most significant effect on controller stability. Several issues associated with incorrect prior knowledge and the sensitivity of controller stability to such an error are also considered. This leads to checkable metrics that can be used by practitioners to evaluate the sensitivity of the controller to given prior knowledge before controller implementation. The issue of model maintenance (that is re-identification of existing models) that will result in improved controller stability in MIMO controllers is then addressed. Posterior knowledge about existing controller performance can be used to re-estimate models. Two novel controller designs result from this study: a multi-model style controller, and an adaptive style controller. Finally, issues regarding closed-loop identification of single-input single-output systems are considered. In particular, it is shown that the direct method of closed-loop identification results in an improved model quality compared to 2-step methods of closed-loop identification.

Acknowledgements

I wish to thank my advisors, Professor John F. MacGregor and Professor Paul A. Taylor, for their supervision, expert guidance and encouragement throughout the course of this work. I would also like to extend my thanks to Professor Bone for his useful suggestions and continuing interest.

The financial assistance provided by Natural Sciences and Engineering Research Council of Canada, and Department of Chemical Engineering are also gratefully acknowledged.

To my new and old colleagues in the Penthouse: Chunliang, François, Ivan, Jose, Jason, Jean-François, Ken, Kevin, Koji, Laura, Manish, Nicki, Pierre, San, Steve, Thanassis, and Tracy, I would like to thank you so much for being such wonderful people. Song, Angela, Jesus, and Alison deserve a special thank you for their friendship, help, encouragement and lively discussion throughout the years. To my old colleagues and friends Hinh and Dalmar, I appreciate your help and friendship in the last decade. To Ruijie, who I have the greatest admiration for: I will always cherish every discussion we have had.

I would like to express my love and gratitude to Tanya, you have always believed in me and stood by my side. Thank you for your love, support and patience throughout my studies, particularly in my final years.

Finally, I dedicate this thesis to my family. My Father, Mother, grand parents, uncles, aunts and cousins whose example I tried to follow. In my eyes they were the greatest example of perseverance and determination. I am grateful to my Mother, whose courage is only surpassed by her love, and whose sacrifices have made it possible for me to reach my goals. I would also like to thank my little sister for caring for me more than I ever knew.

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

This chapter discusses model identification and the role of prior knowledge in model identification. It presents some of the current practices in chemical plant model identification and issues of using prior knowledge in such model identification. It states the objective of this thesis and provides an outline of the chapters to follow. In addition, it states the basis that this thesis is based on.

1.1. Why Model Identification?

Model Identification is the process of building a mathematical representation of a physical system using experimental data. This mathematical model is then used for a variety of reasons in different scientific fields. The commonality between the different application fields is that in all cases the model is an estimate of the physical system and provides valuable information about the system. Perhaps the most intriguing aspect of model identification is the variety of its applications. Model identification has been used to model industrial processes, military aircraft, human voice, and a host of other processes in science and engineering (Ljung 1999; Juang 1994; Söderström and Stoica 1989).

In today's petrochemical plants the need for efficiency and high quality has resulted in sophisticated control, monitoring and optimization applications. At the center of these applications is a model. The capability of the estimated model to describe the dynamic and static behavior of the process accurately is the crucial limitation in these applications (Jørgensen 1988). For example in the case of model-based controllers, it has been concluded that the single most important phase in control application is the model estimation phase (Jørgensen 1988). The model identification phase is not only the most important phase but also the most time consuming phase. Andersen et al. (1991) have suggested that in practice "once an adequate model has been obtained, 80-90% of the implementation is done". This predominance can be mainly attributed to the identification experiments that require significant duration.

1.2. Thesis Objective and Outline

The goal of this thesis is to investigate the use of various types of constraints at different stages of process identification. Several aspects of constraint utilization will be considered: use of prior knowledge in model identification, sensitivity of the solution to uncertainty in that prior knowledge, and addition of acquired (posterior) knowledge. The main contribution of the thesis will be an investigation of the value of using constraints in identification of chemical processes. The research will investigate the following questions:

- What are the possible constraints available in chemical process identification?
- What types of constraints are most valuable in improving model identification?
- How to integrate these constraints into the identification procedure?

In addition, some issues regarding structural constraints in closed-loop system identification are considered (Esmaili et al. 2000).

In Chapter 2, the effect of correct prior knowledge on controller stability is studied. In this chapter a variety of different types of prior knowledge that may be present in chemical processes are introduced and their effect on controller stability analyzed. This is accomplished via a series of Monte Carlo simulations and theoretical analysis. In this chapter, it will be shown that while enforcing some constraints will improve model quality, enforcing other constraints even though they are correct, will degrade model quality.

Chapter 3 is an extension of Chapter 2, but now it is assumed that the prior knowledge may not be correct. This is perhaps a more realistic situation in practice, where prior knowledge exists with a certain level of uncertainty. This leads to a study of the sensitivity of the controller stability to error in prior knowledge. A few different metrics are purposed that may be used to evaluate the sensitivity of the controller stability to errors in prior knowledge before the control scheme is implemented.

A very particular type of posterior knowledge, namely posterior knowledge about the controller stability is used to estimate better models in Chapter 4. Different schemes are proposed in this chapter to use this posterior knowledge in changing the estimated model as the controller operates in modes. It is shown that such a posteriori knowledge is extremely valuable for improving model quality.

The effect of structural constraints on closed-loop identification is considered in Chapter 5. The asymptotic and finite data behavior of some closed-loop identification methods are investigated. Several variations on some two-step identification methods are compared with the direct identification method. Comparisons are made based on the variance of the identified process models both for asymptotic situations and for finite data sets. Process model bias resulting from improper selection of the noise and sensitivity function models is also investigated. In this context, the results support the use of direct identification methods on closed-loop data. This chapter has been presented in this thesis with minor changes compare to how it was published in literature (Esmaili et al. 2000).

1.3. Thesis Convention

The terms prior and posterior knowledge are used extensively in this thesis. In simple terms, knowledge that is independent of all particular experiences is defined as a prior knowledge, as opposed to a posterior knowledge, which is derives from experience alone. An excellent definition of prior and posterior knowledge can be found in Encyclopedia Britannica:

> "The Latin phrases a priori ("from what is before") and a posteriori ("from what is after") were used in philosophy originally to distinguish between arguments from causes and arguments from effects. The first recorded occurrence of the phrases is in the writings of the 14th-century logician Albert of Saxony. Here, an argument a priori is said to be "from causes to the effect" and an argument a posteriori to be "from effects to causes." Similar definitions were

given by many later philosophers down to and including G.W. Leibniz, and the expressions still occur sometimes with these meanings in nonphilosophical contexts. ... Although the use of a priori to distinguish knowledge such as that which we have in mathematics is comparatively recent, the interest of philosophers in that kind of knowledge is almost as old as philosophy itself."

More recently the ideas of prior and posterior knowledge have been used in the Bayesian view of probability. A typical problem of Bayesian inference occurs when one starts with only knowledge of $\mathcal{B}$, but later finds out additional information $\mathcal{C}$. Assuming that $\mathcal{C}$ is relevant information, then the problem is to find how the a priori probabilities are modified to a posteriori probabilities considering this additional information. This results in the Bayes' theorem:

$$P(\mathcal{A} \mid \mathcal{B}\,\mathcal{C}) = P(\mathcal{A} \mid \mathcal{B})\, P(\mathcal{C} \mid \mathcal{A}\,\mathcal{B}) \,/\, P(\mathcal{C} \mid \mathcal{B})$$

where $P(\mathcal{A} \mid \mathcal{B})$ denotes the probability of $\mathcal{A}$ given $\mathcal{B}$

$\mathcal{B}\,\mathcal{C}$ denotes $\mathcal{B}$ and $\mathcal{C}$

$\mathcal{B}$ is the prior knowledge

$\mathcal{C}$ is the posterior knowledge

In this thesis the ideas of prior knowledge ("from causes to the effect") are discussed in chapters 2, 3, and 5, while the ideas of posterior knowledge ("from effects to causes") are talked about in chapter 4.

In practice there are three different approaches to model identification:

- White box modeling: The model is completely estimated based on the first principles and does not use any experimental data. Often such a model is referred to as a mechanistic model.

- Grey box modeling: The model is estimated based on a prior knowledge and experimental data.
- Black box modeling: The model identification is performed exclusively based on experimental data.

White box and black box modeling are two extremes of model identification approaches, in practice some prior knowledge exists about the physical system. In general white box modeling is often infeasible in chemical processes, because of the complexity of the chemical system. However, black box modeling of chemical processes is fairly common (Ljung 1999). The topics in this thesis are mainly along the subject of gray box modeling, with a distinction that in the literature prior knowledge in gray box modeling is usually concerned with the structural non-linearity in the model, compared to this thesis where the prior knowledge is generally concerned with prior knowledge about the parameters of the model. This type of prior knowledge, which is used in this thesis, is more in tune with the classical statistical use of prior knowledge.

The issue of the model quality assessment is at the center of any comparison of model identification methods. It is critical to be clear about what is the measure of model quality and what defines a better or worse model. There are many examples of different model quality measures. For example, Ninness and Goodwin (1995) considered variance and bias in the model identification, in both the time-domain and frequency-domain. A more comprehensive model quality evaluation was performed by Dayal (1996) on FIR type models. The quality of the model, and how it affects controller design, was studied by Garcia and Morari (1985). This work allows the assessment of the stability for a controller based on an estimated model. In essence, this work is concerned with the quality of the estimated directions of the gain matrix. There is no one method of evaluating model quality. Each form of model quality assessment is based on the final use of the estimated model. In this thesis, different methods of estimating model quality will be considered in order to evaluate the estimated model's quality. However, unless otherwise mentioned the term "model quality" is associated with the effect of the estimated model on controller stability (Garcia and Morari 1985).

In order to evaluate the issues associated with controller stability, the MIMO systems that are considered in this thesis tend to be ill-conditioned (not rank deficient), since, in such systems the effect of modeling errors and their reduction through the use of prior knowledge is of much greater importance. However, it is important to note that the results are not particular to ill-conditioned systems since well-conditioned systems with poor signal-to-noise ratio can also result in poor model quality.

Since the issues associated with MIMO controller stability of Garcia and Morari (1985) are independent of process dynamics most of the simulations, of chapters 2, 3, and 4, are performed with steady-state information only. Some simulations were performed to compare the results between the dynamic models and steady-state models (i.e.,, models with no dynamics).

Chapter 2

Effect of correct prior knowledge on multivariable model identification for MIMO controller design and operation

2.1. Introduction

Utilization of prior knowledge in model identification has become an increasingly important research area because such knowledge can improve the model quality and potentially eliminate the need for collection of additional data (Tulleken 1992 and 1993). Improvement in model quality can enhance controller design, process monitoring, fault detection, and process optimization. The prior knowledge will come largely from the process engineers, who possess knowledge about the chemical process.

Perhaps the most significant study on the use of prior knowledge in the identification of chemical processes was performed by Tulleken (1993). His study was limited to parsimonious (ARMAX, AutoRegressive Moving Average eXogenous) models for SISO (Single-Input Single-Output) systems. The main conclusion of his work was that a considerable variance reduction could be achieved at the cost of a small increase in the bias of the parameter estimates, due to the addition of constraints. He concluded that the model estimated using constrained identification methods would be more stable for adaptive controller design. Physical knowledge, such as open-loop stability of the model and sign of the static gain, was used in his work. These types of prior knowledge were transformed into a series of linear inequality constraints that were considered when parameter estimation was performed. The results show noticeable improvement in the gain estimate with short data sets.

A similar type of work was done by Timmons (1992), who considered constrained identification of SISO models for biomedical systems. These models were also estimated in ARMAX form using constrained optimization methods. Then the estimated models were used for MPC style controller design. The constraints considered by Timmons (1992) were similar to the ones used by Tulleken (1992), and comparable conclusions about the stability of the designed controller were reached. A predecessor to the work of Timmons (1992) was Chia (1985), who only considered linear equality constraints in his work, although he considered constraints both in the parameters of the state-space model and the time domain model.

All of the previously mentioned works in this area used simple constraints for SISO systems. This chapter will concentrate on a variety of constraints and their effects on Multi-Input Multi-Output (MIMO) systems, since the effects of constraints on MIMO models have not been reported. In addition, model identification is routinely performed for MIMO chemical processes (Anderson and Kummel 1992, Koung and MacGregor 1994).

Although in the field of model identification there is a limited amount of literature on utilization of prior knowledge, other fields of research have performed extensive research on utilization of prior knowledge. For instance, significant work has been done on solving least-square problems with different types of constraint (Lawson and Hanson, 1995). The methods developed for solving least-square problems have been applied to a variety of fields such as econometrics (Judge et al., 1980) and chemometrics (Bro et al., 1998). The types of constraints used in each of these fields are specific to the prior knowledge available in that field. One of the goals in this work is to formulate the constraints that might prove useful in the chemical process industries. The types of constraints that are applied in other fields can assist in formulating constraints for chemical processes. For example, approximate equality constraints (in some literature known as inexact equality constraints) in least-square are applied in econometrics (Theil, 1963), or unimodality constraints are applied in chemometrics for spectroscopy (Bro et al., 1998). In summary, past research has been performed on model estimation in conjunction with constraints and the types of constraints used are system specific. Research has been performed in different fields on utilization of prior knowledge, such as signal processing (Fogel and Huang, 1982), mechanical engineering (Dasgupta et al., 1988), biomedical engineering (Timmons 1992), econometrics (Judge et al., 1980), and chemometrics (Bro et al., 1998). Only a limited amount of research has been performed on the chemical process; those studies tended to consider SISO systems with parsimonious model structures. This chapter will attempt to deal with various examples of prior knowledge that may be present in chemical process. It will also deal mainly with estimation of non-parsimonious models for medium-sized MIMO systems. No literature was found that dealt with the incorporation of prior knowledge in MIMO model identification for the purposes of control in chemical processes.

In this work, the emphasis is not on the model quality in terms of prediction, since it has been shown that any correct constraint will result in an improvement in prediction quality for SISO systems (Tulleken 1992 and 1993). The measure of quality here is controller stability for MIMO systems (Garcia and Morari 1985), which is independent of controller design or tuning. In practice, this condition is uncheckable because the true model is never known. The main contribution of this chapter is in the propagation of model uncertainty into this stability criterion when the prior knowledge is correct. In some special cases, a closed form solution to this problem is derived; in other cases, Monte Carlo style simulations are used to propagate the model uncertainty.

Traditionally in control literature, the issue of stability is in regards to the s-domain root location, while in this thesis the emphasis is on testing the stability of MIMO systems at low (zero) frequency. Small data sets, low signal-to-noise ratio and changing operating conditions result in unreliable dynamic model for chemical processes. Therefore, in MIMO systems it is often the case that only a matrix of reliable steady-state gains is available (Grosdidier, et al. 1985, Papastathopoulou and Luyben 1990, Pensar and Waller 1993, Skogestad et al. 1988). In this thesis, model quality is evaluated primarily by evaluating the instability resulting from error in the directionality of this steady-state gain matrix.

Two distinct classes of models can be estimated in an identification process: parsimonious and non-parsimonious models. Parsimonious models are typically low order models. They are called parsimonious because the number of parameters employed is small, obeying the principle of parsimony (whereby the minimum number of statistically significant parameters are employed in a model, while still enabling it to represent the true process dynamics). Non-parsimonious models are high order models, which commonly are in the form of a finite impulse response (FIR) model (although not limited to this form). For the case of n_x process inputs and n_y process outputs with only random fluctuations (white noise) as a noise model the multivariable model is:

$$\begin{bmatrix} y_{1,t} \\ y_{2,t} \\ \vdots \\ y_{n_y,t} \end{bmatrix} = \begin{bmatrix} g_{1,1}(q^{-1}) & g_{1,2}(q^{-1}) & \cdots & g_{1,n_x}(q^{-1}) \\ g_{2,1}(q^{-1}) & g_{2,2}(q^{-1}) & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ g_{n_y,1}(q^{-1}) & \cdots & \cdots & g_{n_y,n_x}(q^{-1}) \end{bmatrix} \begin{bmatrix} u_{1,t} \\ u_{2,t} \\ \vdots \\ u_{n_x,t} \end{bmatrix} + \begin{bmatrix} a_{1,t} \\ a_{2,t} \\ \vdots \\ a_{n_y,t} \end{bmatrix} \tag{2.1}$$

$$g_{i,j}(q^{-1}) = \sum_{k=0}^{l_{i,j}} v_{k,i,j} q^{-k}$$

where $y_{i,t}$ is the i^{th} output at time t

$u_{j,t}$ is the j^{th} input at time t

$a_{i,t}$ is the white noise added to the process output at time t

$g_{i,j}(q^{-1})$ is the transfer function relation between the i^{th} output and the j^{th} input

$v_{k,i,j}$ is the k^{th} impulse response coefficient for the $g_{i,j}(q^{-1})$ transfer function

$l_{i,j}$ is the total number of impulse coefficient used for the relationship between the i^{th} output and the j^{th} input.

Where l is chosen to be large enough to adequately approximate the true process dynamics and only a white noise is added to the process outputs. The advantages of such a model are that the parameters can easily be estimated using linear regression methods (Ljung 1999), the only structural assumption is in choosing the number of impulse weights l, and they are not limited to the responses produced by low order (parsimonious) models. The major disadvantage with such a model is the large number of parameters used in model estimation (Ljung 1999).

In this chapter, all of the simulations for dynamic systems utilize a non-parsimonious model in the form of a FIR, unless otherwise stated. In this way, bias issues arising from the choice of an inadequate process model are avoided. Any bias can be attributed to the prior knowledge. Although not illustrated here, in most cases the result can be extended to parsimonious models as well.

2.2. Example of Problem

Assume that the plant is linear and its true model is given by (2.2). This plant is poorly conditioned (condition number of the gain matrix is 100); however, it is not singular.

$$G(s)=\frac{e^{-1s}}{5s+1}\begin{pmatrix}5 & 5\\ 0.1 & 0\end{pmatrix} \tag{2.2}$$

For the purpose of this problem, we assume that there is exact prior knowledge about the process dynamics; hence, there are no model mismatches in the dynamic portion of the models.

Consider the following two estimates of this plant, which have resulted from an identification study:

$$\hat{G}_1(s)=\frac{e^{-1s}}{5s+1}\begin{pmatrix}5.0274 & 5.0369\\ -0.01589 & -0.05799\end{pmatrix} \tag{2.3}$$

$$\hat{G}_2(s)=\frac{e^{-1s}}{5s+1}\begin{pmatrix}5.0275 & 5.0369\\ -0.01821 & 0\end{pmatrix} \tag{2.4}$$

The first model was estimated with no prior knowledge about the gain, while the second model was estimated with a constraint ($g_{2,2} \geq 0$, which is a steady-state prior knowledge about the relationship between the second input and the second output). The additive steady-state model mismatch, which is only based on the estimated steady-state gain matrix ($\hat{G}$) and the true gain matrix (G), is measured by the Frobenius norm:

$$\left\|G - \hat{G}_1\right\|_F = 0.1375$$
$$\left\|G - \hat{G}_2\right\|_F = 0.1269 \qquad (2.5)$$

As expected, the second estimated model, which was estimated with a constraint, has a smaller mismatch than the first model. One would expect that a model with a smaller model mismatch, which in this case was due to addition of prior knowledge, would result in a better controller. If a model-based multivariable controller (such as DMC) is designed based on these models, the response of the closed-loop system to a set-point change is simulated in Figures 2.1 and 2.2. In Figure 2.1, use of the estimated model (equation (2.3)) resulted in a stable controller; however, the estimated model that utilized the prior knowledge (2.4) resulted in an unstable controller shown in Figure 2.2. It will be shown that the second estimated model does not meet the controller stability criteria (CSC) of Garcia and Morari (1985) (for a detailed definition of the CSC see Appendix 1). Therefore, the resulting closed-loop system will be unstable independent of controller design (a similar analysis was performed by Li and Lee 1996) provided that the controller tuning is reasonable. In the case of DMC (Dynamic Matrix Control), this implies that the prediction horizon must be longer than the duration of inverse response (or nonminimum phase), and the weighting matrix for the input and the output are not too aggressive.

This example illustrates that for MIMO systems, a constraint, even though it is correct, will not necessarily result in an improved controller. It also illustrates that there are different measures of model quality (the Frobenius norm and the CSC) and that a

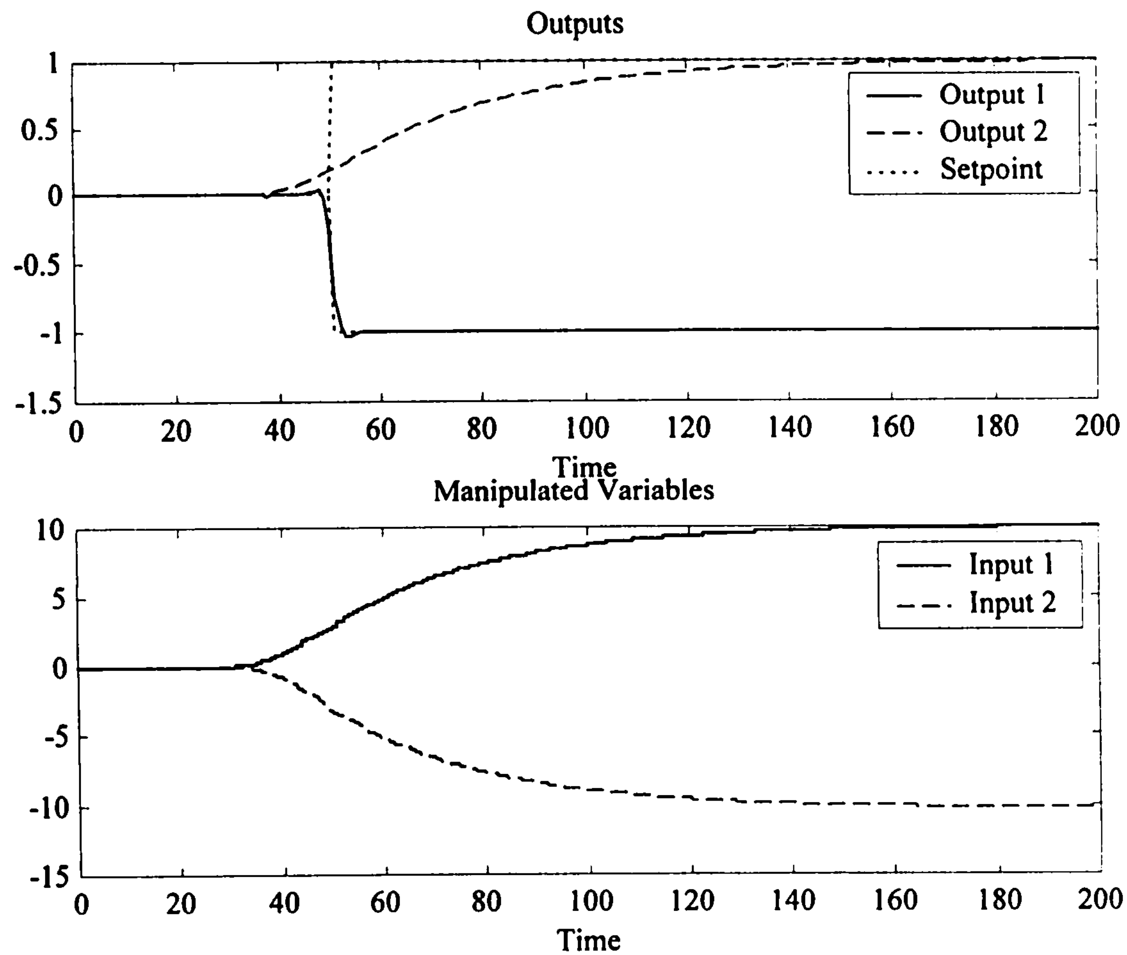


Figure 2.1: A DMC controller performance for the model estimated with no prior knowledge. The controller tuning parameters are: M = 7 (input horizon), P = 20 (output horizon), all the inputs and outputs are weighted equally, and there are no bounds on any input or output.

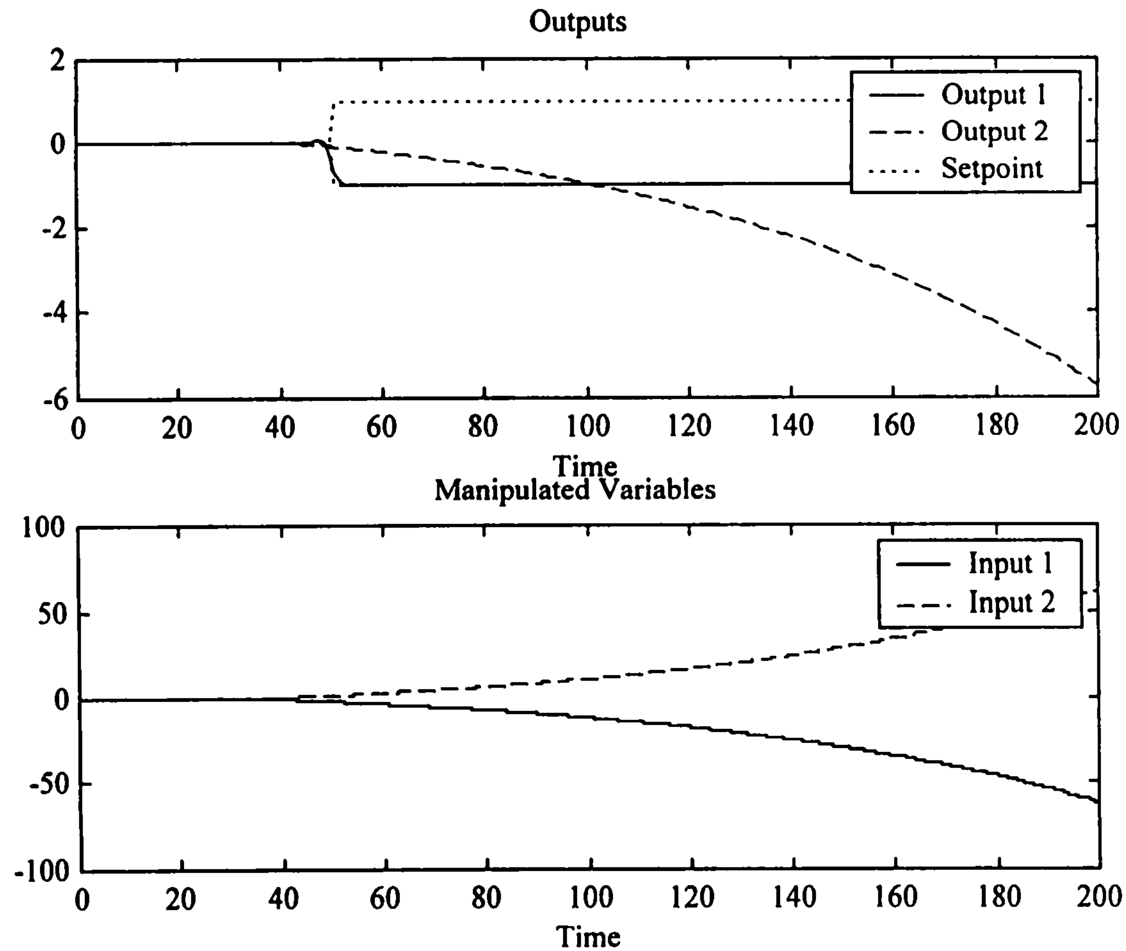


Figure 2.2: A DMC controller performance for the model estimated with a prior knowledge of $g_{2,2} \geq 0$. The controller tuning parameters are: M = 7, P = 20, all the inputs and outputs are weighted equally, and there are no bounds on any input or output.

constraint may improve the model quality by some measures (Frobenius norm) while at the same time deteriorate the model quality by a different measure (CSC). It is also important to distinguish that the Frobenius norm is a univariate measure of model quality, while CSC is a multivariate measure of model quality. The prior knowledge, which in this case stated that $g_{2,2} \geq 0$, was also a univariate prior knowledge. The purpose of this work is to investigate which constraints (multivariate or univariate) improve and/or deteriorate model quality for MIMO controller stability. Actual controller design and tuning will not be considered, since CSC is independent of controller tuning and design (Garcia and Morari (1985), see Appendix 1).

2.3. Analytical Analysis

In this section, the uncertainty of the model is analytically propagated to the controller stability criteria (CSC) of Garcia and Morari (1985). In the first section, the CSC is explained, and the problem associated with propagation of model uncertainty to this criterion is clarified. For 2x2 systems it is possible to observe the distribution of the angle between the gain vectors instead of observing the CSC; this is illustrated in section 2.3.2. In the last section, this methodology is extended to larger systems by propagating the model uncertainty to the determinant of the gain matrix.

2.3.1. Interpretation of the controller stability criteria (CSC)

Model quality can be evaluated in many different ways. The measure of model quality is dependent on the end use of the model. In this work, model quality for MIMO chemical process control design is being considered. The emphasis is on the quality of the steady-state estimation, which is important in chemical processes (McKay et al. 1997, and Andersen et al. 1992). For this purpose, the controller stability criteria (CSC) of Garcia and Morari (1985) will be used. In the past, other researchers have used this criterion for evaluation of model quality in chemical processes (Dayal and MacGregor 1996, Koung 1991, Andersen et al. 1992).

The stability of a multivariable control system using an empirical model can be evaluated using the steady state controller stability criteria (CSC) of Garcia and Morari (1985). They have shown that a diagonal first-order exponential filter such as (2.6) can always stabilize the approximate model inverse $\hat{G}^{-1}$ for some value of α_i as long as condition (2.7) holds.

$$F(z) = diag\left\{\frac{1-\alpha_i}{1-\alpha_i z^{-1}}\right\}, i = 1,\ldots,r, 0 \le \alpha_i < 1 \tag{2.6}$$

$$\mathrm{Re}\left(\lambda_i\left(G\hat{G}^{-1}\right)\right) > 0, \forall i \tag{2.7}$$

This condition is referred to as "integral controllability" (by Morari and Zafiriou 1989) or "robust stability condition" (by Koung and MacGregor 1994), and it presents a steady-state condition for stability (in this thesis, it is referred to as controller stability criteria, CSC). As long as this condition is not violated, a multivariable closed-loop system with no offset can be designed. Notice that CSC (2.7) is independent of the controller design. Controller stability criteria is similar to the SISO condition that the estimated model gain has to have the same sign as the gain of the true model. In the case of SISO system, when the sign of the gain is estimated incorrectly, no negative feedback controller will result in a system with zero offset. In this thesis, a model that results in a control system that satisfies CSC will be referred to as a "stable model" (for more detail on CSC see Appendix 1).

The controller stability criteria (CSC) (2.7) can be used to evaluate the quality of the identified models (2.3) and (2.4).

$$\mathrm{Re}\left(\lambda_i\left(G(0)\hat{G}_1(0)^{-1}\right)\right)=\{2.39,0.99\}>0,\forall i$$
$$\mathrm{Re}\left(\lambda_i\left(G(0)\hat{G}_2(0)^{-1}\right)\right)=\{-5.49,0.99\}\not>0,\forall i \tag{2.8}$$

Based on this, the estimate gain matrix $\hat{G}_1$ passes the CSC, while the second estimated gain matrix $\hat{G}_2$, which incorporates the prior knowledge, fails the CSC. This result confirms the simulation results in Figures 2.1 and 2.2, which shows stable and unstable control systems respectively.

From (2.7) it can be seen that, in any 2x2 system both of the eigenvalues of $G\times\hat{G}^{-1}$ have to be positive (this refers to the real part only) for controller stability criteria (CSC) to hold. Since the trace is the sum of the eigenvalues and the determinant is the product of the eigenvalues, a necessary and sufficient condition for stability is that both the trace and the determinant have to be positive (Koung 1991). Similarly, in the 3x3 system, all the eigenvalues (of $G\times\hat{G}^{-1}$) have to be positive for CSC to hold. A necessary condition for stability in this case is that both the trace and determinant have to be positive; however this is not a sufficient condition, since there are three eigenvalues (Koung 1991).

The goal of the following sections is to propagate the uncertainty of the model into the estimated eigenvalues of $G\times\hat{G}^{-1}$. The effect of the constraints can then be evaluated using this method of propagation. Yet only in some special cases could a closed form solution to this problem be derived. In this chapter, two different methods are used to propagate the model uncertainty into the estimated eigenvalues of $G\times\hat{G}^{-1}$:

- After a series of simplifying assumptions, in certain cases it is possible to derive a closed form analytical solution to this propagation.
- In other cases, which are more realistic, Monte Carlo style simulations can provide insight into this problem.

For 2x2 and 3x3 systems, one can geometrically look at how the gain matrix affects the stability (Koung 1991 has an extensive explanation of the geometrical issues involved). While it is not possible to study the distribution of the eigenvalue for this problem analytically, it is possible to derive a closed form solution to the distribution of the angle between the gain vectors in smaller systems (2x2 and 3x3).

2.3.2. Angle distribution

In a 2x2 system, the angle between two gain vectors can be defined by the following:

$$\alpha = \tan^{-1}\left(\frac{g_{2,1}}{g_{1,1}}\right) - \tan^{-1}\left(\frac{g_{2,2}}{g_{1,2}}\right) \tag{2.9}$$

For example, in the case of the system (2.2), this angle is $\tan^{-1}\left(\frac{0.1}{5}\right) = 1.14°$. In larger systems, the angle refers to the angle between each gain vector and the hyper-plane that defines all the other gain vectors. The importance of this angle in the estimated model can be seen for (2.3) and (2.4). The true model (2.2) has an angle of 1.14°, while the estimated models (2.3) and (2.4) have angles of 0.479° and –0.208° respectively. The fact that the estimated angle for (2.4) has a different sign compared to (2.2) is the reason for the unstable control system seen in Figure 2.2. In more general terms, the effect of the error in the angle can be seen from the geometrical results. The geometrical results suggested that while an error in angle may result in an unstable controller, it is not the sole reason for a system to result in a USC system (Koung 1991).

If the ill-conditioning in the gain matrix was due to a small angle between the gain direction (in the case shown previously, the angle between the two gain directions was 1.14° in a 2x2 system) the probability of a system going unstable can be approximated by the probability of the angle changing sign. This suggests that, in this particular case (2x2 system), one could just look at the probability density function (p.d.f.) of the angle instead of the p.d.f. of the eigenvalues to evaluate CSC. Based on the following assumptions, an analytical expression for the p.d.f. of the angle was derived (Appendix 2):

- The system is ill conditioned because of the angle only.
- The standard deviation in the angle is not large (<50°).
- The cumulative density function (c.d.f.) of the angle is approximately normally distributed, when the angle is zero.

Although the simulation results are not shown here under those assumptions, the analytical result compared very well with Monte Carlo results.

2.3.3. Distribution of the determinant

In the larger systems, one can look at the distribution of the determinant instead of the angle, since there will be multiple angles in systems larger than 2x2. Consider the following for an *nxn* system:

$$\begin{aligned}\det\left(G \times \hat{G}^{-1}\right) &= \lambda_{G\times\hat{G}^{-1},1} \times \lambda_{G\times\hat{G}^{-1},2} \times \cdots \times \lambda_{G\times\hat{G}^{-1},n} \\ &= \det(G) \times \det\left(\hat{G}^{-1}\right) \\ &= \det(G) / \det\left(\hat{G}\right) \\ &= \frac{\lambda_{G,1} \times \lambda_{G,2} \times \cdots \times \lambda_{G,n}}{\lambda_{\hat{G},1} \times \lambda_{\hat{G},2} \times \cdots \times \lambda_{\hat{G},n}}\end{aligned} \tag{2.10}$$

The above expression has to be positive for a system to be stable. This is not a sufficient condition; however, it is a necessary condition. A negative value for the above expression suggests that at least one of the eigenvalues (real part) is negative, resulting in an unstable control system (UCS).

For this condition to hold, the sign of the numerator and the denominator must agree:

$$sign\left(\lambda_{G,1}\times\lambda_{G,2}\times\cdots\times\lambda_{G,n}\right)=sign\left(\lambda_{\hat{G},1}\times\lambda_{\hat{G},2}\times\cdots\times\lambda_{\hat{G},n}\right) \quad (2.11)$$

$$or$$

$$sign\left(\det(G)\right)=sign\left(\det\left(\hat{G}\right)\right)$$

Certainly the above condition is violated when odd numbers of eigenvalues of G and $\hat{G}$ have opposite signs. However, if even-numbered eigenvalues of G and $\hat{G}$ have opposite signs, the above condition will not be violated.

2.3.3.1. Equality constraints

Using the previous expression (2.11), the variance in the gain matrix was propagated to determine the probability of this condition being violated (Appendices 3 and 5). Making the following assumptions:

- $\det\left(\hat{G}\right)\sim N\left(E\left(\det\left(\hat{G}\right)\right),\sigma^2_{\det(\hat{G})}\right)$, this is especially important near $\det\left(\hat{G}\right)=0$.
- It is also assumed that $\sigma_{\det(\hat{G})}$ is relatively small. If $\sigma_{\det(\hat{G})}$ is too large, more than one eigenvalue may change signs.

This results in an expression that estimates the probability of unstable control system (UCS) when there is a correct linear equality constraint and $\det(G)<0$ (see Appendix 5):

$$P\left(\operatorname{Re}\left(\lambda_i\left(G\hat{G}^{-1}\right)\right) \not> 0, \forall i\right) \approx P\left(sign\left(\det(G)\right) \neq sign\left(\det\left(\hat{G}\right)\right)\right)$$
$$\approx P\left(Z > \frac{0 - E\left(\det\left(\hat{G}\right)\right)_{CLS}}{\sqrt{\operatorname{var}\left(\det\left(\hat{G}\right)\right)_{CLS}}}\right)$$

(2.12)

where Z is a standard normal random variable

CLS is the constrained least square estimator

The above expression implies that the probability of the unstable control system (UCS) can be approximated by the probability of the gain matrix determinant changing sign. To use this condition for estimating the probability of unstable control system based on the CSC, the following assumptions have to be made, in addition to the assumptions made in the derivation of the probability expression:

- $P\left(\operatorname{Re}\left(\lambda_{G\times\hat{G}^{-1},i}\right) > 0, \forall i\right) > 0.5$, as $P\left(\operatorname{Re}\left(\lambda_{G\times\hat{G}^{-1},i}\right) > 0, \forall i\right) \rightarrow 1$ the expression (2.12) appears to hold better.
- Only an odd number of $\lambda_{\hat{G},i}$ will change sign. If the ill-conditioning is only caused by angle, only one (an odd number) of $\lambda_{\hat{G},i}$ will change sign (this was illustrated by Koung 1991).

2.3.3.2. Inequality constraints

Exact theoretical result is very difficult to derive for inequality constraint, since such a constraint would certainly result in a non-normal distribution of, at least, the parameter involved in the constraint. Figure 2.3 illustrates the gain matrix element distribution when there is an inequality constraint on one of the gain elements (this distribution was obtained using a Monte Carlo simulation for a 5x5 with 100 realizations,

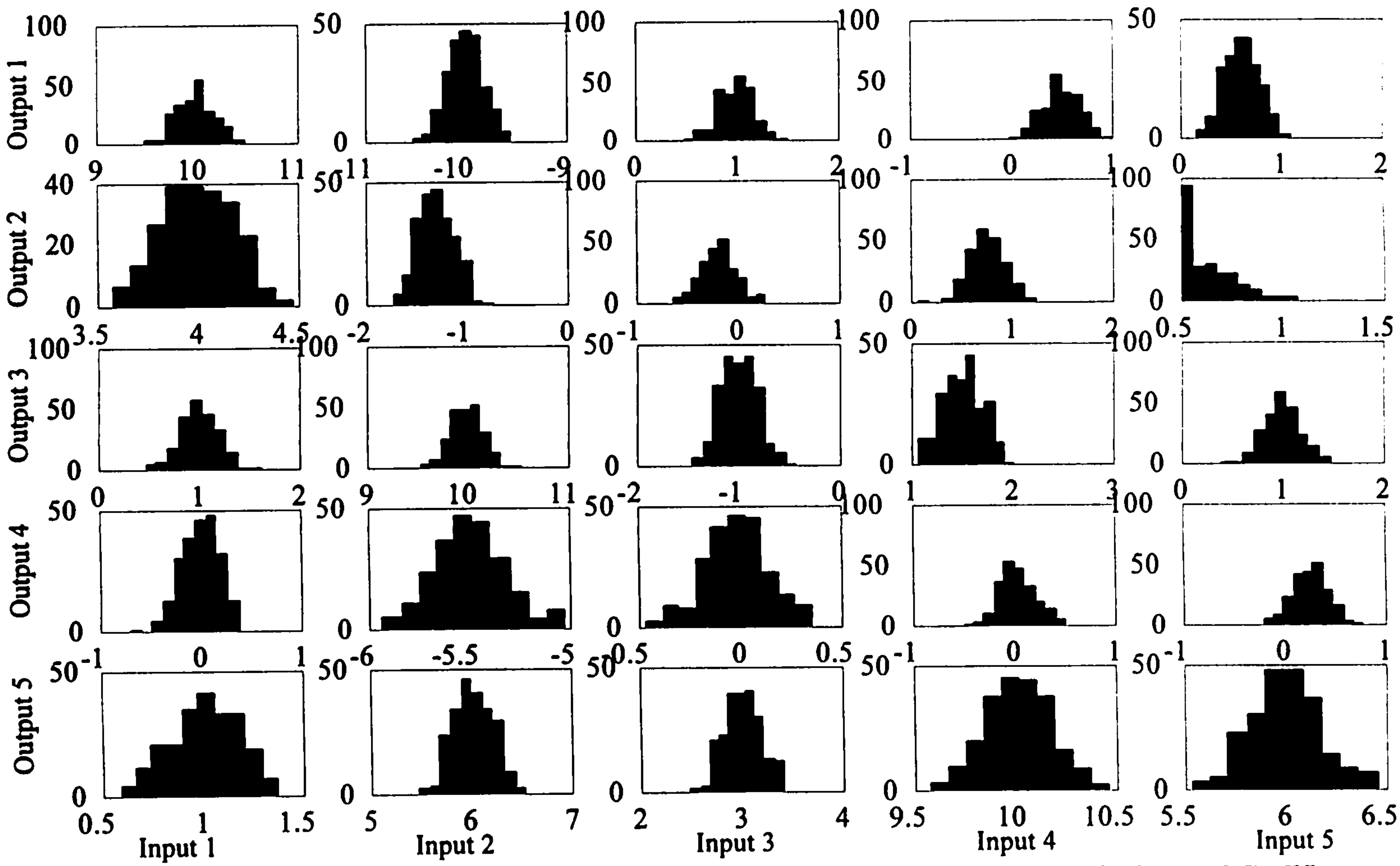


Figure 2.3: The effect of inequality constraint on the distribution of the gain elements (the constraint is $g_{2,5}>0.5$). Where the true gain matrix is G_4 of (2.19) and the PRBS magnitude is ±0.5 (for more detail explanation of the Monte Carlo setting see Appendix 6). The distribution is obtained via a Monte Carlo simulation with 100 realizations and 500 observation in each realization. The Y-axis is the frequency of occurrence while the X-axis is the value of the estimated gain.

which is explained in more detail in section 2.4.1). In this case, the gain element $g_{2,5}$ appears to have a non-normal distribution, while the other parameters appear normally distributed.

Under special conditions, the variance and expected value of gain matrix elements distribution can be estimated theoretically. The variance and mean estimated values might be propagated to the determinant of the gain matrix. Assuming that the determinant is normally distributed, although the elements of the gain matrix are not all normally distributed, the probability that the determinant will change sign can be estimated (Appendices 4, and 5). Observing the distribution of the determinant as shown in Figure 2.4 (this is the determinant of the gain matrix for the Monte Carlo simulation shown in Figure 2.3) can test the validity of this assumption. Therefore the assumptions made are:

- There is no covariance between the gain elements (i.e. the covariance of gain parameter is a diagonal matrix). Covariance terms can be considered in the same framework; however, they complicate the calculations appreciably.
- The determinant of the gain matrix is normally distributed, although individual gain elements may be non-symmetrically (i.e., non-normally) distributed.
- The inequality constraint includes the true value.

This results in an expression that estimates the probability of UCS when there is correct linear inequality constraint and $\det(\hat{G}) < 0$ (see Appendix 5):

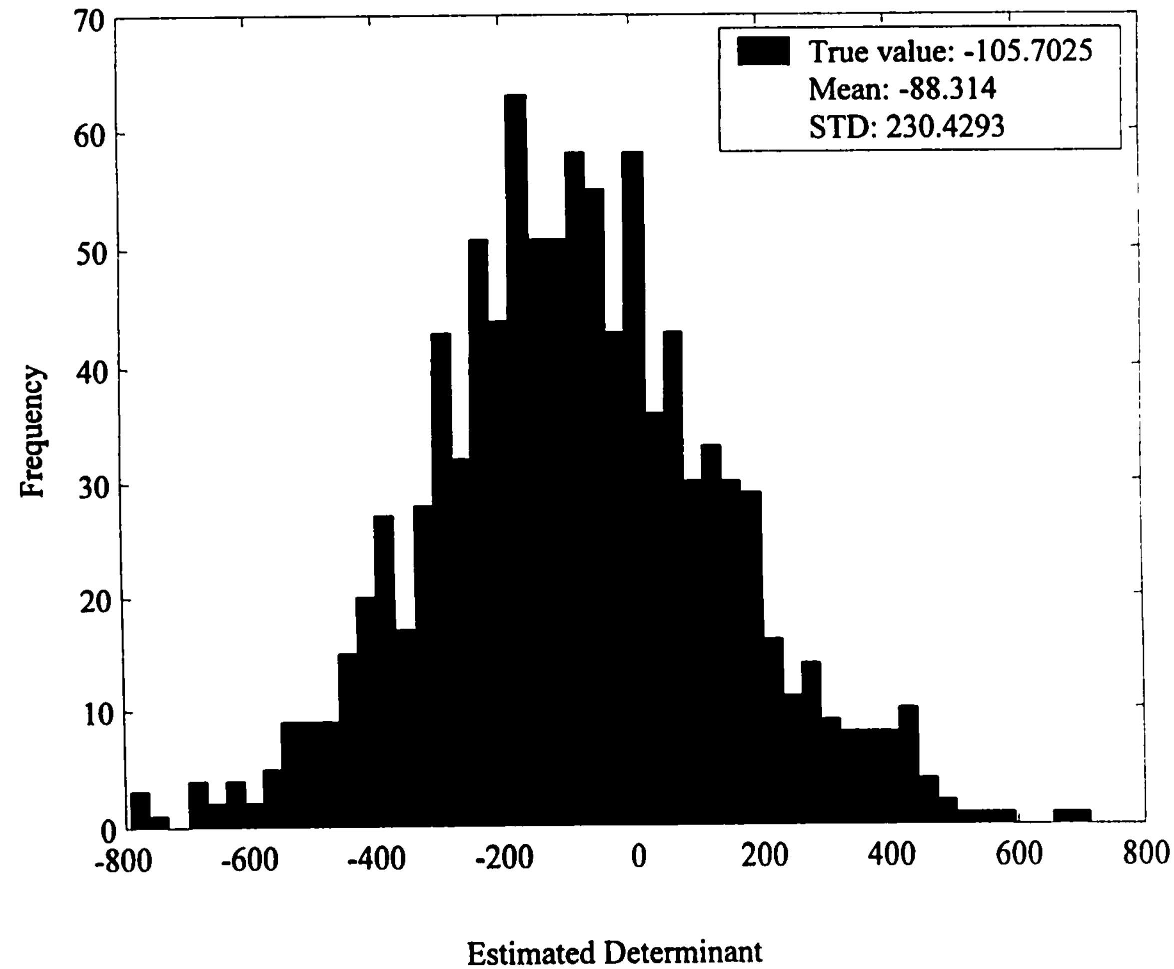


Figure 2.4: The effect of inequality constraint on the distribution of the determinant (the constraint is $g_{2,5}>0.5$). This plot is based on the simulation result shown in the last Figure.

$$P\left(\operatorname{Re}\left(\lambda_i\left(G\hat{G}^{-1}\right)\right) \not> 0, \forall i\right) \approx P\left(sign\left(\det(G)\right) \neq sign\left(\det\left(\hat{G}\right)\right)\right)$$
$$\approx P\left(Z > \frac{0 - E\left(\det\left(\hat{G}\right)\right)_{inequality}}{\sqrt{\operatorname{var}\left(\det\left(\hat{G}\right)\right)_{inequality}}}\right)$$

(2.13)

If the addition of an inequality constraint results in a higher probability of an unstable controller system (UCS) based on CSC, as bounds of this inequality constraint approach the true value (δ→0) the probability of UCS increases (Figure 2.5). The probability of UCS can be greater with inequality constraint compared to no constraint, because there can be a bias in the estimated determinant of the gain matrix ($E\left(\det\left(\hat{G}\right)\right) \neq \det(G)$). In the theoretical part of this work, the expected value of the determinant is assumed to be the determinant of the expected values of the gain matrix elements. This is valid if one assumes that the elements of the gain matrix are independent.

For example, in a 2x2 system with the following gain matrix estimate: $\hat{G} = \begin{pmatrix} \hat{g}_{1,1} & \hat{g}_{1,2} \\ \hat{g}_{2,1} & \hat{g}_{2,2} \end{pmatrix}$ the determinant is $\hat{g}_{1,1}\hat{g}_{2,2} - \hat{g}_{1,2}\hat{g}_{2,1}$ and the expected value of the determinant is $E\left(\hat{g}_{1,1}\hat{g}_{2,2} - \hat{g}_{1,2}\hat{g}_{2,1}\right) = E\left(\hat{g}_{1,1}\hat{g}_{2,2}\right) - E\left(\hat{g}_{1,2}\hat{g}_{2,1}\right)$, and if the elements are independent then the expected value of the determinant is $E\left(\hat{g}_{1,1}\right)E\left(\hat{g}_{2,2}\right) - E\left(\hat{g}_{1,2}\right)E\left(\hat{g}_{2,1}\right)$. Similarly in a 3x3 system:

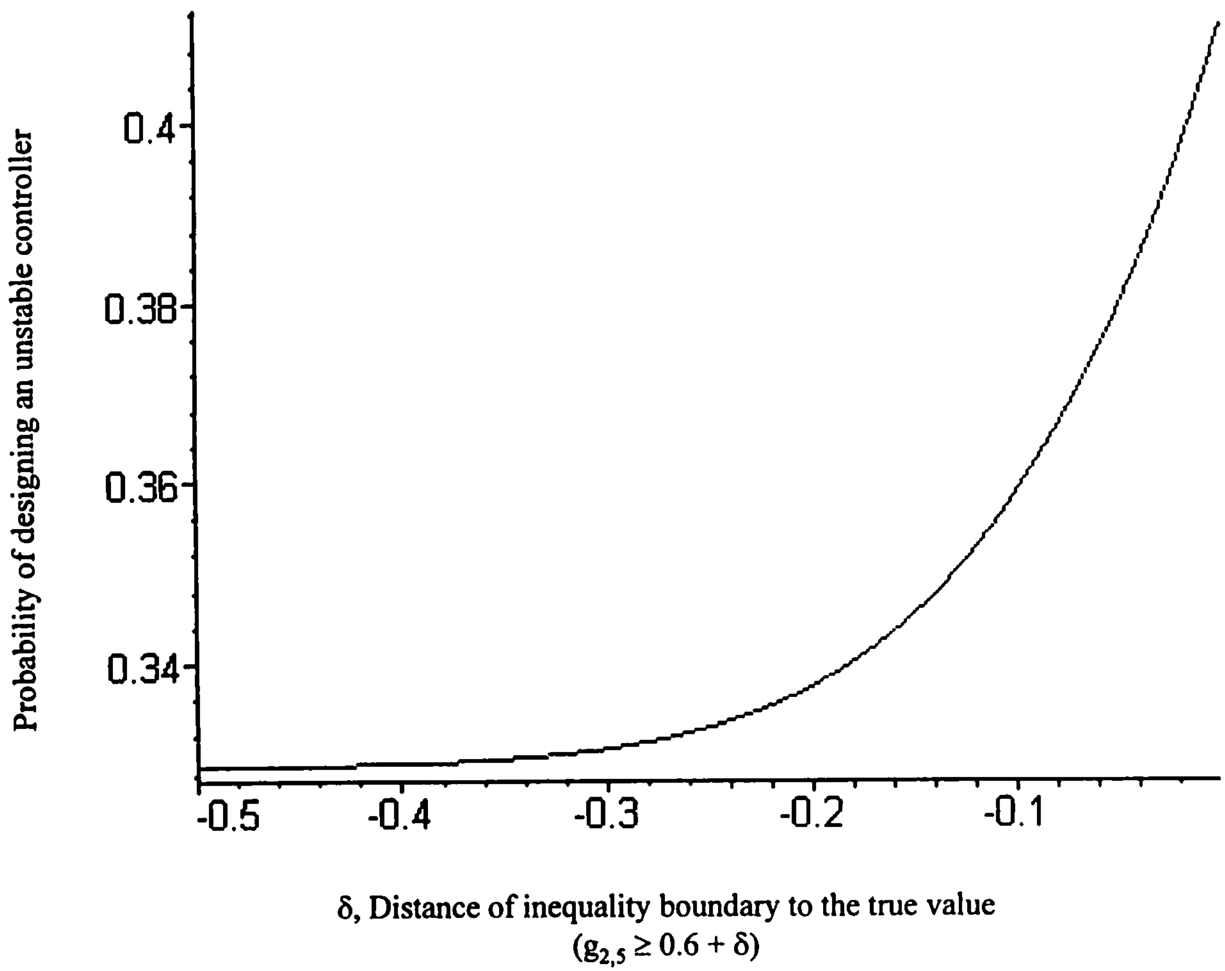


Figure 2.5: The effect of inequality constraint on the probability of UCS estimated analytically for G_4 of (2.19) as the bound of the inequality constraint approaches the true value (the constraint is on $g_{2,5}$)

$$E\left(\det\begin{pmatrix}\hat{g}_{1,1} & \hat{g}_{1,2} & \hat{g}_{1,3}\\ \hat{g}_{2,1} & \hat{g}_{2,2} & \hat{g}_{2,3}\\ \hat{g}_{3,1} & \hat{g}_{3,2} & \hat{g}_{3,3}\end{pmatrix}\right) = E(\hat{g}_{1,1})E(\hat{g}_{2,2})E(\hat{g}_{3,3}) + E(\hat{g}_{1,2})E(\hat{g}_{2,3})E(\hat{g}_{3,1})$$
$$+E(\hat{g}_{2,1})E(\hat{g}_{3,2})E(\hat{g}_{1,3}) - E(\hat{g}_{3,1})E(\hat{g}_{2,2})E(\hat{g}_{1,3})$$
$$-E(\hat{g}_{2,1})E(\hat{g}_{1,2})E(\hat{g}_{3,3}) - E(\hat{g}_{3,2})E(\hat{g}_{2,3})E(\hat{g}_{1,1}) \qquad (2.14)$$

Notice that it is common practice to identify a MISO model from each output individually and then combine the individual models into a combined MIMO model (Ljung 1999). In essence, this methodology assumes that there are no common or correlated parameters among the models for the different outputs. Effectively, this is the same assumption that was made in the derivation of the above equation and is a reasonable assumption.

Therefore, after a series of assumptions, (2.13) can be evaluated for any *nxn* system. This is accomplished by utilizing (2.14) to estimate the expected value of the determinant. The variance of the determinant, required in (2.13), is estimated by propagating the variance expression for inequality-restricted estimator (Judge et al. 1980) of the gain parameters to the determinant of the gain matrix (see Appendix 5 for more detail).

2.4. Simulation results

Different Monte Carlo simulations with several systems were performed to illustrate various issues. In the section 2.4.1, the Monte Carlo simulation approach is described. The means of using the constraint (or prior knowledge) in the optimization problem is illustrated in section 2.4.2. Small systems (2x2) with and without dynamics are considered in sections 2.4.3 and 2.4.4, respectively. In section 2.4.5 a larger system (5x5) with no dynamics is considered. In the last section, the theoretical results of section 2.3 are compared with some of the simulation results. A comprehensive list of the simulations results based on these systems is shown in Appendix 6.

2.4.1. Simulation approach

To evaluate the effect of the prior knowledge on the estimated model, Monte Carlo style simulations were performed. In these simulations, the confidence intervals of the estimated models were estimated. It is important to distinguish that these confidence intervals do not represent confidence interval of individual trials (or realizations); rather they were based on the many trials (or realizations) that make up a Monte Carlo simulation.

A set of k different input signal realizations (with N data points in each realization) resulted in a Monte Carlo type simulation. For each of the k data sets the models were estimated. Utilizing all the k models, confidence intervals and the probability of UCS were calculated. The k different realizations were the same within a set of Monte Carlo simulations but are not constant for all the sets of Monte Carlo simulations.

2.4.2. Methods of implementing constraints

As mentioned previously, for some types of constraints there are analytical solutions. However, all the numerical simulation results are based on solving the problem with a Quadratic Programming (QP) or Sequential Quadratic Programming (SQP) algorithm (depending on if the constraint was linear or non-linear). The MATLAB optimization toolbox was used (for more detail on the optimization methods see Optimization Toolbox (1999)). In the optimization problem, both the analytical gradient and Hessian were provided to speed up the optimization problem. In some cases, the numerical results from the QP were compared with the Constrained Least Squares (CLS) solution to validate the results. These results were in agreement with one another.

2.4.3. 2x2 system with dynamics

The base case 2x2 system with dynamics (as described by (2.15)) had a PRBS with magnitude of 0.1 and switching interval of 4 for both inputs with 1000 data collected in each of 500 Monte Carlo simulations. Both $g_{1,1}(q^{-1})$ and $g_{2,2}(q^{-1})$ were estimated with a 20 parameter FIR; however, $g_{1,2}(q^{-1})$ and $g_{2,1}(q^{-1})$ were estimated using a 10 parameter FIR (20 and 10 are based on the settling time of the process). An AR(1) noise model was estimated for both outputs. The variances of both white noise sequences (a_t) were 1, and they were independent of one another (Appendix 6).

$$\begin{bmatrix} y_{1,t} \\ y_{2,t} \end{bmatrix} = G\left(q^{-1}\right)\begin{bmatrix} u_{1,t} \\ u_{2,t} \end{bmatrix} + N\left(q^{-1}\right)\begin{bmatrix} a_{1,t} \\ a_{2,t} \end{bmatrix} \tag{2.15}$$

$$G_1\left(q^{-1}\right) = \begin{pmatrix} \dfrac{2q^{-3}}{1-.8q^{-1}} & \dfrac{-q^{-2}}{1-.6q^{-1}} \\ \dfrac{q^{-4}}{1-q^{-1}+.25q^{-2}} & \dfrac{-q^{-3}}{1-.9q^{-1}+.6q^{-2}} \end{pmatrix} \tag{2.16}$$

$$N\left(q^{-1}\right) = \begin{pmatrix} \dfrac{1}{1-.7q^{-1}} & 0 \\ 0 & \dfrac{1}{1-.9q^{-1}} \end{pmatrix} \tag{2.17}$$

where $y_{1,t}$ and $y_{2,t}$ are the first and second process outputs at time t

$u_{1,t}$ and $u_{2,t}$ are the first and second process inputs at time t

$N(q^{-1})$ is the noise model

a_t is the i.i.d. white noise with ($a_t \sim N(0,1)$)

The resulting gain matrix for (2.16), which has a condition number of 29, is:

$$G_1 = \begin{pmatrix} 10 & -2.5 \\ 4 & -1.43 \end{pmatrix}$$

The base case simulation considers estimating a FIR model without any prior knowledge (note that the length of FIR terms is not considered a prior knowledge in this work). A summation of the FIR parameters (v) results in the individual gain estimate, which in turn results in the estimated gain matrix.

$$\hat{g}_{i,j} = \sum_k \hat{v}_{k,i,j}$$

where $\hat{g}_{i,j}$ is the element (i, j) of the estimated gain matrix ($\hat{G}$)

$\hat{v}_{k,i,j}$ is the k^{th} estimated impulse response coefficient for $\hat{g}_{i,j}\left(q^{-1}\right)$

The results of the estimated gain matrix for this Monte Carlo simulation are shown in Figure 2.6. The individual points on the plot represent the gain estimated based on a realization (note 500 realizations result in a Monte Carlo simulation in this case), while the ellipse represents the 95% confidence interval of these estimates. The mean of all the realizations is shown by a circle ("o"); the resulting gain vector is denoted by dash line ("----"), and solid lines represent the true gain vectors. Since the mean and the true value overlay one another (in Figure 2.6), there is no noticeable bias between the true model and the mean of all the realizations. The purpose of this plot is primarily to illustrate the directionality of the gain matrix and how the variance in the gain matrix plays a role in the directionality of individual gain matrix estimates. This directionality in turn affects the probability of an unstable control system (for *P(USC)* with no constraint see Appendix 6 Table A.1). The key issue that should be noted in Figure 2.6 (and other similar figures, such as Figures 2.7 and 2.8, which will be shown) is how the mean of the

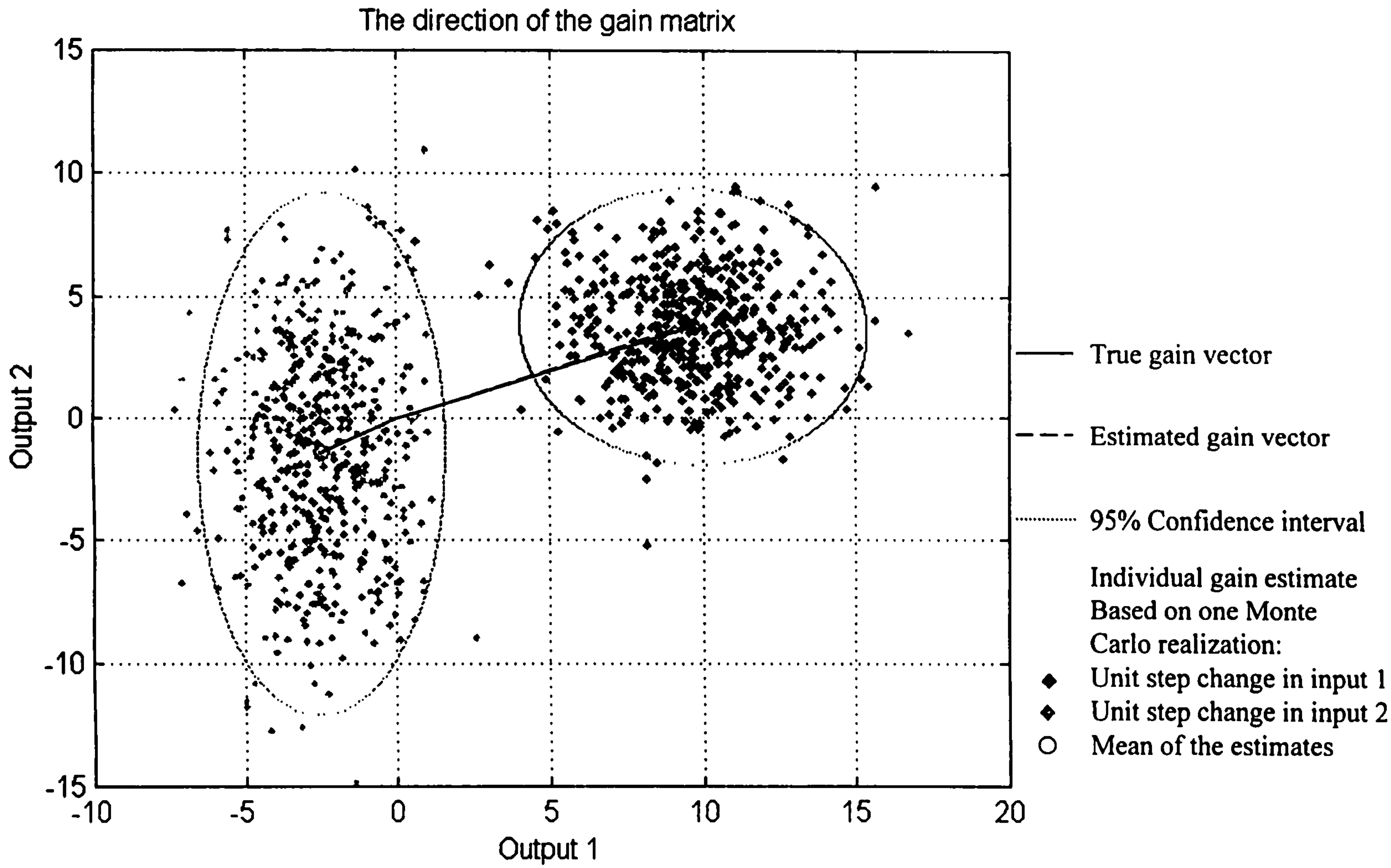


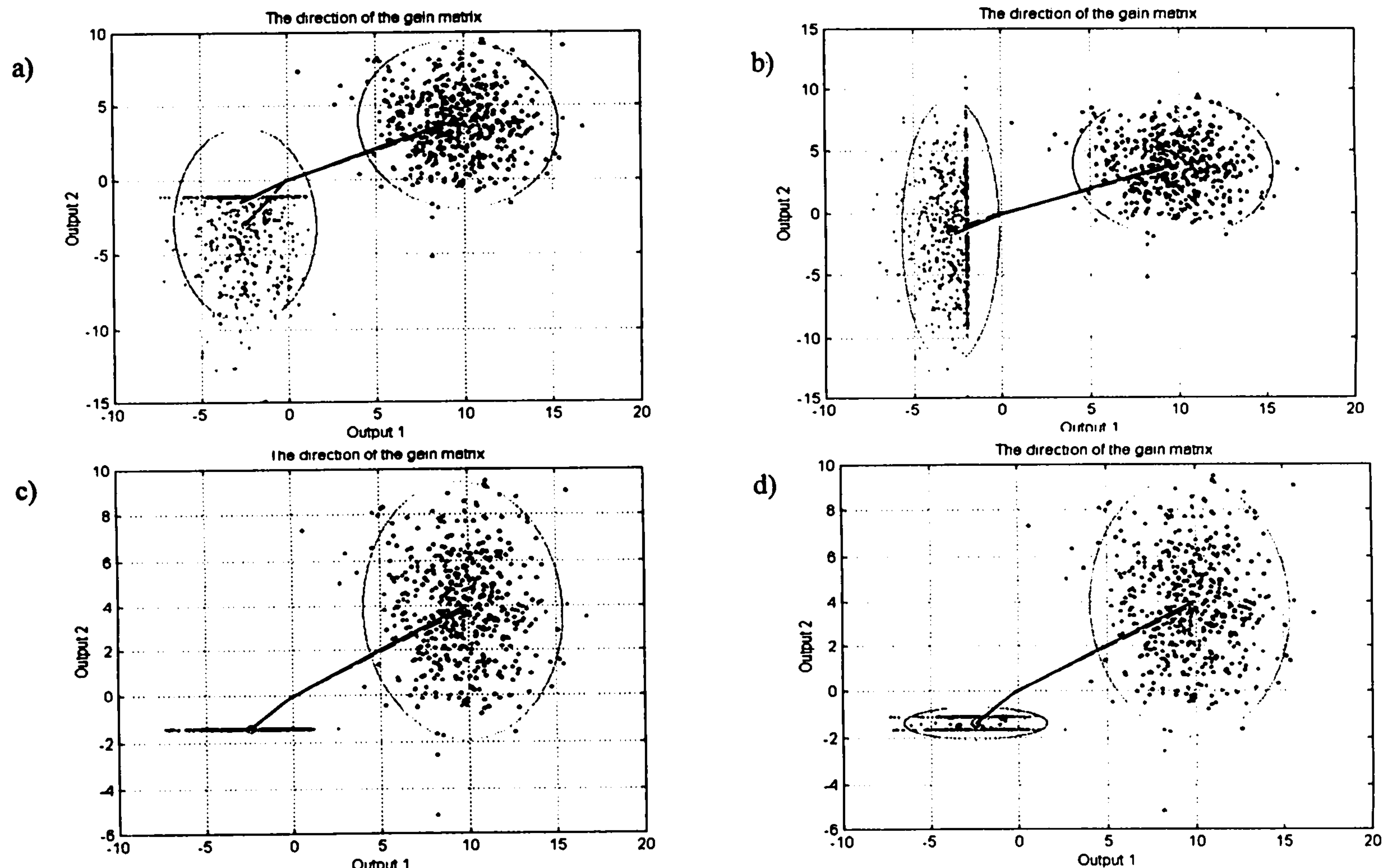
Figure 2.6:The gain estimates for the dynamic 2x2 system with no constraints (the P(USC) in this case is 46.6%)

a)

b)

c)

d)



—— True gain vector – – – Estimated gain vector .... 95% Confidence interval • Individual gain estimate Based on one Monte Carlo realization

Figure 2.7: The effect of different constraints on the direction of the estimated gain matrix. a) $g_{2,2} < -1$ b) $g_{1,2} < -2$ c) $g_{2,2} = -1.43$ d) $-1.12 > g_{2,2} > -1.68$ (see Appendix 6 for P(USC))

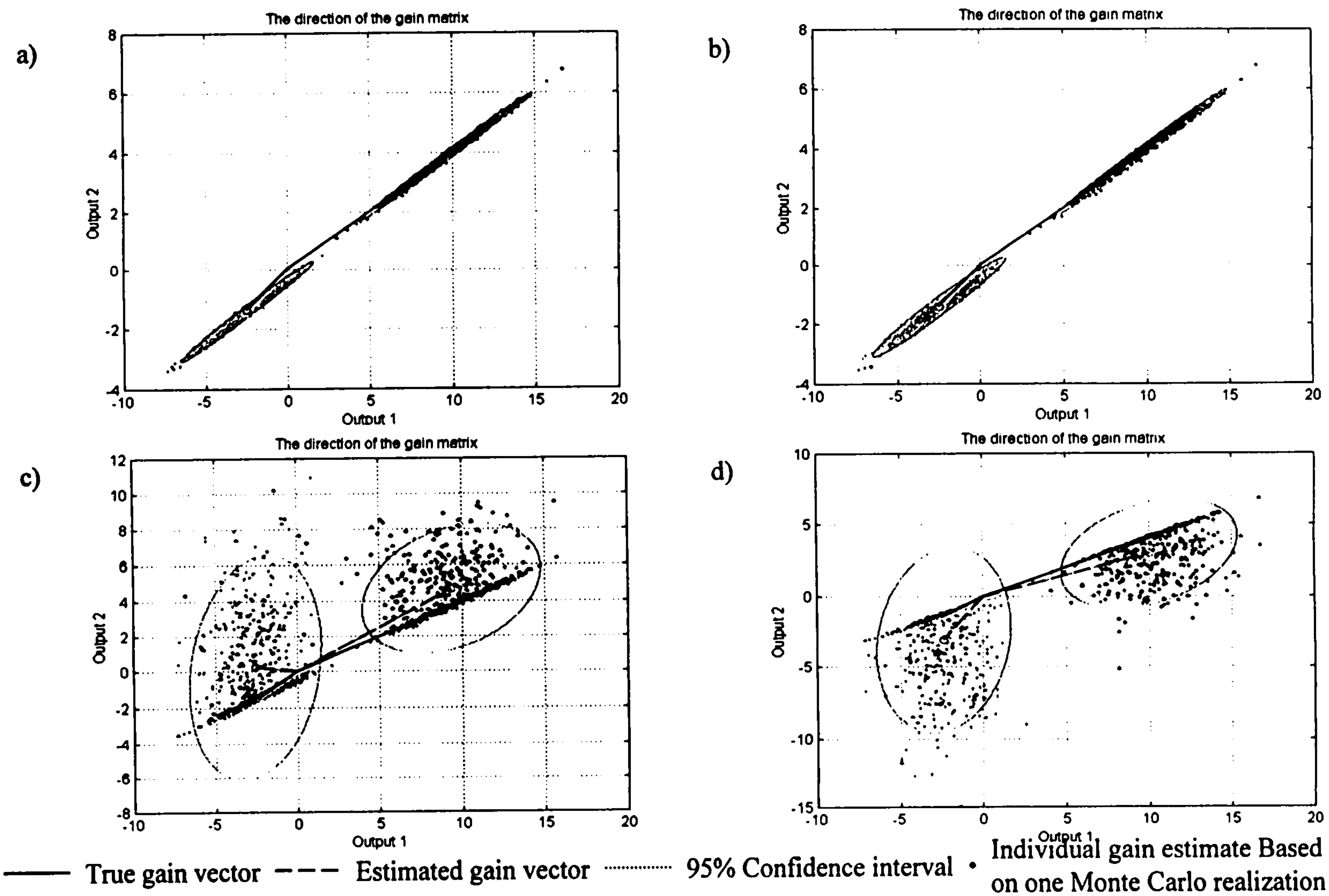


Figure 2.8: The effect of different angel constraints on the direction of the estimated gain matrix. a) $\alpha = -7.97°$ (P(USC) = 0) b) $-9.56° \le \alpha \le -6.37°$ P(USC) = 0) c) $-9.56° \le \alpha$ (P(USC) = 0.562) d) $\alpha \le -6.37°$ (P(USC) = 0.004) (see Appendix 6 for more detail)

estimated gain vectors, which defines the estimated gain directionality, differs from the true gain vectors. One should also note the distribution of the estimated gain elements and how constraints affect this distribution.

The effects of different prior knowledge have been tested on this system. Some prior knowledge involve the steady state portion of the estimated model, while other prior knowledge deal with the dynamic portion of the model. The different types of prior knowledge that were used during model estimation can be categorized into the following:

- Equality and inequality constraint on the gain (such as Figure 2.7 were equality and inequality constraints on two different gain elements are considered for a 2x2 system)
- Equality and inequality constraint on the ratio of gains
- Monotonicity constraints on the step responses
- Constraints on the angle between the gain vectors (such as Figure 2.8 were constraint on the angle for a 2x2 system is considered)

The results of the Monte Carlo simulations (*P(USC)* and the average Frobenius norm between the true gain matrix and the estimated gain matrix) for these types of prior knowledge are shown in Appendix 6.

2.4.4. 2x2 system with no dynamics

In the previous section, 2x2 systems with dynamics were considered, and in this section 2x2 systems with no dynamics are considered. This helps focus our attention on the gain matrix and its effect by eliminating the complexity resulting from the dynamic portion of the model. Incorporating the dynamics becomes a complex problem even for moderate size systems (5x5 may be considered moderate size), which in turn requires the estimation of a large number of parameters (i.e., for a 5x5 problem 5x5x20 = 500 parameters have to be estimated). In the systems with no dynamics, only a few parameters have to be estimated (i.e., for a 5x5 only 25 parameters have to be estimated). This allows a more comprehensive investigation of the effect of prior knowledge on the gain matrix, assuming that the effects are similar between the cases with and without dynamics. The increase in the complexity of the problem, for the cases with dynamics, results in more complicated optimization that requires more computing time.

In the matrix form, a 2x2 system with no dynamics is defined by $Y=G\times X+E$, where Y, X, and E are $2xn$ matrices. In this case, the elements of E are i.i.d. white noise, which are normally distributed with a variance of 1.

A few different 2x2 systems where simulated; their gain matrices are:

$$G_2 = \begin{pmatrix} 5 & 5 \\ 0.1 & 0 \end{pmatrix}, G_3 = \begin{pmatrix} 5 & 5 \\ 0.2 & 0.1 \end{pmatrix} \tag{2.18}$$

Both of these gain matrices have a condition number of 100. The resulting systems were perturbed with 3 levels of PRBS signals (0.025, 0.25, and 2.5), while the variance of the added white noise remained constant at one, to see the effectiveness of constraints at different signal-to-noise ratios. In each realization, 100 data points were collected, and a series of different models were estimated with different constraints (Appendix 6). Many different analyses were performed on the simulations results, of which only some are presented here. The most significant results was in estimation of *P(USC)*. To understand this result in greater depth, some properties regarding the distribution of the angle between the gain vectors were also observed but not reported here (such as mean, median, and standard deviation). These results provide valuable insight into the causes for an increase or decrease in *P(USC)*. The quality of the model for prediction is evaluated using $Mean\left(\left\|\hat{G}_{(k)} - G\right\|_F\right)$ (where $\hat{G}_{(k)}$ is the gain estimate based on the k^{th} realization of the Monte Carlo).

2.4.5. 5x5 system with no dynamics

In larger systems, similar issues arise as they do in smaller systems. For instance, in the 2x2 systems considered, one of the important issues was the angle between the two gain vectors. Similarly, in larger systems the angle between a gain vector and the hyper-plane defined by the other gain vectors plays an important role. In the case of the 2x2 system, one can look at both the trace and the determinant of $G\hat{G}^{-1}$, and if they are both positive, the CSC holds. This is a sufficient and necessary condition for a 2x2 system result in a SCS. However, in the case of a 5x5 system, the condition that both the determinant and the trace of $G\hat{G}^{-1}$ are positive is not a sufficient condition for SCS, rather it is a necessary condition. Considering larger systems will help in assessing if observing changes in determinant sign is a satisfactory means of evaluating CSC. Furthermore, 5x5 systems are perhaps more representative of the typical systems that are encountered in chemical processes. For these reasons the following 5x5 system was considered:

$$G_4 = \begin{pmatrix} 10 & -10 & 1 & .5 & .6 \\ 4 & -1.3 & -.2 & .75 & .6 \\ 1 & 10 & -1 & 1.5 & 1 \\ 0 & -5.5 & 0 & 0 & .25 \\ 1 & 6 & 3 & 10 & 6 \end{pmatrix} \tag{2.19}$$

The eigenvalues of this gain matrix in an increasing order are:

$$\lambda(G_4) = \begin{Bmatrix} -0.39 \\ 0.80 + 2.56i \\ 0.80 - 2.56i \\ 5.36 \\ 7.13 \end{Bmatrix}$$

Based on this, assuming an equal change in the eigenvalues, at first one eigenvalue will change sign and then two more eigenvalues will change sign. This is an important fact since the analytical methods used to estimate *P(USC)* are based on the assumption that an odd number of eigenvalues has changed sign.

The variance of the white noise added to the output was 1 for all the outputs. The inputs are perturbed with PRBS of magnitudes of ±0.25 or ±0.5, depending on the specific case. In each realization of the Monte Carlo simulation, 500 data points are collected. Finally, the model is estimated with a variety of constraints (Appendix 6). Similar analysis as in section 2.4.5 is performed on the simulations results.

2.4.6. Comparison of Theoretical and Simulation Results

A comparison of the theoretical and simulation results is essentially testing the assumptions used in the derivation of the theoretical expressions. The first assumption made is that when the elements of the gain matrix are normally distributed, the probability of an eigenvalue changing sign can be approximated as the probability of the determinant changing sign. This assumption can be tested via performing Monte Carlo simulations, where each realization is a different random matrix ρ, and the two sides of the bottom expression are compared:

$$P\left(\left(\lambda_i\left(G(G+\rho)^{-1}\right)\right) > 0, \forall i\right) \approx \begin{cases} \text{if } \det(G) < 0, P\left(\det(G+\rho) > 0\right) \\ \text{if } \det(G) > 0, P\left(\det(G+\rho) < 0\right) \end{cases} \tag{2.20}$$

Where ρ has the same dimensions as G and is a matrix of random numbers. The elements of ρ are normally distributed with mean zero and a specific variance. The results of the simulations, which can be seen in Figures 2.9-2.11, are based on 30,000 realization and 3 different systems (3x3, 4x4, and 5x5 system). These results show that when the variance of the added noise is less than 0.1 (which is approximately 10% of the value of the elements in the matrix *G*), (2.20) holds well. At higher levels of noise, the

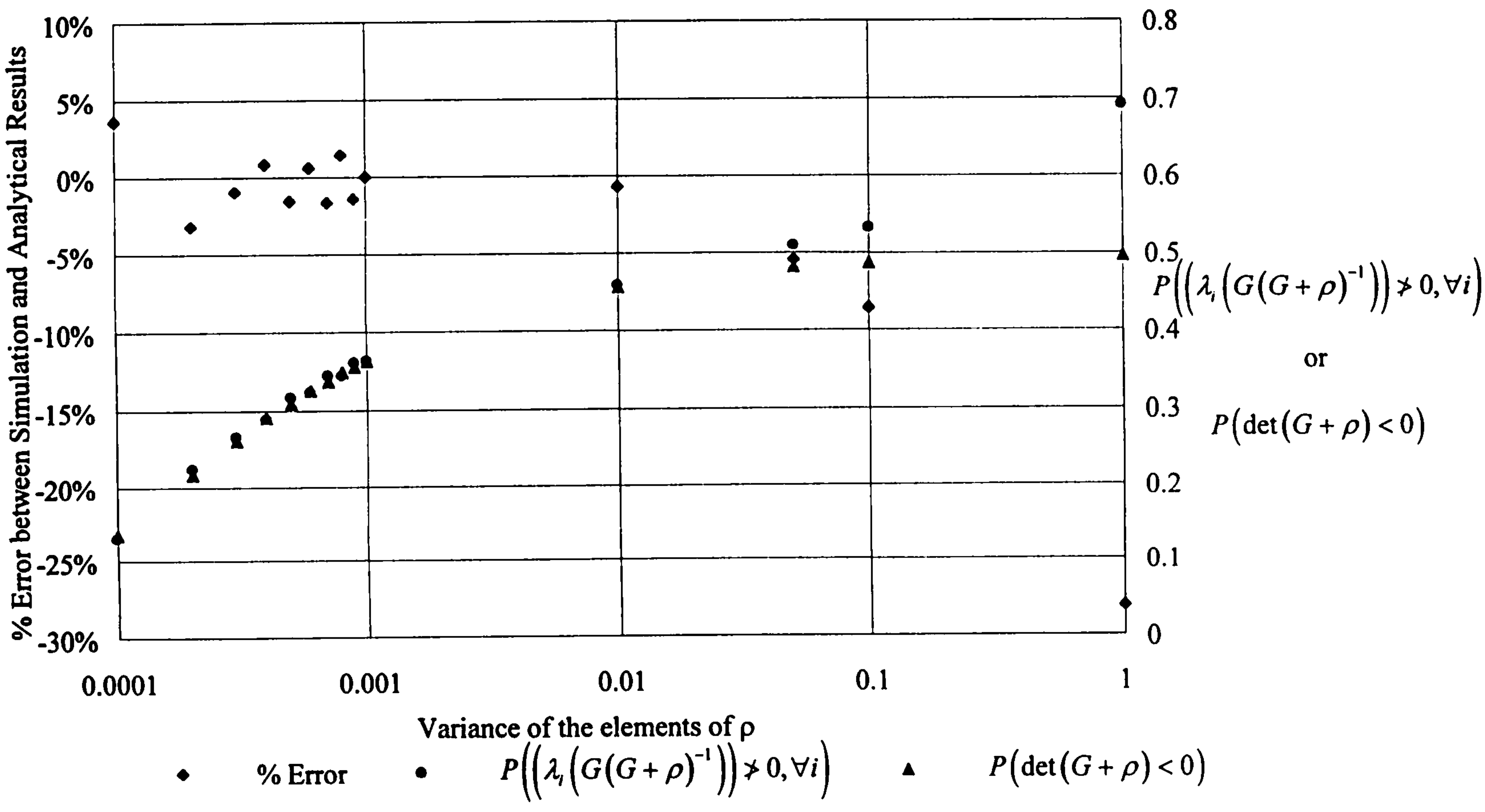


Figure 2.9: Analytical and Simulation Results for a 3x3 system. In this case $G = \begin{pmatrix} 1 & 0.1 & 0.1 \\ 0.1 & 1 & 0.1 \\ 1 & 1 & 0.2 \end{pmatrix}$

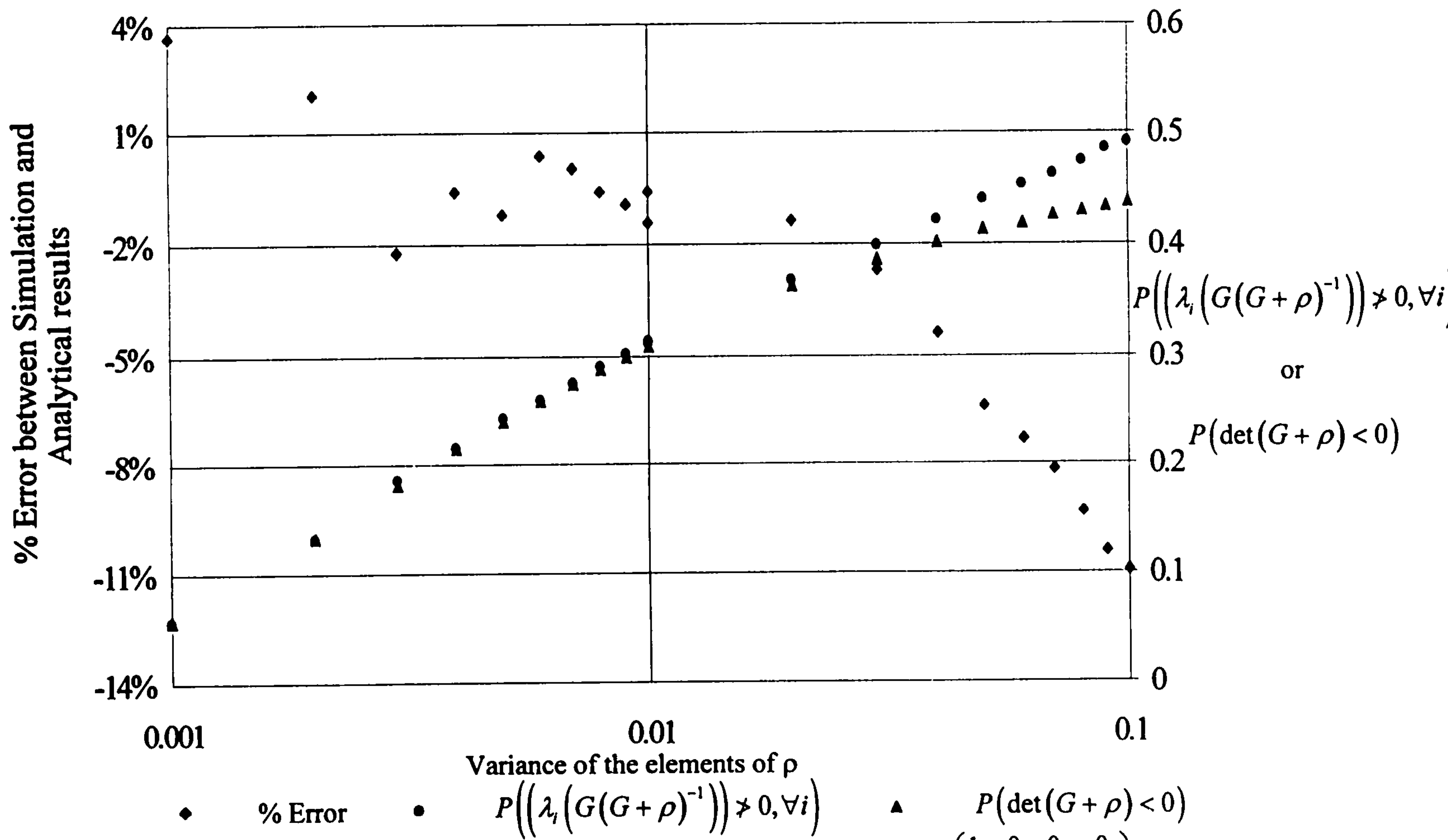


Figure 2.10: Analytical and Simulation Results for a 4x4 system. In this case $G = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0.1 \end{pmatrix}$

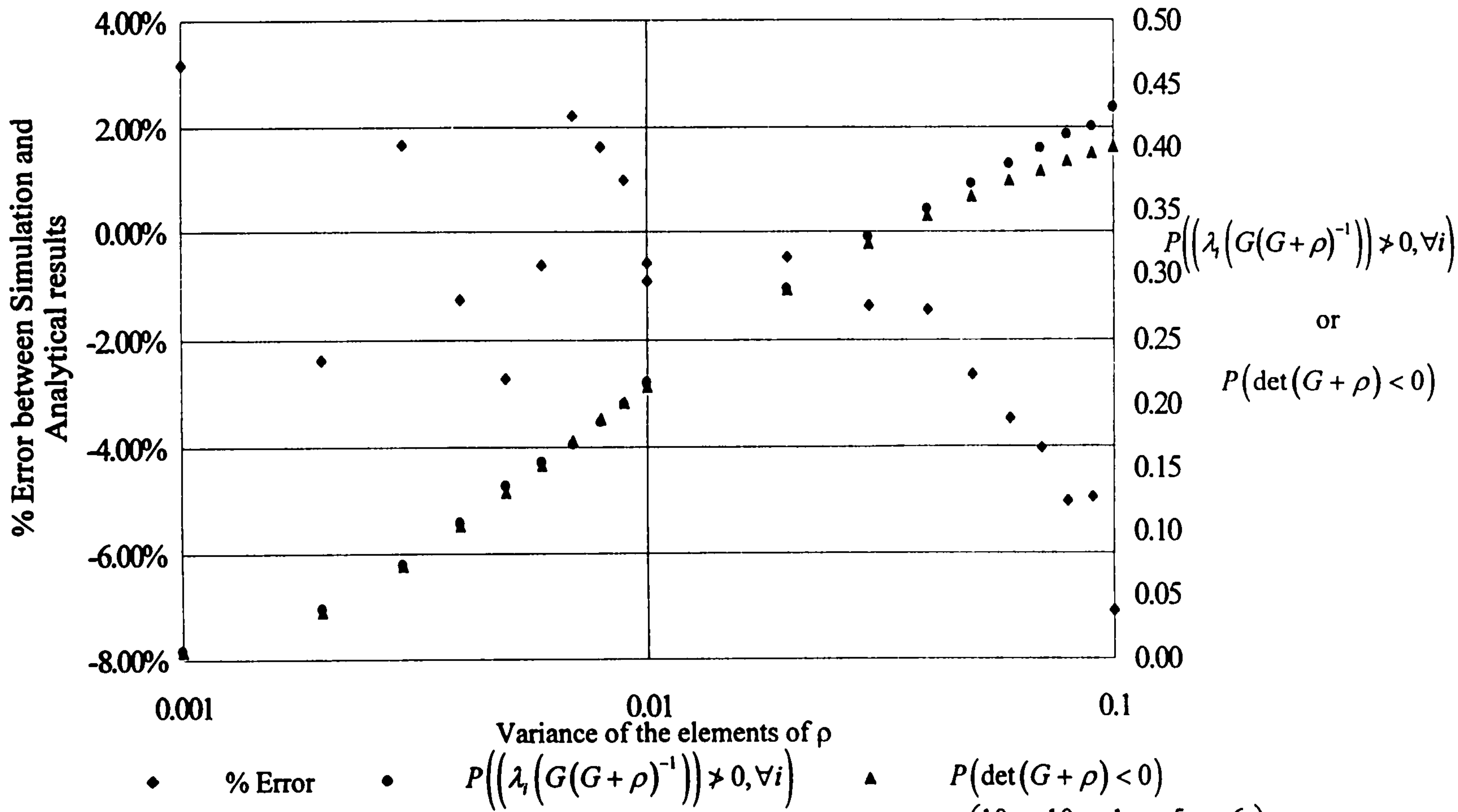


Figure 2.11: Analytical and Simulation Results for a 5x5 system. In this case $G = \begin{pmatrix} 10 & -10 & 1 & .5 & .6 \\ 4 & -1.3 & -.2 & .75 & .6 \\ 1 & 10 & -1 & 1.5 & 1 \\ 0 & -5.5 & 0 & 0 & .25 \\ 1 & 6 & 3 & 10 & 6 \end{pmatrix}$

R.H.S. (right hand side) of (2.20) differ from the L.H.S. (left hand side) of (2.20), since more than one eigenvalue is changing sign. In this thesis, the R.H.S. of (2.20) is estimated analytically (in Appendix 5) while the L.H.S., which is equivalent to the P(SCS), is estimated via Monte Carlo simulations. Therefore, these simulation results illustrate that at low levels of noise, the P(SCS) may be evaluated analytically by estimating the probability of the determinant changing sign in R.H.S. of (2.20).

When prior knowledge is utilized in estimation of the gain matrix, the estimated gain matrix may have non-normal distribution of its elements (Figure 2.3) (i.e., ρ in (2.20) will be non-normal). To see how, in these cases, the simulation results compare with theoretical results; some of the theoretical and numerical results are compared in Appendix 6 Table A.7. They show a very good agreement between the theoretical results and the Monte Carlo simulations for the cases considered.

It is important to note that simulation results are case specific. However, since they validate the theoretical results, at least in some regions that the assumptions hold, the theoretical results can be used to make generalized comments regarding the effect of prior knowledge. Figure 2.12 shows some of the cases where the theoretical result was compared with the simulation results. These results are for different systems with a variety of different constraints (for the detail see Appendix 6 Table A.7), and they compare well. Since the analytical results match the simulation results well, the analytical result can be used to evaluate the effect of different constraints at different signal-to-noise ratio for a variety of systems. For example, Figure 2.5 shows the effect of an inequality constraint as the bound in the inequality approaches the true value of the gain, and Figure 2.13 shows the effect of different constraints as the magnitude of the PRBS increases for the 5x5 system in (2.19).

2.5. Discussion of Results

In this section, the Monte Carlo simulation results based on the systems in sections 2.4.3, 2.4.4, and 2.4.5, and the analytical analysis of section 2.3 are used to describe the effect of different univariate and multivariate constraints. As was mentioned

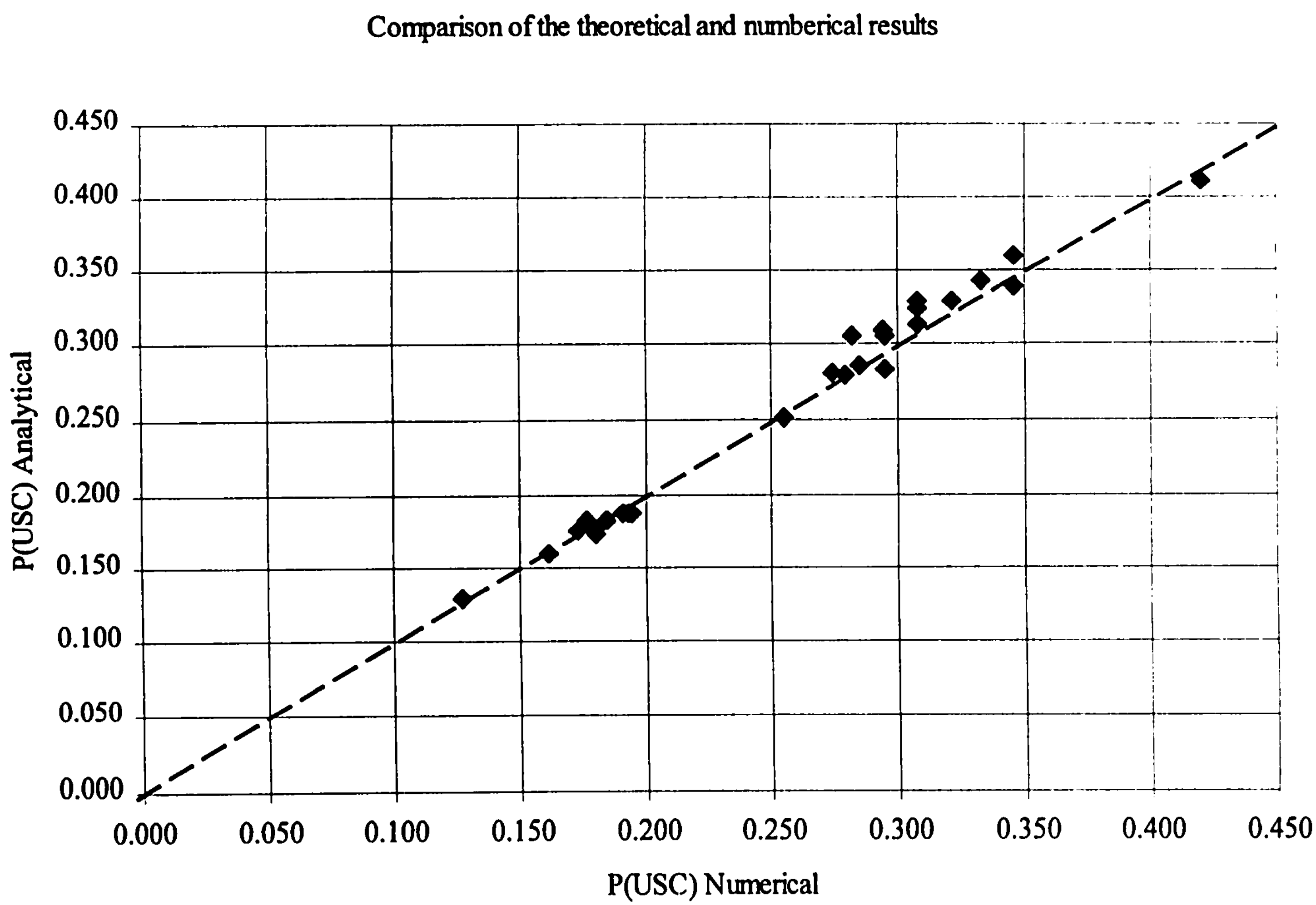


Figure 2.12: Comparison of some of the Monte Carlo simulations with the analytical estimates of P(USC) (for more detail see Appendix 6 Table 7). The dashed line represents a perfect match between the simulation results and the theoretical results.

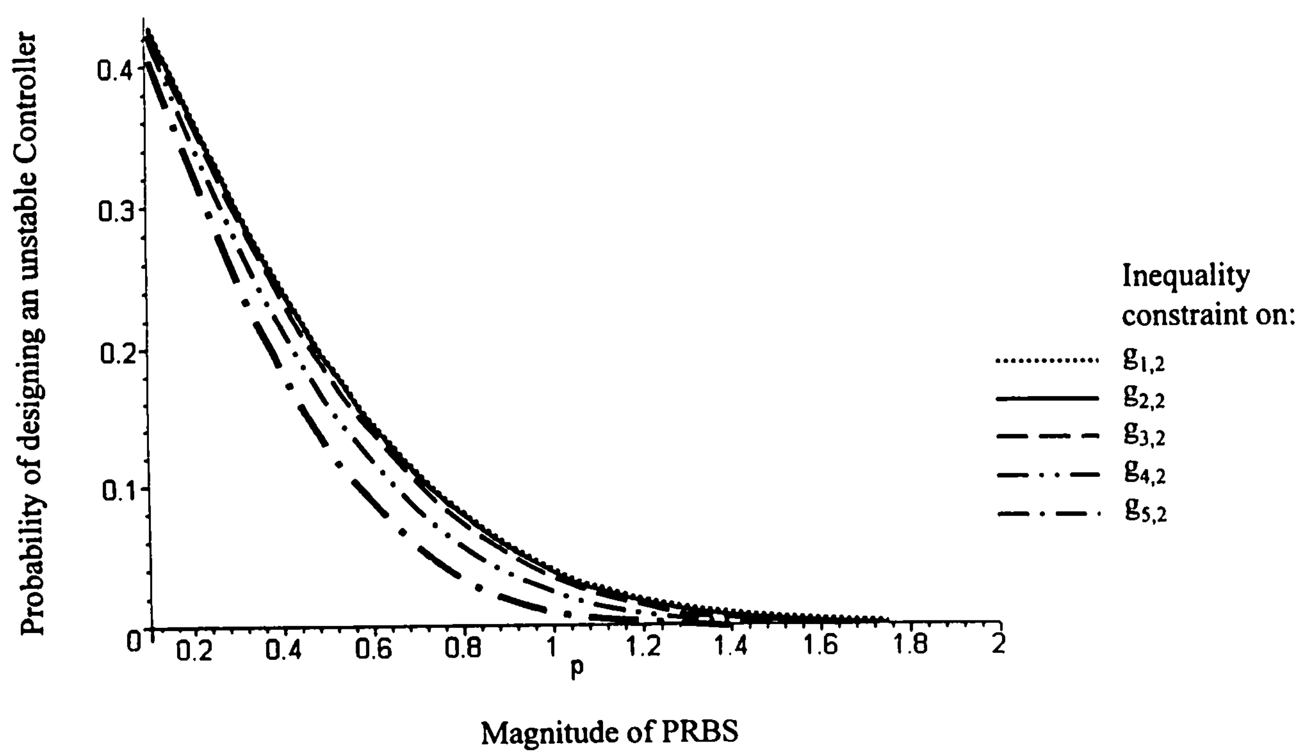


Figure 2.13: The effect of equality constraints on the model quality (based on theoretical results) at different levels of PRBS magnitude (note the curve for $g_{1,2}$ is almost under the $g_{2,2}$ curve)

earlier and concluded by other researchers (Tulleken 1993), addition of any correct prior knowledge will lower the variance of the model parameters. This can also be seen in the mean of the Frobenius norm of any of the simulation results listed in Appendix 6. The issue that will be discussed in the following section is the model quality in terms of CSC, not the variance of the model parameters.

At this point, most of the simulations have considered one constraint at a time (for the most part, the effect of combinations of constraints will not be studied in this thesis). There are two different classes of constraints: univariate and multivariate. A univariate constraint involves one input-output relationship, while a multivariate constraint involves multiple input-output relationships. When dealing with only steady state, a univariate constraint will deal with one gain parameter, while the multivariate constraint is based on multiple gain elements. This classification of constraints can be further divided into linear and non-linear constraints. Any of these constraints can be formulated in terms of equality (exact or inexact) or inequality constraints. It is important to note that this is one way of classifying constraints, with other classifications possible; however, this classification was found useful for this work.

Although a distinction is being made in this work between a multivariate and a univariate constraint, it is important to mention that a set of (or multiple) univariate constraints could have the same effect as a multivariate constraint. Sets of constraints are not directly considered in this work; however, the methodology developed in this chapter (and the next two chapters) can also be extended to consider the effect of sets of constraints.

2.5.1. Univariate constraints

There are many sources of univariate information. For instance, a distillation tower operator realizes that an increase in reflux flow will increase the distillate's light key mole fraction. Such knowledge is very common and may be formulated as an inequality constraint on the gain (i.e., $g_{reflux\text{-}Xp} > 0$). Based on his experience, he may have an idea about the range of response (i.e., $0.02 < g_{reflux\text{-}Xp} < 0.1$). Alternatively, he may even know the actual value of gain (i.e., $g_{reflux\text{-}Xp} = 0.05$). All of these prior knowledge constraints are univariate. Some of them, such as the sign of gain, are common, while others, such as exact value of gain, are rare.

2.5.1.1. Effect of equality constraint

There may be specific prior knowledge, such as the exact value of gain. For example, if an integral controller is between a process input and output, the steady state gain should be zero, and such a prior knowledge could be formulated as a linear univariate equality constraint. Linear univariate equality constraints are easy to implement, as there are closed form solutions for them (such as CLS). The closed form solution can be used to analytically estimate the probability of UCS. These results can be supported by numerical simulation results, as shown previously.

For the systems with no dynamics, the analytical solution suggests that equality constraint on the elements of the gain matrix will always help, hence lowering the probability of UCS. In some of the simulation results, the numerical trials resulted in slightly higher probability UCS for the equality constraint compared to no constraint, but this difference was very small (within the error tolerance). In the cases that the equality constraint fixes the small gain direction, which is crucial to the estimation of the correct angle sign, this constraint was very effective (see Appendix 6 Table A.2). Yet, if the value of the large gain direction is fixed, this has very little effect.

The results in Appendix 5 show that any type of equality constraint will lower the variance of the gain matrix, which in turn lowers the variance of the determinant, since it is assumed that the prior knowledge is correct. There will be no bias in either the gain estimates or determinant. Therefore, as a result of an equality constraint there will be no bias in determinant and the variance in determinant will decrease, which implies the *P(UCS)* will decrease for any system. This was also supported by the Monte Carlo simulations (shown in Appendix 6 Table A.2).

The above discussion is based on the changes in the *P(USC)*; however, one should note that based on individual data sets, it is impossible to know if a correct equality constraint will improve or degrade the model quality. Let us assume that the true gain vectors for a hypothetical 2x2 system is illustrated in Figure 2.14 (a). Also let us assume, that based on an identification experiment on this hypothetical system a model, such as the one shown in Figure 2.14 (b), was estimated. Assuming that there is a prior knowledge about one of the gain elements value (i.e., exact equality constraint), the gain matrices of Figure 2.14 (b) may be re-estimated to result in Figure 2.14 (c). Based on the previous discussion, any equality constraint will on average improve the model quality (i.e., lower *P(USC)*). However, in this case, based on the angle between the two gain vectors, the model in Figure 2.14 (c) is unstable while the model in Figure 2.14 (b) is stable (i.e., enforcing the prior knowledge results in a degradation of model quality). Surely, based on a different identification experiment, the same prior knowledge may not have an effect on the model quality (Figures 2.14 (d) and (e)) or it may improve the model quality in terms of CSC (Figures 2.14 (f) and (g)). One can only be certain that a true equality constraint will improve the model quality (in terms of CSC) if the true model is known a priori. Since the true model will rarely be known a priori to model identification, it would be impossible to know if an equality constraint will improve the model quality for a particular data set. Therefore, the condition that a system has to satisfy in order for an equality constraint to be useful on a data set is an uncheckable condition in real life. However, if the discussion were not based on a data set, rather many different data sets, an equality constraint will on average improve the model quality (i.e., *P(USC)* will decrease), even if the true model is not known (as shown in Appendix 6 Table A.2). On average, the model quality should improve by implementation of a correct equality constraint. It is suggested that such a prior knowledge should be utilized in practice, even if in a particular data set it could result in degradation of model quality.

2.5.1.2. Effect of inequality constraint

In the example mentioned in section 2.5.1, it was noted that the most common prior knowledge is the sign of the gain. In most cases, the operator or process engineer has an idea about the sign of the gain. Such prior knowledge is linear and can be implemented using a QP algorithm.

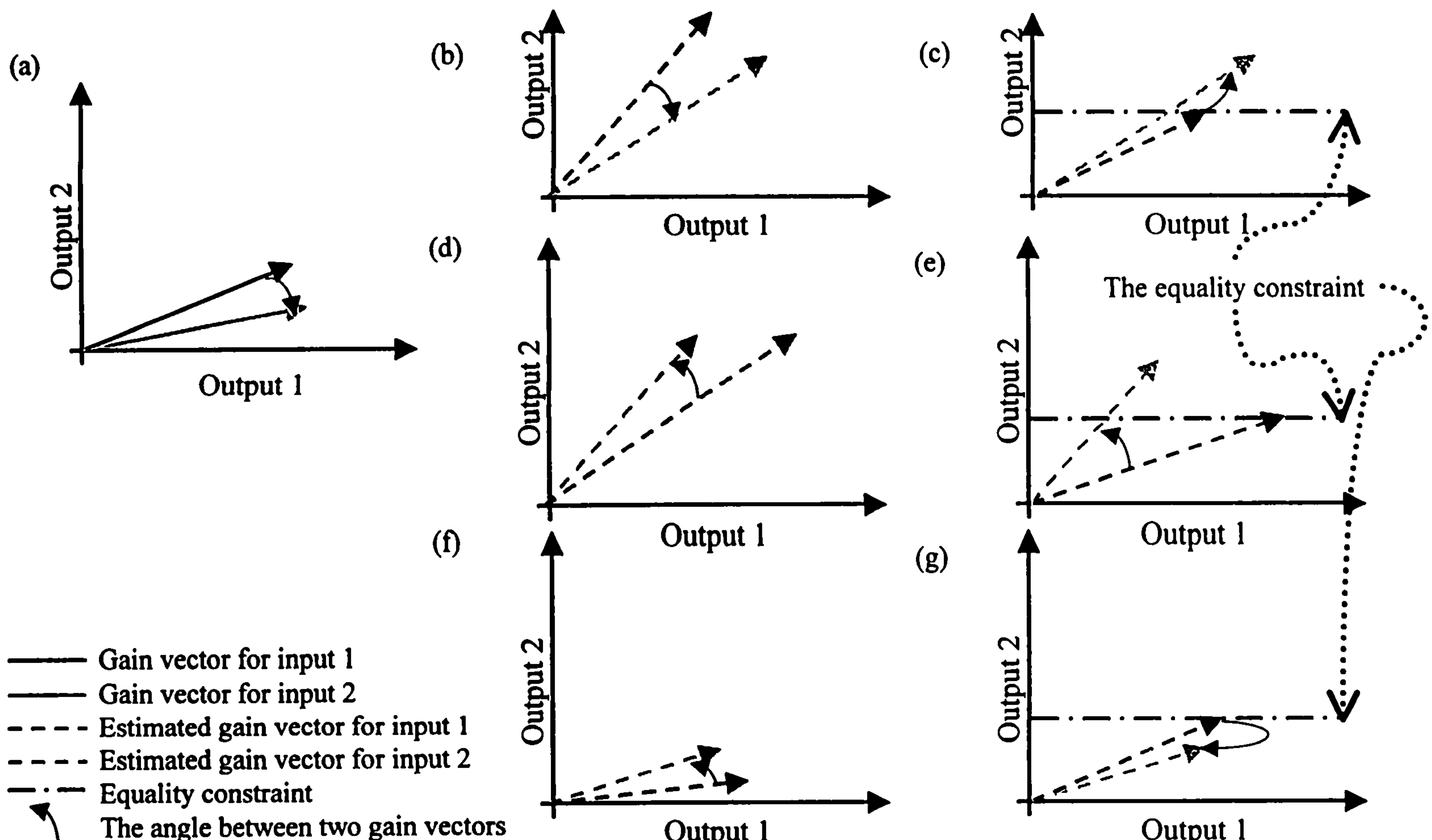


Figure 2.14: The uncheckable condition for an equality constraint shown for a hypothetical 2x2 system. (a) The true gain vectors (b) The first realization of the system resulting in a SCS (c) Implementation of an equality constraint on the first realization resulting in an USC, hence degrading the model quality (d) The second realization of the system resulting in a UCS (e) Implementation of an equality constraint on the second realization resulting in a UCS (f) The third realization of the system resulting in a UCS (g) Enforcing an equality constraint on the third realization resulting in a SCS

After a series of assumptions, an analytical expression for estimating the probability of UCS was devised. This expression is not very accurate in predicting actual values of *P(USC)* (see Appendix 6 Table A.7 for the comparison). It should be used as a means of determining trends rather than exact values, as mentioned in section 2.4.6. Both the analytical and simulation results suggest that an inequality constraint can either improve or degrade the model quality (see Appendix 6 Table A.3). Both dynamic and non-dynamic simulations confirm this. If the inequality constraint is such that it would force the angle of the model (defined as the angle between any gain vector and the hyper-plane defined by all the other gain vectors) to be smaller than that of the true model, the model quality degrades (for example see Figure 2.7 (b)). On the other hand, if it forces the angle of the model to increase over that of the true model, the quality of the model improves. For instance, in the simulation results shown in Figure 2.7 (a) the inequality constraint forces the mean of $\hat{g}_{2,2}$ to be lower than the actual $g_{2,2}$. In turn, this results in an increase of the angle between the two gain directions, which lowers the *P(USC)* for this system.

The uncheckable condition that was mentioned for the equality constraint also applies to the inequality constraint. Consequently, one should note that based on individual data sets, it is impossible to know if a correct inequality constraint will improve or degrade the model quality (see Figure 2.15). The only certainty is that an inequality constraint will improve or degrade the model quality if the true model is known a priori (which would never happen, hence the uncheckable condition). Therefore, the condition that a system has to satisfy in order for an inequality constraint to be useful on a data set is an uncheckable condition in real life.

If the discussion were not based on one data set, rather on many different data sets, then a particular inequality constraint on average may improve or degrade the model quality (i.e., *P(USC)* will increase or decrease as shown in Appendix 6 Table A.2). This can be explained visually using Figure 2.16, where it is assumed that there is no uncertainty associated with gain vector for input 1, and the uncertainty in gain vector for input 2 is illustrated by a circle. For this 2x2 system the P(USC) will be equal to the

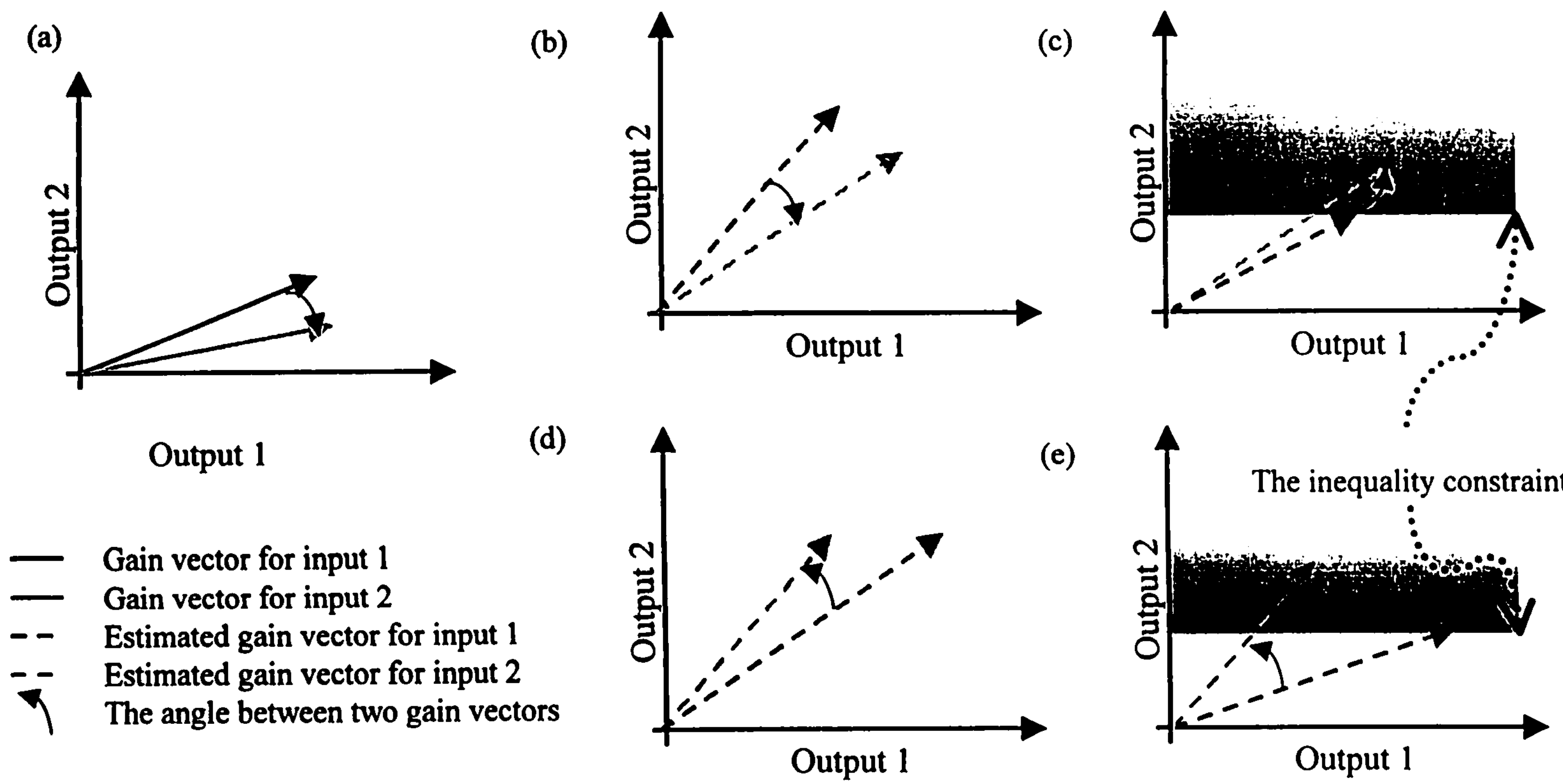


Figure 2.15: The uncheckable condition for an inequality constraint shown for a simple 2x2 system. (a) The true gain vectors (b) The first realization of the system resulting in a SCS (c) Implementation of an inequality constraint on the first realization resulting in an USC, hence degrading the model quality (d) The second realization of the system resulting in a UCS (e) Implementation of an inequality constraint on the second realization resulting in a UCS

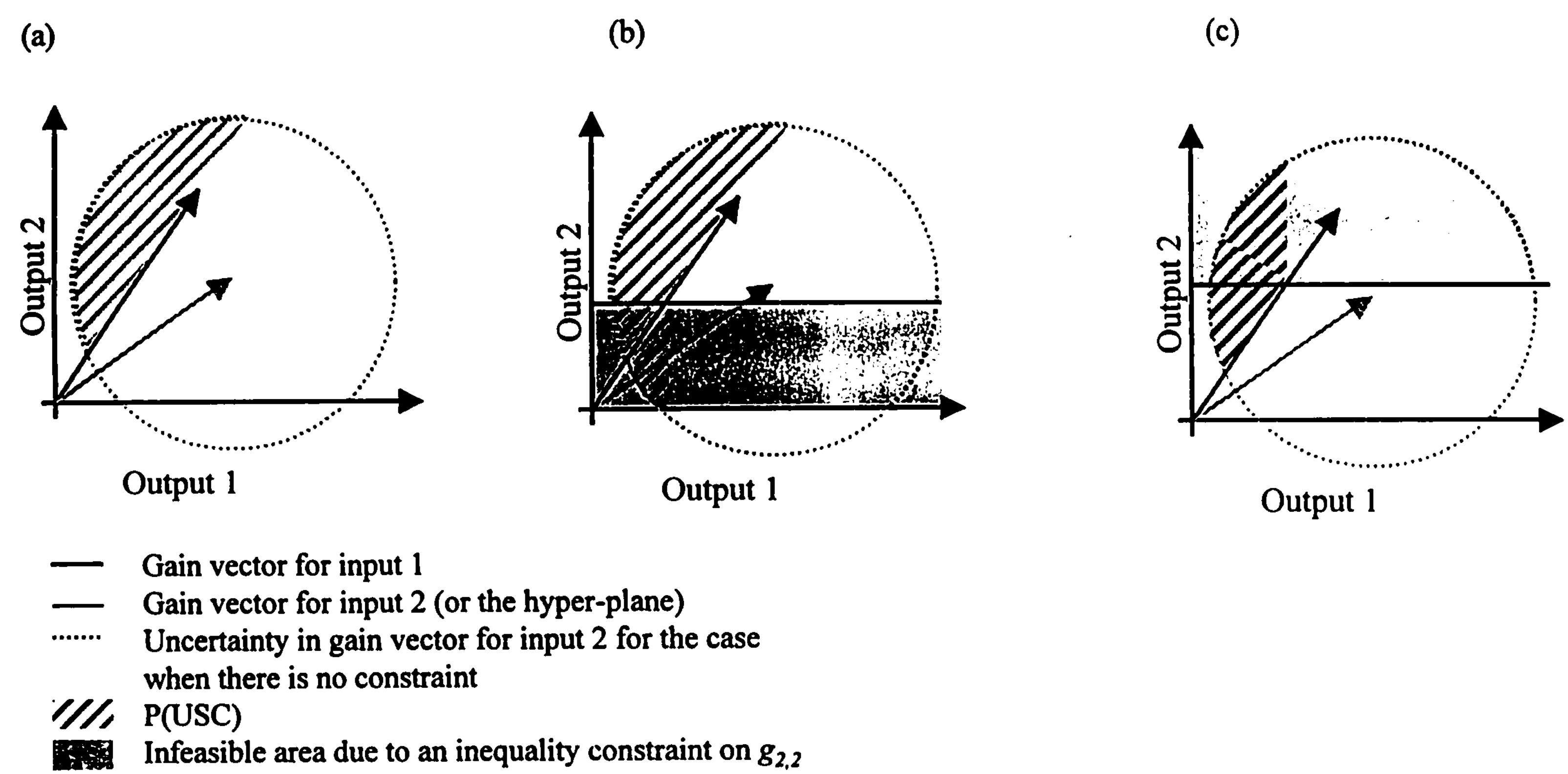


Figure 2.16: The uncheckable condition for an inequality constraint shown for a simple 2x2 system assuming no uncertainty in the gain vector for input 1 and a circle representing the uncertainty in the gain vector for input 2. (a) The initial gain matrix estimated with no constraints (b) The gain matrix estimated with lower bound constraint on $g_{2,2}$ (c) The gain matrix estimated with an upper bound constraint on $g_{2,2}$

probability of the angle between the gain direction being estimated with the incorrect sign. The shaded area shows this probability. As can be seen depending on the direction of the inequality constraint and the directionality of the true gain matrix, the P(USC) may increase (Figure 2.16 (b)) or decrease (Figure 2.16 (c)). Since in real life the true gain directionality will be rarely known a priori, it would be impossible to know if an inequality constraint will improve or degrade model quality on average (hence this is an uncheckable condition). This is the kernel difference between the effect of an equality and inequality constraint on the model quality. Therefore, in practice, an inequality constraint should not be utilized, unless there is very good prior knowledge about the directionality of the true gain matrix.

Although not explicitly mentioned, the effect of simultaneous upper and lower bound constraints on the gain elements was also considered. The effect of having a lower and upper bound constraint, when the distance between the lower and upper bound (δ) is significantly smaller than the standard deviation of the gain element involved ($\sigma_{g_{i,j}}$), was similar to an equality style constraint. However, when δ is significantly larger than the standard deviation of the gain element involved, this type of constraint has no effect. Certainly, there is a transition range ($\frac{\sigma_{g_{i,j}}}{3} < \delta < 3\sigma_{g_{i,j}}$) that the constraint will improve the quality of the model, but not as noticeably as an equality constraint would. This can be seen for a particular situation in Figure 2.17, where the *P(USC)* is evaluated for different distances of upper and lower bounds from the true value. Inexact linear equality constraints, otherwise known as stochastic linear constraints ($g_{i,j} = c + \upsilon, \upsilon \sim N\left(0, \sigma_\upsilon^2\right)$) (Judge et al. 1980), also behave in a similar way to an upper and lower bound constraint. When the uncertainty in the constraint is large (i.e., $\sigma_\upsilon \gg \sigma_{g_{i,j}}$), this type of constraint has no noticeable effect; however, when the uncertainty in the constraint is small (i.e., $\sigma_\upsilon \ll \sigma_{g_{i,j}}$), it behaves similarly to an equality constraint and may be effective in improving model quality.

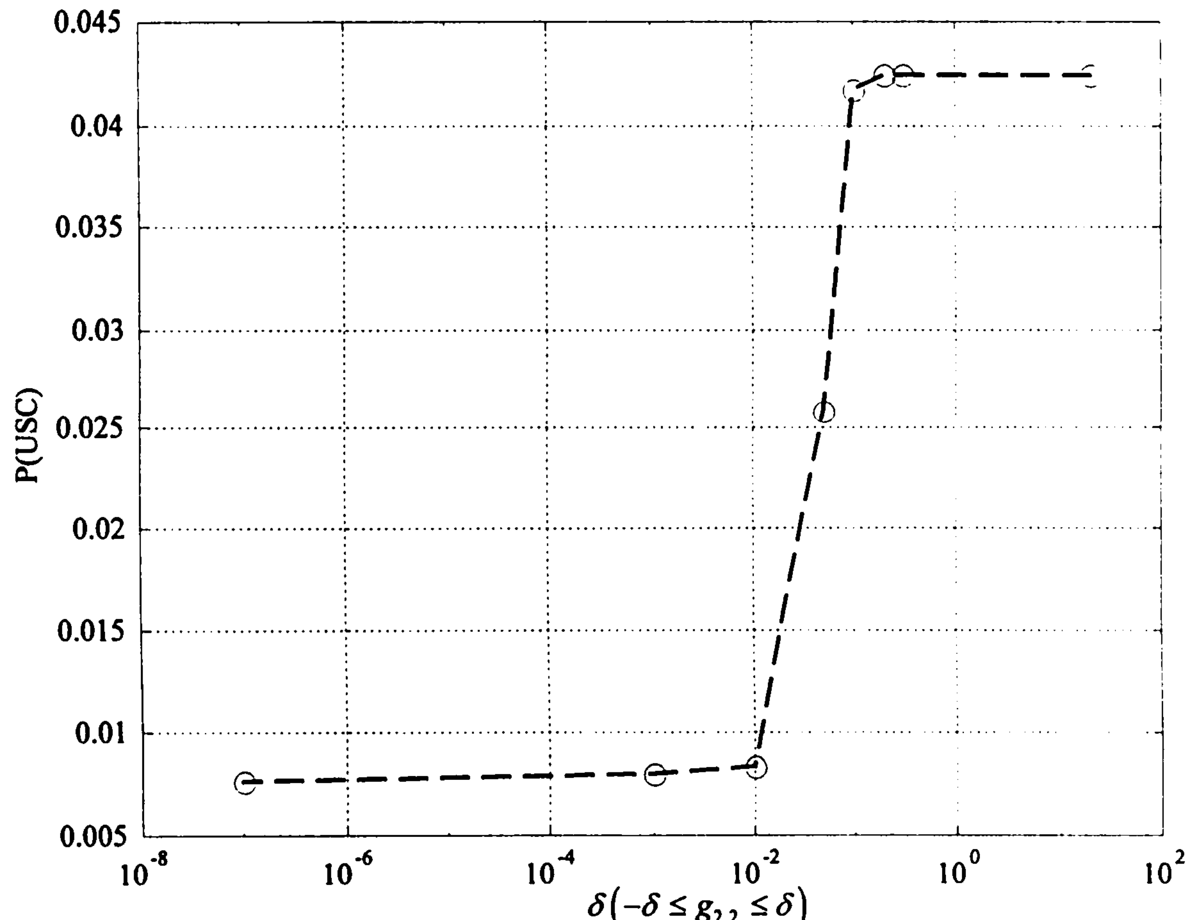


Figure 2.17: The effect of a lower and upper bound inequality constraint on $g_{2,2}$ of G_2 (δ is the distance of the lower and upper bound to the true value of the gain). Note that P(USC) for the case that the constraint is $g_{2,2}=0$ (i.e. $\delta=0$) is 0.0077 (±0.0018). When there is no constraint the P(USC) is .0424 (±0.0042) and the standard deviation of $g_{2,2}$ is 0.04. This plot was obtained using 9308 realization for a Monte Carlo simulation, with each realization having 100 observations, the noise added was i.i.d. N(0, I) and the magnitude of the PRBS was +/-2.5.

2.5.1.3. Monotonicity

A monotonicity constraint on the step response parameters of the model will force the step response parameters to be sequentially increasing or decreasing. This can be implemented by constraints on the FIR parameters of the model, and the resulting least square problem is the Non-Negative Least-Square (NNLS) problem (Lawson and Hanson, 1995). Based on the dynamic simulations for the 2x2 systems, it was seen that monotonicity constraints could produce models with higher or lower probability of UCS (see Appendix 6 Table A.4). These simulation results cannot be confirmed theoretically, since the added noise to the process output is a colored noise. In special cases, when the noise added to the process output is white noise, it would be possible to derive an analytical expression for the *P(USC)* (this is not shown here to save space). The effect of a monotonicity constraint on *P(USC)* is similar to the effect of inequality constraint on the gain. An inequality constraint on the gain directly affects the controller design, while monotonicity constraint affects the gain, in a similar fashion to an inequality constraint, which in turn affects the probability of UCS. As mentioned in the last section, in practice an inequality constraint on the gain will only improve model quality if an uncheckable condition is satisfied. It is evident that the monotonicity constraint has the same weakness.

A second class of constraints similar to monotonicity constraint is a windowing style constraint. In such a constraint the step response is bounded within a window by two different transfer functions. Such a situation may arise if the practitioner has knowledge about both the dynamic and the static portion of the model. His or her knowledge can then be incorporated into two transfer functions and used as constraints in the model identification phase. It is also possible that the practitioner has prior knowledge only about a lower or an upper bound. In such cases, the constraint will have a similar effect on CSC as a monotonicity constraint, since it will act similar to an equality constraint on the gain element. For the situation when there is a prior knowledge about both the upper and lower bound (on the step response model), the result is similar to the case when there is an upper and lower bound on an individual gain. Some simulation results with windowing constraints for system (2.15) are shown in Appendix 6 Table A.4. They illustrate that while some constraints improve the model quality in terms of CSC, other constraints degrade the model quality. In particular, constraints on $g_{2,2}(q^{-1})$ appeared to have severe effects on the model quality. It is interesting to note that constraints on $g_{2,2}(q^{-1})$ had more effect on P(USC) than constraint on $g_{2,2}$ (from Appendix 6 Table A.3). This implies that although windowing constraints have a similar effect on P(USC) as equality constraint on the gain, they appear to have a more severe influence on P(USC). This can be explained by the fact that an upper or lower constraint on the entire transfer function is a more restrictive constraint than a constraint on the gain alone.

2.5.2. Multivariate constraints

A multivariate constraint is defined as a constraint that would involve multiple input-output relationships. Preferably such a constraint would involve all the gain elements and provide information about the system's directionality. Multivariate information is perhaps harder to come by, compared to univariate prior information. It also tends to be more difficult to handle for the optimizer. It will be shown that multivariate constraints are the types of constraints that would have the most effect on the CSC.

2.5.2.1. RGA

A very particular multivariate prior knowledge is the knowledge about the Relative Gain Array (RGA) (Bristol 1966). The RGA is a measure of interaction and is utilized often as a guide for loop pairing in control design. For small systems (2x2), the sign of the RGA may be known a priori; the actual value is rarely known a priori. For the 2x2 systems, the RGA is calculated by (2.21). It can be seen, for 2x2 systems, that an inequality RGA constraint results in a non-convex region that an optimizer could not handle easily (Figure 2.18).

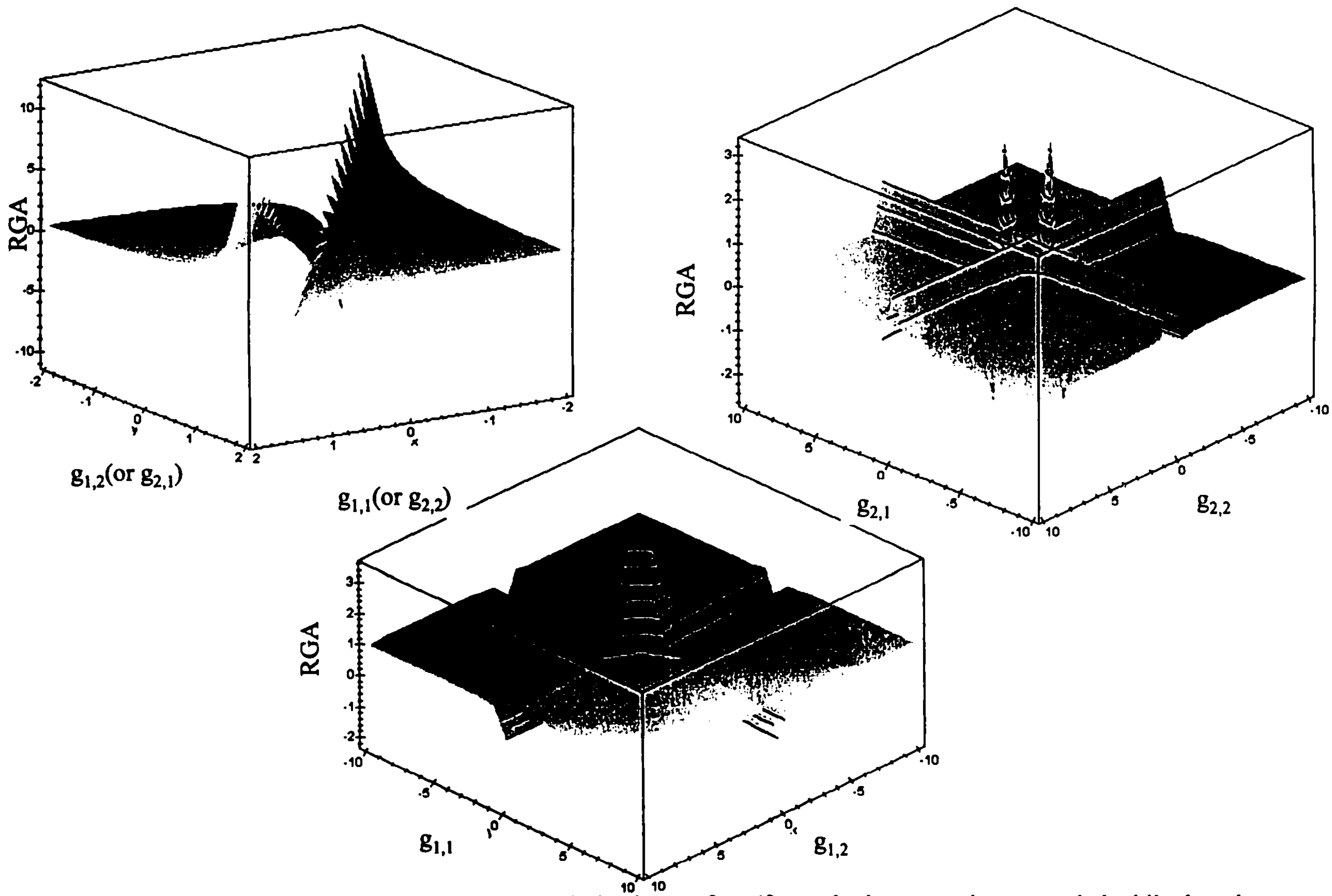


Figure 2.18: The effect of RGA constraint on the optimization surface (for each plot two gains are varied while the other two gains are set to 1)

$$RGA_{2x2} = \frac{1}{1 - \frac{g_{2,1}g_{1,2}}{g_{1,1}g_{2,2}}} \tag{2.21}$$

In order to solve this optimization problem, a GA type optimizer is required. Otherwise, to eliminate this problem, the sign of all the gain elements has to be known a priori (Koung 1991, also came to a similar conclusion). For larger systems (larger than 2x2), the RGA matrix is given by:

$$\Lambda = G \otimes \left(G^{-1}\right)^T \tag{2.22}$$

Where ⊗ is element by element product. For larger systems (larger than 2x2), it is rather rare to know the sign of the RGA matrix elements. Even if such knowledge were available, the application of such a constraint would require multiple non-linear constraints (since there would be one non-linear constraint for each element of the Λ), and the resulting optimization problem would be even more non-convex than before. Other extensions of RGA are the Relative Dynamic Gain Array and the Block Relative Gain. The Relative Dynamic Gain Array (RDGA) accounts for the dynamics of the process as well (Skogestad and Postlethwaite 1996) ($\Lambda\left(e^{i\omega}\right) = G\left(e^{i\omega}\right) \otimes \left(G\left(e^{i\omega}\right)^{-1}\right)^T$). The Block Relative Gain Array (BRG) (Manousiouthakis et al. 1986), which was designed as a guide for determining block-decentralized control structure, is a special summation of the RGA. Prior knowledge in either one of those metrics is both rare and hard to implement.

For the reasons mentioned, an RGA or an RGA-style constraint will not be considered here.

2.5.2.2. *Constraint on Angle*

A constraint on the angle will be non-linear and discontinuous (as shown by non-linear and discontinuous equation (2.9)). A discontinuous constraint is difficult to implement. One solution to this problem is to prevent the optimizer from searching in the region that the constraint is discontinuous. One can accomplish this by simulating a system with high signal-to-noise ratio. In this case, however, since the quality of the data is good, there would be no need for the constraint. Alternatively, a series of assumptions on the magnitude of the gains would also limit the search space (this is similar to RGA_{2x2} case).

A constraint on the angle can take two forms: equality or inequality. In the case of equality constraint, the result is similar to linear equality constraints on the gain elements. In such a case, the constraint will always reduce P(UCS). It would be especially useful if the angle between the two gain vectors is small. In the case of inequality, if the constraint resulted in lowering the collinearity in the model, it improves model quality (Figure 2.8 (d)); however, if it increased the collinearity, such a constraint degrades model quality (Figure 2.8 (c)) (for examples of this see Appendix 6 Table A.5).

2.5.2.3. *Multivariate Linear Constraints*

Some linear constraints have the same effect as constraints on angles. For example, in (2.18) if a constraint is placed on the difference between the $g_{1,2}$ and $g_{2,2}$:

$$g_{1,2} - g_{2,2} > 0 \qquad (2.23)$$

This is similar (although not exactly the same) to stating that the angle between the two gain vectors is positive. Similar to section 2.5.2.2, such a constraint would be very useful since it lowers collinearity. Alternatively, a constraint regarding the addition of the two gain elements, in this particular case, would have little effect. It is difficult to evaluate the effect of multivariate linear constraints that have more than two gain elements involved. In general terms, they are useful in two situations: if they limit the angle between the gain vectors to a particular sign, and/or if they fix a low gain direction. In addition, similar to an equality univariate constraint, a multivariate linear equality constraints will always improve model quality (i.e., lower *P(USC)*) (as shown in Appendix 5).

2.5.2.4. Other Multivariate Constraints

The practitioner may have knowledge of many other possible prior constraints. Some of them are the ratio of two gains, the eigenvalue of the gain matrix, and the determinant of gain matrix. Prior knowledge about the ratio of gain is common in chemical industry. Such a constraint would be implemented as a linear equality or inequality constraint.

It follows from the discussion of importance of the estimated gain matrix eigenvalues that perhaps a constraint on an eigenvalue would be helpful. However, such a constraint would have similar problems to an RGA type constraint (see 2.5.2.1). Furthermore, it would be rather rare for a practitioner to have prior knowledge about the eigenvalues of a process.

Prior knowledge about the determinant of the gain matrix would also be rare. Yet, it is possible that such a prior knowledge will be gained based on a controller performance. This will be discussed in detail in future chapters.

2.6. Conclusions

In this chapter, the effect of correct prior knowledge on MIMO models used for control has been presented. The effects of a variety of constraints for different systems have been considered using Monte Carlo simulations and analytical results. The analytical results have been based on propagation of the gain matrix uncertainty to the angle between the gain vectors or the determinant of the gain matrix. The Monte Carlo simulations have been utilized to check the analytical results and to provide insight into cases where no analytical solution to the problem could be found.

It was determined that the most effective prior knowledge was multivariate prior knowledge. Although a few univariate constraints were helpful, some multivariate constraints were very effective in lowering the *P(USC)*. Such constraints would involve multiple gain elements that were in some way associated with the low-gain direction of the process. In essence, these known multivariate prior constraints were providing crucial information about a particular direction that the estimated model had very little information.

For a univariate constraint to be effective, a prior knowledge about the directionality of the multivariate problem is required. Since, in most cases, such a prior knowledge is not available, this results in an uncheckable condition that has to be satisfied for a constraint to be effective. Therefore, in practice it would be unlikely to be able to predict the effect of a univariate constraint on a single data set. On average (over many data sets), any equality constraint could improve the model quality; however, an inequality constraint on average could improve or degrade the model quality depending on the directionality of the true model, which for the most part is uncheckable. This implies that in practice one should not implement a univariate inequality prior knowledge (even if it is known to be true), unless prior knowledge about the gain directionality exists.

Chapter 3
Model Sensitivity to Prior Knowledge

3.1. Introduction

A methodology for local perturbation (sensitivity) analysis of solution behavior with respect to problem changes is a requirement of any scientific discipline. The sensitivity analysis is an integral part of any solution. The quality of a solution cannot be understood without such information. These techniques are used in a variety of fields to evaluate the sensitivity of the solution (Fiacco 1983). For example, in solving systems of linear equations the condition number of the coefficient matrix is a measure of the sensitivity of the solution to changes in the coefficient matrix. In least square problems the condition number of the covariance matrix is a measure of the sensitivity of the solution. In model identification, analogous techniques such as covariance of the model parameter have been used to assess the sensitivity of models.

In this chapter, the effect of error in prior knowledge on the model quality is considered. The objective is twofold: evaluate the sensitivity of controller stability to error in prior knowledge, and classify which errors have the most effect on this stability. Similar to the previous chapter, these issues are considered for MIMO (multi-input multi-output) systems that are not singular in nature. It is assumed that the system is square (i.e., equal number of input and outputs) and bounds on the process inputs and outputs are not considered. However, contrary to the previous chapter where it was assumed that the prior knowledge is correct, no such assumption is made in this chapter.

The most significant contribution of this chapter is devising checkable metrics to be used in evaluating the effect of error in prior knowledge. The practitioners can then evaluate the sensitivity of a constraint before implementing the controller utilizing such metrics. These metrics evaluate the sensitivity of a univariate constraint on a multivariate problem.

In order to illustrate the effect of incorrect prior knowledge Monte Carlo simulation on a 5x5 system with no dynamics were performed. These simulation results were then compared to the theoretical results. The theoretical results, which are checkable metrics in practice, can then be utilized by a practitioner to perform sensitivity analysis.

As was mentioned in the previous chapter there is no literature on the effect of prior knowledge on MIMO model identification for controller design. Consequently, there is no literature that deals with the effect of incorrect prior knowledge in such a situation. Most of the literature on the effect of incorrect prior knowledge was found in the econometrics field (Judge et al. 1980). In econometrics, extensive research has been performed on the effect of error in a linear constraint in the least square problem. This research has lead to understanding of the parameter distribution in view of error in the prior knowledge. It has even given rise to other parameter estimators such as the pretest estimator, Stein-rule estimator, Stein positive-rule estimator and other Stein-like pretest estimators.

Similar to the previous chapter, in this chapter the emphasis of the model quality is on the controller stability criteria (CSC). The effect of error in the prior knowledge on prediction is similar to the issues covered in the econometric field and they are not discussed here. Although it is shown that contrary to model prediction issues where the directionality of the error in the prior knowledge is not important, in the case of the CSC the directionality of the error in prior knowledge is crucial. Certainly, the issue of the magnitude of the error in prior knowledge is important to both prediction and CSC.

3.2. Example of Problem

Suppose that a process is linear and its true model is given by (3.1). This plant is poorly-conditioned (condition number of the gain matrix is 83); however, it is not singular.

$$G(s) = \frac{e^{-1s}}{10s+1}\begin{pmatrix} 5 & 0.1 \\ -10 & 0.1 \end{pmatrix} \tag{3.1}$$

Such a plant transfer function is constructed of a scalar dynamic model multiplied by a constant matrix (the gain matrix). There are many examples of such transfer function matrices in chemical engineering, including the simplified distillation column studied by Skogestad et al. (1988). Consider the following three estimates of this plant, which have resulted from an identification study:

$$\hat{G}_1(s) = \frac{e^{-1s}}{10s+1}\begin{pmatrix} 4.7336 & -0.6666 \\ -10.2702 & 0.6170 \end{pmatrix} \tag{3.2}$$

$$\hat{G}_2(s) = \frac{e^{-1s}}{10s+1}\begin{pmatrix} 4.7335 & -0.5929 \\ -10.2702 & 0 \end{pmatrix} \tag{3.3}$$

$$\hat{G}_3(s) = \frac{e^{-1s}}{10s+1}\begin{pmatrix} 4.7335 & 0 \\ -10.2702 & 0.5370 \end{pmatrix} \tag{3.4}$$

For simplicity, exact prior knowledge about the process dynamics is assumed available; hence, there is no model mismatch in the dynamic portion of the models. The first model (3.2) was estimated with no prior knowledge about the gain, while the second model (3.3) was estimated with a constraint ($g_{2,2} = 0$, which is slightly incorrect steady-state prior knowledge). The third model (3.4) was estimated with a different constraint ($g_{1,2} = 0$) again slightly incorrect. Both of these prior knowledge have an error of 0.1. The additive steady-state model mismatch between the three estimated models and the true process (measured by the Frobenius norm) is:

$$\left\| G - \hat{G}_1 \right\|_F = 0.99$$
$$\left\| G - \hat{G}_2 \right\|_F = 0.80 \tag{3.5}$$
$$\left\| G - \hat{G}_3 \right\|_F = 0.59$$

Although (3.3) and (3.4) were estimated with an incorrect prior knowledge, they had a smaller mismatch than the first model (3.2). One would expect that a model that has a smaller model mismatch would result in a better controller.

Next, let us assume that a multivariable controller (such as DMC) is designed based on these models. Assuming (3.1) is the true model, the response of the closed-loop system to a sequence of set point changes are simulated in Figures 3.1, 3.2 and 3.3 (the DMC tuning strategy of Shridhar and Cooper (1998) is used for the control horizon tuning (input horizon), *M*, and the prediction horizon tuning (output horizon), *P* and they are the same in all the cases). The estimated model (3.2) is used in designing the controller for the simulation shown in Figure 3.1. This model, which is estimated with no prior knowledge, resulted in an unstable controller. Figures 3.2 and 3.3 show the closed-loop system based on the estimated models (3.3) and (3.4) respectively, which were estimated using the incorrect prior knowledge. The estimated model (3.3) resulted in an unstable controller, while the estimated model (3.4) resulted in a stable controller. Although the error in the prior knowledge for both (3.3) and (3.4) were the same, the resulting control systems based on these models were quite different. It will be shown that the estimated models (3.2) and (3.3) do not meet the controller stability criteria (CSC) of Garcia and Morari (1985). The controller stability criteria is a necessary condition for stability of model based controllers; however, it is not a sufficient condition. If this condition is violated the resulting controller will be unstable independent of the controller tuning (see Appendix 1 for more detail).

This example illustrates that even if prior knowledge is incorrect it could still be useful (i.e., (3.4) results in a stable controller even though there is an error in prior

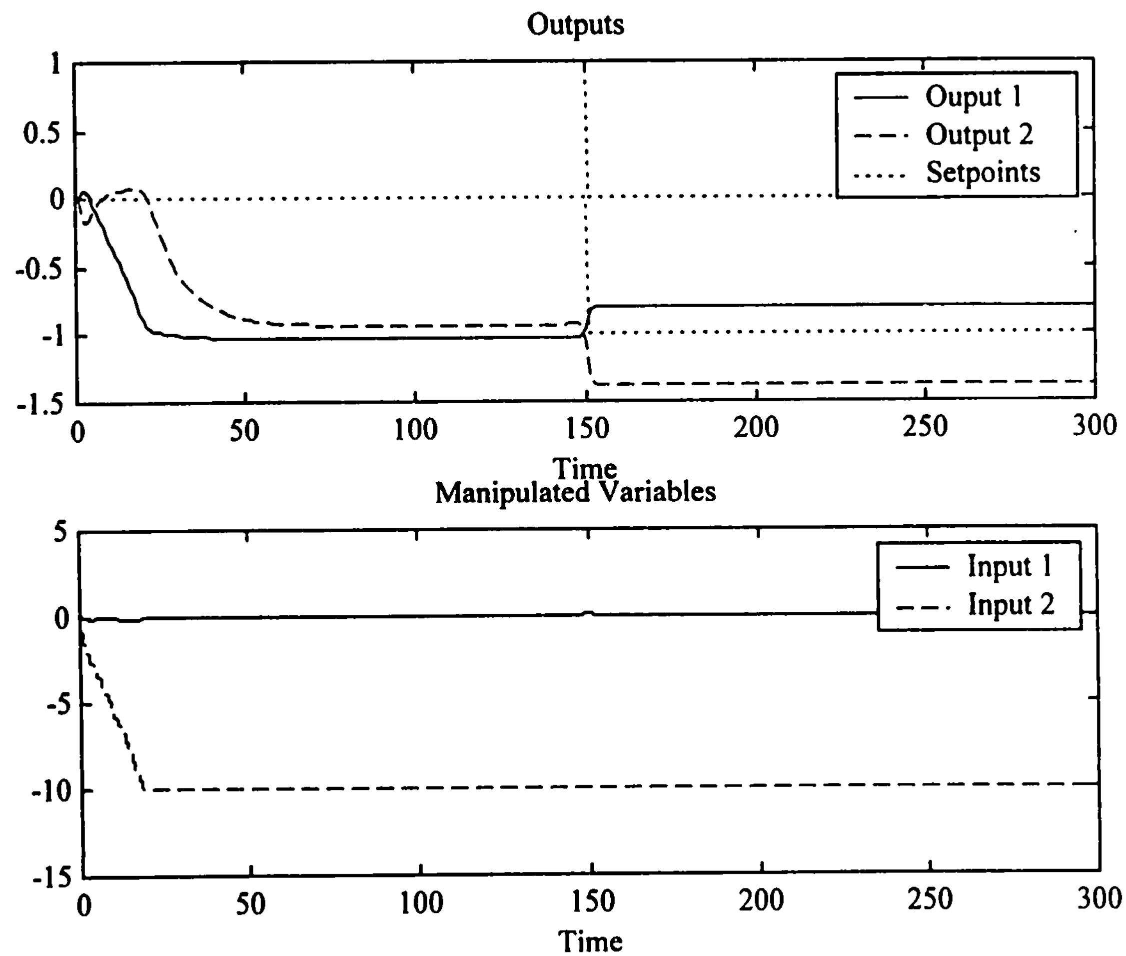


Figure 3.1: A closed-loop response to set-point changes when the DMC is designed using estimated model (2). The input horizon is 10 and the output horizon is 50 with all the inputs and the outputs being equally weighted. There is a bound on the all the inputs and outputs of +/- 10.

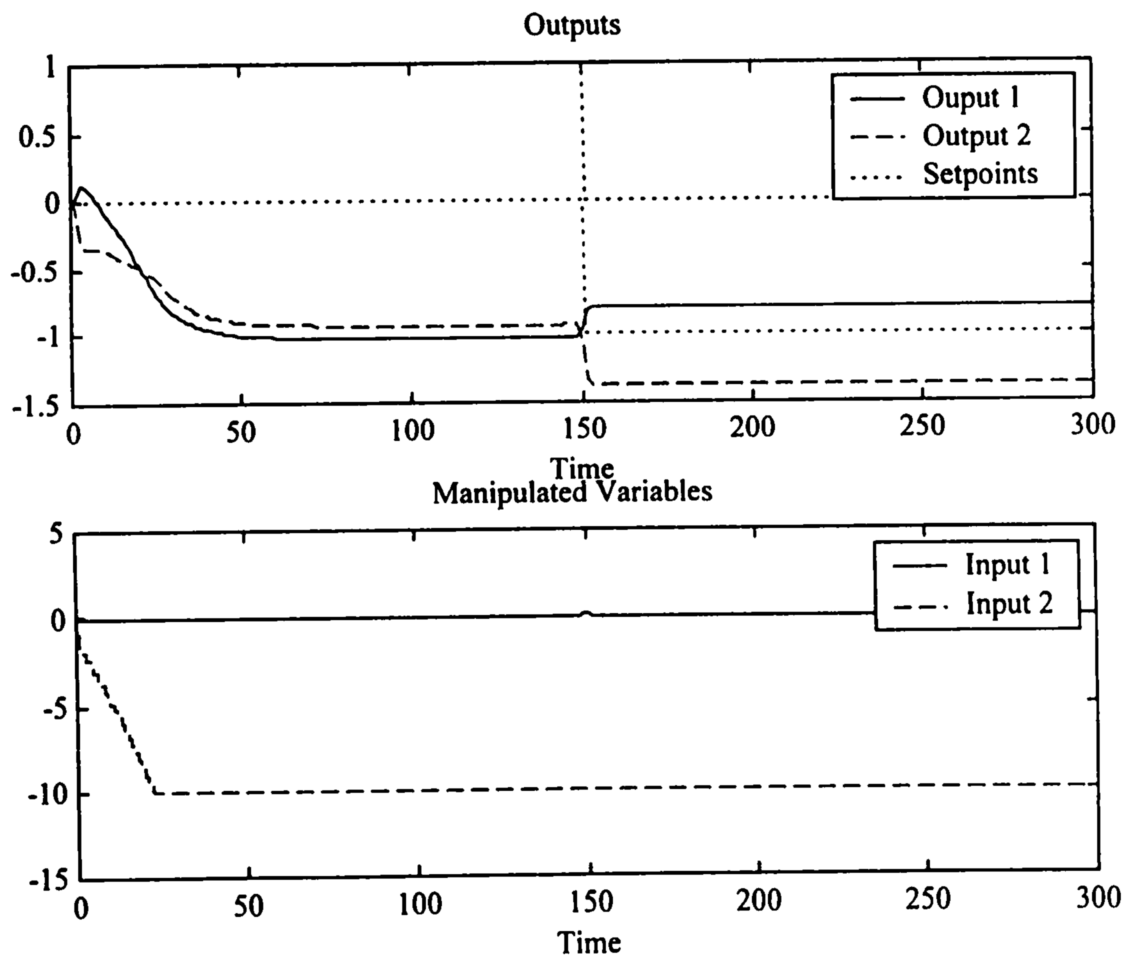


Figure 3.2 : A closed-loop response to set-point changes when the DMC is designed using estimated model (3). The input horizon is 10 and the output horizon is 50 with all the inputs and the outputs being equally weighted. There is a bound on the all the inputs and outputs of +/- 10.

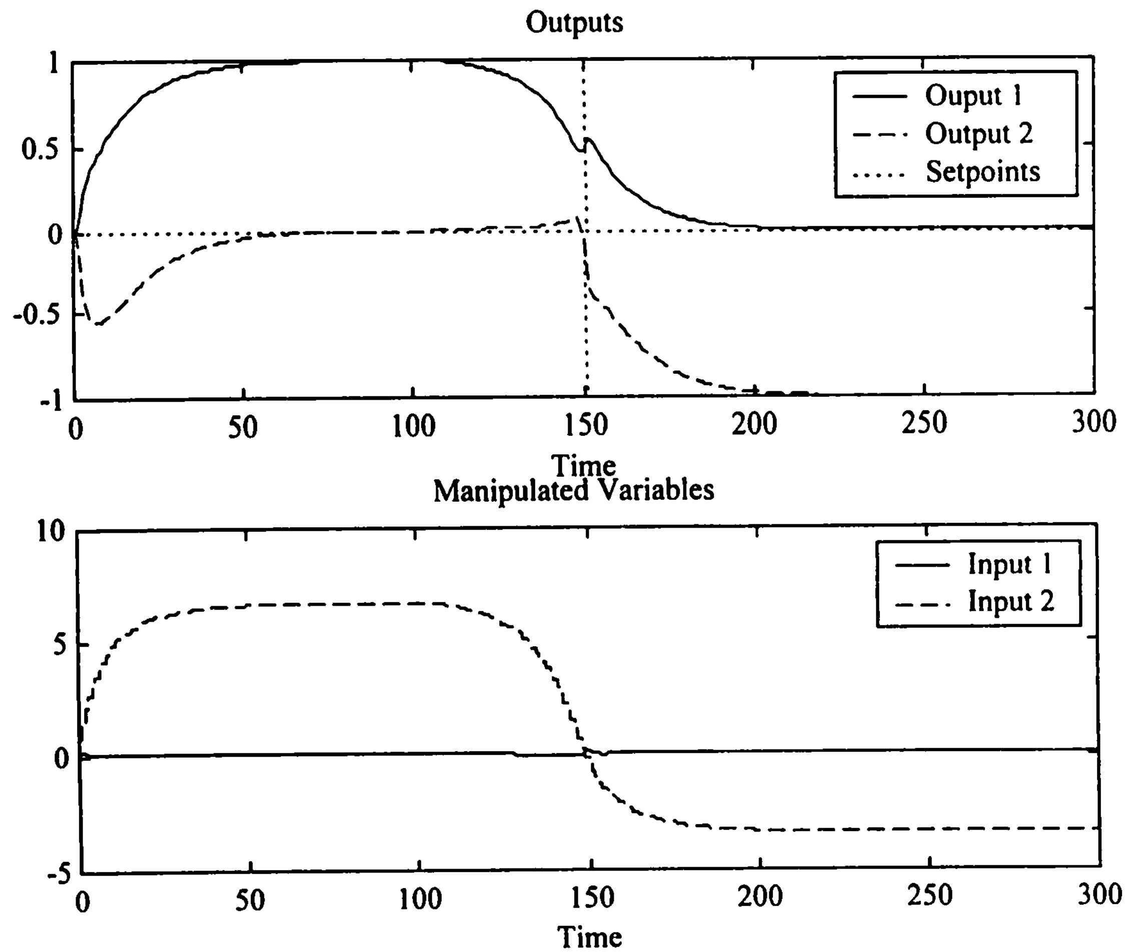


Figure 3.3: A closed-loop response to set-point changes when the DMC is designed using estimated model (4). The input horizon is 10 and the output horizon is 50 with all the inputs and the outputs being equally weighted. There is a bound on the all the inputs and outputs of +/- 10.

knowledge). It also shows that the sensitivity of the CSC to error in prior knowledge is very different for different constraints. The purpose of this work is to investigate the sensitivity of the controller stability criteria (CSC) to error in different constraints. In doing so, a checkable metric is devised that can be used to flag very sensitive constraints.

3.3. Interpretation of the Stability Criteria with an Incorrect Prior Knowledge

In the previous chapter, two different measures of model quality were discussed. In addition, it was mentioned that the stability of a multivariable control system using an empirical model might be evaluated using the steady state controller stability criteria (CSC) (Garcia and Morari, 1985). Garcia and Morari have shown that a diagonal first-order exponential filter such as (3.6) will result in a stable controller that is based on the approximate model inverse $\hat{G}^{-1}$ for some value of α_i as long as condition (3.7) holds.

$$F(z) = diag\left\{\frac{1-\alpha_i}{1-\alpha_i z^{-1}}\right\}, i = 1,\ldots,r, 0 \le \alpha_i < 1 \tag{3.6}$$

$$\mathrm{Re}\left(\lambda_i\left(G\hat{G}^{-1}\right)\right) > 0, \forall i \tag{3.7}$$

As long as this condition is not violated, a multivariable closed-loop system with no offset can be designed (see Appendix 1 for more detail on CSC). The three estimated models (3.2), (3.3) and (3.4) can be tested using (3.7):

$$\begin{aligned}
\mathrm{Re}\left(\lambda_i\left(G\hat{G}_1^{-1}\right)\right) &= \{-0.4076, 0.9376\} \not> 0, \forall i \\
\mathrm{Re}\left(\lambda_i\left(G\hat{G}_2^{-1}\right)\right) &= \{-0.2516, 0.9789\} \not> 0, \forall i \\
\mathrm{Re}\left(\lambda_i\left(G\hat{G}_3^{-1}\right)\right) &= \{1.1194, 0.5272\} > 0, \forall i
\end{aligned}$$

This confirms the simulation results obtained in Figures 3.1, 3.2 and 3.3, namely, that models (3.2) and (3.3) will result in an unstable DMC controller, while (3.4) will result in a stable DMC controller. The interpretation of this criterion for a correct prior knowledge was shown in the previous chapter. It was also mentioned that this is an uncheckable condition, since the true model for a process is never known this condition can not be used in practice. In the case of incorrect prior knowledge, there are two additional issues to consider: the direction of error in the prior knowledge, and the sensitivity of CSC to that error.

In the case of 2x2 systems, the importance of the angle between the gain directions has been discussed in the previous chapter. It has been shown that for CSC the sign of the angle between the gain vectors is a critical issue (this was also illustrated by Koung 1991). The sensitivity of that angle to incorrect prior knowledge is the issue of concern in this section.

The effect of incorrect prior knowledge can be seen geometrically in Figure 3.4. Figure 3.4(a) shows that while an error of $+\delta$ (where $\delta>0$) in the effect of input 1 on output 2, will not change the angle's sign; an error of $-\delta$ will change the sign of the angle and result in an unstable system. This illustrates the fact that not only the magnitude of error (δ), but also the direction of the error is important to the CSC. Figure 3.4(b) illustrates the same magnitude of error in a different gain element (gain between input 1 and output 1). In this case the angle's sign did not change for both $+\delta$ and $-\delta$, which implies that the system is stable and less sensitive to error in this direction. Comparison between Figures 3.4(a) and 3.4(b), illustrate that sensitivity of CSC is different for different constraints. Although this illustrative example is based on a 2x2 system, the results are analogous for any nxn systems (Figures 3.4(c) and (d)).

3.4. Sensitivity Analysis

There are two different steps in performing sensitivity analysis for model identification. The first step is to look at the sensitivity of the estimated model with respect to the data or prior knowledge. The second step is to look at the sensitivity of the

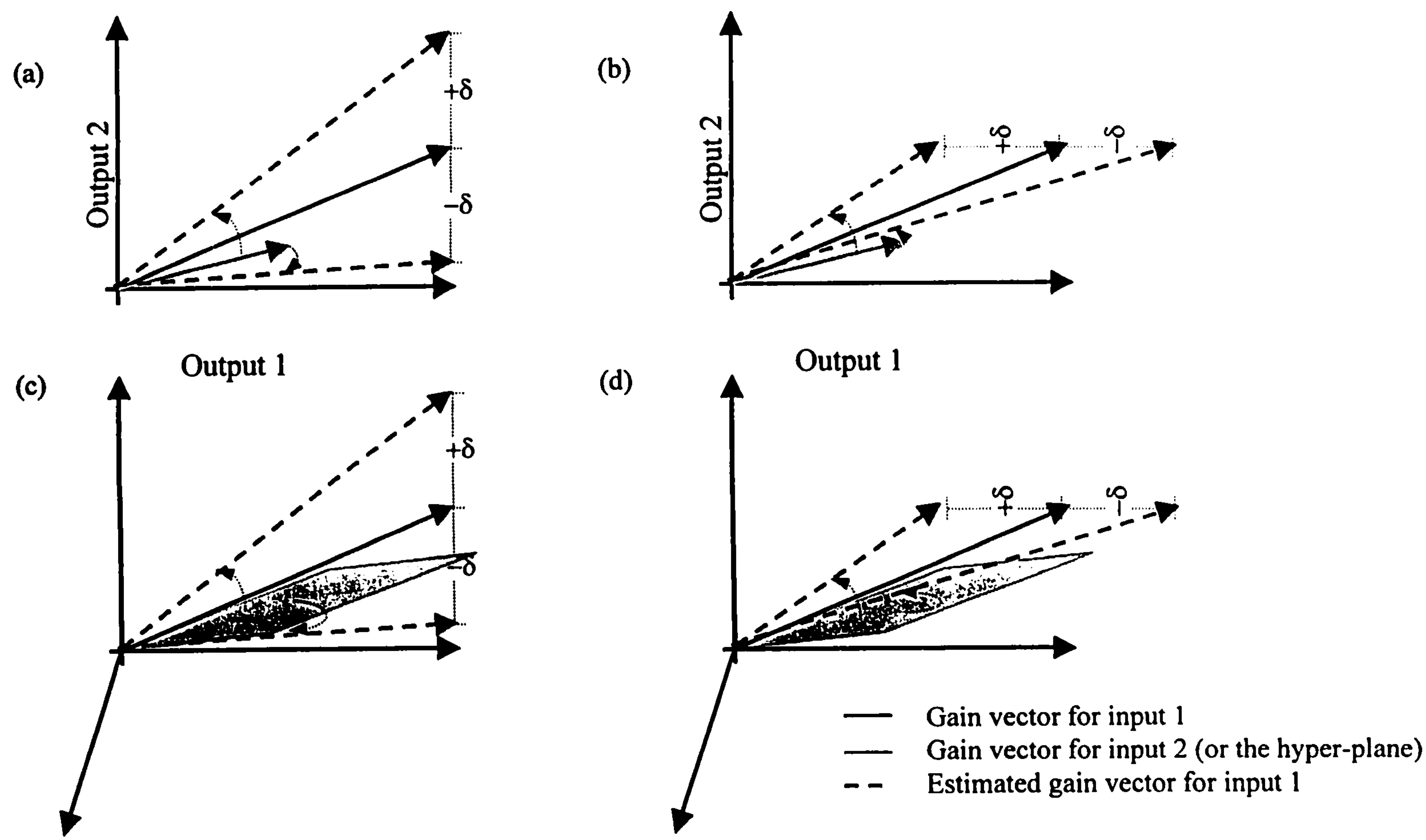


Figure 3.4: Effect of error in prior knowledge on CSC. Controller stability or unstability can be determined by observing the angle between the estimated gain vector and the hyper-plane. If the estimated angle has the same sign as the true angle then a stable controller can be designed (a) Error in the prior knowledge can result in unstable 2x2 system (b) Error in prior knowledge will not result in unstable 2x2 system (c) Error in the prior knowledge can result in unstable nxn system (d) Error in prior knowledge will not result in unstable nxn system.

end use of the model to changes in the estimated model. In our case, the model end use is the design of a controller and the criteria is based on the CSC as stated in (3.7). In other terms:

$$\frac{\partial P(UCS)}{\partial c} = \frac{\partial P(UCS)}{\partial \hat{G}} \frac{\partial \hat{G}}{\partial c} \tag{3.8}$$

where *P(UCS)* is the probability of an unstable control system (USC). In other words it is the probability that the estimated model will violate condition (3.7)

$\hat{G}$ is the estimated gain matrix

c is the constant in the constraint (or prior knowledge)

In section 3.4.2, we are concerned with the sensitivity of the gain matrix to error in prior knowledge, which is given by $\frac{\partial \hat{G}}{\partial c}$ in (3.8). In the section 3.4.3, we are interested in propagating this uncertainty to the CSC, which is given by $\frac{\partial P(UCS)}{\partial c}$ in (3.8).

3.4.1. A Simulation Study

The purpose of this simulation study is to compare the sensitivity of the CSC to errors in different constraints. In the previous chapter, it was illustrated that the conclusions regarding the CSC were similar between the cases with and without process dynamics. This was deduced from the simulation results. It was mentioned that the addition of process dynamic parameters resulted in a more complicated optimization problem and an increase in variance generally (since there are more parameters being estimated). The issues as they pertain to the sensitivity of the CSC to error in prior knowledge were similar between the case with and without dynamics. Consequently, the base case considered here is a system with no dynamics. In the matrix form a system without dynamics is:

$$Y = G \times X + E \tag{3.9}$$

where G is the gain matrix (m x m)

Y, X, and E are the output, input and error matrices respectively (m x n)

m is the number of inputs and outputs (only squared systems are considered)

n is the number of observations

In this case, the elements of E are white noise, which are normally distributed with a unit variance. The base case considered is the following 5x5 gain matrix:

$$G = \begin{pmatrix} 10 & -10 & 1 & .5 & .6 \\ 4 & -1.3 & -.2 & .75 & .6 \\ 1 & 10 & -1 & 1.5 & 1 \\ 0 & -5.5 & 0 & 0 & .25 \\ 1 & 6 & 3 & 10 & 6 \end{pmatrix} \tag{3.10}$$

This gain matrix, which has a condition number of 228 (and a determinant of -106), is neither singular nor badly ill-conditioned. This system is perturbed with a Pseudo Random Binary Signal (PRBS) with a switching time (basic period) of 1 sampling intervals and a magnitude of ± 1 (the perturbation signal does not have to be a PRBS in this case since there is no dynamics to the process or noise models). Unless otherwise specified, 500 data points were collected under open-loop conditions. The white noise error E is normally distributed with a covariance matrix I. The signal-to-noise ratio is defined as the ratio of the effect of the external excitation (which assists in identification) on the input of the process to the effect of the white noise sequence (which hinders identification) on the output of the process. In this case the signal-to-noise ratios for the 5 outputs are:

$$\frac{\sigma_{(Signal)}}{\sigma_{(Noise)}} = \begin{pmatrix} 14.2 & 4.3 & 10.3 & 5.5 & 13.5 \end{pmatrix}$$

A set of 100 different input signal realizations (with 500 data points in each realization) results in a Monte Carlo simulation. For each of the 100 data sets the process was identified by applying the following three identification methods:

1) Ordinary Least Squares (OLS)
2) Constrained Least Squares (CLS)
3) Pretest estimator (critical value of 0.05), which performs an F-test to test the validity of prior knowledge (for more information see Judge et al. 1980).

In the first two methods we compare the effect of prior knowledge on gain estimates and the CSC. The last method minimizes the effect of incorrect prior knowledge by performing a hypothesis test between the data and the prior knowledge. If the hypothesis test passes it will utilize the prior knowledge and use the CLS estimate. If the hypothesis test fails it will not make use of the prior knowledge and use the OLS estimates (Judge et al. 1980).

A set of 100 different realizations was used with the following prior knowledge to estimate the 100 different gain matrices.

$$\hat{g}_{i,j} = c_{i,j} = g_{i,j} + \delta \qquad (3.11)$$

where $g_{i,j}$ is the true gain element that there is a prior knowledge about

$\hat{g}_{i,j}$ is the estimated gain element

$c_{i,j}$ is the prior knowledge about $g_{i,j}$

δ is the error in the prior knowledge ($\delta \in \Re$, in these simulations $\delta \in \{-1, -0.9, \ldots, 0.9, 1\}$)

The different gain matrices where tested to see if they violated (3.7). Based on that, the probability of unstable control system (P(USC)) was estimated. The gain matrices were also used to estimate the MSE(G):

$$MSE(G) = \frac{\sum_{k=1}^{n} \left\| \hat{G}_{(k)} - G \right\|_F}{n}$$

where G is the true gain matrix

$\hat{G}_{(k)}$ is the gain estimate with the k input perturbation

$\|\cdot\|_F$ is the Frobenius norm

n is the number of Monte Carlo realizations (in this case $n = 100$)

There are many ways of assessing model quality (Ninness and Goodwin 1994, 1995). MSE(G) is one measure of model quality, and may be viewed as a measure of model's prediction quality. The CSC indirectly measures quality of the estimated direction of the gain matrix. Figures 3.5-3.7 shows how those model quality vary with incorrect prior knowledge in the form of (3.11).

3.4.2. Sensitivity of Gain Matrix

In this section the sensitivity of the estimated gain matrix to changes in the constraint ($\frac{\partial \hat{G}}{\partial c}$) are estimated. In section 3.4.2.1 this sensitivity is performed for linear style constraints, while in section 3.4.2.2 this is generalized for any constraint.

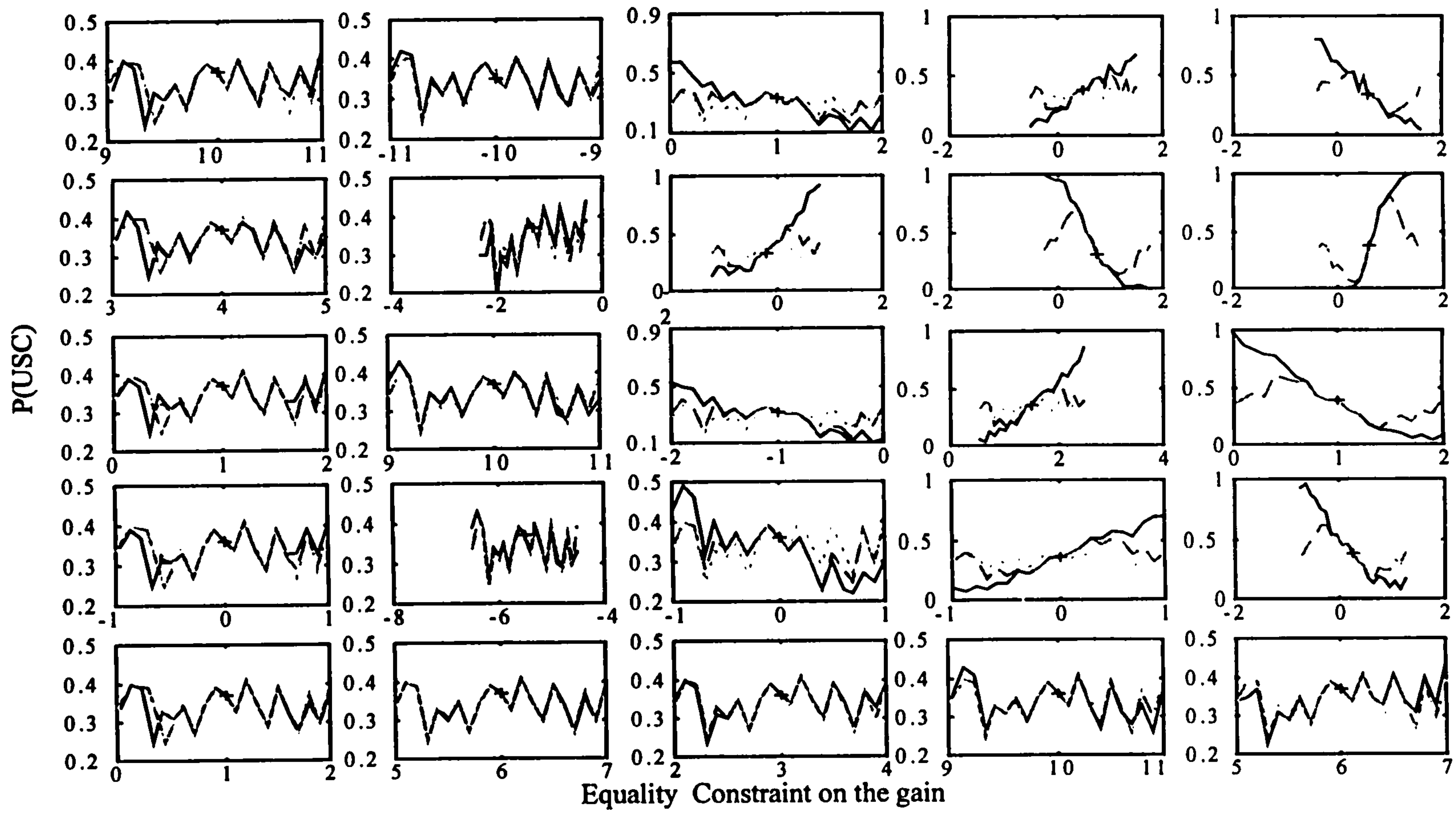


Figure 3.5: The Effect of Equality Constraint on the CSC. Subplot i, j corresponds to an equality constraint on element i, j of the gain matrix. This Figure is based on 525 (25x21) Monte Carlo simulation, where in each Monte Carlo there is a constraint of the form $c_{i,j} = g_{i,j} + \delta$ and there is 100 realization in each Monte Carlo. In each realization the system was perturbed with a PRBS of magnitude ± 0.25 and the covariance of the noise was *I*.

—— CLS
— — Pretest Estimator
OLS
\+ CLS (correct constraint)

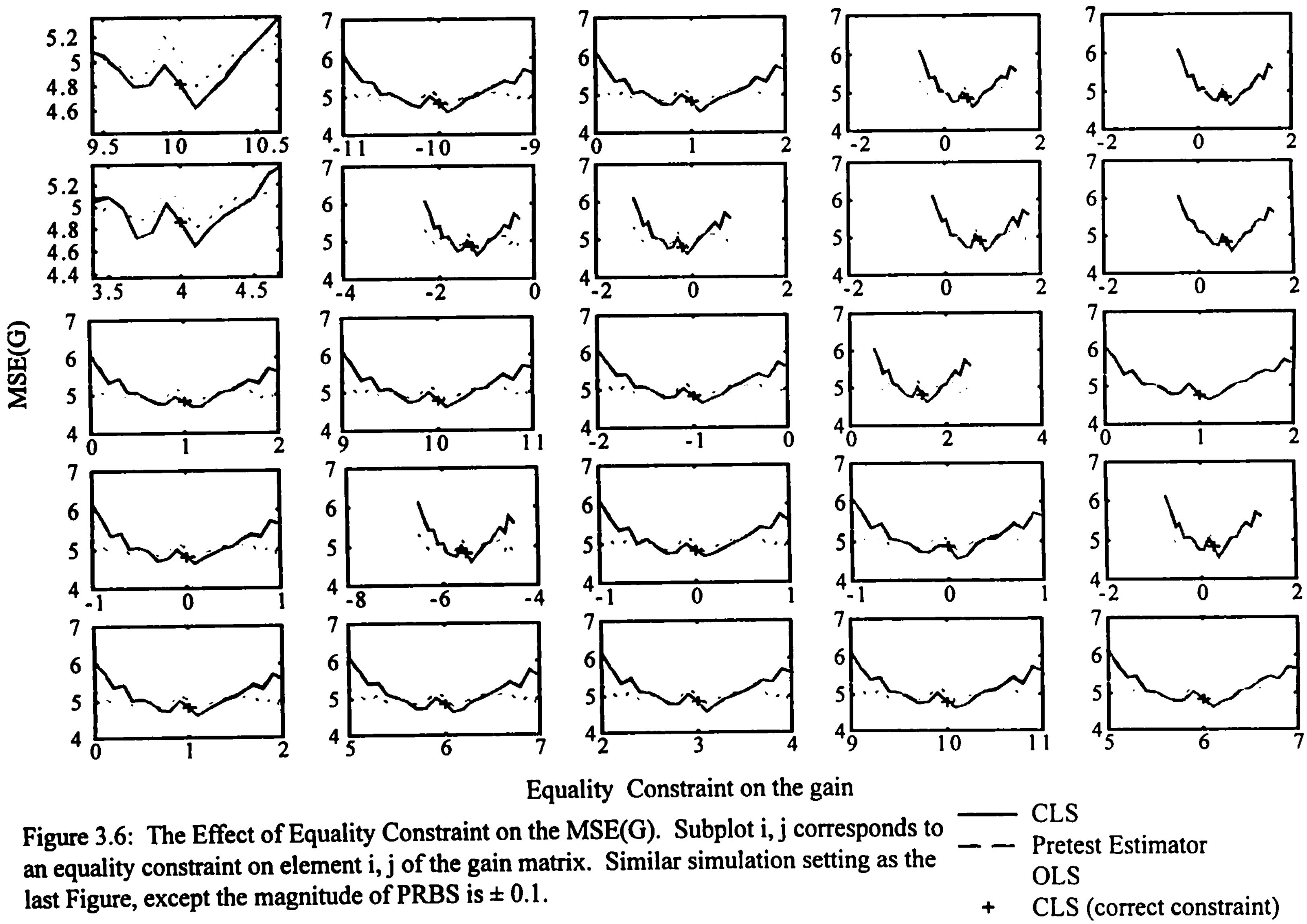


Figure 3.6: The Effect of Equality Constraint on the MSE(G). Subplot i, j corresponds to an equality constraint on element i, j of the gain matrix. Similar simulation setting as the last Figure, except the magnitude of PRBS is ± 0.1.

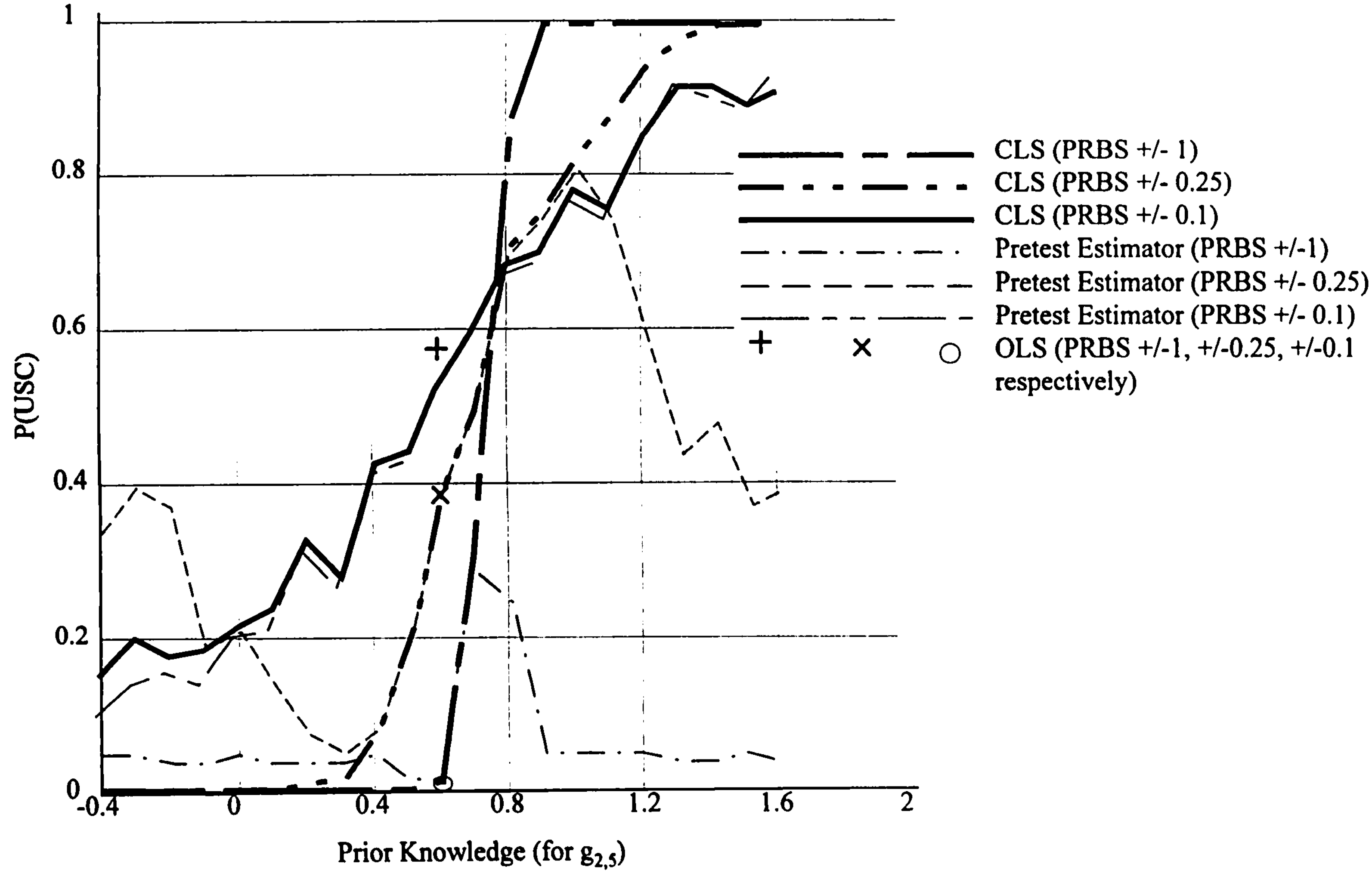


Figure 3.7: The effect of error in prior knowledge as a function of PRBS magnitude on the CSC based on Monte Carlo simulations. In this case the prior knowledge is an equality constraint on $g_{2,5}$. Note the true value for $g_{2,5}$ is 0.6.

3.4.2.1. Constrained Least Square (CLS)

The effect of an incorrect constraint on the MSE of the gain matrix can be seen in Figure 3.6. As expected the correct constraint improves the MSE of the gain matrix slightly. This can be seen in Figure 3.6 where the CLS MSE(G) is lower than the OLS MSE(G) for a correct constraint (illustrated by a + in Figure 3.6, and noticeable in closer view of plots associated with $g_{1,1}$ and $g_{1,2}$), and also when the error in the prior knowledge is less than ±0.5 (i.e., $-0.5<\delta<0.5$). Incorrect specification of the constraint will result in an increase in the MSE(G) for CLS and pretest estimator. The increase in the MSE(G) is dependent on the magnitude of δ. In addition, the MSE(G) responds symmetrically with respect to δ.

The sensitivity of $\hat{G}$, which is related to MSE(G), to δ can be estimated theoretically as well. The equation (A.9) in Appendix 3 can be differentiated with respect to δ this results in:

$$\frac{\partial\hat{\beta}_{\mathrm{H}}}{\partial\delta} = \left(X^T X\right)^{-1} R^T \left(R\left(X^T X\right)^{-1} R^T\right)^{-1} \tag{3.12}$$

where $\hat{\beta}_H$ is the solution to the CLS

R is a matrix of constants defining the linear constraints

δ is the error in the prior knowledge

The above equation illustrates the sensitivity of the CLS solution to error in the prior knowledge. Similarly in the case of an inexact equality constraint (stochastic prior information) it can be shown from (A.12) of Appendix 4 that:

$$\frac{\partial \hat{\beta}_R}{\partial \delta} = \left(X^T X + R^T \Omega^{-1} R\right)^{-1} R^T \Omega^{-1} \tag{3.13}$$

where Ω is the covariance of the inexact linear equality constraint

$\hat{\beta}_R$ is the solution to the least square problem with the inexact linear equality constraint

The sensitivity of the solution to changes in R and Ω can also be estimated in a similar fashion as (3.12) and (3.13).

3.4.2.2. Other Constraints

Comprehensive treatments of linear programming (LP) sensitivity analysis have been done by many researchers (Fiacco 1983; Gal, 1980). Although not as complete as the case with LP, some basic sensitivity results in quadratic programming (QP) have been accomplished (Boot, 1963). Compared to LP and QP, treatment of sensitivity in nonlinear programming (NLP) methodology has only recently begun to materialize (Fiacco 1976, 1983; Wolbert et al. 1992,1994). The basic idea in these works is to estimate the sensitivity using the Karush-Kuhn-Tucker conditions (Wolbert et al. 1992 and 1994).

The sensitivity of CSC to a constraint (a prior knowledge) can be evaluated by performing sensitivity analysis in the optimization routine. In general terms this is accomplished by:

$$\frac{d\left(\det\left(\hat{G}\right)\right)}{dc} = \frac{\det\left(\arg\left\{\begin{matrix} \underset{\hat{G}_+}{Min} SSE \\ s.t. f(\hat{G}) = c + \varepsilon \end{matrix}\right\}\right) - \det\left(\arg\left\{\begin{matrix} \underset{\hat{G}_-}{Min} SSE \\ s.t. f(\hat{G}) = c - \varepsilon \end{matrix}\right\}\right)}{2\varepsilon} \tag{3.14}$$

where *SSE* is the Sum of Squared Error between the actual and predicted outputs.

$\hat{G}_-$ and $\hat{G}_+$ are the gain matrix estimated with change in the constraint of $-\varepsilon$ and $+\varepsilon$

$f(\hat{G}) = c$ is any linear or non-linear constraint

ε is a small change

From the estimated gain matrices, the sensitivity of the determinant to the error in the constraint can be estimated. As was shown in the previous chapter the determinant is directly linked to the CSC.

There are disadvantages to evaluating sensitivity using (3.14). For example, in (3.14), ε has to be greater than the radius of convergence of the optimization method (i.e., the numerical method). If this is not the case, unstable results due to numerical noise inside the radius of convergence will result. Generally this can produce derivatives with wrong signs and magnitudes. The appropriate ε is therefore dependent on the numerical method and will vary from application to application. By looking at the fundamental development via the optimality conditions (Karush-Kuhn-Tucker conditions), the dependence on the numerical method is removed since they will all produce stationary points, regardless of the technique and tuning of the numerical routines. Furthermore, the optimizer will have constraint sets that can change from execution to execution so that when you select an ε your constraint set may change which would dominate the optimization and can change the sensitivity drastically. A small ε that does not result in a changed active set may be too small to give meaningful results. In order to prevent this problem, the sensitivity analysis has to be performed by analytical derivatives as will be shown in the next section.

3.4.2.3. Sensitivity Analysis in Quadratic Programming Problems

Some of the research on sensitivity analysis in quadratic programming (QP) has been performed by Boot (1963), and Theil (1961) who concentrated in the area of econometrics. A conventional quadratic programming problem can be stated as:

$$\underset{x}{Min}\frac{1}{2}x^T Hx + f^T x \tag{3.15}$$

such that:

$$\begin{aligned} &A \cdot x \leq b \\ &Aeq \cdot x = beq \\ &lb \leq x \leq ub \end{aligned} \tag{3.16}$$

Suppose that the subset S out of all constraints (3.16) in the minimization of (3.15) are active. Then if we minimize (3.15) subject to the constraints belonging to S taken as exact equalities, we get the vector x^s, which solves (3.15) and (3.16). The minimization process is accomplished with the use of a Lagrangian multiplier.

$$L = f^T x + \frac{1}{2}x^T Hx + \lambda^s (C_s x - d_s) \tag{3.17}$$

where C_s is a matrix of coefficients of all the active linear constraints

d_s is a vector of constants associated with all the active linear constraints

Differentiating (3.17) with respect to x and λ^s we can solve for x (see Boot 1963 for more detail).

$$x^s = H^{-1}f - H^{-1}C_s^T\left(C_s\, H^{-1}C_s^T\right)^{-1}\left(C_s H^{-1}f - d_s\right) \tag{3.18}$$

The Sensitivity of the solution to changes in the constraint can be seen by:

$$\frac{\partial x^s}{\partial d_s} = H^{-1}C_s^T\left(C_s H^{-1}C_s^T\right)^{-1} \tag{3.19}$$

It can be seen that (3.19) is very similar to (3.12). The main difference is that (3.12) is suitable for constraints that effect one output at a time; while, (3.19) can handle constraints that effect multiple outputs. In addition, (3.19) can be used for non-linear and inequality constraints easier than the CLS method.

3.4.3. Sensitivity of CSC

The sensitivity of CSC to equality constraints on the individual gain elements (as shown in (3.11)) is shown in Figure 3.5. This was obtained using the constrained least square method. Unlike the MSE(G), which responded similarly to both $\delta>0$ and $\delta<0$ (as shown in Figure 3.6), stability is dependent on the direction of the error made in the constraint. This is evident since in one direction the constraint will result in a system that is more ill-conditioned while in the opposite direction the system becomes better conditioned. It can also be seen that the sensitivity of the probability of UCS is not the same for all constraints. This can be seen by the fact that the rate of change is very different for different constraints (i.e., compare constraint on $g_{2,5}$ with $g_{2,1}$)

When the correct constraint is implemented ($\delta = 0$), then the probability of UCS is less than the probability of UCS for OLS estimator (Figure 3.7). In some of the simulations that is not the case, this can be attributed to the number of realizations in a Monte Carlo simulation not being sufficiently large. In the previous chapter, it was shown that for an equality constraint with $\delta = 0$ (i.e., correct prior knowledge) the stability will always be better than the unconstrained case.

The effect of signal-to-noise ratio on the sensitivity of the CSC to error in prior is illustrated in Figure 3.7. In this case the magnitude of the perturbation signal was changed. It can be seen that as the signal-to-noise ratio decrease the sensitivity of the CSC to errors in the prior knowledge also decreases. Clearly as the signal-to-noise ratio decrease the P(USC) increases. In addition, the pretest estimator does little in detecting error in prior knowledge when the signal-to-noise ratio is low. This is due to the fact that the model used by the pretest estimator to validate the constraint has a poor quality when signal-to-noise ratio is low. However, when the signal-to-noise ratio is high the pretest estimator can detect error in prior knowledge and limit sudden increase in P(USC). In contrast, when the error in the constraint improves the model quality (i.e., in the case of Figure 3.7 when $c_{i,j} \rightarrow -\infty$) the pretest estimator again detects an error in constraint and limits the effectiveness of such a constraint. Therefore, the pretest estimator will limit both increases (degradation of model quality) and decreases (improvement of model quality) in P(USC) and for this reason it is not recommended to be used by the practitioner.

In the Monte Carlo simulations that resulted in Figures 3.5 and 3.7 the probability of UCS could be estimated since the true model was known. In real situations, the sensitivity of CSC has to be estimated without the knowledge of the real plant model. In sections 3.4.3.1 and 3.4.3.2, two different solutions are presented for this problem. In 3.4.3.1, the sensitivity of the determinant to changes in constraints are studied; while in 3.4.3.2, the sensitivity of individual eigenvalues to changes in constraints are evaluated.

3.4.3.1. Perturbation of Determinant

3.4.3.1.1. General Sensitivity of Determinant

One way of checking to see how the error in the constraint on a gain parameter would effect the stability, is to observe the effect of that gain element on the determinant of the gain matrix. As was shown in the previous chapter, if the sign of the determinant of the true gain matrix is different from the estimated gain matrix then the system is unstable. Consequently, to see which gain elements have the most effect on the determinant, we can evaluate:

$$\frac{\partial|G|}{\partial G} = |G|(G^T)^{-1} \tag{3.20}$$

which for the 5x5 system (3.10) is:

$$\frac{\partial|G|}{\partial G} = |G|(G^T)^{-1} = \begin{pmatrix} -6.4 & -13 & -107 & 216 & -293 \\ -11 & 38 & 289 & -608 & 833 \\ -0.6 & -19 & -75 & 284 & -416 \\ 14 & 0.8 & -14 & 245 & -405 \\ 1.2 & 0.7 & -5.1 & -18 & 15 \end{pmatrix} \tag{3.21}$$

Note that $\frac{\partial|G|}{\partial G} = |G|(G^T)^{-1}$ is the same as the cofactor of G^T. A zero value for the cofactor would indicate that the sub-matrix used in the calculation of the minor is singular. Similarly, a small cofactor is an indication of poor conditioning of the sub matrix.

From this analysis, one can see that $\frac{\partial|G|}{\partial g_{2,5}} = 833$ is the largest element, suggesting that it is important to have the correct constraint on this element since an error in this constraint will have a significant effect on the stability. This analysis can be confirmed by the Monte Carlo simulations shown previously in Figures 3.5 and 3.7. In contrast, error in a constraint on $g_{3,1}$ has little effect on stability ($\frac{\partial|G|}{\partial g_{3,1}} = -0.6$). This type of analysis has the same weaknesses as was mentioned in the previous chapter regarding utilizing the sign of determinant in estimating the probability of controller stability, namely that it is a sufficient condition for instability but not a necessary condition.

While the error in the constraint has been based on absolute perturbations in the constraints, it can also be considered for a relative perturbation. For this 5x5 system, based in relative error (3.21) can be rewritten as:

$$\frac{\partial|G|}{\partial G}./G = \left(|G|\left(G^T\right)^{-1}\right)/G = \begin{pmatrix} -0.6 & -1.3 & -107 & 434 & -488 \\ -2.6 & 29.1 & 1445 & -811 & 1389 \\ -0.6 & -1.9 & -75 & 189 & -416 \\ \infty & 0.2 & -\infty & \infty & -1618 \\ 1.2 & 0.1 & -1.7 & -1.8 & 2.4 \end{pmatrix} \quad (3.22)$$

In (3.22) ./ is the element by element division. This method of observing the sensitivity provides an estimate of sensitivity based on the percentage error in the constraint. The drawback is that it can not provide a meaningful value when the gain element in question is zero or close to zero. Based on (3.22), the most sensitive constraint excluding the three elements that were infinite (due to the true gain elements being zero), is $g_{4,5}$. $g_{2,5}$, which in (3.22) was the most sensitive constraint, is now the third most sensitive constraint. This is due to the fact that $g_{2,5}$ is 2.4 and 3 times larger in magnitude compare to $g_{4,5}$ and $g_{2,3}$ respectively, which also illustrates how the magnitude of the gain elements play a role in this method of estimating the sensitivity.

Equation (3.20) presents a simple and practical method of checking the sensitivity of CSC to error in equality constraint on the gain. In practice, when G is not known an estimate of G can be used (i.e., $\frac{\partial|\hat{G}|}{\partial\hat{G}}$) and (3.20) will provide the sensitivity of different gain element equality constraint on the CSC.

3.4.3.1.2. Sensitivity of Determinant and CSC

The sensitivity of the CSC to error in the prior knowledge can be evaluated by observing the sensitivity of the determinant to error in the prior knowledge.

$$\frac{d\det\left(\hat{G}\right)}{dc}=\frac{d\det\left(\hat{G}\right)}{d\Delta}\frac{d\Delta}{dc} \tag{3.23}$$

Δ is the perturbation matrix. The first term on the right hand side of the above equation is similar to (3.20). It evaluates the sensitivity of the determinant to changes in the gain matrix elements. The second term on the right hand side, shows the effect of changing constraint on the elements of the gain matrix. This term can be evaluated using (3.19).

A large value for (3.23), implies that the determinant is very sensitive to changes in the constraint. Certainly, such a result is dependent on the quality of data and type of prior knowledge. In contrast, a small value for (3.23) would suggest that the constraint has little effect on the determinant, which would imply that it has little effect on CSC. It is important to note that such values are relative and can not be compared between different systems.

Another method of studying the effect of the constraint on CSC, which would allow comparison between different systems, is to compare the change in the probability of the determinants changing sign. Assuming the determinant of the estimated gain matrix is normally distributed, one can show the following (see Appendix 7):

$$\text{if } \det(G) < 0, \frac{dP\left(\det\left(\hat{G}\right) > 0\right)}{dc} = \frac{dP\left(\det\left(\hat{G}\right) > 0\right)}{d\det\left(\hat{G}\right)} \frac{d\det\left(\hat{G}\right)}{dc} = -P\left(\det\left(\hat{G}\right) = 0\right) \frac{d\det\left(\hat{G}\right)}{dc}$$

$$\text{if } \det(G) > 0, \frac{dP\left(\det\left(\hat{G}\right) < 0\right)}{dc} = \frac{dP\left(\det\left(\hat{G}\right) < 0\right)}{d\det\left(\hat{G}\right)} \frac{d\det\left(\hat{G}\right)}{dc} = P\left(\det\left(\hat{G}\right) = 0\right) \frac{d\det\left(\hat{G}\right)}{dc}$$

(3.24)

Such a method will consider the uncertainty in $\det(\hat{G})$ and the sensitivity of $\det(\hat{G})$ to changes in prior knowledge. Figure 3.8 illustrates how (3.24) encompasses both model uncertainty and sensitivity of determinant. As will be illustrated latter (Figure 3.10), in most cases it can be assumed that the determinant of the estimated gain matrix is normally distributed and $P(\det(\hat{G}) = 0)$ can be estimated assuming normality.

3.4.3.2. Perturbation of Eigenvalues

One of the fundamental problems of this research has been that the eigenvalues of the gain matrix are not differentiable functions of the elements of the gain matrix. However, this does not suggest that individual eigenvalues cannot behave in a locally linear fashion. In this subsection, the sensitivity of a single eigenvalue to perturbation of gain matrix elements is studied. There is extensive work performed on the perturbation of eigenvalue for more detail see Stewart and Sun (1990).

The perturbation theory used for estimating the sensitivity of the eigenvalue of a matrix to perturbation in its element is based on Gerschgorin theorem. The theorem by Gerschgorin (Stewart and Sun 1990) states that:

"let λ be an eigenvalue of an nxn matrix $A = [a_{jk}]$. Then for some integer j ($1 \le j \le n$):

$|a_{jj} - \lambda| \le |a_{j1}| + |a_{j2}| + \cdots + |a_{j,j-1}| + |a_{j,j+1}| + \cdots + |a_{jn}|$"

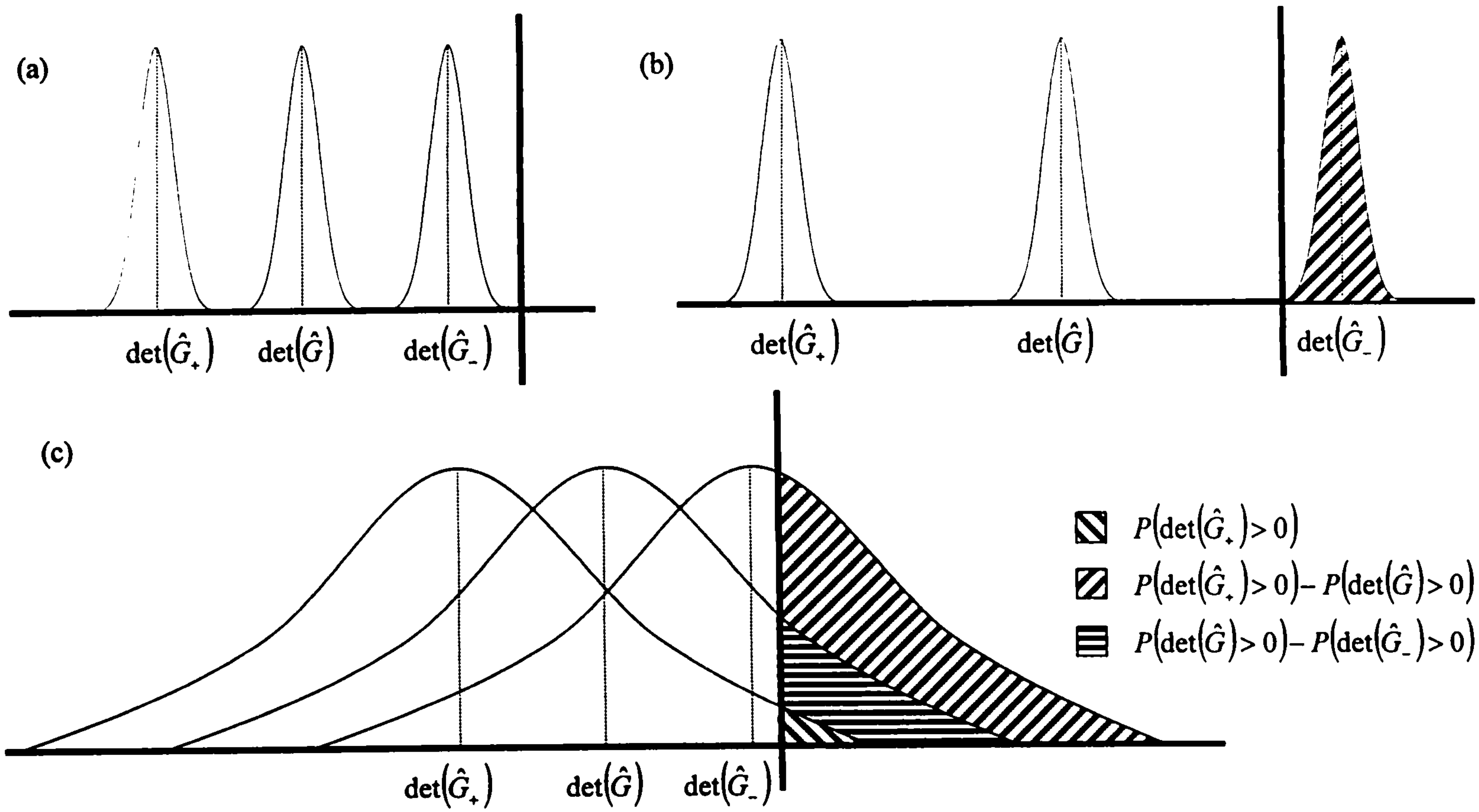


Figure 3.8: Effect of error in prior knowledge on P(USC), when the sensitivity of P(USC) is estimated by the probability of the determinant changing sign. (a) There is relatively a small uncertainty in the determinant of the estimated gain matrix and deviation in the constraint has a small effect on the determinant. (b) There is relatively a small uncertainty in the determinant of the estimated gain matrix; however, deviation in the constraint has a large effect on the determinant. As a result there is a large change in the P(USC) (c) There is a large uncertainty in the determinant of the estimated gain matrix and the deviation in the constraint has a small effect on the determinant

Based on this inclusion theorem a first-order approximation to the perturbed eigenvalue can be estimated (Stewart and Sun 1990):

$$\tilde{\lambda}_i = \lambda_i + \frac{y_i^H \Delta x_i}{y_i^H x_i} + O\left(\|\Delta\|^2\right) \tag{3.25}$$

where Δ is the perturbation matrix

λ_i is the ith eigenvalue of A

$\tilde{\lambda}_i$ is the ith eigenvalue of $A+\Delta$

x_i is a right eigenvector of A

y_i is a left eigenvector of A

$\cdot^H$ is the Hermitian of a matrix

Equation (3.25) is only valid for simple eigenvalues (an eigenvalue whose multiplicity equals one is called a simple eigenvalue). The probability of having non-simple eigenvalues should be very low in the case of gain matrices. The effect of perturbation on the resulting eigenvalues can also be estimated numerically. This can be accomplished by:

$$\tilde{\lambda}_i \approx \lambda_i + \lambda_{i,A+\Delta/2} - \lambda_{i,A-\Delta/2} \tag{3.26}$$

Using (3.26) the sensitivity of the eigenvalue to perturbation in the gain matrix can be estimated. This expression can evaluate the sensitivity of each individual eigenvalue to perturbations in the constraint. (3.23) and (3.24) evaluate the sensitivity of the determinant, which is a multiple of all the eigenvalues, to perturbations in constraints. Evaluating the sensitivity by (3.23) and (3.24) is more general; however, (3.25) and (3.26) evaluates the sensitivity in a much more explicit fashion. The two methods should be used in conjunction with one another as shown in the next example.

Example 3.1: The purpose of this example is to illustrate how in practice sensitivity analysis maybe performed. In this case it is assumed that the true model is not known and only an estimate of the true model is available. The estimated model is based on 500 observations which are collected under the conditions stated in section 3.4.1. Consider four different prior knowledge for the gain matrix (3.19):

1) $g_{2,5} = 0.6$

2) $-0.54g_{2,4} + 0.84g_{2,5} = 0.099$

3) $g_{2,4} - 1.25g_{2,5} = 0$

4) $g_{4,3} = 0$

In this example the sensitivities of the determinant and the eigenvalues of the gain matrix to changes in the constraint of the following form are considered:

$$g_{2,5} = 0.6 \pm \varepsilon$$

$$-0.54g_{2,4} + 0.84g_{2,5} = 0.099 \pm \varepsilon$$

$$g_{2,4} - 1.25g_{2,5} = 0 \pm \varepsilon$$

$$g_{4,3} = 0 \pm \varepsilon$$

The sensitivity of the determinant to changes in the constraint can be estimated using (3.23). The sensitivity of the eigenvalues can be estimated theoretically using (3.25) and numerically using (3.26). The estimated sensitivities based on a data set are shown in Table 3.1. The same result is also illustrated in Figure 3.9 graphically.

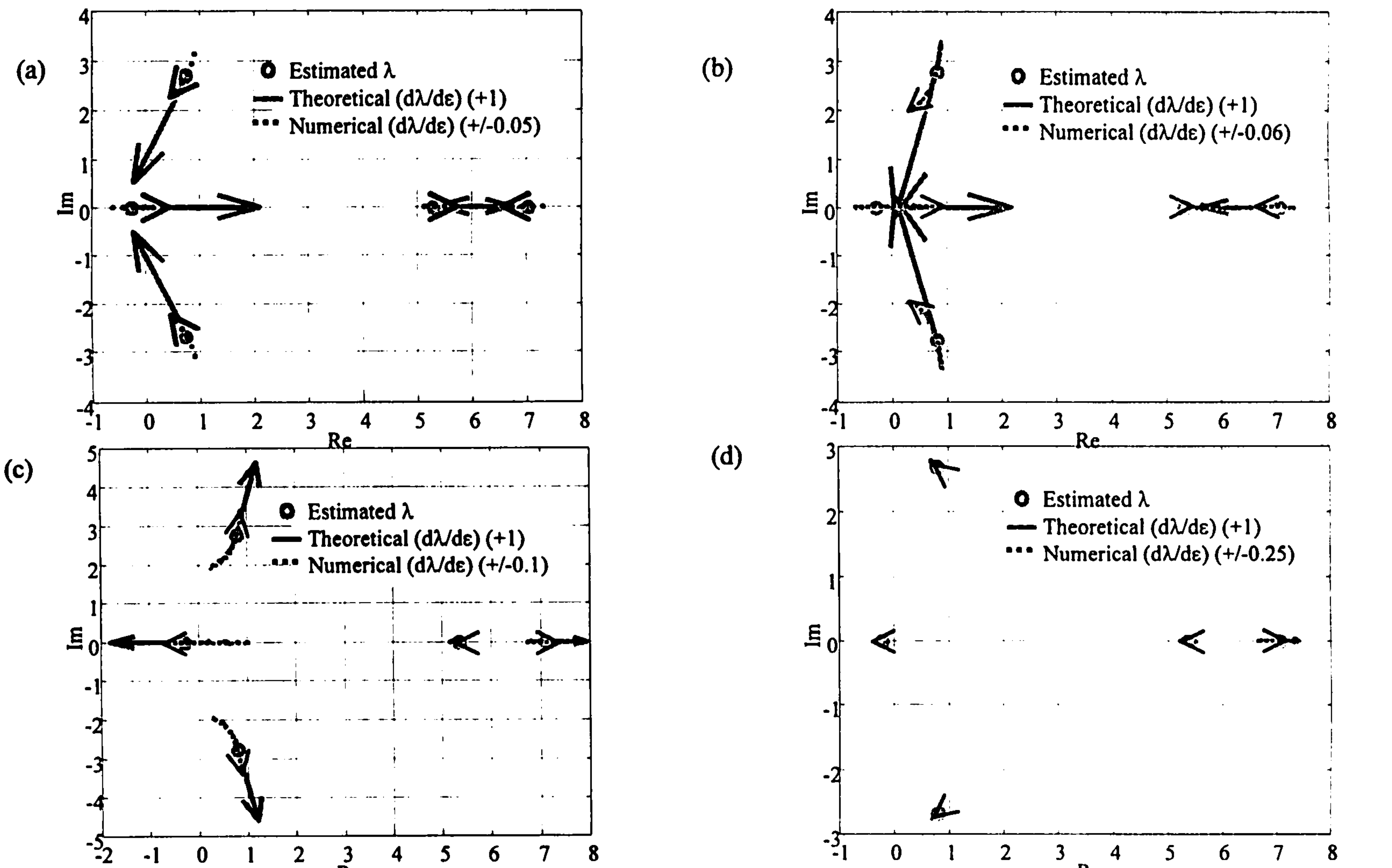


Figure 3.9: Sensitivity of the estimated gain matrix eigenvalues to perturbations in the constraint. This estimate is based on one data set. (The value in the bracket is the amount of error considered in each case)*(a)* $g_{2,5} = 0.6$ *(b)* $-0.54g_{2,4} + 0.84g_{2,5} = 0.099$ *(c)* $g_{2,4} - 1.25g_{2,5} = 0$ *(d)* $g_{4,3} = 0$

Table 3.1: The sensitivity of the eigenvalues and determinants to changes in prior knowledge

Type of Prior Knowledge	$\frac{d\det(\hat{G})}{dc}$	λ_i	$\frac{\tilde{\lambda}_i-\lambda_i}{\varepsilon}$	$\left\|\left(\frac{\tilde{\lambda}_i-\lambda_i}{\varepsilon}\right)\right\| \Big/ \|\lambda_i\|$
$g_{2,5} = 0.6$	792.1	-0.2235	2.3170	10.3653
		0.7765 - 2.6847i	-1.0076 + 2.1516i	0.8501
		0.7765 + 2.6847i	-1.0076 - 2.1516i	0.8501
		5.3476	1.3625	0.2548
		7.0574	-1.6280	0.2307
$-0.54g_{2,4}+0.84g_{2,5} = 0.099$	975.0	-0.2903	2.4773	8.5339
		0.7983 - 2.7593i	-0.7344 + 2.9477i	1.0576
		0.7983 + 2.7593i	-0.7344 - 2.9477i	1.0576
		5.3311	0.4467 - 0.0000i	0.0838
		7.0958	-1.4014 - 0.0000i	0.1975
$g_{2,4} - 1.25g_{2,5} = 0$	-607.0	-0.3002	-1.5042 - 0.0000i	5.0103
		0.7999 - 2.7732i	0.4093 - 1.8573i	0.6589
		0.7999 + 2.7732i	0.4093 + 1.8573i	0.6589
		5.3337	-0.1692 - 0.0000i	0.0317
		7.0996	0.8203 + 0.0000i	0.1155
$g_{4,3} = 0$	-21.20	-0.2306	-0.0569	0.2467
		0.7888 - 2.6891i	-0.0973 - 0.0832i	0.0457
		0.7888 + 2.6891i	-0.0973 + 0.0832i	0.0457
		5.3511	-0.1368 + 0.0000i	0.0256
		7.0378	0.3670 - 0.0000i	0.0521

Therefore based on the sensitivity of the determinant the order of the constraints from the most to least sensitive is: 2, 1, 3 and 4. While based on the smallest eigenvalue the order is slightly different: 1, 2, 3, and 4.

An estimate of the sensitivity of the first and the last prior knowledge was also observed in (3.21). The results in this example match the results shown in (3.21), which show that the sensitivity (based on determinant) of $g_{2,5} = 0.6$ is higher than $g_{4,3} = 0$. Also the sensitivity of the smallest eigenvalue to changes in $g_{2,5} = 0.6$ is higher than $g_{4,3} = 0$. Those results also match the Monte Carlo simulation results that were illustrated previously in Figure 3.5. In addition, based on the sensitivity of the smallest eigenvalue $g_{2,5} = 0.6$ is the most sensitive constraint. It is important to note that prior knowledge in the form of equality constraint on the gains are common in chemical process (see previous chapter). Consequently, it is important to evaluate the sensitivity of such a constraint on CSC.

The second constraint is actually a simplified version of the constraint on the smallest eigenvalue. The effect of a constraint on the smallest eigenvalue is shown in Appendix 8. Other than the case when the eigenvalue is zero, it is rare to have prior knowledge about an eigenvalue. The prior knowledge that the eigenvalue of a system is zero is very critical information. This would suggest that the system is singular and for a square problem, it implies that the output cannot be controlled in one of the dimensions. For the 5x5 system, that is being considered a constraint on the smallest eigenvalue is equivalent to 5 linear equality constraints on the gain matrix:

$$G\times[-0.0239,\ 0.0004,\ 0.0176,\ -0.5402,\ 0.8410]^T = [0.0092,\ -0.0002,\ -0.0068,\ 0.2081,\ -0.3240]^T$$

The second constraint, in the previous example, is derived from the multiplication of the second row of G with the eigenvector associated with the smallest eigenvalue (in the above equation). In addition, only the last two elements of the eigenvector are used (since the first three elements of the eigenvector are small they are not considered). It is rare in chemical processes to have a prior knowledge about the eigenvector; however, it is plausible to have prior knowledge about the relationship between two gain elements, which is the second constraint.

The third constraint is a simplified version of the second constraint and is perhaps more practical and more common style of constraint, since it is simpler. It states that $g_{2,4}$ is 25% larger than $g_{2,5}$. As was illustrated in the previous chapter, prior knowledge on the ratio of gains is common in chemical industry. For example, in distillation towers it is common to know a bound on the ratio of the gain for two adjacent tray temperatures.

In Example 3.1, an estimate of the sensitivity to different constraints was illustrated based on one data set. The confidence interval in such an estimate can be estimated by Jackknifing or Bootstrapping methods. A different approach to estimating the uncertainty in the sensitivity analysis based on many data sets is to perform a Monte Carlo style simulation, which can estimate the bias and variance (since it is assumed that the true model is known). The results of such a simulation are shown in Tables 3.2 and 3.3. Since the mapping of the uncertainty is non-linear the confidence interval is not equal in both directions.

Similar to Example 3.1, the sensitivity of the determinant to perturbations in constraint can also be evaluated using the Monte Carlo simulation. The results for the distribution of the determinant are listed in Table 3.3. Furthermore, the assumption made in derivation of (3.24), about the distribution of the determinant being normal can also be tested. Figure 3.10 shows the distribution of the determinant on a normal probability plot; it appears that the assumption of the determinant being normally distributed is valid. Other simulation results, which are not shown here, illustrate that this assumption is satisfactory until the signal-to-noise ratio becomes very small.

From Tables 3.1 and 3.3, it can be seen that sensitivity of CSC to different types of prior knowledge can be very different. The sensitivity results are also dependent on the signal-to-noise ratio (see Figures 3.7 and 3.11). First, the confidence interval in the estimated sensitivity increases as the ratio of signal-to-noise decreases. This is caused by

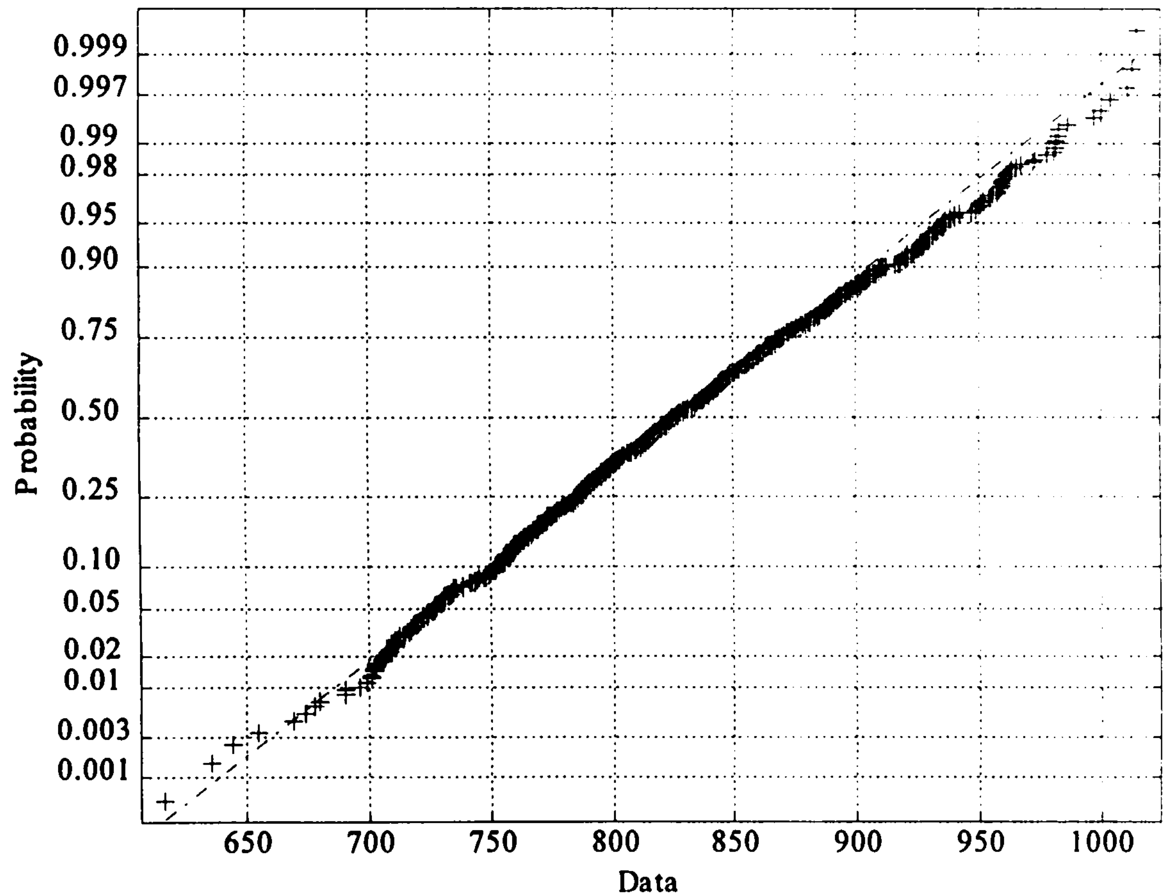


Figure 3.10: Normal probability plot for distribution of the determinant sensitivity (for the constraint $g_{2,5}$ =*0.6*) illustrating that the uncertainty in the determinant sensitivity appears normally distributed. This is based on a Monte Carlo simulation of 1000 realizations with 500 observations used in each realization (PRBS magnitude is +/- 1 and noise covariance of *I*).

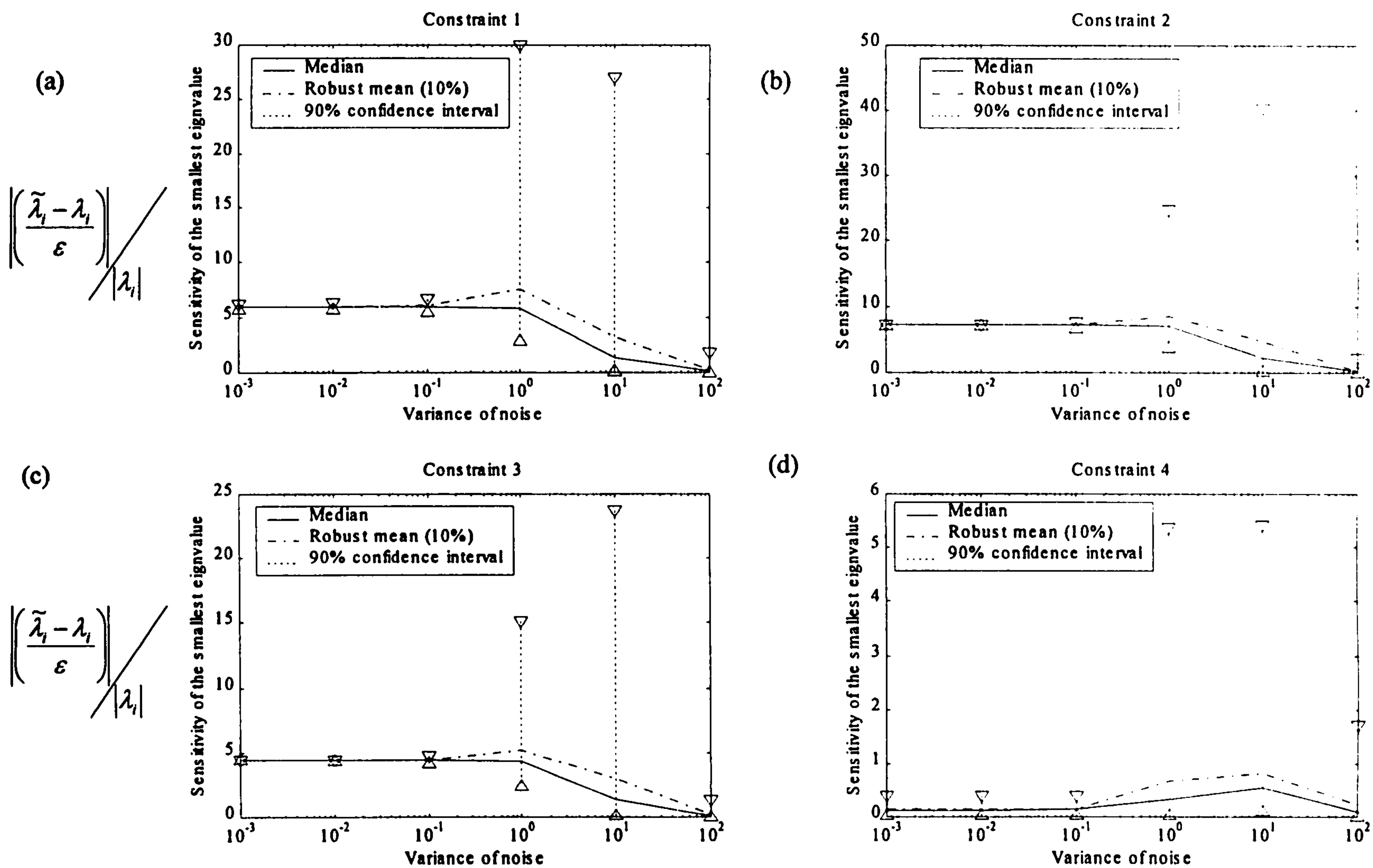


Figure 3.11: The effect of variance of noise on sensitivity of the smallest eigenvalue to changes in the constraint. Each point on the plot is a result of a Monte Carlo style simulation with 1000 realization (PRBS magnitude is +/- 1 and 500 observation in each realization). *(a)* $g_{2,5} = 0.6$ *(b)* $-0.54g_{2,4} + 0.84g_{2,5} = 0.099$ *(c)* $g_{2,4} - 1.25g_{2,5} = 0$ *(d)* $g_{4,3} = 0$

a higher variance in the estimated gain matrix parameters, which in turn increase the confidence interval of the estimated eigenvalue. Second, as the signal-to-noise ratio decreases the uncertainty in the estimated eigenvectors increases which results in some non-linear behavior for the sensitivity of the smallest eigenvalue. Figure 3.11 shows numerically, the effect of signal-to-noise ratio on the sensitivity of the smallest eigenvalue for the constraints in Example 3.1. The effect of non-linearity due to larger uncertainty in the eigenvectors can be seen in Figure 3.11, when the variance of noise is larger or equal to 10. The sensitivity of the eigenvalues to changes in the constraint can also be evaluated theoretically for the case when the variance of the added noise is 0 (Appendix 9). This is accomplished by substituting (3.12) into (3.25). From Appendix 9, it can be seen that the theoretical results for when the variance of the noise is 0, match the numerical results for when the signal-to-noise ratio is high, well in Figure 3.11. Similarly, Figure 3.12 illustrates the effect of signal-to-noise ratio on the sensitivity of the determinant. This demonstrates that as the signal-to-noise ratio decreases the uncertainty in the determinant increases.

Figure 3.11 also illustrates that the sensitivity of the smallest eigenvalue decreases when the variance of noise is higher than 1. This implies that when the signal-to-noise ratio is very small the changes in the constraint have little effect on the smallest eigenvalue. This in turn implies an obvious situation, that when the data quality is very poor the prior knowledge has little effect in improving the quality of the solution.

The important issue to note from this simulation study is that the uncertainty in the sensitivity analysis remains low when the signal-to-noise ratio is high. For this particular simulation study, when the variance of the noise is less than 1, which corresponds to the following signal-to-noise ratio:

$$\frac{\sigma_{(Signal)}}{\sigma_{(Noise)}} = \begin{pmatrix} 14.2 & 4.3 & 10.3 & 5.5 & 13.5 \end{pmatrix}$$

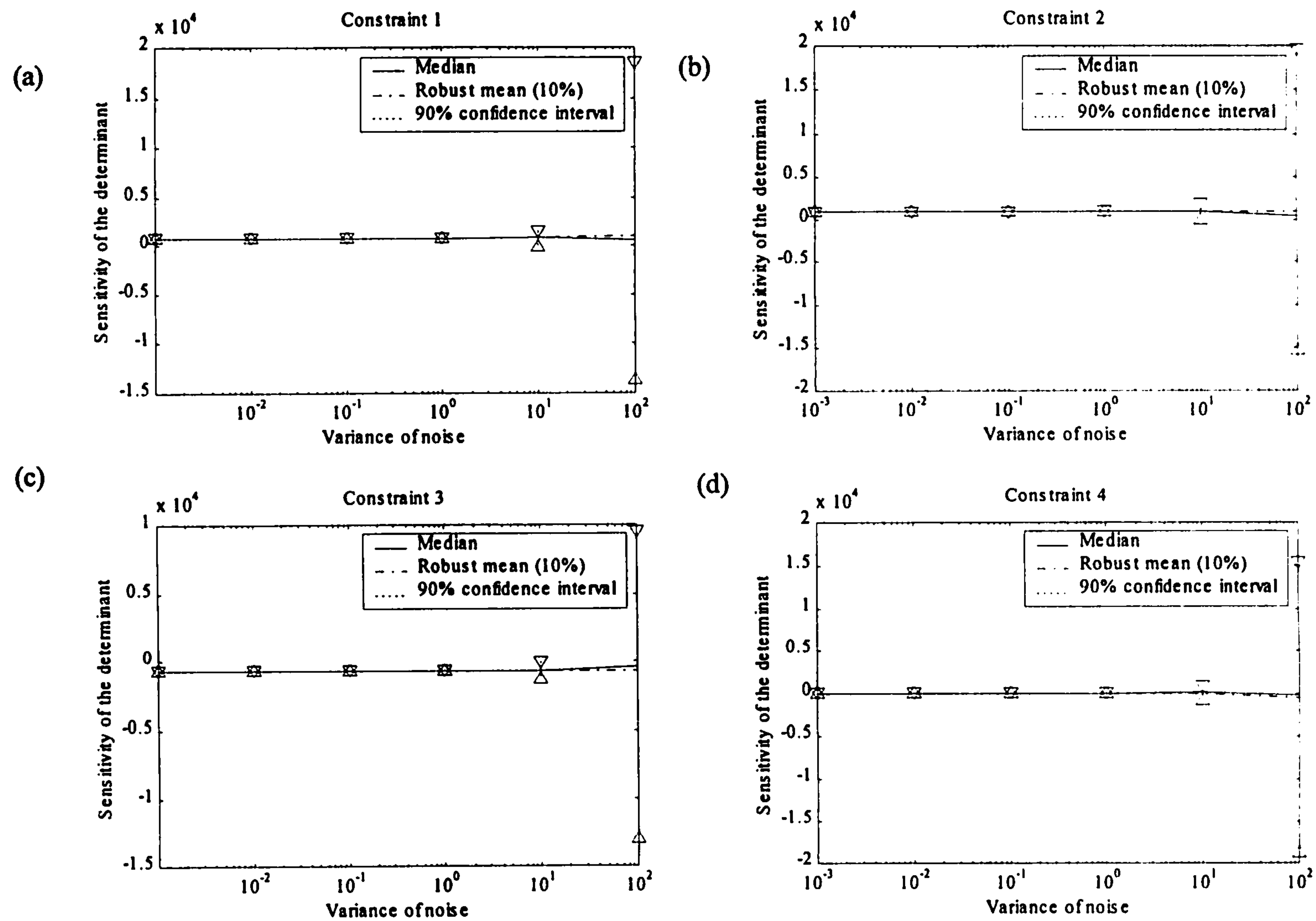


Figure 3.12: The effect of variance of noise on sensitivity of the determinant to changes in the constraint. Each point on the plot is a result of a Monte Carlo style simulation with 1000 realization (PRBS magnitude is +/- 1 and 500 observation in each realization). *(a)* $g_{2,5} = 0.6$ *(b)* $-0.54g_{2,4} + 0.84g_{2,5} = 0.099$ *(c)* $g_{2,4} - 1.25g_{2,5} = 0$ *(d)* $g_{4,3} = 0$

The uncertainty in the sensitivity of the smallest eigenvalue remains low. As a result the probability that the ranking of the constraints based on the sensitivity of the smallest eigenvalues to errors in constraint will be wrong, is fairly low (less than 5%). Consequently this measure of model sensitivity is an appropriate metric in chemical processes.

Table 3.2: The sensitivity of the determinant for the prior knowledge in Example 3.1 based on Monte Carlo style simulation with 1000 realizations (the first value is the mean followed by the 95% confidence interval)

Type of Prior Knowledge	$\frac{d\det(\hat{G})}{dc}$
$g_{2,5} = 0.6$	829.1 ± 126.6
$-0.54g_{2,4}+0.84g_{2,5} = 0.099$	1027.8 ± 129.8
$g_{2,4} - 1.25g_{2,5} = 0$	-641.8 ± 80.48
$g_{4,3} = 0$	-12.41 ± 126.6

Table 3.3: The sensitivity of the eigenvalues for the prior knowledge in Example 3.1 based on Monte Carlo style simulation with 1000 realizations (the first value is the mean followed by the 90% range for each estimate)

Type of Prior Knowledge	λ_i	$\frac{\tilde{\lambda}_i - \lambda_i}{\varepsilon}$	$\left\lvert\left(\frac{\tilde{\lambda}_i - \lambda_i}{\varepsilon}\right)\right\rvert / \lvert\lambda_i\rvert$
$g_{2,5} = 0.6$	-0.3786 (-0.5997, -0.1352)	2.3962 (1.6012, 3.6002)	11.6370 (2.8316, 25.1900)
	0.7979 + 2.5502i (0.8377 + 2.1862i, 0.8308 + 2.8675i)	-1.1355 - 2.4148i (-0.6978 - 1.8760i, -1.1957 - 3.3018i)	1.0183 (0.6755, 1.4959)

	0.7979 - 2.5502i (0.8377 - 2.1862i, 0.8308 - 2.8675i)	-1.1355 + 2.4148i (-0.6978 + 1.8760i, -1.1957 + 3.3018i)	1.0183 (0.6755, 1.4959)
	5.3553 (5.0854, 5.6091)	1.5287 (1.0984, 2.0370)	0.2849 (0.2070, 0.3726)
	7.1232 (6.9294, 7.3088)	-1.6523 (-1.3359, -2.0497)	0.2324 (0.1839, 0.2936)
$-0.54g_{2,4}+0.84g_{2,5} = 0.099$	-0.3808 (-0.5722, -0.1887)	2.8168 (2.0291, 3.8497)	9.5937 (3.7183, 19.5961)
	0.7991 + 2.5492i (0.7870 + 2.2667i, 0.7698 + 2.8298i)	-1.0829 - 3.3215i (-0.8219 - 2.6594i, -1.4991 - 4.1898i)	1.3247 (0.9471, 1.8367)
	0.7991 - 2.5492i (0.7870 - 2.2667i, 0.7698 - 2.8298i)	-1.0829 + 3.3215i (-0.8219 + 2.6594i, -1.4991 + 4.1898i)	1.3247 (0.9471, 1.8367)
	5.3590 (5.0756, 5.6402)	0.8103 (0.3374, 1.3532)	0.1502 (0.0648, 0.2428)
	7.1193 (6.9025, 7.3085)	-1.4607 (-1.1656, -1.8502)	0.2057 (0.1600, 0.2659)
$g_{2,4} - 1.25g_{2,5} = 0$	-0.3801 (-0.5697, -0.1833)	-1.7497 (-1.2565, -2.3896)	6.4121 (2.3059, 12.5066)
	0.7988 + 2.5485i (0.8012 + 2.2648i, 0.7321 + 2.8475i)	0.6476 + 2.1065i (0.5300 + 1.6717i, 0.6660 + 2.7287i)	0.8358 (0.6012, 1.1566)
	0.7988 - 2.5485i (0.8012 - 2.2648i, 0.7321 - 2.8475i)	0.6476 – 2.1065i (0.5300 - 1.6717i, 0.6660 – 2.7287i)	0.8358 (0.6012, 1.1566)
	5.3597 (5.0742, 5.6469)	-0.4009 (-0.1220, -0.7453)	0.0746 (0.0233, 0.1325)
	7.1185 (6.8997, 7.3143)	0.8552 (0.6785, 1.0934)	0.1205 (0.0931, 0.1576)
$g_{4,3} = 0$	-0.3637 (-0.6313, -0.0266)	-0.0698 (-0.0012, -0.5388)	1.4180 (0.0329, 5.6363)

	0.7907 + 2.5436i (0.6524 + 2.1706i, 0.8591 + 2.9263i)	-0.0971 + 0.0892i (-0.0975 + 0.0365i, 0.0313 + 0.3020i)	0.0704 (0.0384, 0.1356)
	0.7907 - 2.5436i (0.6524 - 2.1706i, 0.8591 - 2.9263i)	-0.0971 - 0.0892i (-0.0975 - 0.0365i, 0.0313 – 0.3020i)	0.0704 (0.0384, 0.1356)
	5.3626 (5.0753, 5.6601)	-0.0977 (0.0119, -0.2297)	0.0206 (0.0022, 0.0440)
	7.1154 (6.8556, 7.3513)	0.3641 (0.2579, 0.4703)	0.0512(0.0364, 0.0662)

3.5. Conclusions

In this chapter, the sensitivity of CSC to errors in prior knowledge was studied. This is accomplished by performing two separate sensitivity analyses. First, the sensitivity of the gain matrix to errors in prior knowledge is evaluated. Next, the sensitivity of the CSC to changes in the gain matrix is assessed. The combination of these two sensitivity analyses provides the sensitivity of the CSC to error in prior knowledge.

Since the CSC is an uncheckable criteria in practice, two different checkable criteria are devised which can evaluate the sensitivity of CSC to error in prior knowledge ((3.23) and (3.25)). The first methodology propagates the uncertainty of the estimated model and the error in the prior knowledge to the determinant of the gain matrix. The second method observes the sensitivity of the smallest eigenvalue to changes in the prior knowledge. Using these metrics a practitioner may evaluate the sensitivity of the controller design to error in prior knowledge before implementing the control scheme (this was preformed in Example 3.1).

The metrics used to evaluate the model sensitivity can also be used to suggest to the practitioners which prior knowledge would be most useful. The process engineer can then attempt to determine knowledge about the parameters via other means. If such a prior knowledge is found it could be used to improve the model quality.

Chapter 4
Model Maintenance

4.1. Introduction

At the center of any MPC (model predictive control) application is the estimated model (Qin and Badgwell 1997). This model is usually based on experimental data collected during the identification stage. Hence, the performance of the MPC depends greatly on the quality of the data collected during the identification phase. Some of the qualities of good data sets are: high signal-to-noise ratio, large data sets, and well designed input signals. The quality of the estimated model depends on all of these factors. Due to process limitations, operational and quality constraints, ideal experimental conditions are rare. Non-ideal situations, such as short data sets and low signal-to-noise ratio, result in poorly estimated models. One of the problems that arises due to poor modeling is ill-conditioning of the model (or sub-systems of the model). Such an ill-conditioning could be a natural occurring problem that is system specific (such as two adjacent tray temperatures in a distillation column) (Sagfors and Waller 1995). However, it could also be a manifestation of a poor design of experiment. This problem becomes more evident in larger systems with high sparsity, since in such systems more sub-systems are naturally ill-conditioned. Larger systems with high sparsity are a common occurrence in chemical processes: for example, Hoffman (2000) mentions that in a typical highly interactive fluid catalytic cracking unit (FCCU) approximately 80-85% of the transfer functions are zeros, and Qin et al. (1997) reported MPC applications on systems as large as 603x283 (603 inputs and 283 outputs).

It is hard to estimate whether or not the ill-conditioning of the model is due to poor data or inherent process characteristics, unless a comprehensive study is performed. It appears, based on private communications with practitioners, that perhaps both situations are quiet common. For small systems or sub-systems (2x2, 3x3, and maybe 4x4) is it possible to detect natural ill-conditioning based on engineering knowledge (Hoffman 2000). In larger systems, which have exponentially more sub-systems, it is rare to be able to use engineering knowledge to understand detect all the modeled ill-conditioning.

Once the ill-conditioning has been detected in the model, there are few methods of handling it. Most practical applications by different vendors such as DMC (Dynamic Matrix Control, Cutler and Ramaker 1980), RMPCT (Robust Model Predictive Control Technology, MacArthur 1996), and SMC-Idcom (Shell Multivariable Control-Identification-COMmand, Yousfi and Tournier 1991) remove the ill-conditioning by setting the smallest singular value of the model equal to zero in a variety of means. This procedure is called Singular Value Thresholding (SVT) (MacArthur 1996, Aoyama et al. 1997, and Qin et al. 1997). In SVT, an arbitrary threshold is chosen for the singular values, and if the singular value is lower than this value, it is set to zero. The RMPCT, which is a Honeywell product, is a direct extension of SVT (MacArthur 1996, and Qin et al. 1997). The information about how the threshold value is determined is not mentioned in the literature, so perhaps this material is proprietary. In RMPCT (as well as SMC-Idcom) the ill-conditioning is checked after each controller action. If the sub-matrix (or the reduced system), defined by the variables not at a contraint is ill-conditioned, then the singular values below a certain threshold are ignored and the gain matrix is re-estimated based on the other singular values. In the case of SMC-Idcom, when a high condition number is detected in the sub-matrix, the controller eliminates the low priority CVs until a well-conditioned sub-matrix remains. In controllers that use move suppression technologies (such as DMC and OPC), the move suppression on the input is increased. This results in a better conditioning of the problem in a similar fashion as ridge regression does in least square problems (Shridhar and Cooper 1998). Most vendors use a combination of these three approaches (SVT, prioritizing CVs, and move suppression) to deal with ill-conditioning problems as the constraint set changes.

A slightly different approach to the problem of ill-conditioning was also mentioned by Hoffman (2000); based on my communication with control engineers, this method appears to be common. In this method, the ill-conditioning is removed by "modifying the gains slightly to eliminate one rank" (Hoffman 2000). This modification in the gain matrix elements is performed without any regard to the uncertainty in the gain elements. Although this is perhaps practical for smaller systems (smaller than 4x4), it would be rather difficult to change the gain elements appropriately in larger systems (in his paper he considers a 4x3 system).

The problem of ill-conditioning is two-fold: first, the ill-conditioning has to be detected, then it has to be dealt with in an appropriate manner. The detection of the ill-conditioning is an art in itself. Belsley (1991) suggests 11 different methods of detecting ill-conditioning. Perhaps the most common method of detecting ill-conditioning is the condition number. The condition number of a gain matrix is scale dependent (Brambilla and D'Elia 1992). Unless a suitable scaling is used, the condition number will have no physical meaning, and a high condition number would reflect on numerical characteristics of the estimated model instead of the fundamental process model.

RMPCT uses scaling to improve the systems condition number (MacArthur 1996). A similar method was reported by others (Skogestad and Morari 1988, Grosdidier 1985, and Nguyen et al. 1988) and termed "Minimum Condition Number." In RMPCT, the scaling is performed at the design phase and is based on the following minimization:

$$\gamma^{*} = \min_{D_Y, D_X} \left\{ \kappa\left(D_Y \hat{G} D_X\right) \right\} \tag{4.1}$$

where D_Y and D_X are the scaling of the output and the input respectively. They are diagonal matrices.

κ is the condition number

γ^{*} is the minimum condition number. An estimate of the minumum condition number (Grosdidier et al. 1985):

$$\gamma^{*}\begin{cases} = \|\Lambda\|_1 + \sqrt{\|\Lambda\|_1^2 - 1}, \text{ where } G \text{ is } 2\times 2 \\ \leq 2\max\left[\|\Lambda\|_1, \|\Lambda\|_\infty\right], \text{ where } G \text{ is larger than } 2\times 2 \end{cases}$$

Λ is the RGA (relative gain matrix) of the gain matrix, $\Lambda = G \otimes \left(G^{-1}\right)^T$ where $\otimes$ denotes element by element multiplication

There is no analytical solution to the above optimization problem. If the dimension of G is $n\times n$, $2\times n$ optimization variables must be solved simultaneously. Although an optimization method will give the best condition number, it may require significant time to solve in large systems. Consequently, iterative schemes that will analytically reduce the condition number via scaling have been devised (Nguyen et al. 1988).

While such a scaling would certainly result in a better conditioning of the gain matrix, it has no effect on the probability of the control system being unstable (see Appendix 13). By scaling the gain matrix, the uncertainties in its elements are also scaled. This results in the scaling of the eigenvalue and the uncertainty in the eigenvalue; however, the probability that an eigenvalue will change sign will not change. This is similar to the fact that RGA does not change with scaling (Grosdidier et al. 1985).

Furthermore, such a scaling would only improve the condition number of the entire gain matrix and not necessarily the sub-matrices in the gain matrix. Checking for ill-conditioning in all the sub-systems that may be encountered during operations requires a large number of condition number evaluations (Qin et al. 1997). To check the condition number for all the sub-systems of an *mxm* system the number of condition number evaluations is given by:

$$k''_{\max} = \sum_{i=2\cdots m} \binom{m}{i}, m \geq 2 \tag{4.2}$$

Where k''_{max} is the maximum number of condition numbers to be evaluated. For a 10x10 system, this results in evaluation of 1.83×10^5 condition numbers. In order to eliminate the need to analyze the many different sub-systems and the possibility that a system may go unstable when constraints become active, some authors (Zheng and Morari 1995; Zafiriou and Chiou 1996; Zafiriou 1990) have suggested the replacement of hard constraints in the optimization problem with soft constraints. The disadvantage of this method is that the resulting controller is overly conservative, and it may be infeasible to implement such a scheme for larger systems.

It is important to distinguish between practical and theoretical solutions to the problem of ill-conditioning. The discussion so far has been based on current practices in industry (Hoffman 2000, Qin et al. 1997, and MacArthur 1996); such methods have been practical and easy to implement. Some other solutions to the problem of ill-conditioning were presented by Featherstone et al. (1998) and Maurath et al. (1988).

Featherstone (1998) considered the problem of controlling a paper machine (with dimensions of 101x101 for a detailed description of the system, see Featherstone 1997). In this case, the uncertainty in the gain matrix was propagated to the singular values of the gain matrix. This was accomplished by assuming that the eigenvectors associated with the gain matrix did not have any uncertainty; hence, the only uncertainty was in the eigenvalues of the gain matrix. In addition, it was assumed that the singular values were normally distributed. Based on those assumptions, a confidence interval for each singular value was estimated. If the confidence interval (for a singular value) included zero, then it was assumed that the process could not be controlled in that direction, and a new gain matrix with that singular value set to zero was produced (in this particular paper he used a pseudo-SVD controller, Moore 1986). This idea is very similar to the SVT ideology that some vendors use (RMPCT and SMC-Idcom). There is also a similarity between this work and the general class of pre- and post-compensators, more specifically SVD controllers (Skogestad and Postlethwaite 1996, Zhu and Jutan 1998, Lau et al. 1985). A typical form of a pre- and post-compensator controller is (Skogestad and Postlethwaite 1996):

$$K_C(s) = W_1 K_{CS}(s) W_2 \tag{4.3}$$

where $K_C(s)$ is the controller in the s-domain

K_{CS} (s) is a diagonal control matrix

W_1, and W_2 are the pre- and post-compensators

SVD-controller is a specific type of pre- and post-compensator controller in which:

$W_1 = V$ and $W_2 = U^T$, where the SVD of G is $G = U\Sigma V^T$

In the case of Featherstone (1998) the matrix K_{CS} can actually be a non-diagonal matrix that is not full rank.

Perhaps a more sophisticated solution to this problem is by Maurath et al. (1988), in which PCA is performed on the A matrix, which is the "Dynamics Matrix" in DMC (Cutler et al. 1980). This results in removing the ill-conditioning not only of the gain matrix (which corresponds to zero frequency) but also at other frequencies.

The commonality between all of these methods, whether practical or academic in nature, is to remove ill-conditioning by either removing the uncontrollable direction (i.e., set the smallest eigenvalue to zero) or by de-tuning that direction significantly. Both those solutions are too conservative, since they will result in either a de-tuned controller or one that cannot control in a certain direction. A different approach to this problem is not to remove the low-gain direction but to better estimate the low-gain direction of the model (Koung and MacGregor 1994). In this approach, contrary to other approaches mentioned, it is assumed that the low-gain direction (smallest eigenvalue) is not zero and that there is an advantage for it to be used by the controller. This may be the case if the directionality of the disturbance is in the low-gain direction of the system or if the LP (which may be operating on the top of the controller) sees an advantage in operating in that direction (this will be illustrated in Example 4.4). In the paper by Koung and MacGregor (1994), the emphasis is on the design of experiment for estimating the low-gain direction. In particular, the low direction is re-estimated and the new estimated model is used for control in the full space. In their work, they consider the problem of controlling a 2x2 high-purity distillation column (which is physically ill-conditioned) and a 3x3 system (although their methodology is not limited to this) in the full space.

In this chapter, the purpose is to decide if the gain direction should be removed (i.e., the eigenvalue should be set to zero) or the gain direction should be changed (i.e., flip the sign of the eigenvalue). To keep in tune with the rest of this thesis, the method proposed is to use posterior knowledge about the controller performance. Simply stated, if a MPC controller is unstable, then at least one of its eigenvalues has an incorrect sign. The approach of this chapter is to use this type of posterior knowledge to re-estimate the model and fix up ill-conditioning problems. Contrary to the previous chapters where the prior knowledge is known before the model is estimated, in this chapter the prior knowledge is gained (i.e., there is a posterior knowledge, or run-time knowledge) based on the controller performance. Furthermore, in the previous chapters, it was assumed that there are no bounds on the inputs or the outputs. However, the problem addressed in this chapter is often caused by the fact that there are bounds (or constraints) on the inputs and outputs. Therefore, acquired posterior knowledge is dependent on which set of bounds are active. The main contribution of this chapter will be in developing methods to maintain models after they have been implemented in a control structure.

The advantage of utilizing the methods of this chapter is that it provides an estimate for the smallest eigenvalue with no need for more plant experimentation. This is accomplished by including the posterior knowledge about the controller performance in the model re-estimation phase. This estimate of the low-gain direction may not be very accurate, but the accuracy of this estimate should be sufficient to produce a stable controller with little effort. This will have the added advantage of providing control with an extra dimension compared to the more traditional approach of SVT and SVD controllers, which are currently used in practice. The controller performance may not be as good as the controller performance obtained by estimating the smallest eigenvalue more accurately via a sequential design, as suggested by Koung and MacGregor (1994), which would require sophisticated design of experiment with more plant experiments to be conducted. In essence, the methodology of this chapter is halfway between sequential experimental design of Koung and MacGregor (1994) and the SVT algorithm used in most commercial packages. Furthermore, if the controller design based on the incorporation of the posterior knowledge about the controller performance fails (i.e., the re-estimated model still produces an unstable controller) then the methodology of this chapter is to either perform a sequential design (Koung and MacGregor 1994) or remove the low-gain direction from the model.

4.2. Example of Problem

Suppose that a process is linear and its true model is given by (4.4). This non-square plant, with 3 inputs and 2 outputs, is well-conditioned (condition number of the gain matrix is 4.85).

$$G(s) = \frac{e^{-1s}}{10s+1}\begin{pmatrix} 0.3 & -0.25 & 0.6 \\ 0.5 & -0.3 & 0.4 \end{pmatrix} \tag{4.4}$$

This example transfer function is constructed of a scalar dynamic model multiplied by a constant matrix (the gain matrix). Many examples of such transfer function matrices appear in chemical engineering, including the simplified distillation column studied by many different researchers (Skogestad et al. 1988, 1988, Pensar et al. 1993, and

Grosdidier et al. 1985). Consider the following two estimates of this process, which resulted from two different identification studies:

$$\hat{G}_1(s)=\frac{e^{-1s}}{10s+1}\begin{pmatrix}0.27 & -0.17 & 0.61\\ 0.61 & -0.39 & 0.30\end{pmatrix} \tag{4.5}$$

$$\hat{G}_2(s)=\frac{e^{-1s}}{10s+1}\begin{pmatrix}0.43 & -0.42 & 0.55\\ 0.52 & -0.20 & 0.42\end{pmatrix} \tag{4.6}$$

For simplicity it is assumed that prior knowledge about the process dynamics is available; hence, there are no model mismatches in the dynamic portion of the models. The additive steady-state model mismatch between the two estimated models and the true process (measured by the F-norm) is:

$$\left\|\hat{G}_1-G\right\|_F=0.194 \tag{4.7}$$

$$\left\|\hat{G}_2-G\right\|_F=0.243$$

Note that the estimated model (4.5) has smaller model error compared to (4.6). Next, assume that a multivariable controller (such as DMC) is designed based on these models. Assuming (4.4) is the true model, the responses of the closed-loop system to a sequence of set-point changes are simulated in Figures 4.1, 4.2, and 4.3 (the DMC tuning strategy of Shridhar and Cooper 1998 is used for the control horizon tuning, *M*, and the prediction horizon tuning, *P*). In all the cases, the DMC used had an input horizon (*M*) of 5 and an output horizon (*P*) of 30. The weight matrix for both the input and the output was the identity matrix (*I*). In Figures 4.1(a), 4.2(a), and 4.3(a) there were no constraints on the

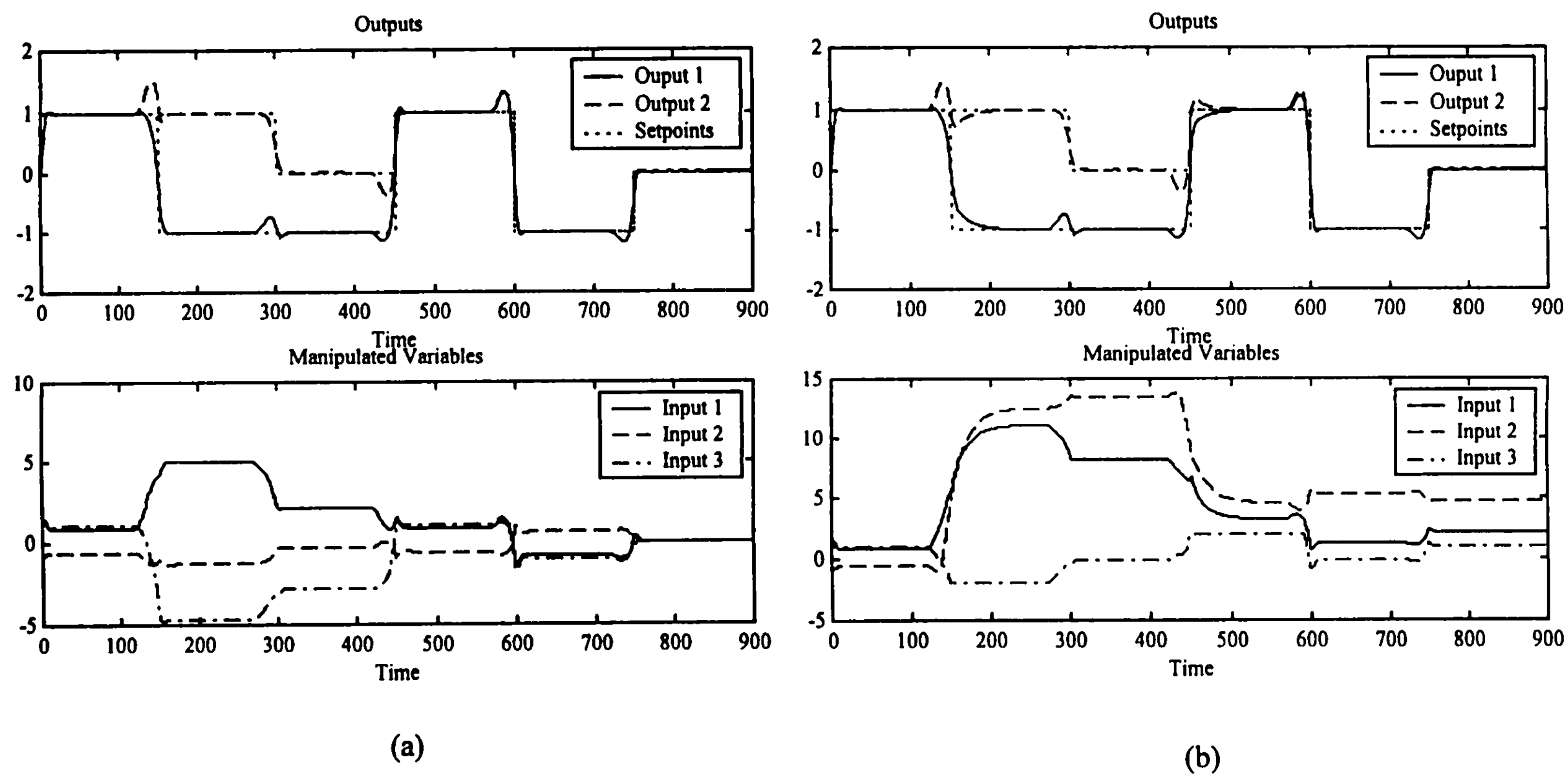
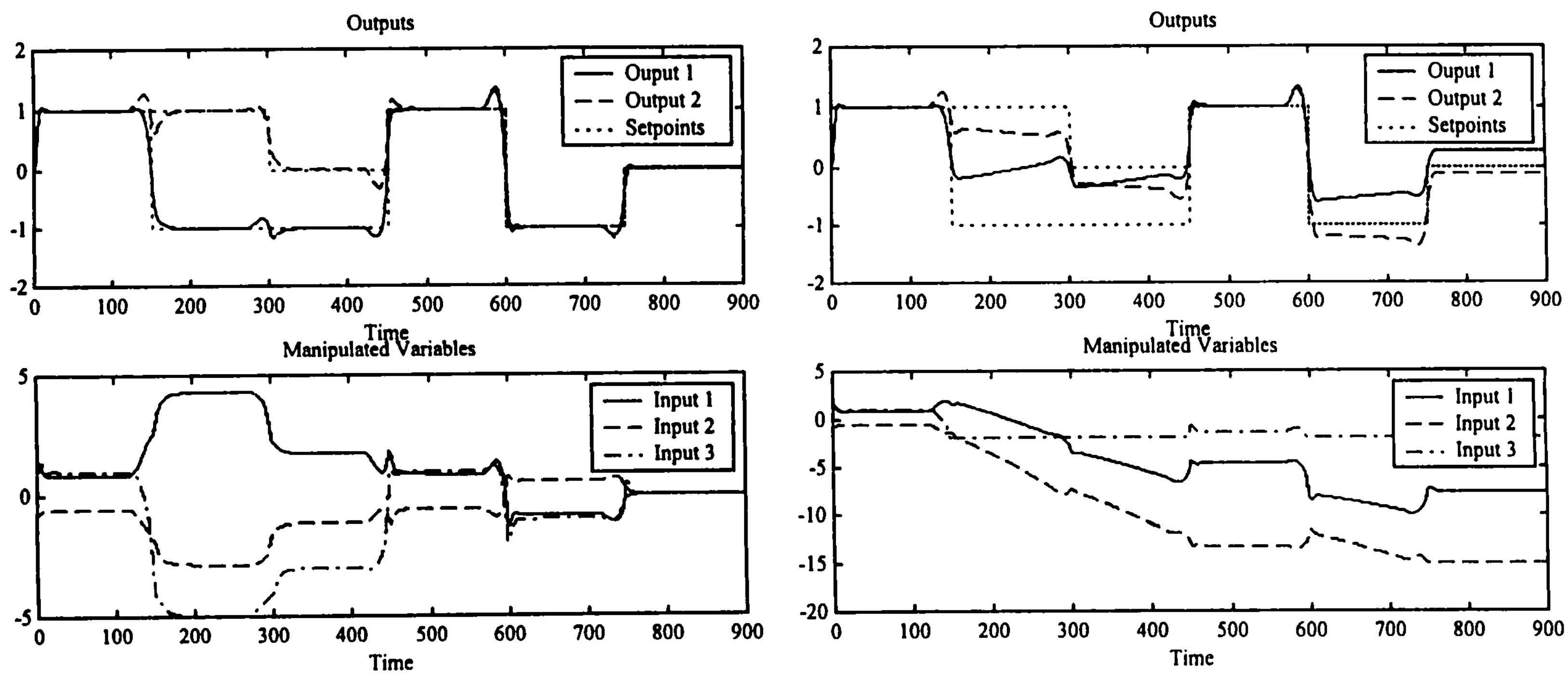


Figure 4.1: Controller performance using the true transfer function as the model (a) with no constraints (b) with constraint of +/- 2 on the 3rd input and +/- 15 on the other inputs



(a) (b)

Figure 4.2: Controller performance using the first estimated model (a) with no constraints (b) with constraint of +/- 2 on the 3rd input and +/- 15 on the other inputs

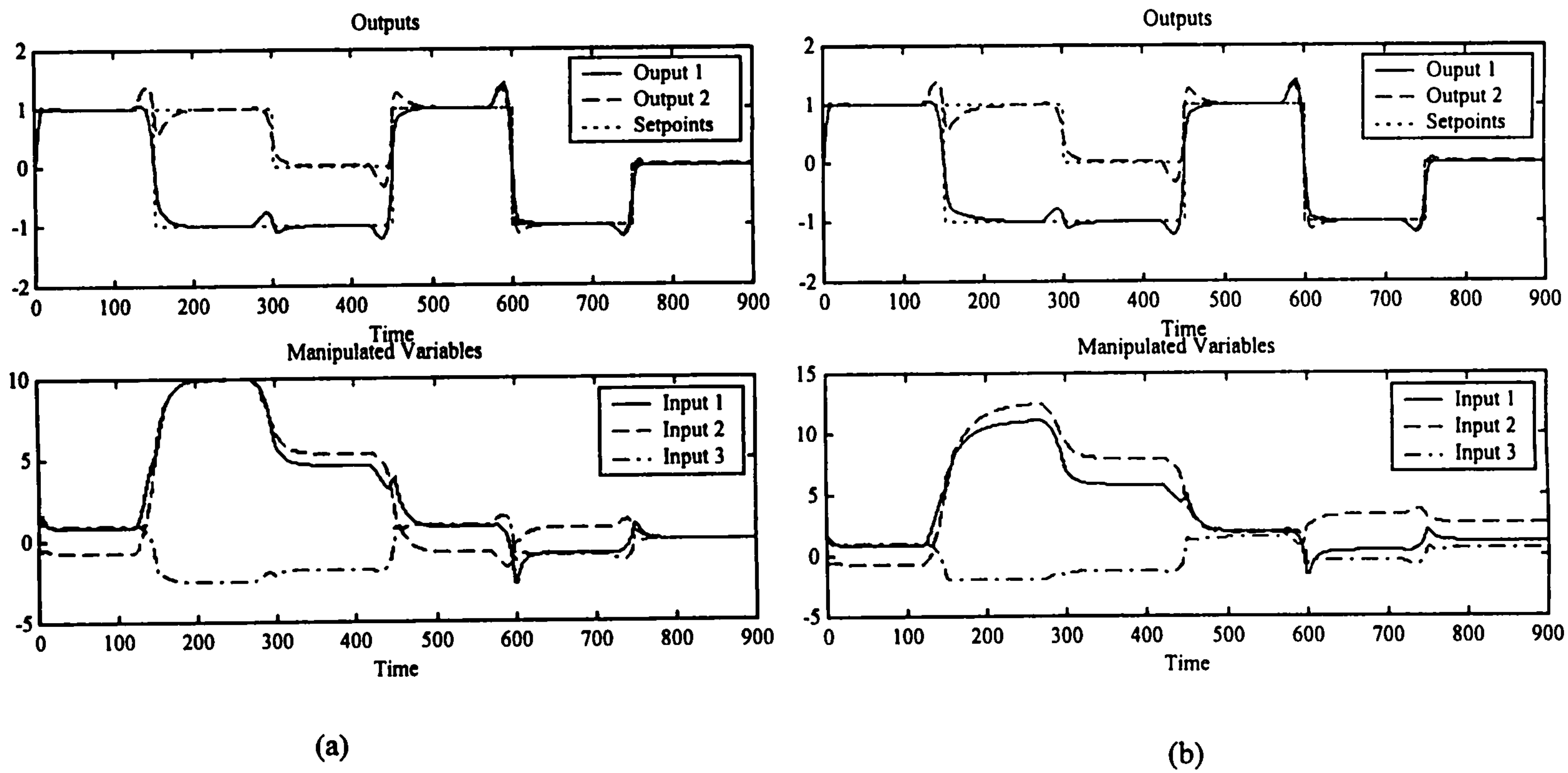


Figure 4.3: Controller performance using the second estimated model (a) with no constraints (b) with constraint of +/- 2 on the 3rd input and +/- 15 on the other inputs

inputs or the outputs; however, in Figures 4.1(b), 4.2(b), and 4.3(b) input 3 was constrained to be between –2 and +2 (other inputs were bounded to ± 15).

Although the two estimated models have similar F-norm errors (4.7), their controller performance is significantly different. The second estimated model (4.6) results in a controller that is stable independent of whether or not there is a constraint active (Figure 4.3). The first estimated model (4.5) results in a stable controller when there is no active constraint (Figure 4.2(a)). In Figure 4.2(b) there is an active constraint for input 3 at different time intervals. During those intervals, the controller is unstable; however, at other times the controller may become stable. From Figures 4.2 and 4.3, it can be seen that although the two estimated models had similar additive errors (4.7), the resulting controllers were significantly different. This result indicates that the closeness between the identified and true process models cannot be judged based on the mismatch magnitude alone. It will be shown that the sub-systems of (4.5) do not meet the controller stability criteria (CSC) of Garcia and Morari (1985). Therefore, the resulting closed-loop system will be unstable, independent of controller design. In addition, such instability is only visible when certain constraints become active. It will be shown that such operational knowledge about controller stability may be used as posterior knowledge in model re-estimation to produce a new model that results in a stable controller.

4.3. Using Controller Stability as Posterior Knowledge in Model Identification

It is very difficult in ill-conditioned systems to accurately estimate the smallest eigenvalue (or singular value in the case of a non-square system) with sufficient accuracy, using traditional experimental design methods (Andersen et al. 1989, Koung et al. 1994). This has resulted in two areas of research. First, as mentioned earlier, there is an emphasis on SVD-style controllers, where the smallest singular value (or eigenvalue) is set to zero (Featherstone 1997, Featherstone et al. 1998, Hoffman 2000, Hovd et al. 1996, MacArthur 1996, Maurath et al. 1988, Moore 1986, Zhu et al. 1998, and others). Consequently, the system will be controlled in a reduced space. This is a conservative design, where the controller has a lower degree of freedom than if it was controlling the full space (this point is illustrated in Example 4.4). One of the consequences of dropping an eigenvalue compared to keeping it, is seen when an LP is running on the top of the controller to optimize the steady-state performance of the plant (Marselle et al. 1982). In order for the process to be at the maximum profit point, the LP should have as many degrees of freedom as possible. Yousfi and Tournier (1991) stated the importance of additional degrees of freedom for the steady state optimization running inside the model predictive control being used in Shell's Multivariable Optimizing Controller (SMOC) (this is also illustrated in Example 4.4). By setting an eigenvalue equal to zero, the degrees of freedom of the optimization problem is reduced by one. Consequently, the process may not be at the optimum point (provided that this particular degree of freedom is involved in the objective function). This result prompted research in the second approach: an experimental design focused on the estimation of the smallest singular value (Koung et al. 1994, and Cooley et al. 2001). While this method will certainly improve the accuracy of the smallest singular value of the full problem, it may not necessarily fix the problems associated with the sub-problem that may arise when constraints become active as the controller is running.

The solution proposed in this chapter is to perform a traditional designed experiment, where all the inputs are perturbed with an un-correlated PRBS under open-loop or closed-loop conditions (MacGregor et al. 1991). Then the model is estimated and is used to implement a MPC. If the controller remains stable throughout its operation, independent of which of the input or output constraints become active, this model satisfies the CSC both in the full space and the different reduced spaces. However, if the system becomes unstable, either initially or after a set of input or output constraints has become active, then the methods proposed in the next few sections can be used to fix up and re-estimate the model.

4.4. The Square problem

Under most situations, the control system operates under some underlying square system, even if the physical process is non-square. This is accomplished by the weighting of the process inputs or outputs or by an optimizer setting set-points on the inputs or outputs and reducing the system. In this section, the premise is that the system being controlled is a square one (non-square systems are discussed in section 4.5). In addition, it is assumed here that the controller has proved to be unstable independent of the controller tuning. Hence, the CSC mentioned in the previous chapters is violated (for detection of USC see Appendix 10). Therefore, at least one of the estimated eigenvalues has the wrong sign. The idea in this section is to use this (posterior knowledge) as a constraint in a parameter re-estimation. It is possible to look at the individual eigenvalues (λ_q) and their confidence intervals (σ_q) to predict which eigenvalues may be the cause of the instability. The sign of the eigenvalues, whose confidence interval include zero, can be changed. This would result in a constraint involving a particular eigenvalue (λ_q) (or a set of eigenvalues) in the parameter re-estimation:

$$\begin{aligned}
&\underset{\hat{\hat{G}}}{Min}\, SSE \\
&s.t. \\
&-\mathrm{Re}\left(\hat{\hat{\lambda}}_q\right)\times sign\left(\mathrm{Re}\left(\hat{\lambda}_q\right)\right) > 0, \left\{\hat{\lambda}_q\right\} = \left\{\hat{\lambda}_i \middle| \mathrm{Re}\left(\hat{\lambda}_i\right) - 2\hat{\sigma}_{\lambda,i} < 0 < \mathrm{Re}\left(\hat{\lambda}_i\right) + 2\hat{\sigma}_{\lambda,i}, \forall i\right\}
\end{aligned} \tag{4.8}$$

where SSE is the Sum of Square Error for all the outputs (i.e.,

$SSE = trace\left(\left(Y - \hat{\hat{G}}X\right)\left(Y - \hat{\hat{G}}X\right)^T\right)$)

$\hat{\lambda}_i$ is the ith eigenvalue of the first estimated gain matrix ($\hat{G}$)

$\hat{\sigma}_{\lambda,i}$ is the standard deviation of the ith eigenvalue ($\hat{\lambda}_i$)

$\hat{\lambda}_q$ is a set of eigenvalues whose confidence interval includes zero

$\hat{\hat{\lambda}}_i$ is the ith eigenvalue of the second estimated gain matrix ($\hat{\hat{G}}$)

Three problems are associated with utilizing (4.8): the optimization problem will be rather complex since the constraint is both non-linear and discontinuous, there will be uncertainty about which set of eigenvalues has caused an unstable system, and it is difficult to match the eigenvalues of the first estimate gain matrix with the eigenvalues of the second estimate gain matrix (i.e., if the eigenvalues are sorted in a sequence, this sequence is not necessarily consistent, this is specially an important issue when two or more eigenvalues have similar magnitudes). Therefore, a similar approach to that used in previous chapters is used here. Instead of changing the sign of individual eigenvalues, the sign of the determinant (which is clearly the product of all the eigenvalues) of the gain matrix is changed (or flipped):

$$\begin{aligned} &\underset{\hat{\hat{G}}}{Min}\, SSE \\ &s.t. -\det\left(\hat{\hat{G}}\right) \times sign\left(\det\left(\hat{G}\right)\right) > 0 \end{aligned} \tag{4.9}$$

As shown in previous chapters, this method will be useful when an odd number of eigenvalues has changed signs. If an even number of eigenvalues has changed signs, the sign of the determinant remains the same and (4.9) will be ineffective.

4.4.1. Monte Carlo Simulations

Monte Carlo simulations have been used extensively by researchers to provide answers to complex problems (Rubinstein 1981). They generally take little time to code, and the algorithm itself does not require a significant amount of memory. Perhaps the most significant computational advantage of Monte Carlo simulations is that they can be implemented as parallel simulations (this was the case with some of the Monte Carlo simulations in this chapter and the next). The disadvantages are that they are slow and determination of convergence is difficult. Figure 4.4 illustrates a Monte Carlo simulation (which is estimating the probability of unstable control system (P(USC))) progress as a function of the number of realizations. It is apparent that as the number of trials (or realizations) increases, the solution accuracy improves. Figures similar to Figure 4.4 were used to assure that all the simulations had reached a satisfactory level of accuracy.

The purpose of the Monte Carlo simulations, in this section, was to estimate the probability of (4.9) not producing a satisfactory model (i.e., the probability that the fix-up would not work). In order to evaluate the effectiveness of (4.9), three different system sizes were considered: 5x5, 10x10, and 20x20. In each realization of the Monte Carlo simulation, the gain matrix elements were determined by random numbers (i.e., producing a random matrix, $G_k(i,j) \in N(0,1),\ \forall i,\ \forall j$). Then a random PRBS signal was used as the process input, while a white noise was added to the process output. Although the gain matrix was chosen randomly, certain characteristics were imposed on the gain matrix: the sparsity of the gain matrix was set to 50% (i.e., at least half of the individual gains where set to zero), the condition number was set to be greater than 1000, the gain matrix and the sub-matrices that would result from activation of constraints were non-singular. This was done to imitate the characteristics of ill-condition gain matrices in chemical processes (Hoffman 2000). It should be noted that this style of Monte Carlo simulation is significantly different from the previous chapters. While in previous chapters the system was unique, in this chapter the system also changes. This was done to assure that the simulation results are not particular to any one system and that they hold in general for any system with these dimensions and characteristics. Due to changes in the system, the Monte Carlo simulations of this chapter required a substantial increase in the number of realizations. Figure 4.5 illustrates a flow chart of the Monte Carlo studies for this chapter.

Table 4.1 illustrates the Monte Carlo simulation results. The percentages of unstable and stable systems were determined based on the number of simulations that

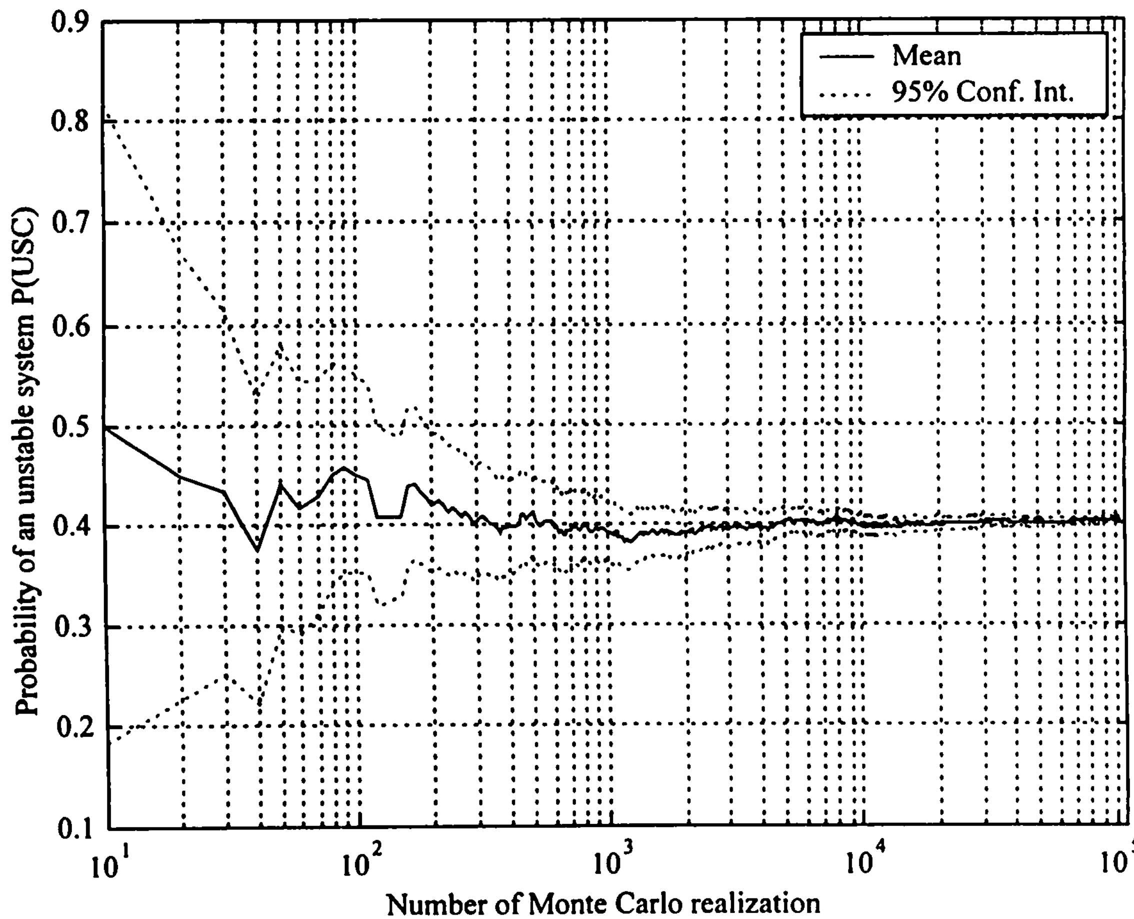


Figure 4.4: Progress of a Monte Carlo simulation. In this case the P(USC) for a random 5x5 system is being estimated (the complete result for this Monte Carlo simulation can be seen in Table 4.1)

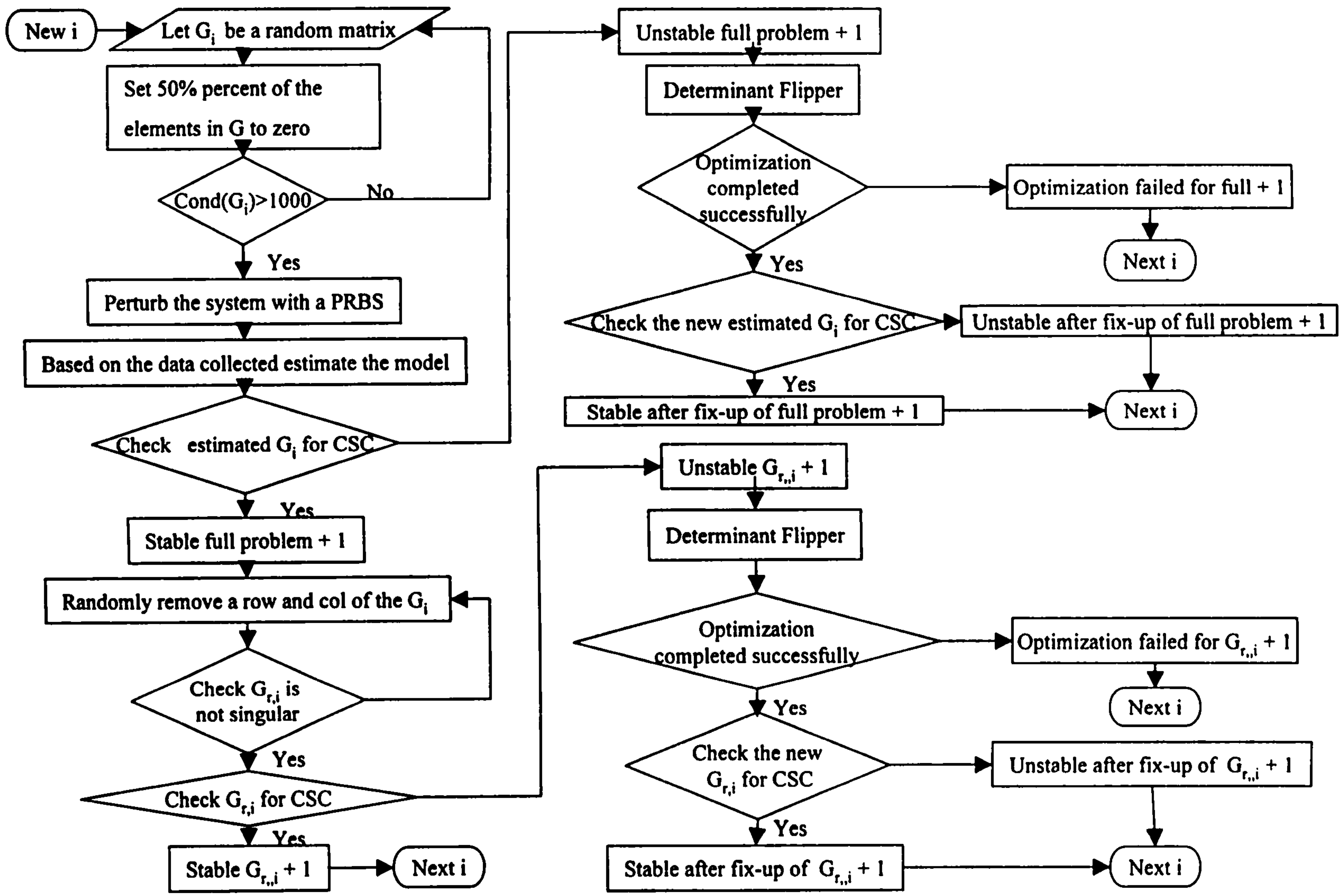


Figure 4.5: A flow chart of the algorithm for the Monte Carlo

failed or passed the CSC respectively. In this case, checking the CSC is possible because these are simulated model identifications and the true transfer function was know. If an estimated model failed the CSC, the model fix-up method of (4.9) was applied to that data set and the model was re-estimated, and the stability of the new model was then re-evaluated. Due to the non-linearity of the constraint in (4.9) and limited computing time, however, the optimization of (4.9) was not always successfully completed (for optimization settings see Appendix 11).

As expected, the probability of unstable systems changes due to changes in the dimension of the system and to the number of data points collected in each trial. For example, for a 5×5 system (under specific conditions listed in Table 4.1) the probability of an UCS is 40%. This number in itself is of little value. The critical issue here is that over 90% of system instability can be fixed with (4.9). In particular, for a 5x5 system 94.5% of the systems with instability where fixed (by flipping the sign of the determinant) with (4.9), while 3% cannot be fixed, and 3% of the simulations have failed (so no judgment can be made about them). If only the simulations that did not fail were considered then the adjusted success rate for the 5x5 systems (with conditions listed in Table 4.1) is 97.0%. Furthermore, the high success rate, which in this case was for the 5×5 system, appears to be independent of the system size.

Although the success rate of over 90% shows little dependence on the system size, it depends greatly on the signal-to-noise ratio and number of data points used in model estimation (for the effect of signal-to-noise ratio on success rate see Table 4.2). The success rate is fundamentally linked to the probability of an odd number of eigenvalues changing sign compared to the probability of an even number of eigenvalues changing sign (P(determinant flipper method success) = 1 – P(even number of eigenvalues changing sign)). The change of sign is defined as a change of sign in the real part of the eigenvalue in this work. In order to get an estimate of this probability, the p.d.f. of the eigenvalues for a random matrix is required. Based on a comprehensive literature review in this area, it was found that for other than special cases, such a distribution could not be estimated analytically. Two special cases were found in which theoretical results for eigenvalue distributions exist. In the first case, the well-known work of Wishart (1928), which has lead to the Wishart distribution (most books in multivariate analysis cover this case), is concerned with the eigenvalue distribution of the covariance matrix. Such matrices by definition are symmetric and semi-positive definite. However, this special case (namely that a gain matrix is symmetric) would rarely occur in chemical processes. The second set of literature on eigenvalue distribution was in the field of random matrices, which is of particular interest to physicists. In this case, the focus is on the eigenvalue distribution of symmetric random Hermitian matrices, where the dimension of the matrix is approaching infinity (see Mehta 1991). Neither of these cases was of relevance to the problem of eigenvalue distribution for gain matrices. Since no theoretical results for eigenvalue distributions were available for cases relevant to chemical engineering, simulations were performed to assess the effect of signal-to-noise ratio on the success rate of (4.9) in producing a SCS (stable control system).

Although no theoretical results for the eigenvalues of random matrices were found, a descriptive analysis can be performed. In a hypothetical case, when no noise is added to the system, the estimated gain matrix will have exactly the same eigenvalues as the true gain matrix (assuming no round-off error). In such a situation, none of the estimated eigenvalues will have the wrong sign (referring to the sign of the real part of the eigenvalue). As more noise is added to the process output (and/or the perturbation in the inputs is decreased), the uncertainty in the gain matrix estimate will increase. In turn, this would result in signs of eigenvalues being estimated incorrectly. As the uncertainty in the gain matrix increases, the number of eigenvalues with an incorrect sign will increase. First (when there is no uncertainty in the gain matrix), there will be no eigenvalues with the wrong sign, then as the uncertainty increases there will be 1, and as the uncertainty increases further, more eigenvalues will change signs. This is illustrated in Figure 4.6 using a simulation study. In this case a 5×5 random matrix was studied, where each element of this matrix is N(0,1) i.i.d., and a white noise of different magnitude is added. This is to represent a true gain matrix (i.e., in this case the 5x5 random matrix) and an estimated gain matrix (i.e., the original 5x5 random matrix plus

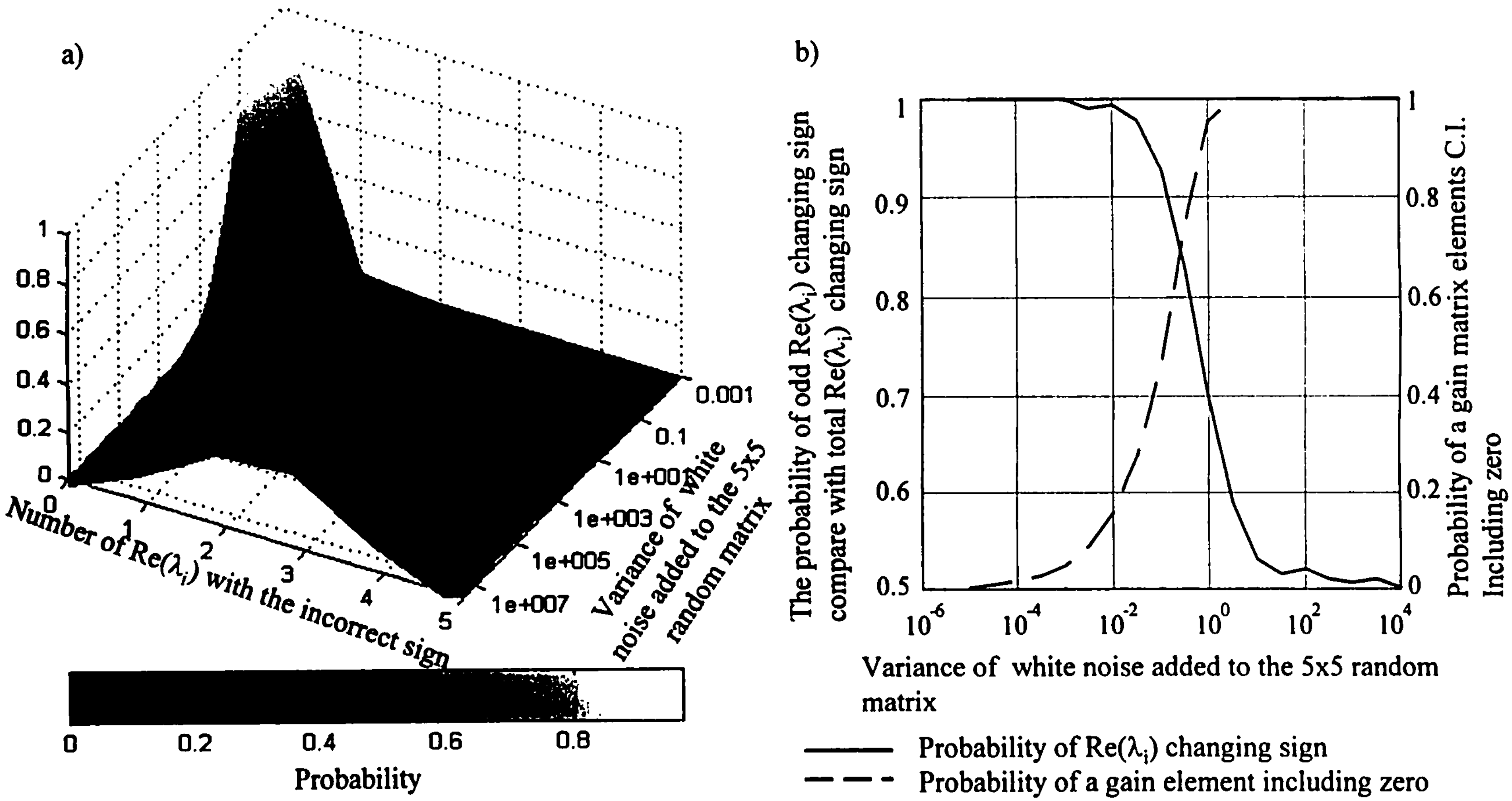


Figure 4.6: The simulation results for both of the graphs are based on 10,000, 5x5 random matrices (with each element being i.i.d. N(0,1)). Where a white noise of different variance is added to the 5x5 random matrix. a) Shows the probability of different number of Re(λ_i), referring to the eigenvalues of the CSC, changing sign as the variance of noise added is increased from 10^{-3} to 10^7. b) shows the probability of odd Re(λ_i) changing sign compare to total Re(λ_i) changing sign in the CSC, as the added white noise increases (this is equivalent as the probability of fix-up method's success). On the right access a measure of model quality is presented to illustrate the (equivalent) quality of model.

another random matrix representing the uncertainty due to noise). Then the eigenvalues are estimated, and the number of eigenvalues with the incorrect signs are counted. Predictably, as more noise is added to the system, the number of eigenvalues with the incorrect signs increases and the probability of odd eigenvalues with the incorrect signs, compared to even eigenvalues with incorrect signs, approaches 0.5. More importantly, when the noise variance is smaller than 0.1 (which is equivalent to 47% of the gain matrix elements having a 95% confidence interval that includes zero) the probability of odd eigenvalues changing signs is greater than 0.8. This implies that at such levels of model uncertainty, re-identifying with (4.9) will result in a SCS with probability of more than 80%. It is the understanding of this author that such levels of model uncertainty are considered poor in chemical processes. Qualitatively speaking, the success rate of producing a SCS using (4.9) will be above 80% for most cases, and in cases where the data quality is reasonable or good, the success rate will be in the 90% range. Furthermore, when the model quality is very poor (i.e., more than 90% of the gain elements have confidence intervals including zero) the probability of success rate using (4.9) approaches 50%. This is due to the distribution of the number of eigenvalues with incorrect sign (illustrate in Figure 4.6 (a)) approaching an asymptotic distribution, where the ratio of odd to even eigenvalues with incorrect sign is about 1. For the rest of this chapter, the success rate of (4.9) in producing a SCS is 94.5% based on the simulations shown in Table 4.1.

4.4.2. Model Fix-up in the Full Dimension

The Monte Carlo simulation results suggest a controller design scheme that could handle ill-conditioned systems without the need for more data collection. In such a scheme, the model is estimated with the available data and a MPC style controller is designed based on this model. The controller is then implemented and its performance monitored. If the controller satisfies the CSC (remember that is an uncheckable condition in practice because we would not know the true transfer function), there exists a tuning for the controller that would result in a stable system (this can be checked in practice by de-tuning the controller, see Appendix 10). If the system is not stabilizable, the process engineer can re-identify the model using (4.9). In effect, the process engineer has utilized the acquired knowledge, that the current control system is unstable, in model re-estimation. As was shown by the Monte Carlo simulation in over 90% of the cases, this would resolve the problem (see Table 4.1). If the resulting system is still unstable, the control engineer can put individual constraints on an even numbers of the eigenvalues changing sign, knowing that odd numbers of eigenvalues are not the cause of UCS. At this stage, the problem is very complex and has similar problems as (4.8). Therefore, it is suggested that if (4.9) does not solve the problem, the process engineer should assume that the process is truly singular and resort to an SVD style controller (Skogestad and Postlethwaite 1996, Zhu and Jutan 1998, Lau et al. 1985). If the resources are available and a full dimension controller is required, model re-identification can be performed, in which the small eigenvalues are specifically perturbed (Koung and MacGregor 1994) and estimated more accurately.

To illustrate this method of model fix-up in the full dimension, consider the following example:

Example 4.1: For the purposes of this example consider a hypothetical 3x3 system (with no dynamics):

$$G = \begin{pmatrix} -0.3165 & -2.4189 & -0.4263 \\ -0.5825 & -2.0410 & 0.2045 \\ -0.9249 & -0.4536 & 1.4856 \end{pmatrix}$$

where the eigenvalues are λ=[0.0029 1.7433 -2.6181] and the condition number is 1436. Three sequences of white noise that are independently and identically distributed (i.i.d.) N(0,1) were added to the process outputs. Consider three different methods of performing the model identification for the purposes of control. In the first case, 80 data points are collected with a traditional PRBS design (Ljung 1999), where the magnitude of the PRBS is ±1 that results in signal-to-noise ratio ($\sigma_{Signal}/\sigma_{Noise}$) of 2.4759, 2.1323, and

1.8078, respectively. In the second case, the sequential experimental design, as illustrated by Koung and MacGregor (1994), was used. In the third case, the method uses the controller stability as a form of posterior knowledge. In all the results shown the confidence intervals are the 95% confidence interval calculated based on the group jackknifing method (which in this thesis will simply be referred to as jackknifing method). Since there are no constraints in the first few cases, the confidence interval have been estimated using both the variance expressions of ordinary least squares (OLS) method and jackknifing method. This provides the mean of comparing the jackknife variance estimator to the variance estimated using OLS. However, to keep consistency with future examples, where there are non-linear constraints, jackknifing will be used in all the cases to estimate the confidence interval. Grouped jackknife works by dividing the total number of observations (n_T) into n_g groups each of size n_h (where $n_T = n_g \times n_h$). Next the estimate based on the ith group of observations removed is calculated ($\hat{\theta}_{-i}$). This results in grouped jackknife variance estimator (Efron and Tibshirani 1993, Shao and Wu 1989):

$$\hat{\sigma}_{\theta}^{2} = \left(\frac{n_g - 1}{n_g} \sum_{i=1}^{n_g} \left(\hat{\theta}_{-i} - \bar{\hat{\theta}} \right)^2 \right)$$

where $\hat{\sigma}_{\theta}^{2}$ is the estimated variance

n_g is the number of groups (in this thesis n_g is 15, it is recommended that this value be larger than the square root of the number of observations, Efron and Tibshirani 1993)

$\hat{\theta}$ is an estimator

$\hat{\theta}_{-i}$ is the ith grouped jackknife replication of $\hat{\theta}$

$\bar{\hat{\theta}}$ is the average of all $\hat{\theta}_{-i}$

Grouped jackknife is very similar to the delete-1 jackknife (which is frequently used), with the advantage that it is computationally less intensive.

The analysis for all the methods is the same for the first half of the identification experiment. Using the first 40 data points, the gain matrix was estimated to be:

$$\hat{G}_1 = \begin{pmatrix} -0.0360 & -2.4887 & -0.5006 \\ -0.4310 & -1.9998 & 0.2936 \\ -0.9885 & -0.5154 & 1.2119 \end{pmatrix} \pm \begin{pmatrix} 0.3601 & 0.4040 & 0.3525 \\ 0.3335 & 0.3591 & 0.3531 \\ 0.4084 & 0.4153 & 0.3812 \end{pmatrix}$$
$$\left\{ \pm \begin{pmatrix} 0.3642 & 0.3568 & 0.3623 \\ 0.3166 & 0.3102 & 0.3150 \\ 0.3441 & 0.3372 & 0.3424 \end{pmatrix} \right\}_{OLS}$$

In the above expression the first matrix is the estimated gain matrix followed by the 95% confidence interval based on group jackknifing. The last matrix is the 95% confidence interval estimated based on the OLS. The accuracy of group jackknife in estimating the confidence interval can be evaluated by comparing the confidence interval based on the OLS to the confidence interval based on group jackknife. Based on the gain matrix estimates ($\hat{G}_1, \hat{G}_2$, and $\hat{G}_{2S}$) and their confidence interval the mean absolute difference between the OLS confidence interval estimate and the jackknife confidence interval is 0.037. This implies an error of about 10% in the confidence interval estimated between jackknifing and OLS.

The individual gain matrix elements (of $\hat{G}_1$) have high uncertainty, with two gain elements confidence interval including zero. This would be expected since the true gain matrix is ill-conditioned and only a small data set is used for model estimation. Note that the application of the CSC would suggest that a controller based on this model would be unstable independent of the controller tuning:

$$\lambda_i\left(G \times \hat{G}_1^{-1}\right) = \{-0.0545 \quad 0.6236 \quad 1.0239\} \not> 0, \forall i$$

Therefore, after the process engineer implements the controller he/she would soon find out that the controller is not stabilizable with its current estimated model (see Appendix 10 for detection of instability). The following are three different methods that he/she may apply:

Method 1: Collecting more data using traditional PRBS design (traditional method)
If 40 more observations are collected in a similar fashion as the first 40 observations, then the estimated model based on the 80 observations is:

$$\hat{G}_2 = \begin{pmatrix} -0.1837 & -2.4816 & -0.5792 \\ -0.4869 & -1.9539 & 0.1828 \\ -1.0267 & -0.3572 & 1.4203 \end{pmatrix} \pm \begin{pmatrix} 0.1840 & 0.2949 & 0.2333 \\ 0.2655 & 0.2361 & 0.3065 \\ 0.2559 & 0.3247 & 0.1952 \end{pmatrix}$$
$$\left\{ \pm \begin{pmatrix} 0.2589 & 0.2569 & 0.2588 \\ 0.2411 & 0.2392 & 0.2410 \\ 0.2646 & 0.2625 & 0.2645 \end{pmatrix} \right\}_{OLS}$$

The new gain estimates have improved, since the confidence intervals are smaller and now only one gain element confidence interval includes zero. Although the model quality has improved, the new model will still result in an unstable control system (UCS):

$$\lambda_i\left(G \times \hat{G}_{2S}^{-1}\right) = \{-0.0502 \quad 0.8088 \quad 1.0609\} \not> 0, \forall i$$

Therefore, after collecting 80 data points, the process engineer may have a better gain estimate; yet, the system remains unstable. More data needs to be collected in order to design a stable control system using this methodology.

Method 2: Sequential D-optimal experimental design method

In the sequential design, an estimate of the V (where $G = U\Sigma V^T$) matrix is used to rotate the input such that the small singular value directions are perturbed more. In this case, the estimate of V based on the first 40 observations is:

$$\hat{V}_1 = \begin{pmatrix} 0.1542 & -0.5756 & 0.8030 \\ 0.9879 & 0.1053 & -0.1142 \\ -0.0189 & 0.8109 & 0.5849 \end{pmatrix}$$

Based on this rotation matrix, the input design will switch between the following:

$$\left\{ \begin{bmatrix} 10.3 \\ -1.2 \\ 8.2 \end{bmatrix}, \begin{bmatrix} -10.9 \\ 1.8 \\ -7.3 \end{bmatrix}, \begin{bmatrix} 11.0 \\ -1.3 \\ 7.2 \end{bmatrix}, \begin{bmatrix} -10.2 \\ 1.7 \\ -8.2 \end{bmatrix}, \begin{bmatrix} 10.2 \\ -1.7 \\ 8.2 \end{bmatrix}, \begin{bmatrix} -11.0 \\ 1.3 \\ -7.2 \end{bmatrix}, \begin{bmatrix} 10.9 \\ -1.8 \\ -7.3 \end{bmatrix}, \begin{bmatrix} -10.3 \\ 1.2 \\ -8.2 \end{bmatrix} \right\}$$

Note that this design, contrary to the traditional PRBS design, is highly correlated. The test signal based on this design is shown in Figure 4.7. Although the magnitude of the test signals are larger than the PRBS used earlier (which had a magnitude of ±1), the output variation remains approximately the same (they are not exactly the same since $\hat{V} \neq V$). Based on this design (and the first 40 observations) a new model was estimated:

$$\hat{G}_{2s} = \begin{pmatrix} -0.0187 & -2.3274 & -0.5426 \\ -0.5294 & -2.0379 & 0.2807 \\ -1.0221 & -0.5068 & 1.4016 \end{pmatrix} \pm \begin{pmatrix} 0.1792 & 0.2412 & 0.2664 \\ 0.2809 & 0.2082 & 0.1949 \\ 0.1952 & 0.2093 & 0.1304 \end{pmatrix}$$

$$\left\{ \pm \begin{pmatrix} 0.2552 & 0.2419 & 0.2438 \\ 0.2197 & 0.2082 & 0.2099 \\ 0.2358 & 0.2235 & 0.2253 \end{pmatrix} \right\}_{OLS}$$

Based on a simple inspection of the confidence intervals of the gain matrix, the model quality for both $\hat{G}_2$ and $\hat{G}_{2s}$ are about the same. In both cases, one gain element

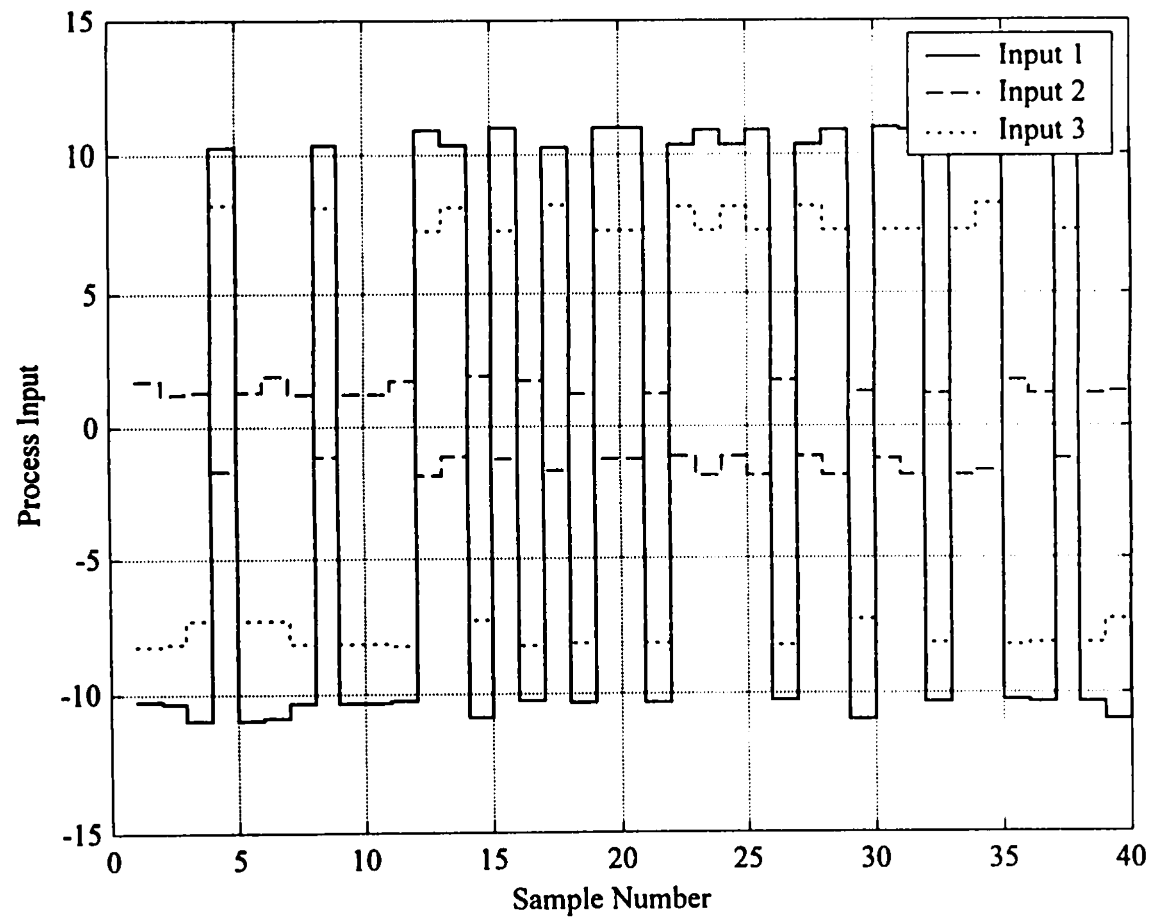


Figure 4.7: The correlated input design used to collect 40 more observations

confidence interval includes zero. This model satisfies the CSC, and if a controller was designed based on this model, it should be stable.

$$\lambda_i\left(G \times \hat{G}_{2S}^{-1}\right) = \{4.7372 \quad 0.1210 \quad 0.9851\} > 0, \forall i$$

As mentioned earlier, this is the result of the weak gain directions being specifically perturbed to produce an improved estimate of the smallest singular value. It should be noted that in this particular situation, (by chance) $\hat{V}$ was a good estimate of V, and as a result the second experimental design perturbed the low-gain direction sufficiently to produce an adequate model. However, in many other simulations (results not shown here), where $\hat{V}$ was a poor estimate of V, the experimental design did not perturb the low-gain direction sufficiently to produce an adequate model. Therefore, there is no assurance that after the second set of data has been collected the estimated model will be stable.

Method 3: Model Fix-up method

Utilizing the posterior knowledge that the controller based on the first data set was unstable, the gain matrix may be re-estimated using (4.9) yielding:

$$\hat{G}_{1f} = \begin{pmatrix} -0.0057 & -2.4893 & -0.4760 \\ -0.4741 & -1.9989 & 0.2586 \\ -0.9677 & -0.5158 & 1.2289 \end{pmatrix} \pm \begin{pmatrix} 0.3001 & 0.4003 & 0.3364 \\ 0.2937 & 0.3472 & 0.3270 \\ 0.3906 & 0.4136 & 0.3635 \end{pmatrix}$$

This re-estimated model is based on the original 40 observations and did not require the collection of more data. This model satisfies the CSC, and a model-based controller with appropriate tuning will result in a stable control system.

$$\lambda_i\left(G \times \hat{G}_{1f}^{-1}\right) = \{0.0721 \quad 1.0244 \quad 18.0280\} > 0, \forall i$$

In summary, both the sequential experimental design and the model fix-up method resulted in a stable system, while the traditional experimental design still resulted in an unstable controller. The sequential design accomplished this via collection of more data (80 data points were used). The model fix-up method used the acquired knowledge about the controller performance and the original data (40 data points were used) to produce a stable model. As mentioned before, there are other measures of evaluating the model quality. Although both the fix-up method and the sequential design method give a stable controller according to the CSC, based on other model quality measures (such as the confidence interval of estimated parameters) the sequential design may be better, since it uses 80 data points compared to 40 data points (in the case of the fix-up method).

4.4.3. Model Fix-up in Reduced Dimensions

In the previous section, it was assumed that none of the process constraints were active. As mentioned earlier, if a constraint on the input or output becomes active, the characteristic of the problem changes. If the full dimension problem is stable, it does not imply that the reduced dimension (or sub-problem) will be stable. In this section, two different methods are suggested to deal with both the full-dimensional problems and the different reduced dimensional problems that may arise due to constraints becoming active.

4.4.3.1. Multimodel Controller

The idea of using a multimodel control for a non-linear process has been studied extensively. In the case of a SISO systems, this style of controller is referred to as a gain scheduler. Similar styles of controllers have been designed for MIMO systems where the model that is utilized for control is dependent on the operating region (Häggblom and Böling 1998). In such a situation, several models that capture system behavior over the different operating regions are estimated and utilized for control.

The objective in this section is not control in different operating regions but control with different sets of active constraints (in different dimensions or spaces). As the controller is operating, constraints may become active and the dimension that it operates in will change. If the controller has satisfactory performance in the original full dimension, there is no guarantee that it will maintain satisfactory performance once it is operating in the reduced (or changed) dimension. One method to combat this problem is to have different models for different dimensions. One can estimate the model for different dimensions in the identification phase. If the performance measure is the CSC, in each of the different dimensions, the identification experiment should perturb the smallest singular value ($\underline{\sigma}$) sufficiently in order to obtain a good estimate of $\underline{\sigma}$ (Koung and MacGregor 1994). This would result in a combinatorial identification problem where the number of identification experiments is given by (4.2). For a small system, where the dimension is less than 4, it may be possible to practically implement this sort of identification; however, for larger systems it is impractical to identify all the sub-models individually. Another possibility is that the process engineer has prior knowledge about which constraints will become active during normal operation; then that prior knowledge can be used to perturb the $\underline{\sigma}$ of those sub-models individually in the identification phase.

A simpler method would be to use the original model and monitor controller performance as constraints become active, or inactive, as the controller moves into different spaces. Assume that the set J and K ($\dim\{K\}=\dim\{J\}$) is the set of manipulated and controlled variables, respectively, that are at a constraint. Then let $G^{J,K}$ be the new gain matrix resulting from removing columns J and rows K from G_f (the full dimension gain matrix). If for a particular set of constraints (J^l, K^l) the resulting controller is stable, the CSC has not been violated, and the sign of $\det\left(\hat{G}^{J^l,K^l}\right)$ is correct. However, if the controller is unstable (for various tuning, see Appendix 10), it is assumed that the CSC is violated and the sign of $\det\left(\hat{G}^{J^l,K^l}\right)$ is incorrect. For a particular sub-system (let $\hat{G}_{r,1} = \hat{G}^{J^*,K^*}$, to simplify notation) that results in UCS, re-identified using the fix-up (4.9) can be implemented:

$$\underset{\hat{\hat{G}}}{Min} SSE \qquad (4.10)$$
$$s.t. -\det\left(\hat{\hat{G}}_r\right) \times sign\left(\det\left(\hat{G}_r\right)\right) > 0$$

More than one set of constraints may result in an UCS. This would result in multiple $\hat{\hat{G}}_r$ models ($\hat{\hat{G}}_{r,1}, \hat{\hat{G}}_{r,2}, \cdots, \hat{\hat{G}}_{r,k}$). This would indicate that as changes in operating dimension occur, different gain matrices should be used.

Based on the simulation results in section 4.4.1, many of the reduced dimension systems may be fixed this way. Similar to the full-dimension problem in the last section, if re-identified using (4.10) does not result in a reduced system that is stable, then either an experimental design focused on this reduced dimension should be performed or an SVD controller should be implemented. The following example illustrates these ideas.

Example 4.2: Consider a new hypothetical system that is 3×3:

$$G = \begin{pmatrix} 0.0573 & -0.2415 & 1.2071 \\ 0.2546 & -0.9267 & -0.1841 \\ -0.0328 & -1.4567 & -0.2617 \end{pmatrix}$$

where separate i.i.d. white noise of variance 1 was added to each process output. This process is well-conditioned (with a condition number of 7.5). The process inputs were perturbed with a PRBS of magnitude ±0.5 and 200 observations were collected. This resulted in a data set where the signal-to-noise ratio based on standard deviations

($\sigma_{Signal}/\sigma_{Noise}$) for each output is 1.2324, 0.9785, and 1.4804, respectively. The resulting model is (where the second expression on the right is the 95% confidence interval estimated by jackknifing):

$$\hat{G}_1 = \begin{pmatrix} 0.0361 & -0.3066 & 1.3320 \\ 0.3062 & -1.2447 & -0.1002 \\ 0.0671 & -1.5634 & -0.2581 \end{pmatrix} \pm \begin{pmatrix} 0.3371 & 0.2866 & 0.2557 \\ 0.2557 & 0.2911 & 0.2543 \\ 0.2767 & 0.3056 & 0.4107 \end{pmatrix}$$

This estimated model satisfies the CSC:

$$\lambda_i\left(G \times \hat{G}_1^{-1}\right) = \{0.9094 \quad 0.8042 \quad 1.2684\} > 0, \forall i$$

If during the operation the first input and first output are at a constraint, the reduced system (let us call this the first reduced system) will violate the CSC, and the controller will become unstable:

$$\lambda_i\left(G_{r1} \times \hat{G}_{1,r1}^{-1}\right) = \{-0.1625 \quad 0.9593\} \not> 0, \forall i$$

Using this as posterior knowledge, the model can be re-estimated using (4.10) with a constraint on this reduced system using the original 200 observations to yield:

$$\hat{G}_2 = \begin{pmatrix} 0.0361 & -0.3065 & 1.3320 \\ 0.3123 & -1.2414 & -0.1682 \\ 0.0623 & -1.5663 & -0.2042 \end{pmatrix} \pm \begin{pmatrix} 0.3371 & 0.2866 & 0.2557 \\ 0.2558 & 0.3011 & 0.2559 \\ 0.2729 & 0.2919 & 0.3235 \end{pmatrix}$$

The re-estimated gain matrix results in a stable system in both the first reduced space and the full space:

$$\lambda_i\left(G \times \hat{G}_2^{-1}\right) = \{0.8897 \quad 0.7921 \quad 1.2635\} > 0, \forall i$$
$$\mathrm{Re}\left(\lambda_i\left(G_{r1} \times \hat{G}_{2,r1}^{-1}\right)\right) = \mathrm{Re}\left(\{0.9653 \pm 1.2784i\}\right) > 0, \forall i$$

In the multi-model framework, $\hat{G}$ will be used for control when the controller is operating in the full space and $\hat{G}_2$ will be used when the constraint on both input 1 and output 1 become active. During a longer period of operation, it is possible for other constraints to become active. For example, a constraint on both input 2 and output 2 (let us call this the second reduced system) may become active. In such a case, neither the first nor the second estimated model can control the system in this reduced space, since they both violate the CSC:

$$\lambda_i\left(G_{r2} \times \hat{G}_{1,r2}^{-1}\right) = \{0.9003 \quad -0.2768\} \not> 0, \forall i$$
$$\lambda_i\left(G_{r2} \times \hat{G}_{2,r2}^{-1}\right) = \{0.8890 \quad -0.3062\} \not> 0, \forall i$$

Using this posterior knowledge, the model may be re-estimated using (4.10) with a constraint on the second reduced system's determinant to yield:

$$\hat{G}_3 = \begin{pmatrix} 0.0213 & -0.3062 & 1.3340 \\ 0.3062 & -1.2448 & -0.1002 \\ -0.0115 & -1.5620 & -0.2496 \end{pmatrix} \pm \begin{pmatrix} 0.3343 & 0.2867 & 0.2621 \\ 0.2557 & 0.2911 & 0.2544 \\ 0.0693 & 0.3052 & 0.4095 \end{pmatrix}$$

While this model will result in a stable system for the second reduced system, it will result in an unstable system for the first reduced system, and a stable full system:

$$\mathrm{Re}\left(\lambda_i\left(G \times \hat{G}_3^{-1}\right)\right) = \mathrm{Re}\left(\{0.8616 \pm 0.0661i \quad 0.9947\}\right) > 0, \forall i$$
$$\lambda_i\left(G_{r1} \times \hat{G}_{3,r1}^{-1}\right) = \{-0.1712 \quad 0.9723\} \not> 0, \forall i$$

$$\lambda_i\left(G_{r2} \times \hat{G}_{3,r2}^{-1}\right) = \{0.8378 \quad 2.9359\} > 0, \forall i$$

Therefore, a multi-model controller in this scheme will switch between the 3 estimated models depending on which space the controller is operating in. Moreover, when the controller encounters a new reduced space it will gain more knowledge. The number of spaces the controller encounters throughout its operation does not limit this method. The more unstable reduced systems that it encounters, the more models it will have.

4.4.3.2. Adaptive Control

An alternative to the multimodel controller, mentioned in the last section, is an adaptive style controller. In an adaptive controller system, the controller parameters are adjusted routinely to compensate for varying process conditions. A linear adaptive control of a non-linear process would in essence alter the parameters of the linear model such that the linear model approximates the non-linear model at that operating point. In this section, a similar ideology is suggested for a MIMO system that changes dimensions. Contrary to the multimodel controller where different models were used for the different dimensions, an adaptive controller (based on this author's definition) will use a single model for all operating dimensions. The model may alter, however, as the controller moves into new dimensions and gains knowledge.

Simply stated, as the controller moves into different dimensions, which is the result of different sets of constraints becoming active (J^i, K^i), knowledge about the CSC in that dimension is gained. When the activation of a constraint results in an UCS, the model is re-estimated with all the posterior knowledge.

$$\underset{\hat{G}}{Min}\, SSE \tag{4.11}$$
$$s.t.\begin{cases} -\det\left(\hat{G}_{r,i}\right) \times sign\left(\det\left(G_{r,i}\right)\right) > 0, i \in \{1, \cdots, k'\} \\ \det\left(\hat{G}_{r,i}\right) \times sign\left(\det\left(G_{r,i}\right)\right) > 0, i \in \{k'+1, \cdots, k''\} \end{cases}$$

where k' is the number of the sub-systems that have resulted in UCS

k'' is the total number of the systems (sub-system or full system) that the controller has operated in

$G_{r,i}$ is the reduced system i that corresponds to a particular set of constraints becoming active (J^i, K^i)

To see how such a global model can be used for control, consider the following example.

Example 4.3: Reconsider the system from Example 4.2. Under the adaptive scheme, the controller gain matrix will adapt to the different spaces as the controller moves through them. In Example 4.2, the controller started in the full space, next moved into the first reduced space (where the first input and output were at a constraint) and then into the second reduced space (where the second input and output were at a constraint). The adaptive controller will have the same model at the start as the multimodel controller had in the full space (namely $\hat{G}_1$). When the system moves to the first reduced space, the model will have to be re-estimated with the following two constraints: a constraint on the full gain matrix determinant to keep its sign the same as $\det(\hat{G}_1)$ and a constraint on the first reduced system to change its sign to be the opposite of $\det(\hat{G}_{1,r1})$. This will be implemented in a similar fashion as in (4.11). The resulting model of this optimization problem will be $\hat{G}_2$ (which is the same $\hat{G}_2$ in Example 4.2). This model is not only stable in the full space but also in the first reduced space. Next, the system moves into the second reduced space. The new model will have to be stable in the full space and both reduced spaces. This is accomplished by applying (4.11) with three constraints: one on the full system and one on each of the two reduced systems. The resulting model is:

$$\hat{G}_4 = \begin{pmatrix} 0.0250 & -0.3063 & 1.3337 \\ 0.3120 & -1.2418 & -0.1644 \\ -0.0112 & -1.5646 & -0.1991 \end{pmatrix} \pm \begin{pmatrix} 0.3376 & 0.2867 & 0.2609 \\ 0.2552 & 0.3004 & 0.2575 \\ 0.0539 & 0.2909 & 0.3232 \end{pmatrix}$$

which is stable in all the spaces that the controller has encountered up to this point:

$$\mathrm{Re}\left(\lambda_i\left(G \times \hat{G}_4^{-1}\right)\right) = \mathrm{Re}\left(\{0.8357 \pm 0.0464i \quad 1.0423\}\right) > 0, \forall i$$

$$\mathrm{Re}\left(\lambda_i\left(G_{r1} \times \hat{G}_{4,r1}^{-1}\right)\right) = \mathrm{Re}\left(\{0.9025 \pm 1.3235i\}\right) > 0, \forall i$$

$$\lambda_i\left(G_{r2} \times \hat{G}_{4,r2}^{-1}\right) = \{0.7796 \quad 3.1553\} > 0, \forall i$$

Two different issues arise from the application of (4.11) to estimate a global model that can handle different dimensions. The first issue is feasibility. As the system operates in k'' different sub-systems, progressively more posterior knowledge about the CSC is gained. The posterior (gained prior) knowledge is added to the model in the form of constraints in (4.11). As the number of sub-systems increases ($k'' \rightarrow k''_{max}$), where the maximum number of sub-systems can be estimated by (4.2), the possibility that the set of constraints will result in no feasible region increases. Unfortunately, there is no easy way of checking if a feasible space exists. Moreover, the feasibility analysis itself results in another optimization problem (a min max optimization to be exact; see Dimitriadis and Pistikopoulos 1995). In addition, if the process is linear time variant (LTV) or non-linear, it is possible that there may be inconsistent constraints. In such a situation, the constraints should be relaxed (i.e., instead of hard constraints use soft constraints). The second issue is after implementation of (4.11), the sub-system could still be unstable or the addition of the constraint on the sub-system may result in a full-system that is unstable. In fact, there are $2^{k''}$ cases that can arise. After implementation of (4.11), when $k''=2$, there are 4 cases to consider:

-Both the sub-system and full-system are stable

-The sub-system is unstable while the full-system is stable

-The sub-system is stable while the full-system is unstable

-Both the sub-system and full-system are unstable

Monte Carlo simulations were performed to estimate the probability of each one of these cases.

In the Monte Carlo simulations of section 4.4.1 the effect of adding one constraint was studied. In this section, the effect of adding two constraints (one on the full dimension problem and one on the reduced dimension problem) is studied. This was accomplished by studying many different 5x5 systems that resulted in an UCS in the full dimension. If a system was UCS in the full dimension (4.9) was implemented, after which, about 94.5% of the systems became stable (see Table 4.1). At this stage, to see the effectiveness of (4.11), one row and column of G (which in the full-space was originally unstable) was randomly removed (to imitate activation of a constraint on an input and output) and the system ($\hat{G}_r$) was tested for CSC. It was determined that about 15.5% of the reduced systems ($\hat{G}_r$) were unstable (see Table 4.3). In these cases, (4.11) was applied to these systems. This resulted in 90.15% (±0.75%) stable systems, which is slightly lower than the 94.50% (±0.22%) for the one constraint case (see Tables 4.1 and 4.2). This illustrates that as more and more constraints (as $k'' \rightarrow k''_{max}$) are added to the optimization problem (4.11) the effectiveness of this methodology will decrease. A conservative estimate of the probability of stable control system (SCS) may be estimated if the effectiveness of adding one constraint is independent of the effectiveness of adding another constraint. Then a lower bound for the probability of (4.11) resulting in a stable model (when both the estimated original full and reduced dimension models were unstable ($k'=k''$)) may be estimated by:

$$p_{r,k''} \geq (p_r)^{k'} \tag{4.12}$$

where $p_{r,k''}$ is the probability of SCS in all the k'' different spaces after the model fix-up of (4.11) is implemented

p_r is the probability of SCS after the model fix-up of (4.9) is used (in this thesis p_r is assumed to be 94.5% see Table 4.1)

k' is the number of unstable systems encountered

k'' is the total number of spaces encountered by the controller

For the case shown in the Monte Carlo result with k'=2 the probability of SCS for both the full space and reduced space calculations based on (4.12) (assuming p_r=94.50% ± 0.22%) is at least 89.30% (±0.42%), which is within the confidence interval of the Monte Carlo results (90.15% ± 0.75%). To explain why (4.12) is a lower bound, consider that if both the reduced system and the full-system are unstable, fixing either the reduced or the full system may fix both systems (i.e., the effectiveness of each constraint is not independent). In fact, fixing the reduced system in many cases will fix the full-system as well (see Appendix 14). Equation (4.12) suggests that as the number of different sub-systems increases, the probability that all systems will remain stable will decrease (this issue was mentioned earlier).

4.4.4. SVD Controllers

It is important to mention that there is a possibility that the system is truly singular (i.e., condition number is infinity). In such a situation, the CSC would certainly be violated since one of the eigenvalues is zero and control in the full output space is impossible. Since no model can produce a stable control system, it would be inappropriate to implement (4.9) or (4.11), and an implementation of (4.9) or (4.11) will certainly result in an UCS. Therefore, it is suggested that if after implementation of (4.9) or (4.11) the control system remains unstable, a SVD style controller should be implemented. In essence, if the system cannot be controlled in the full space, it is assumed that the system is singular and a SVD style controller is used. Four different situations would give rise to this assumption:

1) the system may be fundamentally singular,

2) the non-linearity or time varying nature of a system may cause changes in the eigenvalue sign,

3) an odd number of eigenvalues have the wrong sign,

4) the controller may be too aggressively tuned ($0 \leq \alpha \leq \alpha^*$, see Appendix 1).

Since it would be difficult to distinguish which is the case, a more conservative (and traditional) approach, such as the SVD-controller, should be used (Figure 4.8 illustrates this in more detail).

In this thesis, the emphasis is on model quality evaluation using the CSC. Since an SVD style controller will always result in an UCS, its performance was not compared to the other controller designs that were mentioned. Other measures of model quality could allow comparison of the SVD controller to a multimodel or adaptive controller; however, this was left as a future research area.

If the eigenvalue of the system were small (but not zero), it would certainly be difficult to control in the low-gain direction. Even if the true model is used for control (i.e., assume no error in the estimated model), the controller would have to make large changes in the manipulated variable to reach set point changes of the controlled variable that happen to be in the direction of the small eigenvalue. In certain situations, however, it may be important to control in the low-gain direction of the system. The value of including the small singular value in the model rather than excluding it is shown in the next example.

Example 4.4: In this example, a few different situations are illustrated where it is helpful to keep the small singular value in the estimated model. Since it is not possible to illustrate this point by using the same model for the different cases, different models will be used in each case.

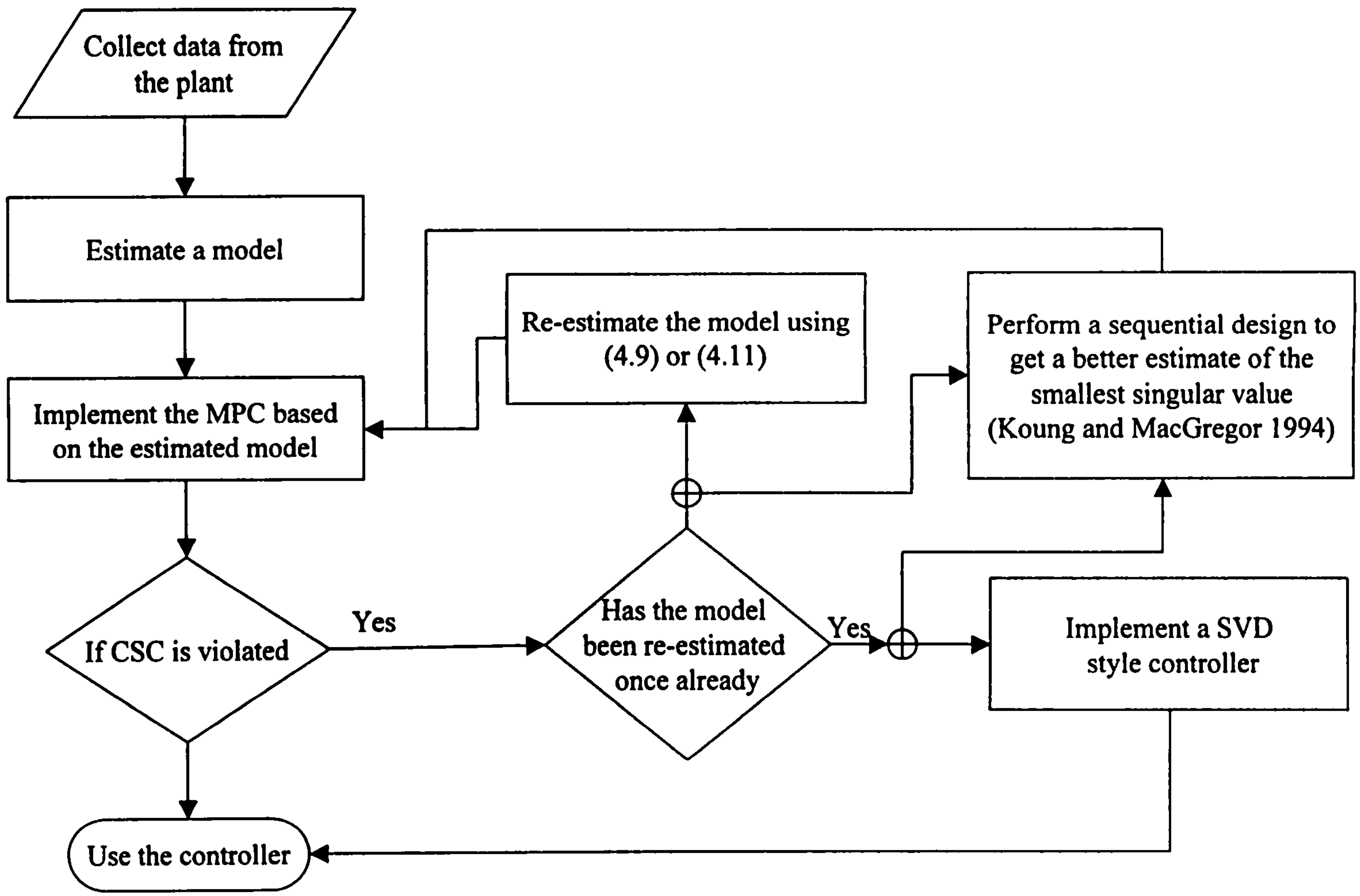


Figure 4.8: A flowchart of different methods of handling problems associated with ill-conditioning (where ⊕ implies or)

Case 1: In today's chemical plants, the MPC is a component of a multi-level hierarchy of control functions. The objective of this hierarchy is to operate the plant at the most economically feasible point (or region). Different vendors use different types of hierarchical control and optimization structure (Richalet et al. 1978, Qin and Badgwell 1996). Loosely stated, they have the following 4 levels (the expression in brackets is the approximate frequency of occurrence):

1) A steady state plant-wide optimization and scheduling (every day)

2) Steady state local optimizations of the unit to minimize cost and ensure quality and quantity of production (every hour).

3) Dynamic multivariable constraint control of the unit (every minute)

4) PID control of valves (every few seconds)

While vital for process operability and safety the economical benefits induced by levels 3 and 4 are usually insignificant (Qin and Badgwell 1996). However, it is possible that control in the low-gain direction of the system (level 3) is crucial to the economics of the system (level 1 and 2). For instance, the direction of the small singular value of the system may greatly influence the objective function of the (local or global) process optimizer. In the extreme case, the process optimizer may be exclusively influenced by the low-gain direction, and other directions may have no effect on its objective function. In the other extreme of this hypothetical situation, the low-gain direction may have no effect on the objective function of the optimizer. It is difficult to estimate the probability of each of these situations in chemical plants, unless an extensive survey is performed. It is the assumption of this author that probably the most common situation is that the low-gain direction has some effect on the objective function of the optimizer (but it is not the sole effect). In such a situation, addition of the low-gain direction will improve the plant economics.

Consider a simple 2x2 system, such a small system will allow us to visualize the control action and its mapping on to the optimization surface. The following 2x2 gain matrix (which was also looked at in chapter 2) is used:

$$G_5 = \begin{pmatrix} 10 & -2.5 \\ 4 & -1.43 \end{pmatrix} \tag{4.13}$$

The condition number of this gain matrix is 29. Based on an identification experiment (PRBS magnitude ±1, switching time of 5, 100 observations, and the covariance of the white noise added to process output is I) the following model was estimated:

$$\hat{G}_5 = \begin{pmatrix} 9.6785 & -3.8541 \\ 4.7452 & -1.6181 \end{pmatrix} \tag{4.14}$$

The implementation of this model will result in an unstable control system since:

$$\lambda_i\left(G_5 \times \hat{G}_5^{-1}\right) = \{-1.9032 \quad 0.8598\} \not> 0, \forall i$$

Given that the estimated model is ill-conditioned (condition number is 51), then a conservative design might assume that the system cannot be controlled in the low-gain direction and therefore might utilize an SVD style controller. This can be accomplished by removing the smallest singular value of $\hat{G}_5$:

$$\hat{G}_5 = \hat{U}_5\hat{\Sigma}_5\hat{V}_5^T, \hat{\Sigma}_5 = diag\left(\hat{\sigma}_{5,1}, \hat{\sigma}_{5,2}\right), \hat{\sigma}_{5,1} \geq \hat{\sigma}_{5,2}$$

Using SVT methodology, the smallest singular value will be set to zero and the new gain matrix recalculated as:

$$\hat{G}_{5,SVD} = \hat{U}_5\hat{\Sigma}_{5,SVD}\hat{V}_5^T, \hat{\Sigma}_{5,SVD} = diag\left(\hat{\sigma}_{5,1}, 0\right), \hat{\sigma}_{5,1} \geq 0$$

In the present example this yields:

$$\hat{G}_{5,SVD} = \begin{pmatrix} 9.7141 & -3.7622 \\ 4.6712 & -1.8091 \end{pmatrix} \tag{4.15}$$

Utilizing the posterior knowledge that the controller based on the first data set was unstable, the gain matrix may be re-estimated using fix-up (4.9):

$$\hat{G}_{5f} = \begin{pmatrix} 9.7864 & -3.7682 \\ 4.7783 & -1.8471 \end{pmatrix} \tag{4.16}$$

Assume that there is an optimizer (such as real-time optimizer, RTO) on the top of the control structure (level 1 and 2 in the control and optimization hierarchy) with the following quadratic objective function:

$$\underset{\overline{y_{sp}}}{Min} \frac{1}{2} \overline{y_{sp}}^T \begin{bmatrix} 1 & 0.5 \\ 0.5 & 0.5 \end{bmatrix} \overline{y_{sp}} + \begin{bmatrix} 1.6 & 1 \end{bmatrix} \overline{y_{sp}}$$

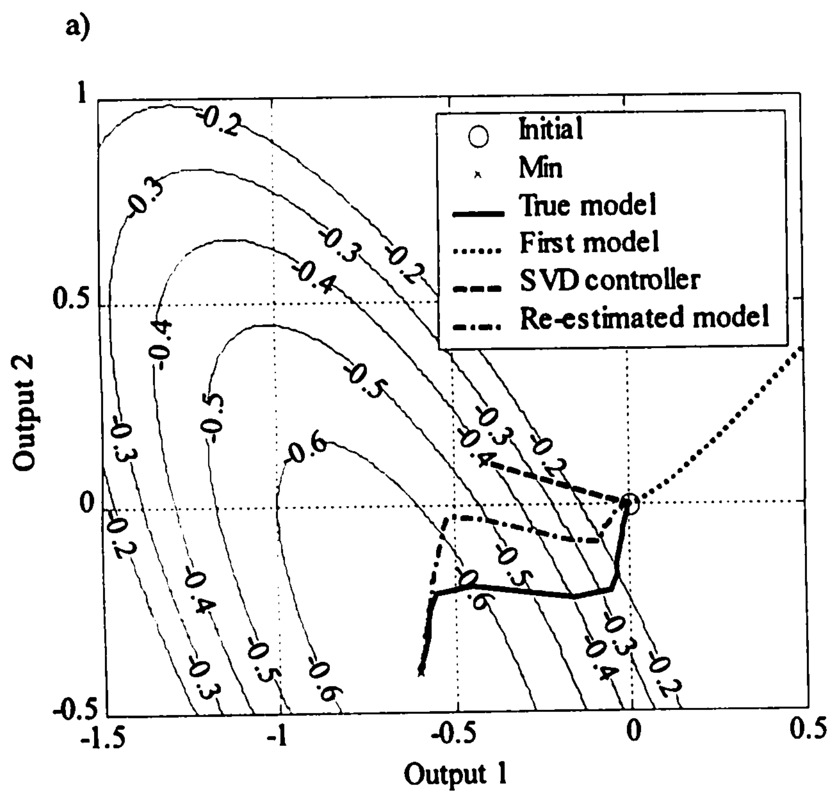
The above optimization problem can be viewed as the cost of operation for this chemical unit. The above optimization problem is a simplified version of what is encountered in chemical processes. In many cases, the above optimization problem includes the manipulated variables, and the objective function is dependent on economical factors that would be changing hourly (resulting in changes of the objective function and the solution to the optimization problem). The minimum in the above optimization problem is:

$$\overline{y_{sp}} = \begin{pmatrix} -0.6 \\ -0.4 \end{pmatrix}$$

Figure 4.9 illustrates how the different controllers (which are the result of using different estimated models) result in different controller performance. The first estimated model $\hat{G}_5$ results in an unstable controller; this was expected since it violates the CSC. The second model, which is based on removing the smallest singular value from $\hat{G}_5$, results in a controller that moves closer to the set point target, which is the solution of the optimization problem; however, it cannot reach the set point target. In this case, the move by the controller lowers the objective function by –0.4 (as shown by the contours in Figure 4.9 a)). Next, the re-estimated model $\hat{G}_{5f}$ reaches the set point, even though it takes a slightly different path compared with the case when the true model is used. In this case, there is a cost saving of –0.68 (or a 70% improvement over the case when the SVD controller was used). Although the process inputs are not shown here, they all remained in the range of ±1.5 during this time period, except for the controller that used the first estimated model $\hat{G}_5$, which went unstable soon after the set point change.

As mentioned earlier, this is a simplified example illustrating that it is possible to have a system that is ill-conditioned in a particular direction while the plant economics dictate that the plant should be moved, at least partially, in that direction. The same results may be obtained with larger systems. While the objective function in the above optimization problem is quadratic (resulting in a QP problem), this is not necessary. One could make a linear (or more non-linear) objective function, which would also require the plant to operate in the low-gain direction.

Case 2: It is important to realize that the systems being considered in real situations are rarely deterministic. Almost always the estimated model is based on a particular data set (realization) and may not be a good representation of the true model (this would be the case if there is a small data set or the variance of the signal to the variance of the noise is low). In such a situation, an estimate of a system that is truly well-conditioned may appear ill-conditioned Conversely, a system that is ill-conditioned may appear well-



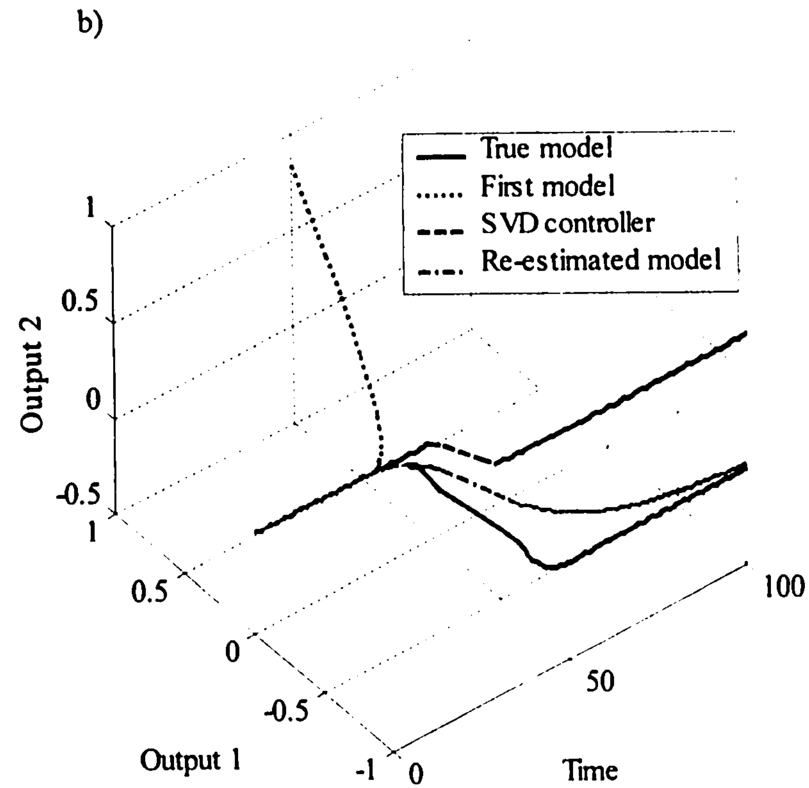


Figure 4.9: Controller performance for example 4.4 case 1 where the set points are the result of the QP problem. There is a bound of +/- 2 on all the inputs and outputs. The following are the controller parameters for the QDMC: M = 7, P = 20, all the inputs and the outputs are weighted equally. Although the inputs are not shown here they were within the range of +/- 1.5 except for the controller with the first model. a) Process output movement in the output space, where the contours are from the QP b) The CVs trajectory as a function of time

conditioned (see Example 4.1 where the original model is ill-conditioned ($\bar{\sigma}/\underline{\sigma}$=1436) but the estimated model ($\hat{G}_2$) is well-conditioned ($\bar{\sigma}/\underline{\sigma}$=62)).

Let us now reconsider the well-conditioned system of Example 4.2 (condition number is 7.5). The first estimate of this model is given by $\hat{G}_1$ (condition number of $\hat{G}_1$ is 10.4), which is based on a particular realization of the PRBS used and the random noise added to the system. Based on a different realization of the PRBS and the random noise added to the system a different model was estimated:

$$\hat{G}_2 = \begin{pmatrix} 0.0937 & -0.3643 & 1.4225 \\ 0.1176 & -1.1315 & -0.1563 \\ 0.1544 & -1.4600 & -0.1176 \end{pmatrix} \pm \begin{pmatrix} 0.2101 & 0.2135 & 0.3311 \\ 0.2136 & 0.2731 & 0.2220 \\ 0.3290 & 0.2036 & 0.2665 \end{pmatrix}$$

Note that for both realizations the following were kept constant: the number of observations collected, the true model, magnitude of the PRBS, and the signal-to-noise ratio. While $\hat{G}_1$ is well-conditioned, $\hat{G}_2$ is very ill-conditioned (condition number of $\hat{G}_2$ is 5758). To implement $\hat{G}_2$ for control purposes, a practitioner may consider using an SVD style controller (i.e., set the small singular value of $\hat{G}_2$ to zero). In such a situation, the controller will not be capable of controlling in the full output space. However, if the $\hat{G}_2$ was used in the controller (i.e., no singular values were set to zero), the controller would be capable of controlling in the full output space (assuming no constraint on the inputs or outputs). In Figure 4.10 this point is illustrated by comparing the two controllers for a set point change in the low-gain direction. In this figure the SVD controller does not reach the setpoints; however, using $\hat{G}_2$ the controller is reaches its setpoints.

Not only is it possible that a well-conditioned system will be estimated as an ill-conditioned system, it is also possible that the well-conditioned system may be estimated with the incorrect sign of one or more eigenvalues. The probability of such an event will

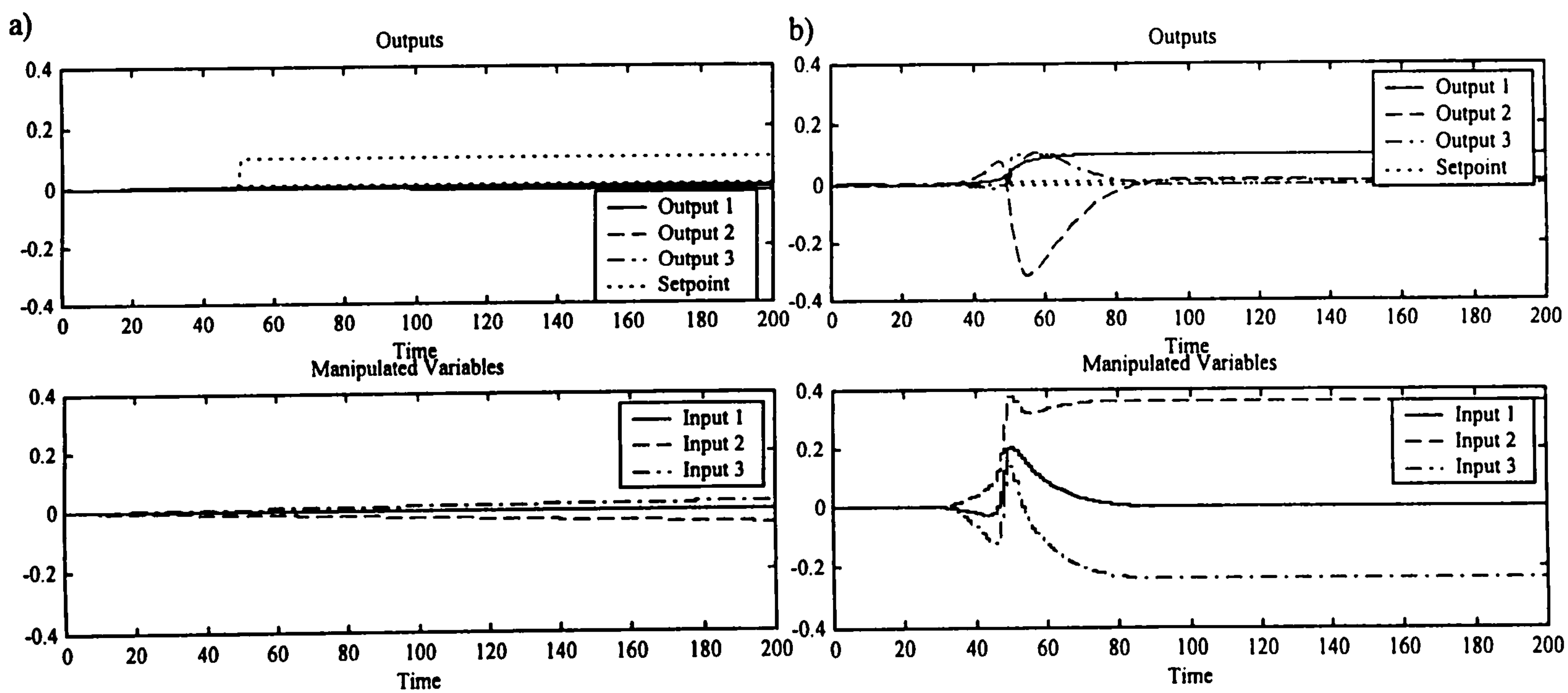


Figure 4.10: QDMC controller performance for example 4.4 case 2 where the set points are in the direction of the small singular value. a) The estimated model used is singular. b) The estimated model used is full rank. The following are the controller parameters for both case: M = 7, P = 20, all the inputs and the outputs are weighted equally, there is a bound of ± 1 on all the inputs and outputs.

increase as the ratio of signal-to-noise and number of observations decreases. In such a situation, the re-estimation method (4.9) can be used to fix the estimated model so that the resulting control system is stable.

Case 3: While the scaling in equation (4.1) results in an improvement of the condition number of the estimated full problem, it may also result in estimated reduced systems that are ill-conditioned unless the scaling using the optimization problem (4.1) is performed again for the reduced system. Yet, it appears that in commercial packages the scaling is only performed for the initial problem (at the design phase MacArthur 1996) and not redone for the reduced problem that may occur due to constraints becoming active.

Consider the following gain matrix:

$$G = \begin{pmatrix} 10 & -10 & 1 & .5 & .6 \\ 4 & -1.3 & -.2 & .75 & .6 \\ 1 & 10 & -1 & 1.5 & 1 \\ 0 & -5.5 & 0 & 0 & .25 \\ 1 & 6 & 3 & 10 & 6 \end{pmatrix}$$

the condition number of this gain matrix is 228. To improve the condition number of this gain matrix one can apply (4.1). The solution of this optimization problem is:

$$D_X = (\ -0.2700 \quad 1.5227 \quad 11.6485 \quad 10.1009 \quad 13.1211) \qquad (4.17)$$

$$D_Y = (\ -1.0750 \quad 1.6768 \quad 0.8685 \quad 1.7810 \quad 0.0821)$$

Then the minimum condition number (based on the above scaling) is 24.

Now assume that inputs 3, 4, 5 and outputs 2, 3, 4 are at a constraint (this may occur due to a certain set point change or disturbance). Then the resulting reduced system is without scaling:

$$G_r = \begin{pmatrix} 10 & -10 \\ 1 & 6 \end{pmatrix}$$

This reduced gain matrix is well-conditioned (condition number is 3). However, if the scaling of (4.17) is used, the reduced system will have a condition number of 109 (the gain matrix is now ill-conditioned). Based on this scaling, one may think that control in the low-gain direction of the reduced system is not possible (i.e., an SVD style controller should be used); however, in reality the system is well-conditioned. In fact, if (4.1) is applied to this reduced system, the minimum condition number will be 1. This case illustrates that due to scaling of the full problem a reduced system may appear to be ill-conditioned, while in reality it is a well-conditioned system.

In summary, one can see that there are advantages and disadvantages to incorporating the small singular value of the gain matrix. The choice of inclusion or exclusion of the small singular values in the problem formulation is case dependent.

4.5. Controller Stability Criteria for Non-Square Gain Matrices

The controller stability criteria (CSC) mentioned in previous chapters is based on the assumption that the system is a square system (the number of inputs is equal to the number of outputs) (Garcia and Morari 1985). In the case of more independent control variables than independent manipulated variables, the CSC will never be satisfied, since the system is uncontrollable to begin with (i.e., the system cannot be controlled in the full output space defined by all the CVs). Conversely, if the number of independent controlled variables (CV) is less than the number of independent manipulated variables, the system may be not completely reachable and a stable controller could be designed. Both of these cases, where the number of independent manipulated variables is not equal to number of independent controlled variables, are referred to as non-square problems in general. In this chapter, a non-square problem refers to the case where the number of manipulated variables is greater than the number of controlled variables.

The controller stability criteria for square problems that was mentioned in previous chapters can be extended to a non-square problems as follows:

$$\mathrm{Re}\left(\lambda_i\left(G_{ns}\hat{G}_{ns}^{+}\right)\right) > 0, \forall i \tag{4.18}$$

where G_{ns} is the true non-square gain matrix

$\hat{G}_{ns}$ is the estimated non-square gain matrix

A^{+} is the left pseudo-inverse of A ($A^{+} = \left(A^T A\right)^{-1} A^T$) (Ogata 1995).

In the case that the gain matrix is a square matrix $G^{-1} = G^{+}$, the condition in (4.18) reduces to the controller stability criteria (CSC) mentioned in previous chapters.

Using the singular value decomposition approach to evaluate the effect of uncertainty on the CSC for the non-square problem results in:

$$\mathrm{Re}\left(\lambda_i\left(U\Sigma V^T\hat{V}\hat{\Sigma}^{+}\hat{U}^T\right)\right) > 0, \forall i \tag{4.19}$$

where $G_{ns} = U\Sigma V^T$

For simplicity, assume that uncertainty in the gain matrix is only in the singular values of it ($U = \hat{U}$ and $V = \hat{V}$). Then the eigenvalues will all be positive or zero:

$$\left(\frac{\sigma_1}{\hat{\sigma}_1}, \frac{\sigma_2}{\hat{\sigma}_2}, \dots, \frac{\sigma_m}{\hat{\sigma}_m}\right) \geq 0 \tag{4.20}$$

This would mean that if there is no model mismatch in U and V, the system will always satisfy the CSC as mentioned in (4.18). Furthermore, it implies that USC is the result of error not only in singular values but also in U and V. (Similar results can be derived for a square problem, see Koung 1991).

The extension of the ideas of CSC to non-square problems requires the evaluation of the singular values. However, if a similar study was to be performed on the singular values as was performed for the eigenvalues (in the square problem), the probability of singular values changing sign is required. Yet, by definition singular values are positive (or zero) and cannot have negative values. This shows that there are no convenient methods of using singular values to generalize (4.8). This has motivated the definition of pseudo-singular values (Hovd et al. 1993,1996, Featherstone et al. 1998, and Featherstone 1997), which by definition can be positive or negative:

$$\breve{G} = U\Sigma V^T = (UD_u)(D_u\Sigma)V^T = U_D\Sigma_D V^T \tag{4.21}$$

where D_u is a diagonal matrix which has each diagonal element either +1 or −1

$$D_u(i,j) = \begin{cases} i \neq j, D_u(i,j) = 0 \\ i = j, D_u(i,i) = sign\left((v_i)^T u_i\right) \end{cases}$$

Σ_D is a diagonal matrix with the diagonal elements referred to as the pseudo-singular values ($\Sigma_D = diag(\sigma_{D,1},\cdots,\sigma_{D,m},0,\cdots,0)$, $\sigma_{D,1} \geq \sigma_{D,2} \geq \cdots \geq \sigma_{D,m} \geq 0$

$\breve{G}$ is made of adding either extra rows or columns of zeros to matrix G to produce a square matrix

u_i and v_i refer to the ith column of the matrix U and V

The issue is how to fix an unstable control system when the system is non-square. The difference between the square problem and the non-square problem is that in the square problem the cause of controller stability can be isolated. In the square problem the

P(UCS) can be mainly attributed to the probability of the sign of the smallest eigenvalue being estimated incorrectly (when the uncertainty in all the eigenvalues is equal). In the non-square problem the P(UCS) can be approximated by the probability that a pseudo-singular value will be equal to zero or change sign (Featherstone et al. 1998). Similar to the derivation of (4.20), the distribution of pseudo-singular values can be estimated by assuming that there is no model uncertainty in U and V. Once it is determined which pseudo-singular value is the most probable cause of the system instability, then a determinant constraint can be placed on the elements in the gain matrix that contribute most to that pseudo-singular value.

The uncertainty in the pseudo-singular values, when errors in U and V are not explicitly taken into account (equivalent to assuming that the covariance of the gain matrix elements is zero) can be estimated by (see Featherstone et al. 1998):

$$\hat{s}_{\sigma_i}^2 = \sum\left[\left(u_i \times v_i^T\right)\otimes\left(u_i \times v_i^T\right)\otimes\hat{\Theta}_{\hat{G}}^2\right] \tag{4.22}$$

where $\hat{\Theta}_{\hat{G}}^2$ is a matrix of the estimated variance in each gain matrix element

$\sum$ is a summation of all the elements of the resulting matrix

$\hat{s}_{\sigma_i}$ is the estimated standard deviation of the ith pseudo-singular value

Certainly the pseudo-singular value is not normally distributed and the assumption that U and V have no uncertainty is not accurate. In order to take these issues into account, when estimating the uncertainty in the pseudo-singular value, a Bootstrapping or Jackknifing method (Efron and Tibshirani 1993, Shao and Wu 1989, and Wehrens et al. 2000) can be used to estimate the uncertainty more accurately.

Equation (4.22) or a Jackknifing method can be used to rank the pseudo-singular values from the most probable to the least probable to change sign. This can be utilized to determine which process ill-conditioning (pseudo-singular value) may be the cause of the non-square system going unstable (let this pseudo-singular value be $\sigma_{D,q}$). Using this knowledge, the elements in the gain matrix that contribute the most to this singular value can be isolated:

$$\Gamma_q = u_q \times v_q^T \tag{4.23}$$

where Γ_q is the contribution matrix of the gain matrix with the same dimensions as G associated with the ith pseudo-singular value. At this stage, it may be possible to isolate a sub-matrix of Γ_q that has a high contribution to the small singular value. If this sub-matrix is square, a constraint can be placed on the determinant of this sub-matrix and the model may be re-estimated, with the posterior knowledge that the controller was unstable with the first estimated model:

$$\begin{aligned} &\underset{\hat{\hat{G}}}{Min}\, SSE \\ &s.t. -\det\left(\hat{\hat{G}}_r\right) \times sign\left(\det\left(\hat{G}_r\right)\right) > 0 \end{aligned} \tag{4.24}$$

where SSE is the Sum of Square Error in the prediction

$\hat{G}, \hat{\hat{G}}$ are the first and second estimates of the gain matrix (G)

$\hat{G}_r$ is the sub-matrix of the first estimated gain matrix ($\hat{G}$) that is found to have a high contribution to $\hat{\sigma}_{D,q}$

$\hat{\hat{G}}_r$ is the second estimate of G_r

Perhaps a more common situation is that the effect of the singular value in question cannot be isolated to a square sub-matrix. In this case, the complete non-square gain matrix should be considered. One approach to this problem is to implement $\det\left(\hat{G}\hat{\hat{G}}^T\right) > 0$ (where $\hat{G}$ is the first estimate, and $\hat{\hat{G}}$ is the second estimate) as a constraint. This is similar to flipping the sign in the square problem. The derivation of "why" $\det\left(\hat{G}\hat{\hat{G}}^T\right) > 0$ should result in a stable model is explained in Appendix 12.

$$\begin{aligned} &\underset{\hat{\hat{G}}}{Min}^{SSE} \\ &s.t. \det\left(\hat{G}\hat{\hat{G}}^T\right) > 0 \end{aligned} \tag{4.25}$$

Similar to flipping the sign of the square problem, this condition is a necessary, but not a sufficient condition for CSC of non-square systems. As a result, the optimization solution resulting from (4.25) will not always result in a SCS. This optimization is similar to (4.9), namely that in most cases it will result in a stable system (but not always).

4.5.1. Geometric interpretation of the non-square problem

Contrary to the square problem where the geometrical interpretation of the CSC was straightforward, in the non-square problem the geometrical interpretation is quite complicated (see Appendix 12). Only in the simple case of one process output can the CSC be explained geometrically.

4.6. Other issues

In both cases, non-square and square, it was implicit that the relation between inputs and the outputs are linear and time-invariant (LTI). This is a crucial assumption, since non-linearities (or time-variance) can cause changes in eigenvalue signs and would result in an alteration of the posterior knowledge. For example, it is feasible that for a particular operating condition $\left[\det(G_1)\right]_{\text{Region }A} < 0$ while in a different operating region $\left[\det(G_1)\right]_{\text{Region }B} > 0$. This would result in a controller having two posterior experiences that contradict one another. If the prior knowledge exists that the system is fundamentally non-linear and the sign of $\det(G)$ can change based on the operating region, a truly adaptive controller can be implemented. In such a case when there is a contradiction between two different posterior behaviors, the oldest posterior knowledge is removed from the optimization problem.

It is important to note that the ease of application for these different optimization problems ((4.24) and (4.25)) are not equivalent. For the same problem, the number of parameters involved in the non-linear constraint (on the determinant) can be different (compare (4.24) and (4.25)). In other cases, the optimization problem is discontinuous (which is the case for (4.8)). This discontinuous optimization problem would be a hard optimization problem to solve. Consequently, in practice, some of these optimizations would require substantial time in both problem formulation and computational time required for solving the problem.

If the process engineer has other prior knowledge, these can also be added to this framework. This would be accomplished by addition of constraints to all of the optimization problems mentioned up to this point. The different types of possible prior knowledge that may exist in a chemical process has been discussed extensively in the previous chapters. As mentioned in the previous chapter, the process engineer should consider the sensitivity of the added prior knowledge on the solution before using the prior knowledge to the optimization.

4.7. Other Applications

Although the main part of this chapter deals with adding constraints on the determinant during model re-estimation in order to improve controller stability in two situations (when the estimated model has the wrong sign of an eigenvalue and when the sub-system becomes active due to a constraint on a variable and violates CSC), the idea has other applications as well. In this section, two other applications of this idea are presented. The first deals with the problem of sensitivity of the controller, when a manipulated variable or a controlled variable fails due to a hardware failure. The second application deals with the situation when additional data is collected for model re-estimation. In this case, both the original estimated model and the controller performance based on this model are used as a form of posterior knowledge in model re-estimation.

4.7.1. Sensor or Actuator Failure

The concept of sensor or actuator failure and its effect on control systems has been studied extensively (Morari et al. 1983; Chang and Yu 1991; Grosdidier et al. 1985). For sensors, this type of problem may be due to a burned out thermocouple, pressure transducer, etc. In the case of actuators, there could be a burned out heater, a stuck valve, etc. These types of failure result in either an irregular signal being sent to the controller or the controller action not being implemented. In either case, the controller does not have accurate information about the state of the system. This can lead to dangerous situations. The solution to this problem in industrial situations is to put the controller in to either offline mode or failure mode (Grosdidier et al. 1985, Morari 1983). In the failure mode, it is desired that the controller maintain stability, despite some of its sensors or actuators failing.

Multiple issues are involved in this problem. First, the failure (or fault) has to be detected. This can be accomplished either by hardware (i.e., temperature measurement of zero indicates failure in the thermocouple) or fault detection software (i.e., erratic changes in the temperature or other patterns or correlations may signal a fault with the thermocouple). Next, the controller in its failure mode will need to control the system in a reduced space (since it has less inputs or outputs). The fundamental question is whether this controller will maintain stability. This has resulted in analysis of sensor failure sensitivity (SFS) and actuator failure sensitivity (AFS) (Grosdidier et al. 1985; Chang and Yu 1991), which combined are termed "Failure Tolerance" by Morari (1983). The more the controller can handle sensor or actuator failure the more "resilient" it is said to be (Morari 1983).

The following assumptions are usually made in any discussion of controller "resilience":

-It is assumed that there is insufficient time for the controller to be re-tuned.

-The sensor or actuator failure has been recognized and removed from the controller calculations.

-Controlled variables are ranked in the order of importance. Consequently, if the controller does not have sufficient degrees of freedom to control all the control variables, it would remove the least important controlled variable.

-The true process gain (G) is not singular.

To see the consequences when certain manipulated or controlled variables are lost, consider the following "switching" matrix:

$$\Delta = diag\{\varsigma_1, \varsigma_2, \cdots, \varsigma_n\} \tag{4.26}$$

where $\varsigma_i = \begin{cases} 1, i \notin J, \text{normal operation of variable } i \\ 0, i \in J, \text{variable } i \text{ is lost} \end{cases}$

J is a subset of variables that have failed. Note that there will be J_u and J_y corresponding to the input and the output respectively

Then the resulting system, which incorporates the failure of sensors and actuators, is given by (subscript u and y represent the input and the output):

$$G_f = \Delta_y G \Delta_u \tag{4.27}$$

where G is the original full dimension gain matrix, which is square

G_f is the systems gain matrix incorporating the failures

When there are no failures in the inputs or the outputs, $\Delta_u = \Delta_y = I$ and $G_f = G$. If the rows and columns that are zero in G_f are removed, the resulting matrix is G_r. For a particular set of failure descriptors (J_u, J_y), if the resulting controller is stable, the CSC has not been violated, and the sign of $\det(\hat{G}_r)$ is correct. However, if the controller is unstable, it is assumed that the CSC is violated and the sign of $\det(\hat{G}_r)$ is incorrect. Similar studies have been performed by other researchers, where low (zero) frequency instability of the controller due to sensor or actuator failure is studied (Chang and Yu 1991). They have shown that the control system is stable if the eigenvalues of $(G_f \hat{G}_f^{-1})$ are in the RHP (Right Half Plane), which in essence is the same as applying the CSC to the reduced system.

Sensor or actuator failure is similar to active constraints on the process input or output. Consequently, the conclusions and results made in 4.4 and 4.5 are applicable to this situation. Namely, multimodel and adaptive controllers as described in 4.4.3.1 and 4.4.3.2 respectively can be used to handle sensor and actuator failures, since the reduced system that results from activation of input or output constraints is the same as failure of sensor or actuator.

4.7.2. Incorporating of New Data Sets

Most of the research in the area of model re-estimation (or iterative model identification) has been concerned with model re-estimation during the identification phase. The approach has been to either: iteratively identify a closed-loop system and then re-design the controller (Schrama and Van den Hof, 1992), or iteratively re-design the perturbation signal used in the identification experiment based on the re-estimated model (Koung et al. 1994, Cooley 1997, and Li et al. 1996). In industrial processes after the original identification (for control) is performed, the controller based on the estimated model is implemented. Traditionally, if the controller performance is poor, the control engineer performs a second test. The test signal used in the second experiment could be designed to perturb the smallest singular value of the process in a D-optimal design (Koung and MacGregor 1994), to perturb a particular frequency range by performing closed-loop identification (Söderström and Stoica 1987), or it could be simply an uncorrelated PRBS with a larger magnitude. In all of these methods, the posterior knowledge about the controller performance after the first identification process is not used in the subsequent identification. In this section, this posterior knowledge is used in the estimation of the new model. Since this posterior knowledge and its inclusion in model re-estimation is independent of the test signal used, the issue of the test signal design and its implications are not discussed here.

While poor controller performance may be one reason for performing a second identification test, there may be other reasons for collecting more data. One reason is if a new manipulated or controlled variable is added to the process. Since an experiment has to be performed on one of the MVs or on all the MV's related to the new CV, one may want to take this opportunity to re-estimate the other relationships better. Typically in such a system, there are the following characteristics (which is used as the basis of this work):

-There are 2 data sets.

-The first data set contains no measurement for one or more of the MVs or CVs.

-In the second data set there is a measurement for all the MVs and CVs.

After the second data set has been collected, a new model with larger dimension than the original model has to be estimated. The new model has to incorporate information from both of the data sets. This can be accomplished in several different ways. The first data set was used to estimate the original (reduced) system ($\hat{G}_r$), which is now a part of the new (full) system ($\hat{G}_f$). Then the full system can be estimated with the second data set, by presetting the parameters that correspond to the reduced system ($\hat{\hat{G}}_r = \hat{G}_r$, where, $\hat{G}_r, \hat{\hat{G}}_r$ are the estimated model based on the first and second data sets respectively). This would result in not a very good model, since $\hat{G}_r$ is independent of the second data set. A number of variations of this method are also plausible (which would result in a better model). For example, in the second model estimation, instead of presetting the exact values of the reduced system, they can be set as an upper and lower bound ($\hat{G}_r - 2\Theta < \hat{\hat{G}}_r < \hat{G}_r + 2\Theta$, where Θ is matrix of the estimated standard deviation in each gain matrix element based on the first data set), or as a soft (stochastic) constraint that incorporates similar information. To implement the soft constraint, the objective function has to be augmented to incorporate the posterior knowledge. A similar method would be to minimize the total SSE (that is based on the first data set (*SSE$_r$*) and the second data set (*SSE$_f$*)) over both data sets simultaneously, while also incorporating knowledge about controller stability from the first data set in the reduced system. Incorporating the posterior knowledge about the CSC in this minimization problem will result in:

$$\underset{\hat{\hat{G}}_f}{Min}\left(SSE_r + SSE_f\right) \tag{4.28}$$

$$s.t. -\det\left(\hat{\hat{G}}_r\right) \times sign\left(\det\left(\hat{G}_r\right)\right) > 0$$

where $\hat{G}_r$ is the reduced systems gain estimate based on the first data set

$\hat{\hat{G}}_f$ and $\hat{\hat{G}}_r$ are the full and reduced systems gain estimates, respectively, based on both data sets

Variations of (4.28) can also be considered in which the SSE is from the two data sets are weighted differently. This may be done if the first data set was collected long ago and is less relevant. Alternatively, if the process is thought to be non-linear, the operating region for one of the data sets may not be as relevant. In addition, if other prior knowledge about this system exists, they may be added to (4.28) as constraints.

Example 4.5: Consider the gain matrix of Example 4.1.

$$G_r = \begin{pmatrix} -0.3165 & -2.4189 & -0.4263 \\ -0.5825 & -2.0410 & 0.2045 \\ -0.9249 & -0.4536 & 1.4856 \end{pmatrix}$$

Assume that this is the original (reduced) system. To each process output an i.i.d. white noise with variance of 1 is added (N(0,1)). An identification experiment is performed where the process inputs are perturbed with independent PRBS's of magnitude ±0.5 and 500 observations are collected. This results in a data set where the signal-to-noise ratio in terms of standard deviation ($\sigma_{Signal}/\sigma_{Noise}$) for each output is 1.2382, 1.0662, and 0.9039, respectively. The estimated gain matrix based on one such data set is:

$$\hat{G}_{r,1} = \begin{pmatrix} -0.3374 & -2.2811 & -0.3941 \\ -0.5923 & -1.8835 & 0.1179 \\ -0.9493 & -0.2416 & 1.4065 \end{pmatrix} \pm \begin{pmatrix} 0.1276 & 0.1666 & 0.1826 \\ 0.2699 & 0.0935 & 0.2296 \\ 0.2122 & 0.1981 & 0.2482 \end{pmatrix}$$

Since this model satisfies the CSC, the resulting controller would be stable:

$$\lambda_i\left(G_r \times \hat{G}_{r,1}^{-1}\right) = \{0.1275 \quad 0.9152 \quad 1.0142\} > 0, \forall i$$

Now let a new manipulated and a new controlled variable be added to the system (the first row and column represent this addition). The true gain matrix for the (4x4) system is:

$$G_f = \begin{pmatrix} -1.0059 & -0.4237 & -2.4609 & -0.2678 \\ 0.6831 & -0.3165 & -2.4189 & -0.4263 \\ -0.3019 & -0.5825 & -2.0410 & 0.2045 \\ -1.4155 & -0.9249 & -0.4536 & 1.4856 \end{pmatrix}$$

The variance of the noise added to the new output is 1 (N(0,1)). In order to estimate a new model for the new (full) problem, 500 new observations were collected (PRBS magnitude was ±0.5 for all inputs, with the resulting signal-to-noise ratio ($\sigma_{Signal}/\sigma_{Noise}$) of 1.3527, 1.2845, 1.0768, and 1.1480 for each output). The resulting model using only the second data set is:

$$\hat{G}_{f,1} = \begin{pmatrix} -0.9554 & -0.3869 & -2.4129 & -0.2711 \\ 0.6761 & -0.2212 & -2.2931 & -0.5948 \\ -0.3706 & -0.5697 & -2.0462 & 0.3173 \\ -1.4352 & -0.9084 & -0.6111 & 1.4829 \end{pmatrix} \pm \begin{pmatrix} 0.1802 & 0.1464 & 0.2206 & 0.1530 \\ 0.2718 & 0.1026 & 0.1278 & 0.2131 \\ 0.1320 & 0.1292 & 0.1708 & 0.2101 \\ 0.1887 & 0.1392 & 0.2388 & 0.1895 \end{pmatrix}$$

This new model based on the second data set alone will result in an unstable controller:

$$\mathrm{Re}\left(\lambda_i\left(G_f \times \hat{G}_{f,1}^{-1}\right)\right) = \mathrm{Re}\left(\{-0.0213 \quad 1.1162 \quad 1.0510 \pm 0.0497i\}\right) \not> 0, \forall i$$

If both data sets are combined together, similar to (4.28) but with no constraint on the determinant, the resulting model will be:

$$\hat{G}_{f,2} = \begin{pmatrix} -0.9554 & -0.3869 & -2.4129 & -0.2711 \\ 0.6782 & -0.2810 & -2.2884 & -0.4931 \\ -0.3727 & -0.5753 & -1.9667 & 0.2187 \\ -1.4371 & -0.9197 & -0.4277 & 1.4473 \end{pmatrix} \pm \begin{pmatrix} 0.1802 & 0.1464 & 0.2206 & 0.1531 \\ 0.2717 & 0.0478 & 0.0570 & 0.0998 \\ 0.1280 & 0.0579 & 0.0792 & 0.1007 \\ 0.1909 & 0.0701 & 0.1121 & 0.0896 \end{pmatrix}$$

Once again this model will not result in a stable controller for the full system:

$$\mathrm{Re}\left(\lambda_i\left(G_f \times \hat{G}_{f,2}^{-1}\right)\right) = \mathrm{Re}(\{-0.0844 \quad 1.0868 \quad 1.0153 \pm 0.0225i\}) \not> 0, \forall i$$

However, if the posterior knowledge that the controller performance of the reduced system is stable and the model is re-estimated using this knowledge (as shown in equation (4.28)) then the resulting model using both data sets is:

$$\hat{G}_{f,3} = \begin{pmatrix} -0.9554 & -0.3869 & -2.4129 & -0.2711 \\ 0.6782 & -0.2710 & -2.2908 & -0.4878 \\ -0.3727 & -0.5881 & -1.9637 & 0.2118 \\ -1.4371 & -0.9145 & -0.4290 & 1.4502 \end{pmatrix} \pm \begin{pmatrix} 0.1802 & 0.1464 & 0.2206 & 0.1531 \\ 0.2716 & 0.0699 & 0.0607 & 0.0848 \\ 0.1276 & 0.0287 & 0.0817 & 0.0912 \\ 0.1907 & 0.0656 & 0.1116 & 0.0842 \end{pmatrix}$$

Which satisfies the CSC and will result in a stable control system:

$$\mathrm{Re}\left(\lambda_i\left(G_f \times \hat{G}_{f,3}^{-1}\right)\right) = \mathrm{Re}(\{0.1357 \quad 1.1433 \quad 0.9824 \quad 1.0318\}) > 0, \forall i$$

Combining the two data sets lowers the variance in the original (reduced) gain matrix elements, since the two data sets combined contain 1000 observations that can potentially be used to estimate G_r. However, combining the two data sets has little effect on the gain matrix elements associated with the new manipulated and controlled variables. More importantly, the combination of the two data sets by themselves is not sufficient to produce a stable controller. The posterior knowledge of the controller performance in the reduced space, with the first model, has to be utilized to produce a stable controller in the full space. This can be attributed to the second data set producing an estimate of the smallest eigenvalue with the incorrect sign. The sign of this eigenvalue remains incorrect even after the addition of the first data set. Only the posterior knowledge about the controller performance or addition of more data can produce a stable controller.

It is important to note that the above example is just one example, and it is possible that even after incorporating the posterior knowledge the full system could remain unstable. In such a case, another constraint can be added to (4.28) in the form of a constraint on the full determinant. As mentioned earlier, there is always the possibility that the constraint on the determinant may not result in a stable system (i.e., the Monte Carlo simulations in Table 4.1 showed a success rate of 94.5% not 100%).

4.8. Conclusions

This chapter is concerned with posterior (or acquired) knowledge, in contrast to previous chapters where the issue of prior knowledge is studied. In particular, this chapter is concerned with posterior knowledge about the actual controller stability when it uses the estimated model. Such posterior knowledge is then utilized in model re-estimation.

Practitioners and researchers in the past have dealt with the problem of control of an ill-conditioned system by either removing the low-gain direction from the control problem (MacArthur 1996) or estimating the low-gain direction more accurately (Koung and MacGregor 1994). In the first situation, the controller will not be capable of controlling the system in the low-gain direction, while in the second situation, the controller will be capable of limited control in the full output space; however, the second approach requires the collection of more data in a very specific re-identification phase (compared to the case when the posterior knowledge about the controller stability is used). The key benefit in incorporating such posterior knowledge about the controller stability into the re-estimated model is that the controller will be capable of limited control in the full output space without the need to collect more data. It is important to note that while the addition of the posterior knowledge will improve the model quality based on the CSC, it may not improve the model quality based on other metrics. Collection of more data in the identification phase, as suggested by Koung and MacGregor (1994) will lower uncertainty in model parameters (i.e., improve model quality based on a different metric).

As different constraints on the process input and output become active, the process moves into different sub-spaces. Two different control designs are suggested that would incorporate the posterior knowledge about the controller stability in a particular sub-space into the estimated model. The first style of controller is referred to as a multimodel controller, where different models are used for control in different sub-spaces resulting in multiple models being used by the controller. In the second approach, the same model is used independent of which sub-space the controller is operating in. In this approach, the model is adaptively changed as more posterior knowledge about the controller stability is gained. The advantages and disadvantages of each approach have been discussed. This approach is also suited to incorporating knowledge of sensor/actuator failure, and to combining identification data sets, which contain a different number of MVs or CVs.

Table 4.1: The effect of model fix-up on random matrices

Magnitude of the PRBS is 1, there are 100×[number of system outputs] observations in each realization, and the standard deviation of the added noise is 0.1. This results in an average signal-to-noise ($\sigma_{Signal}/\sigma_{Noise}$) ratio for the 5x5, 10x10, and 20x20 of 22, 31, and 45 respectively (for other simulation settings see Figure 4.5)

	System Size	5x5	10x10	20x20
	Number of Monte Carlo realizations [2]	105880	10000	510
	Number of observations in each realization	500	1000	2000
	Median of the condition number of the gain matrix	2602	2094	2019
Before Fix-up [3]	%Unstable [4]	40.3%	27.3%	15.1%
	95% C.I. [1]	0.3%	0.9%	3.2%
	%Stable	59.7%	72.7%	84.9%
	95% C.I.	0.3%	0.9%	3.2%
After Fix-up	%Unstable	2.9%	0.7%	2.6%
	95% C.I.	0.2%	0.3%	3.6%
	%Stable[6]	94.5%	93.1%	84.4%
	95% C.I.	0.2%	1.0%	8.3%
	%Optimization Failing [5]	2.6%	6.2%	13.0%
	95% C.I.	0.2%	0.9%	7.7%
	Adjusted %Stable[7]	97.0%	99.2	97.0%

[1] C.I. is the confidence interval assuming normal distribution

[2] Unfortunately, the number of Monte Carlo realizations was not the same in the

different cases due to time and computer limitations

[3] The stability analysis of the full matrix before the fix-up was implemented

[4] The % of unstable system is the ratio of the number of simulations that failed the CSC to the total number of Monte Carlo realizations

[5] The % of optimization failing is based on the number of optimizations that did not converge (for optimization convergence criteria see Appendix 11)

[6] Note that the % stable appears to be 94.5% for all the system sizes considering the C.I.

[7] The adjusted % stable has been calculated by removing the % of failed optimizations from the calculation (i.e., Adjusted % Stable = % Stable/(1- %Optimization Failing))

Table 4.2: Effect of signal-to-noise ratio on effectiveness of model fix-up

	PRBS Magnitude	0.1	1	10
	Number of Monte Carlo Realizations [2]	1000	105880	540
Before Fix-up [3]	%Unstable [4]	51.3%	40.3%	10.7%
	%Stable	48.7%	59.7%	89.3%
	95% C.I. [1]	3.2%	0.3%	2.7%
After Fix-up	%Unstable	16.8%	2.9%	1.7% [7]
	%Stable	83.2%	94.5%	98.3% [7]
	%Optimization Failing [5]	0.0%	2.6%	0.0% [7]
	95% C.I. [6]	3.3%	0.2%	3.4%

[1] C.I. is the confidence interval assuming normal distribution and applies to both the unstable and stable (before fix-up) case

[2] Unfortunately, the number of Monte Carlo realizations was not the same in the different cases due to computer limitations

[3] The stability analysis of the full matrix before the fix-up was implemented

[4] The percentage of unstable systems is the ratio of the number of simulations that failed the CSC to the total number of Monte Carlo realizations

[5] The % of optimization failing is based on the number of optimizations that did not converge

[6] C.I. is the confidence interval assuming normal distribution and applies to the unstable, the stable and optimization failing (after fix-up) case. It does not apply to the cases where the probability is 0%.

[7] In this case, for each Monte Carlo realization, when the optimization failed, the optimization problem was retried with a different initial guess. This was done to increase the probability of finding a solution to the optimization problem.

Table 4.3: Stability analysis of the reduce system [1]

		Percentage	95% Confidence Interval
Reduced System	%Stable	84.5%	0.4%
	%Unstable	15.5%	0.4%
After Fix-up of the Full and Reduced System	%All Stable	90.2%	0.8%
	%Unstable-full and Stable-Reduce	0.3%	0.1%
	%Stable-full and Unstable-Sub	1.3%	0.3%
	%All Unstable	0.8%	0.2%

[1] This is the reduced system from the 5x5 system shown in second column of Table 4.1

Chapter 5

Direct and two-step methods for closed-loop identification a comparison of asymptotic and finite data set performance

5.1. Introduction

Many of the key theoretical and practical issues in closed-loop identification, such as identifiability conditions and identification methods in the frequency and time domains, were formulated in the early 1970's (Akaike 1968, Söderström et al. 1975, Box and MacGregor 1974, 1976, Caines and Chan 1975). While these early works focused on identification of adequate models describing the true process and disturbance, recent works have focused on the advantage of using closed-loop identification experiments to provide more robust control (Hjalmarsson et al. 1996).

In closed-loop identification, as in any identification problem, the choice of adequate model structures to characterize the process dynamics and the disturbance is an important issue. Iterative model building approaches involving model structure identification, parameter estimation and model checking (Box and MacGregor 1974, Box and Jenkins 1976, and Ljung 1987), can produce efficient, unbiased parsimonious models from either open or closed-loop data. The difficulty associated with identifying and checking for adequate parsimonious model structures can be avoided by using non-parsimonious finite impulse or step response models, with large number of parameters. Because of the large number of parameters, the bias is minimized, but model variance is increased compared to parsimonious models.

Most methods for identifying models using data from closed-loop experiments are variants of the following three approaches:

i) direct identification using prediction error methods to directly fit input/output models to the closed-loop data,

ii) indirect identification where a model is built between the output and the external variable exciting the process, and then the process model is calculated using prior knowledge of the controller equation,

iii) joint input/output identification where both the input and output variables are modeled as a function of the external exciting variable and the disturbance innovations, and then the process and disturbance models are extracted.

Details on all these approaches are given in Ljung (1987), Söderström and Stoica (1989). More recently, a series of two-step identification procedures have been proposed as a variants of the joint identification approach (Van den Hof and Schrama 1993, Huang and Shah 1997). In these approaches the joint input/output identification problem is broken into two open-loop identification problems, the first problem is to fit a model to the input to yield an estimate of the closed-loop sensitivity function, and then, using this estimate to filter the input or output, the second problem is to identify the process model from the filtered data. By breaking up the identification problem into two open-loop problems, the two-step methods have also been shown to be asymptotically unbiased. Direct identification also gives asymptotically unbiased results, provided that adequate disturbance and transfer function model structures are used, and identified simultaneously (Ljung 1987, MacGregor and Fogal 1995). Another motivation that is claimed for the two-step approach is that a disturbance model is not need (Van den Hof and Schrama 1993, Huang. and Shah 1997). This follows from Ljung (1987) who proved that open-loop identification will give an asymptotically unbiased estimates of the process model even if the disturbance model is inadequate, provided that the disturbance is stationary. However, with finite data sets and for disturbances that approach nonstationarity, the two-step approach can produce biased results, and provide estimates with larger variance.

The main distinction between the direct and two-step methods is how they diminish the effect of the feedback correlation in the closed-loop data. Both methods achieve this by filtering the data. In the direct method, both the input and output data is filtered with the inverse of the estimated disturbance model. In the two-step methods, either the input data (1993) or the output data (Huang. and Shah 1997) is filtered with the estimated closed-loop sensitivity function. Both the direct and 2-step methods can be used to provide asymptotically unbiased results with parsimonious or non-parsimonious model structures, provided that the model structures contain the true process model and either the true disturbance model (direct method), or the true sensitivity function model (two-step method). Using non-parsimonious model structure makes both methods easier to use in practice, at the expense of increasing the variance of the parameter estimates.

The purpose of this chapter is to explore variance issues and some bias issues, when closed-loop identification is performed with non-parsimonious process models of sufficiently high order, by comparing the direct and two-step identification approaches in terms of their asymptotic properties, and their behavior with finite data sets. Section 2 presents the various estimation methods, and their asymptotic properties. Section 3 presents simulations studies used to evaluate these asymptotic properties, and to provide comparisons using finite data set. Discussion and conclusions are given in the final section.

5.2. Closed-Loop Identification Approaches

A simple closed-loop system is shown in Figure 5.1 where d_k is a designed external excitation (dither) imposed on top of the process input (or the set-point), and $D_k = H(q^{-1})e_k$ represents the unmeasured disturbances in the system during the identification experiment.

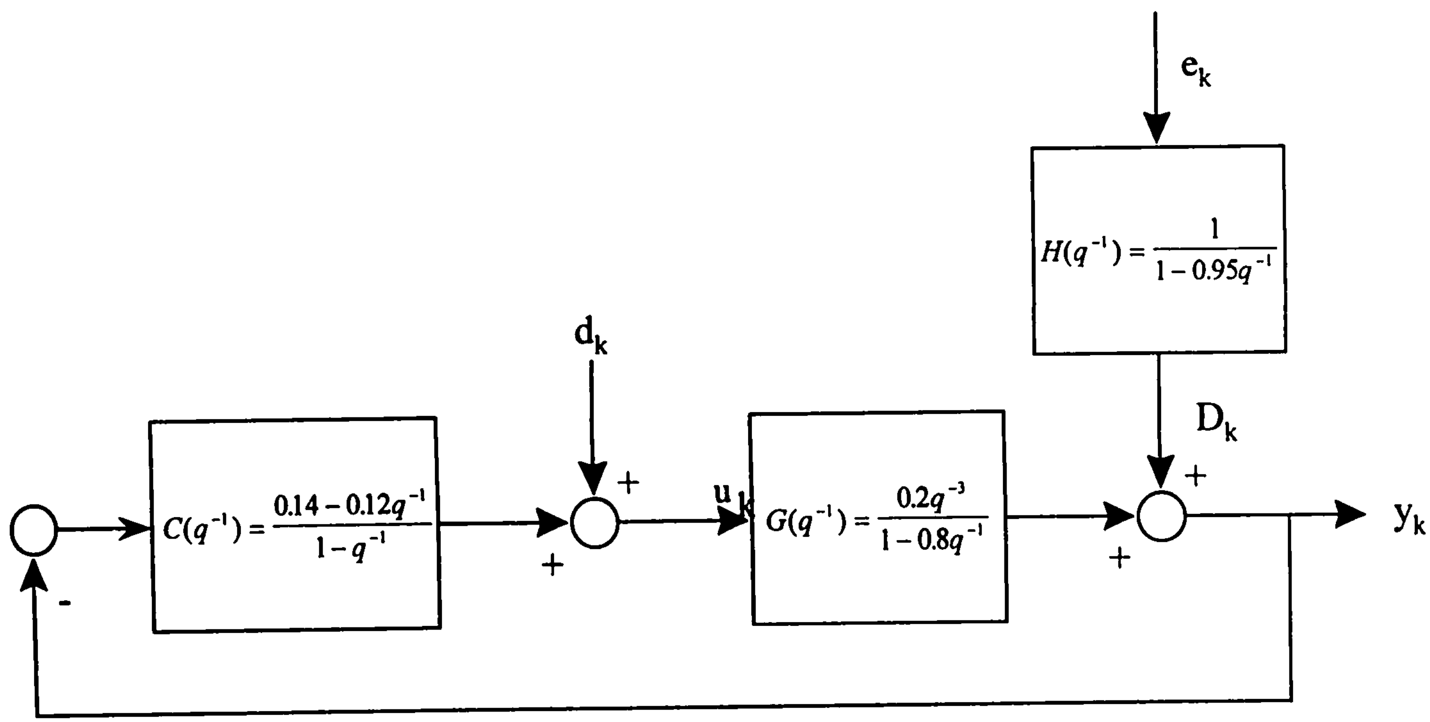


Figure 5.1: Closed-loop system used as the basis for the simulations

5.2.1. Parameter Estimation Algorithms

i) Direct Identification involves estimating the parameters in the process, $G(q^{-1})$, and disturbance, $H(q^{-1})$, models directly from closed-loop input/output data in exactly the same manner as done with open-loop data. For the model

$$y_k = G(q^{-1})u_k + H(q^{-1})e_k \tag{5.1}$$

the parameters in G and H are simultaneously estimated by the Prediction Error method

$$\underset{G,H}{Min}\sum_{k=1}^{n}(y_k - \hat{y}_{k|k-l})^2 = \sum_{k=1}^{n}\hat{e}^2_{k|k-l} \tag{5.2}$$

If the e_k are normally distributed with an adequate model structure for both $G(q^{-1})$ and $H(q^{-1})$, and the prediction horizon is l = 1, then this method provides Maximum Likelihood estimates of the model parameters. These parameter estimates are asymptotically unbiased and achieve the Cramer-Rao lower bound on variance for this model structure. If parsimonious model structures are used, then the lowest parameter variance is obtained, but attention must be paid to testing for the adequacy of the structures and iterating if inadequacies are found (Box and Jenkins 1976, Ljung 1987).

ii) Joint Input/Output Identification involves expressing the output and input variables in terms of the independent external exciting variables d_k and e_k as follows:

$$y_k = G(q^{-1})S(q^{-1})d_k + H(q^{-1})S(q^{-1})e_k \tag{5.3}$$

$$u_k = S(q^{-1})d_k + H(q^{-1})S(q^{-1})C(q^{-1})e_k \tag{5.4}$$

where $S(q^{-1}) = \left[1 + G(q^{-1})C(q^{-1})\right]^{-1}$ is the sensitivity function. If independently parameterized models are fitted by prediction error methods,

$$y_k = G_y(q^{-1})d_k + H_y(q^{-1})\varepsilon_{yk} \tag{5.5}$$

$$u_k = G_u(q^{-1})d_k + H_u(q^{-1})\varepsilon_{uk} \tag{5.6}$$

then the process and disturbance models G and H can be determined by comparing equations (5.5) and (5.6) with (5.3) and (5.4). For example

$$\hat{G}\left(q^{-1}\right) = \hat{G}_y(q^{-1})\left[\hat{G}_u(q^{-1})\right]^{-1} \tag{5.7}$$

since $\hat{G}_u(q^{-1})$ is an estimate of the sensitivity function $S(q^{-1})$.

iii) Two-Step Methods are variations of the joint method described above. As in the joint method, the model for the input in (5.6) is first fitted to the closed-loop data often assuming that $H_y(q^{-1}) = 1$. The estimate $\hat{G}_u(q^{-1}) = \hat{S}(q^{-1})$ is then used in the output equation (5.3) in either of two ways. Van den Hof and Schrama use it to filter the external dither signal as $d_k^F = \hat{S}(q^{-1})d_k$ and then fit the model

$$y_k = G(q^{-1})d_k^F + \varepsilon_k \tag{5.8}$$

by least squares. Huang and Shah[11] filter the output as $y_k^F = \left[\hat{S}(q^{-1})\right]^{-1} y_k$ and fit the model

$$y_k^F = G(q^{-1})d_k + \varepsilon_k \tag{5.9}$$

The essential difference between the joint input/output identification method and the two-step methods is the way the sensitivity function estimate $\hat{S}(q^{-1})$ is used. In the two-step methods the estimate of the sensitivity function $\hat{S}(q^{-1})$ is used directly to filter d_k, or its inverse is used to filter y_k, before fitting the model for y_k. In the joint input/output method a model is identified by fitting y_k and dividing by $\hat{S}(q^{-1})$ afterwards.

5.2.2. Parsimonious and Non-Parsimonious Models

In the last section, different methods of closed-loop identification were introduced. Each method can identify two different distinct classes of models: parsimonious and non-parsimonious models. Parsimonious models are typically low order models. They are called parsimonious because the number of parameters employed is small, obeying the principal of parsimony (Box and Jenkins 1976) (whereby the minimum number of statistically significant parameters are employed in a model, while still enabling it to represent the true process dynamics). Non-parsimonious models are high order models, which commonly are in the form of a finite impulse response (FIR) model:

$$G(q^{-1}) = \sum_{i=0}^{m} g_i q^{-i} \tag{5.10}$$

where m is large enough to adequately approximate the true process dynamics. The advantages of such a model are that: the only structural determination is choosing the number of impulse weights m, and they are not limited to the responses produced by low order (parsimonious) models.

In this chapter, all of the simulations utilize a non-parsimonious model in the form of a FIR, unless otherwise stated. In this way, we avoid bias issues arising from the choice of an inadequate process model. Any bias arising from the methods will then result from other issues such as inadequate models for the disturbance or sensitivity function. Therefore, in this chapter we are primarily focusing on variance issues of the closed-loop identification methods.

5.2.3. Asymptotic Variance Expressions

Asymptotic variance expressions for the various open and closed-loop identification methods have been developed. For open-loop identification Ljung (1985) developed the following expression

$$Var\left(G\left(e^{i\omega}\right)\right) = \frac{n}{N}\frac{\Phi_D(\omega)}{\Phi_u(\omega)} \tag{5.11}$$

where $\Phi_D(\omega)$ and $\Phi_u(\omega)$ are the spectra of the disturbance and input respectively. This expression shows that the asymptotic variance is proportional to the signal to noise ratio at any frequency. This expression, and those that follow for closed-loop identification are asymptotic in both N, the number of observations, and n, the model order. It therefore applies to non-parsimonious, high order models of the ARX and FIR type.

For closed-loop identification of G and H, Gevers et al. (1996) have shown that the corresponding asymptotic variance expression for $\hat{G}\left(e^{i\omega}\right)$ is

$$Var\left(G\left(e^{i\omega}\right)\right) = \frac{n}{N}\frac{\Phi_D(\omega)}{\left|S(\omega)\right|^2\Phi_d(\omega)} = \frac{n}{N}\frac{\Phi_D(\omega)}{\Phi_u^d(\omega)} \tag{5.12}$$

where $\Phi_u^d(\omega) = \left|S(\omega)\right|^2\Phi_d(\omega)$ is the spectrum of that part of the input signal arising from the external dither, i.e. $u_k^d = S_o(q^{-1})d_k$. Gevers et al. (1996) have shown that this asymptotic expression is valid for both the direct and joint input/output identification methods, and hence for the two-step counterparts of the latter.

5.2.4. Open-Loop vs. Closed-Loop Experiments

The above asymptotic results show that when the input power is limited, the $Var(\hat{G})$ will generally be larger for closed-loop experiments than for open-loop experiments. This follows from the fact that:

$$\Phi_u^d(\omega) < \Phi_u(\omega) \qquad \forall \omega \tag{5.13}$$

However, one purpose of performing experiments under closed-loop is to limit the output power. Under this limitation the situation is much different. Consider the output spectra under open and closed-loop operation.

Open-Loop: $$\Phi_y = |G|^2 \Phi_u + \Phi_D \tag{5.14}$$

Closed-Loop: $$\Phi_y = |G|^2 |S|^2 \Phi_d + |S|^2 \Phi_D \tag{5.15}$$

If we were to choose the spectrum of our designed dither signal for closed-loop identification d_k to be such that its contribution to the input u_k^d was equal to the input signal applied in the open-loop situation, i.e.

$$\Phi_u^d = |S|^2 \Phi_d = \Phi_{u|open-loop} \tag{5.16}$$

then the asymptotic variances of $\hat{G}$ given by equations (5.11) and (5.12) would be identical in both the open and the closed-loop identification. However, under this condition the variance of the output signal will generally be smaller in the closed-loop case. Comparing equations (5.14) and (5.15) under condition (5.16), the components of Φ_y arising from the input signal (first term) will be identical in both the open and closed-loop case under the design (5.16). However, the contribution to the output spectrum Φ_y arising from the disturbance (second term) will be less in the closed-loop case at all frequencies where $|S|^2 < 1$. Since, by virtue of the controller design, $|S|^2$ is generally less than one over most of the important low and intermediate frequencies, this implies that the output variance under closed-loop identification will be lower in this frequency range and in general the overall $Var(y) = \frac{1}{2\pi}\int \Phi_y e^{i\omega} d\omega$ will be lower in the closed-loop experiment.

Therefore, when the output power is limited, closed-loop identification will generally give better identification (lower $Var(\hat{G})$ for the same output variance) than open-loop identification.

5.3. Simulation Studies

The objective of the simulation studies is to investigate some of the assumptions made in the derivation of the asymptotic expressions (5.11) and (5.12), and to examine the precision of $\hat{G}$ in non-asymptotic situation (limited data length and low signal to noise ratio) for several of the identification methods.

5.3.1. Base Case Simulation

The base case simulation is a simple first order process with a gain of 1, a time constant of 4.46, a dead time of 3, and the sampling interval of 1. The simulated transfer functions of the process and the noise model are given respectively by:

$$G(q^{-1}) = \frac{0.2q^{-3}}{1-0.8q^{-1}} \tag{5.17}$$

$$H(q^{-1}) = \frac{1}{1-0.95q^{-1}} \tag{5.18}$$

The controller of this process is a detuned PI controller with a gain of 0.12 and the integral time of 6. The controller transfer function is:

$$C(q^{-1}) = \frac{0.14-0.12q^{-1}}{1-q^{-1}} \tag{5.19}$$

The external excitation is a Pseudo-Random Binary Signal (PRBS) (Ljung 1987, Söderström and Stoica 1989) with a switching time (basic period) of 3 sampling intervals and a magnitude of ±2 in all the simulations. Unless otherwise specified, 5000 data points were collected under closed-loop conditions. The white noise used as the input to the noise model is normally distributed with a variance of 1. The signal to noise ratio is defined as the ratio of the effect of the external excitation (which assists in identification) on the input of the process to the effect of the white noise sequence (which hinders identification) on the input of the process. In the base case simulation the ratio of the signal to noise at the input and output in terms of ratios of standard deviations were:

$$\sqrt{\frac{\mathrm{var}(Sd)}{\mathrm{var}(SCHe)}} = 1.16 \quad \text{or} \quad \sqrt{\frac{\mathrm{var}(SGd)}{\mathrm{var}(SHe)}} = 0.38 \tag{5.20}$$

A set of 100 different input signal realizations (with 5000 data points in each realization) results in a Monte Carlo type simulation. For each of the 100 data sets the process was identified by applying the following four identification methods:

i) direct method with an AR(3) noise model

ii) two-step method of Van den Hof and Schrama

iii) two-step method, given the true sensitivity function

iv) two-step method, given the true sensitivity function and estimating an AR(3) noise model

The first two methods compare the two-step method, as formulated by Van den Hof and Schrama (1993), with the direct method of closed-loop identification. The third method eliminates the uncertainty caused through estimation of the sensitivity function, by providing the correct sensitivity function for the second step of the two-step approach. The last method investigates the advantages of simultaneously estimating a noise model and its effect on the estimated transfer function model.

In the cases where the estimate of the sensitivity function was required, a 15 parameter finite impulse response (FIR) model was implemented unless otherwise specified. This was found to be adequate by further simulations. The structure of the non-parsimonious dynamic model was a FIR model. For FIR models "Ordinary Least Squares" was used to estimate the model parameters. When a noise model was simultaneously estimated along with the FIR model, the "Generalized Least Squares" method was used. In the case of parsimonious models an iterative "Gauss-Newton" algorithm was utilized. (For more information on parameter estimation methods see Ljung 1987, Söderström and Stoica 1989).

The different methods of closed-loop identification were performed using each of the 100 sets of data independently. The means of the estimated parameters from the 100 identifications were used to obtain the Nyquist plot of the estimated model. The approximate 100(1-α)% joint confidence regions for the real and imaginary components in $G(e^{i\omega})$ at different frequencies were determined by:

$$\left(\hat{G}(e^{i\omega})-\overline{G}(e^{i\omega})\right)^T \hat{\Pi}^{-1}(\omega)\left(\hat{G}(e^{i\omega})-\overline{G}(e^{i\omega})\right) \leq \frac{(N-1)p}{N(N-p)}F_{\alpha;p,N-p} \tag{5.21}$$

where $\hat{G}(e^{i\omega})$ is the estimated transfer function

$\overline{G}(e^{i\omega})$ is the mean of $\hat{G}(e^{i\omega})$ over the 100 simulations

$\hat{\Pi}$ is the sample covariance matrix of $\hat{G}(e^{i\omega})$, given by:

$$\hat{\Pi}(\varpi) = \frac{1}{N}\sum_{i=1}^{N}\left(\hat{G}_i(e^{i\varpi}) - \overline{G}_i(e^{i\varpi})\right)\left(\hat{G}_i(e^{i\varpi}) - \overline{G}_i(e^{i\varpi})\right)^T$$

p is the degree of freedom

N is the number of Monte Carlo simulations (100)

α is the level of significance used in the test (0.05 in this study)

It was assumed here that an elliptical (Normal theory) confidence region described the confidence contours adequately. This is a reasonable assumption based on the simulation results shown in Figure 5.2. In this Figure, each input signal realization results in one point at each frequency. A more exact method of determining the confidence region could be by performing Likelihood profiling as is described by Watts (1994).

The evaluation of the precision of the identified models was based on the area of the joint confidence regions at the different frequencies in the Nyquist plots. The areas of the confidence regions are proportional to the determinant of $\hat{\Pi}(\omega)$ in (5.20). A Nyquist plot illustrating the variation in the joint confidence region for an estimated parsimonious model using the direct method is shown in Figure 5.2.

For a medium sized data set of 1000 observations, Figure 5.3 shows a Nyquist plot of the parameter means and the confidence regions for a non-parsimonious 15 parameter FIR model using different methods of closed-loop identification. The area of the contours are representative of the uncertainty of the estimated model at the given frequency. In the next section a series of plots similar to Figure 5.3 are compiled together to study the effect of number of observations and model order on different methods of closed-loop identification.

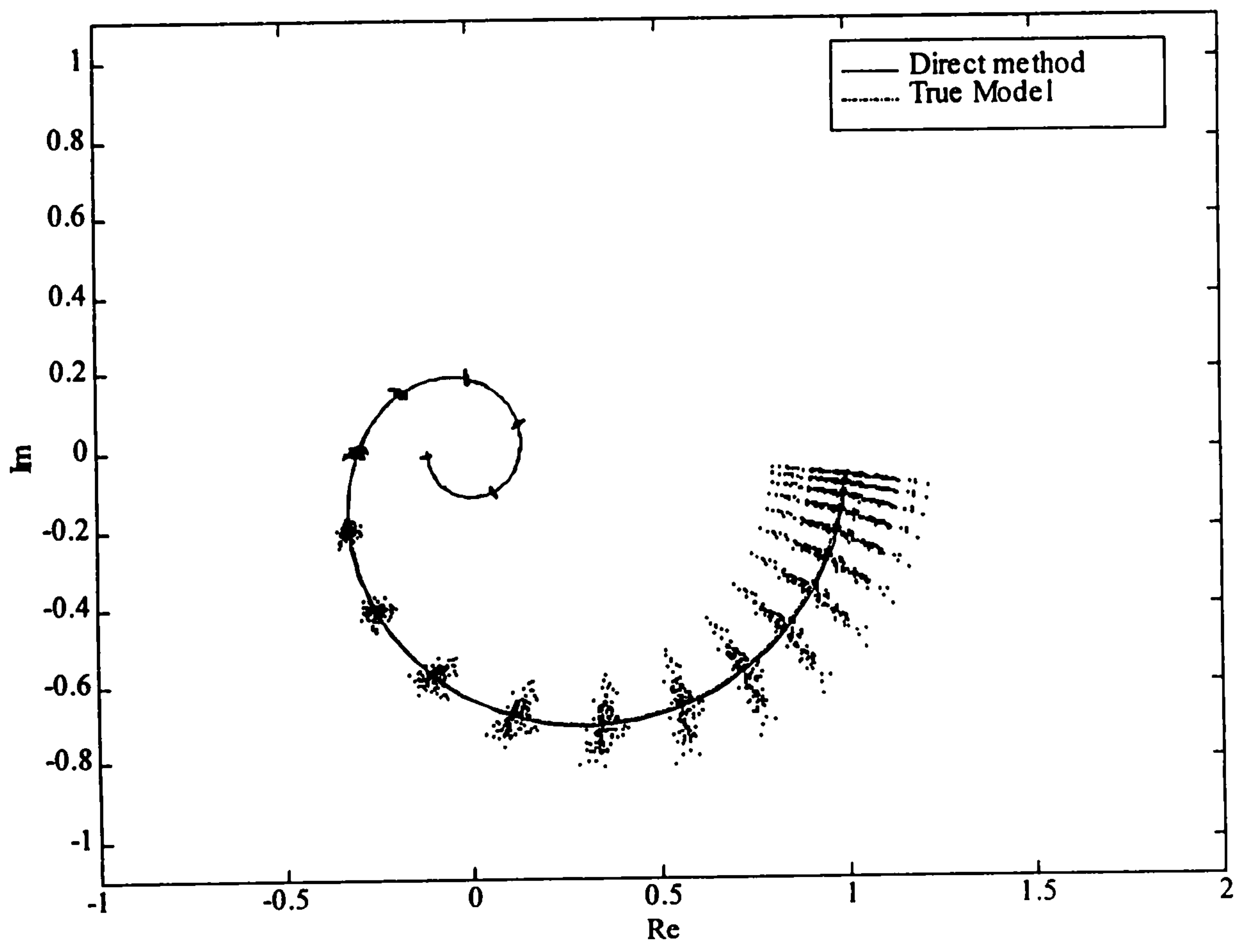


Figure 5.2: A Nyquist plot showing the variance of the estimated model using the direct method and parsimonious model at different frequencies (n=2, N=5000, ω is a discrete frequency log scaled between 0.01 to 3.1).

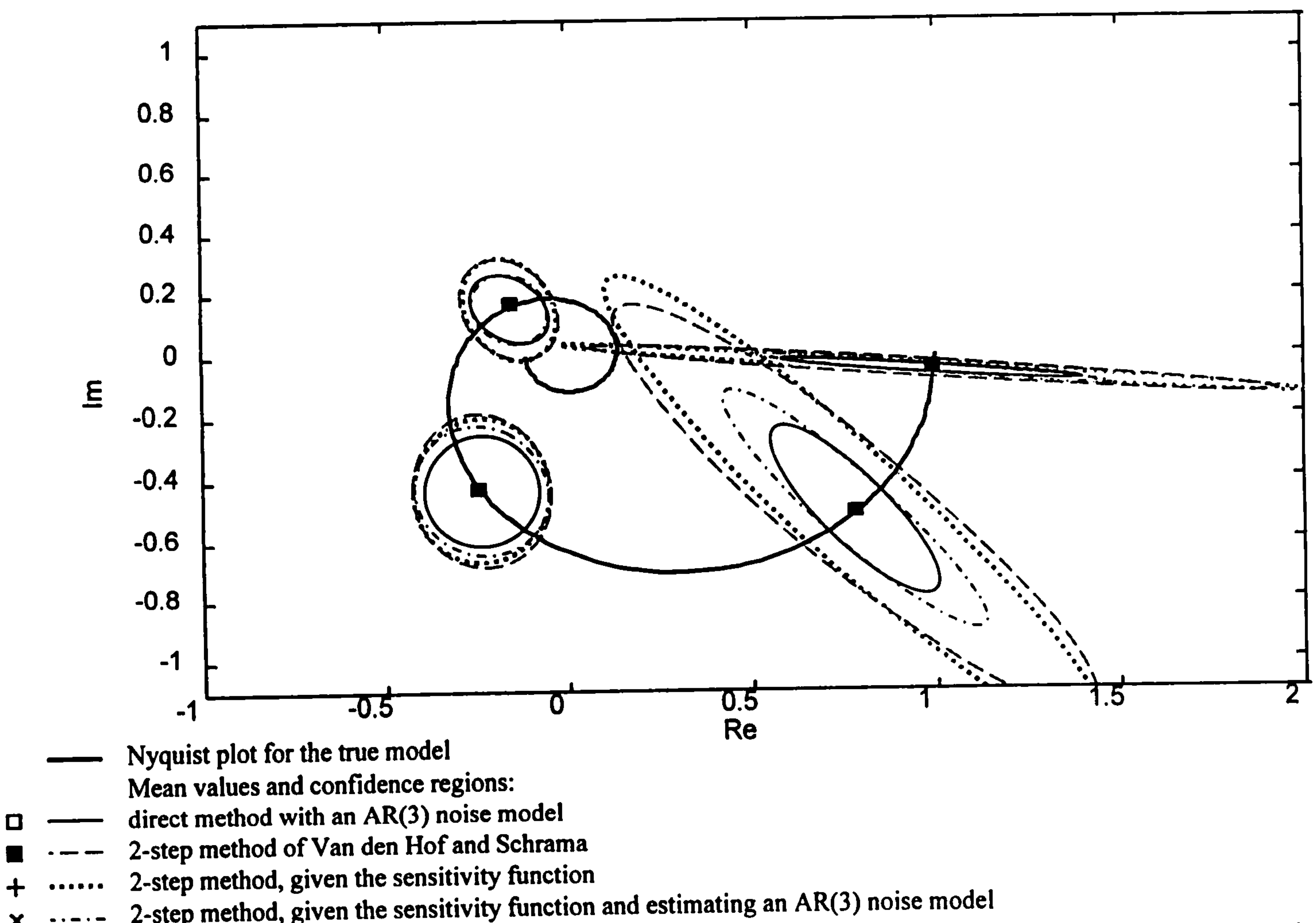


Figure 5.3: A Nyquist plot showing the variance of the estimated model using different methods (n=15, N=1000,ω=[0.01, 0.1, 0.4, 1]).

5.3.2. Evaluation of Asymptotic Results

The approximate closed-loop variance expression in (5.11) is only valid asymptotically as both the number of observations (*N*) and the model order (*n*) became large. Figures 5.4 and 5.5 show the effect of increasing the FIR model order and the number of observations respectively for the direct identification method. The asymptotic expression (5.11) suggests that the area of the confidence region should increase in a manner directly proportional to the model order (*n*) and decrease in a manner inversely proportional to number of observations (*N*).

In Figure 5.4, plots of the (*Area/n*) versus frequency shows that for $n \geq 60$ the plot appears to be independent of *n*, confirming the dependence suggested in (5.11). In Figure 5.5, plots of ($Area \times N$) verses frequency also appears to show convergence as *N* increases. Little effect is seen in increasing *N* from 500 to 5000 observations.

Comparison of the two-step and direct methods of closed-loop identification at different frequencies are shown in Figures 5.6 and 5.7. Two things are apparent regarding the asymptotic behavior of the two-step methods versus the direct method:

i) The direct method appears to approach the asymptotic behavior faster than the two-step methods.
ii) The different variations of the two-step methods never quite achieve the same lower variance asymptote achieved by the direct method as predicted by (5.12), particularly at low frequency. This is probably a result of assumptions made in deriving (5.12).

5.3.3. Assessment of Non-Asymptotic Results

In chemical processes, the data sets are generally of limited duration and the noise models often tend to be near non-stationary in nature. The effect of not using a noise model in identification for these situations could be large. The fact that the two-step methods have higher asymptotic variance at lower frequencies increases the chance that the gain of the process would be estimated imprecisely with a limited data set, which is a key issue in controller design.

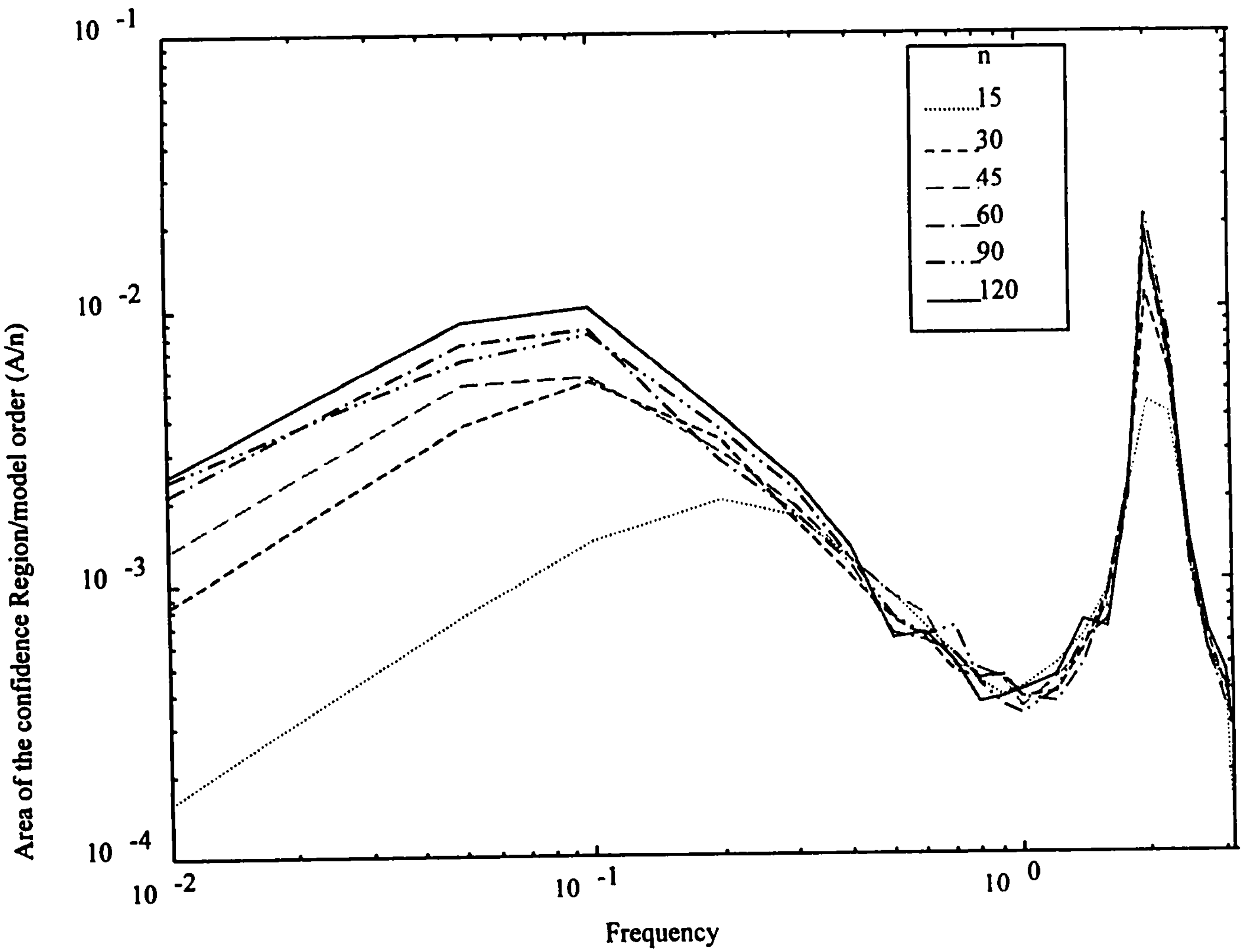


Figure 5.4: The trend of the confidence region as model order is increased, for the direct method (N=5000).

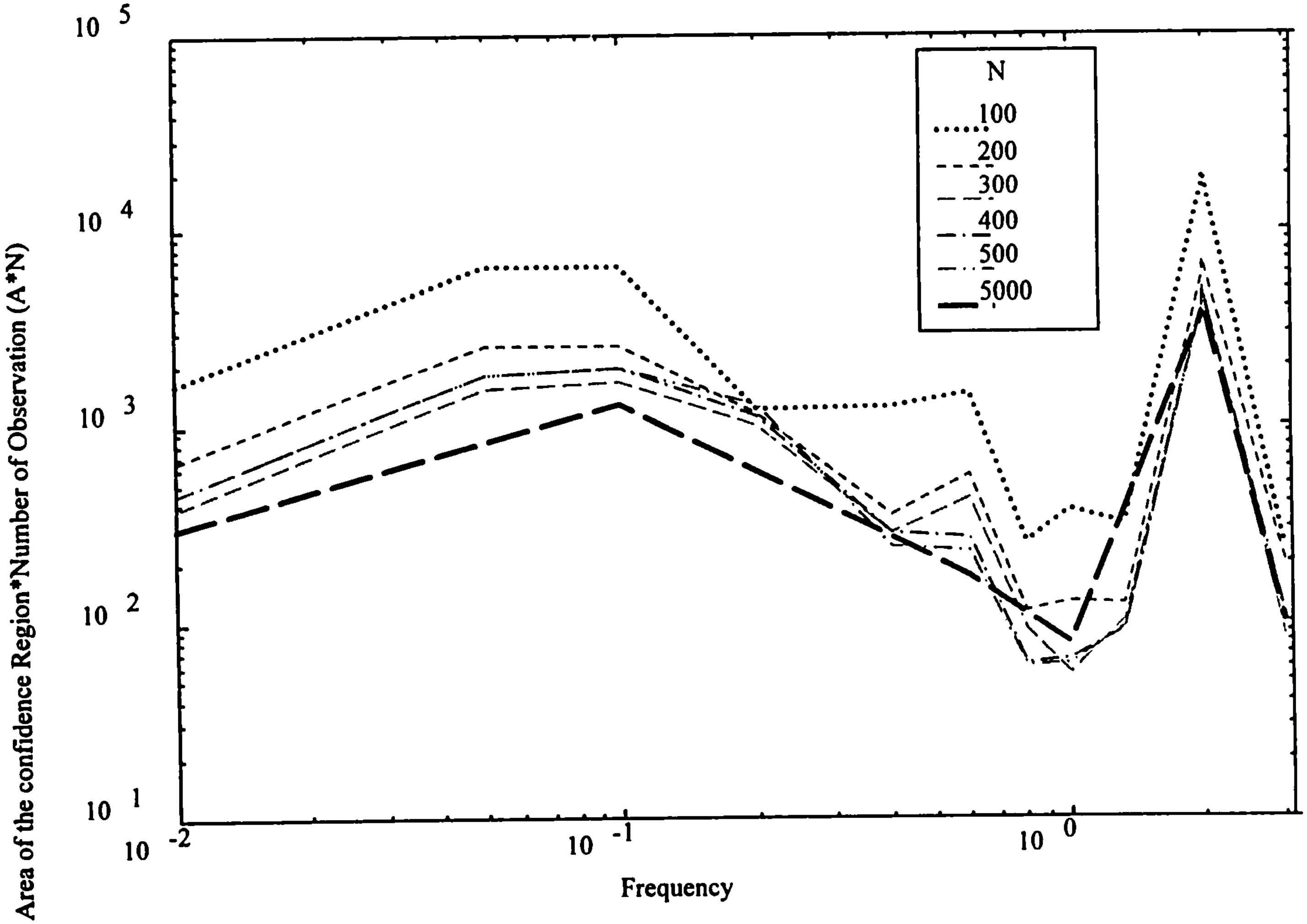


Figure 5.5: The trend of the confidence region as data point is increased, for the direct method(n=45).

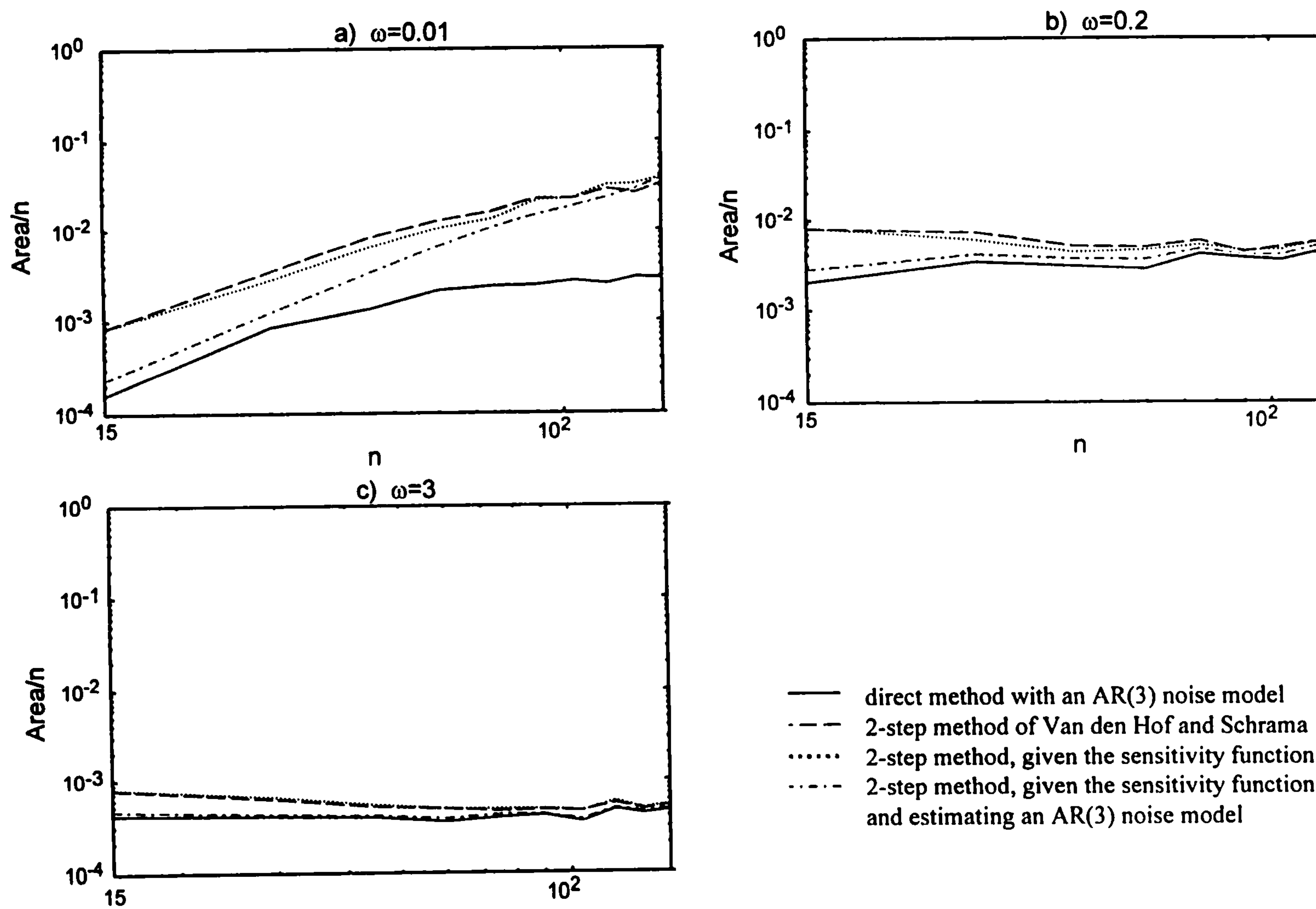


Figure 5.6: The effect of the model order on the different methods of closed-loop identification (N=5000).

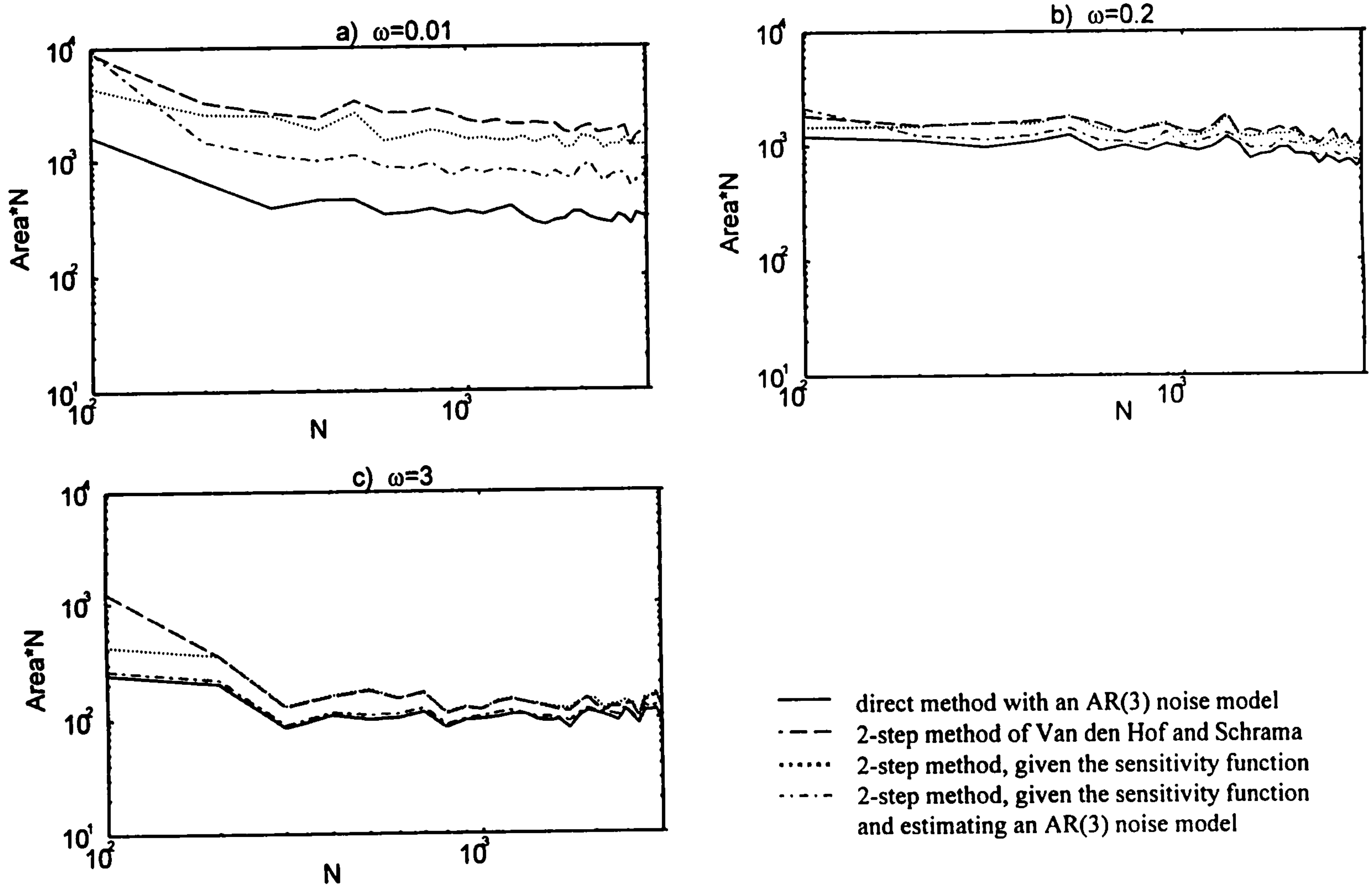


Figure 5.7: The effect of the number of observations on the different methods of closed-loop identification (n=45).

The following results can be deduced from Figures 5.6 and 5.7 under non-asymptotic conditions (low n and N), and from the confidence regions plotted in Figure 5.3:

i) The two-step method as suggested by Van den Hof and Schrama (1993) consistently performed worse than the direct method.

ii) As expected, the variance of the estimated model decreased slightly when the error introduced in the first step of the two-step method was eliminated by providing the true S. However, the improvement is seen to be small, except at very small N and low ω where the small sample size leads to larger estimation error for S.

iii) The most noticeable improvement in the two-step method came when one simultaneously identified a noise model along with the process model. In this case the results were very similar to the direct method, as one would expect.

Although the results shown in this chapter are for a specific SISO system, they should hold for MIMO systems as well. The MIMO identification using two-step method is discussed by Eek et al. (1994) and Barrs (1994).

5.3.4. Effect of Higher Order Transfer Function

Apart from a simple first order process model and first order autoregressive noise model, other linear transfer function models were also considered. For example, the following 3rd order transfer function model, with equal poles of 0.65 and a gain of 1:

$$G(q^{-1}) = \frac{0.0429q^{-3}}{1 - 1.95q^{-1} + 1.2675q^{-2} - 0.2746q^{-3}} \tag{5.22}$$

This transfer function was simulated with the settings mentioned in the section titled "Base Case Simulation", with the same controller and noise model as the base case simulation. The result of this simulation, shown in Figure 5.8, suggests that the observations made about the simple transfer function comparisons appear valid for more

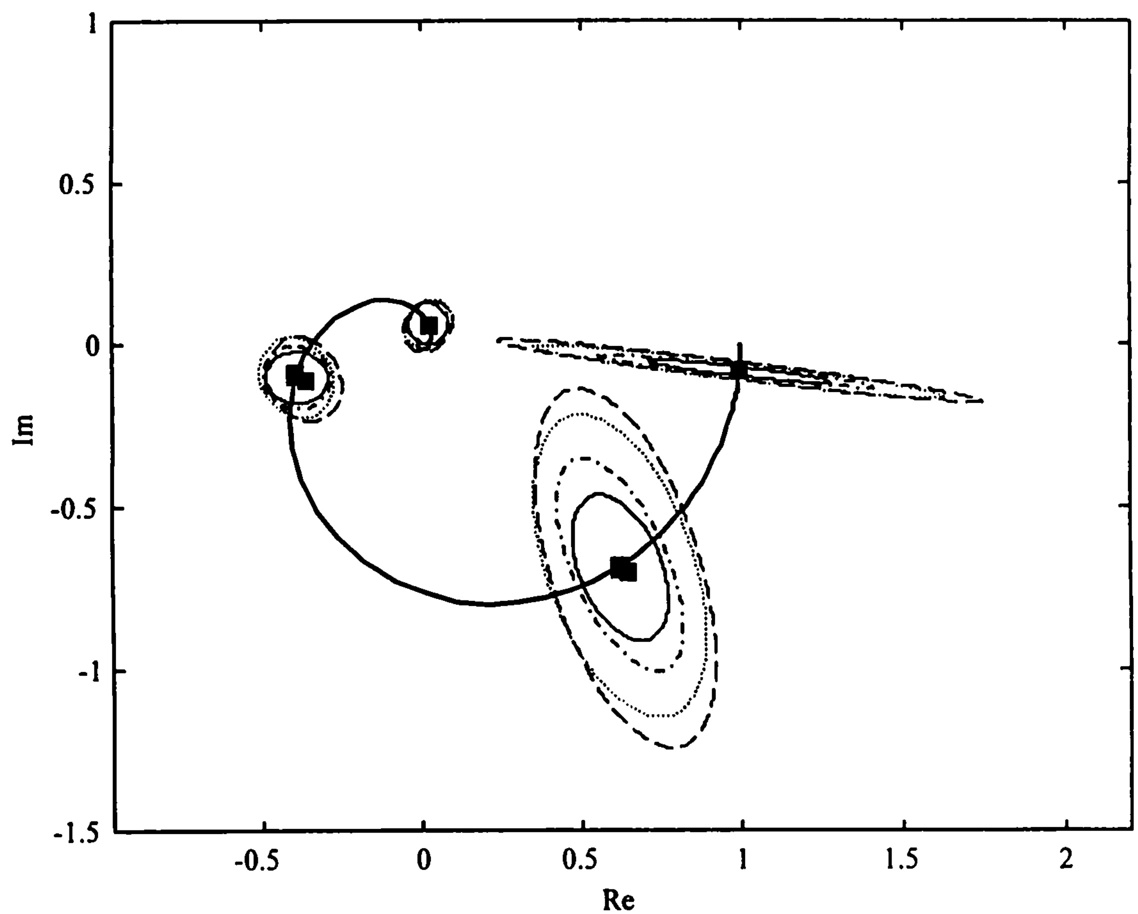


Figure 5.8: A Nyquist plot showing the variance and bias of the estimated dynamic model (n=25, N=5000,ω=[0.01, 0.1, 0.4, 1])(for notations see Figure 5.3).

complicated transfer functions. This is expected since the data is fitted to a non-parsimonious FIR model, which is capable of fitting models with a variety of responses.

5.3.5. Effect of Controller Tuning

All the previous simulations that were considered had a moderately tuned PI controller (5.19). A minimum variance controller (MVC), which is a highly tuned controller, was also considered with the dynamic model (5.17) and noise model (5.18). The controller transfer function is:

$$C(q^{-1}) = \frac{4.287 - 3.429q^{-1}}{1 - 0.857q^{-3}} \tag{5.23}$$

This MVC was simulated with the settings mentioned in the section titled "Base Case Simulation". The simulation result, illustrated in Figure 5.9(a), shows that the direct method again consistently gave the best results. Other simulations were also performed, which corroborated that the direct method produces a lower variance in the estimated transfer function compared to the 2-step method, independent of the controller performance and the order of the process.

5.3.6. Effect of Noise Model or Sensitivity Function

In the simulation study above (using controller (5.23)), Figure 5.9(a) also shows that the 2-step method appears to exhibit a bias in the transfer function estimate at low frequencies (ω = [0.01 0.1]). However, when the length of the FIR model used to estimate the sensitivity function in the first step of the 2-step approach was reduced from m = 15 to m = 5 (m = 3 is theoretically adequate in this case) the equivalent Monte Carlo simulation, presented in Figure 5.9(b), shows that the bias has disappeared. A possible explanation for this is as follows. Using an unnecessarily high order FIR model in the first step leads to an unbiased, but a high variance estimate of the sensitivity function.

a) Direct method with AR(3) noise model and 2-step method with 15 parameter FIR for sensitivity function

b) Direct method with AR(1) noise model and 2-step method with 5 parameter FIR for sensitivity function

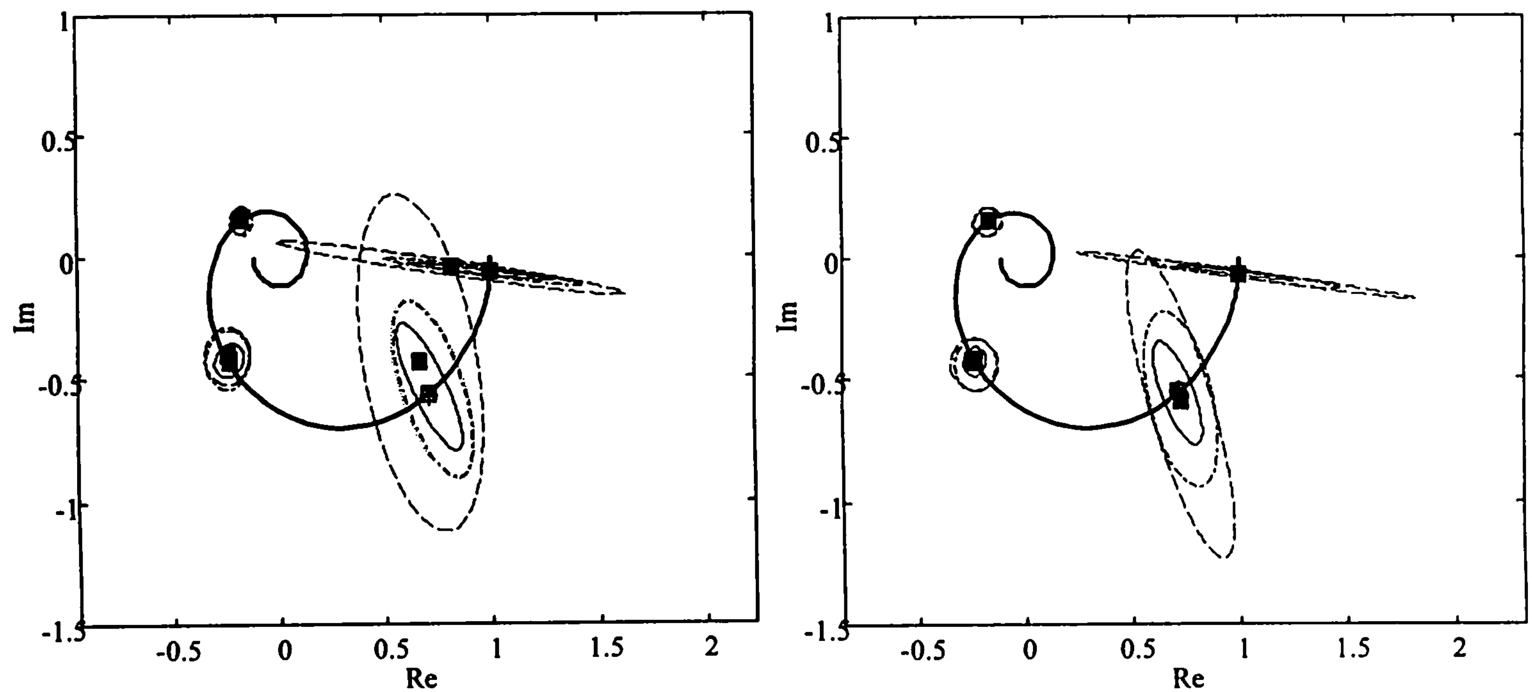


Figure 5.9: A Nyquist plot showing the variance and bias of the estimated model for the MVC (n=25, N=5000,ω=[0.01, 0.1, 0.4, 1]) (for notations see Figure 5.3).

This estimated sensitivity function is then fixed and used to filter the data for the second step and the process FIR model is estimated conditional on this sensitivity function. The observed bias in the process FIR model is apparently due to the nonlinear propagation (see equation (5.3)) of the larger errors in the sensitivity function estimates into the second step.

On the other hand, all the methods of closed-loop identification exhibit a bias error if the process model is inadequate. If the order of the FIR model (m) in (5.10) is not large enough, both the direct method and the 2-step method will display bias in the estimated transfer function (Ljung 1987). In the direct method of closed-loop identification, an appropriate noise model is also required to obtain an unbiased estimate of the transfer function (MacGregor and Fogal 1995). Similarly, the 2-step method requires an appropriate length of a FIR model for the sensitivity function to yield an unbiased estimate of the transfer function.

To see the effect of using an inadequate noise or sensitivity model, consider again the base case simulation with process model (5.17) and the more detuned controller (5.19). We now change the base case disturbance AR(1) model to the ARMA model:

$$H(q^{-1}) = \frac{0.7 + 0.3q^{-1}}{1 - 0.95q^{-1}} \tag{5.24}$$

However, the noise model used in the estimation in the direct method was still set to an AR(1), which is inadequate in this case. Furthermore, the sensitivity function required in the 2-step method was estimated by a 5 parameter FIR, once again an inappropriate model (m =15 is adequate). The resulting model estimates illustrate bias in both methods, as illustrated in Figure 5.10. However, if an appropriate noise model was used in the direct method and the sensitivity function was estimated with an adequate FIR model no bias is apparent (Figure 5.11).

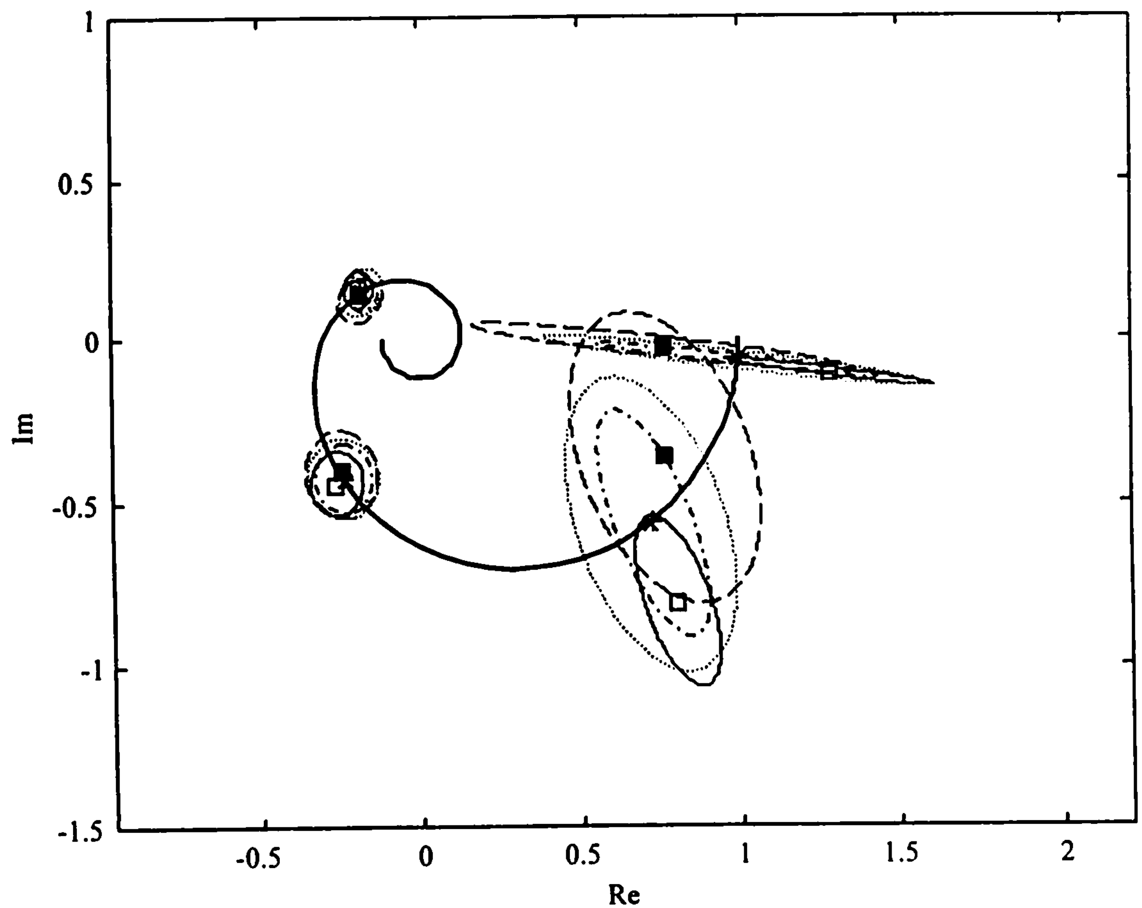


Figure 5.10: A Nyquist plot showing the variance and bias of the estimated dynamic model with an inappropriate model structure when the ARMA(1,1) noise model is used (n=25, N=5000,ω=[0.01, 0.1, 0.4, 1]) (for notations see Figure 5.3).

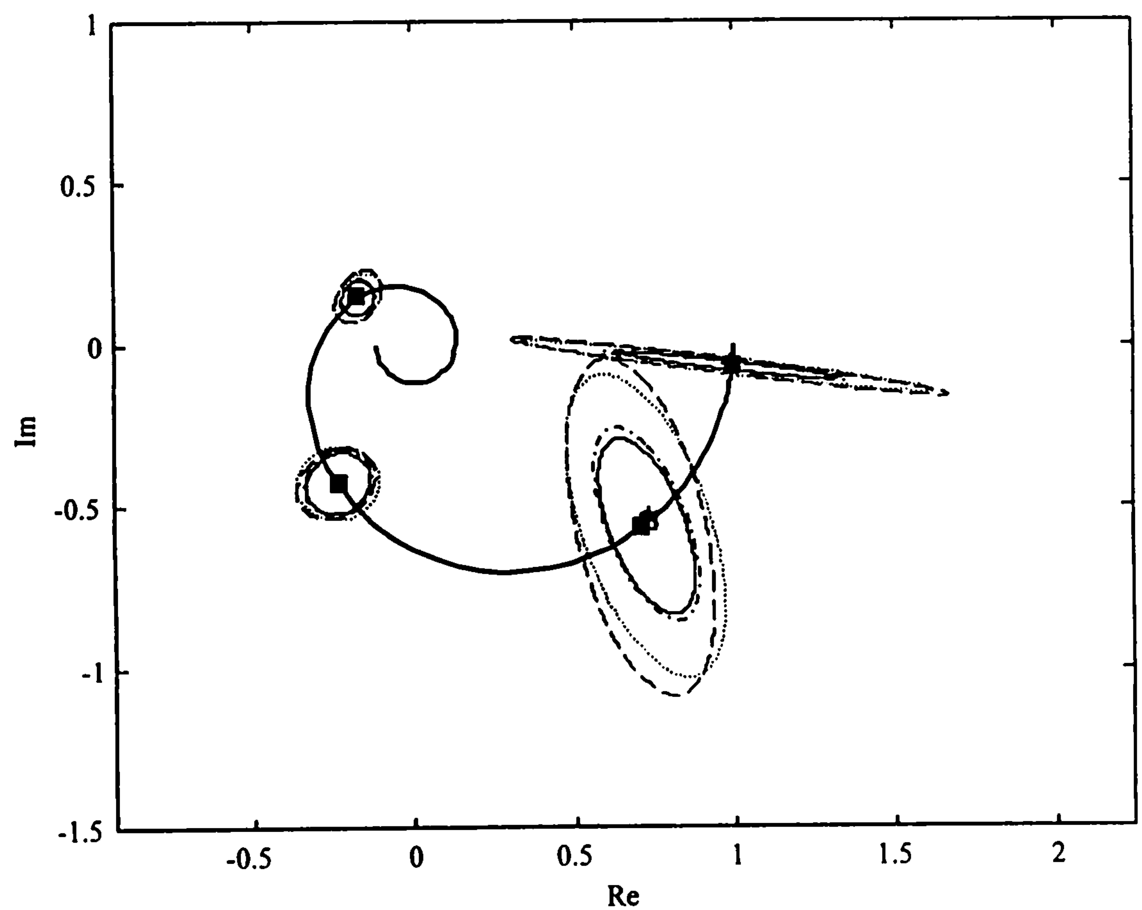


Figure 5.11: A Nyquist plot showing the variance and bias of the estimated dynamic model with an appropriate model structure when the ARMA(1,1) noise model is used (n=25, N=5000,ω=[0.01, 0.1, 0.4, 1]) (for notations see Figure 5.3).

These simulations illustrate that both the direct and the 2-step methods of closed-loop identification behave similarly with respect to bias error when the model orders for the disturbance or sensitivity function are inadequate (under-parameterized). However, the two-step methods can also exhibit bias when the sensitivity model is over-parameterized.

5.4. Conclusions

Some approaches used for closed-loop identification were reviewed and their similarities discussed. In particular, several variations of a common two-step method were investigated and compared with direct identification from an asymptotic and a finite data set viewpoint. The two-step method has been advocated because it leads to asymptotically unbiased estimates and avoids the necessity of simultaneously identifying a noise model. It does this by converting the closed-loop identification problem into two sequential, open-loop identification problems. The direct identification method also leads to asymptotically unbiased results, but at the expense of having to simultaneously identify a disturbance model. Both methods can be used with parsimonious or non-parsimonious model structures, and both can produce biased results if the model structures are inadequate. Approximate variance expressions of Gever et al. (1996) imply that both methods should achieve the same asymptotic variance, but simulations showed that the two-step approaches never quite achieved the same lower bound as the direct approach.

The asymptotic and finite data behavior of those approaches was investigated via Monte Carlo simulation studies, and the main observations were as follows. The direct identification method always gave better results (i.e., more precise estimates of $G(e^{i\omega})$ at all frequencies) for both the asymptotic and finite data situations. The greatest difference was observed with finite data sets. To investigate some of the sources of uncertainty in the two-step method, several variations of it were investigated. Errors introduced during the estimation of the sensitivity function S in the first step were seen to be generally small when the order of the model for S was well chosen. However, bias in the process model estimates was seen to arise when the FIR model for the sensitivity function was either too short (an under-parameterized model structure) or too long (an over-parameterized model structure). For finite data sets, a noticeable improvement in the two-step method was obtained by simultaneously identifying disturbances models in each step. However, this latter modification would appear to negate any advantage of the two-step approach.

In this study, we did not consider bias issues resulting from using inadequate low order parsimonious models for the process, nor did we consider the effects of the estimation errors on the robustness of any subsequent method, which uses these models (e.g., controller design). We used high order non-parsimonious (FIR) models for the process, where bias due to model structure inadequacy was not an issue, and our objective was to obtain the best estimate of the true process dynamics. Our studies focused on the variance of the process model for both asymptotic and non-asymptotic conditions, and on bias arising from poor choice of the disturbance and sensitivity function models. In the above context, our results would support the use of the direct method for closed-loop identification.

Chapter 6
Conclusions

In this thesis, issues on the use of prior and posterior knowledge in the identification of linear models for MIMO systems that lead to stable controller designs (i.e., satisfying stability condition) are investigated. The stability condition used is determined solely by the steady-state mismatch between the true process model and the estimated process model. The advantage of examining the estimated model in the above fashion is that the stability of the closed-loop system is completely determined by the estimated model used and is not affected by the controller designs or tuning. Therefore, the issues discussed are mainly concerned with model estimation and re-estimation rather than controller design or tuning. Chapters 2 and 3 are a study of the effect of the prior knowledge on controller stability, while Chapter 4 is a design of a stable controller based on posterior knowledge. Chapter 5 is a study of model estimation resulting from closed-loop identification.

The primary motivation behind this work has been utilization of prior (or posterior) knowledge in model identification for MIMO ill-conditioned systems. It is assumed that this model will be mainly used in a model-based controller (such as DMC), although it is not limited to this application. The systems considered are similar to the ones encountered in chemical industry, where prior knowledge about model parameters are common. This thesis provides an extensive analysis of using readily available chemical process prior (or posterior) knowledge in model identification.

The contribution is both theoretical and practical in nature. The issues explored in the area of the eigenvalue distribution and the propagation of model uncertainty to the determinant of the gain matrix throughout this thesis are perhaps the most significant theoretical contributions. The methodologies derived in Chapter 4 regarding model maintenance appear to have the greatest practical contribution. Other practical contributions include (but are not limited to) methods of utilizing prior knowledge in model identification, evaluating the effect of different prior knowledge on controller stability, presenting methods for evaluating the sensitivity of the model based controllers to prior knowledge, and provide a better understanding of different methods of closed-loop identification.

The effect of correct prior knowledge on the stability of the controller system is studied in Chapter 2. Some prior knowledge usually exists in chemical processes. The effect of this prior knowledge on the controller stability for ill-conditioned systems was the motivation of this chapter. The results suggest that not all types of prior knowledge should be used in the model estimation. In addition, it was determined that prior knowledge that provides information about the low-gain direction of the system is the most valuable form of prior knowledge. In contrast, linear inequality constraints (when they are true) may improve or degrade model quality (in terms of stability), depending on whether they satisfy or do not satisfy an uncheckable condition.

In real situations there may be error in the prior knowledge; this leads to the ideas discussed in Chapter 3, where the sensitivity of the controller stability to error in prior knowledge is studied. This results in useful checkable metrics that can be used by the practitioner in evaluating the effect of error in prior knowledge on controller stability before the control system is implemented.

Chapter 4 introduces new ideas on model maintenance for MIMO model-based control. This methodology is discussed in the context of posterior knowledge (runtime knowledge or gained knowledge) in model identification. The controller stability is used as a form of posterior knowledge in model re-estimation to produce a stable model based controller. This approach is especially useful in control of ill-conditioned systems or large systems where many input and output bound constraints are active at any given time. The same methodology of using posterior knowledge in model re-estimation was shown to be applicable in two other areas: incorporation of new experimental data and controller resilience. The contribution of this chapter is in the realization that the plant's closed-loop performance is a form of gained knowledge (or posterior knowledge) and that this knowledge can be used as posterior knowledge in model re-estimation to improve controller performance.

Finally, different methods of closed-loop identification and the issues associated with them are discussed in Chapter 5. This leads to a better understanding of the different methods of closed-loop identification under asymptotic and non-asymptotic conditions. It illustrates that while different methods may behave similarly under asymptotic conditions (i.e., infinite data and infinite model order), under non-asymptotic conditions the direct method of closed-loop identification performs better (in terms of variance and bias in the parameters) than the 2-step method of closed-loop identification.

References

Akaike, H., "On the use of a Linear Model for the Identification of Feedback Systems", Ann. Inst. Stat. Math, 20, 425-439, 1968.

Andersen, H.W., and M. Kummel, "Evaluating Estimation of Gain Directionality Part1: Methodology", Journal of Process Control, 2(2), 59-66, 1992.

Anderson, H.W., Kummel, M., and S.B. Jørgensen, "Dynamics and Identification of a Binary Distillation Column", Chemical Engineering Science, 44(11), 2571-2581, 1989.

Aoyama, A., Bell, G., and S.P. Walsh, "Implementation Issues in Quadratic Model Predictive Control", Proceedings of the American Control Conference, New Mexico, 3375-3379, 1997.

Baars, G.E.V., "Close loop system identification of an industrial wind turbine system and a preliminary validation result", Selected Topics in Identification, 7, 63-70, 1994.

Belsley, D.A., *Conditioning Diagnostics: Collinearity and Weak Data in Regression*, John Wiley & Sons, New York, 1991.

Boot, J.C.G., "On Sensitivity Analysis in Convex Quadratic Programming Problems", Operations Research, 11(5), 771-786, 1963.

Box, G., and G.M. Jenkins, *Time Series Analysis, forecasting and control*, Revised Edition, Prentice-Hall, Englewood Cliffs, NJ, 1976.

Box, G., and J.F. MacGregor, "Parameter Estimation with Closed-Loop Operating Data", Technometrics, 18(4), 371-380, 1976.

Box, G., and J.F. MacGregor, "The analysis of closed-loop dynamic stochastic systems", Technometrics, 15(4), 391-398, 1974.

Brambilla, A., and L. D'Elia, "Multivariable Control for Distillation Columns in the Presence of Strong Directionality and Model Errors", Industrial Engineering Chemistry Research, 31, 536-543, 1992.

Bristol, E.H., "On a New Measure of Interaction for Multivariable Process Control", IEEE Transaction of Automatic Control, 11(133), 1966.

Bro, R., and N.D., Sidiropoulos, "Least Squares Algorithms Under Unimodality and Non-negativity Constraints", Journal of Chemometrics, vol. 12, 1998.

Caines, P., and C. Chan, "Feedback between Stationary Stochastic Processes", IEEE transactions on automatic control, 20, 498-508, 1975.

Chang, J.W., and C.C. Yu, "Integrity of Inverse-Based Multivariable Controllers: Conventional Feedback Structure", Industrial Engineering Chemistry Research, 30, 1516-1524, 1991.

Chia, T., "Parameter Identification and Sate Estimation of Constrained Systems", Ph.D. Thesis, Case Western Reserve University, 1985.

Cooley, B.L., "Identification and control: Estimating, shaping, and accounting for uncertainty", Auburn University, Ph.D. Thesis, 1997.

Cooley, B.L., and J.H. Lee, "Control-relevant experiment design for multivariable systems described by expansions in orthonormal bases", 37(2), 273-281, 2001.

Cutler, C.R., and B.L. Ramaker "Dynamic Matrix Control - A Computer Control Algorithm", The Joint Automatic Control Conference Preprints, IEEE, New York, Wp5-B, 1980.

Dasgupta, S., Anderson, B., and R.J., Kaye, "Identification of Physical Parameters in Structured Systems", Automatica, 24(2), 1988.

Dayal, B.S. "Multivariate Statistical Regression Methods for Process Modelling and Experimental Design", Ph.D. Thesis, McMaster University, 1996.

Dayal, B.S. and J.F., MacGregor "Identification of Finite Impulse Response Models: Methods and Robustness Issues", Industrial Engineering and Chemistry Research, 35, 4078-4090, 1996.

Dimitriadis, V.D., and E.N. Pistikopoulos, "Flexibility Analysis of Dynamic Systems", Industrial Engineering Chemistry Research, 34, 4451-4462, 1995.

Doma, M.J., "The Identification of Multivariable Step Response Models from Closed-Loop Data", Master Thesis, University of McMaster, 1995.

Edgar, T.F. and D.M. Himmelblau, *Optimization of Chemical Processes*, McGraw-Hill, 1988.

Eek, R.A., Both, J.A., and P.M.J. Van den Hof, "Closed-loop identification of a continuous crystallization process", Selected Topics in Identification, 7, 71-82, 1994.

Efron, B., and R.J. Tibshirani, *An Introduction to the Bootstrap*, Chapman & Hall, New York, 1993.

Esmaili, A., MacGregor, J.F., and P.A. Taylor, "Direct and two-step methods for closed-loop identification: a comparison of asymptotic and finite data set performance", Journal of Process Control, 10, 525-537, 2000.

Featherstone, A.P. "Control Relevant Identification of Sheet and Film Processes", Ph.D. Dissertation, University of Illinois at Urbana-Champaign, 1997.

Featherstone, A.P., and R.D. Braatz, "Integrated Robust Identification and Control of Large-Scale Processes", Industrial Engineering and Chemistry Research, 37, 1998.

Fiacco, A.Y., "Sensitivity Analysis for Nonlinear Programming Using Penalty Methods", Mathematical Programming, 10, 287-311,1976.

Fiacco, A.Y., *Introduction to Sensitivity and Stability Analysis in Nonlinear Programming*, Academic Press, New York, 1983.

Fletcher, R., *Practical Methods of Optimization*, 2nd Edition, Wiley, Toronto, 1987.

Fogel, E., and Y.F. Huang, "On the Value of Information in System Identification-Bounded Noise Case", Automatica, Vol. 18, No. 2, 1982.

Gal, T. "A 'Histogramme" of Parametric Programming" Journal of Operational Research Society 31, 449-451 (1980).

Garcia, C.E., and M. Morari, "Internal Model Control. 2. Design Procedure for Multivariable Systems", Industrial Engineering Chemistry Fundamentals, Vol. 24, 1985.

Gevers, M., Ljung, L., and P.M.J. Van den Hof, "Asymptotic variance expressions for closed-loop identification and their relevance in identification for control", Selected Topics in Identification, Modelling and Control, 9, 9-15, 1996.

Grosdidier, P., Morari, M., and B.R. Holt, "Closed-Loop Properties from Steady-Sate Gain Information", Industrial Engineering Chemistry Fundamental, 24, 221-235, 1985.

Haggblom, K.E., and J.M. Boling, "Multimodel Identification for Control of an Ill-Conditioned Distillation Column", Journal of Process Control, 8(3), 1998

Hjalmarsson, H., Gevers, M., and F. De Bruyne, "For Model-Based Control Design, Closed-Loop Identification Gives Better Performance", Automatica, 32(12), 1659-1673, 1996.

Hoffman, D.W. "Process Control: Achieving the Benefits" ADCHEM 2000, Pisa Italy June 14-16.

Hovd, M., and S. Skogestad, "Procedure for Regulatory Control Structure Selection with Application to the FCC Process", AIChE Journal, 39(12), 1938-1953, 1993.

Hovd, M., Braatz, R.D., and S. Skogestad, "SVD Controllers for H_2-, H_∞- and μ-optimal Control", Automatica, 33(3), 433-439, 1997.

Huang, B., and S. Shah, "Closed-loop Identification: A Two-step Approach", Journal of Process Control, 7(6), 425-438, 1997.

Judge, G.C., and T.A. Yancey, *Improved Methods of Inference in Econometrics*, North-Holland, New York, 1986

Judge, G.G., Griffiths, W.E., Hill, R.C., and T.C., Lee, "The Theory and Practice of Econometrics", John Wiley and Sons, New York, 1980.

Koung, C.W., "Analysis and identification of Robust Linear Models for Multivariable Control", Ph.D. Thesis, McMaster University, 1991.

Koung, C.W., and J.F. MacGregor, "Identification for Robust Multivariable Control: The Design of Experiments", Automatica, 30(10), 1541-1554, 1994.

Lau, H., Alvarez, J., and K.F. Jensen, "Synthesis of Control Structures by Singular Value Analysis: Dynamic Measures of Sensitivity and Interaction", AIChE Journal, 31(3), 427-439, 1985.

Lawson, C. L., and R. J. Hanson, *Solving Least Squares Problems*, SIAM, Philadelphia, 1995.

Li, W., and J.H. Lee, "Control Relevant Identification of Ill-Conditioned Systems: Estimation of Gain Directionality", Computers and Chemical Engineering, 20(8), 1996.

Ljung, L., "Asymptotic Variance Expressions for Identified Black-Box Transfer Function Models", IEEE transactions on automatic control, 30(9), 834-843, 1985.

Ljung, L., *System Identification Theory for the User*, 2nd edition, Prentice-Hall, Englewood Cliffs, NJ, 1999.

Ljung, L., *System Identification Theory for the User*, Prentice-Hall, Englewood Cliffs, NJ, 1987.

MacArthur, J.W., "RMPCT: A New Robust Approach to Multivariable Predictive Control For the Process Industries", Control Systems, 53-60, 1996.

MacGregor, J.F., and D. Fogal, "Closed Loop Identification: The Role of the Noise Model and Pre-filters", Journal of Process Control, 5, 163-171, 1995.

MacGregor, J.F., Kourti, T., and J.V. Kresta, "Multivariate Identification: A Study of Several Methods", IFAC International Symposium, ADCHEM '91 Proceedings, Pergamon Press, Tolouse, France, 369-375, 1991.

Manousiouthakis, V., Savage, R., and Y. Arkun, "Synthesis of Decentrilized Process Control Structures Using the Concepts of Block Relative Gain", AIChE Journal, 32, 991-1003, 1986.

Marselle, D., Morari, M., and D.F. Rudd, "Design of Resilient Processing Plants II: Design and Control of Energy Management Systems", Chemical Engineering Science, 37(2), 259-270, 1982.

Maurath, P.R., Laub, A.J., Seborg, D.E. and D.A. Mellichamp "Predictive Controller Design by Principal Components Analysis", Industrial Engineering and Chemistry Research, 27,1988.

McKay, B., Willis, M., and G. Barton, "Steady-State Modelling of Chemical Process Systems Using Genetic Programming", Computers and Chemical Engineering, 21(9), 1997.

Mehta, M.L., *Random Matrices,* Revised and Enlarged Second Edition, Academic Press Inc., 1991.

Mijares, G., and C.D. Holland, "Comments on "Internal Model Control. 2. Desigen Procedure for Multivariable Systems"", Industrial Engineering Chemistry Research, 26, 632-633, 1987.

Mijares, G., Cole, J.D., Naugle, N.W., and C.D. Holland, "A New Criterion for the Pairing of Control and Manipulated Variables", AIChE Journal, 32(9), 1439-1449, 1986.

Moore, C. "Application of Singular Value Decomposition to the Design, Analysis, and Control of Industrial Processes", Proceedings of the American Control Conference, IEEE Press, Piscataway, NJ, 643-652, 1986.

Morari, M., "Design of Resilient Processing Plants III: A General Framework for the Assessment of Dynamic Resilience", Chemical Engineering Science, 38(11), 1881-1891, 1983.

Morari, M., "Response to Comments on "Internal Model Control. 2. Desigen Procedure for Multivariable Systems"", Industrial Engineering Chemistry Research, 26, 633, 1987.

Morari, M., and Zafiriou E., *Robust Process Control,* Prentice Hall, Englewood Cliffs, New Jersey, 1989.

Nguyen, T.C., Barton, G.W., Perkins, J.D., and R.D. Johnston, "A condition Number Scaling Policy for Stability Robustness Analysis", AIChE Journal, 34(7), 1200-1206, 1988.

Ninness, B., and G.C. Goodwin, "Estimation of Model Quality", Automatica, 31(12), 1771-1797, 1995.

Ninness, B., and G.C. Goodwin, "Estimation of Model Quality", IFAC System Identification, Copenhagen, Denmark, 1994.

Ogata, K., *Discrete-Time Control Systems*, Prentice Hall International Editions, 1995.

Papastathopoulou, H.S., and W.L. Luyben, "A New Method for the Derivation of Steady-State Gains for Multivariable Processes", Industrial Engineering Chemistry Research, 29(3), 1990.

Pensar, J.A., and K.V. Waller, "Steady-State-Gain Identification for Multivariable Process Control", Industrial Engineering Chemistry Research, 32, 2012-2016, 1993.

Qin, S.J., and T.A. Badgwell, "An Overview of Industrial Model Predictive Control Technology", 5th International Conference of Chemical Process Control, 93(316), 1997.

Richalet, J., Rault, A., Testud, J.L., and J. Papon, "Model Predictive Heuristic Control: Application to Industrial Processes", Automatica, 14, 413-428, 1978.

Rubinstein R.Y. *Simulation and the Monte Carlo Method*, John Wiley & Sons, Toronto, 1981.

Sagfors, M.F. and K.V., Waller, "Dynamic Low-Order Models for Capturing Directionality in Nonideal Distillation", Industrial & Engineering Chemistry Research, 34(6), 2038-2050, 1995.

Schrama, R., and P. Van den Hof, "Iterative Identification and Control Design: A Worked out Example", Selected Topics In Identification, Modeling and Control, 5, 93-100, 1992.

Seber, G.A.F., *Linear Regression Analysis*, John Wiley & Sons, New York, 1977.

Shao, J., and C.F.J., Wu, "A General Theory for Jackknife Variance Estimation", The Annals, of Statistics, 17(3), 1176-1197, 1989.

Shridhar, R., and D.J. Cooper, "A Tuning Strategy for Unconstrained Multivariable Model Predictive Control", Industrial & Engineering and Chemistry Research, 37, 4003-4016, 1998.

Skogestad, S., and I., Postlethwaite, *Multivariable Feedback Control: Analysis and Design*, John Wiley & Sons Ltd., New York, 1996.

Skogestad, S., and M. Morari, "LV-Control of a High Purity Distillation Column", Chemical Engineering Science, 43, 33-48, 1988.

Skogestad, S., and M. Morari, "Understanding the Dynamic Behavior of Distillation Columns", Industrial & Engineering Chemistry Research, 27, 1848-1862, 1988.

Skogestad, S., Morari, M. and J.C. Doyle "Robust control of Ill-conditioned Plants: High Purity Distillation", IEEE Transaction of Automatic Control, 33, 1092-1105, 1988.

Söderström, T., and P. Stoica, *System Identification*, Prentice Hall, New York, 1989.

Söderström, T., Gustavsson, I., and L. Ljung, "Identifiability conditions for linear systems operating under closed-loop", International Journal of Control, 21, 243-255, 1975.

Stewart, G.W., and J. Sun, *Matrix Perturbation Theory*, Academic Press Inc., New York,1990.

Theil, H., "On the Use of Incomplete Prior Information in Regression Analysis", American Statistical Association Journal, 399-414, June 1963.

Theil, H., *Economic Forecasts and Policy*, North-Holland Publishing Company (2nd edition), Amsterdam, 1961.

Timmons, W.D., "Constrained Identification for Adaptive Control: Application to Biomedical Systems", Ph.D. Thesis, Case Western Reserve University, 1992.

Tulleken, H.J., "Grey-box Modelling and Identification Topics", Doctoral Thesis, Delft University of Technology, 1992.

Tulleken, H.J., "Grey-box Modelling and Identification using Physical Knowledge and Bayesian Techniques", Automatica, 29(2), 285-308, 1993.

Van den Hof, P., and R. Schrama, "An Indirect Method for Transfer Function Estimation from Closed Loop Data", Automatica, 29(6), 1523-1527, 1993.

Watts, D.G., "Estimating Parameters in Nonlinear Rate Equations", Canadian Journal of Chemical Engineering, 72, 701-710, 1994.

Wehrens, R., Putter, H., and L.M.C. Buydens, "The Bootstrap: a Tutorial", Chemometrics and Intelligent Laboratory Systems, 54, 35-52, 2000.

Wishart, J., "The Generalized Product Moment Distribution in Samples from a Normal Multivariate Population", Biometrika, 20A, 32-52, 1928.

Wolbert, D., Joulia, X., Koehret, B., and L.T., Biegler, "Flowsheet optimization and optimal sensitivity using analytical derivatives", Computers Chemical Engineering, 18(11/12), 1083-1095, 1994.

Wolbert, D., Joulia, X., Koehret, B., and L.T., Biegler, "Optimal Flowsheet sensitivity in sensitivity oriented environment", European symposium on computer aided process engineering-2 ESCAPE-2, Toulouse, France, s117-s122, October 1992.

Yousfi, C., and R., Tournier, "Steady State Optimization Inside Model Predictive Control", American Control Conference, TP8 18:00, 1866-1870, 1991.

Zafiriou, E., "Robust Model Predictive Control of Process with Hard Constraints", Computers and Chemical Engineering, 14(4/5), 359-371, 1990.

Zafiriou, E., and H.-W. Chiou, "On the Dynamic Resiliency of Constrained Processes", Computers and Chemical Engineering, 20(4), 347-355, 1996.

Zheng, A., and M. Morari, "Stability of Model Predictive Control with Mixed Constraints", IEEE Transaction of Automatic Control, 40(10), 1818-1823, 1995.

Zhu, Z.X., and A. Jutan, "Robust Multivariable Control Using an SVD-based Controller", Chemical Engineering Science, 53(6), 1145-1151, 1998.

Nomenclature

The following is a list of symbols and the acronyms that have been used in this thesis. This is not an exhaustive list, all symbols and acronyms have been explained in the main body of the thesis.

Symbols:

Roman Letters:

$\mathcal{A}$	= occurrence of event $\mathcal{A}$ (chapter 1)
A	= area of the 95% confidence area (chapter 5)
A	= is a matrix of coefficients involved in the linear inequality constraints (chapter 3)
Aeq	= is a matrix of coefficients involved in the linear equality constraints (chapter 3)
$a_{i,t}$	= white noise added to the process output at time t (chapter 2)
b	= a vector of constants involved with inequality constraints in QP (chapter 3)
beq	= a vector of constants involved with equality constraints in QP (chapter 3)
$\mathcal{B}$	= prior knowledge (chapter 1)
c	= a constant resulting from the prior knowledge (chapter 3)
C_s	= the coefficient of the active constraint (chapter 3)
$\mathcal{C}$	= the posterior knowledge (chapter 1)
d_k	= dither (or the test signal) (chapter 5)
D_X	= scaling matrix of the input (chapter 4)
D_Y	= scaling matrix of the output (chapter 4)

E	= a matrix of the white noise
f	= a vector of constants (chapter 3)
$f(\cdot)$	= any linear or non-linear constraint (chapter 3)
e_k	= error between the predicted output and the true output (chapter 5)
$F(q^{-1})$	= diagonal matrix, where the diagonal elements are a 1st order fileter
$F_{\alpha;p,N-p}$	= denotes the F-distribution with the critical value exceeding probability α (chapter 5)
G	= true gain matrix ($G = G(0)$)
$G(s)$	= true process transfer function matrix in s-domain
$G(q^{-1})$	= true process transfer function matrix in terms of the backward shift operator
$g_{i,j}$	= gain matrix element corresponding to row i and column j
$g_{i,j}(q^{-1})$	= the transfer function relationship between the i^{th} output and the j^{th} input (chapter 2)
$G_{r,i}$	= is the reduced system i that corresponds to a particular set of constraint being active (J^i, K^i)
G	= transfer function for a SISO system (chapter 5)
$\hat{G}_r$	= first estimated reduce system's gain matrix (chapter 4)
$\hat{G}$	= the estimated gain matrix ($\hat{G} = \hat{G}(0)$)
$\hat{\hat{G}}$	= the re-estimated gain matrix
$\breve{G}$	= is made of adding either extra rows or columns of zeros to matrix G to produce a square matrix
H	= Hessian of a QP problem (chapter 3)
H	= noise model (chapter 5)
I	= identity matrix
J^i	= ith set of MVs whose constraints are active (chapter 4)
J	= is the set of MVs whose constraints are active (chapter 4)
K	= is the set of CVs whose constraints are active (chapter 4)

$K_C(s)$	= controller in the s-domain (chapter 4)
$K_{CS}(s)$	= is a diagonal control matrix (chapter 4)
k''_{max}	= the maximum number of sub-sytems.(chapter 4)
K^i	= ith set of CVs whose constraints are active (chapter 4)
k'	= the number of the sub-systems that have resulted in UCS (chapter 4)
k''	= the total number of the systems (sub-system or full system) that the controller has operated in (chapter 4)
l	= prediction horizon (chapter 5)
lb	= a lower bound on the optimization variables (chapter 3)
$l_{i,j}$	= total number of impulse coefficient used to for the relationship between the i^{th} output and the j^{th} input
M	= input horizon in the QDMC or DMC controller
N	= the number of observations (chapter 5)
$N(q^{-1})$	= noise model (chapter 2)
n	= the model order (chapter 5)
n_x	= number of process inputs
n_y	= number of process outputs
p	= the degree of freedom (chapter 5)
P	= output (or prediction) horizon in the QDMC or DMC controller
p_r	= probability of a the reduce system being unstable (chapter 4)
q^{-1}	= backward shift operator
R	= a matrix of constants defining the linear constraints (chapter 3)
S	= a subset of active constraints (chapter 3)
S	= sensitivity function (chapter 5)
SSE	= Sum of Square Error for all the outputs (i.e. $SSE = trace\left(\left(Y-\hat{G}X\right)^T\left(Y-\hat{G}X\right)\right)$)
ub	= upper bound on the optimization variables (chapter 3)
$u_{j,t}$	= j^{th} input at time t (chapter 2)

$v_{k,i,j}$ = k^{th} impulse response coefficient for the $g_{i,j}(q^{-1})$ transfer function (chapter 2)

u_i = process input at time i (chapter 5)

W = compensator matrix for SVD style controllers

x = the right eigenvector of A (chapter 3)

X = a matrix of the inputs, where each row representing an input

y = the left eigenvector of A (chapter 3)

y_i = process output at time i (chapter 5)

Y = a matrix of the outputs, where each row represents an output

$y_{i,t}$ = i^{th} output at time t (chapter 2)

Z = is a standard normal random variable

Greek Letters

α = the level of significance used in the test (0.05 in this study) (chapter 5)

α_i = the parameter in a first order filter associated with the i^{th} output (chapter 3 and 4)

β = solution to OLS

$\hat{\beta}_H$ = solution to the CLS

$\hat{\beta}_R$ = the solution to the least square problem with the inexact linear equality constraint

γ^* = minimum condition number

δ = magnitude of error in the prior knowledge (chapter 3)

δ = distance between lower and upper bound in a constraint (chapter 2)

Δ = the perturbation matrix (chapter 3)

Δ = "switching" matrix associated with sensor or actuator failure (chapter 4)

ε = a small change (chapter 3)

ε_k = prediction error in joint input/output identification (chapter 5)

κ	= condition number
$\tilde{\lambda}$	= eigenvalue of A+E
λ^s	= Lagrangian multiplier
λ_q	= a set of eigenvalues whose confidence interval includes zero
$\hat{\hat{\lambda}}$	= are the eigenvalues of the second estimated gain matrix ($\hat{\hat{G}}$)
$\hat{\lambda}_i$	= i^{th} eigenvalue of the first estimated gain matrix ($\hat{G}$)
Λ	= RGA matrix ($\Lambda = G \otimes (G^{-1})^T$)
$\hat{v}_{k,i,j}$	= k^{th} estimated impulse response coefficient for $\hat{g}_{i,j}(q^{-1})$
ν	= is a random number $\upsilon \sim N(0, \sigma_\upsilon^2)$
$\hat{\Pi}$	= sample covariance matrix of $\hat{G}(e^{i\omega})$ (chapter 5)
ρ	= a matrix of random numbers
σ_q	= standard deviation of the λ_q
$\sigma_{\det(\hat{G})}$	= standard deviation of the determinant of the estimated gain matrix
$\hat{\sigma}_{\lambda,i}$	= standard deviation of the ith eigenvalue ($\hat{\lambda}_i$)
$\underline{\sigma}$	= smallest singular value
$\bar{\sigma}$	= largest singular value
$\sigma_{Signal}/\sigma_{Noise}$	= standard deviation of the signal divided by the standard deviation of the noise, otherwise known as signal-to-noise ratio
Φ	= spectrum of a signal (chapter 5)
ω	= frequency with the range of 0 to π, where $\omega = \pi$ corresponds to the sampling interval (chapter 5)
Ω	= the covariance of the inexact linear equality constraint

Acronyms

ARMAX	Autoregressive moving average exogenous variables model
c.d.f.	Cumulative density function
CSC	Controller stability criteria
CV	Controlled variable
DMC	Dynamic Matrix Control, Cutler and Ramaker 1980
GA	Genetic algorithm
i.i.d.	independently identically distributed
L.H.S.	Left hand side
LP	Linear Programming
LTV	linear time variant
MIMO	Multi-Input Multi-Output
MISO	Multi-Input Single-Output
MPC	Model Predictive Control
MV	Manipulated Variable
p.d.f.	probability density function
PCA	Principal Component Analysis
PRBS	Pseudo random binary signal
QDMC	Quadratic Dynamic Matrix Control
RBS	Random binary signal
RGA	Relative gain array
RHP	Right half plane
R.H.S.	Right hand side
RMPCT	Robust Model Predictive Control Technology
SCS	Stable Control System
SISO	Single-Input Single-Output
SMC-Idcom	Shell Multivariable Control-Identification-COMmand
SMOC	Shell's Multivariable Optimizing Controller
SVD	Singular value decomposition
SVT	Singular Value Thresholding

USC	Unstable Control System

Appendices

Appendix 1: The Controller Stability Criteria

The controller stability criteria (CSC), which in this thesis is used as a form of model quality metric, was first published by Garcia and Morari (1985). This paper was later followed by two correspondences by Mijares and Holland (1987) and Morari (1987). A similar idea was also developed by Mijares et al. (1986), but for systems that are more restrictive.

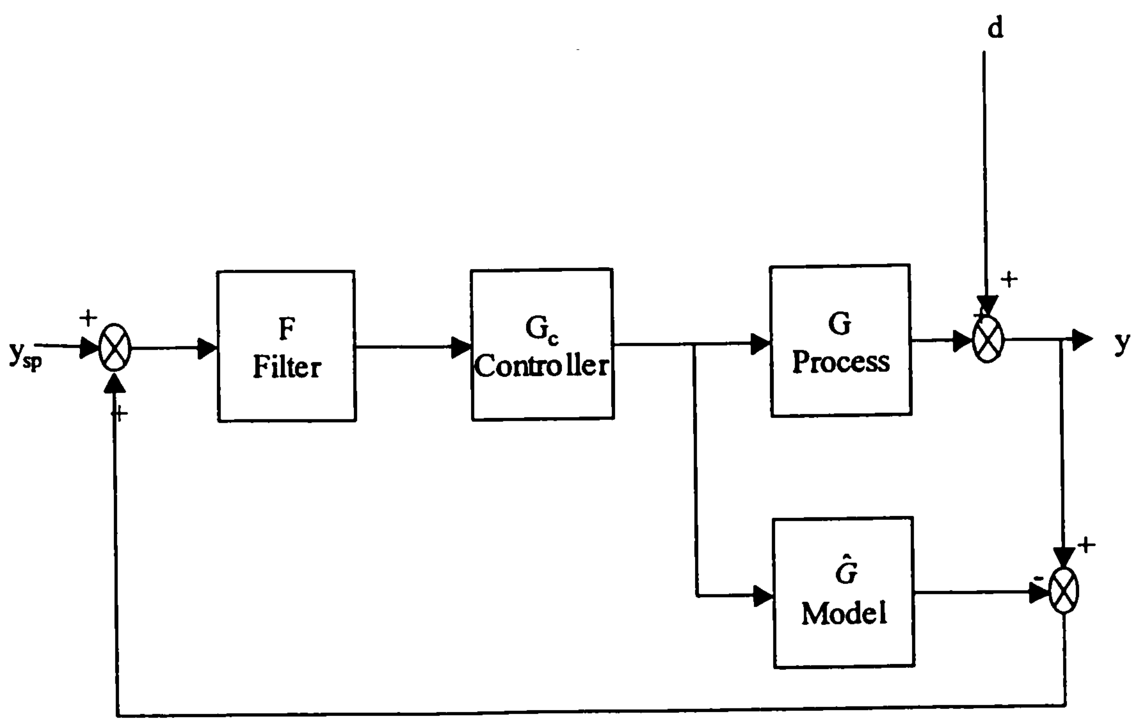


Figure A.1 Internal Model Control (IMC) structure with filter for CSC

Traditionally, in the presence of plant/model mismatch, the controller stability has been attained by addition of a filter in the IMC structure (Figure A.1). In which case, a diagonal first-order filter with a specific time constant is all that is needed for robustness (and stability):

$$F(z) = diag\left\{\frac{1-\alpha_i}{1-\alpha_i z^{-1}}\right\}, 0 \le \alpha_i < 1, \forall i \tag{A.1}$$

There exists an α^* such that the system is closed-loop stable for all α_i in the open interval $\alpha^* \le \alpha_i < 1, \forall i$, if and only if the model plant mismatch satisfies the following condition (this condition is referred to as the controller stability criteria in this thesis):

$$\mathrm{Re}\{\lambda_i(G\hat{G}^{-1})\} > 0, \forall i \tag{A.2}$$

where $\lambda_i(A)$ denotes the *i*th eigenvalue of A

G is the true plant's gain matrix, which is open-loop stable

$\hat{G}$ is the estimated gain matrix

Note that the violation of the above condition is similar to the case that in a SISO closed-loop system has positive feedback (i.e., the gain is estimated with the incorrect sign). For the SISO system, this is a very severe error. For the MIMO system, this condition requires that all the eigenvalues of $G\hat{G}^{-1}$ lie on the right half plane (RHP) (of the complex plane).

At this time a few terms are defined. The condition provided by (A.2) is defined as the controller stability criteria (CSC). If a system satisfies this condition, it is said to be a stable control system (SCS). Using the uncertainty in the estimated model and the true model, the probability of a stable control system (P(SCS)) can be estimated. Conversely, a system that does not satisfy the CSC is an unstable control system (UCS). The probability of such an event is the probability of unstable control system (P(UCS)).

The CSC (A.2) is independent of the controller design. If the real part of one or more of the eigenvalues is negative, the closed-loop system will not be stabilizable using any controller designed, which uses the estimated model.

Koung (1991) simulations illustrate that an improvement in the CSC will also improve other metrics of model quality. In his case, he uses the $\left\|\hat{G}G^{-1} - I\right\|_2$, which is associated with the closed-loop performance. His simulation results show that a lower probability of unstable control system also lowers $\left\|\hat{G}G^{-1} - I\right\|_2$ which in turn improves the closed-loop controller performance. The simulation results by Dayal (1996) shows a similar model quality improvement for small gains theorem (i.e., if the model quality based on small gains theorem improves, so will the probability of stable control systems). In this thesis, the emphasis is on the model quality evaluation based on the CSC, since this measure of model quality assessment is the minimum model quality that is required for MIMO MPC design.

Appendix 2: Propagation of Model Uncertainty to the Angle between Gain Vectors for 2x2 Systems

The angle between two gain vectors (α) can be defined by:

$$\alpha = \tan^{-1}\left(\frac{g_{1,2}}{g_{1,1}}\right) - \tan^{-1}\left(\frac{g_{2,2}}{g_{2,1}}\right) \tag{A.3}$$

The above expression may be used to propagate the uncertainty of the gain elements to estimate the uncertainty in the angle using Taylor series approximation of the above function:

$$\mathrm{var}(\alpha)=\frac{g_{1,2}^2\sigma_{1,1}^2}{g_{1,1}^4\left(1+\frac{g_{1,2}^2}{g_{1,1}^2}\right)^2}+\frac{g_{2,2}^2\sigma_{2,1}^2}{g_{2,1}^4\left(1+\frac{g_{2,2}^2}{g_{2,1}^2}\right)^2}+\frac{\sigma_{1,2}^2}{g_{1,1}^2\left(1+\frac{g_{1,2}^2}{g_{1,1}^2}\right)^2}+\frac{\sigma_{2,2}^2}{g_{2,1}^2\left(1+\frac{g_{2,2}^2}{g_{2,1}^2}\right)^2}$$
$$-2\frac{\sigma_{[1,1][2,1]}g_{1,2}g_{2,2}}{\left(g_{1,1}^2+g_{1,2}^2\right)\left(g_{2,1}^2+g_{2,2}^2\right)}-2\frac{\sigma_{[1,1][1,2]}g_{1,2}g_{1,1}}{\left(g_{1,1}^2+g_{1,2}^2\right)}+2\frac{\sigma_{[1,1][2,2]}g_{1,2}g_{2,1}}{\left(g_{1,1}^2+g_{1,2}^2\right)\left(g_{2,1}^2+g_{2,2}^2\right)}$$
$$+2\frac{\sigma_{[2,1][1,2]}g_{2,2}g_{1,1}}{\left(g_{2,1}^2+g_{2,2}^2\right)\left(g_{1,1}^2+g_{1,2}^2\right)}-2\frac{\sigma_{[2,1][2,2]}g_{2,2}g_{2,1}}{\left(g_{2,1}^2+g_{2,2}^2\right)^2}-2\frac{\sigma_{[1,2][2,2]}g_{1,1}g_{2,1}}{\left(g_{1,1}^2+g_{1,2}^2\right)\left(g_{2,1}^2+g_{2,2}^2\right)}$$

where $g_{i,j}$ is the gain element (i,j) of the gain matrix

$\sigma_{[i,j][k,l]}$ is the covariance associated with $g_{i,j}$ and $g_{k,l}$

α is the angle between the two gain vectors

The above expression evaluates the variance in the angle between the gain vectors, given the uncertainty in the gain elements. Although this equation is for the angle in 2×2 systems, similar expressions can be derived for angles in 3×3 systems as well. In larger than 2×2 systems, there are multiple angles to be considered. In each case, the angle is defined as the angle between the associated gain vector and the hyper-plane defined by all the other gain vectors.

Appendix 3: Linear Equality Constraints

If the model that is being estimated is assumed to have uncorrelated noise, then the identification problem is simplified to a Least Square problem (Ljung 1999). In which case, the FIR type model can be written as:

$$Y = X\beta + \varepsilon \tag{A.4}$$

Where Y is the process outputs in the form of a matrix $(l \times 1)$

X is the process inputs and their lagged values ($l \times k$) (see Ljung 1999, for the exact structure of such a matrix) note that this is different from how X is defined in the thesis. This was due to X as it is defined here being typical in statistical literature; however, the way it is defined in the thesis is typical to control literature

β is the FIR parameters ($k \times 1$)

ε is the white noise sequence in the form of a matrix ($l \times 1$)

In this case, the linear equality constraints may be written as:

$$A\beta = c \tag{A.5}$$

where A is a matrix ($j \times k$) of known prior information that expresses the structure of the information

c is a vector ($j \times 1$) of known elements

Then the Least Square minimization problem is:

$$\begin{aligned} &\text{Min } \varepsilon^T \varepsilon \\ &\text{s.t. } A\beta = c \end{aligned} \tag{A.6}$$

The Lagrangian function of the above minimization problem can be written as:

$$L(\beta,\lambda) = \varepsilon^T \varepsilon + \left(\beta^T A^T - c^T\right)\lambda \tag{A.7}$$

Taking the derivatives and setting them to zero results in:

$$\frac{\partial L}{\partial \beta} = -2X^T Y + 2X^T X\beta + \mathrm{A}^T \lambda = 0 \tag{A.8}$$

Using those equations, Seber (1977) has shown that the solution to this least square problem is given by $\hat{\beta}_H$:

$$\hat{\beta}_H = \hat{\beta} + \left(X^T X\right)^{-1} A^T \left[A\left(X^T X\right)^{-1} A^T\right]^{-1} \left(c - A\hat{\beta}\right) \tag{A.9}$$

where $\hat{\beta}$ is the solution to the original least square problem with no constraints (hence $\hat{\beta} = \left(X^T X\right)^{-1} X^T Y$). The covariance of the above expression (from Judge et. al. 1980) is given by:

$$E\left[\left(\hat{\beta}_H - E\left(\hat{\beta}_H\right)\right)\left(\hat{\beta}_H - E\left(\hat{\beta}_H\right)\right)^T\right] = \sigma^2\left[\left(X^T X\right)^{-1} - \left(X^T X\right)^{-1} A^T \left[A\left(X^T X\right)^{-1} A^T\right]^{-1} A\left(X^T X\right)^{-1}\right]$$

where $\sigma^2 = \varepsilon^T \varepsilon$

(A.10)

This suggests, as expected, that the least square with the equality constraint has a lower variance in its parameters than when there is no constraint.

Appendix 4: Linear Inexact Equality Constraint

While in the last case it was assumed that the equality constraint is exact, there are cases when inexact prior information is provided. In such a case, the constraint may be written as:

$$A\beta + v = c \tag{A.11}$$

where v is an unobservable random vector ($j \times 1$) which is normally distributed with 0 mean and known covariance of Ω ($j \times j$). Then the point estimator of β is:

$$\hat{\beta}_R = \left(\sigma^{-2} X^T X + \sigma^{-2} A^T \Omega^{-1} A\right)^{-1} \left(\sigma^{-2} X^T Y + \sigma^{-2} A^T \Omega^{-1} c\right) \tag{A.12}$$

with a covariance matrix:

$$E\left[\left(\hat{\beta}_R - E(\hat{\beta}_R)\right)\left(\hat{\beta}_R - E(\hat{\beta}_R)\right)^T\right] = \sigma^2 \left[\left(X^T X\right)^{-1} + A^T \Omega^{-1} A\right]^{-1} \tag{A.13}$$

To estimate the $\hat{\beta}_R$ value of σ is required. However, this will not be known until the regression is performed. It was determine (Judge et. al. 1980) that a good estimate for σ is:

$$\hat{\sigma}^2 = \frac{\left(Y - X\hat{\beta}\right)^T \left(Y - X\hat{\beta}\right)}{l - k} \tag{A.14}$$

where l is the number of observations

k is the number of parameters

Using this method, uncertainty in constraints can be transformed and utilized in parameter estimation. This method can also be used in providing initial guess for implementation of upper and lower bound soft constraints problems. For example, if it is known a priori that β_1 lies between –½ and ½ and β_2 lies between –¼ and ¼. Also assuming that this is the 95% confidence interval, which would correspond to two times the σ, and the parameters are normally distributed with mean 0, the range of $\beta_1 = \pm\sqrt{\frac{1}{16}}$ and that of $\beta_2 = \pm\sqrt{\frac{1}{64}}$ result in the covariance matrix of:

$$\Omega = \begin{pmatrix} \frac{1}{16} & 0 \\ 0 & \frac{1}{64} \end{pmatrix} \tag{A.15}$$

Then the estimate of the parameters using equation (A.12) and (A.15) can be used as an initial guess to a QP. This will assist the optimizer in finding a solution to the problem faster.

Appendix 5: Variance of the determinant

It was shown in (A.10) the variance of the β_H when estimated using CLS is estimated by:

$$E\left[\left(\hat{\beta}_H - E(\hat{\beta}_H)\right)\left(\hat{\beta}_H - E(\hat{\beta}_H)\right)^T\right] = \sigma^2\left[\left(X^TX\right)^{-1} - \left(X^TX\right)^{-1}A^T\left[A\left(X^TX\right)^{-1}A^T\right]^{-1}A\left(X^TX\right)^{-1}\right]$$

where $\sigma^2 = \varepsilon^T\varepsilon$

(A.16)

in the case of MIMO systems for the gain matrix this can be written as:

$$\mathrm{var}(\hat{G}) = \Sigma_{\hat{G},CLS}$$

$$= \begin{pmatrix} \sigma_1^2\left[(X^TX)^{-1} - (X^TX)^{-1}A_1^T\left[A_1(X^TX)^{-1}A_1^T\right]^{-1}A_1(X^TX)^{-1}\right],\ 0, 0, \cdots, 0 \\ 0,\ \sigma_2^2\left[(X^TX)^{-1} - (X^TX)^{-1}A_2^T\left[A_2(X^TX)^{-1}A_2^T\right]^{-1}A_2(X^TX)^{-1}\right],\ 0, \cdots, 0 \\ \vdots \\ 0, \cdots, 0,\ \sigma_{n_y}^2\left[(X^TX)^{-1} - (X^TX)^{-1}A_{n_y}^T\left[A_{n_y}(X^TX)^{-1}A_{n_y}^T\right]^{-1}A_{n_y}(X^TX)^{-1}\right] \end{pmatrix}$$

(A.17)

where σ_i is the standard deviation of the i^{th} outputs noise

A_i is the linear constraint associated with output i

$\Sigma_{\hat{G}}$ is the covariance of the gain matrix ($n_y \times n_y$)

Similarly, in the case of inexact prior knowledge the covariance matrix of the gain matrix is:

$$\mathrm{var}(\hat{G}) = \Sigma_{\hat{G},OLS} = \begin{pmatrix} \sigma_1^2\left[(X^TX)^{-1}\right],\ 0, 0, \cdots, 0 \\ 0,\ \sigma_2^2\left[(X^TX)^{-1}\right],\ 0, \cdots, 0 \\ \vdots \\ 0, \cdots, 0,\ \sigma_{n_y}^2\left[(X^TX)^{-1}\right] \end{pmatrix} \quad \text{(A.18)}$$

$$\operatorname{var}\left(\hat{G}\right) = \Sigma_{\hat{G},inexact} = \begin{pmatrix} \sigma_1^2\left[\left(X^TX\right)+A_1^T\Omega_1^{-1}A_1\right]^{-1},\ 0,0,\ \cdots,0 \\ 0,\ \sigma_2^2\left[\left(X^TX\right)+A_2^T\Omega_2^{-1}A_2\right]^{-1},\ 0,\ \cdots,0 \\ \vdots \\ 0,\ \cdots,0,\ \sigma_{n_v}^2\left[\left(X^TX\right)+A_{n_v}^T\Omega_{n_v}^{-1}A_{n_v}\right]^{-1} \end{pmatrix}$$

(A.19)

In the case of least square with inequality constraint on a parameter the variance of that parameter is:

$$\operatorname{var}\left(\hat{\beta}_{i,inequality}\right) = \sigma^2 + \frac{\delta_i^2}{2}P\left(\chi^2_{(1)} \geq \frac{\delta_i^2}{\sigma^2}\right) - \frac{\sigma^2}{2}P\left(\chi^2_{(3)} \geq \frac{\delta_i^2}{\sigma^2}\right)$$

(A.20)

where δ is the distance of the inequality constraint to the equality constraint (when $A\beta \geq c$ then $A\beta = c + \delta$)

When the inequality constraint is on multiple parameters, the above equation becomes significantly more complicated (Judge and Yancey 1986). In this case, the variance of the gain matrix was assumed to be the same as $\Sigma_{G,OLS}$; however, the diagonal element in $\Sigma_{G,OLS}$, which corresponds to the inequality constraint, is replaced with the above equation. Furthermore, it is assumed that all the covariance elements associated with the variable with the inequality constraint are zero. Because of the inequality constraint, the distribution of the parameter is no longer normal (or symmetrical) and a bias in the mean value results:

$$E\left(\hat{\beta}_{i,\,inequality}\right)=\begin{cases}\hat{\beta}_{i,OLS}+\left(\dfrac{\delta_i}{2}\right)P\left(\chi^2_{(1)}\geq\dfrac{\delta_i^2}{\sigma^2}\right)-\dfrac{\sigma^2}{\sqrt{2\pi}}P\left(\chi^2_{(2)}\geq\dfrac{\delta_i^2}{\sigma^2}\right),\text{ if }\delta_i<0\\ \hat{\beta}_{i,OLS}+\delta_i-\left(\dfrac{\delta_i}{2}\right)P\left(\chi^2_{(1)}\geq\dfrac{\delta_i^2}{\sigma^2}\right)-\dfrac{\sigma^2}{\sqrt{2\pi}}P\left(\chi^2_{(2)}\geq\dfrac{\delta_i^2}{\sigma^2}\right),\text{ if }\delta_i\geq 0\end{cases}$$

(A.21)

Note that when $\delta < 0$ the direction of the inequality is correct. In all of the covariance matrices stated above, an estimate of the output noise variance is required. It was determined (Judge et. al. 1980) that a good estimate for σ in all the cases is provided by (A.14).

The variance of the elements of the gain matrix can be propagated to the determinant of the gain matrix using Taylor series expansion:

$$\mathrm{var}\left(\det\left(\hat{G}\right)\right)=\sum\left(g'^T g'\Sigma_{\hat{G}}\right) \tag{A.22}$$

where $g'=\left(\dfrac{\partial|G|}{\partial G_{1,1}},\ \cdots,\ \dfrac{\partial|G|}{\partial G_{1,n_x}},\ \dfrac{\partial|G|}{\partial G_{2,1}},\ \cdots,\ \dfrac{\partial|G|}{\partial G_{n_x,n_y}}\right)$

The summation in the above equation is a summation of all the elements of the resulting matrix. The above equation may be simplified assuming that the off diagonals of the covariance matrix of the gain matrix are zero.

As was mentioned previously, if the elements are independent, the expected value of the determinant can be estimated by:

$$E\left(\det\left(\hat{G}\right)\right) = E\left(\begin{vmatrix} \hat{G}_{1,1} & \cdots & \hat{G}_{1,m} \\ \vdots & \ddots & \vdots \\ \hat{G}_{m,1} & \cdots & \hat{G}_{m,m} \end{vmatrix}\right) = \begin{vmatrix} E\left(\hat{G}_{1,1}\right) & \cdots & E\left(\hat{G}_{1,m}\right) \\ \vdots & \ddots & \vdots \\ E\left(\hat{G}_{m,1}\right) & \cdots & E\left(\hat{G}_{m,m}\right) \end{vmatrix}$$

(A.23)

Using the above expected value of the determinant and the variance of determinant; assuming that the determinant is normally distributed, the probability of the determinant changing sign can be easily estimated. This result in an estimate of P(UCS):

$$P(UCS) \approx \begin{cases} P\left(Z > \dfrac{0 - E\left(\det\left(\hat{G}\right)\right)}{\sqrt{\operatorname{var}\left(\det\left(\hat{G}\right)\right)}}\right), \text{ if } \det(G) < 0 \\ P\left(Z < \dfrac{0 - E\left(\det\left(\hat{G}\right)\right)}{\sqrt{\operatorname{var}\left(\det\left(\hat{G}\right)\right)}}\right), \text{ if } \det(G) > 0 \\ \text{The matrix is rank deficient, } \text{if } \det(G) = 0 \end{cases}$$

(A.24)

This method of propagation of the uncertainty to estimate the probability of UCS can also be used for other estimators (as long as an estimate of variance and mean is available) such as: Ridge-Regression, Pretest estimator, James and Stein estimator, Positive rule estimator, and other methods.

Appendix 6: Monte Carlo Simulation Results

Many different systems with and without dynamics were considered to evaluate the effect of prior knowledge on the quality of the model. The simulation results presented are based on one 2x2 system with dynamics, two 2x2 systems without dynamics and one 5x5 system without dynamics. Other simulation results, which are not presented here, were based on 5x5 systems with dynamics, 10x10 and 20x20 systems without dynamics. In each Monte Carlo, study 40 to 100 different constraints were considered. As would be expected due to the large amount of simulation results, only a very small portion of the result is presented here. The following is a list of the systems and the Monte Carlo settings that the simulation results presented here are based on.

Simulation setting 1:

The following 2x2 system appears to show a result similar to more complicated 5x5 systems (whose simulation results are not shown here):

$$G_1\left(q^{-1}\right)=\begin{pmatrix} \dfrac{2q^{-3}}{1-.8q^{-1}} & \dfrac{-q^{-2}}{1-.6q^{-1}} \\ \dfrac{q^{-4}}{1-q^{-1}+.25q^{-2}} & \dfrac{-q^{-3}}{1-.9q^{-1}+.6q^{-2}} \end{pmatrix}$$

$$N\left(q^{-1}\right)=\begin{pmatrix} \dfrac{1}{1-.7q^{-1}} & 0 \\ 0 & \dfrac{1}{1-.9q^{-1}} \end{pmatrix}$$

Therefore,

$$G_1=\begin{pmatrix} 10 & -2.5 \\ 4 & -1.43 \end{pmatrix}$$

The white noise used by the noise model was an i.i.d. which was distributed N(0,1). The following are some of the other settings for this Monte Carlo simulation:

- PRBS magnitude: 0.1
- PRBS switching time: 4
- Number of observations collected in each realization: 1000
- Number of Monte Carlo realizations: 500

Simulation setting 2:

In this case, the system was simulated with no dynamics with the following gain matrix:

$$G_2 = \begin{pmatrix} 5 & 5 \\ 0.1 & 0 \end{pmatrix}$$

- PRBS magnitude: 0.025
- Number of observations collected in each realization: 100
- Number of Monte Carlo realizations: 1000
- Variance of the added noise to all the outputs: 1

Simulation setting 3:

In this case, the signal-to-noise ratio was increased. The same simulation setting as 2 was used, with the following changes:

- PRBS magnitude: 0.25
- Number of Monte Carlo realization: 5000

Simulation setting 4:

Once again the signal-to-noise ratio was increased again. The same simulation setting as 2 was used, with the following changes:

- PRBS magnitude: 2.5
- Number of Monte Carlo realization: 9308

Simulation setting 5:

A different 2x2 gain matrix was considered:

$$G_3 = \begin{pmatrix} 5 & 5 \\ 0.2 & 0.1 \end{pmatrix}$$

- PRBS magnitude: 0.025
- Number of observations collected in each realization: 100
- Number of Monte Carlo realizations: 5000
- Variance of the added noise to all the outputs: 1

Simulation setting 6:

In this case, the same system as simulation setting 5 was considered with a higher signal-to-noise ratio. The changes to simulation setting 5 were:

- PRBS magnitude: 0.25

Simulation setting 7:

The signal-to-noise ratio was increased once more, to evaluate the effect of high signal-to-noise ratio on the effectiveness of using prior knowledge. The same simulation setting as 5 was used, with the following changes:

- PRBS magnitude: 2.5

Simulation setting 8:

The following 5x5 system was considered to see the effect of constraint for larger systems:

$$G_4 = \begin{pmatrix} 10 & -10 & 1 & .5 & .6 \\ 4 & -1.3 & -.2 & .75 & .6 \\ 1 & 10 & -1 & 1.5 & 1 \\ 0 & -5.5 & 0 & 0 & .25 \\ 1 & 6 & 3 & 10 & 6 \end{pmatrix}$$

- PRBS magnitude: 0.25
- Number of observations collected in each realization: 500
- Number of Monte Carlo realizations: 221
- Variance of the added noise to all the outputs: 1

Simulation setting 9:

In this case, inequality constraints on different gain element relation were enforced. The inequality constraints were formulated in such a fashion that they were 0.1 (in magnitude) away from the true value. The simulation setting were similar to 8, with the following changes:

- Number of Monte Carlo realizations: 1000

Simulation setting 10:

In this case, inequality constraints on different gain element were enforced. The inequality constraints were formulated in such a fashion that they were 0.001 (in magnitude) away from the true value. The simulation settings were similar to 8, with the following changes:

- Number of Monte Carlo realizations: 2000

Simulation setting 11:

The same simulation setting as 8 was used, with the following changes:

- PRBS magnitude: 0.5
- Number of Monte Carlo realization: 787

Table A.1: The effect of model uncertainty on P(USC) (base case)

Gain Matrix Number	Simulation Setting	Type Constraint	of P(USC) based on Monte Carlo	MSEG
1	1	No Constraint	0.466	5.440
2	2	No Constraint	0.569	7.450
2	3	No Constraint	0.414	0.757
2	4	No Constraint	0.042	0.075
3	5	No Constraint	0.572	7.559
3	6	No Constraint	0.442	0.765
3	7	No Constraint	0.038	0.075
4	8	No Constraint	0.326	0.894
4	9	No Constraint	0.308	0.885
4	10	No Constraint	0.332	0.887
4	11	No Constraint	0.191	0.441

Table A.2: Effect of equality constraint on P(USC)

System Number	Simulation setting	Type Constraint	of P(USC)	Mean(norm)
1	1	$g_{1,1} = 10$	0.462	4.953
1	1	$g_{1,2} = -2.5$	0.480	5.150
1	1	$g_{2,1} = 4$	0.478	4.821
1	1 *	$g_{2,2} = -1.429$	0.254	3.270
2	2	$g_{1,1} = 5$	0.534	6.353
2	2	$g_{1,2} = 5$	0.538	6.324
2	2	$g_{2,1} = .1$	0.543	6.362
2	2	$g_{2,2} = 0$	0.541	6.250
2	3	$g_{1,1} = 5$	0.415	0.643
2	3	$g_{1,2} = 5$	0.414	0.640
2	3 *	$g_{2,1} = .1$	0.388	0.640
2	3 *	$g_{2,2} = 0$	0.394	0.643
2	4 *	$g_{2,1} = .1$	0.007	0.065
2	4 *	$g_{2,2} = 0$	0.008	0.065
3	5	$g_{1,1} = 5$	0.542	6.390
3	5	$g_{1,2} = 5$	0.545	6.393
3	5	$g_{2,1} = .2$	0.517	6.429
3	5	$g_{2,2} = .1$	0.560	6.425
3	6 *	$g_{2,1} = .2$	0.408	0.645
3	6 *	$g_{2,2} = .1$	0.410	0.651
3	7 *	$g_{2,1} = .2$	0.009	0.063
3	7 *	$g_{2,2} = .1$	0.005	0.063
4	8	$g_{1,1} = 10$	0.326	0.876
4	8	$g_{1,2} = -10$	0.326	0.876
4	8	$g_{1,3} = 1$	0.321	0.874
4	8	$g_{1,4} = .5$	0.326	0.878
4	8	$g_{1,5} = .6$	0.326	0.875
4	8	$g_{2,1} = 4$	0.321	0.875
4	8	$g_{2,2} = -1.3$	0.326	0.877
4	8	$g_{2,3} = -.2$	0.326	0.879
4	8 *	$g_{2,4} = .75$	0.294	0.873
4	8 *	$g_{2,5} = .6$	0.285	0.875
4	8	$g_{4,1} = 0$	0.326	0.873
4	8	$g_{4,3} = 0$	0.326	0.874
4	8	$g_{4,4} = 0$	0.317	0.874

* Cases where substantial improvement was noticed in P(USC)

Table A.3 (a): Effect of inequality constraint on P(USC)

System Number	Simulation setting	Type Constraint	of P(USC)	Mean(norm)
1	1	$g_{1,1} \leq 12$	0.468	5.33
1	1	$g_{1,2} \leq -2$	0.484	5.29
1	1	$g_{2,1} \leq 4.8$	0.456	5.17
1	1 *	$g_{2,2} \leq -1.12$	0.218	4.28
1	1	$8 \leq g_{1,1} \leq 12$	0.466	5.22
1	1	$-3 \leq g_{1,2} \leq -2$	0.474	5.18
1	1	$3.2 \leq g_{2,1} \leq 4.8$	0.468	4.89
1	1 *	$-1.68 \leq g_{2,2} \leq -1.12$	0.288	3.28
1	1	$g_{1,1} \geq 8$	0.462	5.33
1	1	$g_{1,2} \geq -3$	0.454	5.33
1	1	$g_{2,1} \geq 3.2$	0.476	5.16
1	1 **	$g_{2,2} \geq -1.68$	0.534	4.44
1	1 *	$g_{2,2} \leq -1$	0.236	4.29
1	1 *	$g_{2,2} \leq -.5$	0.338	4.35
1	1	$g_{2,2} \leq 0$	0.436	4.42

Table A.3 (b): Effect of inequality constraint on P(USC)

System Number	Simulation setting	Type Constraint	of P(USC)	Mean(norm)
2	2	$g_{1,1} > 4.9$	0.540	6.871
2	2	$g_{1,2} > 4.9$	0.538	6.871
2	2 *	$g_{2,1} > 0$	0.450	6.920
2	2 **	$g_{2,2} > -.3$	0.656	6.853
2	2 **	$g_{2,2} > -.2$	0.664	6.851
2	2 **	$g_{2,2} > -.1$	0.667	6.849
2	2 **	$g_{2,2} > -.01$	0.677	6.849
2	2 **	$g_{2,2} > -.001$	0.677	6.849
2	2 **	$g_{2,2} > -1e-7$	0.677	6.849
2	2 *	$.3 > g_{2,2}$	0.457	6.855
2	2 *	$.2 > g_{2,2}$	0.451	6.853
2	2 *	$.1 > g_{2,2}$	0.441	6.851
2	2 *	$.01 > g_{2,2}$	0.435	6.851
2	2 *	$1e-3 > g_{2,2}$	0.434	6.851
2	2 *	$1e-7 > g_{2,2}$	0.433	6.851
2	2	$5.1 > g_{1,1}$	0.564	6.933
2	2	$5.1 > g_{1,2}$	0.570	6.904
2	2 **	$0.2 > g_{2,1}$	0.660	6.893
2	3 *	$g_{2,1} > 0$	0.327	0.704
2	3 **	$g_{2,2} > -.3$	0.431	0.722
2	3 **	$g_{2,2} > -.2$	0.449	0.711
2	3 **	$g_{2,2} > -.1$	0.473	0.701
2	3 **	$g_{2,2} > -.01$	0.512	0.697
2	3 **	$g_{2,2} > -.001$	0.517	0.697
2	3 **	$g_{2,2} > -1e-7$	0.517	0.697
2	3 *	$.3 > g_{2,2}$	0.381	0.727
2	3 *	$.2 > g_{2,2}$	0.359	0.716
2	3 *	$.1 > g_{2,2}$	0.329	0.707
2	3 *	$.01 > g_{2,2}$	0.293	0.703
2	3 *	$1e-3 > g_{2,2}$	0.291	0.703
2	3 *	$1e-7 > g_{2,2}$	0.291	0.703
2	4	$g_{2,2} > -.1$	0.042	0.075
2	4	$.1 > g_{2,2}$	0.042	0.075
2	4 *	$.01 > g_{2,2}$	0.007	0.070
2	4 *	$1e-3 > g_{2,2}$	0.005	0.070
2	4 *	$1e-7 > g_{2,2}$	0.005	0.070

Table A.3 (c): Effect of inequality constraint on P(USC)

System Number	Simulation setting	Type Constraint	of P(USC)	Mean(norm)
3	5	$g_{1,2} > 4.9$	0.541	6.981
3	5 *	$g_{2,1} > 0.1$	0.431	6.966
3	5 **	$g_{2,2} > 0$	0.678	6.963
3	5 *	$.2 > g_{2,2}$	0.457	7.022
3	5	$5.1 > g_{1,1}$	0.567	6.976
3	5	$5.1 > g_{1,2}$	0.575	6.972
3	5 **	$0.2 > g_{2,1}$	0.659	7.023
3	6	$g_{1,1} > 4.9$	0.444	0.710
3	6	$g_{1,2} > 4.9$	0.440	0.710
3	6 *	$g_{2,1} > 0.1$	0.346	0.708
3	6 **	$g_{2,2} > 0$	0.512	0.713
3	6 *	$.2 > g_{2,2}$	0.349	0.711
3	6	$5.1 > g_{1,1}$	0.440	0.711
3	6	$5.1 > g_{1,2}$	0.444	0.712
3	6 **	$0.2 > g_{2,1}$	0.507	0.711
3	7	$g_{1,1} > 4.9$	0.038	0.075
3	7	$g_{1,2} > 4.9$	0.038	0.075
3	7	$g_{2,1} > 0.1$	0.037	0.075
3	7	$g_{2,2} > 0$	0.038	0.075
3	7	$.2 > g_{2,2}$	0.038	0.074
3	7	$5.1 > g_{1,1}$	0.038	0.075
3	7	$5.1 > g_{1,2}$	0.038	0.074
3	7	$0.2 > g_{2,1}$	0.038	0.075

Table A.3 (d): Effect of inequality constraint on P(USC)

System Number	Simulation setting	Type Constraint	of P(USC)	Mean(norm)
4	9	$g_{1,1} > 10$	0.308	0.877
4	9	$g_{1,2} > -10$	0.308	0.876
4	9	$g_{1,3} > 1$	0.308	0.880
4	9 **	$g_{1,4} > 0.5$	0.346	0.877
4	9	$g_{1,5} > 0.6$	0.308	0.878
4	9	$g_{2,1} > 4$	0.308	0.874
4	9	$g_{2,2} > -1.3$	0.308	0.874
4	9 **	$g_{2,3} > -0.2$	0.333	0.876
4	9	$g_{2,4} > 0.75$	0.295	0.879
4	9 **	$g_{2,5} > 0.6$	0.346	0.879
4	9	$g_{3,1} > 1$	0.308	0.877
4	9	$g_{3,2} > 10$	0.308	0.877
4	9	$g_{3,3} > -1$	0.308	0.879
4	9	$g_{3,4} > 1.5$	0.321	0.876
4	9 *	$g_{3,5} > 1$	0.282	0.877
4	9	$g_{4,1} > 0$	0.308	0.876
4	9	$g_{4,2} > -5.5$	0.308	0.878
4	9	$g_{4,3} > 0$	0.308	0.877
4	9	$g_{4,4} > 0$	0.321	0.881
4	9	$g_{4,5} > 0.25$	0.295	0.880
4	9	$g_{5,1} > 1$	0.308	0.877
4	9	$g_{5,2} > 6$	0.308	0.876
4	9	$g_{5,3} > 3$	0.308	0.878
4	9	$g_{5,4} > 10$	0.308	0.880
4	9	$g_{5,5} > 6$	0.308	0.879

* Cases where substantial improvement was noticed in P(USC) due to an equality constraint which affected the low-gain direction

** Cases where model deterioration was noticed in P(USC) due to an equality constraint which affected the low-gain direction

Table A.4: Effect of Monotonicity and Windowing Constraint

System Number	Simulation setting	Type of Constraint	P(USC)	Mean(norm)
1	1	monotonicity $g_{1,1}(q^{-1})$	0.458	5.560
1	1	monotonicity $g_{1,2}(q^{-1})$	0.500	5.430
1	i	monotonicity $g_{2,1}(q^{-1})$	0.500	5.360
1	1	$g_{1,1}(q^{-1}) \leq 2.92q^{-2}/(1-.757q^{-1})$	47	5.33
1	1	$1.36q^{-2}/(1-.83q^{-1}) \leq g_{1,1}(q^{-1})$	46.2	5.32
1	1	$1.36q^{-2}/(1-.83q^{-1}) \leq g_{1,1}(q^{-1}) \leq 2.92q^{-2}/(1-.757q^{-1})$	46.8	5.16
1	1	$g_{1,2}(q^{-1}) \geq -1.41q^{-1}/(1-.52q^{-1})$	45.2	5.34
1	1	$g_{1,2}(q^{-1}) \leq -.695q^{-1}/(1-.65q^{-1})$	51.6	5.48
1	1	$-1.41q^{-1}/(1-.52q^{-1}) \leq g_{1,2}(q^{-1}) \leq -.695q^{-1}/(1-.65q^{-1})$	47.6	5.17
1	1	$g_{2,1}(q^{-1}) \leq 1.68q^{-3}/(1-.65q^{-1})$	44.8	5.24
1	1	$g_{2,1}(q^{-1}) \geq .83q^{-3}/(1-.74q^{-1})$	48.4	5.21
1	1	$.83q^{-3}/(1-.74q^{-1}) \leq g_{2,1}(q^{-1}) \leq 1.68q^{-3}/(1-.65q^{-1})$	46.8	4.87
1	1	$g_{2,2}(q^{-1}) \geq -2q^{-2}/(1-.5q^{-1})$	51.8	4.81
1	1 *	$g_{2,2}(q^{-1}) \leq -.2q^{-2}/(1-.8q^{-1})$	12.2	4.58
1	1 *	$-2q^{-2}/(1-.5q^{-1}) \leq g_{2,2}(q^{-1}) \leq -.2q^{-2}/(1-.8q^{-1})$	14.8	3.65

Table A.5: Effect of Constraint on the Angle

System Number	Simulation setting	Type of Constraint	P(USC)	Mean(norm)
1	1	$\alpha = -7.97$	0	0
1	1	$\alpha \leq -6.37$	0.4	0.4
1	1	$\alpha \geq -9.56$	56.2	56.2
1	1	$-9.56 \leq \alpha \leq -6.37$	0	0
1	1	$\alpha \leq -4$	0.4	3.91
1	1	$\alpha \leq -2$	0.8	3.92
1	1	$\alpha \leq -1$	1	3.92
1	1	$\alpha \leq 0$	14.8	3.93

Table A.6: Effect of Multivariate Linear Constraint
Equality constraint (and upper and lower bound constraints)

System Number	Simulation setting	Type of Constraint	P(USC)	Mean(norm)
1	1	$g_{1,1}/g_{1,2} = -4$	0.466	5.14
1	1	$g_{1,1}/g_{2,1} = 2.5$	0.468	4.83
1	1 *	$g_{1,1}/g_{2,2} = -7$	0.290	3.29
1	1	$g_{1,2}/g_{2,1} = -.625$	0.478	4.99
1	1 *	$g_{1,2}/g_{2,2} = 1.75$	0.292	3.36
1	1 *	$g_{2,1}/g_{2,2} = -2.8$	0.274	3.34
1	1	$-4.8 \leq g_{1,1}/g_{1,2} \leq -3.2$	0.472	5.16
1	1	$2 \leq g_{1,1}/g_{2,1} \leq 3$	0.464	4.90
1	1 *	$-8.4 \leq g_{1,1}/g_{2,2} \leq -5.6$	0.276	3.30
1	1	$-.7 \leq g_{1,2}/g_{2,1} \leq -.5$	0.478	4.99
1	1 *	$1.4 \leq g_{1,2}/g_{2,2} \leq 2.1$	0.298	3.33
1	1 *	$-3.16 \leq g_{2,1}/g_{2,2} \leq -2.24$	0.264	3.32
3	5	$g_{2,1}-g_{2,2} = .1$	0.527	6.463
3	5 *	$g_{2,1}+g_{2,2} = .3$	0.519	6.396
3	5 *	$g_{1,1}-g_{1,2} = 0$	0.514	6.405
3	5	$g_{1,1}+g_{1,2} = 0$	0.552	6.386
3	6 *	$g_{2,1}-g_{2,2} = .1$	0.013	0.647
3	6	$g_{2,1}+g_{2,2} = .3$	0.445	0.650
3	6	$g_{1,1}-g_{1,2} = 0$	0.443	0.648
3	6	$g_{1,1}+g_{1,2} = 0$	0.442	0.652
3	7 *	$g_{2,1}-g_{2,2} = .1$	0.000	0.063
3	7	$g_{2,1}+g_{2,2} = .3$	0.038	0.063
3	7	$g_{1,1}-g_{1,2} = 0$	0.039	0.063
3	7	$g_{1,1}+g_{1,2} = 0$	0.038	0.063
4	8	$g_{1,1}+g_{1,2} = 0$	0.326	0.875
4	8	$g_{1,1}+g_{1,3} = 11$	0.326	0.874
4	8	$g_{1,1}+g_{1,4} = 10.5$	0.335	0.876
4	8	$g_{1,1}+g_{1,5} = 10.6$	0.303	0.876
4	8	$g_{1,2}+g_{1,3} = -9$	0.326	0.874
4	8	* $g_{4,2}+g_{4,4} = -5.5$	0.299	0.877
4	8	$g_{4,2}+g_{4,5} = -5.25$	0.321	0.874
4	8	$g_{4,3}-g_{4,4} = 0$	0.330	0.877
4	8	* $g_{4,3}-g_{4,5} = 0.25$	0.299	0.876
4	8	$g_{4,4}-g_{4,5} = 0.25$	0.312	0.874

Table A.7: Comparison of some numerical results with analytical results for the 5x5 system with no dynamics

Gain Matrix Number	Simulation Setting	Type of Constraint	P(USC) based on Monte Carlo	P(USC) based on the p.d.f. of the det(G)
4	8	$g_{2,1} = 4$	0.321	0.329
4	8	$g_{2,4} = .75$	0.294	0.309
4	8	$g_{2,5} = .6$	0.285	0.285
4	8	$g_{3,1} = 1$	0.321	0.329
4	8	$g_{3,2} = 10$	0.321	0.329
4	9	No Constraint	0.308	0.329
4	9	$g_{1,1} > 10$	0.308	0.329
4	9	$g_{1,2} > -10$	0.308	0.328
4	9	$g_{1,3} > 1$	0.308	0.323
4	9	$g_{1,4} > 0.5$	0.346	0.339
4	9	$g_{1,5} > 0.6$	0.308	0.313
4	9	$g_{2,1} > 4$	0.308	0.328
4	9	$g_{2,2} > -1.3$	0.308	0.329
4	9	$g_{2,3} > -0.2$	0.333	0.342
4	9	$g_{2,4} > 0.75$	0.295	0.283
4	9	$g_{2,5} > 0.6$	0.346	0.359
4	9	$g_{3,5} > 1$	0.282	0.305
4	9	$g_{4,5} > 0.25$	0.295	0.306
4	10	$g_{2,4} > 0.75-0.001$	0.255	0.251
4	10	$g_{2,5} > 0.6-0.001$	0.420	0.410
4	10	$g_{3,5} > 1-0.001$	0.279	0.279
4	10	$g_{4,5} > 0.25-0.001$	0.275	0.281
4	11	No Constraint	0.191	0.188
4	11	$g_{1,1} = 10$	0.193	0.188
4	11	$g_{2,4} = .75$	0.161	0.160
4	11	$g_{2,5} = .6$	0.127	0.128
4	11	$g_{2,4} = .75$	0.173	0.175
4	11	$g_{2,5} = .6$	0.194	0.188
4	11	$g_{4,5} = 0.3$	0.180	0.176
4	11	$g_{4,2}+g_{4,5} = -5.25$	0.184	0.182
4	11	$g_{4,1}-g_{4,5} = 0.25$	0.177	0.183
4	11	$g_{4,4}-g_{4,5} = -0.25$	0.180	0.173

Appendix 7: Derivative of the Normal Cumulative Density Function

To estimate the sensitivity of the P(USC) to changes in the prior knowledge, the derivative of the normal cumulative density function for the determinant of the gain matrix is required. This can be obtained by assuming that the determinant of the gain matrix is normally distributed (this is illustrated to be assumption in chapters 2 and 4) then:

$$P\left(\det\left(\hat{G}\right)>0\right)=\int_0^\infty \frac{1}{\sqrt{2\pi}\sigma_{\hat{G}}}e^{-\frac{1}{2}\left(\frac{x-E\left(\det\left(\hat{G}\right)\right)}{\sigma_{\hat{G}}}\right)^2}dx \tag{A.25}$$

Using Leibnitz's rule results in:

$$\begin{aligned}\frac{dP\left(\det\left(\hat{G}\right)>0\right)}{d\det\left(\hat{G}\right)}&=\frac{1}{\sqrt{2\pi}\sigma_{\hat{G}}}e^{-\frac{1}{2}\left(\frac{\infty-E\left(\det\left(\hat{G}\right)\right)}{\sigma_{\hat{G}}}\right)^2}dx-\frac{1}{\sqrt{2\pi}\sigma_{\hat{G}}}e^{-\frac{1}{2}\left(\frac{0-E\left(\det\left(\hat{G}\right)\right)}{\sigma_{\hat{G}}}\right)^2}dx\\&=-\frac{1}{\sqrt{2\pi}\sigma_{\hat{G}}}e^{-\frac{1}{2}\left(\frac{0-E\left(\det\left(\hat{G}\right)\right)}{\sigma_{\hat{G}}}\right)^2}dx\\&=-P\left(\det\left(\hat{G}\right)=0\right)\end{aligned} \tag{A.26}$$

Using this result, the sensitivity of the probability of the determinant to changes in the constraint may be estimated by:

$$\text{if } \det(G)<0, \frac{dP\left(\det\left(\hat{G}\right)>0\right)}{dc}=\frac{dP\left(\det\left(\hat{G}\right)>0\right)}{d\det\left(\hat{G}\right)}\frac{d\det\left(\hat{G}\right)}{dc}=-P\left(\det\left(\hat{G}\right)=0\right)\frac{d\det\left(\hat{G}\right)}{dc}$$

$$\text{if } \det(G)>0, \frac{dP\left(\det\left(\hat{G}\right)<0\right)}{dc}=\frac{dP\left(\det\left(\hat{G}\right)<0\right)}{d\det\left(\hat{G}\right)}\frac{d\det\left(\hat{G}\right)}{dc}=P\left(\det\left(\hat{G}\right)=0\right)\frac{d\det\left(\hat{G}\right)}{dc}$$

(A.27)

The above expression suggests that the sensitivity of P(USC) to changes in prior knowledge is directly proportional to $P\left(\det\left(\hat{G}\right)=0\right)$.

Appendix 8: Sensitivity of the Determinant to Changes in Eigenvalues

The metric that is used in evaluating the sensitivity of the CSC to perturbation in the constraint is based on the determinant of the gain matrix. A constraint on the eigenvalue of the gain matrix will be in the following form:

$$Gx_i = \lambda_i x_i \tag{A.28}$$

This is based on the fundamental definition of an eigenvalue (where x is the right hand eigenvector corresponding to λ_i). When G is a nxn gain matrix, this results in n equality constraint. Assuming that the eigenvalues are independent, the sensitivity of an eigenvalue constraint on the determinant can be estimated by:

$$\frac{\partial\left|\hat{G}\right|}{\partial\lambda_i}=\frac{\partial\prod_{j=1}^{n}\lambda_j}{\partial\lambda_i}=\prod_{j=1}^{i-1}\lambda_j\times\prod_{j=i+1}^{n}\lambda_j \tag{A.29}$$

where $\hat{G}$ is the estimated gain matrix, which is $n \times n$.

λ_i is the i th eigenvalue of $\hat{G}$ $(i \le n)$

As expected, the application of this equation to the smallest eigenvalue would suggest that the determinant is most sensitive to constraint on this eigenvalue compared to the other eigenvalues. For the 5x5 gain matrix stated in chapter 3, the sensitivity of the determinant to the smallest eigenvalue is 274.4 compared to the largest eigenvalue's sensitivity of −14.8. As expected, this would suggest that the CSC is most sensitive to changes in the constraint based on the smallest eigenvalue.

Appendix 9: Theoretical estimation of the eigenvalue sensitivity

The sensitivity of the solution to changes in the constraint for linear equality is given (from (A.9)) by:

$$\frac{\partial \hat{\beta}_H}{\partial c} = \left(X^T X\right)^{-1} A^T \left[A\left(X^T X\right)^{-1} A^T\right]^{-1} \tag{A.30}$$

in the case of MIMO systems for the gain matrix, this can be written as:

$$\frac{\partial \hat{G}}{\partial c} = \begin{pmatrix} \left(X^T X\right)^{-1} A_1^T \left[A_1 \left(X^T X\right)^{-1} A_1^T\right]^{-1} \\ \vdots \\ \left(X^T X\right)^{-1} A_{n_y}^T \left[A_{n_y} \left(X^T X\right)^{-1} A_{n_y}^T\right]^{-1} \end{pmatrix} \tag{A.31}$$

where A_i is the linear constraint associated with output i

n_y is the number of process outputs

In addition, considering the sensitivity of the eigenvalues to changes in gain matrix (as shown in chapter 3):

$$\tilde{\lambda} = \lambda + \frac{y^H \Delta x}{y^H x} + O\left(\|\Delta\|^2\right) \qquad \text{(A.32)}$$

where Δ is the perturbation matrix

λ is the eigenvalue of G

$\tilde{\lambda}$ is the eigenvalue of $G+E$

x is the right eigenvector of G

y is the left eigenvector of G

$\cdot^H$ is the Hermitian of a matrix

Substituting (A.31) into (A.32) results in the following:

$$\tilde{\lambda} - \lambda = \frac{y^H \begin{pmatrix} \left(X^T X\right)^{-1} A_1^T \left[A_1 \left(X^T X\right)^{-1} A_1^T\right]^{-1} \\ \vdots \\ \left(X^T X\right)^{-1} A_m^T \left[A_m \left(X^T X\right)^{-1} A_m^T\right]^{-1} \end{pmatrix} x}{y^H x} + O\left(\|E\|^2\right) \qquad \text{(A.33)}$$

Assuming that the left and right eigenvectors do not change even though the constraint will, the sensitivity of the eigenvalue due to changes in the constraint can be estimated. For Example 3.1 (in chapter 3):

$X^T X = 500 \times I_5$

$y^H = [-0.2461 \quad 0.7400 \quad -0.4442 \quad -0.4392 \quad 0.0404]$

$x^H = [-0.0239 \quad 0.0004 \quad 0.0176 \quad -0.5402 \quad 0.8410]$

This results in the sensitivity of the smallest eigenvalue to be:

Type of Prior Knowledge	$\frac{\tilde{\lambda}_i - \lambda_i}{\varepsilon}$	$\left(\frac{\tilde{\lambda}_i - \lambda_i}{\varepsilon}\right) \Big/ \lambda_i$
$g_{2,5} = 0.6$	2.3094	5.9953
$-0.54g_{2,4}+0.84g_{2,5} = 0.099$	2.7486	7.1354
$g_{2,4} - 1.25g_{2,5} = 0$	-1.7054	-4.4273
$g_{4,3} = 0$	-0.0288	0.0748

The difference between this analysis and the sensitivity analysis in Example 3.1 (of chapter 3) is that in this case the true gain matrix is used in estimating the eigenvectors. However, in Example 3.1 an estimate of the gain matrix (which was based on a data set) was used. Even though the methods are different, the results of the Monte Carlo simulation shown in Table 3.2 compare with these results well. As the signal-to-noise ratio decreases, the estimated eigenvectors would have a larger variance and there would be a larger difference between the true eigenvectors and the estimated eigenvector. Consequently, their uncertainty also propagates non-linearly to the estimated sensitivity. This results in a non-linear behavior in the sensitivity analysis as the signal-to-noise ratio decreases.

Appendix 10: Detecting Unstable Control System

The detection of an UCS is not a trivial issue. A few different situations may suggest in practice that the controller is unstable where in fact the system is stable. Since all real systems have bounds on the process inputs and outputs, if the system does not satisfy the CSC, one of the outputs will move to their bound. In fact, only in a hypothetical system with no bounds can you be certain that the CSC has been violated. In such a system, there are no bounds on the inputs or the outputs, and if the system is an UCS, at least one of the inputs will monotonically increase or decrease (and approach $\pm\infty$).

One of the situations that may produce the same results as UCS is if the set points for the process outputs are unreachable. In such a case, the system cannot reach the set points because of constraints on the process inputs, even if the controller uses the true model. To illustrate this point, consider a set point vector of $\overrightarrow{y_{sp}}$. Then if the system is unstable in the direction of $\overrightarrow{y_{sp}}$, the system will exhibit unstability behavior even if a small change is made in that direction (i.e., $\overrightarrow{y_{sp}} \times \varepsilon$, $\varepsilon \to 0$). However, if the system is unreachable in that direction, there exists an $\varepsilon^{*}<1$, such that the set point $\overrightarrow{y_{sp}} \times \varepsilon^{*}$ is reachable. In practice this can be used to distinguish if the system violates CSC or if the set points $\overrightarrow{y_{sp}}$ are unreachable, by making small changes in direction $\overrightarrow{y_{sp}}$ and seeing if the system is stable or not.

Another possibility is that the controller is aggressively tuned (i.e., $\alpha^{*} \not\leq \alpha_i, \forall i$ in equation (A.2)). To test if this is the case, the controller can be detuned (i.e., $a_i \to 1, \forall i$). Iff the system remains unstable after detuning, can it be said that the estimated model is truly an UCS. In this author's opinion, this is not a serious issue since in most chemical systems (specially for large MIMO that this thesis concentrates on) the controllers are very detuned.

Let us consider another case, and its consequences, where the control engineer thinks that there is UCS (i.e., accepts the hypothesis that there is a UCS); however, the estimated model satisfies the CSC. In such a case, the system has not reached $\overrightarrow{y_{sp}}$ and some of the process inputs or outputs are at their bounds. In such a situation, the methodology in this thesis suggests that the model should be re-estimated with a constraint on the determinant:

$$\underset{\hat{G}}{Min}\, SSE$$
$$s.t. -\det\left(\hat{\hat{G}}\right) \times sign\left(\det\left(\hat{G}\right)\right) > 0$$

The new model is then to be used for control. Since the original model had truly not violated the CSC, the new model will result in an USC. As a result, some of the process inputs and/or outputs will move to their bounds. Notice that this is the same situation that we started with. In effect, the control system is no better or worst than its original form. This analysis is analogous to a Type II error in statistics. The hypothesis is that the control system is UCS. If we accept this hypothesis when it should be rejected, a Type II error has been made. The consequences of this type of error are not significant in our situation.

Perhaps an artificial system with a pattern-recognitioning algorithm may be used to detect UCS (this would be a good topic for future research). In this thesis, it is assumed that in most cases the engineering knowledge of the process engineer will be sufficient to detect UCS. Furthermore, it is assumed that in future better methods of detecting unstable control systems will be devised. Certainly the issue of detecting UCS is not dealt with here in the comprehensive nature that it requires. Even in the methodology described above one can find exceptional situations where the above methodology will not be capable of detecting unstable control systems (i.e. when there is a very high noise in the process output or the system is very oscillatory in nature).

Appendix 11: The Optimization Settings

All of the simulations in this thesis were performed with MATLAB. In the cases where dynamic models were estimated with constraint (mainly in Chapter 2), the Sequential Quadratic Programming (SQP) algorithm (E04UCF) of the Numerical Algorithms Group Ltd. (NAG) foundation toolbox Version 1.0.3 (R11 06-Jun-1998) was used. In this case, the gradient of both the objective function and constraint were provided. The maximum number of iterations performed was set to 200. Other parameters related to the optimizer were left at their default value.

For the cases where the model was estimated without any dynamics (as was the case in Chapters 2, 3, and 4), the SQP algorithm of the MATLAB optimization toolbox version 2.0 (R11 09-Oct-1998) was used (file name fmincon). In this case, the optimization settings were as follows:

- Gradient of both the objective function and the constraint were provided
- The maximum function evaluation allowed was set to 10^4
- The maximum number of iterations allowed was set to 200

Appendix 12: Stability of Non-Square Systems

As mentioned in chapter 4, the non-square system considered in this thesis is one in which the number of manipulated variables (n_x) exceeds the number of controlled variables (n_y). In this appendix, the controller stability criteria (CSC) for such systems are studied. In addition, for simple cases the CSC of a non-square system is explained.

Consider the following matrix manipulation:

$$\left(\hat{G}\hat{G}\right)^{+} = \left(\hat{G}^{T}\right)^{+}\hat{G}^{+}$$
$$\hat{G}^{T}\left(\hat{G}\hat{G}^{T}\right)^{+} = \hat{G}^{+}$$

Assuming that $\hat{G}\hat{G}^T$ is full rank, then $\hat{G}\hat{G}^T$ is positive definite and:

$$\hat{G}^T\left(\hat{G}\hat{G}^T\right)^{-1} = \hat{G}^+$$

Where in this appendix it is assumed that G is a non-square matrix. Substituting the above expression into the CSC for non-square systems results in:

$$\mathrm{Re}\left(\lambda_i\left(G\hat{G}^T\left(\hat{G}\hat{G}^T\right)^{-1}\right)\right) \geq 0, \forall i \tag{A.34}$$

For simplification, assume

$$Q = G\hat{G}^T\left(\hat{G}\hat{G}^T\right)^{-1}$$

then,

$$\mathrm{Re}(\lambda_i(Q)) > 0, \forall i$$

Then the following are necessary (and sufficient in the case of only 2 outputs) conditions for system stability:

$$\det(Q) = \lambda_1 \times \lambda_2 \times \cdots \times \lambda_{n_y} > 0 \tag{A.35}$$

$$trace(Q) = \lambda_1 + \lambda_2 + \cdots + \lambda_{n_y} > 0 \tag{A.36}$$

Consider the first condition, which is on the determinant of Q:

$$\det(Q) = \lambda_1 \times \lambda_2 \times \cdots \times \lambda_{n_y} > 0$$
$$= \det\left(G\hat{G}^T \left(\hat{G}\hat{G}^T\right)^{-1} \right)$$
$$= \det\left(G\hat{G}^T\right) \times \det\left(\left(\hat{G}\hat{G}^T\right)^{-1}\right)$$

Since $\hat{G}\hat{G}^T$ is positive definite, condition (A.35) can be simplified to:

$$\det\left(G\hat{G}^T\right) > 0$$

Note that the above condition by itself is a necessary but not a sufficient condition for CSC in non-square problems (when the number of outputs is greater than 1). Furthermore, in the case that the gain matrix was square, this condition is similar to a condition shown previously, which states that the determinant of the estimated model has to have the same sign as the determinant of the true model.

Consider a simple case where there are multiple inputs (n_x) but only one output, then let the true and the estimated gain matrix be:

$$G = \begin{pmatrix} g_{1,1} & g_{1,2} & \cdots & g_{1,n_x} \end{pmatrix}$$

$$\hat{G} = \begin{pmatrix} \hat{g}_{1,1} & \hat{g}_{1,2} & \cdots & \hat{g}_{1,n_x} \end{pmatrix}$$

Substituting the above expression into (A.34) gives:

$$\mathrm{Re}\left(\lambda_i\left(\begin{pmatrix} g_{1,1} & g_{1,2} & \cdots & g_{1,n_r}\end{pmatrix}\times\begin{pmatrix}\hat{g}_{1,1}\\ \hat{g}_{1,2}\\ \vdots\\ \hat{g}_{1,n_r}\end{pmatrix}\left(\begin{pmatrix}\hat{g}_{1,1} & \hat{g}_{1,2} & \cdots & \hat{g}_{1,n_r}\end{pmatrix}\times\begin{pmatrix}\hat{g}_{1,1}\\ \hat{g}_{1,2}\\ \vdots\\ \hat{g}_{1,n_r}\end{pmatrix}\right)^{-1}\right)\right)\geq 0,\forall i$$

$$\mathrm{Re}\left(\lambda_i\left(\frac{g_{1,1}\hat{g}_{1,1}+g_{1,2}\hat{g}_{1,2}+\cdots+g_{1,n_r}\hat{g}_{1,n_r}}{\hat{g}_{1,1}^2+\hat{g}_{1,2}^2+\cdots+\hat{g}_{1,n_r}^2}\right)\right)\geq 0,\forall i$$

Since the elements of both the estimated and the true gain matrix are real numbers, and $\hat{g}_{1,1}^2+\hat{g}_{1,2}^2+\cdots+\hat{g}_{1,n_r}^2 \neq 0$ then the above expression simplifies to:

$$g_{1,1}\hat{g}_{1,1}+g_{1,2}\hat{g}_{1,2}+\cdots+g_{1,n_r}\hat{g}_{1,n_r} > 0 \tag{A.37}$$

The above condition is a sufficient and necessary condition, which applies to systems with one output. Now consider the simple case that there are 2 process input and 1 process output, then equation (A.37) becomes:

$$g_{1,1}\hat{g}_{1,1}+g_{1,2}\hat{g}_{1,2} > 0$$

The implication of the above equation is that any one-gain element estimate can have the wrong sign and the system will still be stable. Furthermore, if one gain element has the wrong sign, the other gain element has to compensate for this error. However, if both of the gain elements have the wrong sign, the system will certainly be unstable. Alternatively, if both gain elements have the correct sign, the system will be stable. Utilizing (A.37), a similar analysis can be performed for any MISO system.

Consider a more realistic case, where there are 3 inputs and 2 outputs.

$$G = \begin{pmatrix} g_{1,1} & g_{1,2} & g_{1,3} \\ g_{2,1} & g_{2,2} & g_{2,3} \end{pmatrix}$$

$$\hat{G} = \begin{pmatrix} \hat{g}_{1,1} & \hat{g}_{1,2} & \hat{g}_{1,3} \\ \hat{g}_{2,1} & \hat{g}_{2,2} & \hat{g}_{2,3} \end{pmatrix}$$

Substituting the above expression into (A.34) results in the following Q matrix:

$$Q = \begin{pmatrix} g_{1,1}\hat{g}_{1,1} + g_{1,2}\hat{g}_{1,2} + g_{1,3}\hat{g}_{1,3} & g_{1,1}\hat{g}_{2,1} + g_{1,2}\hat{g}_{2,2} + g_{1,3}\hat{g}_{2,3} \\ \hat{g}_{1,1}g_{2,1} + \hat{g}_{1,2}g_{2,2} + \hat{g}_{1,3}g_{2,3} & g_{2,1}\hat{g}_{2,1} + g_{2,2}\hat{g}_{2,2} + g_{2,3}\hat{g}_{2,3} \end{pmatrix} \times \begin{pmatrix} \hat{g}_{1,1}^2 + \hat{g}_{1,2}^2 + \hat{g}_{1,3}^2 & \hat{g}_{1,1}\hat{g}_{2,1} + \hat{g}_{1,2}\hat{g}_{2,2} + \hat{g}_{1,3}\hat{g}_{2,3} \\ \hat{g}_{1,1}\hat{g}_{2,1} + \hat{g}_{1,2}\hat{g}_{2,2} + \hat{g}_{1,3}\hat{g}_{2,3} & \hat{g}_{2,1}^2 + \hat{g}_{2,2}^2 + \hat{g}_{2,3}^2 \end{pmatrix}^{-1} \tag{A.38}$$

Substituting (A.38) into (A.35) the above equation results in:

$$\begin{aligned} \det\left(G\hat{G}^T\right) = & -\hat{g}_{1,1}\hat{g}_{2,2}g_{1,2}g_{2,1} + \hat{g}_{1,1}\hat{g}_{2,2}g_{1,1}g_{2,2} + \hat{g}_{1,1}g_{1,1}g_{2,3}\hat{g}_{2,3} - \hat{g}_{1,1}g_{1,3}\hat{g}_{2,3}g_{2,1} \\ & - \hat{g}_{2,2}g_{1,2}g_{2,3}\hat{g}_{1,3} + \hat{g}_{2,2}g_{1,3}\hat{g}_{1,3}g_{2,2} + g_{1,2}g_{2,3}\hat{g}_{2,3}\hat{g}_{1,2} + g_{1,2}\hat{g}_{2,1}g_{2,1}\hat{g}_{1,2} \\ & - g_{1,3}\hat{g}_{2,3}\hat{g}_{1,2}g_{2,2} + g_{1,3}\hat{g}_{1,3}\hat{g}_{2,1}g_{2,1} - g_{1,1}g_{2,2}\hat{g}_{1,2}\hat{g}_{2,1} - g_{1,1}g_{2,3}\hat{g}_{1,3}\hat{g}_{2,1} \\ & > 0 \end{aligned} \tag{A.39}$$

Note that as mentioned previously this condition is necessary but not sufficient. Combining (A.39) and (A.36) will result in a sufficient and necessary condition for CSC of a 2 output system. It is rather difficult to describe what (A.39) implies geometrically, even if the polar coordinates are used. To perform this geometrical analysis, let the gain matrices be represented in the polar coordinate by:

$$G = \begin{pmatrix} l_1 \cos(\alpha_1) & l_2 \cos(\alpha_2) & l_3 \cos(\alpha_3) \\ l_1 \sin(\alpha_1) & l_2 \sin(\alpha_2) & l_3 \sin(\alpha_3) \end{pmatrix}$$

$$\hat{G} = \begin{pmatrix} \eta_1 l_1 \cos(\alpha_1 + \delta_1) & \eta_2 l_2 \cos(\alpha_2 + \delta_2) & \eta_3 l_3 \cos(\alpha_3 + \delta_3) \\ \eta_1 l_1 \sin(\alpha_1 + \delta_1) & \eta_2 l_2 \sin(\alpha_2 + \delta_2) & \eta_3 l_3 \sin(\alpha_3 + \delta_3) \end{pmatrix}$$

Where η is the error in the length

δ is the error in the angle

After substituting the above expressions into (A.39) the result is still too complicated for geometrical interpretation. Perhaps a more elegant method of geometrical interpretation exists; however, this was not realized here.

Appendix 13: Effect of Scaling on Determinant

Assume that G is the un-scaled gain matrix, and G_S is the scaled gain matrix.

$$\hat{G}_S = D_L \hat{G} D_R \tag{A.40}$$

Then the determinant of the above expression is:

$$\det(\hat{G}_S) = \det(D_L)\det(\hat{G})\det(D_R) \tag{A.41}$$

The expected value of the determinant when the gain matrix is scaled is given by:

$$E\left(\det\left(\hat{G}\right)\right)=E\left(\left|\begin{pmatrix}\hat{G}_{1,1} & \cdots & \hat{G}_{1,n_x}\\ \vdots & \ddots & \vdots\\ \hat{G}_{n_y,1} & \cdots & \hat{G}_{n_y,n_x}\end{pmatrix}\right|\right)=\left|\begin{pmatrix}E\left(\hat{G}_{1,1}\right) & \cdots & E\left(\hat{G}_{1,n_x}\right)\\ \vdots & \ddots & \vdots\\ E\left(\hat{G}_{n_y,1}\right) & \cdots & E\left(\hat{G}_{n_y,n_x}\right)\end{pmatrix}\right| \tag{A.42}$$

Then the expected value of the determinant for the scaled gain matrix is:

$$\begin{aligned}E\left(\det\left(\hat{G}_S\right)\right)&=E\left(\det\left(D_L\right)\right)\left|\begin{pmatrix}E\left(\hat{G}_{1,1}\right) & \cdots & E\left(\hat{G}_{1,n_x}\right)\\ \vdots & \ddots & \vdots\\ E\left(\hat{G}_{n_y,1}\right) & \cdots & E\left(\hat{G}_{n_y,n_x}\right)\end{pmatrix}\right|E\left(\det\left(D_R\right)\right)\\ &=\det\left(D_L\right)\det\left(D_R\right)\left|\begin{pmatrix}E\left(\hat{G}_{1,1}\right) & \cdots & E\left(\hat{G}_{1,n_x}\right)\\ \vdots & \ddots & \vdots\\ E\left(\hat{G}_{n_y,1}\right) & \cdots & E\left(\hat{G}_{n_y,n_x}\right)\end{pmatrix}\right|\end{aligned} \tag{A.43}$$

Similarly, the variance of the determinant for the scaled gain matrix is:

$$\begin{aligned}\operatorname{var}\left(\det\left(\hat{G}_S\right)\right)&=\operatorname{var}\left(\det\left(D_L\right)\det\left(\hat{G}\right)\det\left(D_R\right)\right)\\ &=\left(\det\left(D_L\right)\det\left(D_R\right)\right)^2\operatorname{var}\left(\det\left(\hat{G}\right)\right)\end{aligned} \tag{A.44}$$

Then the probability of unstable control system for the scaled gain matrix is:

$$P\left(UCS_{\hat{G}_S}\right) \approx \begin{cases} P\left(Z > \dfrac{0 - \det(D_L)\det(D_R)E\left(\det\left(\hat{G}\right)\right)}{\sqrt{\left(\left(\det(D_L)\det(D_R)\right)^2 \operatorname{var}\left(\det\left(\hat{G}\right)\right)\right)}} = \dfrac{E\left(\det\left(\hat{G}\right)\right)}{\sqrt{\operatorname{var}\left(\det\left(\hat{G}\right)\right)}} \right), \\ if \ \det\left(\hat{G}\right) < 0 \\ P\left(Z < \dfrac{0 - \det(D_L)\det(D_R)E\left(\det\left(\hat{G}\right)\right)}{\sqrt{\left(\left(\det(D_L)\det(D_R)\right)^2 \operatorname{var}\left(\det\left(\hat{G}\right)\right)\right)}} = \dfrac{E\left(\det\left(\hat{G}\right)\right)}{\sqrt{\operatorname{var}\left(\det\left(\hat{G}\right)\right)}} \right), \\ if \ \det\left(\hat{G}\right) > 0 \\ \qquad \text{The matrix is rank deficient, } if \ \det\left(\hat{G}\right) = 0 \end{cases}$$

(A.45)

Since the probability of unstability estimated using equation (A.45) and P(USC) in Appendix 5 are the same, then scaling has no effect on P(USC). This result suggests that scaling may assist in the parameter estimation phase of the problem, by making the optimization problem less ill-conditioned (as is the case in RMPCT, MacArthur 1996); however, it will not have any effect on improving the quality of the model in terms of CSC.

Appendix 14: Effect of fixing the reduce system on the full system

Let G be a $\mathbb{R}^{n\times n}$ matrix. Then the determinant of G can be estimated using its minor's along any jth column:

$$\det(G) = g_{1,j}(-1)^{1+j}\det\left(G_{r,1,j}\right) + \cdots + g_{i,j}(-1)^{i+j}\det\left(G_{r,i,j}\right) + \cdots + g_{n,j}(-1)^{n+j}\det\left(G_{r,n,j}\right)$$

where $g_{i,j}$ is the i, jth element of matrix G

$G_{r,i,j}$ is the reduce matrix and is equivalent to the minor of entry $g_{i,j}$ of matrix G

Let $\det(G) < 0$ and an estimate of the full gain matrix be $\hat{G}$ (where $\det(\hat{G}) > 0$). This would suggest that the estimated model would result in an UCS for the full system. Now consider the reduce system $G_{r,i,j}$ and let $\det(G_{r,i,j}) < 0$ and based on the $\hat{G}$ let $\det(\hat{G}_{r,i,j}) > 0$. Hence the estimated model will result in an UCS for the reduce system as well. Fixing this reduce system, such that its determinant has the correct sign will result in $\hat{\hat{G}}_{r,i,j}$ (where $\det\left(\hat{\hat{G}}_{r,i,j}\right) < 0$). Then an expansion of $\det\left(\hat{\hat{G}}\right)$ will result in:

$$\det\left(\hat{\hat{G}}\right) = \hat{\hat{g}}_{i,j}(-1)^{i+j}\det\left(\hat{\hat{G}}_{r,i,j}\right) + \sum_k \hat{\hat{g}}_{k,j}(-1)^{k+j}\det\left(\hat{\hat{G}}_{r,k,j}\right)$$
$$k \in \{1,2,\cdots,i-1,i+1,\cdots,n\}$$

Since $\det\left(\hat{\hat{G}}_{r,i,j}\right) < 0$ then $\det\left(\hat{\hat{G}}\right) < 0$ iff:

$$-\hat{\hat{g}}_{i,j}(-1)^{i+j}\det\left(\hat{\hat{G}}_{r,i,j}\right) < \sum_k \hat{\hat{g}}_{k,j}(-1)^{k+j}\det\left(\hat{\hat{G}}_{r,k,j}\right)$$

Alternatively, this condition can be rewritten as:

$$\begin{cases} \text{if } \hat{\hat{g}}_{i,j}(-1)^{i+j} > 0, \det\left(\hat{\hat{G}}_{r,i,j}\right) > \dfrac{\sum_k \hat{\hat{g}}_{k,j}(-1)^{k+j}\det\left(\hat{\hat{G}}_{r,k,j}\right)}{\hat{\hat{g}}_{i,j}(-1)^{i+j}} \\ \text{if } \hat{\hat{g}}_{i,j}(-1)^{i+j} < 0, \det\left(\hat{\hat{G}}_{r,i,j}\right) < \dfrac{\sum_k \hat{\hat{g}}_{k,j}(-1)^{k+j}\det\left(\hat{\hat{G}}_{r,k,j}\right)}{\hat{\hat{g}}_{i,j}(-1)^{i+j}} \end{cases}$$

The above expression suggests a condition where fixing the reduce system will fix the full system's determinant as well. It illustrates that this would happen if the effect of the minor (associated with this particular reduce system) is larger than the effect of all the other minors summed up. Similar style of expression can be derived for when more than one constraint on the input and output are active. The above condition will result in a SCS when $\det(G) < 0$ and $\det(G_{r,i,j}) < 0$ similar conditions can be derived for other combinations as well (i.e., $\det(G) < 0$ and $\det(G_{r,i,j}) > 0$).