Adaptive Input Reconstruction with Application to Model Refinement, State Estimation, and Adaptive Control

by

Anthony M. D'Amato

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Aerospace Engineering)
in The University of Michigan
2012

Doctoral Committee:

Professor Dennis S. Bernstein, Chair
Professor Pierre T. Kabamba
Professor Ilya V. Kolmanovsky
Professor Aaron J. Ridley

He that is taught only by himself has a fool for a master.

-Ben Jonson

It was the Law of the Sea, they said. Civilization ends at the waterline. Beyond that, we all enter the food chain, and not always right at the top.

-Hunter S. Thompson

© Anthony Mario D'Amato 2012

All Rights Reserved

To Mom and Dad

Special thanks to my advisor, Dennis Bernstein

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

ABSTRACT

Adaptive Input Reconstruction with Application to Adaptive Control, Model Refinement, and State Estimation

by

Anthony M. D'Amato

Chair: Dennis S. Bernstein

Input reconstruction is the process of using the output of a system to estimate its input. In some cases, input reconstruction can be accomplished by determining the output of the inverse of a model of the system whose input is the output of the original system. Inversion, however, requires an exact and fully known analytical model, and is limited by instabilities arising from nonminimum-phase zeros.

The main contribution of this work is a novel technique for input reconstruction that does not require model inversion. This technique is based on a retrospective cost, which requires a limited number of Markov parameters. Retrospective cost input reconstruction (RCIR) does not require knowledge of nonminimum-phase zero locations or an analytical model of the system.

RCIR provides a technique that can be used for model refinement, state estimation, and adaptive control. In the model refinement application, data are used to refine or improve a model of a system. It is assumed that the difference between the model output and the data is due to an unmodeled subsystem whose interconnection with the modeled system is inaccessible, that is, the interconnection signals cannot be

measured and thus standard system identification techniques cannot be used. Using input reconstruction, these inaccessible signals can be estimated, and the inaccessible subsystem can be fitted. We demonstrate input reconstruction in a model refinement framework by identifying unknown physics in a space weather model and by estimating an unknown film growth in a lithium ion battery. The same technique can be used to obtain estimates of states that cannot be directly measured.

Adaptive control can be formulated as a model-refinement problem, where the unknown subsystem is the idealized controller that minimizes a measured performance variable. Minimal modeling input reconstruction for adaptive control is useful for applications where modeling information may be difficult to obtain. We demonstrate adaptive control of a seeker-guided missile with unknown aerodynamics.

CHAPTER I

Introduction

1.1 Model Refinement

Identification of linear time-invariant systems is a fundamental problem in systems theory, and available methods to solve this problem include frequency-domain techniques [2], time-series methods [3], and state-space algorithms [4, 5]. These techniques assume different model structures, in practice, the choice of model structure is guided by the intended use of the model.

Regardless of the desired model structure, the amount of available data and the quality of that data (that is, the level of noise that corrupts the data) directly impact the fidelity of the identified model. For data corrupted by stationary noise, we expect a fundamental tradeoff between the amount of available data and the noise level, where a weakness in quantity or quality can, at least to some extent, be offset by a strength in the other. In addition to the amount of data, identification methods may be sensitive to additional issues, such as the type of noise, a priori estimates of the system order and relative degree, and, in the MIMO case, coupling strength between the inputs and outputs. All of these issues must be assessed within the context of the computational burden of the competing algorithms.

A variation of the model identification problem is the case in which an initial model is available, and data is used to refine the initial model to obtain an improved

fit to the data, specifically, by identifying poorly modeled dynamics. This problem has been extensively studied within the context of finite element modeling [6, 7, 8], and has received some attention within the systems and control literature [9, 10, 11]. Figure 1.1 illustrates the model refinement problem. Part of the physical

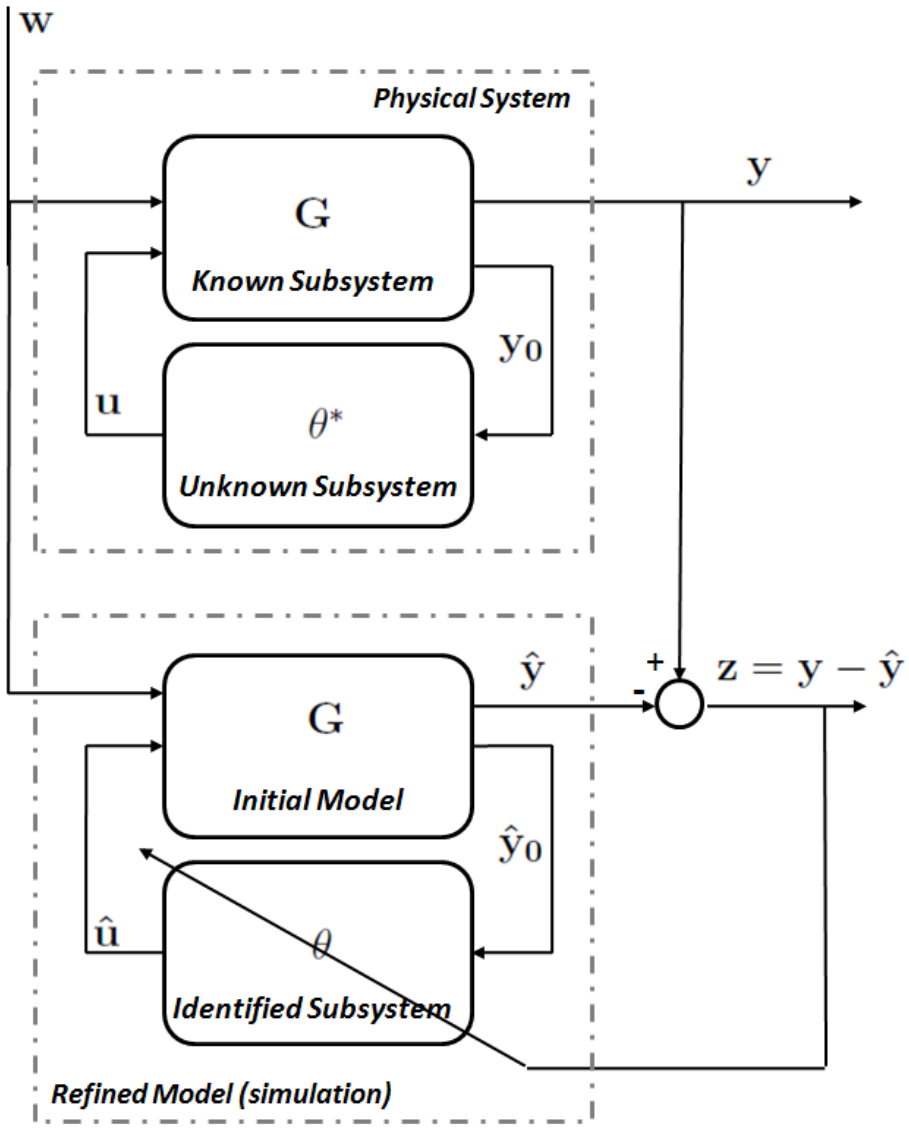


Figure 1.1: Model refinement architecture. The goal is to update θ based on the residual between the truth system and the model. The resulting initial model, in closed loop with θ, better approximates the physical system.

system is known, but there exists an unknown subsystem which we wish to identify. Note that traditional system identification techniques are not applicable, since u is an inaccessible signal within the system. To solve this problem, we propose to reconstruct u using adaptive feedback input reconstruction techniques, which then allow an estimate of the unknown subsystem to be identified.

1.2 State Estimation

The classical Kalman filter is the optimal state estimator for linear systems under process and sensor noise with zero mean and finite second moments. Implementation of the optimal estimator under these idealized conditions depends on knowledge of the linear dynamics, noise covariances, and inputs. When these assumptions are not satisfied, the accuracy of the Kalman filter can be severely degraded. Consequently, a problem of longstanding interest is to develop estimators that are robust to these uncertainties [12, 13, 14, 15]. A more proactive approach is to implement an adaptive state estimator, where the goal is to identify unknown inputs during system operation and use this information to tune the estimator on-line [13, 14]. Figure 1.2 illustrates an adaptive state and input estimation architecture. The state and input estimation problem is really a subset of model refinement, where the states of the known system have the same physical meaning as the states of the initial model.

1.3 Adaptive Control

Feedback has a long history in control of dynamical systems, from steam engine governors to automotive cruise controls. The basic principle behind feedback control is to adjust a control signal based on what the output of the system is doing.

Three areas of interest in feedback control are stabilization, command following, and disturbance rejection. The goal in stabilization is to bound the state of a system

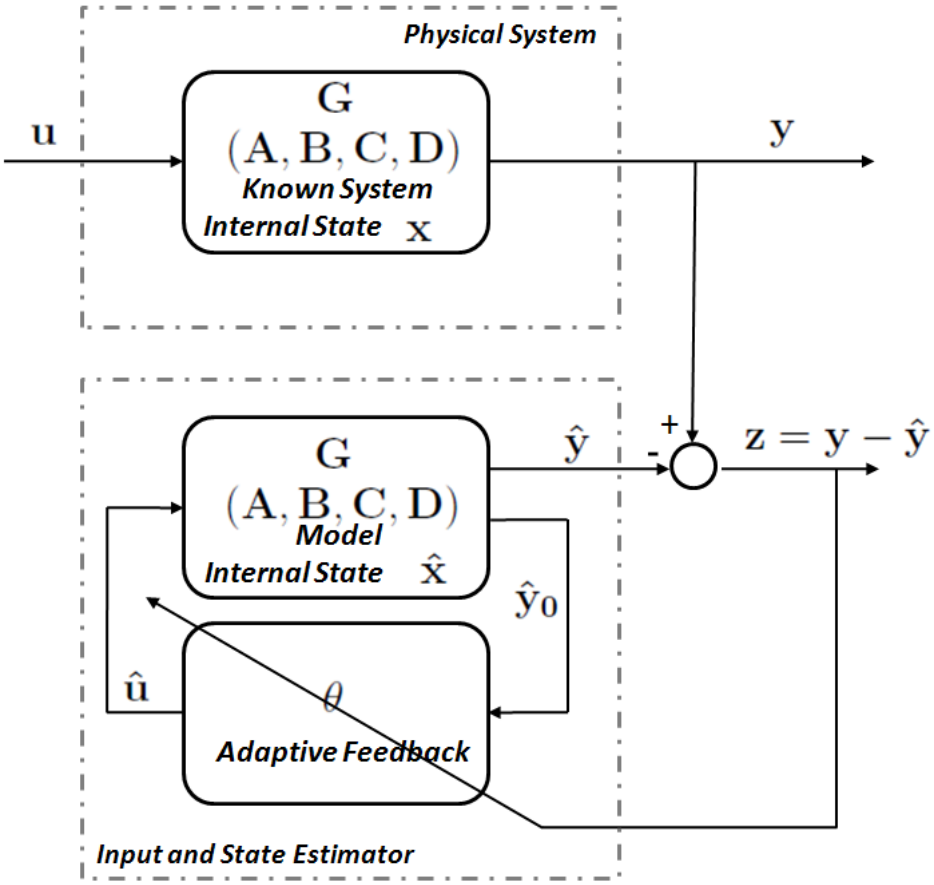


Figure 1.2: State estimation architecture. The goal is to update θ based on the residual between the physical system and the model, a subset of model refinement. If the states of the truth model have the same physical meaning as the estimator states, then the model refinement architecture results in a state estimator for systems with unknown harmonic inputs.

and drive its response to zero. In command following, the goal is to force the response of a system to follow a trajectory determined by the user. In disturbance rejection, the goal is to drive the response of the plant to a desired trajectory in the presence of an unknown exogenous signal. Furthermore, these objectives can be combined, for example, command following in the presence of a disturbance.

In adaptive control, the coefficients of a feedback controller are updated based on

an update law and a measured performance variable. The robustness of an adaptive controller is linked to the required modeling information needed by the update law. Adaptive controllers are intrinsically robust to plant information that is not required for the adaptation law.

Figure 1.3 illustrates an adaptive control architecture, formulated like a model refinement problem. The notation is modified to highlight the fact that there is no simulated system, only a real system. We wish to minimize the residual between the real system, which we wish to control, and the ideal system, which is a fiction. The ideal system is simply a copy of the real system, but it is connected in feedback with an idealized controller, which results in the performance $z^* = 0$.

1.4 Input Reconstruction

The common problem in the model refinement, state estimation and adaptive control architectures presented in the previous sections, is the inaccessibility of an unknown input. Therefore, the common solution to these problems is estimation or reconstruction of the unknown input. But what does it mean to reconstruct an input?

Input reconstruction is the process of estimating an unknown input given the output and modeling information from the system. Although input reconstruction is gaining interest among researchers [16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27], this topic is generally overshadowed by other areas of control theory, such as system identification, state estimation and control. We argue however, that these areas are fundamentally linked. In traditional applications of identification, estimation and control, the input is generally assumed to be known, therefore input estimation is not required. But what happens when the input is unknown, as in the model refinement frameworks? In this case, traditional system identification techniques fail, state estimation error now reflects the lack of knowledge of the system inputs, and it is difficult to implement a feedback control without knowledge of disturbance spectra.

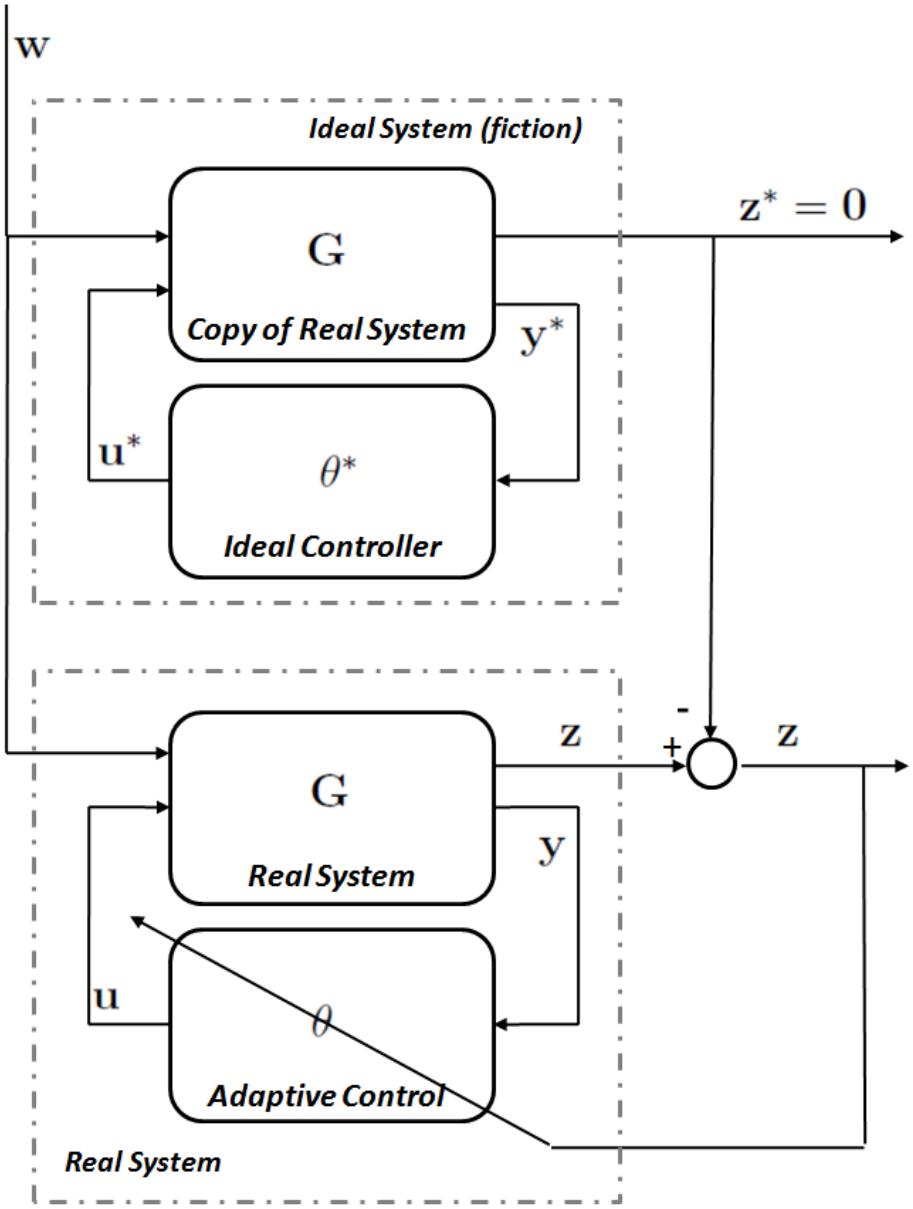


Figure 1.3: Adaptive control architecture. The goal is to update θ based on the residual between the real system and the ideal system. Adaptive control differs from model refinement and state estimation in that the ideal system is a fiction with idealized performance.

But how do we go about estimating an input, and what information about the system do we need to know? In its simplest form, input reconstruction can be ac-

complished, given a model, by taking its inverse and using the system output as the input to the inverse model. Figure 1.4 demonstrates this concept, where we assume that G is invertible and exactly proper ($D \neq 0$). However, this does not guarantee that $u - \hat{u}$ will becomes small as more data is gathered.

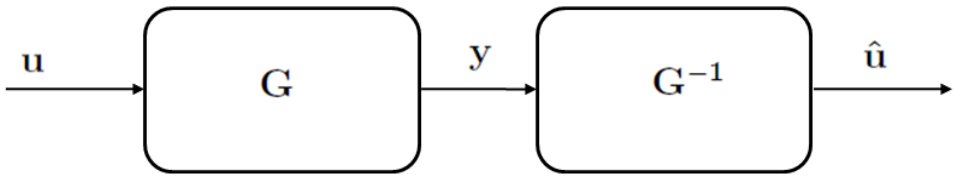


Figure 1.4: Assuming that the model G is invertible and exactly proper, the output of the system, y, can be used as the input to G^{-1}.

In fact, consider the problem of estimating the input, $u(k)$, and the state, $x(k)$, for the system

$$x(k+1) = Ax(k) + Bu(k), \tag{1.1}$$

$$y(k) = Cx(k) + Du(k), \tag{1.2}$$

$$u(k) = D^{-1}[y(k) - Cx(k)], \tag{1.3}$$

where D is nonsingular and (1.3) is obtained by solving (1.2) for $u(k)$. Note that the state $x(k)$ is unknown, therefore (1.3) cannot to be used to obtain $u(k)$ directly.

Using the inverse system approach, we build the estimator system

$$\hat{x}(k+1) = (A - BD^{-1}C)\hat{x}(k) + BD^{-1}y(k), \tag{1.4}$$

$$\hat{y}(k) = C\hat{x}(k) + D\hat{u}(k), \tag{1.5}$$

$$\hat{u}(k) = D^{-1}(y(k) - C\hat{x}(k)). \tag{1.6}$$

Subtracting (1.4) from (1.1), and (1.6) from (1.3), yields the state and input estima-

tion errors

$$e(k+1) = (A - BD^{-1}C)e(k), \tag{1.7}$$

$$\tilde{u}(k) = D^{-1}e(k), \tag{1.8}$$

where $e(k) = x(k) - \hat{x}(k)$, and $\tilde{u}(k) = u(k) - \hat{u}(k)$.

Immediately, we recognize the eigenvalues of $A - BD^{-1}C$ are the transmission zeros of (A, B, C, D). Assuming that the initial state $x(0)$ is unknown, the errors (1.7) and (1.8) get smaller if all the open loop transmission zeros are within the unit circle. Likewise, if (A, B, C, D) is nonminimum-phase, then the errors become unbounded.

The estimator (1.4)–(1.6) is an asymptotic input reconstruction technique. This techqniue differs from other types of input reconstruction techniques, such as those found in [27, 22, 23, 26], in that the initial conditions, and inputs before the estimator is started are never recovered, (and no attempt is made to do so). As data is gathered, we hope to drive the estimated input closer to the true input. As a by-product, the state of the system will also be recovered.

Beyond the problem associated with nonminimum-phase zeros, most input reconstruction techniques require an exact analytical model of the system, however many interesting input reconstruction problems do not have analytical representations. Consider a computational fluid dynamics model used to predict space weather conditions. The goal is to use data gathered on orbit to estimate solar intensity, which drives physical process in the thermosphere and ionosphere. In this case, inversion of an analytical model is impossible; the model we have access to is only capable of producing an output given an input, much like a black box. In Figure 1.5 an adaptive feedback architecture for input reconstruction, where the estimated input $\hat{u}$ is the output of a model whose parameters are updated based on the residual between the

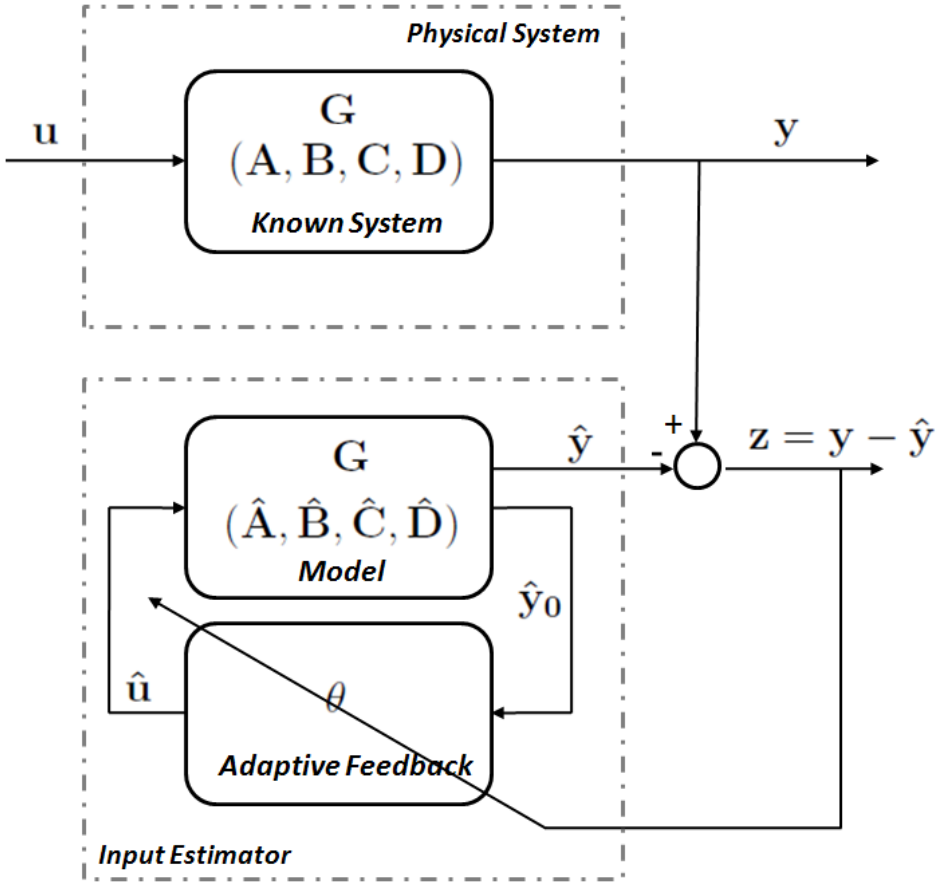


Figure 1.5: Adaptive input reconstruction architecture. The estimated input $\hat{u}$ is the output of the model θ, where the parameters of θ are updated by an adaptive update law using the residual between the data and the model output.

physical data and the model output.

In this dissertation, the original contribution is the development of an adaptive asymptotic input reconstruction method for nonminimum-phase systems with harmonic inputs. The adaptation law is based on a retrospective performance, that is, the input estimates are updated based on previously measured outputs.

We will demonstrate that for open-loop asymptotically stable plants with nonminimum -phase zeros, or for minimum-phase plants, adaptive asymptotic input recon-

struction can be accomplished using minimal modeling information. In most cases, a single Markov parameter is required. This enables us to reconstruct inputs even when a full analytical model is unavailable.

Finally, we use input reconstruction as a fundamental link between model refinement, (a subset of system identification), state estimation, and adaptive control. We formulate each of these problems as a special case of asymptotic input reconstruction, and both linear and nonlinear applications are demonstrated, such as identification of Wiener and Hammerstein systems, application to identification of unknown physics in space weather applications, Li-ion battery health monitoring, and adaptive control of a missile.

1.5 Dissertation Outline

In Chapter II, we introduce input reconstruction as a process where the inputs to a system are estimated using the measured system output. As described earlier in this chapter, one way to achieve this goal is to invert the system model and cascade delays to guarantee that the inverse is proper. The standing issue in input reconstruction lies in the inversion of nonminimum-phase systems, since the inverse model is unstable. We consider two methods for achieving input reconstruction despite the presence of nonminimum-phase zeros. First, we develop an open-loop partial inversion of the system model using a finite number of frequency points, where the partial inverse is a finite impulse response model and therefore is guaranteed to be asymptotically stable. Second, we examine a closed-loop approach that uses an infinite impulse response model. We demonstrate both methods on several illustrative examples.

In Chapter III, we present a two-step method for identifying SISO Hammerstein systems. First, using a persistent input with retrospective cost optimization, we estimate a parametric model of the linear system. Next, we pass a single harmonic signal through the system. We use l-delay input reconstruction with the parametric

model of the linear system to estimate the inaccessible intermediate signal. Using the estimate of the intermediate signal we estimate a nonparametric model of the static nonlinearity, which is assumed to be only piecewise continuous. This method is demonstrated on several numerical and experimental examples of increasing complexity. The algorithms and results presented in this chapter appear in [28].

In Chapter IV, we consider the problem of it data-based model refinement, where we assume the availability of an initial model, which can incorporate both physical laws and empirical observations. With this initial model as a starting point, our goal is to use additional measurements to refine the model. In particular, components of the model that are poorly modeled can be updated, thereby resulting in a higher fidelity model. We consider two special cases, namely, system emulation and subsystem identification. In the former case, the main system is assumed to be uncertain and we seek an estimate of the unknown subsystem that allows the overall model to approximate the true system. In this case, there is no expectation that the constructed subsystem model approximates the unknown subsystem. In the latter case, we assume that the main system is accurately modeled and we seek an estimate of the unknown subsystem that approximates the unknown subsystem. The algorithms and results presented in this chapter appear in [29].

In Chapter V, we apply the method developed in Chapter IV to a large-scale model based on first-principles physics, specifically, the the Global Ionosphere-Thermosphere Model (GITM). The goal is to estimate unknown physical processes in the ionosphere and thermosphere. Using GITM as the truth model, we demonstrate that measurements can be used to identify unknown physics. Specifically, we estimate static thermal conductivity parameters, as well as a dynamic cooling process. The algorithms and results presented in this chapter appear in [30].

In Chapter VI, we apply the method developed in Chapter IV to health management of Li-ion batteries. The health of a Li-ion battery depends on knowledge

of certain battery internal dynamics (e.g., lithium consumption and film growth at the solid-electrolyte interface) whose inputs and outputs are not directly measurable with noninvasive methods. This presents a problem of identification of inaccessible subsystems. To address this problem, we apply the retrospective-cost subsystem identification (RCSI) method. As a first step, this Chapter presents a simulation-based study that assumes as the truth model of the battery an electrochemistry-based battery charge/discharge model of Doyle, Fuller, and Newman, and later augmented with a battery-health model by Ramadass. First, this truth model is used to generate the data needed for the identification study. Next, the film-growth component of the battery-health model is assumed to be unknown, and the identification of this inaccessible subsystem is performed using RCSI. The results show that the subsystem identification method can identify the film growth quite accurately when the chemical reactions leading to film growth are consequential.

In Chapter VII, we present the dual to Chapter III, that is, we present a two-step method for identifying SISO Wiener systems. First, using a single harmonic input, we estimate a nonparametric model of the static nonlinearity, which is assumed to be only piecewise continuous. Second, using the identified nonparametric map, we use retrospective cost optimization to identify a parametric model of the linear dynamic system. This method is demonstrated on several examples of increasing complexity. The algorithms and results presented in this chapter appear in [31].

In Chapter VIII, we develop a method for obtaining state estimates for a possibly nonminimum-phase system in the presence of an unknown harmonic input. We construct a state estimator based on the system model, and then introduce an estimator input provided by an adaptive feedback model whose goal is to drive the estimated output to the measured output despite the presence of the unknown harmonic input. Using input reconstruction based on a retrospective surrogate cost, we reconstruct the unknown harmonic input. Using the reconstructed input we update the parameters

of the adaptive model using recursive least squares identification. We then extend the method to nonlinear systems. The performance of this method is compared with the Kalman filter for linear examples, as well as with the extended and unscented Kalman filters for nonlinear examples. The algorithms and results presented in this chapter appear in [32].

In Chapter IX, we develop a direct adaptive controller for discrete-time multi-input, multi-output, possibly nonminimum-phase systems with unknown nonminimum -phase zeros. This controller is based on a retrospective performance objective, where the controller is updated using either batch or recursive least squares. The adaptive controller requires limited modeling information about the system, specifically, Markov parameters from the control input to the performance variables. If the system is either asymptotically stable or minimum phase, then a single Markov parameter often suffices. If the system is unstable and nonminimum phase, then additional Markov parameters may be required. Stability of the algorithm is analyzed using a time-and-frequency-domain approach. Numerical examples are given to demonstrate disturbance rejection and command following problems, with unknown disturbance and command spectra. The algorithm and results presented in this chapter are submitted to the AIAA Journal of Guidance and Control.

In Chapter X, we apply extensions of retrospective cost adaptive control (RCAC) to a 2D missile model considered in prior works [33] as a benchmark test of adaptive control methods. The dynamics of the missile are highly nonlinear, and instantaneous linearizations are nonminimum phase due to nose sensing and tail actuation. The results that we present in this chapter show that the RCAC controller provides results that are comparable to a highly tuned autopilot based on aerodynamic modeling, whereas the RCAC controller does not use knowledge of the missile's aerodynamics. These results significantly improve the results obtained on the same problem using an earlier version of RCAC, presented at the 2010 GNC.

In Chapter IX, the coefficients of a strictly proper infinite-impulse-response controller are adaptively updated based on a measured performance variable. In Chapter XI, we restrict the controller to a proportional-integral-derivative (PID) structure, which we use for setpoint tracking. The adaptive law updates the coefficients of a PID controller based on the tracking error $z(k)$, reducing the tuning requirement of a typical PID setup. Unlike traditional PID tuning frameworks, this adaptive method requires limited modeling information; in most cases, a single Markov parameter is sufficient. Furthermore, if actuator saturation is present, the effects of integrator windup are mitigated, eliminating the need for anti-windup schemes. We demonstrate the method on several example of increasing complexity, including cases in which windup occurs in traditional PID setups, and for multi-input multi-output problems.

Finally, in Chapter XII, we discuss conclusions and future work in input reconstruction, model refinement, state estimation with unknown inputs, and adaptive control.

CHAPTER II

Adaptive Forward-Propagating Input Reconstruction for Nonminimum Phase Systems

2.1 Introduction

Unlike state estimation, where the goal is to use measured outputs to estimate unknown internal states, the goal of input reconstruction is to use measured outputs to estimate unknown inputs. Although not as well known as the state-estimation problem, input reconstruction has been studied for several decades, and interest continues up to the present time [27, 22, 26, 23, 20, 34, 21, 24, 25, 16, 17, 18, 19].

Early research focused on the problem of reconstructing the input given knowledge of the initial state of the system, while more recent techniques have focused on the problem of input reconstruction when the initial state is unknown. The latter problem is more challenging when zeros are present in the system since, for a suitable initial condition and input sequence, the output can be identically zero, thus making it impossible to unambiguously reconstruct the input.

When the initial condition is unknown and zeros are present in the plant, it is, however, possible to generically reconstruct the input asymptotically [22, 26, 23]. The simplest case occurs when the system is minimum phase, that is, all transmission zeros

are stable. In this case, the input reconstruction error decays with time [27]. The case of transmission zeros on the unit circle is intractable since the input reconstruction error is persistent. Finally, the case of nonminimum-phase transmission zeros is the most interesting, since the error decays as the reverse system propagates forward, that is, the input-reconstruction estimates are propagated backwards in time [22, 26, 23]. In practice, this means that the ability to reconstruct inputs to a nonminimum-phase system entails a delay in obtaining the input estimates.

In this chapter we propose a new approach to input reconstruction that is based entirely on forward propagation of the input estimate. This approach is especially suitable for harmonic inputs, that is, inputs that consist of sinusoids of different frequencies, and is applicable to plants with arbitrary zeros. Inversion techniques for nonminimum-phase systems are used in [35, 36] for tracking and iterative learning control.

We present two algorithms for forward-propagating input reconstruction. In open-loop forward-propagating input reconstruction, we construct a finite-impulse-response (FIR) transfer function that approximates the left inverse of the plant by minimizing the fit error at the frequencies that are present in the input signal. Although this method is applicable even for nonminimum-phase systems, the drawback is that the input spectrum must be known.

For the case in which the input spectrum is uncertain, we consider a closed-loop forward-propagating input-reconstruction technique, where the error is used to adapt the FIR approximate inverse based on the output residual. The adaptation algorithm uses a surrogate performance variable and a retrospective cost function. This technique is used in retrospective cost adaptive control (RCAC) [37, 38] as well as retrospective cost model refinement [39, 40, 41]. This chapter thus shows that, in addition to being of interest for its own sake, the adaptive forward-propagating input-reconstruction has implications beyond the problem of input reconstruction.

In this chapter we first provide a problem formulation for the forward-propagating input reconstruction problem. Next, we develop an open-loop method for input reconstruction in the presence of nonminimum-phase zeros using an asymptotically stable, partial inverse. We demonstrate the open-loop method on several examples. In Section 2.5 we introduce an adaptive closed-loop method using residual-based optimization to determine a partial inverse of the system. We then demonstrate the method on several illustrative examples.

2.2 Problem Formulation

Consider the MIMO discrete-time system

$$x(k+1) = Ax(k) + Bu(k), \tag{2.1}$$

$$y(k) = Cx(k), \tag{2.2}$$

where $A \in \mathbb{R}^{n\times n}$ is asymptotically stable, $B \in \mathbb{R}^{n\times m}$, and $C \in \mathbb{R}^{p\times n}$, $x(k) \in \mathbb{R}^n$, $y(k) \in \mathbb{R}^p$, and $u(k) \in \mathbb{R}^m$. Next, we define

$$G(\mathbf{q}) \stackrel{\triangle}{=} C(\mathbf{q}I - A)^{-1}B \in \mathbb{R}^{p\times m}(\mathbf{q}), \tag{2.3}$$

where $\mathbf{q}$ is the forward shift operator and $y(k) = G(\mathbf{q})u(k)$. We assume that $G(\mathbf{q})$ has full normal column rank, which implies that $m = \text{rank } B \leq \text{rank } C = p$.

We assume that $u(k)$ is a harmonic signal with frequencies in the set $\mathbf{\Omega} \stackrel{\triangle}{=} \{\Omega_1, \ldots, \Omega_h\}$, where $0 < \Omega_i < \pi$ rad/sample for all $i = 1, \ldots, h$. Next, let $F(\mathbf{q}) \in \mathbb{R}^{m\times p}(\mathbf{q})$ and define the signals

$$\hat{u}(k) \stackrel{\triangle}{=} F(\mathbf{q})y(k), \tag{2.4}$$

$$\hat{y}(k) \stackrel{\triangle}{=} G(\mathbf{q})\hat{u}(k), \tag{2.5}$$

where $\hat{u}(k) \in \mathbb{R}^m$, $\hat{y}(k) \in \mathbb{R}^p$. Furthermore,

$$z(k) \triangleq \hat{y}(k) - y(k), \tag{2.6}$$

$$\tilde{u}(k) \triangleq \hat{u}(k) - u(k). \tag{2.7}$$

The goal is to determine $F(\mathbf{q})$ such that $\tilde{u}(k)$ is small. Note that $\tilde{u}(k)$ is unknown, since $u(k)$ is unknown.

2.3 Open-Loop Forward-Propagating Input Reconstruction

Figure 2.1 shows the open-loop architecture in which $F(\mathbf{q})$ is cascaded with $G(\mathbf{q})$ to give estimates $\hat{u}(k)$ of $u(k)$. To determine $F(\mathbf{q})$ we consider the strictly proper

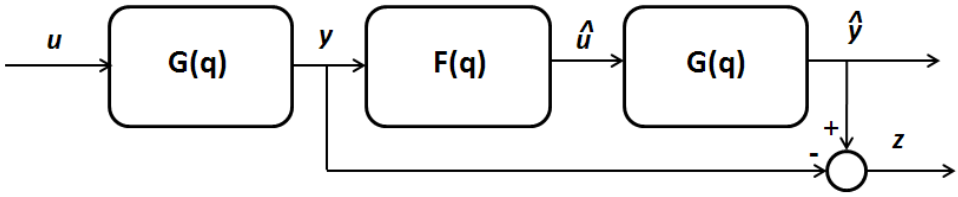


Figure 2.1: Open-loop architecture for input reconstruction. In this setup, the performance z, is not used to improve the estimate $\hat{u}$ of u.

finite-impulse-response (FIR) system

$$F(\mathbf{q}) = \sum_{i=1}^{\ell} \mathbf{q}^{-i} \beta_i \in \mathbb{R}^{m \times p}, \tag{2.8}$$

where $\beta_i \in \mathbb{R}^{m \times p}$, for all $i = 1, \ldots, \ell$. The goal is to determine $\beta_1, \ldots, \beta_\ell$ such that

$$J(\mathbf{\Omega}) \triangleq \sqrt{\text{tr Re}[(\varepsilon^*(\mathbf{\Omega})\varepsilon(\mathbf{\Omega}))]}, \tag{2.9}$$

is minimized where

$$\varepsilon(\mathbf{\Omega}) = \left[\begin{array}{ccc} F(e^{\jmath\Omega_1}) - G^\dagger(e^{\jmath\Omega_1}) & \cdots & F(e^{\jmath\Omega_h}) - G^\dagger(e^{\jmath\Omega_h}) \end{array} \right], \tag{2.10}$$

$(\cdot)^*$ denotes the complex conjugate transpose, and

$$G^\dagger(\mathbf{q}) = [G^{\mathrm{T}}(\mathbf{q})G(\mathbf{q})]^{-1}G^{\mathrm{T}}(\mathbf{q}), \tag{2.11}$$

which is improper.

To determine $\beta_1, \ldots \beta_\ell$ we choose the frequency set $\bar{\mathbf{\Omega}} = \{\bar{\Omega}_1, \ldots, \bar{\Omega}_l\}$. If $\mathbf{\Omega}$ is known, then we can set $\bar{\mathbf{\Omega}} = \mathbf{\Omega}$. If, however, $\mathbf{\Omega}$ is unknown, then $\bar{\mathbf{\Omega}}$ can be chosen to approximate or cover $\mathbf{\Omega}$. We then minimize $J(\bar{\mathbf{\Omega}})$ with respect to $\beta_1, \ldots, \beta_\ell$. We thus have the linear relation

$$\Psi = \Lambda\Phi, \tag{2.12}$$

where $\Psi \in \mathbb{R}^{m\times jp}$, $\Lambda \in \mathbb{R}^{m\times lp}$, $\Phi \in \mathbb{R}^{lp\times lp}$,

$$\Psi \triangleq \left[\mathrm{Re}[G^\dagger(e^{\jmath\bar{\Omega}_1})]\ \mathrm{Im}[G^\dagger(e^{\jmath\bar{\Omega}_1})] \cdots\ \mathrm{Re}[G^\dagger(e^{\jmath\bar{\Omega}_l})]\ \mathrm{Im}[G^\dagger(e^{\jmath\bar{\Omega}_l})]\right], \tag{2.13}$$

$$\Lambda \triangleq \left[\begin{array}{ccc} \beta_1 & \cdots & \beta_\ell \end{array} \right], \tag{2.14}$$

and

$$\Phi \triangleq \left[\begin{array}{cccccc} \mathrm{Re}[e^{-\jmath\bar{\Omega}_1}] & \mathrm{Im}[e^{-\jmath\bar{\Omega}_1}] & \cdots & \mathrm{Re}[e^{-\jmath\bar{\Omega}_l}] & \mathrm{Im}[e^{-\jmath\Omega_l}] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathrm{Re}[e^{-\ell\jmath\bar{\Omega}_1}] & \mathrm{Im}[e^{-\ell\jmath\bar{\Omega}_1}] & \cdots & \mathrm{Re}[e^{-\ell\jmath\bar{\Omega}_l}] & \mathrm{Im}[e^{-\ell\jmath\bar{\Omega}_l}] \end{array} \right]. \tag{2.15}$$

The least-squares solution of (2.12) is given by

$$\Lambda_{\text{opt}} = \Psi\Phi^{\text{T}}(\Phi\Phi^{\text{T}})^{-1}, \tag{2.16}$$

which minimizes $J(\bar{\mathbf{\Omega}})$. Note that if $G(\mathbf{q})$ is nonminimum phase, then $G^{\dagger}(\mathbf{q})$ is unstable. However, the poles of $F(\mathbf{q})$ are located at the origin, and therefore $F(\mathbf{q})$ is asymptotically stable.

2.4 Open-loop examples

Consider the asymptotically stable, nonminimum-phase plant $G(\mathbf{q}) = \frac{(\mathbf{q}-0.4)(\mathbf{q}-1.5)}{(\mathbf{q}-0.5\pm 0.5j)(\mathbf{q}-0.7)}$. We demonstrate input reconstruction for an unknown three-tone signal.

In the first example, we assume that the selected fit frequencies $\bar{\mathbf{\Omega}}$ coincide with the input frequencies $\mathbf{\Omega}$. In the second example, we assume that $\mathbf{\Omega}$ is a subset of $\bar{\mathbf{\Omega}}$, and, finally, $\mathbf{\Omega}$ is not a subset of $\bar{\mathbf{\Omega}}$.

In each case the unknown input is $u(k) = 0.01\sin(\Omega_1 k)+0.63\sin(\Omega_2 k)+1.3\sin(\Omega_3 k)$, where $\Omega_1 = 0.5$, $\Omega_2 = 0.7$, and $\Omega_3 = 0.9$ rad/sample.

Example 2.4.1. (SISO NMP, $\mathbf{\Omega} = \bar{\mathbf{\Omega}}$) Figure 2.2 is the frequency response comparison of F and $G^{\dagger}$, where the red vertical lines denote the frequencies $\bar{\Omega}_i$. At the frequencies $\bar{\Omega}_i$ the fit error is small, namely, $\|\varepsilon(\mathbf{\Omega})\| = 9.66 \times 10^{-12}$. Figure 2.3 compares the input $u(k)$ and the estimated input $\hat{u}(k)$. The peak of the transient not shown is ± 10. Figure 2.4 shows the residual plots of $\tilde{u}(k)$ and $z(k)$.

Example 2.4.2. (SISO NMP, $\mathbf{\Omega}$ is a subset of $\bar{\mathbf{\Omega}}$) For this example we choose $\bar{\Omega}_1 = 0.2$, $\bar{\Omega}_2 = 0.5$, $\bar{\Omega}_3 = 0.6$, $\bar{\Omega}_4 = 0.7$, $\bar{\Omega}_5 = 0.9$, and $\bar{\Omega}_6 = 1.1$. Figure 2.5 is the frequency response comparison of F and $G^{\dagger}$, where the red vertical lines denote the frequencies $\bar{\mathbf{\Omega}}$. The fit error is $\|\varepsilon(\mathbf{\Omega})\| = 6.57 \times 10^{-4}$. Figure 2.6 compares the input $u(k)$ and the estimated input $\hat{u}(k)$. The peak of the transient not shown is $\pm 1 \times 10^4$. Figure 2.7 shows the residual plots of $\tilde{u}(k)$ and $\tilde{y}(k)$.

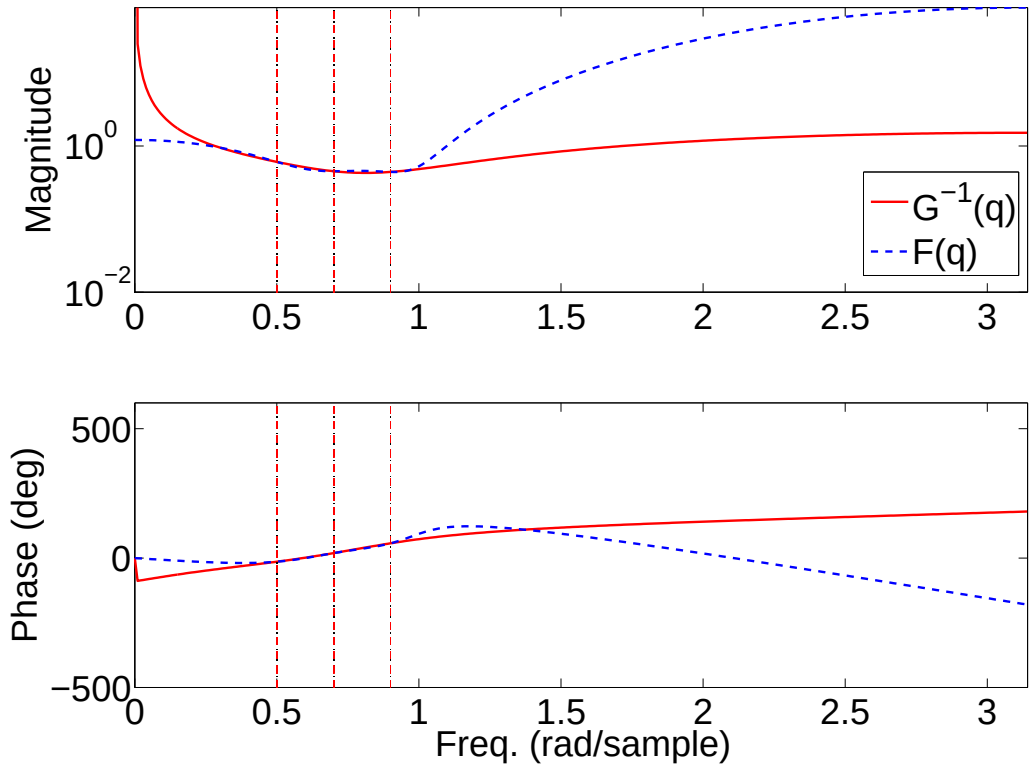


Figure 2.2: Frequency response comparison of F and $G^{\dagger}$, where the red vertical lines denote the frequencies $\bar{\Omega}_i$. In this example, $\bar{\mathbf{\Omega}} = \mathbf{\Omega}$.

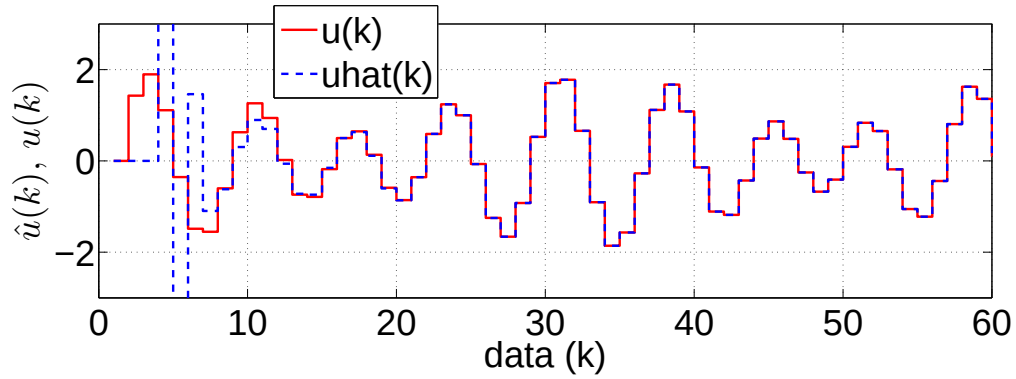


Figure 2.3: Comparison between $\hat{u}(k)$ and $u(k)$. After a transient whose peak excursion is approximately ± 10, $\tilde{u}(k)$ is small.

Example 2.4.3. (SISO NMP, $\mathbf{\Omega}$ is not a subset of $\bar{\mathbf{\Omega}}$) For this example we choose $\bar{\Omega}_1 = 0.2$, $\bar{\Omega}_2 = 0.57$, $\bar{\Omega}_3 = 0.6$, $\bar{\Omega}_4 = 0.76$, $\bar{\Omega}_5 = 0.95$, and $\bar{\Omega}_6 = 1.1$. Figure 2.8 is the

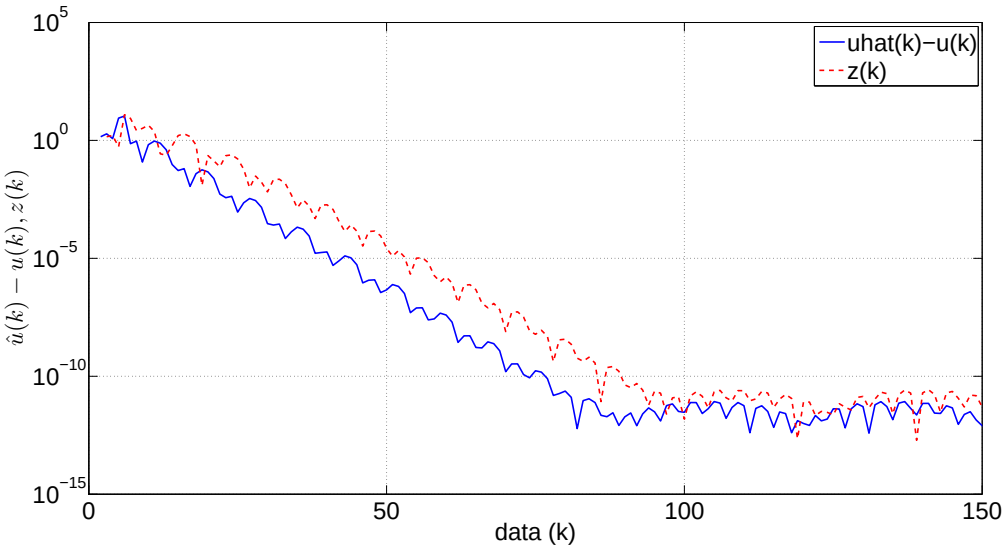


Figure 2.4: Residuals $\tilde{u}(k)$ and $z(k)$ show that both the input reconstruction error and output residual become small.

frequency response comparison of F and $G^{\dagger}$, where the red vertical lines denote the frequencies $\bar{\mathbf{\Omega}}$. The fit error is $\|\varepsilon(\mathbf{\Omega})\| = 0.14$. Next we choose $\bar{\Omega}_1 = 0.01$, $\bar{\Omega}_2 = 0.2$, $\bar{\Omega}_3 = 0.75$, $\bar{\Omega}_4 = 0.9$, $\bar{\Omega}_5 = 1.5$. Figure 2.9 compares the input $u(k)$ and the estimated input $\hat{u}(k)$. The peak of the transient not shown is approximately $\pm 5 \times 10^3$. Figure 2.10 shows the residual plots of $\tilde{u}(k)$ and $\tilde{y}(k)$.

2.5 Closed-Loop Forward-Propagating Input Reconstruction

We now consider the case where the output residual $z(k)$ is used to adaptively tune the parameters of $F(\mathbf{q})$ as in Figure 2.11. We start by letting $\hat{u}(k)$ be the output of the strictly proper adaptive feedback system of order ℓ, given by

$$\hat{u}(k) = \sum_{i=1}^{\ell} M_i(k)\hat{u}(k-i) + \sum_{i=1}^{\ell} N_i(k)z(k-i), \tag{2.17}$$

where, for all $i = 1, \ldots, \ell$, $M_i(k) \in \mathbb{R}^{m\times m}$ and $N_i(k) \in \mathbb{R}^{m\times p}$. Note that we now take $z(k)$ to be the input the input to $F(\mathbf{q})$. The goal is to update $M_i(k)$ and $N_i(k)$

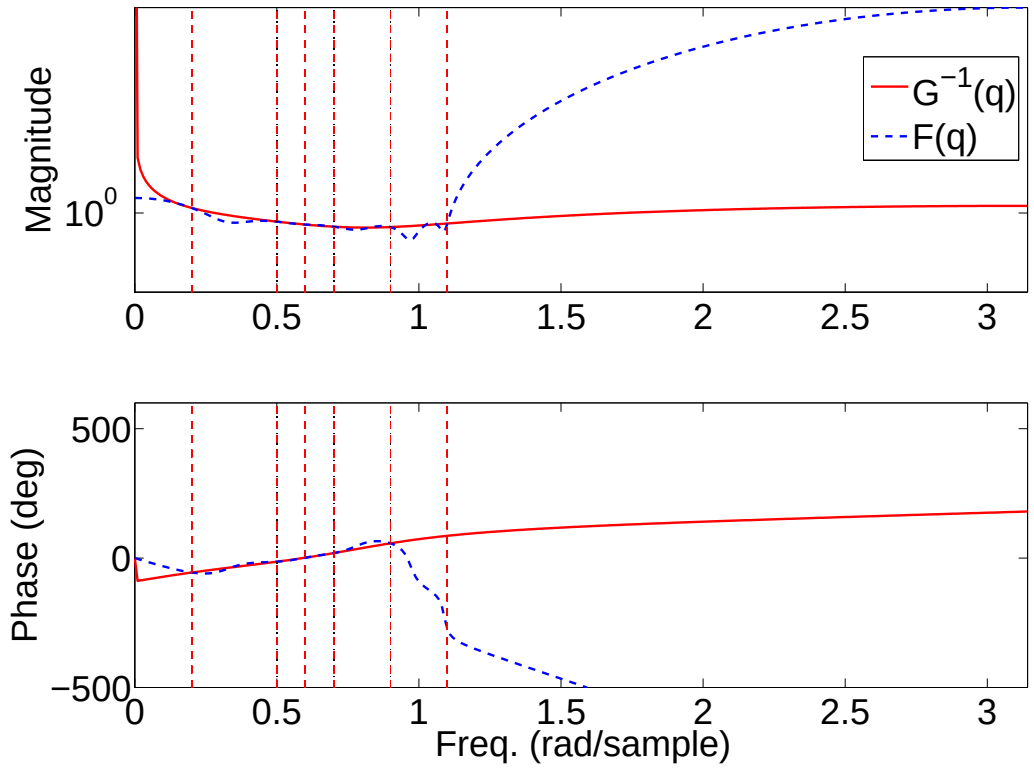


Figure 2.5: Frequency response comparison of F and $G^{\dagger}$, where the dashed vertical lines denote the frequencies $\bar{\mathbf{\Omega}}$, and the dotted vertical lines are the frequencies $\mathbf{\Omega}$.

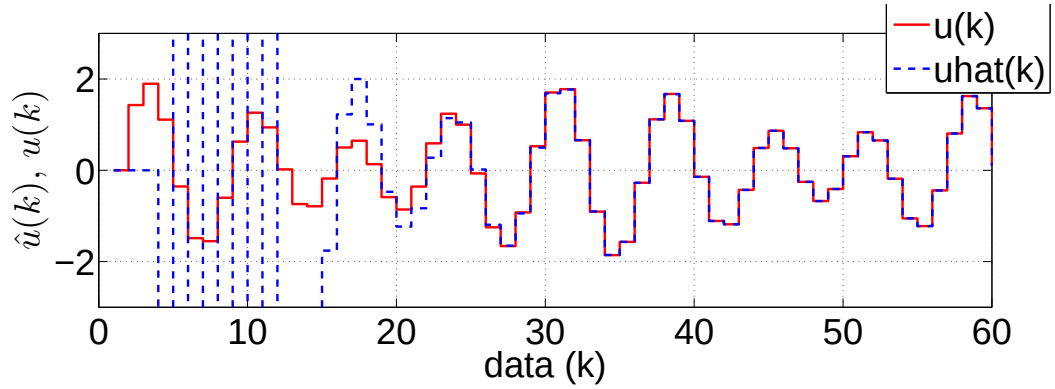


Figure 2.6: Comparison between $\hat{u}(k)$ and $u(k)$. After a transient whose peak excursion is approximately $\pm 1 \times 10^4$, $\tilde{u}(k)$ is small.

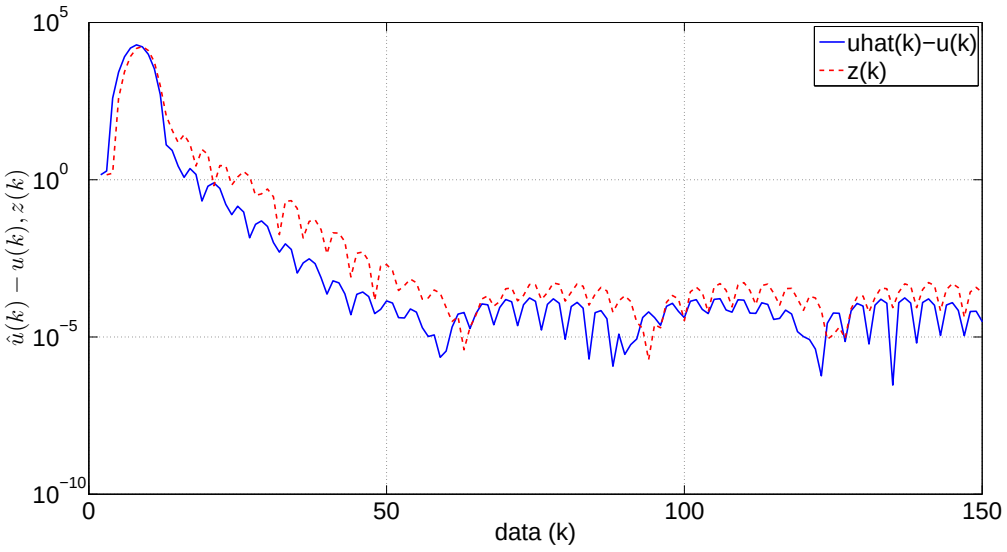


Figure 2.7: Residuals $\tilde{u}(k)$ and $z(k)$ show that both the input reconstruction error and output residual become small.

using the measured output error $z(k)$. Note that the coefficients of $F(\mathbf{q})$ are $M_i(k)$ and $N_i(k)$.

2.5.1 Input Reconstruction using a Retrospective Cost

For $i \geq 1$, define the Markov parameter H_i of (A, B, C) given by

$$H_i \triangleq CA^{i-1}B. \tag{2.18}$$

For example, $H_1 = CB$ and $H_2 = CAB$. Let r be a positive integer. Then, for all $k \geq r$,

$$\hat{x}(k) = A^r \hat{x}(k-r) + \sum_{i=1}^{r} A^{i-1} B \hat{u}(k-i), \tag{2.19}$$

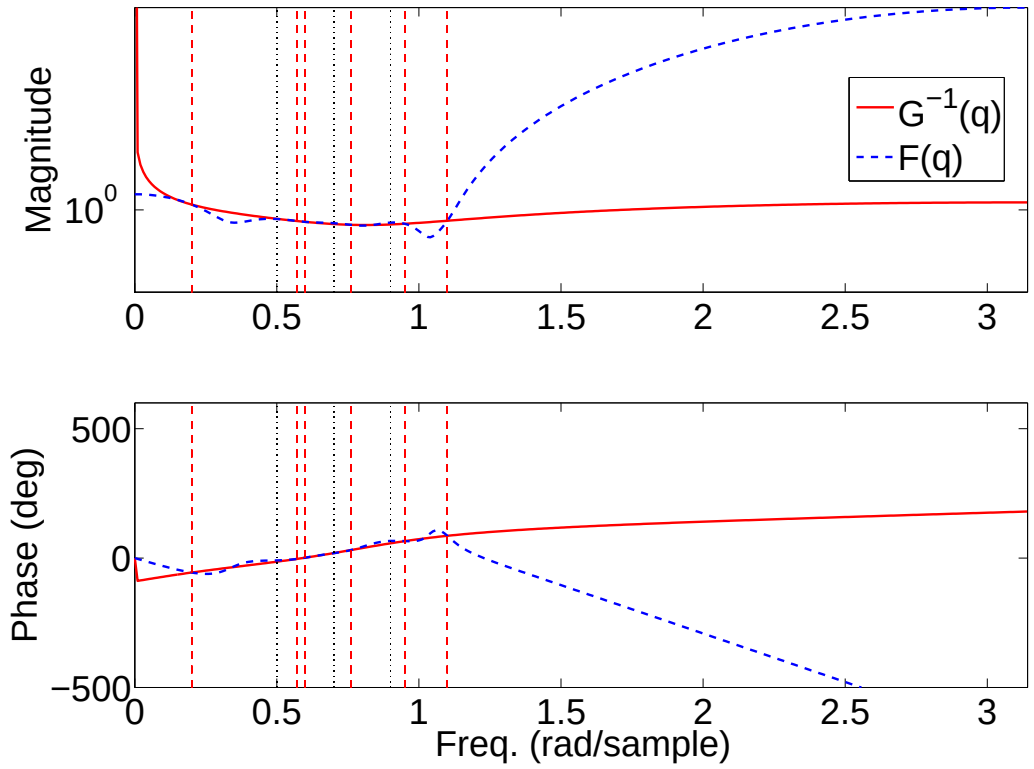


Figure 2.8: Frequency response comparison of F and $G^{\dagger}$, where the dashed vertical lines denote the frequencies $\bar{\mathbf{\Omega}}$, and the dotted vertical lines are the frequencies $\mathbf{\Omega}$.

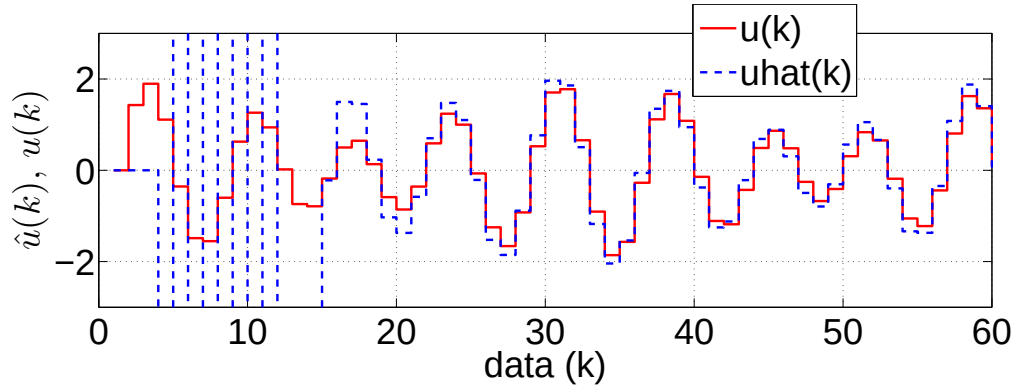


Figure 2.9: Comparison between $\hat{u}(k)$ and $u(k)$. After a transient whose peak excursion is approximately $\pm 5 \times 10^3$, $\tilde{u}(k)$ is small.

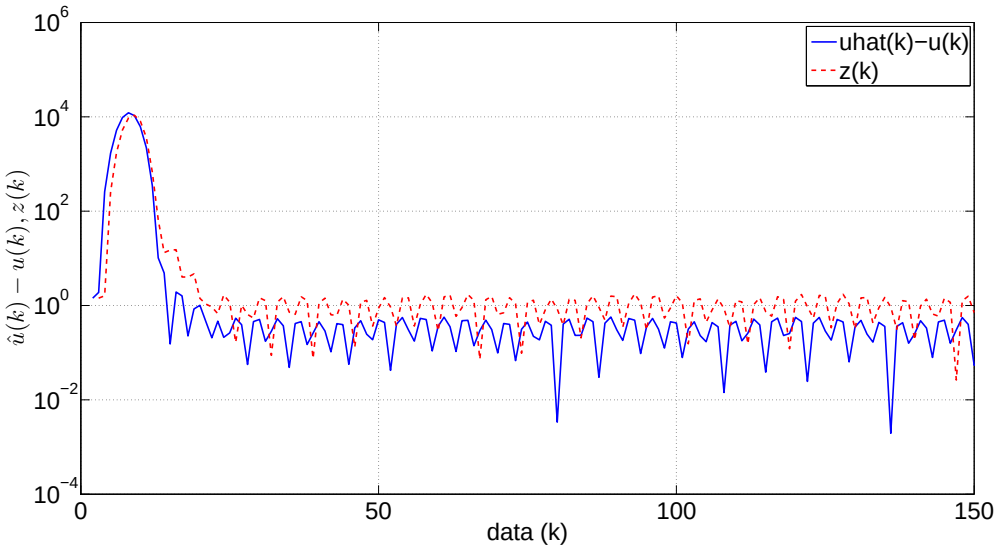


Figure 2.10: Residuals $\tilde{u}(k)$ and $z(k)$ show that both the input reconstruction error and output residual become small.

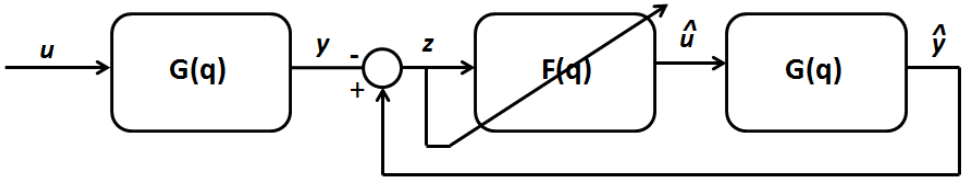


Figure 2.11: Adaptive architecture for input reconstruction.

and thus

$$z(k) = CA^r\hat{x}(k-r) - y(k) + \bar{H}\hat{\bar{U}}(k-1), \tag{2.20}$$

where

$$\bar{H} \triangleq \left[\begin{array}{ccc} H_1 & \cdots & H_r \end{array} \right] \in \mathbb{R}^{p\times rm}$$

and

$$\hat{\bar{U}}(k-1) \triangleq \begin{bmatrix} \hat{u}(k-1) \\ \vdots \\ \hat{u}(k-r) \end{bmatrix}.$$

Next, we rearrange the columns of $\bar{H}$ and the components of $\hat{\bar{U}}(k-1)$ and partition the resulting matrix and vector so that

$$\bar{H}\hat{\bar{U}}(k-1) = \mathcal{H}'\hat{U}'(k-1) + \mathcal{H}\hat{U}(k-1), \tag{2.21}$$

where $\mathcal{H}' \in \mathbb{R}^{p\times(rm-l_{\hat{U}})}$, $\mathcal{H} \in \mathbb{R}^{p\times l_{\hat{U}}}$, $\hat{U}'(k-1) \in \mathbb{R}^{rm-l_{\hat{U}}}$, and $\hat{U}(k-1) \in \mathbb{R}^{l_{\hat{U}}}$. Then, we can rewrite (2.20) as

$$z(k) = \mathcal{S}(k) + \mathcal{H}\hat{U}(k-1), \tag{2.22}$$

where

$$\mathcal{S}(k) \triangleq CA^r\hat{x}(k-r) - y(k) + \mathcal{H}'\hat{U}'(k-1). \tag{2.23}$$

For example, $\bar{H} = \begin{bmatrix} H_1 & H_2 & H_3 & H_4 & H_5 \end{bmatrix}$,

$$\mathcal{H}' = \begin{bmatrix} H_1 & H_2 & H_4 \end{bmatrix}, \quad \hat{U}'(k-1) = \begin{bmatrix} \hat{u}(k-1) \\ \hat{u}(k-2) \\ \hat{u}(k-4) \end{bmatrix},$$

and

$$\mathcal{H} = \begin{bmatrix} H_3 & H_5 \end{bmatrix}, \quad \hat{U}(k-1) = \begin{bmatrix} \hat{u}(k-3) \\ \hat{u}(k-5) \end{bmatrix}.$$

Note that the decomposition of $\bar{H}\hat{\bar{U}}(k-1)$ in (2.21) is not unique. Let s be a positive integer. Then for $i = 1, \ldots, s$, we replace $\mathcal{H}$, $\hat{U}(k-1)$, $\mathcal{H}'$, and $\hat{U}'(k-1)$ in (2.21) with $\mathcal{H}_j \in \mathbb{R}^{p \times l_{\hat{U}_j}}$, $\hat{U}_j(k-1) \in \mathbb{R}^{l_{\hat{U}_j}}$, $\mathcal{H}'_j \in \mathbb{R}^{p \times (rm - l_{\hat{U}_j})}$, and $\hat{U}'_j(k-1) \in \mathbb{R}^{rm - l_{\hat{U}_j}}$, respectively, such that (2.21) becomes

$$\bar{H}\hat{\bar{U}}(k-1) = \mathcal{H}'_j \hat{U}'_j(k-1) + \mathcal{H}_j \hat{U}_j(k-1). \tag{2.24}$$

Therefore, for $j = 1, \ldots, s$, we can rewrite (2.22) as

$$z(k) = \mathcal{S}_j(k) + \mathcal{H}_j \hat{U}_j(k-1), \tag{2.25}$$

where

$$\mathcal{S}_j(k) \stackrel{\triangle}{=} CA^r \hat{x}(k-r) + y(k) + \mathcal{H}'_j \hat{U}'_j(k-1). \tag{2.26}$$

Next, let $0 \leq k_1 \leq k_2 \leq \cdots \leq k_s$. Replacing k by $k - k_j$ in (2.25) yields

$$z(k-k_j) = \mathcal{S}_j(k-k_j) + \mathcal{H}_j \hat{U}_j(k-k_j-1). \tag{2.27}$$

Now, by stacking $z(k-k_1), \ldots, z(k-k_s)$, we define the *extended performance*

$$Z(k) \stackrel{\triangle}{=} \begin{bmatrix} z(k-k_1) \\ \vdots \\ z(k-k_s) \end{bmatrix} \in \mathbb{R}^{sp}. \tag{2.28}$$

Therefore,

$$Z(k) \triangleq \tilde{\mathcal{S}}(k) + \tilde{\mathcal{H}}\hat{\tilde{U}}(k-1), \tag{2.29}$$

where

$$\tilde{\mathcal{S}}(k) \triangleq \begin{bmatrix} \mathcal{S}_1(k-k_1) \\ \vdots \\ \mathcal{S}_s(k-k_s) \end{bmatrix} \in \mathbb{R}^{sp} \tag{2.30}$$

and $\hat{\tilde{U}}(k-1)$ has the form

$$\hat{\tilde{U}}(k-1) \triangleq \begin{bmatrix} \hat{u}(k-q_1) \\ \vdots \\ \hat{u}(k-q_g) \end{bmatrix} \in \mathbb{R}^{gm}, \tag{2.31}$$

where $k_1 \leq q_1 < q_2 < \cdots < q_g \leq k_s + r$. The vector $\hat{\tilde{U}}(k-1)$ is formed by stacking $\hat{U}_1(k-k_1-1), \ldots, \hat{U}_s(k-k_s-1)$ and removing copies of repeated components, and $\tilde{\mathcal{H}} \in \mathbb{R}^{sp \times gm}$ is constructed according to the structure of $\hat{\tilde{U}}(k-1)$. For example, with $s = 2$, $k_1 = 0$, and $k_2 = 2$, stacking $\hat{U}_1(k-1) = \begin{bmatrix} \hat{u}(k-1) \\ \hat{u}(k-2) \end{bmatrix}$ and $\hat{U}_2(k-3) = \hat{u}(k-3)$ results in

$$\hat{\tilde{U}}(k-1) = \begin{bmatrix} \hat{u}(k-1) \\ \hat{u}(k-2) \\ \hat{u}(k-3) \end{bmatrix}, \quad \tilde{\mathcal{H}} = \begin{bmatrix} H_1 & H_2 & H_3 \\ 0 & H_1 & H_2 \\ 0 & 0 & H_3 \end{bmatrix}. \tag{2.32}$$

Note that $\tilde{\mathcal{H}}$ consists of the entries of $\mathcal{H}_1, \ldots, \mathcal{H}_s$ arranged according to the structure of $\tilde{U}(k-1)$.

Next, we define the *retrospective performance*

$$\hat{z}(k-k_j) \triangleq \mathcal{S}_j(k-k_j) + \mathcal{H}_j U_j^*(k-k_j-1), \tag{2.33}$$

where the past input estimates $\hat{U}_j(k-k_j-1)$ in (2.27) are replaced by the retrospectively optimized input estimates $U_j^*(k-k_j-1)$, which are determined below. In analogy with (2.28), the *extended retrospective performance* is defined as

$$\hat{Z}(k) \triangleq \begin{bmatrix} \hat{z}(k-k_1) \\ \vdots \\ \hat{z}(k-k_s) \end{bmatrix} \in \mathbb{R}^{sp} \tag{2.34}$$

and thus is given by

$$\hat{Z}(k) = \tilde{\mathcal{S}}(k) + \tilde{\mathcal{H}}\tilde{U}^*(k-1), \tag{2.35}$$

where the components of $\tilde{U}^*(k-1) \in \mathbb{R}^{l_{\hat{U}}}$ are the components of $U_1^*(k-k_1-1), \ldots, U_s^*(k-k_s-1)$ ordered in the same way as the components of $\hat{\tilde{U}}(k-1)$. Subtracting (2.29) from (2.35) yields

$$\hat{Z}(k) = Z(k) - \tilde{\mathcal{H}}\hat{\tilde{U}}(k-1) + \tilde{\mathcal{H}}\tilde{U}^*(k-1). \tag{2.36}$$

Finally, we define the *retrospective cost function*

$$J(\tilde{U}^*(k-1), k) \triangleq \hat{Z}^{\mathrm{T}}(k) R_1(k) \hat{Z}(k) + \eta(k) \tilde{U}^{*\mathrm{T}}(k-1) R_2(k) \tilde{U}^*(k-1), \tag{2.37}$$

where $R_1(k) \in \mathbb{R}^{ps \times ps}$ is a positive-definite performance weighting, $R_2(k) \in \mathbb{R}^{gm \times gm}$ is a positive-definite input estimate weighting, and $\eta(k) \geq 0$ is a regularization weighting. The goal is to determine retrospective input estimates $\tilde{U}^*(k-1)$ that would have

provided better performance than the estimated inputs $\hat{U}(k-1)$ that were applied to the system. The retrospectively optimized estimated input values $\tilde{U}^*(k-1)$ are then used to update the controller. Substituting (2.36) into (2.37) yields

$$J(\tilde{U}^*(k-1),k) = \tilde{U}^{*\mathrm{T}}(k-1)\mathcal{A}(k)\tilde{U}^*(k-1) + \tilde{U}^{*\mathrm{T}}(k-1)\mathcal{B}^{\mathrm{T}}(k) + \mathcal{C}(k), \quad (2.38)$$

where

$$\mathcal{A}(k) \triangleq \tilde{\mathcal{H}}^{\mathrm{T}} R_1(k)\tilde{\mathcal{H}} + \eta(k)R_2(k), \quad (2.39)$$

$$\mathcal{B}(k) \triangleq 2\tilde{\mathcal{H}}^{\mathrm{T}} R_1(k)[Z(k) - \tilde{\mathcal{H}}\hat{\tilde{U}}(k-1)], \quad (2.40)$$

$$\mathcal{C}(k) \triangleq Z^{\mathrm{T}}(k)R_1(k)Z(k) - 2Z^{\mathrm{T}}(k)R_1(k)\tilde{\mathcal{H}}\hat{\tilde{U}}(k-1) + \hat{\tilde{U}}^{\mathrm{T}}(k-1)\tilde{\mathcal{H}}^{\mathrm{T}}R_1(k)\tilde{\mathcal{H}}\hat{\tilde{U}}(k-1). \quad (2.41)$$

If either $\tilde{\mathcal{H}}$ has full column rank or $\eta(k) > 0$, then $\mathcal{A}(k)$ is positive definite. In this case, $J(\tilde{U}^*(k-1),k)$ has the unique global minimizer

$$\tilde{U}^*(k-1) = -\frac{1}{2}\mathcal{A}^{-1}(k)\mathcal{B}(k), \quad (2.42)$$

which is the retrospectively optimized estimated inputs.

The regularization weighting $\eta(k)$ can be used to bound the retrospectively optimized estimated inputs $\tilde{U}^*(k-1)$ and thus indirectly bound the estimated inputs $\hat{\tilde{U}}(k)$. For example, $\eta(k)$ may be performance based

$$\eta(k) = \eta_0(k)||Z(k)||_2^2 \quad (2.43)$$

or error based

$$\eta(k) = \eta_0(k)||\tilde{U}^*(k-2) - \hat{\tilde{U}}(k-2)||_2^2, \quad (2.44)$$

where $\eta_0(k) \geq 0$. Alternatively, the retrospectively optimized inputs can be bounded directly by using a saturation function, where $\eta(k) \equiv 0$ in (2.39) and (2.42) is replaced by

$$\tilde{U}^*(k-1) \stackrel{\triangle}{=} \mathrm{sat}_{[a,b]}[-\frac{1}{2}\mathcal{A}^{-1}(k)\mathcal{B}(k)], \tag{2.45}$$

where $\mathrm{sat}_{[a,b]}(\zeta)$ is the component-wise saturation function defined for scalar arguments by

$$\mathrm{sat}_{[a,b]}(\zeta) \stackrel{\triangle}{=} \begin{cases} b, & \text{if } \zeta \geq b, \\ \zeta, & \text{if } a < \zeta < b, \\ a & \text{if } \zeta \leq a, \end{cases} \tag{2.46}$$

where $a < b$ are the component-wise saturation levels.

2.6 Adaptive Feedback Update

The reconstructed input (2.17) can be expressed as

$$\hat{u}(k) = \theta(k)\phi(k-1), \tag{2.47}$$

where

$$\theta(k) \stackrel{\triangle}{=} [M_1(k) \ \cdots \ M_\ell(k) \ N_1(k) \ \cdots \ N_\ell(k)] \in \mathbb{R}^{m \times \ell(m+p)} \tag{2.48}$$

and

$$\phi(k-1) \triangleq \begin{bmatrix} \hat{u}(k-1) \\ \vdots \\ \hat{u}(k-\ell) \\ y(k-1) \\ \vdots \\ y(k-\ell) \end{bmatrix} \in \mathbb{R}^{\ell(m+p)}. \tag{2.49}$$

Next, we define the recursive least squares (RLS) cost function

$$\begin{aligned} J_{\mathrm{R}}(\theta(k)) \triangleq & \sum_{i=q_g+1}^{k} \lambda^{k-i} \|\phi^{\mathrm{T}}(i-q_g-1)\theta^{\mathrm{T}}(k) - u^{*\mathrm{T}}(i-q_g)\|^2 \\ & + \lambda^k (\theta(k)-\theta(0)) P^{-1}(0) (\theta(k)-\theta(0))^{\mathrm{T}}, \end{aligned} \tag{2.50}$$

where $\|\cdot\|$ is the Euclidean norm and, for some $\varepsilon \in (0,1)$, $\lambda(k) \in (\varepsilon, 1]$ is the forgetting factor, and $P(0) \in \mathbb{R}^{\ell(m+p)\times\ell(m+p)}$ is positive definite. Minimizing (2.50) yields

$$\begin{aligned} \theta^{\mathrm{T}}(k) \triangleq & \, \theta^{\mathrm{T}}(k-1) + \beta(k) P(k-1)\phi(k-q_g-1) \\ & \cdot [\phi^{\mathrm{T}}(k-q_g-1)P(k-1)\phi(k-q_g-1) + \lambda(k)]^{-1} \\ & \cdot [\theta(k-1)\phi(k-q_g-1) - u^*(k-q_g)]^{\mathrm{T}}, \end{aligned} \tag{2.51}$$

where $\beta(k)$ is either 0 or 1. When $\beta(k) = 1$, the controller is allowed to adapt, whereas, when $\beta(k) = 0$, the adaptation is off. $P(k)$ is updated by

$$\begin{aligned} P(k) \triangleq & \, (1-\beta(k))P(k-1) + \beta(k)\lambda^{-1}(k)P(k-1) - \beta(k)\lambda^{-1}(k)P(k-1)\phi(k-q_g-1) \\ & \cdot [\phi^{\mathrm{T}}(k-q_g-1)P(k-1)\phi(k-q_g-1) + \lambda(k)]^{-1}\phi^{\mathrm{T}}(k-q_g-1)P(k-1). \end{aligned} \tag{2.52}$$

We initialize $P(0) = \gamma I$, where $\gamma > 0$. Furthermore, the updates (2.51) and (2.52) are based on the g^{th} component of $\tilde{U}^*(k-1)$. However any or all of the components of $\tilde{U}^*(k-1)$ may be used in the update of $\theta(k)$ and $P(k)$.

2.6.1 $\tilde{\mathcal{H}}$ as an Approximation of $G(\mathbf{q})$

In this section, we give frequency-domain conditions on $\tilde{\mathcal{H}}$ and $\hat{\tilde{U}}(k-1)$ such that $\hat{u} - u(k)$ will become small. Note that this analysis is the input reconstruction analog to the stability development in Chapter IX. Some of the assumptions used in Proposition II.1 are motivated by the more complete analysis in Chapter IX.

First we introduce $G_{\text{FIR}}(\mathbf{q})$, which is an FIR filter whose numerator coefficients are the Markov parameters of $G_{zu}(\mathbf{q})$ that comprise $\tilde{\mathcal{H}}$. The structure of $G_{\text{FIR}}(\mathbf{q})$ depends on the components of $\hat{\tilde{U}}(k-1)$ that are used to update $\theta(k)$ and the structure of $\tilde{\mathcal{H}}$. For example, in the SISO case assume that $\tilde{\mathcal{H}} = [H_3 \ H_2 \ H_1]^{\text{T}}$ and $g = q_g = 3$, which results in $\hat{\tilde{U}}(k-1) = \hat{u}(k-q_g)$. Then

$$G_{\text{FIR}}(\mathbf{q}) = \frac{H_1\mathbf{q}^2 + H_2\mathbf{q} + H_3}{\mathbf{q}^3}. \tag{2.53}$$

Note that $G_{\text{FIR}}(\mathbf{q})$ is only use an analysis tool, it is not required to implement the retrospective input reconstruction technique.

Furthermore, let the unknown signal $u(k)$ be a sinusoid whose frequency is Θ. For this analysis we assume that $\beta(k) = 1$, only when the state of the estimator system reaches a harmonic steady state, and $\beta(k) = 0$ otherwise. Specifically, we only update the controller coefficients and error covariance when the state of the estimator system reaches harmonic steady state and let

$$\nu = \beta(0) + \cdots + \beta(k), \tag{2.54}$$

where ν is the number of controller and error covariance updates. Note, in practice,

we update the controller coefficients at each step k.

Proposition II.1. *Let $\tilde{U}^*(k-1)$ be given by (2.42), and let $m = p = 1$. Next, assume that the frequency Θ does not coincide with any zeros of $G_{\rm FIR}$. Furthermore, assume that (2.5) reaches harmonic steady state for all ν, and assume that $\hat{u}(k) - u^*(k) \to 0$ as $k \to \infty$ and that $\eta(k)$ is chosen such that $\eta(k) \to 0$, as $k \to \infty$, which implies $\hat{z}(k) \to 0$, as $k \to \infty$, (see Fact. 9.5.2) then if*

$$\left| 1 - \frac{G(e^{\jmath\Theta})}{G_{\rm FIR}(e^{\jmath\Theta})} \right| < 1, \tag{2.55}$$

then $z(k) \to 0$ as $k \to \infty$.

In harmonic steady state we have

$$z_\nu = -G(e^{\jmath\Theta})u + G(e^{\jmath\Theta})u^*_\nu + G(e^{\jmath\Theta})g_\nu, \tag{2.56}$$

where z_ν, u^*_ν, g_ν are phasors, and $g_\nu \stackrel{\triangle}{=} u_\nu - \hat{u}_\nu$. In view of the assumption that $\hat{u}(k) - u^*(k) \to 0$ as $k \to \infty$, we assume that g_ν is negligible and omitted for simplicity.

Next, the retrospective cost in harmonic steady state is,

$$\hat{z}_\nu \stackrel{\triangle}{=} z_{\nu-1} - G_{\rm FIR}(e^{\jmath\Theta})u^*_{\nu-1} + G_{\rm FIR}(e^{\jmath\Theta})u^*_\nu, \tag{2.57}$$

$$\hat{z}_\nu = -G(e^{\jmath\Theta})u + G(e^{\jmath\Theta})u^*_{\nu-1} - G_{\rm FIR}(e^{\jmath\Theta})u^*_{\nu-1} + G_{\rm FIR}(e^{\jmath\Theta})u^*_\nu, \tag{2.58}$$

$$\hat{z}_\nu = [G(e^{\jmath\Theta}) - G_{\rm FIR}(e^{\jmath\Theta})]u^*_{\nu-1} + G_{\rm FIR}(e^{\jmath\Theta})u^*_\nu - G(e^{\jmath\Theta})u. \tag{2.59}$$

Solving (2.59) for u^*_ν yields,

$$u^*_\nu = G^{-1}_{\rm FIR}(e^{\jmath\Theta}) \left[\hat{z}_\nu + G(e^{\jmath\Theta})u - [G(e^{\jmath\Theta}) - G_{\rm FIR}(e^{\jmath\Theta})]u^*_{\nu-1} \right]. \tag{2.60}$$

Substituting (2.60) into (2.56) yields,

$$z_\nu = [1 - G(e^{\jmath\Theta})G_{\mathrm{FIR}}^{-1}(e^{\jmath\Theta})][-G(e^{\jmath\Theta})u + G(e^{\jmath\Theta})u_{\nu-1}^*] + G(e^{\jmath\Theta})G_{\mathrm{FIR}}^{-1}(e^{\jmath\Theta})\hat{z}_\nu. \quad (2.61)$$

Using this process we write z_ν in terms of u_0^* as

$$\begin{aligned} z_\nu &= [1 - G(e^{\jmath\Theta})G_{\mathrm{FIR}}^{-1}(e^{\jmath\Theta})]^\nu[-G(e^{\jmath\Theta})u + G(e^{\jmath\Theta})u_0^*] \\ &\quad + \sum_{i=0}^{\nu}[1 - G(e^{\jmath\Theta})G_{\mathrm{FIR}}^{-1}(e^{\jmath\Theta})]^i[G(e^{\jmath\Theta})G_{\mathrm{FIR}}^{-1}(e^{\jmath\Theta})]\hat{z}_{\nu-i}. \end{aligned} \quad (2.62)$$

It follows from (2.62) that

$$\begin{aligned} |z_\nu| &\leq \left|1 - G(e^{\jmath\Theta})G_{\mathrm{FIR}}^{-1}(e^{\jmath\Theta})\right|^\nu \left|-G(e^{\jmath\Theta})u + G(e^{\jmath\Theta})u_0^*\right| \\ &\quad + \left|\sum_{i=0}^{\nu}[1 - G(e^{\jmath\Theta})G_{\mathrm{FIR}}^{-1}(e^{\jmath\Theta})]^i[G(e^{\jmath\Theta})G_{\mathrm{FIR}}^{-1}(e^{\jmath\Theta})]\hat{z}_{\nu-i}\right|. \end{aligned} \quad (2.63)$$

Therefore, since $\left|1 - \frac{G(e^{\jmath\Theta})}{G_{\mathrm{FIR}}(e^{\jmath\Theta})}\right| < 1$, it follows that $\left|1 - \frac{G(e^{\jmath\Theta})}{G_{\mathrm{FIR}}(e^{\jmath\Theta})}\right|^\nu \to 0$ as $\nu \to \infty$, then $|z_\nu| \to 0$ as $\nu \to \infty$.

Condition (2.55) has a simple geometric interpretation, namely, $G_{\mathrm{FIR}}(e^{\jmath\Theta})$ must lie in a half plane that contains $G(e^{\jmath\Theta})$ and whose boundary is perpendicular to $|G(e^{\jmath\Theta})|$ and passes through $\frac{1}{2}|G(e^{\jmath\Theta})|$. Figure 2.12 illustrates the region of admissible $G_{\mathrm{FIR}}(e^{\jmath\Theta})$ for a given $|G(e^{\jmath\Theta})|$ and frequency.

From Proposition II.1, we note that if Θ coincides with a zero of G_{FIR}, then the system remains at its open-loop performance value. Furthermore, this analysis can be done for multiple frequencies.

The above analysis is based on the assumption that the state of the system reaches harmonic steady state after each period of adaptation. This assumption is an approximation invoked to facilitate the analysis. In fact, RCAC adapts at each step, and thus the state does not reach harmonic steady state. The examples in the next sec-

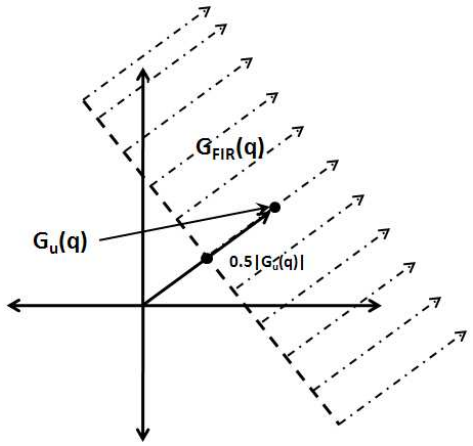


Figure 2.12: The dashed region on the complex plane illustrates the region of admissible $G_{\rm FIR}(e^{\jmath\Theta})$ for a given $|G(e^{\jmath\Theta})|$ and frequency Θ as determined by 2.55. The admissible region is a half plane.

tion show that this condition is sufficient but not necessary, and thus provides a conservative estimate of the allowable uncertainty that can be tolerated in the FIR approximation error.

2.7 Adaptive Examples

We now re-consider the same plant and unknown input as in Section 2.4. We now apply the adaptive algorithm in place of the open-loop input reconstruction technique. For all of the following examples we choose the RCAC parameters $\ell = 10$, $\eta_0 = 0.01$, and $P(0) = 1$.

Example 2.7.1. (SISO NMP, $\tilde{\mathcal{H}} = H_1$) In the first example we choose $\tilde{\mathcal{H}} = H_1$. Figure 2.13 compares the frequency response of $G(\mathbf{q})$ and $G_{\rm FIR}(\mathbf{q}) = \frac{H_1}{\mathbf{q}}$. Note from the phase comparison that the phase error at frequency Ω_1 is approaching 90 degrees. Figure 2.14 compares the effect of the large phase error, where a large transient is experienced before settling. Figure 2.15 compares the residual $\hat{u}(k) - u(k)$ and $z(k)$. We note that the performance does not reach a lower bound as in the open-loop case, but continues to decrease.

Example 2.7.2. (SISO NMP, $\tilde{\mathcal{H}} = [H_3\ H_2\ H_1]^{\rm T}$) In the first example we choose $\tilde{\mathcal{H}} =$

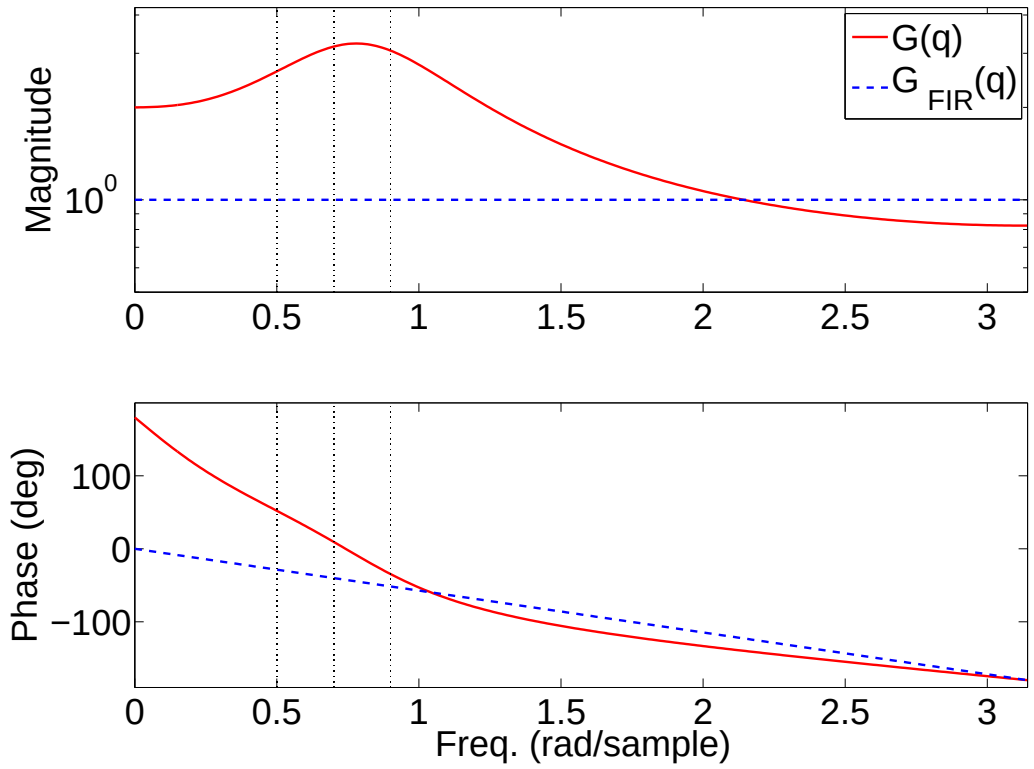


Figure 2.13: Frequency response comparison of G and G_{FIR}, where the red vertical lines denote the frequencies Ω_i.

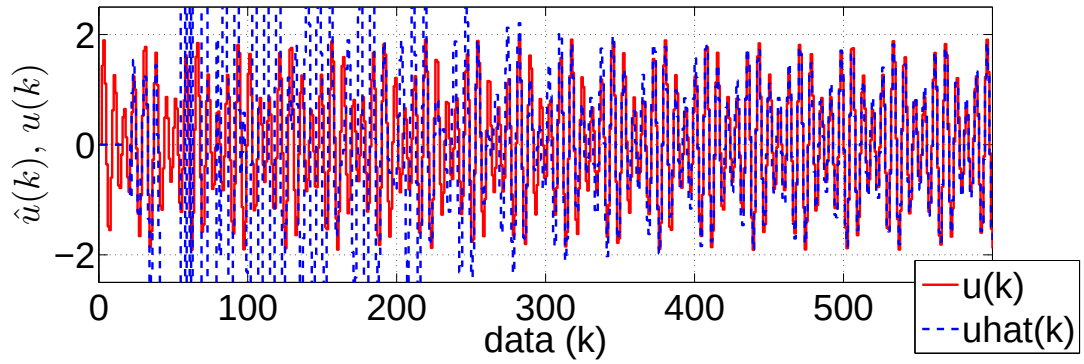


Figure 2.14: Comparison between $\hat{u}(k)$ and $u(k)$. After a transient whose peak excursion is approximately $\pm 4 \times 10^5$, $\tilde{u}(k)$ is small.

$[H_3 \ H_2 \ H_1]^{\mathrm{T}}$. Figure 2.16 compares the frequency response of $G(\mathbf{q})$ and $G_{\mathrm{FIR}}(\mathbf{q}) = \frac{H_1\mathbf{q}^2+H_2\mathbf{q}+H_3}{\mathbf{q}^3}$. Note from the phase comparison that the phase error is smaller than

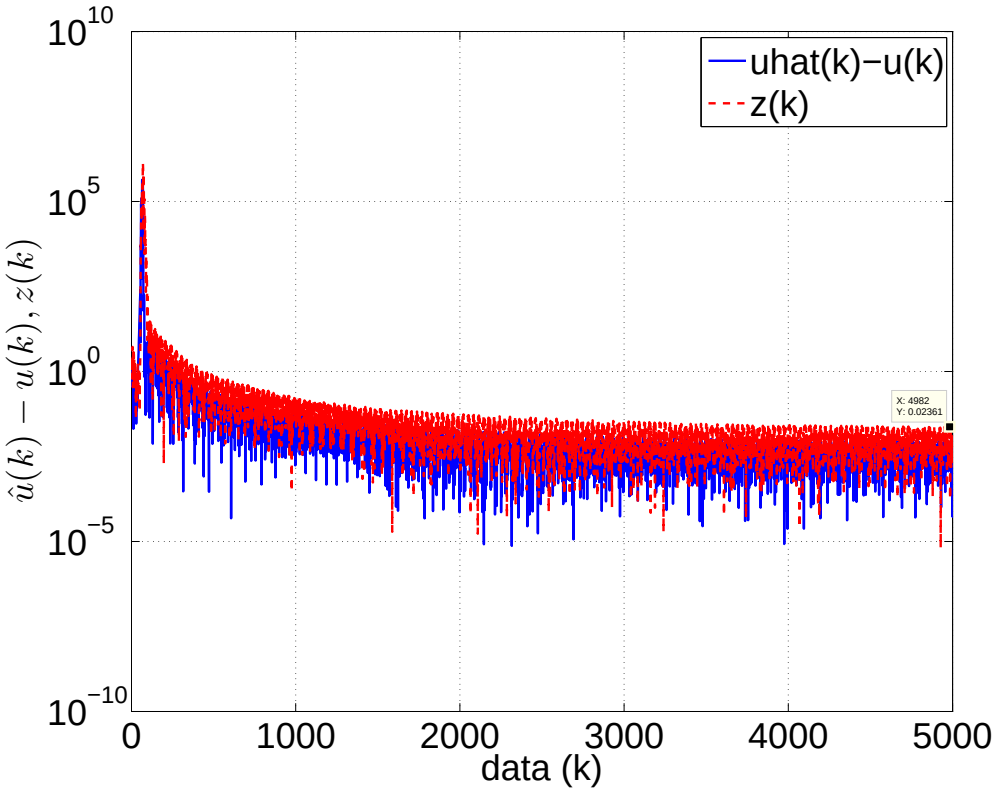


Figure 2.15: Residuals $\hat{u}(k) - u(k)$ and $z(k)$ show that both the input reconstruction error and the output residual become small.

the previous example.

Figure 2.17 compares the improved transient performance over the previous example since G_{FIR} is a better approximation of G at the relevant frequencies.

Figure 2.18 shows the residual $\hat{u}(k) - u(k)$ and $z(k)$.

2.8 Conclusions

In this chapter we presented two methods for forward-propagating input reconstruction for nonminimum-phase systems. First we developed an open-loop method, which uses a finite impulse response (FIR) model to approximate the inverse of the system at a finite number of frequencies. This FIR model was then used with the mea-

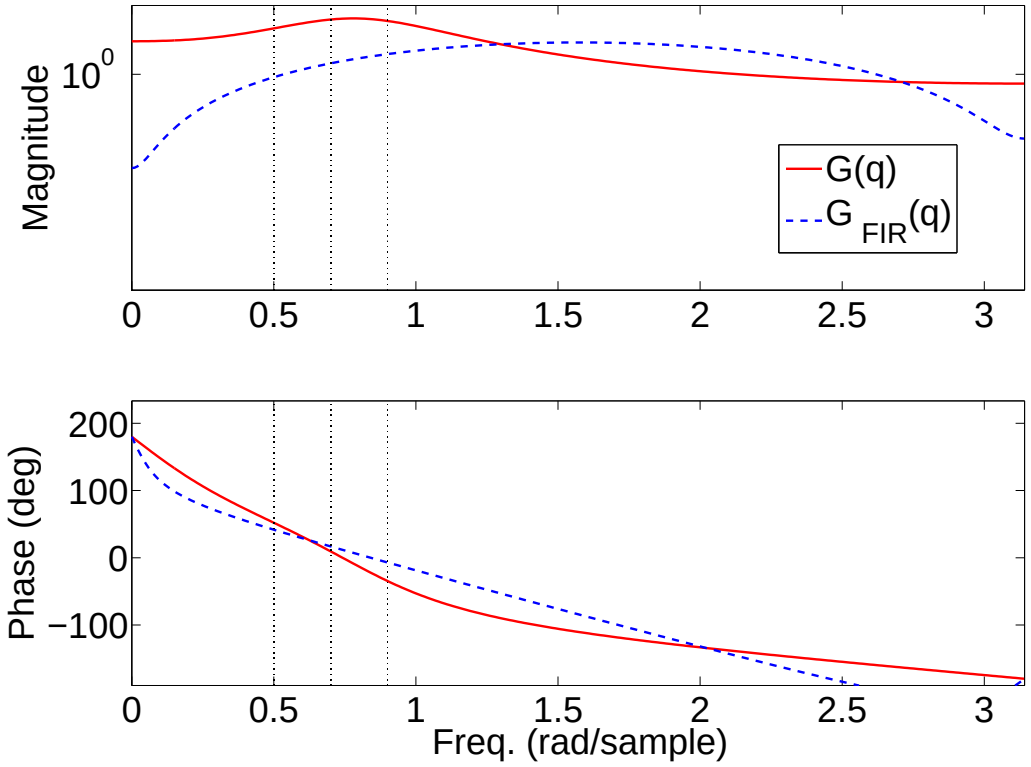


Figure 2.16: Frequency response comparison of G and G_{FIR}, where the red vertical lines denote the frequencies Ω_i.

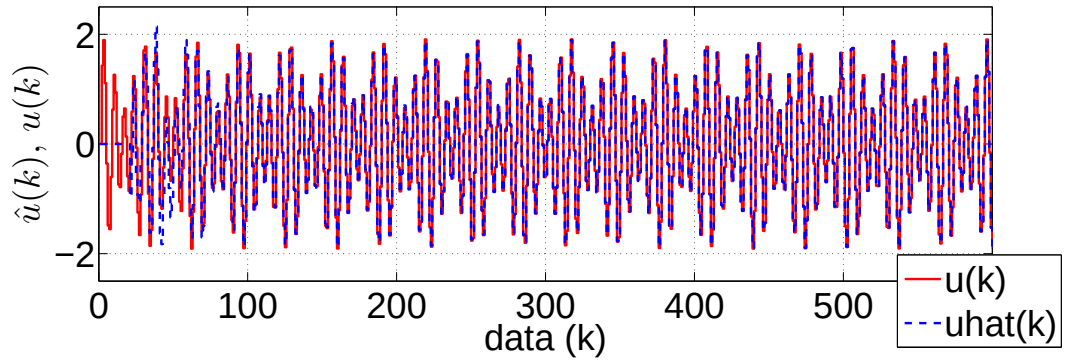


Figure 2.17: Comparison between $\hat{u}(k)$ and $u(k)$. Note that in this case $\tilde{u}(k)$ is small, with no transient.

sured system output to estimate the system input. Next, we presented a closed-loop method, which uses the output residual to adaptively tune the coefficients of an infi-

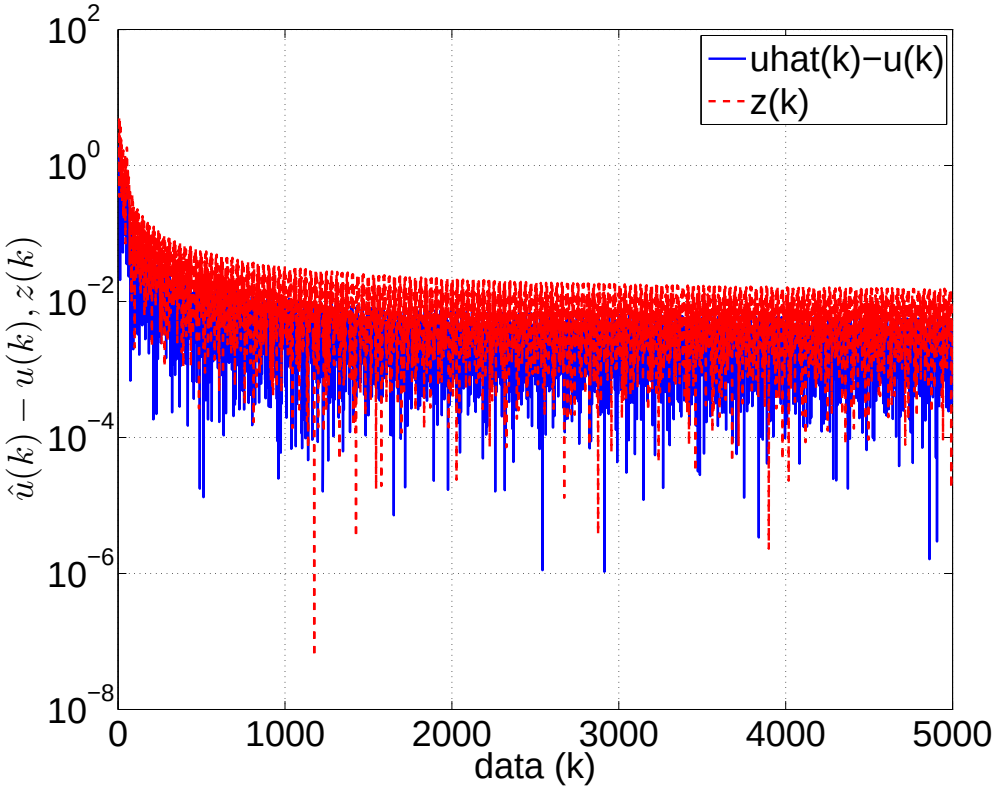


Figure 2.18: Residuals $\hat{u}(k) - u(k)$ and $z(k)$ show that both the input reconstruction error and the output residual become small.

nite impulse response transfer function to estimate the system input. The closed-loop method is able to minimize the error $\tilde{u}(k)$ despite inaccuracy in the system estimate at the frequencies of the unknown input.

Future research will focus on the effect of sensor noise on the accuracy of the input reconstruction. Finally, the ability to adaptively estimate the phase shift of the transfer function at the input frequency as well as the phase of the input may have applications to the area of phase and frequency estimation.

CHAPTER III

Application of Input Reconstruction to Semi-Parametric Identification of Hammerstein Systems

3.1 Introduction

Nonlinear model structures involving a single linear dynamic block and a single nonlinear static block comprise a natural first step in nonlinear system identification. A nonlinear mapping at the input yields a Hammerstein model, while a nonlinear mapping at the output yields a Wiener model. The literature on system identification for these models structures is extensive, and shows that nonlinear system identification for these problems remains a challenging and useful area of research. Representative references on Hammerstein and Wiener system identification include [42, 43, 44] and [42, 45, 46, 43, 39], respectively.

The starting point for this Chapter on Hammerstein-system identification is the semiparametric approach developed in [39] for identifying Wiener systems. This approach involves two steps and is semiparametric, which, as described in [47], refers to the fact that the nonlinear block is estimated nonparametrically, whereas the linear dynamics are identified parametrically. In the first step, a single harmonic is applied to the system to determine the phase shift of the output of the linear system relative

the input to the linear system; this information is then used to construct a nonparametric approximation of the nonlinearity. Next, using knowledge of the output nonlinearity, which is not assumed to be invertible, retrospective cost optimization (RCO) is used to estimate the parameters of the linear model. RCO was originally developed for adaptive control [48, 49, 50], and has subsequently been applied to various identification problems, including model refinement [39, 11].

In this chapter we develop a two-step semiparametric technique for identifying single-input, single-output (SISO) Hammerstein systems. In the first step we use a sufficiently rich signal to estimate the linear dynamics of the system. We then use retrospective cost optimization to estimate the parametric model of the linear dynamics, although alternative techniques [51] can be used for this step, such as output-error modeling methods. When an initial model of the linear system is available, retrospective cost optimization can utilize this information. In this chapter we do not assume that an initial model is available.

For the second step, we apply a single harmonic input signal, and measure the output once the trajectory of the system reaches steady state. We then use input reconstruction, which is based on l-delay left invertibility of the linear parametric model [27]. By using input reconstruction with the identified linear parametric model, we estimate the inaccessible intermediate signal. We examine the estimate of the intermediate signal (which is not harmonic due to the nonlinearity) relative to the input, and use the symmetry properties of these signals to estimate the nonharmonic phase shift. This estimate allows us to infer the phase shift of the unmeasured intermediate signal (that is, the output of the nonlinearity) and thus reconstruct this signal up to an arbitrary amplitude. By plotting the reconstructed intermediate signal versus the input signal, we thus obtain a nonparametric approximation of the nonlinear block of the system.

The contents of this chapter are as follows. In Section 2 we define the Ham-

merstein identification problem. A method for parametric identification of the linear time-invariant dynamics using retrospective cost optimization is reviewed in Section 3, while a method for nonparametric identification of the static nonlinearity using input reconstruction and single harmonic input is presented in Section 4. These methods are demonstrated on numerical and experimental examples in sections 5 and 6, respectively. Concluding remarks are presented in Section 7.

3.2 Problem Formulation

Consider the block-structured Hammerstein model shown in Figure 3.1a, with input $u(k) \in \mathbb{R}$ and intermediate signal

$$v(k) = \mathcal{H}(u(k)), \tag{3.1}$$

where $\mathcal{H} : \mathbb{R} \longmapsto \mathbb{R}$ is the static nonlinearity, and $\mathcal{L}$ is the SISO discrete-time linear time-invariant dynamic system

$$x(k+1) = Ax(k) + Bv(k), \tag{3.2}$$

$$y(k) = Cx(k), \tag{3.3}$$

where $y(k) \in \mathbb{R}$ is the output, $x(k) \in \mathbb{R}^n$ is the state vector, and k is the sample index.

We assume that $\mathcal{L}$ is asymptotically stable and $\mathcal{H}$ is piecewise continuous. Note that we do not assume that $\mathcal{H}$ is invertible, one-to-one, continuous, or $\mathcal{H}(0) = 0$. Also, we assume that $v(k)$ is not accessible, and that $x(0)$ is unknown and possibly nonzero.

Figure 3.1b shows the scaled-domain modification $\mathcal{L}_\lambda(\nu) \triangleq \mathcal{L}\left(\frac{\nu}{\lambda}\right)$ of $\mathcal{L}$, where λ is a nonzero real number. Therefore, $\mathcal{L}_\lambda(\lambda v) = \mathcal{L}(v)$. Each value of λ scales both the

gain of $\mathcal{L}$ and the domain of $\mathcal{H}$. However, λ is not identifiable.

(a) (b)

Figure 3.1: (a) Block-structured Hammerstein model, where u is the input, v is the intermediate signal, y is the output, $\mathcal{H}$ is a static nonlinearity, and $\mathcal{L}$ is a discrete-time linear time-invariant dynamic system. (b) An equivalent scaled model, where λ is a scaling factor and $\mathcal{L}_\lambda$ is a scaled-domain modification of $\mathcal{L}$ satisfying $\mathcal{L}_\lambda(\lambda v) = \mathcal{L}(v)$. The scaling factor λ is not identifiable.

3.3 Parametric Identification of the Linear Time-Invariant Dynamics

Using a sufficiently rich input u and measuring the output y of the Hammerstein system, we identify a model of $\mathcal{L}$ given by $\hat{\mathcal{L}}$ using retrospective cost optimization (RCO). The RCO algorithm is presented in [11] together with guidelines for choosing its tuning parameters, namely, n_c, p, and α. We do not assume that the initial state of $\mathcal{L}$ is zero.

Consider the adaptive feedback architecture for $\hat{\mathcal{L}}$ shown in Figure 3.2, where $\hat{\mathcal{L}}_{\mathrm{m}}$ denotes the initial model with input $w \in \mathbb{R}$ and output $\hat{y} \in \mathbb{R}$, and where $\hat{\mathcal{L}}_\Delta$ denotes the feedback delta model with inputs u, $\hat{y} \in \mathbb{R}$ and output w. The goal is to adaptively tune $\hat{\mathcal{L}}_\Delta$ so that the performance variable

$$z(k) \triangleq y(k) - \hat{y}(k) \tag{3.4}$$

is minimized in the presence of the identification signal u. For simplicity, we choose

$\hat{\mathcal{L}}_{\mathrm{m}}$ to be the one-step delay $1/\mathbf{z}$. In the case that information is known about the linear system, an initial model can be used in place of the unit delay.

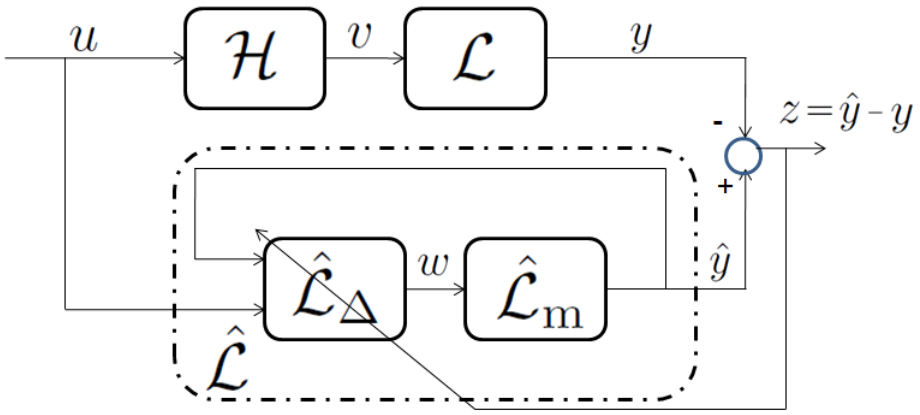


Figure 3.2: Identification architecture for Hammerstein models using RCO.

When an initial model of the linear system is not available, various system identification methods can be used to obtain the parametric estimate of $\mathcal{L}$, such as output-error modeling methods [51]; see Example 6.3. To identify a parametric model $\hat{\mathcal{L}}$ for the linear system using the signals u and y, we assume that $\mathcal{H}$ is approximately linear for the domain of u used to drive y. In a sense, we ignore the nonlinearity $\mathcal{H}$, the validity of this assumption is investigated in [52].

3.4 Nonparametric Identification of the Static Nonlinearity

Consider the harmonic input signal

$$u(k) = A_0 \sin(\omega_0 k T_s) = A_0 \sin(\Omega_0 k), \tag{3.5}$$

where A_0 is the amplitude, ω_0 is the angular frequency in rad/sec, T_{s} is the sample period in sec/sample, and $\Omega_0 \triangleq \omega_0 T_{\mathrm{s}}$ is the angular sample frequency in rad/sample.

The intermediate signal is

$$v(k) = \mathcal{H}(u), \tag{3.6}$$

and the output signal is

$$y(k) = G(\mathbf{z})\mathcal{H}(u), \tag{3.7}$$

where $G(\mathbf{z}) = C(\mathbf{z}I - A)^{-1}B$ is a transfer function representation of $\mathcal{L}$. To obtain the nonparametric estimate $\hat{\mathcal{H}}$ of the nonlinearity $\mathcal{H}$, we require an estimate $\hat{v}(k)$ of the inaccessible intermediate signal $v(k)$. To obtain $\hat{v}(k)$, we use input reconstruction [26, 27]. Together, $\hat{\mathcal{L}}$ and $\hat{\mathcal{H}}$ comprise a *semiparametric model* of the Hammerstein system.

3.4.1 Input Reconstruction

With an estimate $\hat{\mathcal{L}}$ of the linear system $\mathcal{L}$, we pass (3.5) through (3.1)-(3.3). Next we wish to obtain an estimate $\hat{v}$ of the intermediate signal v. To obtain $\hat{v}$, we use input reconstruction, which depends on the l-delay invertibility of the estimate of $G(\mathbf{z})$.

Let l be a nonnegative integer. Then $G(\mathbf{z})$ is *l-delay invertible* if there exists a proper transfer function $G_l(\mathbf{z})$ (called an *l-delay inverse of $G(\mathbf{z})$*) such that $G_l(\mathbf{z})G(\mathbf{z}) = \mathbf{z}^{-l}$ for almost all $\mathbf{z} \in \mathbb{C}$ [27]. For a SISO system, G is l-delay invertible for all $l \geq d$, where d is the relative degree of $G(\mathbf{z})$.

Using input reconstruction we obtain

$$\hat{v}_l(k) = \hat{G}^{-1}(\mathbf{z})\mathbf{z}^{-l}y(k), \tag{3.8}$$

where $\hat{v}_l(k) = \hat{v}(k-l)$ for $k \geq l$. For the case $l = d$, the estimate $\hat{v}(k)$ is in

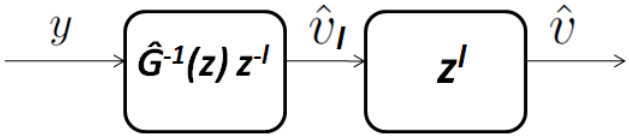


Figure 3.3: Input reconstruction. Using l-delay invertibility of the estimated linear system, the intermediate signal can be reconstructed.

nonharmonic phase with the true intermediate signal $v(k)$, where nonharmonic phase is defined in Section 3.4.4. We also consider the case $l \neq d$, since d is not assumed to be known. Using the harmonic input u and $\hat{v}$ we can determine the nonharmonic phase shift, and then determine an estimate of the nonparametric map of the nonlinearity.

3.4.2 Signal Symmetry

Note that the continuous-time harmonic signal $\sin(\omega_0 t)$ is symmetric in the intervals $\left[0, \frac{1}{2}T_0\right]$ and $\left[\frac{1}{2}T_0, T_0\right]$ about the points $\frac{1}{4}T_0$ and $\frac{3}{4}T_0$, respectively, where $T_0 \triangleq \dfrac{2\pi}{\omega_0}$ is the period of the harmonic input. To preserve symmetry for the sampled signal (3.5) about the points $\frac{1}{4}T_0$ and $\frac{3}{4}T_0$, we assume that $\Omega_0 = \dfrac{\pi}{2m}$, where m is a positive integer. Thus $N_0 \triangleq 4m = \dfrac{T_0}{T_s}$ is the period of the discrete-time input (3.5). With this choice of Ω_0, the sampled signal $u(k)$ is symmetric in the intervals $\left[0, \frac{1}{2}N_0\right]$ and $\left[\frac{1}{2}N_0, N_0\right]$ about the points $\frac{1}{4}N_0$ and $\frac{3}{4}N_0$, respectively. Furthermore, $q \triangleq d - l$ is an integer, that is, the estimated intermediate signal $\hat{v}(k)$, which is shifted relative to $u(k)$ due to $d - l$, the error in the relative degree between $\mathcal{L}$ and $\hat{\mathcal{L}}$, is symmetric about $\frac{1}{4}N_0 + q$ in the interval $\left[q, \frac{1}{2}N_0 + q\right]$ and about $\frac{3}{4}N_0 + q$ in the interval $\left[\frac{1}{2}N_0 + q, N_0 + q\right]$.

Next, we note that the intermediate signal v, which is not generally harmonic, possesses the same symmetry as u on the same intervals. By exploiting knowledge of this symmetry, we can identify the *nonharmonic phase shift* of $\hat{v}$ relative to u. Since $\hat{v}$ is not sinusoidal, the nonharmonic phase shift of $\hat{v}$ relative to u refers to the shifting

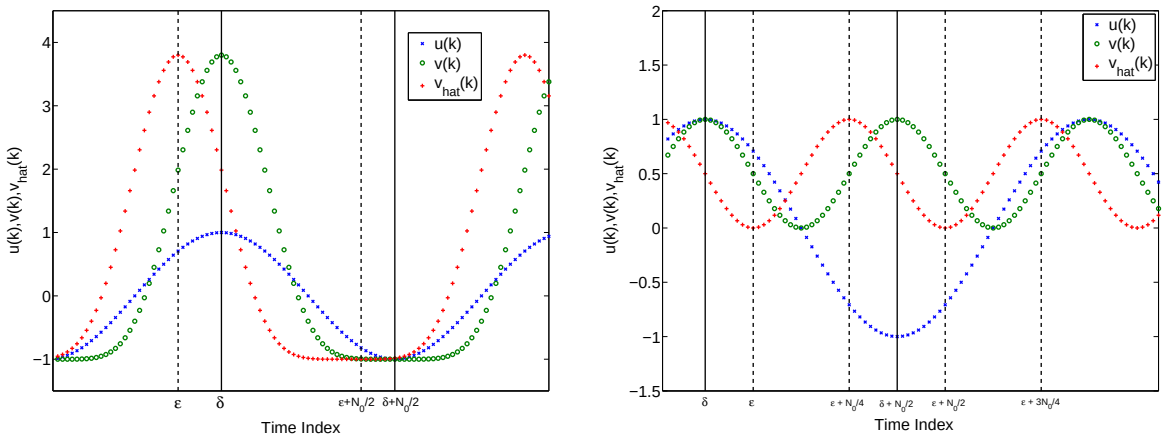
of the symmetric portions of $\hat{v}$ relative to the symmetric portions of u. Knowledge of this nonharmonic phase shift allows us to determine v up to a constant multiple, specifically, $\hat{v}$ is shifted relative to u by a known number of samples.

To clarify the above discussion, we present two examples using $A_0 = 1$, $m = 18$ (so that $\Omega_0 = \pi/36$). First, consider the polynomial nonlinearity $v = \mathcal{H}(u) = 0.6(u+1)^3 - 1$, which is neither even nor odd. Figure 3.4a illustrates the resulting signals $u(k)$, $v(k)$, $\hat{v}(k)$ in harmonic steady state, where the delay q, between $v(k)$ and $\hat{v}(k)$, is added to simulate modeling inaccuracy. Note that u and v are symmetric about the discrete-time index δ in the interval $\left[\delta - \frac{1}{4}N_0, \delta + \frac{1}{4}N_0\right]$ and about $\delta + \frac{1}{2}N_0$ in the interval $\left[\delta + \frac{1}{4}N_0, \delta_2 + \frac{3}{4}N_0\right]$. Likewise, $\hat{v}$ is symmetric about the discrete-time index ε in the interval $\left[\varepsilon - \frac{1}{4}N_0, \varepsilon + \frac{1}{4}N_0\right]$ and about $\varepsilon + \frac{1}{2}N_0$ in the interval $\left[\varepsilon + \frac{1}{4}N_0, \varepsilon + \frac{3}{4}N_0\right]$.

Second, we consider the even polynomial nonlinearity $v = \mathcal{H}(u) = u^2$. Figure 3.4b illustrates the resulting signals $u(k)$, $v(k)$, and $\hat{v}(k)$ in harmonic steady state. The signal u of Figure 3.4b is equal to the signal u shown in Figure 3.4a. However, in addition to the two points of symmetry shown in Figure 3.4a, note that v and $\hat{v}$ have two additional points of symmetry, specifically, v is symmetric about $\delta + \frac{1}{4}N_0$ in the interval $\left[\delta, \delta + \frac{1}{2}N_0\right]$ and about $\delta + \frac{3}{4}N_0$ in the interval $\left[\delta + \frac{1}{2}N_0, \delta + N_0\right]$, and $\hat{v}$ is symmetric about $\varepsilon + \frac{1}{4}N_0$ in the interval $\left[\varepsilon, \varepsilon + \frac{1}{2}N_0\right]$ and about $\varepsilon + \frac{3}{4}N_0$ in the interval $\left[\varepsilon + \frac{1}{2}N_0, \varepsilon + N_0\right]$.

3.4.3 Symmetry Search Algorithm

We now review from [39] an algorithm for determining ε from $\hat{v}$. We then use ε to estimate the nonharmonic phase shift of $\hat{v}$ relative to u. For convenience, we assume that the harmonic steady state begins at $k = 0$. Consider the signal $\hat{v}$ shown in Figure 3.5, and let $n \geq 6m$ denote the width of the data window so that it includes at least one and a half periods. To encompass a complete signal period, we construct



(a) (b)

Figure 3.4: Illustration of the symmetry properties of the signals u, v, and $\hat{v}$. For (a) the non-even polynomial nonlinearity is $v = \mathcal{H}(u) = 0.6(u+1)^3 - 1$ and (b) the even polynomial nonlinearity is $v = \mathcal{H}(u) = u^2$. For both cases, u and v are symmetric about δ in the interval $\left[\delta - \frac{1}{4}N_0, \delta + \frac{1}{4}N_0\right]$ and about $\delta + \frac{1}{2}N_0$ in the interval $\left[\delta + \frac{1}{4}N_0, \delta + \frac{3}{4}N_0\right]$, while $\hat{v}$ is symmetric about ε in the interval $\left[\varepsilon - \frac{1}{4}N_0, \varepsilon + \frac{1}{4}N_0\right]$ and about $\varepsilon + \frac{1}{2}N_0$ in the interval $\left[\varepsilon + \frac{1}{4}N_0, \varepsilon + \frac{3}{4}N_0\right]$. In addition, for the case of an even polynomial nonlinearity shown in (b), v and $\hat{v}$ have two additional points of symmetry, specifically, v is symmetric about $\delta + \frac{1}{4}N_0$ in the interval $\left[\delta, \delta + \frac{1}{2}N_0\right]$ and about $\delta + \frac{3}{4}N_0$ in the interval $\left[\delta + \frac{1}{2}N_0, \delta + N_0\right]$, and $\hat{v}$ is symmetric about $\varepsilon + \frac{1}{4}N_0$ in the interval $\left[\varepsilon, \varepsilon + \frac{1}{2}N_0\right]$ and about $\varepsilon + \frac{3}{4}N_0$ in the interval $\left[\varepsilon + \frac{1}{2}N_0, \varepsilon + N_0\right]$.

a sliding window with $N_0 + 1$ data points. The window is divided into quarters as shown in Figure 3.5.

Next, for $k = 0, \ldots, n - N_0$, define

$$\beta_1(k) \triangleq \sum_{i=1}^{2m-1} \left|\hat{v}\left(k + i - 1\right) - \hat{v}\left(k + 2m - i + 1\right)\right|, \tag{3.9}$$

which is the sum of the absolute difference in magnitude for each pair of candidate symmetric points in the first and second quarters about the point $k + \frac{1}{4}N_0$ for the

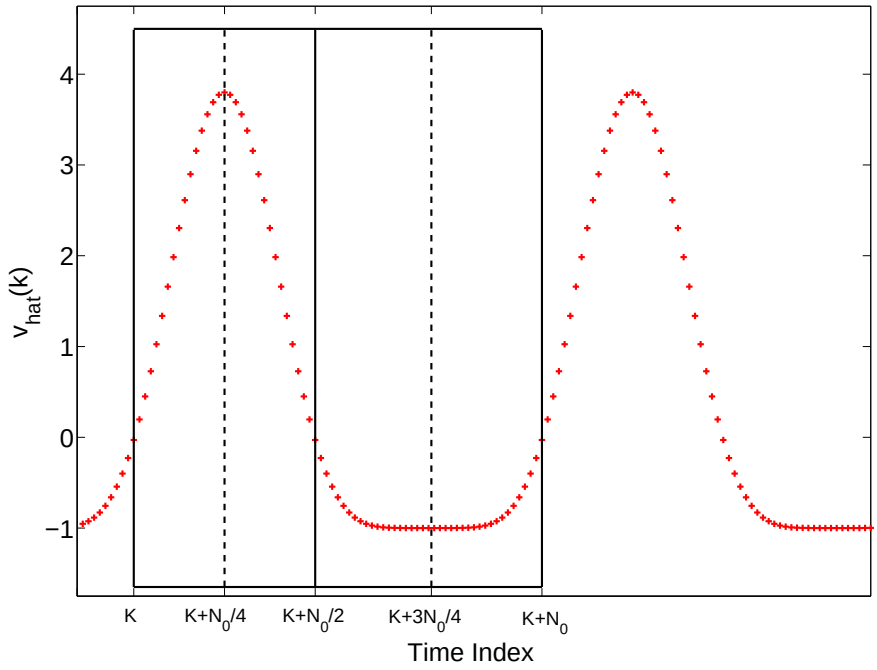


Figure 3.5: Illustration of the symmetry search algorithm. The solid line box comprises the sliding window of length $N_0 + 1$ starting at time k, while the dashed lines indicate the windowed points of symmetry.

sliding window starting at time step k. Likewise, for $k = 0, \ldots, n - N_0$, define

$$\beta_2(k) \stackrel{\triangle}{=} \sum_{i=1}^{2m-1} |\hat{v}\,(k + 2m + i - 1) - \hat{v}\,(k + 4m - i + 1)|, \tag{3.10}$$

for each pair of candidate symmetric points in the third and fourth quarters about the point $k + \frac{3}{4}N_0$. The values of β_1 and β_2 quantify the symmetry error about the points $k + \frac{1}{4}N_0$ and $k + \frac{3}{4}N_0$, respectively, for each allowable value of k. Thus, using (3.9) and (3.10), we define the *symmetry error index* $\beta(k) \stackrel{\triangle}{=} \beta_1(k) + \beta_2(k)$, corresponding to the sliding window starting at point k, for $k = 0, \ldots, n - N_0$.

For $k = 0, \ldots, n - N_0$, let $k_0 < N_0$ be the minimizer of $\beta(k)$. We use knowledge of k_0 to determine the location of the points of symmetry ε and $\varepsilon + \frac{1}{2}N_0$ for the sliding window starting at point k_0. In particular, since k_0 is the starting point of the window that minimizes β and since ε and $\varepsilon + \frac{1}{2}N_0$ are, respectively, the quarter point and

three quarter point of the same window, it follows that

$$\varepsilon = k_0 + \frac{1}{4}N_0, \quad \varepsilon + \frac{1}{2}N_0 = k_0 + \frac{3}{4}N_0. \tag{3.11}$$

To illustrate the symmetry search algorithm, we reconsider the example considered in Figures 3.4a and 3.5, where $v = \mathcal{H}(u) = 0.6(u+1)^3 - 1$. Note that $\mathcal{H}$ is not even. Figure 3.6a shows the values of β calculated for $\hat{v}(k)$ on the interval $[k_0, k_0 + 2N_0]$. Since, in Figure 3.6a, the data window of $\hat{v}(k)$ is selected to start at $k_0 = \varepsilon - \frac{1}{4}N_0$, the minimum values of $\beta(k)$ occur at k_0 and $k_0 + N_0$, where $k_0 + N_0$ is the start of the next period and, thus, need not be considered. Thus, using the unique minimizer k_0 of $\beta(k)$, it follows that the locations of the points of symmetry are given by (3.11).

Next, for the even nonlinearity $v = \mathcal{H}(u) = u^2$ considered in Figure 3.4b, Figure 3.6b shows the values of $\beta(k)$ calculated for $v(k)$ on the interval $[k_0, k_0 + 2N_0]$. In this case, the minimum values of $\beta(k)$ occur at k_0, $k_0 + \frac{1}{2}N_0$, and $k_0 + N_0$, where $k_0 + N_0$ is the start of the next period and, thus, need not be considered. Thus, using k_0, it follows that the locations of the points of symmetry are given by (3.11). Also, using $k_0 + \frac{1}{2}N_0$, we obtain two additional points of symmetry given by

$$\varepsilon + \frac{1}{4}N_0 = k_0 + \frac{1}{2}N_0, \quad \varepsilon + \frac{3}{4}N_0 = k_0 + N_0. \tag{3.12}$$

This ambiguity is due to the fact that ε and $\varepsilon + \frac{1}{2}N_0$ are the midpoints of two identical symmetric portions of $\hat{v}$. Thus, the start of the data window within which the function has the symmetry properties illustrated in Figure 3.5 can be taken as either k_0 or $k_0 + \frac{1}{2}N_0$. Note that the second minimizer $k_0 + \frac{1}{2}N_0$ appears only for even nonlinearities.

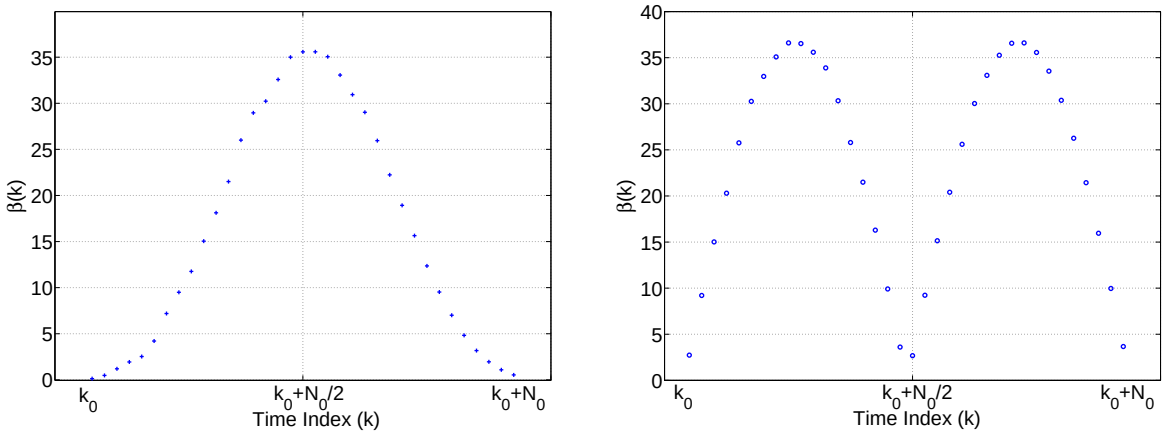


Figure 3.6: Illustration of the symmetry error index $\beta(k)$ given by (3.9). The values of $\beta(k)$ are shown for two static nonlinearities, namely, (a) a non-even polynomial and (b) an even polynomial.

3.4.4 Nonparametric Approximation of the Static Nonlinearity

Using δ, which is assumed to be known from the harmonic input u, and the estimate of ε obtained from $\hat{v}$ in Section 3.4.3, we now determine an estimate $\hat{\phi}$ of the nonharmonic phase shift of $\hat{v}$ relative to u by $\hat{\phi} \triangleq \Omega_0(\varepsilon - \delta)$, which is an estimate of $d - l$. Moreover, define the virtual signal

$$\tilde{v}(k) \triangleq \hat{v}\left(k + \frac{\hat{\phi}}{\Omega_0}\right), \tag{3.13}$$

which is an approximation of the intermediate signal v. Note that the amplitude of $\tilde{v}(k)$ is irrelevant due to the scaling factor λ shown in Figure 3.1b. Using $\tilde{v}$ and u, the nonparametric estimate of $\mathcal{H}$ is given by

$$\hat{\mathcal{H}} \triangleq \{(u(k_0), \tilde{v}(k_0)), (u(k_0+1), \tilde{v}(k_0+1)), \ldots, (u(n), \tilde{v}(n))\}, \tag{3.14}$$

where each pair $(u(k), \tilde{v}(k),)$, for $k = 0, \ldots, n$, determines a value of the nonpara-

metric estimate $\hat{\mathcal{H}}$ of $\mathcal{H}$.

Figure 3.6 shows that, depending on the type of nonlinearity, $\beta(k)$ has either one or two minima within each period. For a non-even polynomial nonlinearity, $\beta(k)$ has one minimum within each period. Therefore, the estimate of the nonharmonic phase shift has two candidate values, namely, $\hat{\phi}$ and $\hat{\phi}+\pi$. For an even nonlinearity, $\beta(k)$ has two minima within each period. Therefore, the estimate of the nonharmonic phase shift has four candidate values, namely, $\hat{\phi}$, $\hat{\phi}+\frac{\pi}{2}$, $\hat{\phi}+\pi$, and $\hat{\phi}+\frac{3\pi}{2}$. However, for the even case, $\hat{\phi}$ and $\hat{\phi}+\pi$ yield the same nonparametric model $\hat{\mathcal{H}}$, while $\hat{\phi}+\frac{\pi}{2}$ and $\hat{\phi}+\frac{3\pi}{2}$ yield the same $\hat{\mathcal{H}}$.

Therefore, in both the non-even and even cases, there are two candidate nonparametric estimates of $\mathcal{H}$, both of which are constructed using (3.13) and (3.14). In practice q is small compared to N_0, therefore, it is reasonable to assume that ϕ is the correct nonharmonic phase shift candidate for estimating $\mathcal{H}$.

3.5 Simulated Examples

To demonstrate semiparametric Hammerstein model identification, we consider two static nonlinearities, namely, a non-even case and an even case. For both examples, we choose G to have poles $0.34\pm0.87\jmath, -0.3141\pm0.9\jmath, 0.05\pm0.3122\jmath, -0.6875$ and zeros $0.14\pm0.97\jmath, -0.12\pm0.62\jmath, -0.89$ with monic numerator and denominator. Also, $u(k)$ is chosen to be a realization of zero-mean Gaussian white noise with standard deviation $\sigma_u = 3.5$.

Example 3.5.1. (Non-even Polynomial) Consider $\mathcal{H}$ defined by

$$v = \mathcal{H}(u) = u^3 + 4u + 7. \tag{3.15}$$

The parameters for nonparametric identification of $\mathcal{H}$ are $m = 500$ and $A_0 = 5$. Figure 3.7a shows the frequency response of the true dynamic model G and the identified

model using RCO. The RCO parameters used to identify the linear dynamic system are set as $n_c = 9$, $p = 1$, and $\alpha = 1$. Figure 3.7b compares the true nonlinearity with the identified nonlinearity estimated using input reconstruction.

Example 3.5.2. (Even Polynomial) Consider $\mathcal{H}$ defined by

$$v = \mathcal{H}(u) = 7u^4 + u^2. \tag{3.16}$$

The parameters for nonparametric identification of $\mathcal{H}$ are $m = 500$ and $A_0 = 5$. Figure 3.8a shows the frequency response of the true dynamic model G and the identified model using RCO. The RCO parameters used to identify the linear dynamic system are set as $n_c = 9$, $p = 1$, and $\alpha = 1$. Figure 3.8b compares the true nonlinearity (blue line) with the identified nonlinearity estimated using input reconstruction (red crosses).

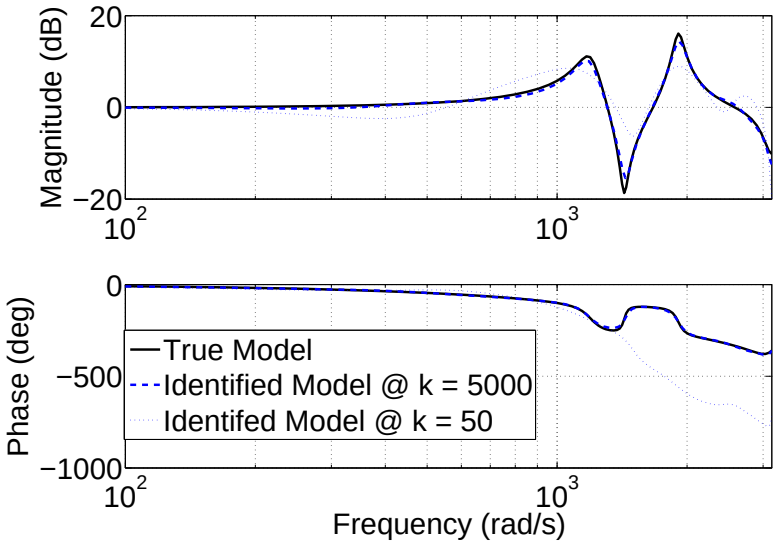
To illustrate the ambiguity discussed in Section 3.4.4, we select the incorrect nonharmonic phase shift, specifically, $\hat{\phi} + \frac{\pi}{2}$, which is represented by the black circles in Figure 3.8b. Note that the incorrect nonharmonic phase shift produces an erroneous nonparametric model of the nonlinearity.

3.6 Experimental Examples

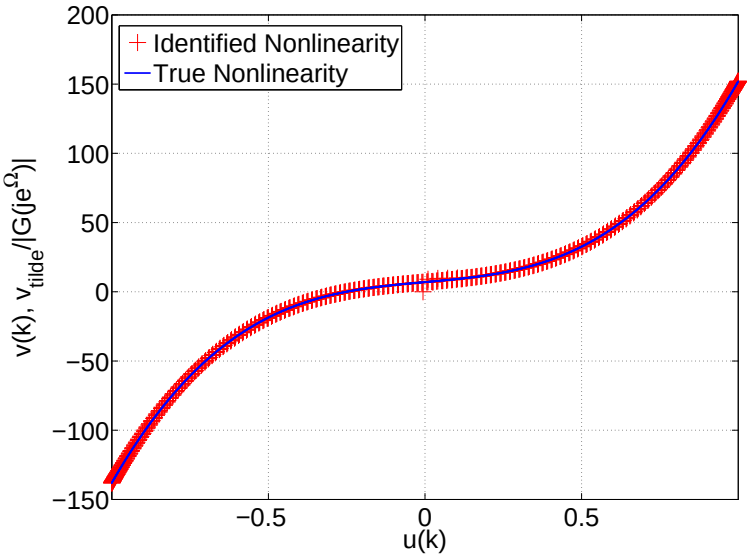
We now present experimental examples using a resistor-inductor-capacitor (RLC) circuit. The true parametric model of the RLC circuit is generated from first principles, where $R = 250\ \Omega$. $L = 55$ mH, $C = 23.5\ \mu$F, and

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ \frac{-1}{LC} & \frac{-R}{L} \end{bmatrix} x + \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} v, \tag{3.17}$$

$$y = \begin{bmatrix} 0 & R \end{bmatrix} x, \tag{3.18}$$

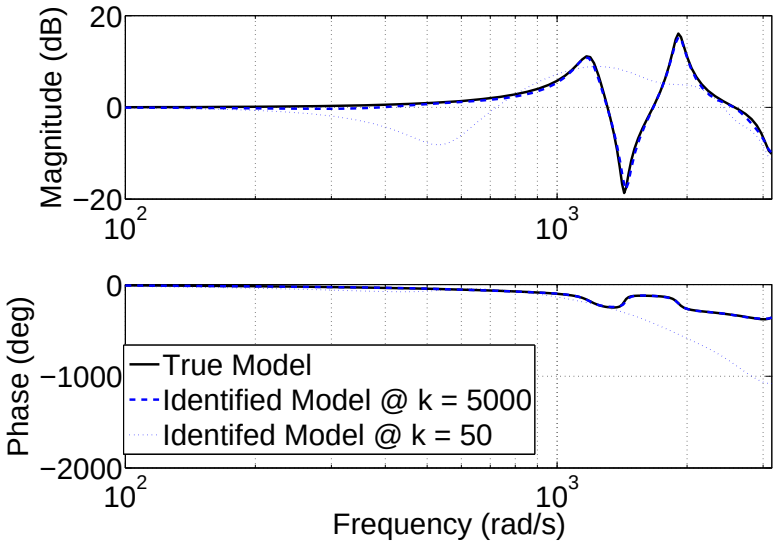


(a)

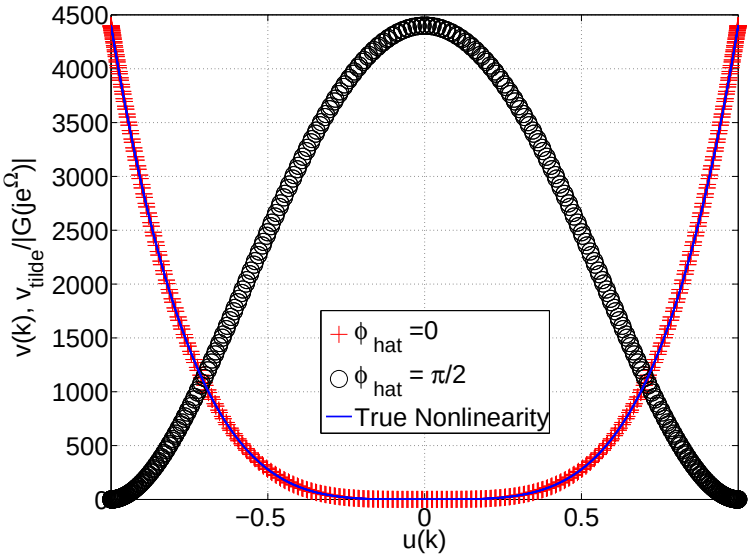


(b)

Figure 3.7: (a) Frequency response comparison of the true G and the identified LTI system, where k is the number of data points used to determine the identified model. For $k = 5000$, the traces for the true and identified models almost coincide. (b) Identified nonlinearity versus true nonlinearity, where $m = 500$ and $A_0 = 5$. The argument of the identified nonlinearity is scaled by $\frac{1}{|G(e^{j\Omega_0})|}$ to facilitate comparison with the true nonlinearity (3.15)



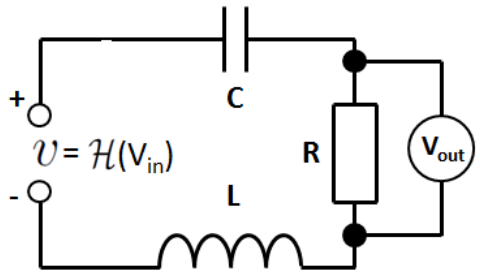
(a)



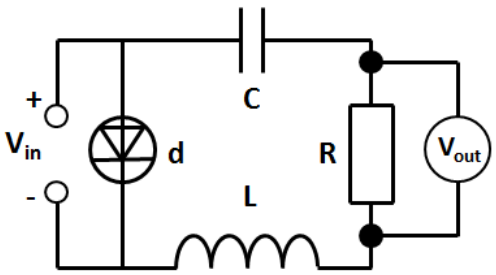
(b)

Figure 3.8: (a) Frequency response comparison of the true G and the identified LTI system, where k is the number of data points used to determine the identified model. (b) Identified nonlinearities versus true nonlinearity, where $m = 500$ and $A_0 = 5$. The argument of the identified nonlinearity is scaled by $\frac{1}{|G(e^{j\Omega_0})|}$ to facilitate comparison with the true nonlinearity (3.16). The red crosses represent the identified nonlinearity, the black circles, represent the identified nonlinearity using the incorrect nonharmonic phase shift.

where $x \in \mathbb{R}^2$ is the state vector, which is the circuit charge and current. For the following examples, G is a discrete time transfer function representation of (3.17)–(3.18), with a sampling rate of $Ts = 0.0001$. Figure 3.9a shows the RLC circuit, where the nonlinearity is a saturation in the actuation voltage.



(a)



(b)

Figure 3.9: (a) Block diagram representation of the series RLC circuit, where the input voltage is modified by $\mathcal{H}$. For this example, $\mathcal{H}$ is a saturation function. (b)A series RLC circuit in parallel with a diode. The resulting system is a Hammerstein system where the diode can be represented as a static nonlinearity and the series RLC circuit is the linear model.

Example 3.6.1. (Saturation) Consider $\mathcal{H}$ defined by

$$v = \mathcal{H}(u) = \begin{cases} u, & \text{if } -1 < u < 1; \\ 1, & \text{if } u \geq 1; \\ -1, & \text{if } u \leq 1. \end{cases} \tag{3.19}$$

The parameters for nonparametric identification of $\mathcal{H}$ are $m = 500$ and $A_0 = 5$. Figure 3.10a shows the frequency response of the true dynamic model G, and the identified model using RCO. The RCO parameters used to identify the linear dynamic system are set as $n_c = 9$, $p = 1$, and $\alpha = 1$. Figure 3.10b compares the true nonlinearity with the identified nonlinearity estimated using input reconstruction.

We now reconsider the RLC circuit in Figure 3.9b, where a diode is presented in parallel with the circuit. The diode modifies the input voltage according to a nonlinear function $\mathcal{H}$. We first determine $\mathcal{H}$ experimentally, using a circuit where the diode is the sole component. $\mathcal{H}$ is approximatively given by

$$v = \mathcal{H}(u) = \begin{cases} u, & \text{if } u < 0.07; \\ 0.07, & \text{if } u \geq 0.07. \end{cases} \tag{3.20}$$

We view (3.20) as the truth model of $\mathcal{H}$.

Example 3.6.2. (Diode in Parallel with RLC Circuit) Consider $\mathcal{H}$ which is given by (3.20), which is in parallel with the linear dynamic system given by (3.17) and (3.18). In this example, the diode is assumed to be inaccesible, namely, it can not be directly measured. The RCO parameters used to identify the linear dynamic system are set as $n_c = 3$, $p = 1$, and $\alpha = 1$. Figure 3.11a shows the frequency response of the true dynamic model G, and the identified model using RCO. For nonparametric identification of $\mathcal{H}$, $m = 500$ and $A_0 = 0.919$. Figure 3.11b compares with true nonlinearity and the identified nonlinearity estimated using input reconstruction.

Example 3.6.3. (Diode in Parallel with RLC Circuit) We now revisit the diode problem without using RCO, by fitting an output error model (OEM) of the form

$$y(k) = \frac{B(\mathbf{z})}{F(\mathbf{z})}u(k) + e(k), \tag{3.21}$$

where $B(\mathbf{z})$ and $F(\mathbf{z})$ are polynomials. The coefficients of $B(\mathbf{z})$, and $F(\mathbf{z})$ are determined by minimizing the error term $e(k)$, using a maximum likelihood method.

Figure 3.12a shows the frequency response of the true dynamic model G and the identified model using the OEM fit. The identified nonlinearity using input reconstruction and the actual nonlinearity are shown in Figure 3.12b.

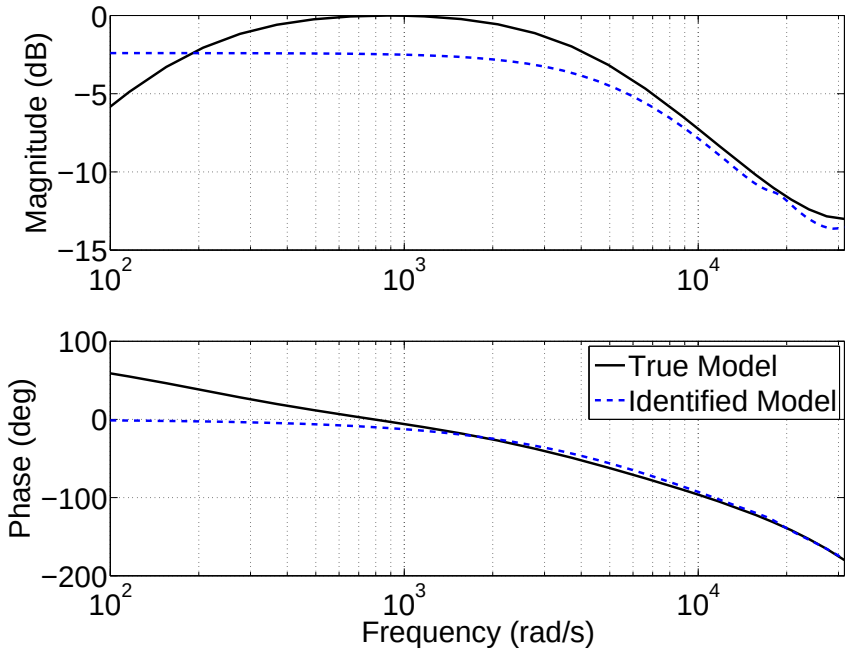
3.7 Conclusion

In this chapter we develop a two-step method to identify semiparametric models for SISO discrete-time Hammerstein systems. We assume that the linear dynamic block is asymptotically stable, and the static nonlinearity is piecewise continuous.

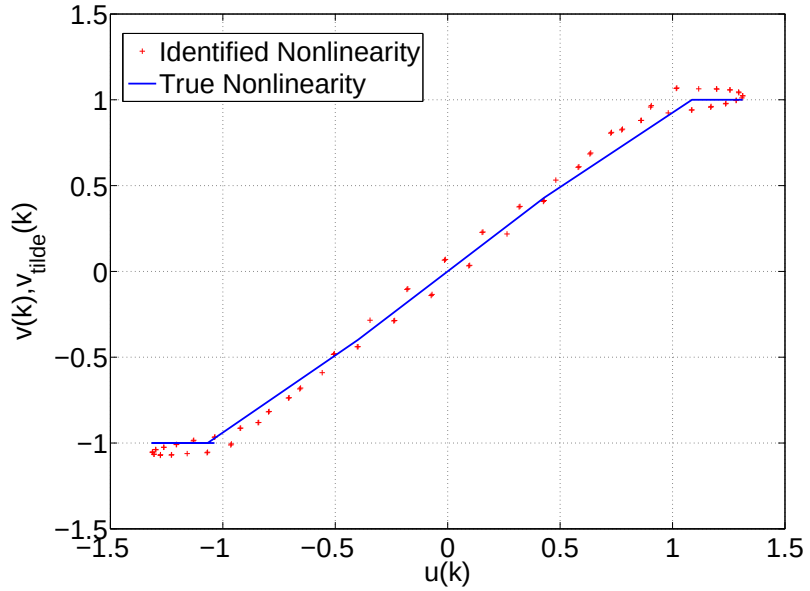
First, we identify a parametric model of the linear dynamic system using a sufficiently rich input. We identify the parametric model using retrospective cost optimization and, in one example, using an output error model.

Second, we choose a single harmonic input and measure the system output when the state trajectory is in harmonic steady state. Using the system output and input reconstruction we estimate the intermediate signal, which may be shifted compared to the true intermediate signal, since the relative degree of the linear system is unknown. We exploit symmetry properties of the estimated intermediate signal compared to the input, which we use to approximate the nonharmonic phase shift and, therefore, estimate the delay between the estimate and true intermediate signal. Using the estimate of the intermediate signal, a nonparametric model of the static nonlinearity is obtained.

This method is effectively demonstrated on two simulated examples. Furthermore, two experimental examples are presented, namely, an RLC circuit with saturation at the input, and an RLC circuit containing a diode.

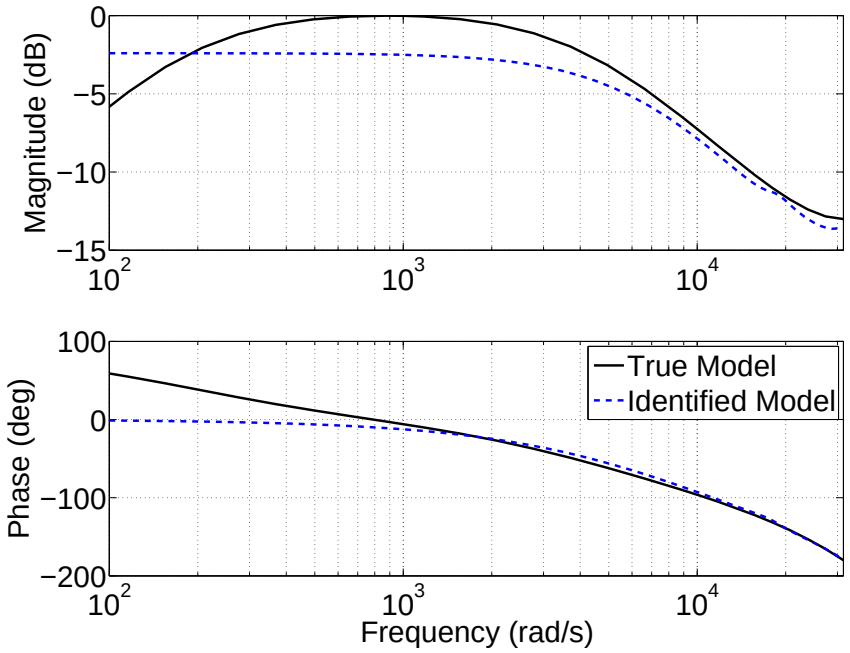


(a)

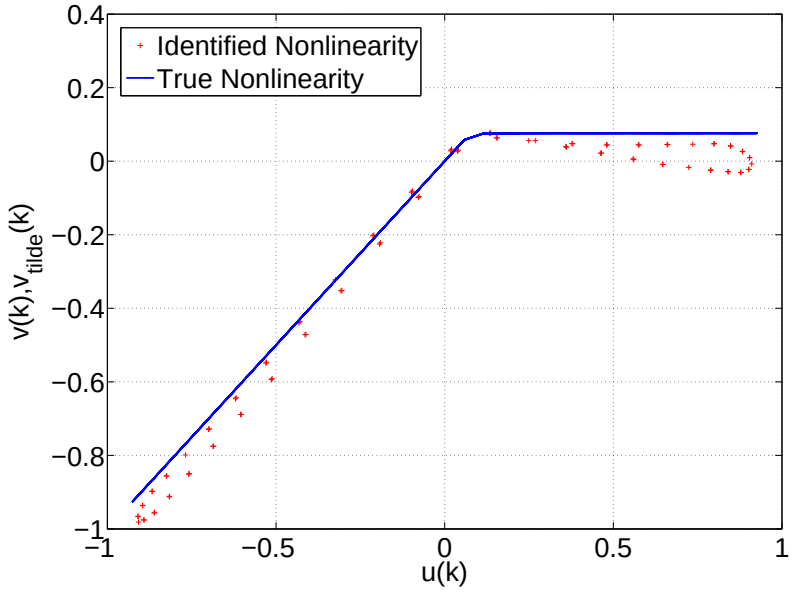


(b)

Figure 3.10: (a) Frequency response comparison of the true G and the identified LTI system, where k is the number of data points used to determine the identified model. For $k = 5000$, the traces for the true and identified models almost coincide. (b) Identified nonlinearity versus true nonlinearity, where $m = 500$ and $A_0 = 5$. The argument of the identified nonlinearity is scaled by $\frac{1}{\left|G(e^{\jmath\Omega_0})\right|}$ to facilitate comparison with the true nonlinearity (3.19).

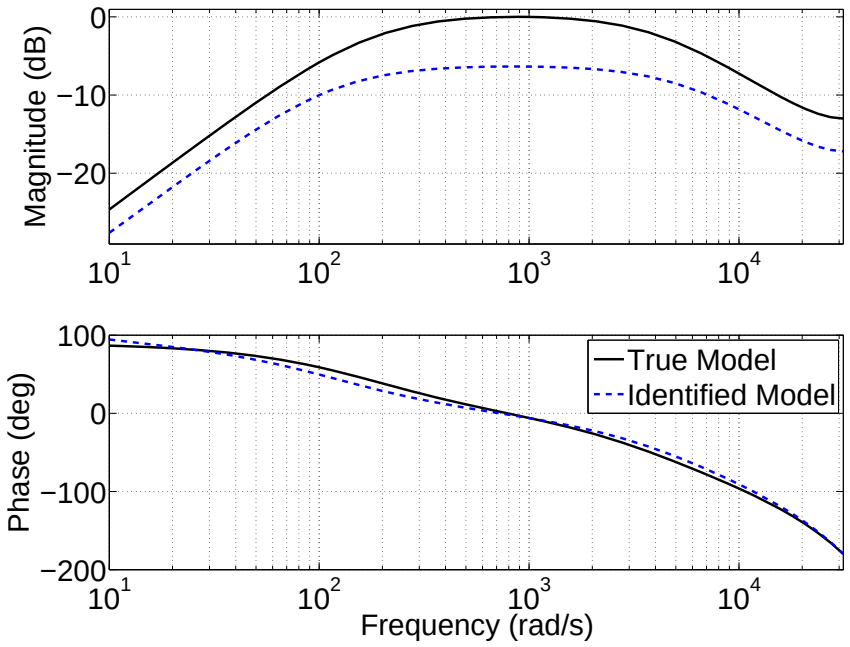


(a)

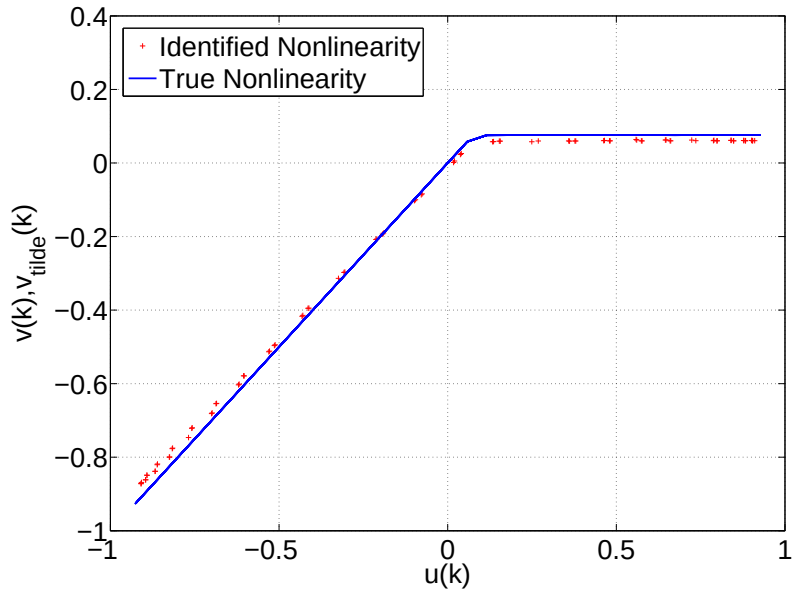


(b)

Figure 3.11: (a) Frequency response comparison of the true G and the identified LTI system the lines for the true and identified models almost coincide. (b) Identified nonlinearity versus true nonlinearity, where $m = 500$ and $A_0 = 0.919$. The argument of the identified nonlinearity is scaled by $\frac{1}{|G(e^{j\Omega_0})|}$ to facilitate comparison with the true nonlinearity (3.20)



(a)



(b)

Figure 3.12: (a) Frequency response comparison of the true G and the identified LTI system, where the identified linear system is an output error model (OEM) fit (b) Identified nonlinearity versus true nonlinearity, where $m = 500$ and $A_0 = 5$. The argument of the identified nonlinearity is scaled by $\frac{1}{|G(e^{j\Omega_0})|}$ to facilitate comparison with the true nonlinearity (3.20)

CHAPTER IV

Model Refinement

4.1 Introduction

In this chapter we consider the problem of data-based model refinement, where we assume the availability of an initial model, which may incorporate both physical laws and empirical observations. The components of the initial model may have varying degrees of fidelity, reflecting knowledge or ignorance of the relevant physics as well as the availability of data. With this initial model as a starting point, our goal is to use additional measurements to refine the model. In particular, we wish to update the components of the model that are poorly modeled, thereby resulting in a higher fidelity model [11, 53, 54, 55, 56].

System identification is typically concerned with the construction of a model of the entire system from measured inputs to measured outputs. In contrast, our goal is to identify only a subsystem of the model, where the remainder of the model is not modified. One motivation for this objective is to improve understanding of the physics of the poorly modeled subsystem despite its low accessibility. Here, accessibility refers to the availability of measurements or estimates of the inputs and outputs of the unknown subsystem. This lack of accessibility leads to a nonstandard system identification problem.

This chapter goes beyond [11, 53, 54, 55, 56] in two ways. First, the model refine-

ment algorithm described in Section II is based on the extension of the retrospective cost adaptive control (RCAC) algorithm described in [57]. The algorithm in [57] requires knowledge of a limited number of Markov parameters of the plant, and thus simplifies earlier versions of RCAC described in [50, 58, 59, 60]. Therefore, the algorithm in [57] improves the model refinement technique described in [11, 55, 41]. Furthermore, this chapter encompasses multiple versions of the model refinement problem, including system emulation and subsystem identification. In the former case, we seek an estimate of the unknown subsystem that allows the overall model to approximate the true system. In this case, there is no expectation that the constructed subsystem model approximates the unknown subsystem. In contrast, in the latter case, we seek an estimate of the unknown subsystem that approximates the unknown subsystem.

4.2 Problem Formulation

Consider the MIMO discrete-time main system

$$x(k+1) = Ax(k) + Bu(k) + D_1 w(k), \tag{4.1}$$

$$y(k) = Cx(k), \tag{4.2}$$

$$y_0(k) = E_1 x(k) + v(k), \tag{4.3}$$

where $x(k) \in \mathbb{R}^n$, $y(k) \in \mathbb{R}^{l_y}$, $y_0(k) \in \mathbb{R}^{l_{y_0}}$, $u(k) \in \mathbb{R}^{l_u}$, $w(k) \in \mathbb{R}^{l_w}$, and $k \geq 0$. The main system (4.1)–(4.3) is interconnected with the unknown subsystem modeled by

$$u(k) = G_{\mathrm{s}}(q)y(k), \tag{4.4}$$

where q is the forward shift operator. The system (4.1)–(4.4) represents the true system. We assume that the excitation signal $w(k)$ is known. $v(k)$ denotes measurement noise.

Next, we assume a model of the main system of the form

$$\hat{x}(k+1) = \hat{A}\hat{x}(k) + \hat{B}\hat{u}(k) + \hat{D}_1 w(k), \tag{4.5}$$

$$\hat{y}(k) = \hat{C}\hat{x}(k), \tag{4.6}$$

$$\hat{y}_0(k) = \hat{E}_1\hat{x}(k), \tag{4.7}$$

where $\hat{x}(k) \in \mathbb{R}^{\hat{n}}$, $\hat{y}(k) \in \mathbb{R}^{l_{\hat{y}}}$, $\hat{y}_0(k) \in \mathbb{R}^{l_{y_0}}$, $\hat{u}(k) \in \mathbb{R}^{l_{\hat{u}}}$. The model of the main system is interconnected with the subsystem model

$$\hat{u}(k) = \hat{G}_{\mathrm{s}}(q)\hat{y}(k). \tag{4.8}$$

The goal is to estimate a subsystem model $\hat{G}_{\mathrm{s}}(q)$ that minimizes a cost function based on the performance variable

$$z(k) \stackrel{\triangle}{=} \hat{y}_0(k) - y_0(k) \in \mathbb{R}^{l_z} \tag{4.9}$$

We estimate $\hat{G}_{\mathrm{s}}(q)$ by retrospectively reconstructing the signal $\hat{u}(k)$ that minimizes the performance at the current time step. The reconstruction of $\hat{u}(k)$ uses minimal modeling information about the true system (4.1)–(4.3), namely, a limited number of Markov parameters. We then use $\hat{u}(k)$ and $\hat{y}(k)$ to construct $\hat{G}_{\mathrm{s}}(q)$. Figure 4.1 illustrates the model-refinement architecture, which includes system emulation and subsystem identification as special cases. Table 4.2 indicates the switch positions for various model-refinement architectures.

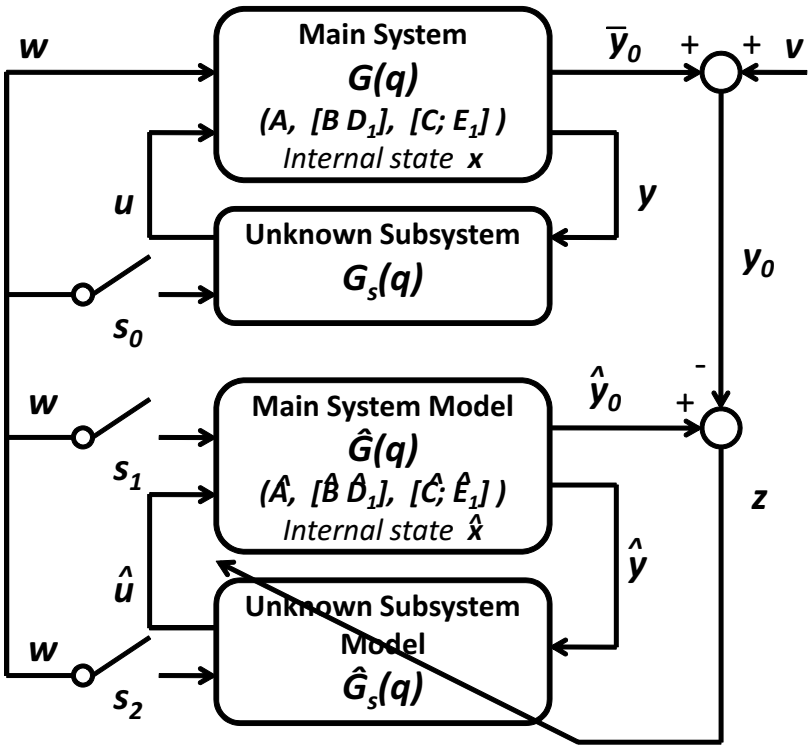


Figure 4.1: Model-refinement architectures. The switches s_0, s_1 and s_2 are used to define different architectures.

Case	s_0	s_1	s_2	Remarks
1	0	0	1	System emulation without subsystem excitation
2	0	1	0	System emulation without subsystem excitation. If A, B, C are known, this case is subsystem identification
3	1	0	1	System emulation with subsystem excitation
4	1	1	1	System emulation with subsystem excitation. If A, B, C are known, this case is subsystem identification

Table II. Switch positions for various model-refinement architectures. A switch in position 1 indicates

the switch is closed, whereas a switch in position 0 indicates it is open.

The goal of system emulation is to determine a subsystem model $\hat{G}_{\mathrm{s}}(q)$ such that the closed-loop frequency response of the true system (from w to y_0) matches the closed-loop frequency response of the system model (from w to $\hat{y}_0$). Since the matrices A, B, C are unknown, the matrices $\hat{A}, \hat{B}, \hat{C}$ in the main system model are approximations of A, B, C. The accuracy of this approximation determines how well the constructed subsystem model approximates the unknown subsystem. In the idealized case of subsystem identification, where A, B, C are known exactly, we set $\hat{A} = A$, $\hat{B} = B$, and $\hat{C} = C$ and use architectures 2 and 4 from Table 4.2 to obtain a subsystem model $\hat{G}_{\mathrm{s}}(q)$ that approximates the unknown subsystem $G_{\mathrm{s}}(q)$. However, the less stringent objective of system emulation is to obtain a model of the unknown subsystem such that the closed-loop model approximates the true closed-loop system.

4.3 Retrospective Surrogate-Cost-Based Signal Reconstruction

We begin by defining Markov parameters of the main system model $\hat{G}(q)$. For $i \geq 1$, let

$$H_i \triangleq \hat{E}_1 \hat{A}^{i-1} \hat{B}. \tag{4.10}$$

Therefore, $H_1 = \hat{E}_1 \hat{B}$ and $H_2 = \hat{E}_1 \hat{A} \hat{B}$. Let r be a positive integer. Then, for all $k \geq r$,

$$\hat{x}(k) = \hat{A}^r \hat{x}(k-r) + \sum_{i=1}^{r} \hat{A}^{i-1} \hat{B} \hat{u}(k-i) + \sum_{i=1}^{r} \hat{A}^{i-1} \hat{D}_1 w(k-i), \tag{4.11}$$

and thus

$$z(k) = \hat{E}_1 \hat{A}^r \hat{x}(k-r) - y_0(k) + \sum_{i=1}^{r} \hat{E}_1 \hat{A}^{i-1} \hat{D}_1 w(k-i) + \bar{H}\hat{\bar{U}}(k-1), \tag{4.12}$$

where

$$\bar{H} \triangleq \begin{bmatrix} H_1 & \cdots & H_r \end{bmatrix} \in \mathbb{R}^{l_z \times r l_{\hat{u}}}$$

and

$$\hat{\bar{U}}(k-1) \triangleq \begin{bmatrix} \hat{u}(k-1) \\ \vdots \\ \hat{u}(k-r) \end{bmatrix}.$$

Next, we rearrange the columns of $\bar{H}$ and the components of $\hat{\bar{U}}(k-1)$ and partition the resulting matrix and vector so that

$$\bar{H}\hat{\bar{U}}(k-1) = \mathcal{H}'\hat{U}'(k-1) + \mathcal{H}\hat{U}(k-1), \tag{4.13}$$

where $\mathcal{H}' \in \mathbb{R}^{l_z \times (r l_{\hat{u}} - l_{\hat{U}})}$, $\mathcal{H} \in \mathbb{R}^{l_z \times l_{\hat{U}}}$, $\hat{U}'(k-1) \in \mathbb{R}^{r l_{\hat{u}} - l_{\hat{U}}}$, and $\hat{U}(k-1) \in \mathbb{R}^{l_{\hat{U}}}$. Then, we can rewrite (4.12) as

$$z(k) = \mathcal{S}(k) + \mathcal{H}\hat{U}(k-1), \tag{4.14}$$

where

$$\mathcal{S}(k) \triangleq \hat{E}_1 \hat{A}^r \hat{x}(k-r) - y_0(k) + \sum_{i=1}^{r} \hat{E}_1 \hat{A}^{i-1} \hat{D}_1 w(k-i) + \mathcal{H}'\hat{U}'(k-1). \tag{4.15}$$

For example, $\bar{H} = \begin{bmatrix} H_1 & H_2 & H_3 & H_4 & H_5 \end{bmatrix}$,

$$\mathcal{H}' = \begin{bmatrix} H_1 & H_2 & H_4 \end{bmatrix}, \quad \hat{U}'(k-1) = \begin{bmatrix} \hat{u}(k-1) \\ \hat{u}(k-2) \\ \hat{u}(k-4) \end{bmatrix},$$

and

$$\mathcal{H} = \begin{bmatrix} H_3 & H_5 \end{bmatrix}, \quad \hat{U}(k-1) = \begin{bmatrix} \hat{u}(k-3) \\ \hat{u}(k-5) \end{bmatrix}.$$

Note that the decomposition of $\bar{H}\hat{\bar{U}}(k-1)$ in (4.13) is not unique. Let s be a positive integer. Then for $i = 1, \ldots, s$, we replace $\mathcal{H}$, $\hat{U}(k-1)$, $\mathcal{H}'$, and $\hat{U}'(k-1)$ in (4.13) with $\mathcal{H}_j \in \mathbb{R}^{l_z \times l_{\hat{U}_j}}$, $\hat{U}_j(k-1) \in \mathbb{R}^{l_{\hat{U}_j}}$, $\mathcal{H}'_j \in \mathbb{R}^{l_z \times (rl_{\hat{u}} - l_{\hat{U}_j})}$, and $\hat{U}'_j(k-1) \in \mathbb{R}^{rl_{\hat{u}} - l_{\hat{U}_j}}$, respectively, such that (4.13) becomes

$$\bar{H}\hat{\bar{U}}(k-1) = \mathcal{H}'_j\hat{U}'_j(k-1) + \mathcal{H}_j\hat{U}_j(k-1). \tag{4.16}$$

Therefore, for $j = 1, \ldots, s$, we can rewrite (4.14) as

$$z(k) = \mathcal{S}_j(k) + \mathcal{H}_j\hat{U}_j(k-1), \tag{4.17}$$

where

$$\mathcal{S}_j(k) \triangleq \hat{E}_1\hat{A}^r\hat{x}(k-r) + y_0(k) + \sum_{i=1}^{r} \hat{E}_1\hat{A}^{i-1}\hat{D}_1 w(k-i) + \mathcal{H}'_j\hat{U}'_j(k-1). \tag{4.18}$$

Next, let $0 \leq k_1 \leq k_2 \leq \cdots \leq k_s$. Replacing k by $k - k_j$ in (4.17) yields

$$z(k-k_j) = \mathcal{S}_j(k-k_j) + \mathcal{H}_j\hat{U}_j(k-k_j-1). \tag{4.19}$$

Now, by stacking $z(k-k_1),\ldots,z(k-k_s)$, we define the *extended performance*

$$Z(k) \triangleq \begin{bmatrix} z(k-k_1) \\ \vdots \\ z(k-k_s) \end{bmatrix} \in \mathbb{R}^{sl_z}. \tag{4.20}$$

Therefore,

$$Z(k) \triangleq \tilde{\mathcal{S}}(k) + \tilde{\mathcal{H}}\hat{\tilde{U}}(k-1), \tag{4.21}$$

where

$$\tilde{\mathcal{S}}(k) \triangleq \begin{bmatrix} \mathcal{S}_1(k-k_1) \\ \vdots \\ \mathcal{S}_s(k-k_s) \end{bmatrix} \in \mathbb{R}^{sp} \tag{4.22}$$

and $\hat{\tilde{U}}(k-1)$ has the form

$$\hat{\tilde{U}}(k-1) \triangleq \begin{bmatrix} \hat{u}(k-q_1) \\ \vdots \\ \hat{u}(k-q_g) \end{bmatrix} \in \mathbb{R}^{gl_{\hat{u}}}, \tag{4.23}$$

where $k_1 \leq q_1 < q_2 < \cdots < q_g \leq k_s + r$. The vector $\hat{\tilde{U}}(k-1)$ is formed by stacking $\hat{U}_1(k-k_1-1),\ldots,\hat{U}_s(k-k_s-1)$ and removing copies of repeated components, and $\tilde{\mathcal{H}} \in \mathbb{R}^{sl_z \times gl_{\hat{u}}}$ is constructed according to the structure of $\hat{\tilde{U}}(k-1)$. For example, with $s=2$, $k_1=0$, and $k_2=2$, stacking $\hat{U}_1(k-1) = \begin{bmatrix} \hat{u}(k-1) \\ \hat{u}(k-2) \end{bmatrix}$ and $\hat{U}_2(k-3) = \hat{u}(k-3)$ results in

$$\hat{\tilde{U}}(k-1) = \begin{bmatrix} \hat{u}(k-1) \\ \hat{u}(k-2) \\ \hat{u}(k-3) \end{bmatrix}, \quad \tilde{\mathcal{H}} = \begin{bmatrix} H_1 & H_2 & H_3 \\ 0 & H_1 & H_2 \\ 0 & 0 & H_3 \end{bmatrix}. \tag{4.24}$$

Note that $\tilde{\mathcal{H}}$ consists of the entries of $\mathcal{H}_1, \ldots, \mathcal{H}_s$ arranged according to the structure of $\tilde{U}(k-1)$.

Next, we define the *retrospective performance*

$$\hat{z}(k-k_j) \stackrel{\triangle}{=} \mathcal{S}_j(k-k_j) + \mathcal{H}_j U_j^*(k-k_j-1), \tag{4.25}$$

where the past input estimates $\hat{U}_j(k-k_j-1)$ in (4.19) are replaced by the retrospectively optimized input estimates $U_j^*(k-k_j-1)$, which are determined below. In analogy with (4.20), the *extended retrospective performance* is defined as

$$\hat{Z}(k) \stackrel{\triangle}{=} \begin{bmatrix} \hat{z}(k-k_1) \\ \vdots \\ \hat{z}(k-k_s) \end{bmatrix} \in \mathbb{R}^{sl_z} \tag{4.26}$$

and thus is given by

$$\hat{Z}(k) = \tilde{\mathcal{S}}(k) + \tilde{\mathcal{H}}\tilde{U}^*(k-1), \tag{4.27}$$

where the components of $\tilde{U}^*(k-1) \in \mathbb{R}^{l_{\hat{\tilde{U}}}}$ are the components of $U_1^*(k-k_1-1), \ldots, U_s^*(k-k_s-1)$ ordered in the same way as the components of $\hat{\tilde{U}}(k-1)$. Subtracting (4.21) from (4.27) yields

$$\hat{Z}(k) = Z(k) - \tilde{\mathcal{H}}\hat{\tilde{U}}(k-1) + \tilde{\mathcal{H}}\tilde{U}^*(k-1). \tag{4.28}$$

Finally, we define the *retrospective cost function*

$$J(\tilde{U}^*(k-1),k) \triangleq \hat{Z}^{\mathrm{T}}(k)R_1(k)\hat{Z}(k) + \eta(k)\tilde{U}^{*\mathrm{T}}(k-1)R_2(k)\tilde{U}^*(k-1), \tag{4.29}$$

where $R_1(k) \in \mathbb{R}^{l_z s \times l_z s}$ is a positive-definite performance weighting, $R_2(k) \in \mathbb{R}^{gl_{\hat{u}} \times gl_{\hat{u}}}$ is a positive-definite input estimate weighting, and $\eta(k) \geq 0$ is a regularization weighting. The goal is to determine retrospective subsystem output estimates $\tilde{U}^*(k-1)$ that would have provided better performance than the estimated subsystem outputs $\hat{U}(k-1)$ that were used in the model refinement. The retrospectively optimized subsystem outputs $\tilde{U}^*(k-1)$ are then used to update the controller. Substituting (4.28) into (4.29) yields

$$J(\tilde{U}^*(k-1),k) = \tilde{U}^{*\mathrm{T}}(k-1)\mathcal{A}(k)\tilde{U}^*(k-1) + \tilde{U}^{*\mathrm{T}}(k-1)\mathcal{B}^{\mathrm{T}}(k) + \mathcal{C}(k), \tag{4.30}$$

where

$$\mathcal{A}(k) \triangleq \tilde{\mathcal{H}}^{\mathrm{T}}R_1(k)\tilde{\mathcal{H}} + \eta(k)R_2(k), \tag{4.31}$$

$$\mathcal{B}(k) \triangleq 2\tilde{\mathcal{H}}^{\mathrm{T}}R_1(k)[Z(k) - \tilde{\mathcal{H}}\hat{\tilde{U}}(k-1)], \tag{4.32}$$

$$\mathcal{C}(k) \triangleq Z^{\mathrm{T}}(k)R_1(k)Z(k) - 2Z^{\mathrm{T}}(k)R_1(k)\tilde{\mathcal{H}}\hat{\tilde{U}}(k-1) + \hat{\tilde{U}}^{\mathrm{T}}(k-1)\tilde{\mathcal{H}}^{\mathrm{T}}R_1(k)\tilde{\mathcal{H}}\hat{\tilde{U}}(k-1). \tag{4.33}$$

If either $\tilde{\mathcal{H}}$ has full column rank or $\eta(k) > 0$, then $\mathcal{A}(k)$ is positive definite. In this case, $J(\tilde{U}^*(k-1),k)$ has the unique global minimizer

$$\tilde{U}^*(k-1) = -\frac{1}{2}\mathcal{A}^{-1}(k)\mathcal{B}(k), \tag{4.34}$$

which is the retrospectively optimized subsystem outputs.

The regularization weighting $\eta(k)$ can be used to bound the retrospectively opti-

mized subsystem outputs $\tilde{U}^*(k-1)$ and thus indirectly bound estimated subsystem outputs $\hat{\tilde{U}}(k)$. For example, $\eta(k)$ may be performance based

$$\eta(k) = \eta_0(k)||Z(k)||_2^2 \tag{4.35}$$

or error based

$$\eta(k) = \eta_0(k)||\tilde{U}^*(k-2) - \hat{\tilde{U}}(k-2)||_2^2, \tag{4.36}$$

where $\eta_0(k) \geq 0$. Alternatively, the retrospectively optimized subsystem outputs can be bounded directly by using a saturation function, where $\eta(k) \equiv 0$ in (4.31) and (4.34) is replaced by

$$\tilde{U}^*(k-1) \stackrel{\triangle}{=} \text{sat}_{[a,b]}[-\frac{1}{2}\mathcal{A}^{-1}(k)\mathcal{B}(k)], \tag{4.37}$$

where $\text{sat}_{[a,b]}(\zeta)$ is the component-wise saturation function defined for scalar arguments by

$$\text{sat}_{[a,b]}(\zeta) \stackrel{\triangle}{=} \begin{cases} b, & \text{if } \zeta \geq b, \\ \zeta, & \text{if } a < \zeta < b, \\ a & \text{if } \zeta \leq a, \end{cases} \tag{4.38}$$

where $a < b$ are the component-wise saturation levels.

4.3.1 Subsystem Modeling

The subsystem output $\hat{u}(k)$ is given by the exactly proper time-series model of order n_c given by

$$\hat{u}(k) = \sum_{i=1}^{n_c} M_i(k)\hat{u}(k-i) + \sum_{i=0}^{n_c} N_i(k)\hat{y}(k-i) + \sum_{i=0}^{n_c} O_i(k)w(k-i), \tag{4.39}$$

where, for all $i = 1, \ldots, n_c$, $M_i(k) \in \mathbb{R}^{l_{\hat{u}} \times l_{\hat{u}}}$, $N_i(k) \in \mathbb{R}^{l_{\hat{u}} \times l_{\hat{y}}}$ and $O_i(k) \in \mathbb{R}^{l_{\hat{u}} \times l_w}$. The subsystem output (4.39) can be expressed as $\hat{u}(k) = \theta(k)\phi(k-1)$, where $\theta(k) \in \mathbb{R}^{l_{\hat{u}} \times n_c(l_{\hat{u}}+l_{\hat{y}}+l_w)}$ is

$$\theta(k) \triangleq [M_1(k) \ \cdots \ M_{n_c}(k) \ N_1(k) \ \cdots \ N_{n_c}(k) \ O_1(k) \ \cdots \ O_{n_c}(k)], \text{and}$$

$$\phi(k-1) \triangleq \begin{bmatrix} \hat{u}(k-1) \\ \vdots \\ \hat{u}(k-n_c) \\ \hat{y}(k-1) \\ \vdots \\ \hat{y}(k-n_c) \\ w(k-1) \\ \vdots \\ w(k-n_c) \end{bmatrix} \in \mathbb{R}^{n_c(l_{\hat{u}}+l_{\hat{y}}+l_w)} \tag{4.40}$$

Note if $s_2 = 0$ then $w(k)$ and O_i are removed from $\hat{u}(k)$, $\theta(k)$, and $\phi(k-1)$.

4.3.2 Recursive Least Squares Update of $\theta(k)$

We define the cumulative cost function

$$\begin{aligned} J_R(\theta(k)) \triangleq & \sum_{i=q_g+1}^{k} \lambda^{k-i} \|\phi^T(i-q_g-1)\theta^T(k) - u^{*T}(i-q_g)\|^2 \\ & + \lambda^k(\theta(k)-\theta(0))P^{-1}(0)(\theta(k)-\theta(0))^T, \end{aligned} \tag{4.41}$$

where $\|\cdot\|$ is the Euclidean norm and, for some $\varepsilon \in (0,1)$, $\lambda(k) \in (\varepsilon, 1]$ is the forgetting factor, and $P(0) \in \mathbb{R}^{n_c(l_{\hat{u}}+l_z)\times n_c(l_{\hat{u}}+l_z)}$ is positive definite. Minimizing (4.41) yields

$$\begin{aligned}\theta^{\mathrm{T}}(k) \triangleq\ & \theta^{\mathrm{T}}(k-1) + \beta(k)P(k-1)\phi(k-q_g-1) \\ & \cdot [\phi^{\mathrm{T}}(k-q_g-1)P(k-1)\phi(k-q_g-1) + \lambda(k)]^{-1} \\ & \cdot [\theta(k-1)\phi(k-q_g-1) - u^*(k-q_g)]^{\mathrm{T}},\end{aligned} \tag{4.42}$$

where $\beta(k)$ is either 0 or 1. When $\beta(k) = 1$, the controller is allowed to adapt, whereas, when $\beta(k) = 0$, the adaptation is off. $P(k)$ is updated by

$$\begin{aligned}P(k) \triangleq\ & (1-\beta(k))P(k-1) + \beta(k)\lambda^{-1}(k)P(k-1) - \beta(k)\lambda^{-1}(k)P(k-1)\phi(k-q_g-1) \\ & \cdot [\phi^{\mathrm{T}}(k-q_g-1)P(k-1)\phi(k-q_g-1) + \lambda(k)]^{-1}\phi^{\mathrm{T}}(k-q_g-1)P(k-1).\end{aligned} \tag{4.43}$$

We initialize $P(0) = \gamma I$, where $\gamma > 0$. Furthermore, the updates (4.42) and (4.43) are based on the g^{th} component of $\tilde{U}^*(k-1)$. However any or all of the components of $\tilde{U}^*(k-1)$ may be used in the update of $\theta(k)$ and $P(k)$.

4.4 Numerical Examples

We now consider numerical examples with various model-refinement architectures to illustrate the effect of noise and model uncertainty on the emulation of the closed-loop system and, where applicable, the identification of the unknown subsystem. For all examples in this section, RCO is turned on after 100 steps. The level of measurement noise varies for each example, where $v = \mathcal{N}(\mu_v, \sigma_v^2)$ means that the output noise signal v is Gaussian white noise with mean μ_v and variance σ_v^2. We define SNR $\triangleq \frac{\sigma_{\bar{y}_0}^2}{\sigma_v^2}$, where $\sigma_{\bar{y}_0}^2$ is the variance of the output signal $\bar{y}_0$. The case number in each example refers to the positions of the switches in Figure 4.1 as described in

Table 4.2. For all examples, the subsystem model parameters $\theta(k)$ are initialized at zero. For convenience, let $G(q)$ represent the main system, let $G_{\rm cl}(q)$ represent the closed-loop system from w to $\bar{y}_0$, and let $G_{\rm s}(q)$ represent the unknown subsystem.

We consider the spring-mass-damper system shown in Figure 4.2. For $i = 1, 2, 3$, let q_i be the position of $i^{\rm th}$ mass, and let m_i be the mass of the $i^{\rm th}$ block. For $i = 1, 2, 3, 4$, let k_i be the stiffness of the $i^{\rm th}$ spring, and let c_i be the damping coefficient of the $i^{\rm th}$ damper. Finally, let w be the force applied to the second block.

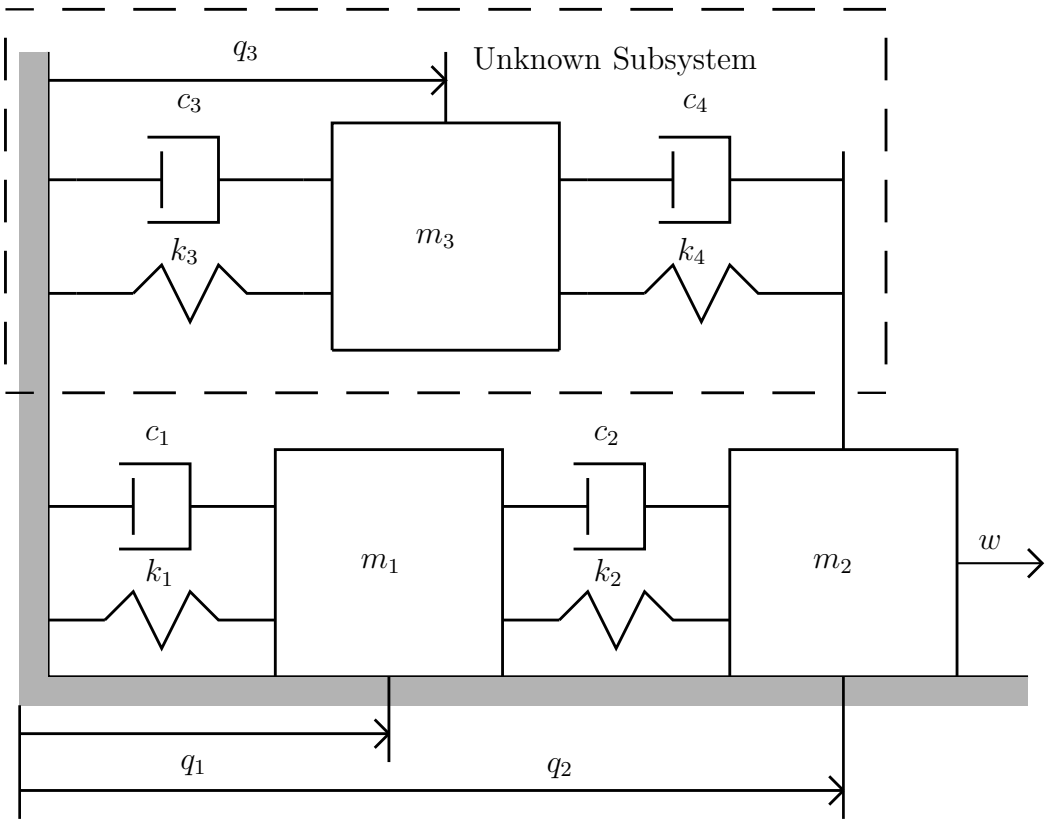


Figure 4.2: Spring-mass-damper system with main system and unknown subsystem.

The discretized equations of motion of the main system are

$$x(k+1) = Ax(k) + Bu(k) + D_1 w(k), \tag{4.44}$$

$$y(k) = Cx(k), \tag{4.45}$$

$$y_0(k) = E_1 x(k) + \nu(k), \tag{4.46}$$

where $x(k) = \begin{bmatrix} q_1(k) & q_2(k) & q_4(k) & q_5(k) \end{bmatrix}^{\mathrm{T}}$,

$$A = \begin{bmatrix} 1 & 0 & T_s & 0 \\ 0 & 1 & 0 & T_s \\ \frac{-T_s(k_1+k_2)}{m_1} & \frac{T_s k_2}{m_1} & 1 - \frac{T_s(c_1+c_2)}{m_1} & \frac{T_s c_2}{m_1} \\ \frac{T_s k_2}{m_2} & \frac{-T_s(k_2+k_4)}{m_2} & \frac{T_s c_2}{m_2} & 1 - \frac{T_s(c_2+c_4)}{m_2} \end{bmatrix},$$

$$D_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{T_s}{m_2} \end{bmatrix}, C = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}^{\mathrm{T}}, E_1 = \begin{bmatrix} 0 \\ \frac{T_s k_4}{m_3} \\ 0 \\ \frac{T_s c_4}{m_3} \end{bmatrix}^{\mathrm{T}}.$$

The discretized equations of motion of the unknown subsystem are

$$x_s(k+1) = A_s x_s(k) + B_s y(k), \tag{4.47}$$

$$u(k) = C_s x(k), \tag{4.48}$$

where

$$x_s(k) = \begin{bmatrix} q_3(k) \\ q_6(k) \end{bmatrix}, A_s = \begin{bmatrix} 1 & T_s \\ -\frac{T_s(k_3+k_4)}{m_3} & 1 - \frac{T_s(c_3+c_4)}{m_3} \end{bmatrix},$$

$$B_s = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, C_s = \begin{bmatrix} \frac{T_s k_4}{m_2} \\ \frac{T_s c_4}{m_2} \end{bmatrix}.$$

Furthermore, $T_s = 0.25$, $m_1 = 4$, $m_2 = 2$, $m_3 = 10$, $k_1 = 12$, $k_2 = 2$, $k_3 = 4$, $k_4 = 6$, $c_1 = 4$, $c_2 = 2$, $c_3 = 5$, and $c_4 = 3$.

Example IV.1. (*Case 1, A, B, C unknown,* SNR = 100). Since A, B, and C are unknown, we choose $\hat{A}$, $\hat{B}$, and $\hat{C}$ such that $\hat{G}(q)$ is stable and minimum phase, but otherwise arbitrarily. More specifically, we choose

$$\hat{A} = \begin{bmatrix} -0.039 & -0.029 \\ 0.023 & 0.0023 \end{bmatrix}, \hat{B} = \begin{bmatrix} 0.003 \\ 0.098 \end{bmatrix},$$
$$\hat{C} = \begin{bmatrix} 2.5 & -0.78 \end{bmatrix}. \tag{4.49}$$

Moreover, $\mu_w = 0$ and $\sigma_w^2 = 5$. For this example we take $n_c = 20$, $\bar{\eta} = 0$, $\beta = 0.01$, and $\tilde{\mathcal{H}} = H_1$, which is the first Markov parameter of $\hat{G}(q)$. The parameters of the model refinement algorithm are chosen such that $z(k)$ is minimized. Figure 4.3 shows that the estimated frequency response of the closed-loop system $\hat{G}_{\mathrm{cl}}(q)$ approximates the closed-loop frequency response of the true system $G_{\mathrm{cl}}(q)$. Next, we run this example with three different SNR values for 5000 time steps. Figure 4.4 shows that, as the SNR increases, the frequency response of $\hat{G}_{\mathrm{cl}}$ provides an improved approximation of the frequency response of G_{cl}.

Example IV.2. (*Case 3, A, B, C unknown,* SNR = 100). The architecture for this example is different from the architecture of Case 1 only in that the unknown subsystem $G_{\mathrm{s}}(q)$ has the additional input w, and hence

$$x_s(k+1) = A_s x_s(k) + B_s \begin{bmatrix} y(k) & w(k) \end{bmatrix}^{\mathrm{T}},$$

where,

$$B_s = \begin{bmatrix} 0 & 0 \\ 1 & T_s/m_3 \end{bmatrix}.$$

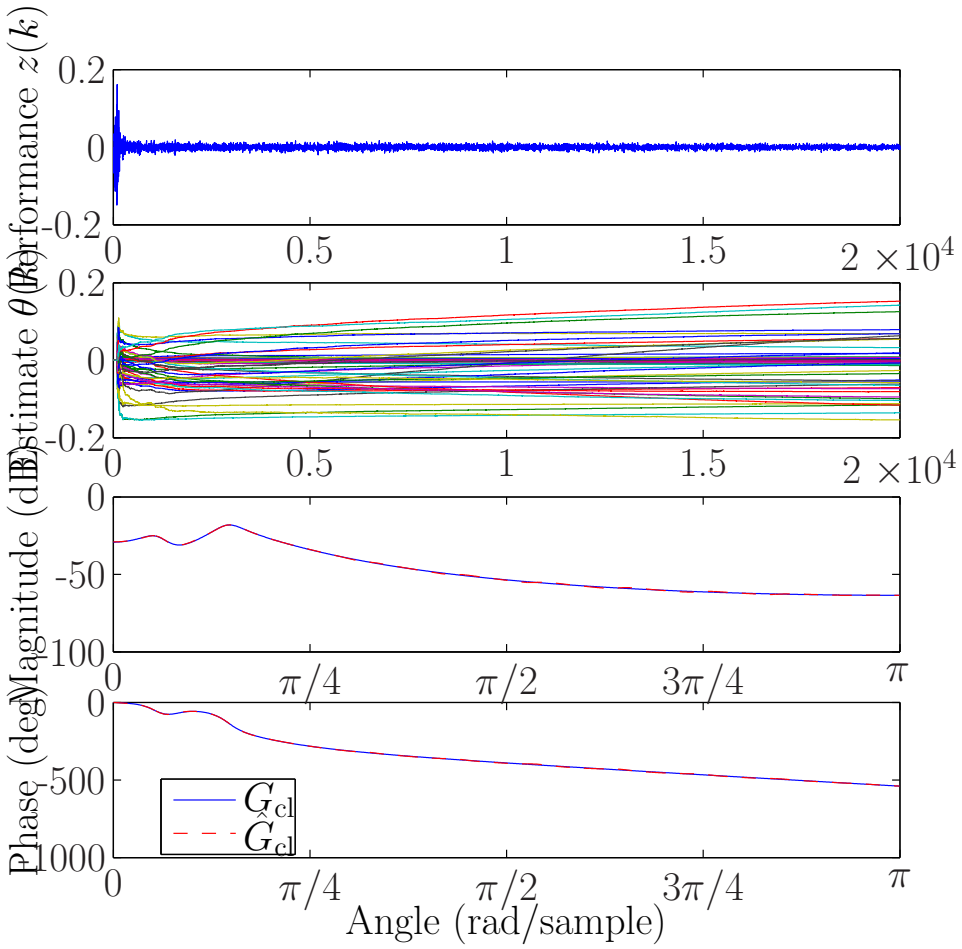


Figure 4.3: The RCO algorithm is turned on at $k = 100$ steps. The closed-loop frequency response of $\hat{G}_{\rm cl}$ is indistinguishable from the frequency response of $G_{\rm cl}$.

Furthermore, we let $\mu_w = 0$ and $\sigma_w^2 = 5$. Since A, B, C are unknown, we choose $\hat{A}, \hat{B}, \hat{C}$ as in (4.49). For this example we take $n_{\rm c} = 20$, $\bar{\eta} = 0$, $\beta = 0.01$, and $\tilde{\mathcal{H}} = H_1$. Figure 4.5 shows that, as the SNR increases, the accuracy of the frequency response of $\hat{G}_{\rm cl}$ improves.

Example IV.3. (*Case 2, A, B and C known*). First, we investigate the effect of the amount of data on the identification of $G_{\rm cl}$ and $G_{\rm s}$ using Case 2 architecture when A, B, and C are known. For this example, $\mu_w = 0$, $\sigma_w^2 = 10$ and there is no noise. Furthermore, we let $n_{\rm c} = 12$, $\bar{\eta} = 0$, $\beta = 0.01$, and $\tilde{\mathcal{H}} = H_3$. Figures 4.6 and 4.7 show that as the amount of data increases, the accuracy of the frequency responses of $\hat{G}_{\rm cl}$ and $\hat{G}_{\rm s}$ improve. Note that the frequency response of $\hat{G}_s$ cannot approximate

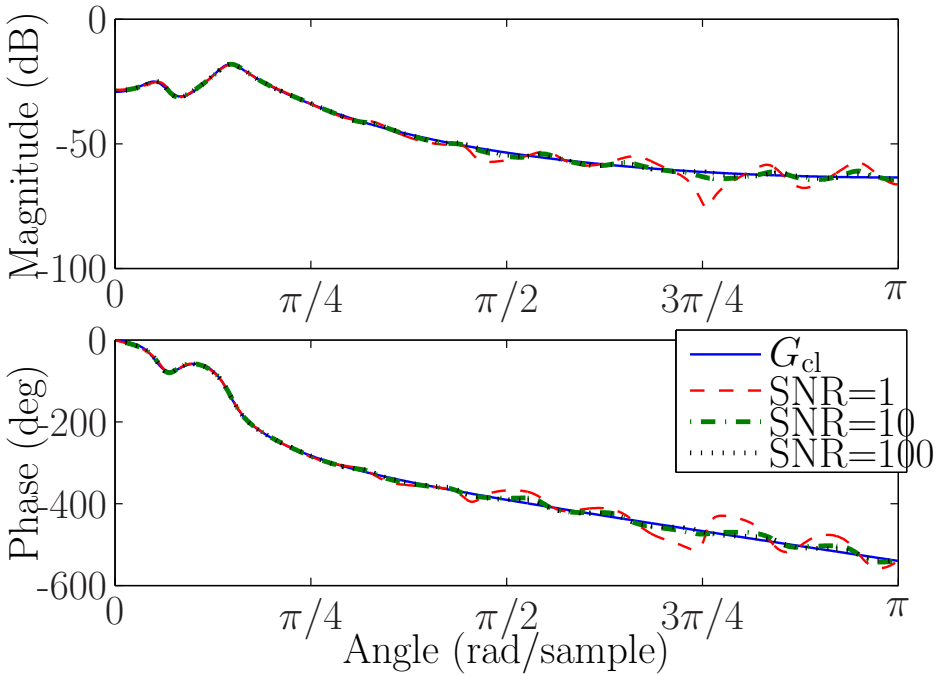


Figure 4.4: As the SNR increases, the accuracy of the frequency response of $\hat{G}_{\mathrm{cl}}$ improves.

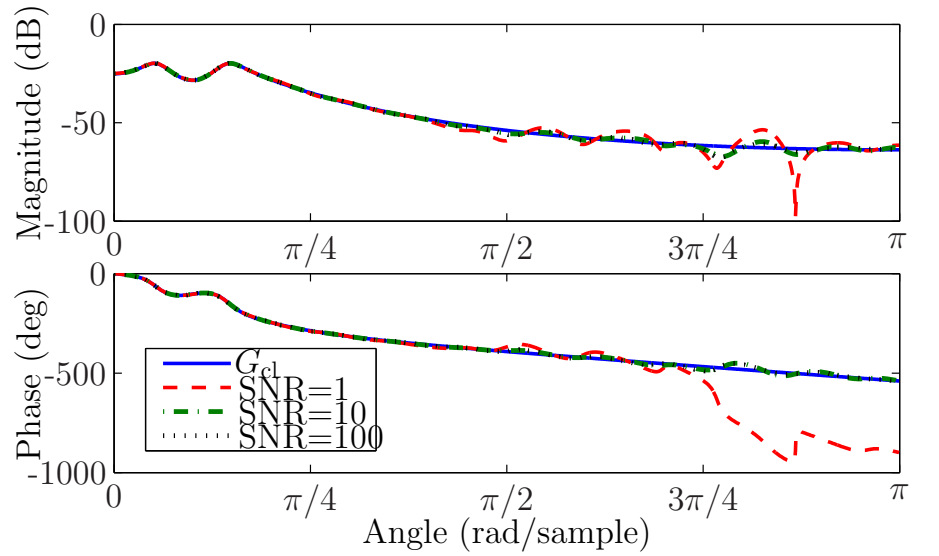


Figure 4.5: As the SNR increases, the frequency response of $\hat{G}_{\mathrm{cl}}$ more closely approximates the frequency response of G_{cl}.

G_s above 0.75 radians/sample because the transfer function that multiplies G_s in G_{cl} rolls off above this frequency. Next, we investigate the effect of SNR on Case 2 architecture when A, B, and C are known. The parameters are the same as in the

previous example. Figures 4.8 and 4.9 show that as the SNR increases, the accuracy of the frequency responses of $\hat{G}_{\text{cl}}$ and $\hat{G}_{\text{s}}$ improve.

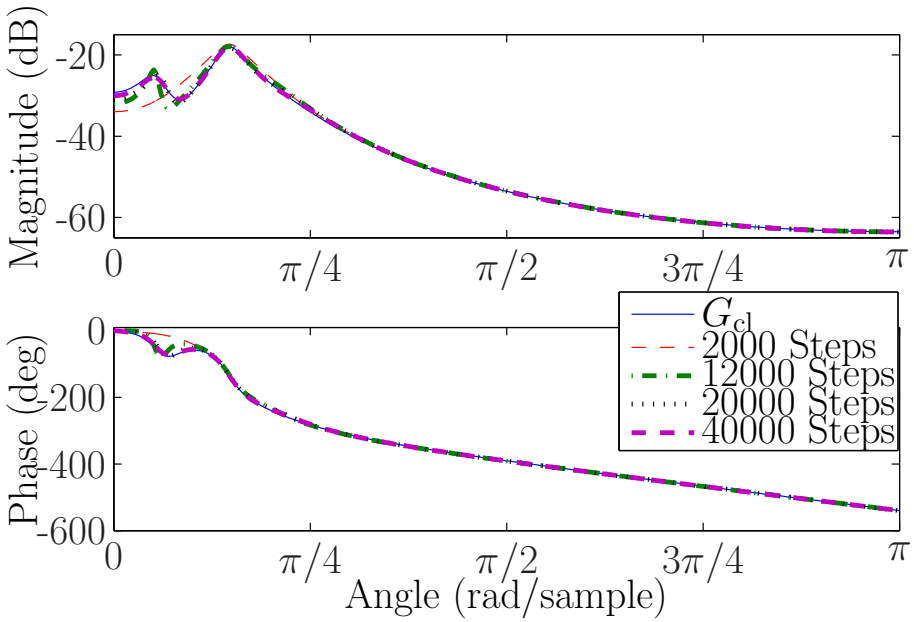


Figure 4.6: As the amount of data increases, the frequency response of $\hat{G}_{\text{cl}}$ more closely approximates the frequency response of G_{cl}.

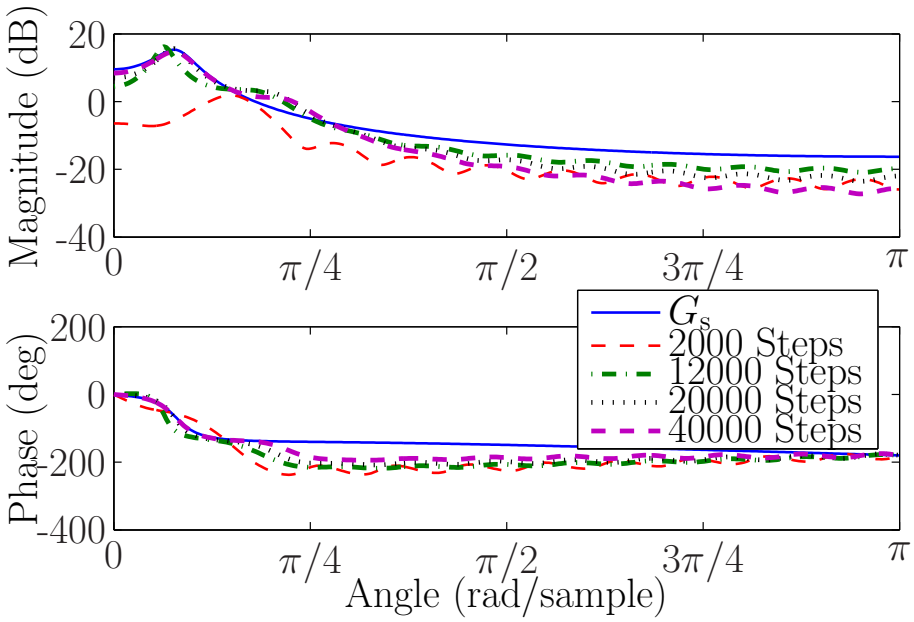


Figure 4.7: As the amount of data increases, the frequency response of $\hat{G}_{\text{s}}$ more closely approximates the frequency response of G_{s}.

Example IV.4. (*Case 2, A uncertain, B and C known, SNR=100*). In this example we investigate the effect of uncertainty in A. Uncertainty in A is introduced by

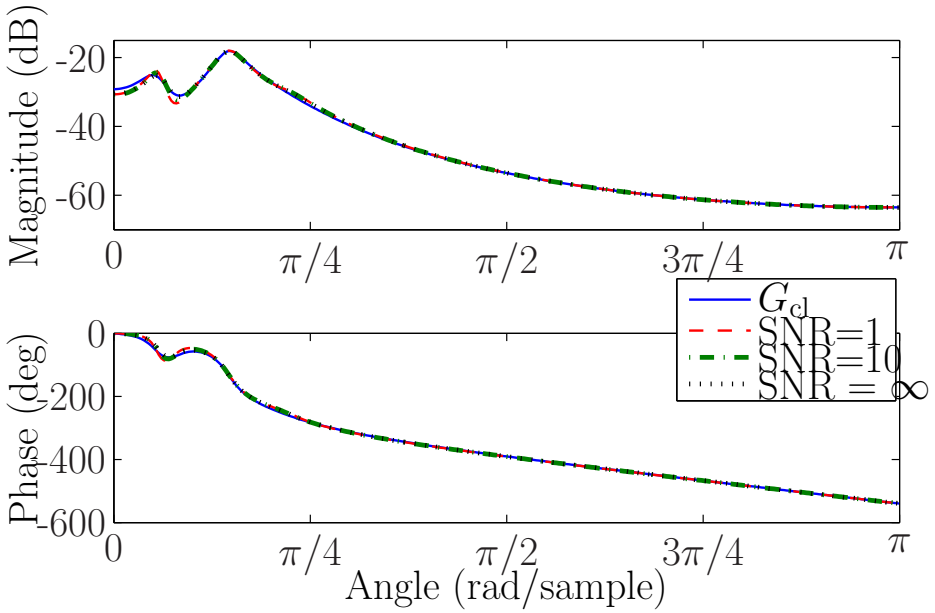


Figure 4.8: As the SNR increases, the frequency response of $\hat{G}_{\text{cl}}$ more closely approximates the frequency response of G_{cl}.

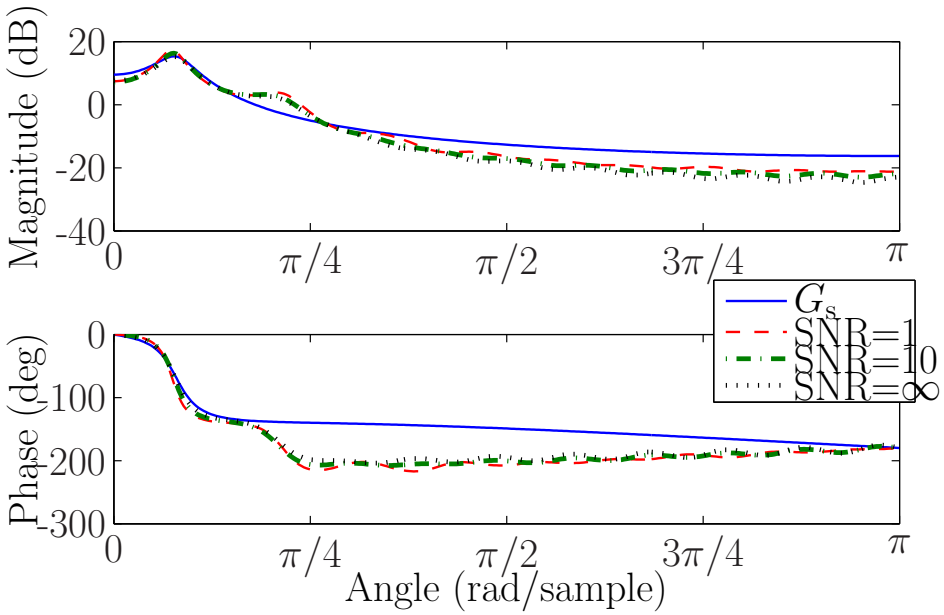


Figure 4.9: As the SNR increases, the frequency response of $\hat{G}_{\text{s}}$ more closely approximates the frequency response of G_{s}.

scaling the damping coefficient c_2 by an unknown scale factor α. Thus, $\hat{A}$ is obtained by replacing c_2 in A by αc_2. For this example, $\mu_w = 0$ and $\sigma_w^2 = 5$. Furthermore, we let $n_{\text{c}} = 12$, $\bar{\eta} = 0$, $\beta = 0.01$, and $\tilde{\mathcal{H}} = H_3$. Figures 4.10 and 4.11 show that as the uncertainty in A decreases (that is, α approaches 1), the frequency responses of $\hat{G}_{\text{cl}}$ and $\hat{G}_{\text{s}}$ more closely approximate the frequency responses of G_{cl} and G_{s}, respectively.

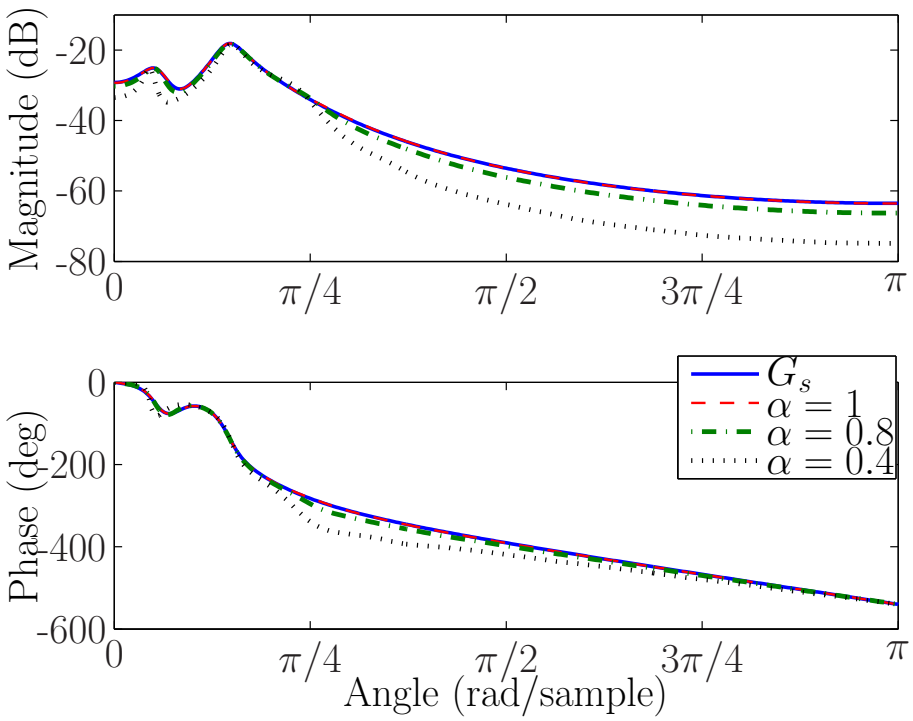


Figure 4.10: Estimate of $G_{\rm cl}$ with uncertain $\hat{c}_2 = \alpha c_2$. As α approaches 1, the frequency response of $\hat{G}_{\rm cl}$ more closely approximates the frequency response of $G_{\rm cl}$.

4.5 Conclusions

This chapter has focused on the problem of model refinement, where data are used to improve the accuracy of a subsystem model connected by feedback to a given main system model. In particular, the objective is system emulation, where the goal is to estimate a subsystem model in order to provide a combined system model that has improved accuracy relative to the main system alone. The inputs and outputs of the unknown subsystem are not assumed to be accessible, and thus standard system identification techniques are not applicable. We applied retrospective cost optimization, which reconstructs the input to the main system from the unknown subsystem. The main system may be well known or uncertain. In the latter case, there is no expectation that the estimated subsystem model approximates the unknown subsystem. However, if the main system is known exactly, then the estimated subsystem may

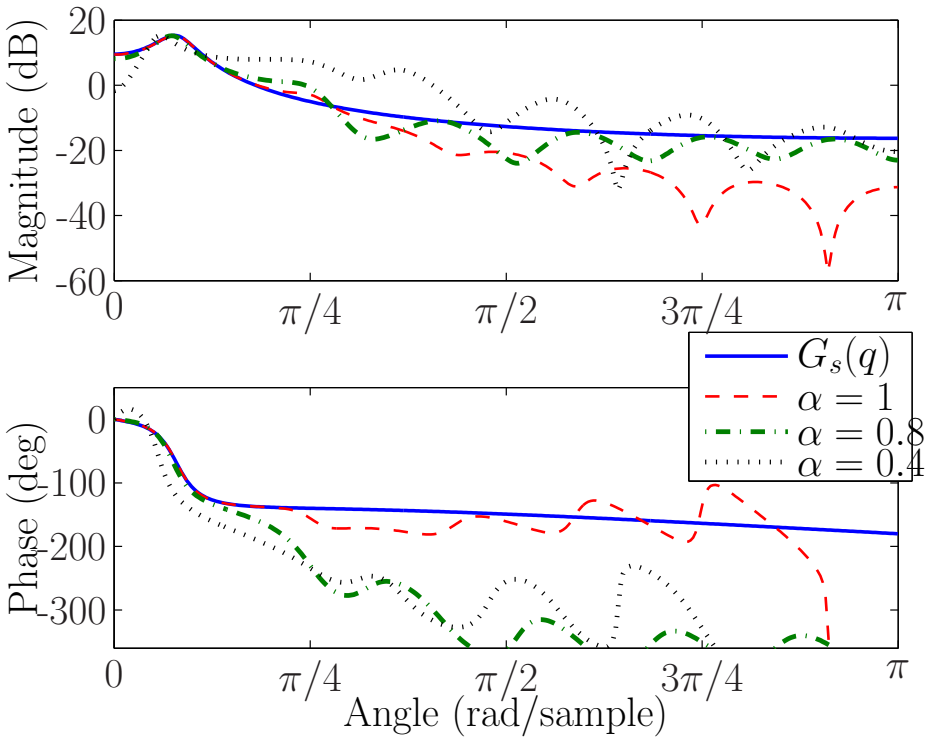


Figure 4.11: Estimate of G_s with uncertain $\hat{c}_2 = \alpha c_2$. As α approaches 1, the frequency response of $\hat{G}_s$ more closely approximates the frequency response of G_s.

provide a useful estimate of the unknown subsystem. Several numerical examples were used to illustrate the approach. The performance of the algorithm was assessed in terms of the closeness of the frequency response plots. The ultimate goal of this work is to provide a tool that engineers and scientists can use to improve the accuracy of large-scale models and estimate unknown subsystems that are difficult to model due to the inaccessibility of their inputs and outputs.

CHAPTER V

Application of Model Refinement to the Ionosphere and Thermosphere

5.1 Introduction

Models serve a variety of purposes by capturing different phenomena at varying levels of resolution. High-resolution models are desirable when the goal is to understand scientific phenomena or assimilate data, whereas a coarser model may be preferable when the goal is to capture critical details in an efficient manner, for example, for fast prediction or control. Consequently, the fidelity of a model must be gauged against its intended usage.

Figure 5.1: The goal of this work is to use data to improve the accuracy of an initial model. In other words, initial model + data = improved model.

Most models are constructed from collections of interconnected subsystem models, which in turn are based on a combination of physical laws and empirical observations.

For example, the core of a model might be the Navier-Stokes or magnetohydrodynamic equations, while source terms, such as chemistry, heating, and friction, may be modeled using either first-principles submodels or empirical relations that have different levels of accuracy, self-consistency, and complexity. Physical laws embody first-principles knowledge, whereas empirical observations may include relations that are based on the statistical analysis of data, for example, regression. Physics can provide the backbone of a model, while empirical relations can flesh out details such as sub-grid-scale phenomena that are beyond the ability of analytical modeling.

When input-output data are available, an empirical model can be constructed by means of system identification methods. In particular, techniques for constructing linear dynamic models that relate measured inputs to measured outputs are well developed [4, 51, 61]. A challenging extension is to develop methods for nonlinear system identification. Since nonlinear models can have a vast range of structures, the problem of nonlinear system identification requires the choice of a suitable model structure as well as an algorithm that uses data to tune the parameters of the model. Candidate model structures range from unstructured black-box models, such as neural networks, to gray-box and white-box models, where some or all of the structure of the model is specified [62, 63, 64, 65]. The chosen model structure is assumed to be identifiable from the available measurements, which means that its independent parameters can be unambiguously estimated from sufficiently persistent data.

The ability to identify a component or subsystem of a system depends on *accessibility,* which refers to the availability of the inputs and outputs of the subsystem. The highest degree of accessibility arises when both the input and output of the unknown subsystem are measured. In the case of Hammerstein and Wiener gray-box model structures, a static nonlinear mapping is cascaded with a dynamic linear subsystem, but the intermediate signal is assumed to be unavailable for identification [66]. If a static or dynamic subsystem is completely inaccessible in the sense that neither its in-

put nor its output is measured, then the identification problem becomes significantly more challenging.

The uncertain physics of a subsystem may range from the simplest case of an unknown parameter (such as a diffusion constant), to a multivariable spatially dependent static mapping (such as a conductivity tensor or boundary conditions), to a fully dynamic relationship among multiple variables (such as reaction kinetics). The difficulty of identifying these phenomena from empirical data depends on the accessibility of the subsystem, while the ability to use data to update a model despite limited accessibility is the goal of model refinement.

Model refinement begins with an initial model, which may incorporate both physical laws and empirical observations. The components of the initial model may have varying degrees of fidelity, reflecting knowledge or ignorance of the relevant physics as well as the availability of data. With this initial model as a starting point, the goal is to use additional measurements to refine the model. Components of the model that are poorly modeled can be updated, thereby resulting in a higher fidelity model, as shown in Figure 5.1. This problem is variously known as model correction, empirical correction, model refinement, model calibration, or model updating, and relevant literature includes [6, 8, 10, 7] on finite-element modeling, [67, 68, 69] on meteorology, [9] on feedback control, as well as algorithms [11, 53, 54] with application to health monitoring [55, 56]. Model refinement is thus a specialized version of identification, which is typically concerned with the construction of a model of the entire system.

When cast in the form of a block diagram, the model refinement problem has the form of an adaptive control system [11, 53, 54, 55, 70]. This resemblance suggests that adaptive control methods may be effective for tackling the model refinement problem. To do this, we require techniques for adaptive control that are sufficiently general and computationally tractable to address the features of large-scale physically meaningful applications. We thus apply the retrospective-cost adaptive control (RCAC)

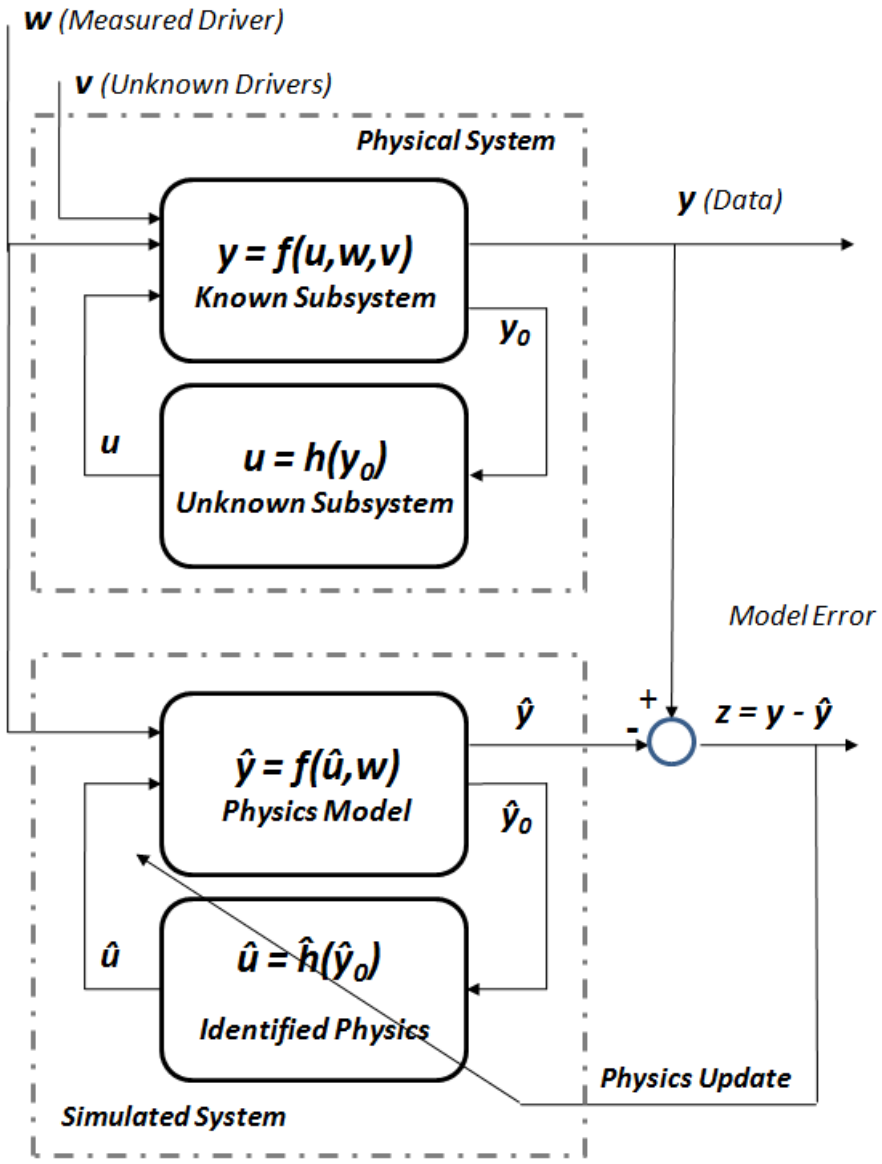


Figure 5.2: This block diagram illustrates the model refinement problem, where the goal is to identify the "Unknown Subsystem" of the "Physical System." By depicting this problem as a block diagram, it becomes evident that the model refinement problem is equivalent to a problem of adaptive command following.

technique [48, 50, 49], which differs from standard adaptive control approaches in several ways. Specifically, RCAC requires minimal modeling information concerning

the known portion of the system and is applicable to a wide range of adaptive control problems, including stabilization, command following, disturbance rejection, and model following. RCAC utilizes a surrogate cost function that entails a closed-form quadratic (and thus convex) optimization step. The controller update requires information about only the zeros of the system; no information about the poles is needed. Furthermore, the control update requires knowledge of only the nonminimum-phase zeros of the system. For model refinement, the relevant adaptive control problem is adaptive disturbance rejection, where the "disturbance" to be rejected is the unknown external excitation signal. Model refinement based on RCAC is called *adaptive model refinement*.

In this chapter we formulate adaptive model refinement for linear systems. We then demonstrate the method on a linear numerical example as well as on an experimental setup. Next, we apply adaptive model refinement to a first-principles model of the ionosphere and thermosphere. Specifically, we use the Global Ionosphere Thermosphere Model (GITM) [71] to provide a known initial model. We then use data from a "truth model" version of GITM in order to refine the initial model. Although the techniques developed in [48, 50] apply to linear systems, this chapter shows that model refinement based on RCAC can be effective for large-scale nonlinear systems such as GITM. Additional relevant literature on retrospective cost optimization includes [72, 73, 74, 31, 28, 75, 76, 77].

GITM is a 3-dimensional spherical (global Earth) code that solves the Navier-Stokes equations for the thermosphere. GITM is different from other models of the atmosphere [78, 79, 80] in that it solves the full vertical momentum equation instead of assuming that the atmosphere is in hydrostatic equilibrium, where the pressure gradient is balanced by gravity. While this assumption is valid for the majority of the atmosphere, in the auroral zone, where significant energy is dumped into the thermosphere on short time-scales, vertical accelerations often occur. This heating

causes strong vertical winds that can significantly lift the atmosphere [81].

The grid structure within GITM is fully parallel and covers the entire surface of the Earth by using a block-based two-dimensional domain decomposition in the horizontal coordinates [82]. The number of latitude and longitude blocks can be specified at run time in order to modify the horizontal resolution. GITM has been run on up to 256 processors with a resolution as fine as 0.31° latitude by 2.5° longitude over the entire globe with 50 vertical levels, covering a vertical domain from 100 km to roughly 600 km [71]. This flexibility can be used to validate consistency by running model refinement at various levels of resolution.

First principles models of the atmosphere are strongly influenced by unknowns such as thermal conductivity coefficients and cooling processes. These effects cannot be directly measured at each altitude, and thus they are inaccessible. We identify these subsystems, which are assumed to be unknown or uncertain, using data from simulated satellites on orbit. We then correct the uncertain model to demonstrate the feasibility of implementing the adaptive model refinement technique. A preliminary version of some of the results in this chapter have appeared in the conference papers [83, 84].

In Section 2, we describe the adaptive model refinement problem for subsystem identification. In Section 3, a linear problem formulation is cast using transfer functions to represent the initial model and the unknown subsystem. In Section 4, we present retrospective cost optimization as a method for obtaining an estimate of the unknown subsystem. In Section 5, the technique is demonstrated on linear numerical examples, as well as an experimental example. In Section 6, we apply the technique to a nonlinear example, specifically, parameter estimation and dynamic subsystem identification in the ionosphere and thermosphere.

5.2 Adaptive Model Refinement for Subsystem Identification

Figure 5.2 shows a block diagram of the model refinement problem. Each block is labeled to denote its uncertainty status. The blocks labeled "Known Subsystem" and "Unknown Subsystem" represent the physical system, whose inputs include known and unknown inputs, known as drivers. These subsystems are connected through feedback, which captures the fact that each subsystem impacts the other. The majority of the dynamics of the system are assumed to be included in the "Known Subsystem", while the "Unknown Subsystem" includes static or dynamic maps that are poorly known. Both the input y_0 and the output u of the "Unknown Subsystem" are assumed to be unavailable, and thus this subsystem is not accessible. The objective is to use data to better understand the "Unknown Subsystem". The unknown drivers v, which are unmeasured excitations to the system, may corrupt the estimated model of the unknown subsystem, despite the model error signal z tending to zero.

The lower part of the diagram in Figure 5.2 constitutes the "Simulated System." The "Physics Model," which is implemented in computation, captures the dynamics of the "Known Subsystem" and serves as the initial model. The "Physics Model" is interconnected by feedback with the block labeled "Identified Physics," which is refined by the "Physics Update" procedure, which is denoted by the diagonal arrow. The "Physics Update" is a tuning procedure that recursively identifies the unknown physics as data become available to provide a model of the "Unknown Subsystem". This tuning procedure is driven by the model-error signal z, which is the difference between the data y from the "Physical System" and the computed output $\hat{y}$ of the "Simulated System."

5.3 Linear Problem Formulation

From Figure 5.2, we consider a transfer function representation of the known subsystem $y = f(u, w)$, which is modeled by

$$\begin{bmatrix} y \\ y_0 \end{bmatrix} = \begin{bmatrix} G_{wy} & G_{uy} \\ G_{wy_0} & G_{uy_0} \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix} = G \begin{bmatrix} w \\ u \end{bmatrix}, \tag{5.1}$$

where G is the known initial model, y is the output data, w is the measured input signal, y_0 is the input to the unknown subsystem h, and u is the output of h. Furthermore, $u = h(y_0)$ is represented by the transfer function

$$\begin{aligned} u &= G_\Delta \begin{bmatrix} y_0 \\ w \end{bmatrix} \\ &= \begin{bmatrix} G_{\Delta,y_0} & G_{\Delta,w} \end{bmatrix} \begin{bmatrix} y_0 \\ w \end{bmatrix} \\ &= G_{\Delta,y_0} y_0 + G_{\Delta,w} w. \end{aligned} \tag{5.2}$$

We stress that h is not accessible, that is, measurements of the signals u and y_0 are not available, and thus G_Δ cannot be identified using standard techniques. From (5.1) and (5.2), we obtain the closed-loop transfer function from w to y given by

$$y = \left[G_{wy} + G_{uy}\left(G_{\Delta,y_0}\left[I - G_{uy_0}G_{\Delta,y_0}\right]^{-1}\left[G_{wy_0} + G_{uy_0}G_{\Delta,w}\right] + G_{\Delta,w}\right)\right] w. \tag{5.3}$$

The goal is to estimate the unknown subsystem $\hat{G}_\Delta$ such that the simulated system

$$\hat{y} = \left[G_{wy} + G_{uy}\left(\hat{G}_{\Delta,y_0}\left[I - G_{uy_0}\hat{G}_{\Delta,y_0}\right]^{-1}\left[G_{wy_0} + G_{uy_0}\hat{G}_{\Delta,w}\right] + \hat{G}_{\Delta,w}\right)\right] w \tag{5.4}$$

matches the physical system, that is, the model error signal

$$z = y - \hat{y} \tag{5.5}$$

is small.

To identify the feedback term G_Δ using the given initial model G, we use an adaptive feedback model structure to identify $\hat{G}_\Delta = [\hat{G}_{\Delta,y_0}\ \hat{G}_{\Delta,w}]$. To enforce model matching, we minimize the model error signal z in the presence of the measured signal w. In particular, we use RCAC in a disturbance-rejection architecture. The only signals available to RCAC are the measurement y, the simulated system inputs and outputs u, $\hat{y}$, $\hat{y}_0$, and the model error signal z.

5.4 Retrospective Cost Optimization

To model the "Identified Physics", consider a strictly proper time-series model of order n_c, such that (5.2) is given by

$$u(k) = \sum_{i=1}^{n_c} M_i(k)u(k-i) + \sum_{i=0}^{n_c} N_i(k)\hat{y}_0(k-i) + \sum_{i=0}^{n_c} L_i(k)w(k-i), \tag{5.6}$$

where, for all $i = 1, \ldots, n_c$, $M_i : \mathbb{N} \to \mathbb{R}^{l_u \times l_u}$, $N_i : \mathbb{N} \to \mathbb{R}^{l_u \times l_y}$ and $L_i : \mathbb{N} \to \mathbb{R}^{l_u \times l_w}$ are determined by the adaptive law presented below. Equation (5.6) can be expressed as

$$u(k) = \theta(k)\phi(k), \tag{5.7}$$

where

$$\theta(k) \triangleq \begin{bmatrix} N_1(k) & \cdots & N_{n_c}(k) & L_1(k) & \cdots & L_{n_c}(k) & M_1(k) & \cdots & M_{n_c}(k) \end{bmatrix}$$

and

$$\phi(k) \triangleq \Big[\; \hat{y}_0^{\mathrm{T}}(k-1) \;\; \cdots \;\; \hat{y}_0^{\mathrm{T}}(k-n_{\mathrm{c}}) \;\; w^{\mathrm{T}}(k-1) \;\; \cdots \;\; w^{\mathrm{T}}(k-n_{\mathrm{c}}) \;\; u^{\mathrm{T}}(k-1) \;\; \cdots \;\; u^{\mathrm{T}}(k-n_{\mathrm{c}}) \; \Big]^{\mathrm{T}} \in \mathbb{R}^{n_{\mathrm{c}}(l_u+l_y+l_w)}.$$

Next, we represent (5.5) as the time-series model from u and w to z given by

$$z(k) = y(k) - \left[\sum_{i=1}^{n} -\alpha_i y(k-i) + \sum_{i=d}^{n} \beta_i u(k-i) + \sum_{i=0}^{n} \gamma_i w(k-i) \right], \tag{5.8}$$

where $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$, $\beta_d, \ldots, \beta_n \in \mathbb{R}^{l_z \times l_u}$, $\gamma_0, \ldots, \gamma_n \in \mathbb{R}^{l_z \times l_w}$, and the relative degree d is the smallest non-negative integer i such that the ith Markov parameter of G_{yu} is nonzero, where the Markov parameters are the components of the system's impulse response [4, 51].

Next, we define the *retrospective performance*

$$\hat{z}(\hat{\theta}, k) \triangleq y(k) - \left[\sum_{i=1}^{n} -\alpha_i y(k-i) + \sum_{i=d}^{n} \beta_i \theta(k-i) \phi(k-i) + \sum_{i=0}^{n} \gamma_i w(k-i) + \sum_{i=d}^{\nu} \bar{\beta}_i \left[\hat{\theta} - \theta(k-i) \right] \phi(k-i) \right], \tag{5.9}$$

where $\nu \geq d$, $\hat{\theta} \in \mathbb{R}^{l_u \times (n_c(l_y+l_u))}$ is an optimization variable used to derive the adaptive law, and $\bar{\beta}_d, \ldots, \bar{\beta}_\nu \in \mathbb{R}^{l_z \times l_u}$. RCAC uses a retrospective performance measure, in which the performance measurement is modified based on the difference between the actual past control inputs and the recomputed past control inputs. The parameters ν and $\bar{\beta}_d, \ldots, \bar{\beta}_\nu$ must capture the information included in the first nonzero Markov parameter and the nonminimum-phase zeros from u to z [58]. In this chapter, we let $\bar{\beta}_d, \ldots, \bar{\beta}_\nu$ denote Markov parameters of the transfer function from u to z. Alternative choices of the parameters ν and $\bar{\beta}_d, \ldots, \bar{\beta}_\nu$ are discussed in [58].

Next, subtracting (5.8) from (5.9) yields

$$\hat{z}(\hat{\theta}, k) = z(k) + \sum_{i=d}^{n} \beta_i \left[\hat{\theta} - \theta(k-i)\right] \phi(k-i). \tag{5.10}$$

Defining $\hat{\Theta} \triangleq \text{vec}\, \hat{\theta} \in \mathbb{R}^{n_c l_u (l_y + l_u + l_w)}$ and $\Theta(k) \triangleq \text{vec}\, \theta(k) \in \mathbb{R}^{n_c l_u (l_y + l_u + l_w)}$, it follows that

$$\begin{aligned} \hat{z}(\hat{\Theta}, k) &= z(k) + \sum_{i=d}^{n} \Phi_i^{\mathrm{T}}(k) \left[\hat{\Theta} - \Theta(k-i)\right] \\ &= z(k) - \sum_{i=d}^{n} \Phi_i^{\mathrm{T}}(k) \Theta(k-i) + \Psi^{\mathrm{T}}(k) \hat{\Theta}, \end{aligned} \tag{5.11}$$

where, for $i = d, \ldots, n$,

$$\Phi_i(k) \triangleq \phi(k-i) \otimes \beta_i^{\mathrm{T}} \in \mathbb{R}^{(n_c l_u (l_y + l_u + l_w)) \times l_z},$$

where "vec" denotes the column-stacking operator, $\otimes$ represents the Kronecker product [85], and

$$\Psi(k) \triangleq \sum_{i=d}^{n} \Phi_i(k).$$

We now consider the *retrospective cost function*

$$J(\hat{\Theta}, k) \triangleq \hat{z}^{\mathrm{T}}(\hat{\theta}, k) R_1(k) \hat{z}(\hat{\theta}, k) + \text{tr} \left[R_2(k) \left(\hat{\theta} - \theta(k)\right)^{\mathrm{T}} R_3(k) \left(\hat{\theta} - \theta(k)\right) \right], \tag{5.12}$$

where $R_1(k) \triangleq I_{l_z}$, $R_2(k) \triangleq \alpha(k) I_{n_c (l_y + l_u + l_w)}$, and $R_3(k) \triangleq I_{l_u \times l_u}$. Using Kronecker algebra, (5.12) can be written as the quadratic form

$$J(\hat{\Theta}, k) = c(k) + b^{\mathrm{T}}(k) \hat{\Theta} + \hat{\Theta}^{\mathrm{T}} A(k) \hat{\Theta},$$

where

$$A(k) \triangleq \Phi(k)\Phi^{\mathrm{T}}(k) + \alpha(k)I,$$

$$b(k) \triangleq 2\Phi(k)\left[\sum_{i=d}^{n} \Theta^{\mathrm{T}}(k-i)\Phi_i(k) + z^{\mathrm{T}}(k)\right] - 2\alpha(k)\Theta(k),$$

$$c(k) \triangleq \left[\sum_{i=d}^{n} \Phi_i^{\mathrm{T}}(k)\Theta(k-i)\right] R_1(k) \left[\sum_{i=d}^{n} \Theta^{\mathrm{T}}(k-i)\Phi_i(k)\right] + \mathrm{tr}\left[R_2(k)\theta^{\mathrm{T}}(k)R_3(k)\theta(k)\right].$$

Since $A(k)$ is positive definite, $J(\hat{\Theta}, k)$ has the strict global minimizer

$$\hat{\theta} = \frac{1}{2}\mathrm{vec}^{-1}(A(k)^{-1}b(k)). \tag{5.13}$$

The gain update law is to set $\theta(k+1)$ to the global minimizer (5.13), that is,

$$\theta(k+1) = \hat{\theta}. \tag{5.14}$$

The coefficients of the time series (5.6) given by (5.14) contain information about the unknown subsystem, such as its poles, zeros, time constants, and frequency response.

RCAC requires the selection of several parameters. Specifically, n_{c} is the estimated order of the unknown subsystem, while ν is the number of Markov parameters obtained from the known model. The adaptive update law (5.14) is based on the quadratic cost function (5.12), which involves the time-varying weighting parameter $\alpha(k) > 0$, referred to as the *learning rate* since it affects the convergence speed of the adaptive model refinement algorithm.

The methodology for choosing these parameters is as follows. For dynamic subsystem identification, the subsystem order n_c is typically unknown. In this case, it is convenient to overestimate the subsystem order. For parameter estimation, choosing $n_c = 0$ is a natural choice in (5.6), since the resulting G_Δ is static. The number ν of Markov parameters is usually chosen to be 1; however, a larger value is typically

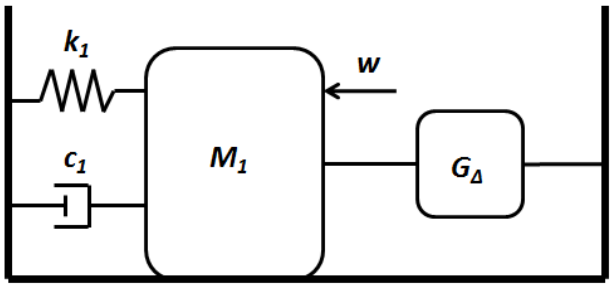


Figure 5.3: A single-degree-of-freedom mass-spring-damper system connected to an unknown impedance.

needed if nonminimum-phase zeros are present in the initial model [49].

5.5 Linear Examples

5.5.1 Dynamic Subsystem Estimation

Consider the mass-spring-damper structure shown in Figure 5.3 modeled by

$$m_1\ddot{q} + c_1\dot{q} + k_1 q = w, \tag{5.15}$$

where m_1, c_1, k_1 are the known mass, damping, and stiffness, respectively, and w is a force input. As shown in Figure 5.3, the mass is also connected to an unknown impedance G_Δ, which applies force to the mass in response to the velocity of the mass. We obtain the state space representation of the known subsystem

$$\begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = A_c \begin{bmatrix} q \\ \dot{q} \end{bmatrix} + B_c u + D_{1,c} w, \tag{5.16}$$

$$y = C \begin{bmatrix} q \\ \dot{q} \end{bmatrix}, \tag{5.17}$$

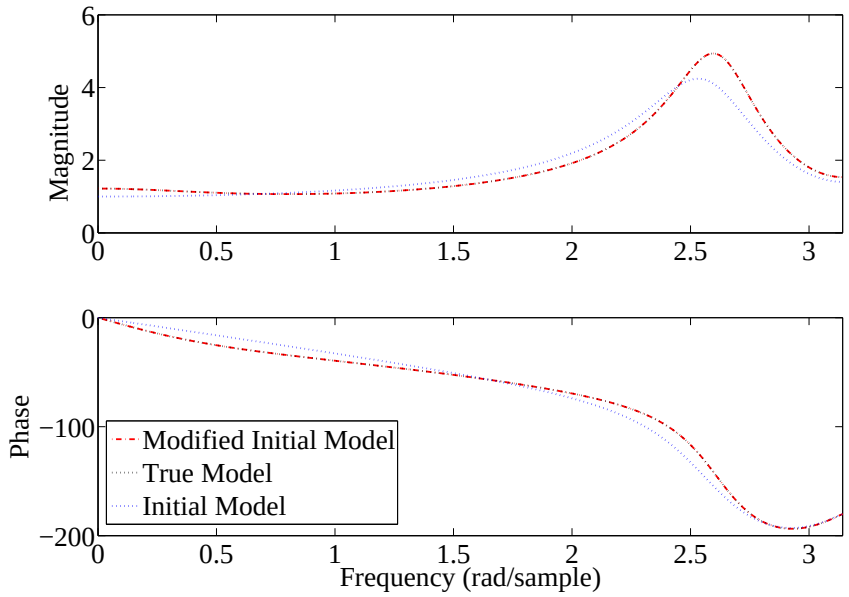
where q and $\dot{q}$ are the position and velocity, respectively, of the mass, and

$$A_c = \begin{bmatrix} 0 & 1 \\ -\frac{k_1}{m_1} & -\frac{c_1}{m_1} \end{bmatrix}, \quad B_c = D_{1,c} = \begin{bmatrix} 0 \\ \frac{1}{m_1} \end{bmatrix}, \quad C = \begin{bmatrix} 0 & 1 \end{bmatrix}. \tag{5.18}$$

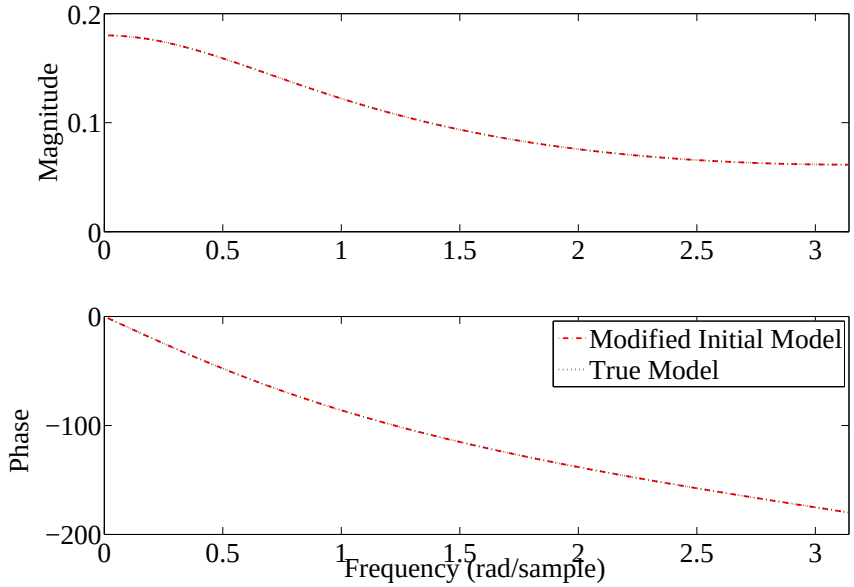
Finally, we write the system in transfer function form $G(s) = C(sI - A_c)^{-1}B_c$, where s is the Laplace transform variable and the closed-loop transfer function from w to y with the unknown impedance $u = G_\Delta y$ is $G_{\rm cl}(s) = \frac{G(s)}{1-G(s)G_\Delta(s)}$.

To demonstrate adaptive model refinement, we choose $m_1 = 1 \times 10^{-4}$, $k_1 = 1$, $c_1 = 5.0275 \times 10^{-4}$, and $G_\Delta(s) = \frac{(s+30)(s+60)}{(s+20)(s+50)(s+10)}$. Next, the continuous-time system (5.17) is converted to discrete time using $A = e^{A_c T_s}$ and $B = A_c^{-1}[A - I]B$, where $T_s = 0.1$ sec is the sample time. The resulting discrete time transfer function is $\bar{G}(z) = C(zI - A)^{-1}B$, where z is the Z-transform variable. Furthermore, $\bar{G}_\Delta(z)$ denotes the discretized transfer function of $G_\Delta(s)$.

Next, we choose $n_c = 5$, which is an overestimate of the order of $\bar{G}_\Delta$, $\alpha = 1$, and $\nu = 10$, that is, we use 10 Markov parameters of $\bar{G}(z)$. Figure 5.4(a) compares the frequency responses of the initial model and the closed-loop model consisting of the initial model and the subsystem estimate $\hat{\bar{G}}_\Delta$ of $\bar{G}_\Delta$. The difference between the initial model and the closed-loop model is reduced by including the estimate $\hat{\bar{G}}_\Delta$ of $\bar{G}_\Delta$; in fact, the estimated closed-loop model frequency response is almost identical to the frequency response of the "Physical System". Figure 5.4(b) compares the frequency responses of $\bar{G}_\Delta$ and $\hat{\bar{G}}_\Delta$.



(a)



(b)

Figure 5.4: (a) compares the frequency response of the initial model $\bar{G}(z)$, the closed loop $\bar{G}_{\text{cl}}(z)$, and the estimated closed loop using the identified unknown feedback $\hat{\bar{G}}_{\text{cl}}(z)$. (b) compares the frequency response of the unknown feedback and the identified feedback.

5.5.2 Static Parameter Estimation

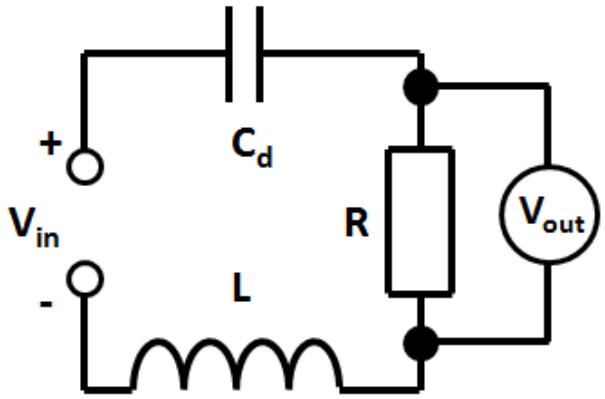


Figure 5.5: A series resistor-inductor-capacitor (RLC) circuit, where voltage is measured across the resistor. The inductance L and the capacitance C_d are assumed to be uncertain.

To demonstrate adaptive model refinement for parameter estimation, we consider the series resistor-inductor-capacitor (RLC) circuit shown in Figure 5.5 modeled by

$$L\ddot{x} + R\dot{x} + \frac{1}{C_{\mathrm{d}}}x = u, \tag{5.19}$$

where L, C_{d}, and R are the inductor, capacitor, and resistor values, respectively, and w is the input voltage. A state space representation of the circuit is given by

$$\begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{1}{LC_{\mathrm{d}}} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} u, \tag{5.20}$$

$$y = \begin{bmatrix} 0 & R \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \end{bmatrix}, \tag{5.21}$$

where q and $\dot{q}$ are the charge and current, respectively, of the circuit. Next, we write the state space equations for the circuit with an uncertainty ΔC_{d} in the capacitance

and ΔL in the inductance as

$$\begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{1}{(L+\Delta L)(C_{\rm d}+\Delta C_{\rm d})} & -\frac{R}{L+\Delta L} \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{L+\Delta L} \end{bmatrix} w, \tag{5.22}$$

$$y = \begin{bmatrix} 0 & R \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \end{bmatrix}, \tag{5.23}$$

where $u = G_{\Delta_q} q + G_{\Delta_{\dot{q}}} \dot{q} + G_{\Delta_w} w$. Estimates $\Delta\hat{C}_{\rm d}$ of $\Delta C_{\rm d}$ and $\Delta\hat{L}$ of ΔL can be obtained from the adaptive model refinement estimates $\hat{G}_{\Delta,q}$ of $G_{\Delta,q}$, $\hat{G}_{\Delta,\dot{q}}$ of $G_{\Delta,\dot{q}}$, and $\hat{G}_{\Delta,w}$ of $G_{\Delta,w}$ by means of

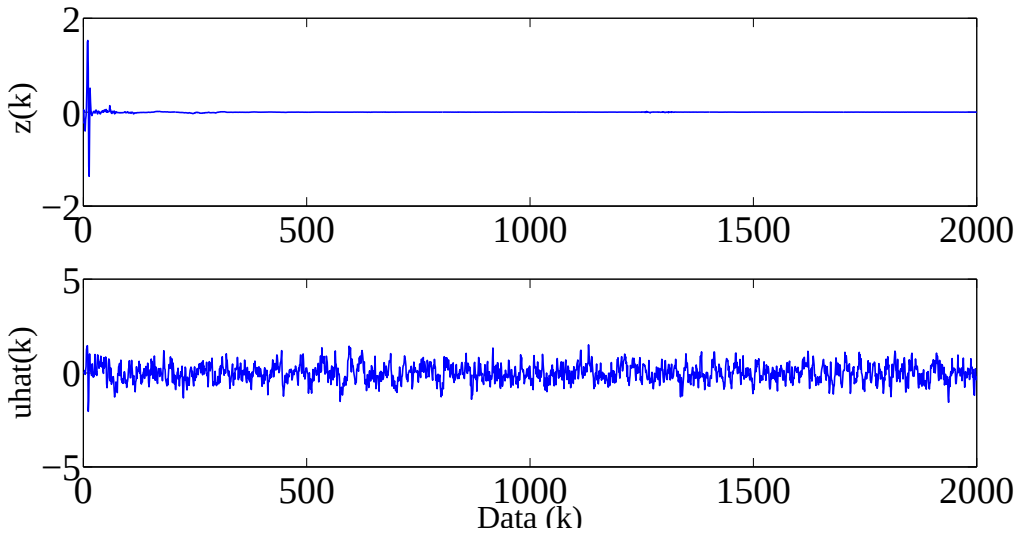
$$\Delta\hat{L} = \frac{L}{\hat{G}_{\Delta,w}} - L = \frac{-RL}{-R + \hat{G}_{\Delta,\dot{x}}} - L, \tag{5.24}$$

$$\Delta\hat{C}_{\rm d} = -L \left(\frac{-1}{C_{\rm d}} + \hat{G}_{\Delta,x} \right)^{-1} (L + \Delta L)^{-1} - C_{\rm d}. \tag{5.25}$$

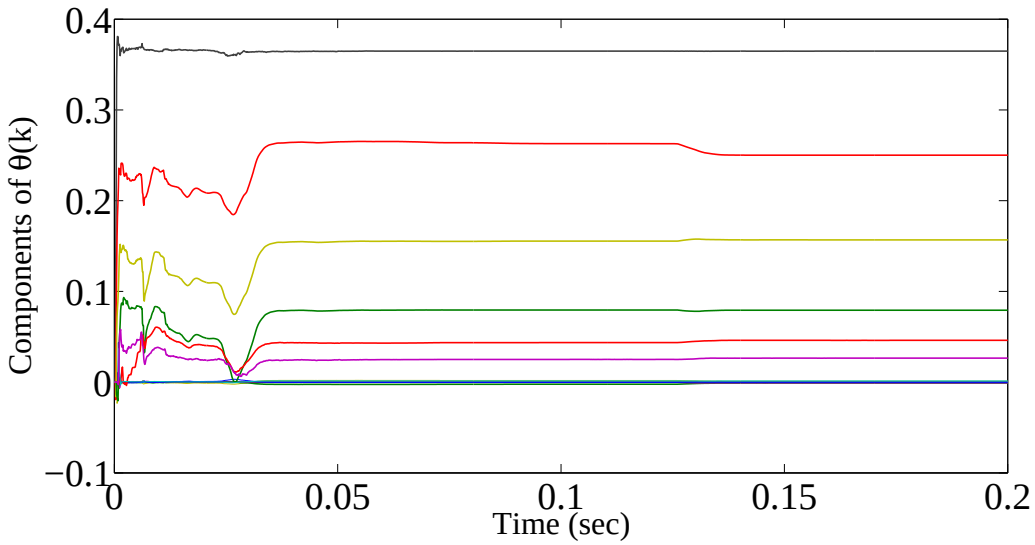
Next, we assemble a circuit with $R = 250\ \Omega$, $L + \Delta L = 55$ mH, and $C_{\rm d} + \Delta C_{\rm d} = 23.5\ \mu$F. We assume that we do not have knowledge of either $\Delta C_{\rm d}$ or ΔL, but only the initial estimates $C_{\rm d} = 1$ F and $L = 2\ \mu$H. The model (5.21) is similarly discretized. We drive the circuit using zero-mean, Gaussian white noise, and we measure the voltage across the resistor.

We implement RCAC to obtain estimates of the transfer functions $\hat{\hat{G}}_{\Delta,q}$, $\hat{\hat{G}}_{\Delta,\dot{q}}$, and $\hat{\hat{G}}_{\Delta,w}$. Figure 5.6(a) shows the history of the model error signal z. Figure 5.7(a) compares the frequency responses of the initial model, the actual system, and the refined model, in discrete time.

Next, we generate and record the driving signal and system output. Figure 5.8(a) shows the history of the model error signal z for the experimental setup. Figure 5.9(a) compares the discrete-time frequency responses of the initial model, the actual system, and the refined model, for the experimental setup.



(a)



(b)

Figure 5.6: (a) shows the history of the model error signal $z = \hat{y} - y$ and output $\hat{u}$ of the estimated subsystem for the simulated RLC circuit. (b) shows the components of the subsystem model $\theta(k)$ as functions of time. Note that z tends to zero as k becomes large, which indicates that the output of the simulated model approaches the output of the experimental circuit.

5.6 Application of Adaptive Model Refinement to Ionospheric Parameter Estimation

We now apply adaptive model refinement to a nonlinear example. We consider the problem of using upper atmospheric mass-density measurements, as can be ob-

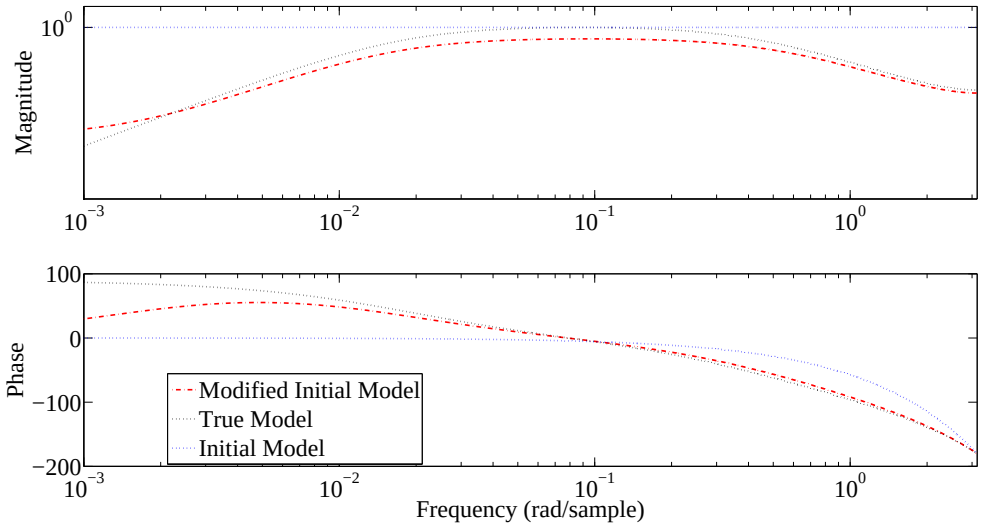
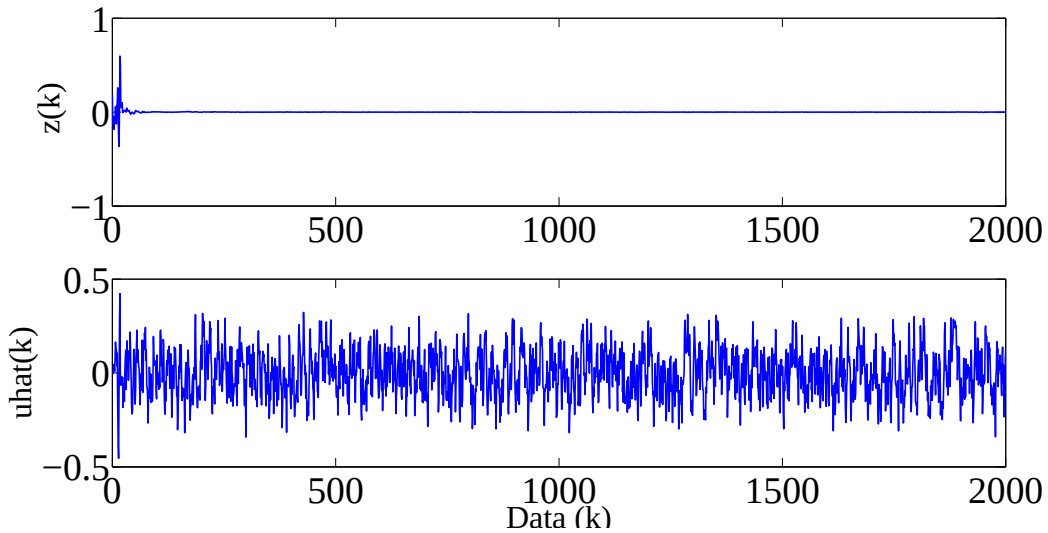


Figure 5.7: This plot compares the frequency responses of the initial model $\bar{G}$ (blue dotted line), the closed-loop $\bar{G}_{\text{cl}}$ (black dotted line), and the estimated closed-loop $\hat{\bar{G}}_{\text{cl}}$ (red dotted line) for the simulated RLC circuit.

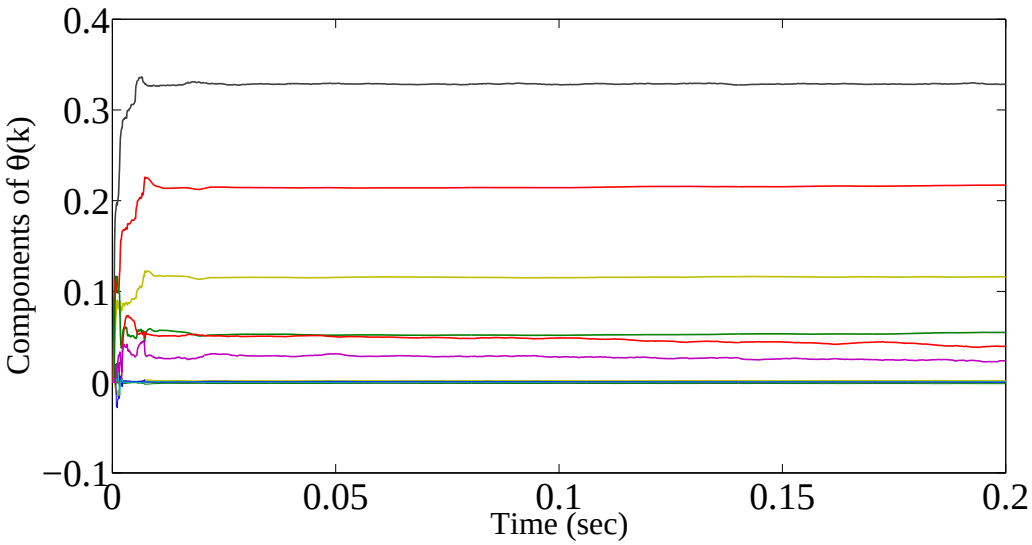
tained from a satellite, to estimate the thermal conductivity of the thermosphere. This problem is challenging due to the fact that we do not assume the availability of measurements that can serve as inputs or outputs to the "Unknown Subsystem," which models thermal conductivity. In other words, the objective of the identification in this application is a subsystem whose physics are inaccessible relative to the available measurements.

We use GITM to simulate the chemistry and fluid dynamics in a one-dimensional (1D) column in the ionosphere-thermosphere. The temperature structure of the thermosphere depends on various factors, such as the Sun's intensity in extreme ultraviolet (EUV) wavelengths, eddy diffusion in the lower thermosphere, radiative cooling of the O_2 and NO, frictional heating, and the thermal conductivity.

The structure of the thermal conductivity is $\lambda = AT^s$, where A and s are the thermal conductivity and rate coefficients, respectively. The thermal conductivity may depend on chemical constituents (e.g., N_2, O_2, and O). Uncertainty concerning



(a)



(b)

Figure 5.8: (a) shows the history of the model error signal $z = \hat{y} - y$ and the output $\hat{u}$ of the estimated subsystem for the experimental RLC circuit. (b) shows the components of the subsystem model $\theta(k)$ as functions of time. Note that z tends to zero as k becomes large, which indicates that the output of the simulated model approaches the output of the experimental circuit.

the values of A and s [86] can strongly influence the temperature structure. The need to estimate these coefficients is motivated by Figure 5.11 from [1], where published values of these coefficients are shown to vary depending on the reference source. For

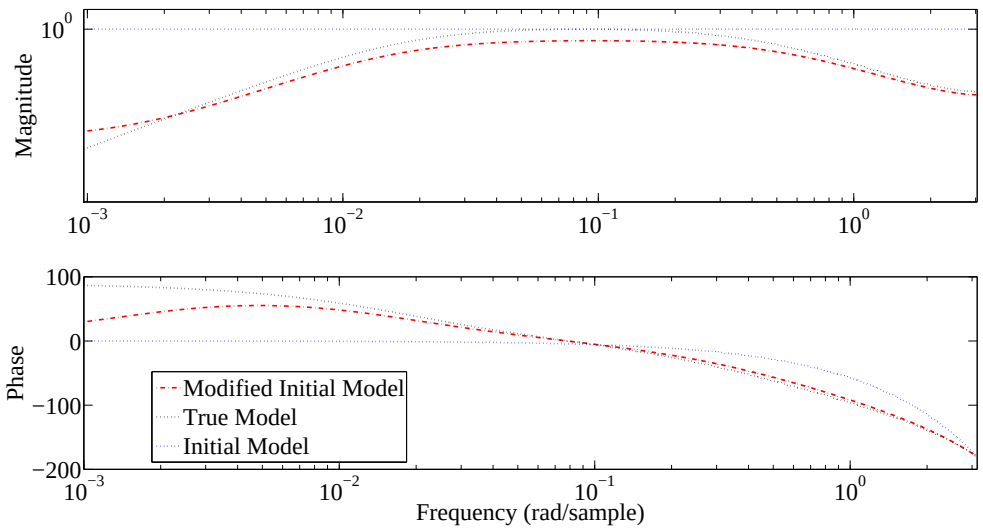


Figure 5.9: This plot compares the frequency responses of the initial model $\bar{G}$ (blue dotted line), the closed-loop $\bar{G}_{\text{cl}}$ (black dotted line), and the estimated closed-loop $\hat{\bar{G}}_{\text{cl}}$ (red dotted line) for the experimental RLC circuit.

illustration, we assume that the true value of A is the mean of the range of values, and we seek estimates of A that are within this range.

To estimate the unknown thermal conductivity coefficient A, we apply adaptive model refinement to simulated measurements of neutral mass density provided by 1D GITM. We do this by running a "truth model," from which we extract mass-density data at 400-km altitude, which is a typical altitude for satellites. The thermal conductivity coefficient is initialized to be zero, and its value is updated recursively. Figure 5.12 shows the evolution of the estimate $\hat{A}$ of the thermal conductivity A as more data become available. The estimate $\hat{A}$ is seen to converge to a neighborhood of the true value within about 0.6×10^4 data points.

To further illustrate the model refinement method, we now assume that both the thermal conductivity A and the rate coefficient s are unknown. The parameters A and s are initialized as zero, and are updated simultaneously and recursively. Figure 5.13 shows the update of the estimates. Both estimates converge to within a neighborhood

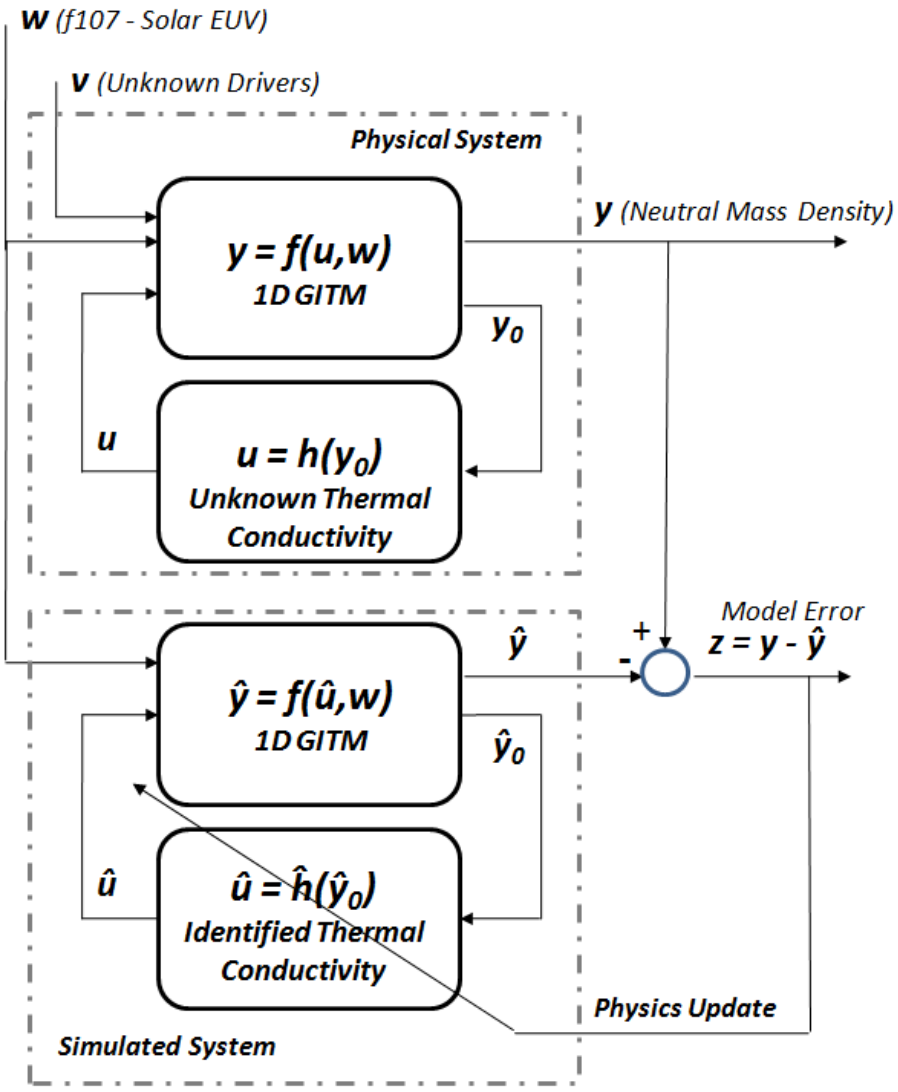


Figure 5.10: This block diagram for adaptive model refinement specializes Figure 5.2 to a model of the ionosphere-thermosphere. Simulated data are generated by using the 1D Global Ionosphere-Thermosphere Model (GITM), where the thermal conductivity is assumed to be unknown. The goal is to estimate the thermal conductivity by using measurements of the neutral mass density. This problem is challenging due to the low accessibility of the unknown physics relative to the available measurements w and y.

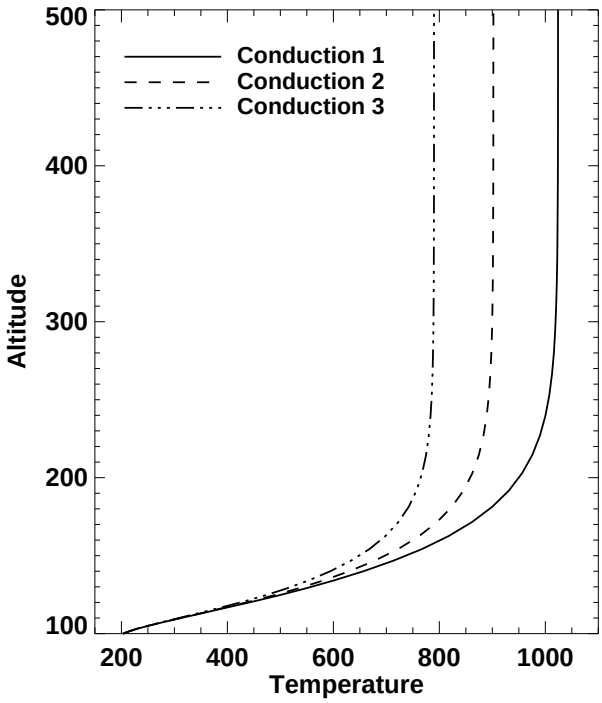


Figure 5.11: Steady-state globally averaged temperature structure using three published conductivity values [1].

of the true values within 0.6×10^5 data points.

The performance gains attributed to the refined parameters are shown in Figure 5.14. The upper figure shows the model error signal z for the GITM "truth" model and an initial GITM model whose thermal conductivity coefficient is set to zero. Within the simulated model, this value prevents energy deposited in one layer of the atmosphere from remaining in that layer. The lower plot of Figure 5.14 illustrates the reduction in model error obtained by including the identified coefficients, thereby accounting for the thermal conductivity of this species. The benefits of refining the GITM model are evident by the improvement in model accuracy as determined by z.

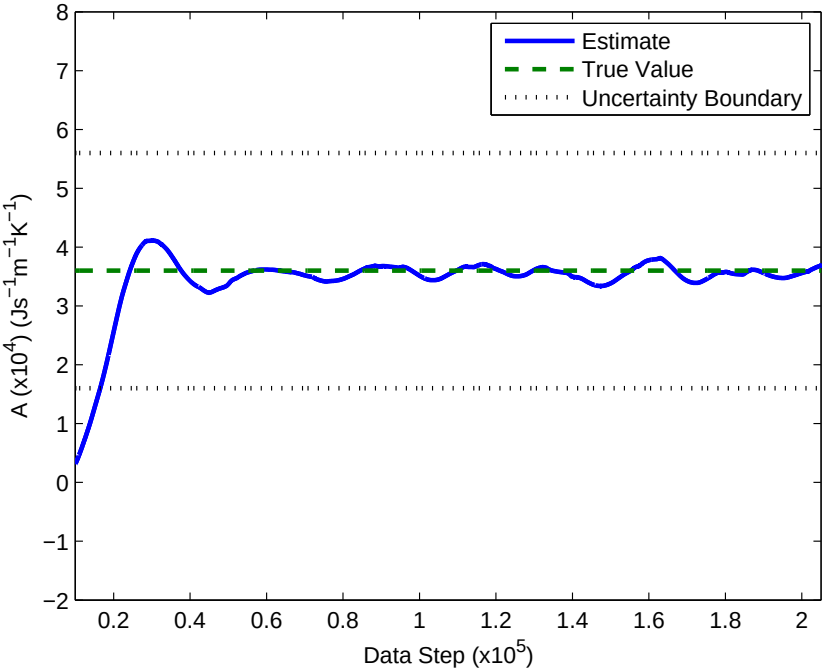


Figure 5.12: This plot shows the true and estimated thermal conductivity coefficient. The initial guess for the thermal conductivity is zero, while the actual thermal conductivity is set to be the mean of the range of uncertainty. The estimate $\hat{A}$ of A converges to a neighborhood of the true value of A within about 0.6×10^5 data points. The lack of final convergence is due to nonlinearities in the dynamics of the system. However, the oscillations are well within the uncertainty bounds, which reflect the range of published values for this coefficient.

5.7 Application of Adaptive Model Refinement to Ionospheric Dynamics Estimation

To illustrate adaptive model refinement in the case of an unknown dynamic subsystem, the NO radiative cooling is removed from GITM to provide an initial model, but is retained in GITM for the truth model. The goal is to reproduce the missing process. This is nontrivial since the functional form of the cooling is assumed to be unknown as are the dynamics. We assume only that something is missing from the energy equation, and that this is most likely a function of temperature. The dynamics of the cooling are estimated at three different altitudes, connecting the other altitudes

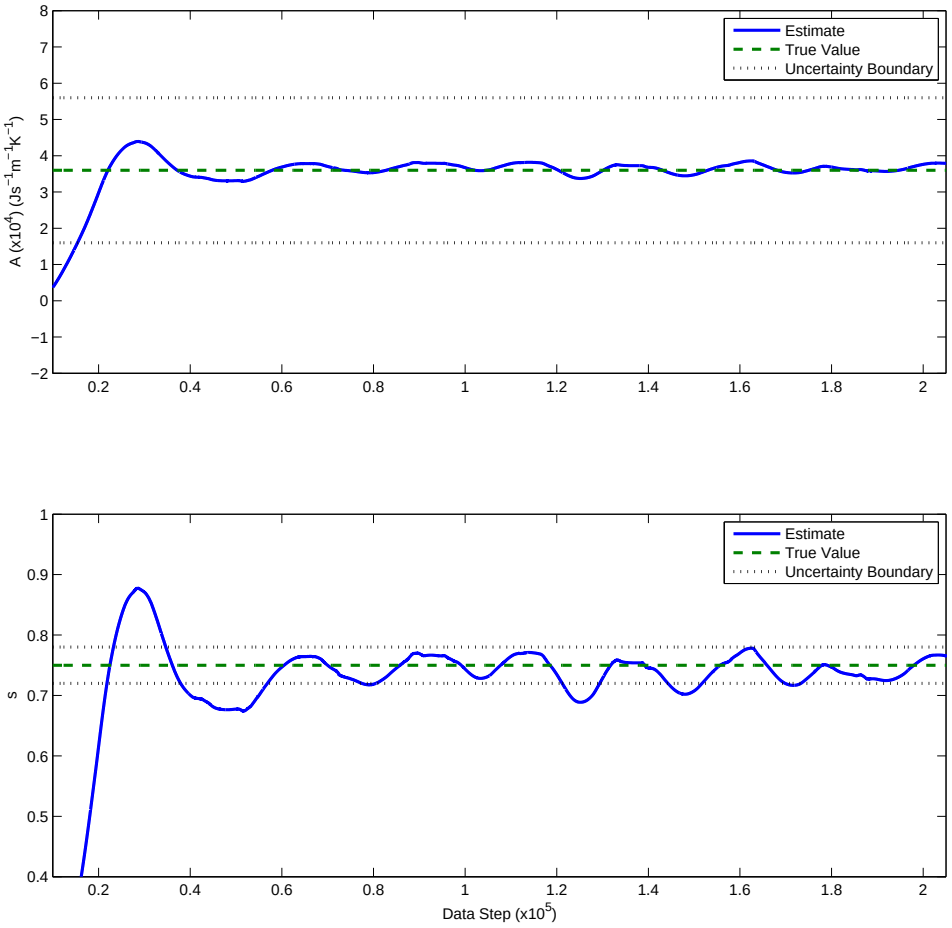


Figure 5.13: These plots show the true and estimated thermal conductivity coefficient as well as the true and estimated rate coefficient. The initial guesses for both coefficients are zero. The estimates converge to a neighborhood of the true value within about 0.6×10^5 data points. The estimates are also within the uncertainty limits.

through linear interpolation, which is an approximation, but illustrates the technique. Nothing else about the energy sink is assumed. The thermospheric density is utilized as data at 407 km altitude from a simulated GITM truth model, which includes NO cooling. Applying adaptive model refinement with temperature as the input to the "Unknown Physics," Figures 5.15 and 5.16 demonstrate that this technique captures

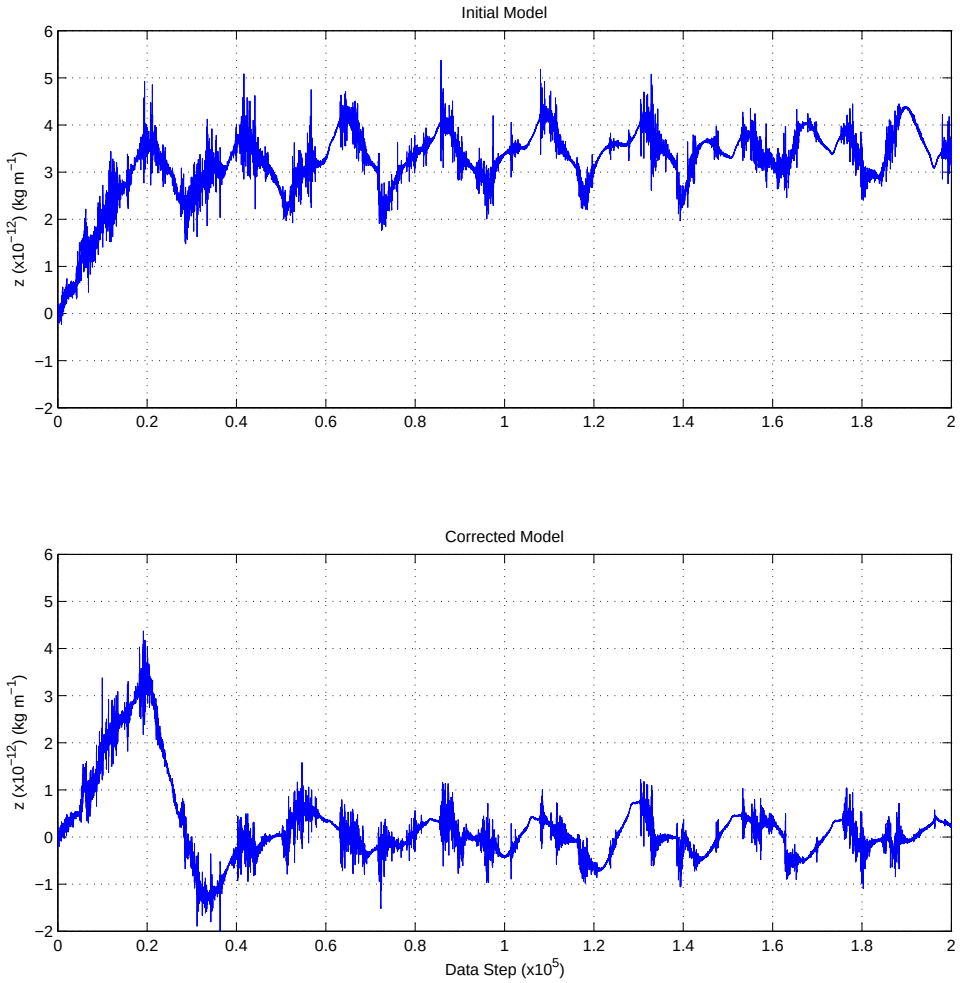


Figure 5.14: The upper figure shows the model error signal z for the difference in neutral mass density output between the GITM truth model and the GITM initial model. The lower figure shows the difference in neutral mass density output between the GITM truth model and the refined GITM model. By utilizing empirically refined estimates of the thermal conductivity and rate coefficient, the model error is reduced.

the actual dynamics in the system. The height profile of the cooling matches the actual cooling. Furthermore, the temporal variation of the maximum cooling matches the cooling simulated by the model.

To reproduce the dynamics of the cooling, three linear dynamic equations are de-

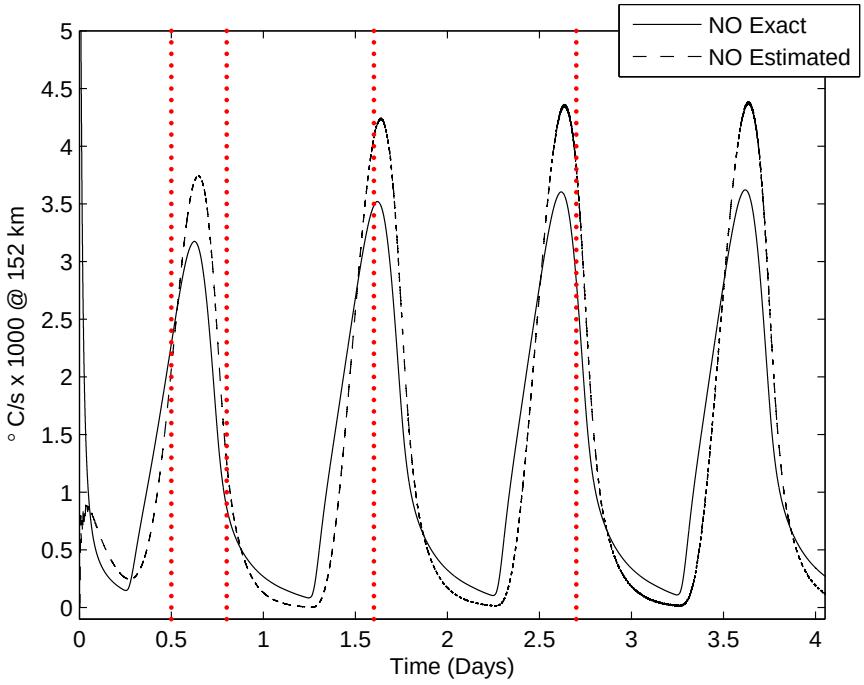
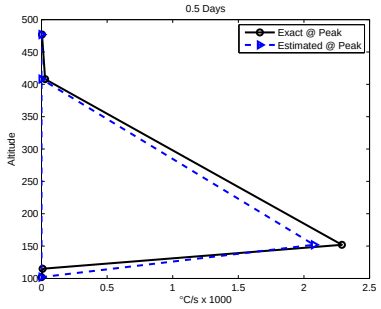


Figure 5.15: This plot shows the difference between the actual NO cooling included in the GITM truth model and the cooling in the refined GITM model as a function of time at a specific altitude (152 km). The vertical dashed lines are the time instances at which the altitude versus NO cooling plots in Figure 5.16 are taken.

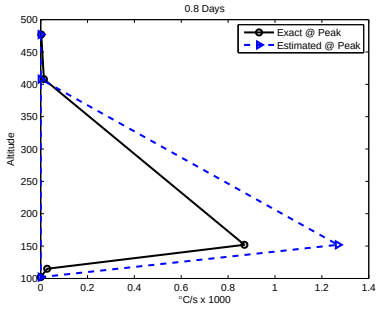
rived, one for each of the three chosen altitudes. This yields a profile that resembles the natural logarithm of the NO density [87, 88], indicating that this may be the source of the cooling, which it actually is. Figure 5.17 compares of the model without correction versus the model with correction, both of which are baselined against the truth model. Without data-based model refinement, the estimated density measurements degrade as time increases.

5.8 Conclusions

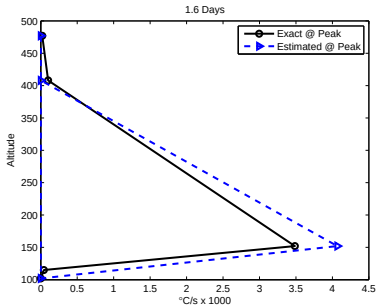
In this chapter we presented an adaptive model refinement technique for improving the fidelity of models using empirical data. Model refinement presents challenges relative to standard input-output system identification, specifically, a lack of acces-



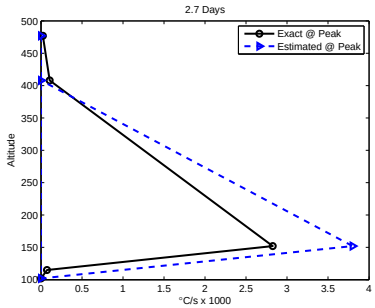
(a) NO cooling as function of altitude at 0.5 days.



(b) NO cooling as function of altitude at 0.8 days.



(c) NO cooling as function of altitude at 1.6 days.



(d) NO cooling as function of altitude at 2.7 days.

Figure 5.16: These plots show the difference between the actual NO cooling included in the truth model and the cooling estimated by the adaptive model refinement as a function of altitude at a given time. Cooling is along the horizontal axis, while altitude is along the vertical axis. The blue dashed line is the estimated value. The measured data are taken at an altitude of 407 km. The vertical dashed lines in Figure 5.15 are the time instances at which the altitude versus NO cooling plots (a)–(d) are taken.

sibility to the signals that are used by standard system identification to identify the unknown subsystem. For model refinement we use retrospective cost optimization to identify the unknown subsystem. We presented a problem formulation for the linear case, and demonstrated the method on a numerical example and an experimental setup. We then demonstrated the feasibility of the method in refining a nonlinear model of the ionosphere and thermosphere using the Global Ionosphere-Thermosphere Model (GITM). We demonstrated how uncertain parameters are identified when the structure of the uncertain model is known. Furthermore, we demonstrated how unknown dynamics are identified from data when the internal structure of the unknown

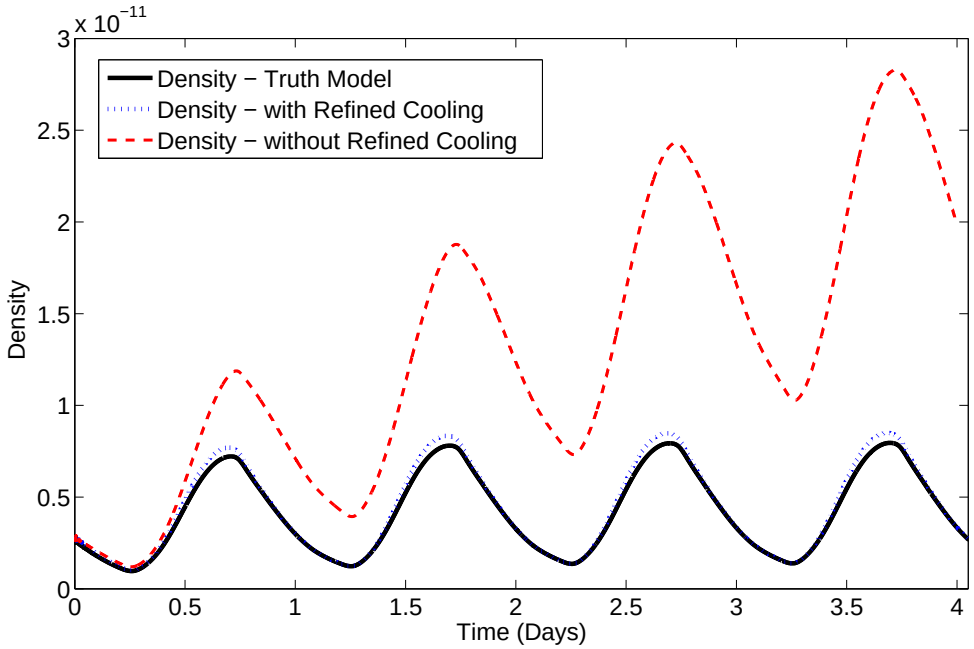


Figure 5.17: This plot shows the difference between the density measurements for the initial model, where no correction is made, and the model with the refined subsystem versus the truth model. With adaptive model refinement, the refined model is able to track the truth model, whereas, if no correction is made, the density measurements degrade as time increases.

subsystem is unknown. This technique can thus be used to refine and improve an initial model (or models, if several are hypothesized) that is either uncertain or erroneous. In turn, the improved model can provide a more accurate foundation for data assimilation aimed at wind and density estimates in the presence of solar storm disturbances.

CHAPTER VI

Application of Model Refinement to L_i Ion Batteries

6.1 Introduction

Due to their higher energy density compared to their lead-acid and nickel-metal-hydride counterparts, Lithium-ion (Li-ion) batteries have found a wide range of applications from handheld electronic devices to electrified vehicles. Understanding and optimally managing their health is critical for improving their reliability, durability, and cost.

Li-ion batteries have various degradation mechanisms depending on which combination of anode, cathode, electrolyte, and dopant chemistries are used. Provided the battery's minimum and maximum voltages are not exceeded, the predominant degradation mechanism in Li-ion batteries with lithium-iron-phosphate ($LiFePO_4$) cathodes is Solid-Electrolyte Interface (SEI) film formation in the anode [89]. This mechanism affects battery State of Health (SoH) in two ways, namely, the film resists intercalation current, increasing internal resistance, and film creation consumes Li-ions, decreasing battery capacity. Film formation depends on how the battery is charged, discharged, and stored.

To control degradation, it is necessary to predict how charging, discharging, and

storage patterns affect SoH and then modify these patterns subject to operating constraints and objectives. Noninvasive methods, such as methods based on equivalent-circuit models for capacity and power, have been used to correlate static parameters of SEI film thickness [90, 91, 92]. However, these methods do not model the dynamics of film growth and thus do not allow the battery-management system to modify charging patterns based on predicted future health. On the other hand, direct measurements of film growth require invasive methods that destroy the battery and are thus not applicable during the lifetime of the battery. Therefore, the goal of this chapter is to use system identification to construct an empirical film-growth model as a noninvasive approach to battery-health diagnostics.

The dynamics of film growth constitute a subsystem of the overall battery model, and thus the goal is to identify the dynamics of the film-growth subsystem while taking advantage of a given model of the main battery subsystem. However, identification of the film-growth subsystem is challenging due to the fact that its inputs and outputs are not available from noninvasive measurements. In this case, we say that the subsystem is *inaccessible*. Figure 6.1 illustrates the subsystem identification problem, where the input y_0 and output u of the Unknown Subsystem are not measured.

To address the inaccessible subsystem identification problem, we apply *retrospective-cost subsystem identification* (RCSI) developed in [11, 53, 54, 29]. The investigation of RCSI for noninvasive battery health diagnostics is motivated by the method's ability to estimate an inaccessible cooling submodel within an ionosphere-thermosphere model [41].

Figure 6.1 shows the RCSI framework. In this chapter we adopt a simulation-based approach and consider as the "Physical" System in Fig. 6.1 the Doyle-Fuller-Newman (DFN) battery model [93, 94] augmented with the Ramadass battery-health model [95] (DFN+R). The DFN+R model is considered as the "truth" model, with

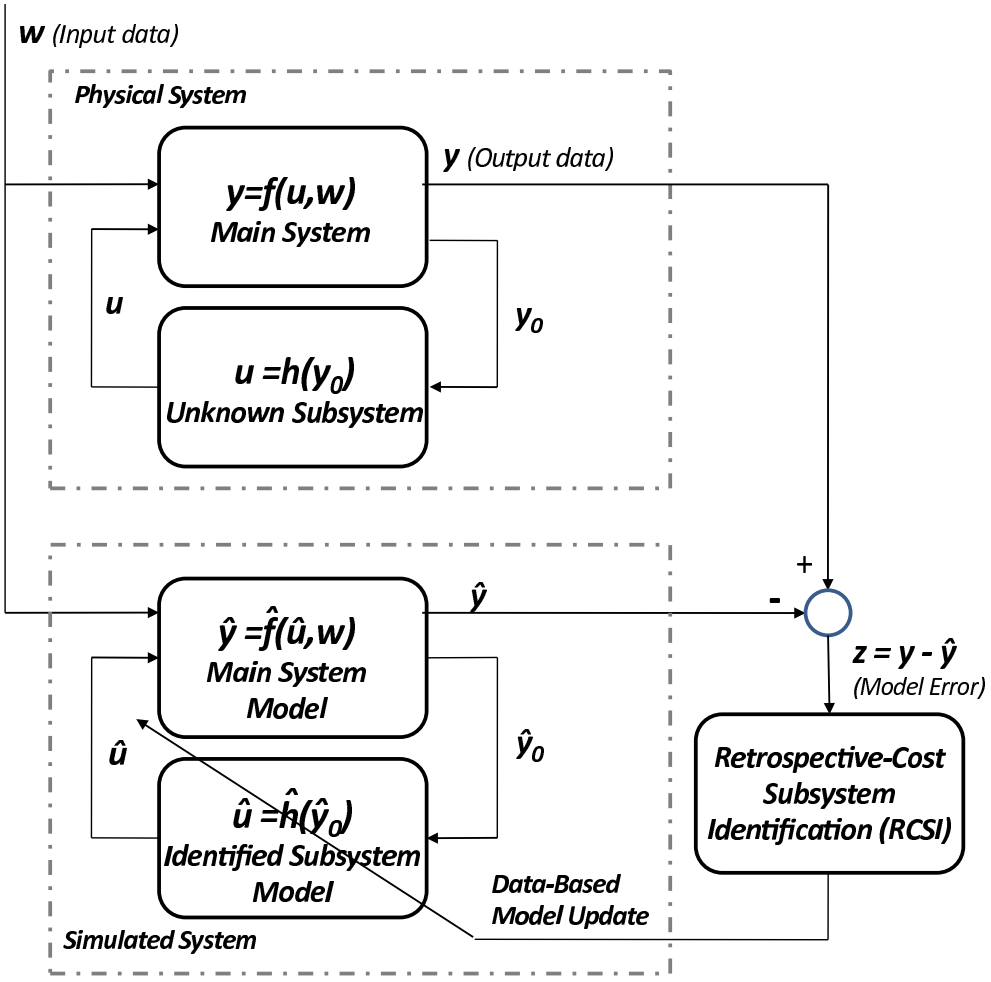


Figure 6.1: Identification of an unknown inaccessible subsystem whose input y_0 and output u are not measured. The only measured data are w and y.

the DFN model together with the Li consumption component of the battery-health model as the Main System, and the film-growth component of the battery-health model as the Unknown Subsystem. This truth model is then used for a simulation-based demonstration of RCSI where the goal is to identify the film-growth portion of the battery-health dynamics. To do this, we first simulate the DFN+R battery model to obtain data for use in subsystem identification. Next, we remove the film-growth component of the battery-health model, treating it as unknown, and use the DFN model augmented with the Li consumption component of the battery-health model as the Main System Model for RCSI. Hence, in this chapter, the Main System and Main System Model blocks shown in Fig. 6.1 are identical. We then apply RCSI to identify the film-growth subsystem model. For validation, we compare the output of the actual battery-health subsystem with the output of the battery-health subsystem model obtained from RCSI.

The rest of the chapter is organized as follows. Section 6.2 gives an overview of the battery model used in this study. The RCSI method is reviewed in Section 6.3. Section 6.4 demonstrates the application of RCSI to the film-growth idetification problem. Conclusions are given in Section 6.5.

6.2 Battery Model

6.2.1 Dynamics of Charging and Discharging

The DFN model is an electrochemical battery model that captures concentration and potential distributions across the width of the cell as well as concentration profiles in the porous electrodes of the anode and cathode. The model is described in [93, 94, 95]. This section summarizes the model equations, which constitute a system of nonlinear partial differential algebraic equations.

Li-ion battery cells consist of an anode, separator, and cathode sandwiched be-

tween current collectors. Both the anode and cathode are made of porous solid material immersed in an electrolyte solution. When the battery is fully charged, most of the Li-ions occupy interstitial sites on the solid material in the anode. As the battery discharges, the Li-ions leave these interstitial sites, entering the electrolyte solution. The Li-ions then migrate through the solution from the anode to the separator and then to the cathode. Eventually, the Li comes to rest at interstitial sites on the solid in the cathode. When a Li-ion leaves its interstitial site in the anode, an electron is freed to flow through the circuit producing useful work. When this electron reaches the cathode it causes a different Li-ion to bond with a cathode interstitial site. Charging the battery is the same process in reverse, except that the circuit provides energy rather than consumes it.

The DFN model captures local Li-ion concentrations and potentials using coupled partial differential equations (PDEs). The PDEs account for the linear diffusion of Li-ions in the electrolyte, spherical diffusion of Li-ions in the solid, and the spatially distributed electrochemical reactions driving them to transfer between the solution and the solid. The remainder of this section briefly outlines the mathematical equations behind these phenomena.

The concentration $c_2(x,t)$ of Li-ions within the electrolyte is governed by Fick's law of linear diffusion combined with an intercalation current density term J transferring Li-ions between the solution and solid as modeled by

$$\varepsilon_2 \frac{\partial c_2}{\partial t} = \nabla(d_2^{eff} \nabla c_2) + \frac{1 - t^+}{F} J \tag{6.1}$$

The intercalation current density J also acts as an input to the dynamics of Li-ion diffusion within the solid. This diffusion occurs at every point in the anode and cathode and can be modeled using a spherical, radially symmetric diffusion law given

by

$$\frac{\partial c_{1,j}}{\partial t} = \frac{D_{1,j}}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial c_{1,j}}{\partial r}\right) \tag{6.2}$$

The total intercalation current density J equals the main intercalation reaction current density J_1 plus any additional intercalation current density J_s representing side reactions in the battery. The main intercalation reaction current density J_1 is driven by potential differences between the solid and electrolyte solution, and governed by the Butler-Volmer equation

$$J_1 = a_j i_{0,j}\left(e^{\frac{\alpha_{a,j}F}{RT}\eta_j} - e^{-\frac{\alpha_{a,j}F}{RT}\eta_j}\right) \tag{6.3}$$

$$i_{0,j} = k_j\left(c_{1,j}^{max} - c_{1,j}^{S}\right)^{\alpha_{a,j}}\left(c_{1,j}^{S}\right)^{\alpha_{c,j}}\left(c_2\right)^{\alpha_{a,j}} \tag{6.4}$$

The over-potentials in these equations equal the differences between the solid and solution potentials minus the reference potentials for the main intercalation reaction, which in turn depend on the local State of Charge (SoC) according to

$$\eta_p = \phi_1 - \phi_2 - u_{p,ref} \tag{6.5}$$

$$\eta_n = \phi_1 - \phi_2 - u_{n,ref} - \frac{J}{a_n}R_{SEI} \tag{6.6}$$

Since the potentials and over-potentials can change much faster than the Li-ion concentrations, they are assumed to respond instantaneously. The solid potential is governed by Ohm's law with a term governing the charge transfer due to intercalation as given by

$$\nabla\left(\sigma_j^{eff}\nabla\phi_{1,j}\right) - J = 0 \tag{6.7}$$

Similarly, the solution potential is governed by Ohm's law, the intercalation current

density, and the charge carried by the ions in solution as modeled by

$$\nabla \left(\kappa^{eff} \nabla \phi_2\right) + J + \nabla \left(\kappa_D \nabla \ln(c_2)\right) = 0 \tag{6.8}$$

This system of equations governs the dynamics of charging and discharging in the Li-ion cell. When the DFN model is discretized, it becomes a system of Differential Algebraic Equations (DAEs), where the differential equations govern the diffusion dynamics and the algebraic equations constrain the potentials and intercalation current.

6.2.2 Battery-Health Submodel

The truth model for battery health used later in this chapter is based on a side reaction that simultaneously increases the anode SEI resistance and consumes cyclable Li-ions [95]. This side reaction is given by

$$\eta_s = \phi_1 - \phi_2 - u_{sd,ref} - \frac{J}{a_n} R_{film} \tag{6.9}$$

$$J_s = -i_{sd,0} a_n e^{-\frac{\alpha F}{RT}\eta_s} \tag{6.10}$$

The side reaction creates a resistive film at a rate proportional to the side reaction current density, that is,

$$\frac{\partial \delta_{film}}{\partial t} = -\frac{J_s M_p}{a_n \rho_p F} \tag{6.11}$$

The resistive film adds to the internal resistance of the anode, thereby negatively affecting battery performance as modeled by

$$R_{film} = R_{SEI} + \frac{\delta_{film}}{K_p} \tag{6.12}$$

Additionally, this model consumes cyclable Li-ions through the intercalation side current J_s and (6.1), resulting in capacity fade.

For subsystem identification, this health model is separated into two components, namely, film growth and Li-ion consumption. The film growth portion is identified by RCSI, whereas the Li-ion consumption piece is assumed to be part of the main model. Specifically, J_s is computed from the main model using (6.10), which is taken as an input to the RCSI algorithm. However, (6.11) and (6.12) are unknown to the RCSI algorithm; it is RCSI's task to create a model that represents these equations based on the simulated data it receives.

6.3 Retrospective-Cost Subsystem Identification

This section describes the RCSI method that is used to identify the inaccessible film-growth subsystem of the battery.

6.3.1 Retrospective Surrogate Cost-Based Signal Construction

Consider the MIMO discrete-time system

$$x(k+1) = f(x(k)) + g_u(u(k)) + g_w(w(k)), \tag{6.13}$$

$$y(k) = h(x(k)), \tag{6.14}$$

$$y_0(k) = h_0(x(k)), \tag{6.15}$$

where $x(k) \in \mathbb{R}^n$, $y^*(k) \in \mathbb{R}^{l_{y^*}}$, $y_0^*(k) \in \mathbb{R}^{l_{y_0}}$, $u(k) \in \mathbb{R}^{l_u}$, $w(k) \in \mathbb{R}^{l_w}$, and $k \geq 0$. Next,

$$u(k) = G(y_0(k)), \tag{6.16}$$

where $G(\cdot)$ is an unknown subsystem. The system (6.13)–(6.15) and (6.16) represent the real battery.

Next, we construct a model of the real system from the DFN model

$$\hat{x}(k+1) = f(\hat{x}(k)) + g_u(\hat{u}(k)) + g_w(w(k)), \tag{6.17}$$

$$\hat{y}(k) = h(\hat{x}(k)), \tag{6.18}$$

$$\hat{y}_0(k) = h_0(\hat{x}(k)), \tag{6.19}$$

$$z(k) = \hat{y}(k) - y(k), \tag{6.20}$$

where $\hat{x}(k) \in \mathbb{R}^n$, $\hat{y}(k) \in \mathbb{R}^{l_y}$, $z(k) \in \mathbb{R}^{l_z}$, $\hat{y}_0(k) \in \mathbb{R}^{l_{y_0}}$, $\hat{u}(k) \in \mathbb{R}^{l_u}$, and

$$\hat{u}(k) = \hat{G}(\hat{y}_0(k)), \tag{6.21}$$

where $\hat{G}(\cdot)$ is an estimate of $G(\cdot)$.

Next let A, B, D_1, and E_1 be the linear counterparts of f, g_u, g_w, and h and respectively. For $i \geq 1$, define the Markov parameters

$$H_i \triangleq E_1 A^{i-1} B. \tag{6.22}$$

Let r be a positive integer. Then, for all $k \geq r$,

$$\hat{x}(k) = A^r \hat{x}(k-r) + \sum_{i=1}^{r} A^{i-1} B \hat{u}(k-i) + \sum_{i=1}^{r} A^{i-1} D_1 w(k-i), \tag{6.23}$$

and thus

$$z(k) = E_1 A^r \hat{x}(k-r) + \sum_{i=1}^{r} E_1 A^{i-1} D_1 w(k-i) - y(k) + \bar{H}\bar{U}(k-1), \tag{6.24}$$

where

$$\bar{H} \triangleq \left[\begin{array}{ccc} H_1 & \cdots & H_r \end{array} \right] \in \mathbb{R}^{l_z \times r l_u}$$

and

$$\bar{U}(k-1) \triangleq \left[\hat{u}^{\mathrm{T}}(k-1) \ \cdots \ \hat{u}^{\mathrm{T}}(k-r) \right]^{\mathrm{T}}.$$

Next, we rearrange the columns of $\bar{H}$ and the components of $\bar{U}(k-1)$ and partition the resulting matrix and vector so that

$$\bar{H}\bar{U}(k-1) = \mathcal{H}'U'(k-1) + \mathcal{H}U(k-1), \tag{6.25}$$

where $\mathcal{H}' \in \mathbb{R}^{l_z \times (r l_u - l_U)}$, $\mathcal{H} \in \mathbb{R}^{l_z \times l_U}$, $U'(k-1) \in \mathbb{R}^{r l_u - l_U}$, and $U(k-1) \in \mathbb{R}^{l_U}$. Then, we can rewrite (6.24) as

$$z(k) = \mathcal{S}(k) + \mathcal{H}U(k-1), \tag{6.26}$$

where

$$\mathcal{S}(k) \triangleq E_1 A^r \hat{x}(k-r) + \sum_{i=1}^{r} E_1 A^{i-1} D_1 w(k-i) - y(k) + \mathcal{H}'U'(k-1). \tag{6.27}$$

Next, we rewrite (6.26) with a delay of k_j time steps, where $0 \leq k_1 \leq k_2 \leq \cdots \leq k_s$, in the form

$$z(k-k_j) = \mathcal{S}_j(k-k_j) + \mathcal{H}_j U_j(k-k_j-1), \tag{6.28}$$

where (6.27) becomes

$$\mathcal{S}_j(k-k_j) \triangleq E_1 A^r \hat{x}(k-k_j-r) + \sum_{i=1}^{r} E_1 A^{i-1} D_1 w(k-k_j-i)$$
$$- y(k-k_j) + \mathcal{H}'_j U'_j(k-k_j-1)$$

and (6.25) becomes

$$\bar{H}\bar{U}(k-k_j-1) = \mathcal{H}'_j U'_j(k-k_j-1)$$
$$+ \mathcal{H}_j U_j(k-k_j-1), \tag{6.29}$$

where $\mathcal{H}'_j \in \mathbb{R}^{l_z \times (rl_u - l_{U_j})}$, $\mathcal{H}_j \in \mathbb{R}^{l_z \times l_{U_j}}$, $U'_j(k-k_j-1) \in \mathbb{R}^{rl_u - l_{U_j}}$, and $U_j(k-k_j-1) \in \mathbb{R}^{l_{U_j}}$. Now, by stacking $z(k-k_1), \ldots, z(k-k_s)$, we define the *extended performance*

$$Z(k) \triangleq \left[z^{\mathrm{T}}(k-k_1) \ \cdots \ z^{\mathrm{T}}(k-k_s)\right]^{\mathrm{T}} \in \mathbb{R}^{sl_z}. \tag{6.30}$$

Therefore,

$$Z(k) \triangleq \tilde{\mathcal{S}}(k) + \tilde{\mathcal{H}}\tilde{U}(k-1), \tag{6.31}$$

where

$$\tilde{\mathcal{S}}(k) \triangleq \left[\mathcal{S}(k-k_1) \ \cdots \ \mathcal{S}(k-k_s)\right]^{\mathrm{T}} \in \mathbb{R}^{sl_z}, \tag{6.32}$$

$\tilde{\mathcal{H}} \in \mathbb{R}^{sl_z \times l_{\tilde{U}}}$, and $\tilde{U}(k-1) \in \mathbb{R}^{l_{\tilde{U}}}$. The vector $\tilde{U}(k-1)$ is formed by stacking $U_1(k-k_1-1), \ldots, U_s(k-k_s-1)$ and removing repetitions of components. The coefficient matrix $\tilde{\mathcal{H}}$ consists of the entries of $\mathcal{H}_1, \ldots, \mathcal{H}_s$ arranged according to the structure of $\tilde{U}(k-1)$. Furthermore, we assume that the last entry of $\tilde{U}(k-1)$ is a component of $\hat{u}(k-r)$.

Next, we define the *surrogate performance*

$$z^*(k-k_j) \triangleq \mathcal{S}_j(k-k_j) + \mathcal{H}_j U_j^*(k-k_j-1), \tag{6.33}$$

where the actual past subsystem outputs $U_j(k-k_j-1)$ in (6.28) are replaced by the surrogate subsystem outputs $\hat{U}_j^*(k-k_j-1)$. The *extended surrogate performance* for (6.33), which is defined as

$$Z^*(k) \triangleq \left[z^{*\mathrm{T}}(k-k_1) \ \cdots \ z^{*\mathrm{T}}(k-k_s)\right]^{\mathrm{T}} \in \mathbb{R}^{sl_z}, \tag{6.34}$$

is given by

$$Z^*(k) = \tilde{\mathcal{S}}(k) + \tilde{\mathcal{H}}\tilde{U}^*(k-1), \tag{6.35}$$

where the components of $\tilde{U}^*(k-1) \in \mathbb{R}^{l_{\tilde{U}^*}}$ are components of $\hat{U}_1^*(k-k_1-1), \ldots, \hat{U}_s^*(k-k_s-1)$ ordered in the same way as the components of $\tilde{U}^*(k-1)$. Subtracting (6.31) from (6.35) yields

$$Z^*(k) = Z(k) - \tilde{\mathcal{H}}\tilde{U}(k-1) + \tilde{\mathcal{H}}\tilde{U}^*(k-1). \tag{6.36}$$

Finally, we define the *retrospective cost function*

$$\bar{J}(\tilde{U}^*(k-1), k) \triangleq Z^{*\mathrm{T}}(k)R(k)Z^*(k) + \eta(k)\tilde{U}^{*\mathrm{T}}(k-1)\tilde{U}^*(k-1), \tag{6.37}$$

where $R(k) \in \mathbb{R}^{l_z s \times l_z s}$ is a positive-definite performance weighting and $\eta(k) \geq 0$. The goal is to determine refined subsystem outputs $\hat{\tilde{U}}(k-1)$ that would have provided better performance than the subsystem outputs $U(k)$ that were applied to the system. The refined subsystem output values $\hat{\tilde{U}}(k-1)$ are subsequently used to update the

subsystem estimate.

Substituting (6.36) into (6.37) yields

$$\bar{J}(\tilde{U}^*(k-1),k) = \tilde{U}^*(k-1)^{\mathrm{T}}\mathcal{A}(k)\tilde{U}^*(k-1) + \tilde{U}^{*\mathrm{T}}(k-1)\mathcal{B}^{\mathrm{T}}(k) + \mathcal{C}(k), \tag{6.38}$$

where

$$\mathcal{A}(k) \triangleq \tilde{\mathcal{H}}^{\mathrm{T}}R(k)\tilde{\mathcal{H}} + \eta(k)I_{l_{\tilde{U}}}, \tag{6.39}$$

$$\mathcal{B}(k) \triangleq 2\tilde{\mathcal{H}}^{\mathrm{T}}R(k)[Z(k) - \tilde{\mathcal{H}}\tilde{U}(k-1)], \tag{6.40}$$

$$\mathcal{C}(k) \triangleq Z^{\mathrm{T}}(k)R(k)Z(k) - 2Z^{\mathrm{T}}(k)R(k)\tilde{\mathcal{H}}\tilde{U}(k-1) + \tilde{U}^{\mathrm{T}}(k-1)\tilde{\mathcal{H}}^{\mathrm{T}}R(k)\tilde{\mathcal{H}}\tilde{U}(k-1). \tag{6.41}$$

If either $\tilde{\mathcal{H}}$ has full column rank or $\eta(k) > 0$, then $\mathcal{A}(k)$ is positive definite. In this case, $\bar{J}(\tilde{U}^*(k-1),k)$ has the unique global minimizer

$$\tilde{U}^*(k-1) = -\frac{1}{2}\mathcal{A}^{-1}(k)\mathcal{B}(k). \tag{6.42}$$

6.3.2 Film Resistance Model

The estimated subsystem output $\hat{u}(k)$ is given by the exactly proper time-series model of order n_{c} given by

$$\hat{u}(k) = \sum_{i=1}^{n_{\mathrm{c}}} M_i(k)\hat{u}(k-i) + \sum_{i=0}^{n_{\mathrm{c}}-1} N_i(k)\hat{y}_0(k-i), \tag{6.43}$$

where, for all $i = 1,\ldots,n_{\mathrm{c}}$, $M_i(k) \in \mathbb{R}^{l_u \times l_u}$ and $N_i(k) \in \mathbb{R}^{l_u \times l_{\hat{y}_0}}$. The subsystem output (6.43) can be expressed as

$$\hat{u}(k) = \theta(k)\phi(k-1), \tag{6.44}$$

where $\theta(k) \in \mathbb{R}^{l_u \times n_c(l_u+l_{y_0})}$, is

$$\theta(k) \triangleq [M_1(k) \ \cdots \ M_{n_c}(k) \ N_1(k) \ \cdots \ N_{n_c}(k)] \tag{6.45}$$

and $\phi(k-1) \in \mathbb{R}^{n_c(l_u+l_{\hat{y}_0})}$,

$$\phi(k-1) \triangleq \left[\hat{u}^{\mathrm{T}}(k-1) \ \cdots \ \hat{u}^{\mathrm{T}}(k-n_c) \ \hat{y}_0^{\mathrm{T}}(k-1) \ \cdots \ \hat{y}_0^{\mathrm{T}}(k-n_c)\right]^{\mathrm{T}}. \tag{6.46}$$

Next, let d be a positive integer such that $\tilde{U}^*(k-1)$ contains $u^*(k-d)$. We define the cumulative cost function

$$\begin{aligned} J_{\mathrm{R}}(\theta(k)) \triangleq & \sum_{i=d+1}^{k} \lambda^{k-i} \| \phi^{\mathrm{T}}(i-d-1)\theta^{\mathrm{T}}(k) \\ & - u^{*\mathrm{T}}(i-d) \|^2 + \lambda^k (\theta(k)-\theta(0)) P^{-1}(0) (\theta(k)-\theta(0))^{\mathrm{T}}, \end{aligned} \tag{6.47}$$

where $\phi(k-d)$ is given by (6.46) and $\lambda(k) \in (0,1]$ is the forgetting factor. Minimizing (6.47) yields

$$\begin{aligned} \theta^{\mathrm{T}}(k) = & \alpha(k)\theta^{\mathrm{T}}(0) + [1-\alpha(k)][\theta^{\mathrm{T}}(k-1) + P(k-1) \\ & \cdot \phi(k-d-1)[\phi^{\mathrm{T}}(k-d-1)P(k-1)\phi(k-d-1) \\ & + \lambda(k)]^{-1}(\hat{u}(k-d) - \phi^{\mathrm{T}}(k-d-1)\theta^{\mathrm{T}}(k-1))]. \end{aligned} \tag{6.48}$$

The error covariance is updated by

$$\begin{aligned} P(k) = & \alpha(k)P(0) + [1-\alpha(k)][\lambda^{-1}(k)P(k-1) \\ & - \lambda^{-1}(k)P(k-1)\phi(k-d-1) \\ & [\phi^{\mathrm{T}}(k-d-1)P(k-1)\phi(k-d-1) + \lambda(k)]^{-1} \qquad (6.49) \\ & \cdot \phi^{\mathrm{T}}(k-d-1)P(k-1)], \qquad (6.50) \end{aligned}$$

where $\alpha(k) \in (0,1)$ is an algorithm reset, that is, when $\alpha(k) = 1$ $\theta(k)$ and $P(k)$ are reset to their initial values. Furthermore we initialize the error covariance matrix as $P(0) = \beta I$, where $\beta > 0$.

6.4 Numerical Simulation of the Application of RCSI for Film-Growth Subsystem Identification

We now present a numerical simulation of the application of RCSI to the film-growth identification problem. To this end, the DFN model together with the Li consumption component of the Ramadass battery-health model is considered as the Main System Model of the RCSI framework shown in Figure 6.1, and the film-growth component of the battery-health model is considered as the Unknown Subsystem to be identified. The adoption of this film-growth subsystem identification problem into the general RCSI framework is illustrated in Fig. 6.2. The Unknown Film-Growth Subsystem $G(\cdot)$ is connected to the Main Battery System $(f(\cdot), g_w(\cdot), g_u(\cdot), h(\cdot), h_0(\cdot))$ by feedback, which captures the fact that the film is driven by the intercalation side current, while the film impacts the local overpotential of the main reaction, restricting battery current. Note that neither the input $u(k)$ nor the output $y_0(k)$ of the film-growth subsystem is measured.

Although in practice the data, namely, battery terminal voltage and current, would be obtained from a physical experiment, the results in this section are based on simulations. To obtain simulated test data, the DFN+R battery model was simulated under repeated Constant-Current, Constant-Voltage (CCCV) cycling from 2 to 3.6 V at a 2.5 C-rate. The parameters for the DFN model are taken from [96]. For the Ramadass health model the parameters are assumed to be $u_{sd,ref} = 0.4$, $i_{sd,0} = 4 \times 10^{-9}$, $M_p = 7.3 \times 10^4$, $\rho_p = 2.1 \times 10^3$, and $K_p = 1$. The film-growth subsystem was then removed from the truth model in accordance with the assumption that it

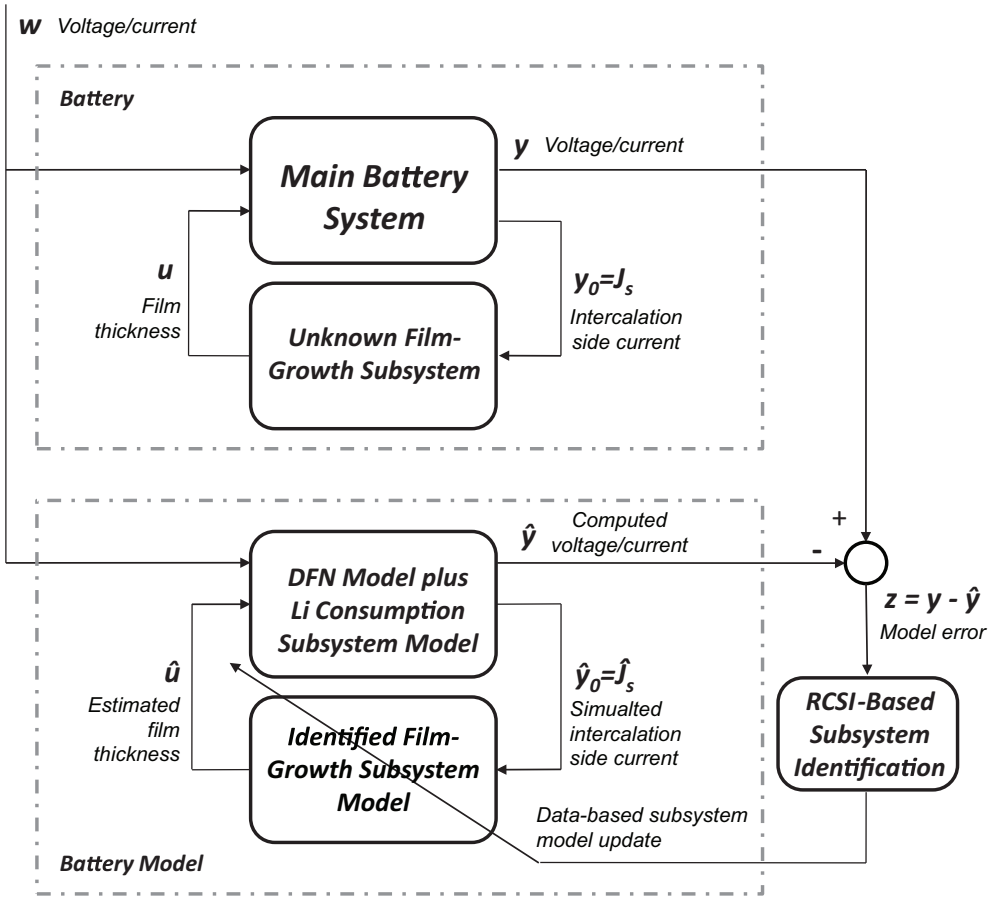


Figure 6.2: Specialization of Figure 6.1 to the film-growth identification problem.

is unknown. RCSI was then tasked with identifying the dynamics of the Unknown Film-Growth Subsystem. The controller and tuning parameters where chosen to be $n_c = 7$, $\eta(k) = 0$, $P(0) = 5 \times 10^{-7}$. In the absence of estimates of A, B, and E_1, we choose $\tilde{\mathcal{H}} = [\hat{H}_1]$, where $\hat{H}_1 = 0.01$. Next, $\hat{y}_0(k)$ is the intercalation side current J_s, and y is the system output voltage or current depending on the cycle. Finally, $\alpha(k) = 1$ at the start of each cycle, that is, $P(k)$ and $\theta(k)$ are re-initialized at the start of each charging cycle.

Figure 6.3 shows the true (that is, truth-model) film thickness as given by DFN+R model and the film thickness as estimated by RCSI. The film-thickness estimates show that the film-thickness subsystem dynamics are not identifiable during intervals of operation within which the intercalation side current J_s is close to zero. However, when J_s is large, RCSI produces a useful estimate of the film thickness that is close to the true film thickness. Figure 6.3 provides a magnified view of Fig. 6.3, which shows that the estimates of film thickness provided by RCSI correspond closely to the true film thickness during intervals in which J_s is large. The resulting identified film-growth model can be used to identify unknown physics, validate hypothesized physics, or predict the future behavior of the battery.

Figure 6.5 (a) shows the time history of θ, where θ are the coefficients of the estimated transfer function between the estimated intercalation side current and estimated film resistance. We note that θ is reset to zero at the start of each cycle. 6.5 (b) shows the normalized estimated intercalation side current and the normalized estimated film resistance. Since the process between intercalation side current and the estimated film resistance is modeled using transfer functions, when the input (intercalation side current) to the transfer function is approximately zero, the film resistance estimate will tend to zero, which means the estimates when the side current is small are unreliable.

Figure 6.6 (a) shows the pole-zero plot of the estimated subsystem at $t = 470$ mins.

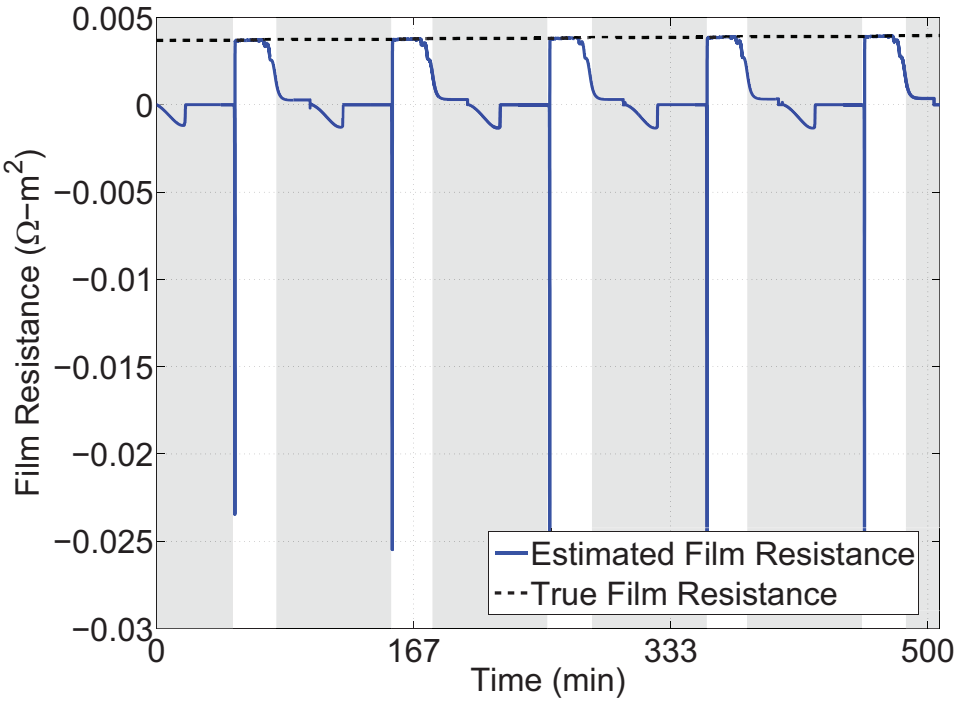


Figure 6.3: True film thickness and the film thickness as estimated by RCSI. Shaded regions indicate when the intercalation side current J_s is close to zero.

The estimated subsystem is approximately a finite impulse response (FIR) system, note the grouping of poles at the origin. Furthermore, Figure 6.6 (b) shows the impulse response of the subsystem, which supports the observation that the subsystem is approximately FIR.

6.5 Conclusions and Future Research

As the first step of a noninvasive solution to battery-health diagnostics, we applied RCSI to the problem of estimating the SEI film-growth subsystem of a battery model for which the main system is the DFN model augmented with a Li consumption model. Assuming that the main system model is accurate and the measurements are noise free, RCSI was able to reproduce the output of the "truth" film-growth model when the intercalation side current J_s is not close to zero. Future research will focus on the case in which the DFN model is uncertain (that is, does not match the true

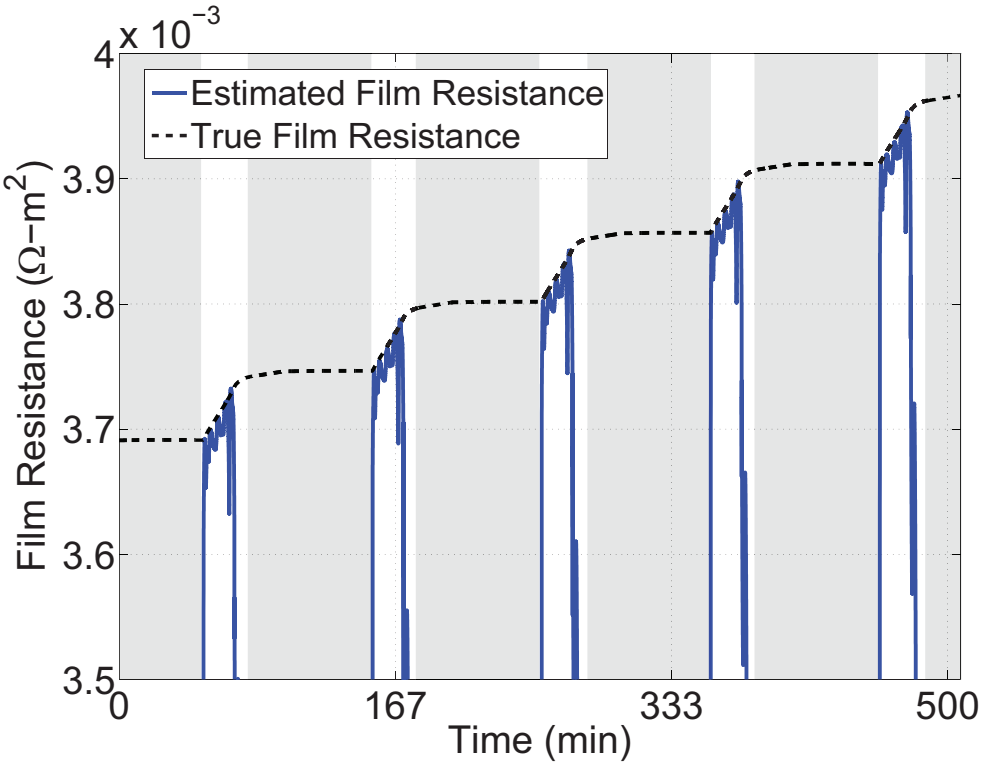


Figure 6.4: A magnified view of Figure 6.3.

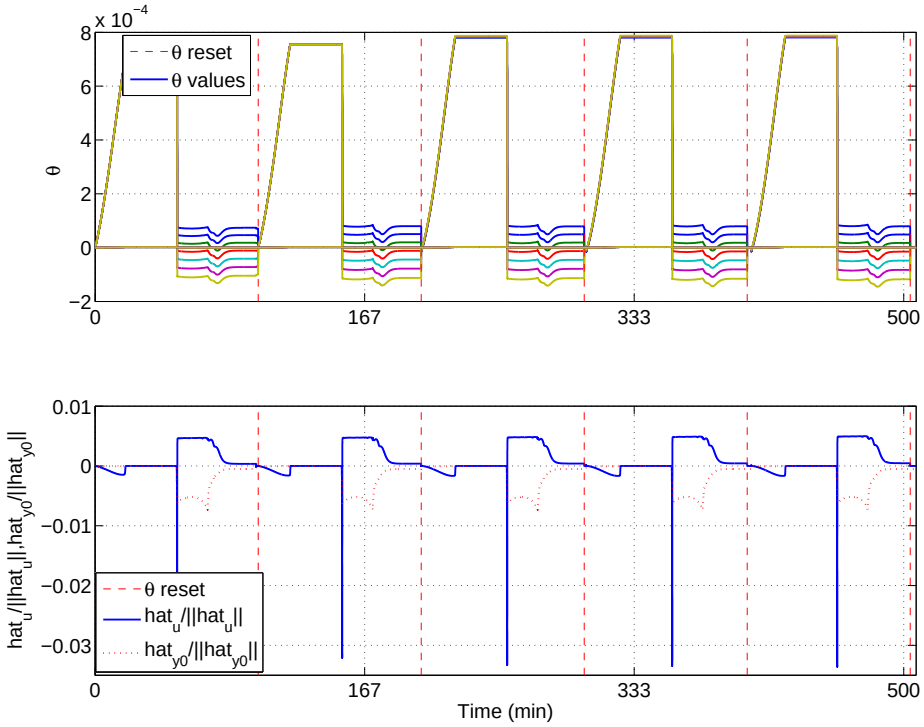


Figure 6.5: (a) is the time evolution of θ. These parameters are the coefficients of the transfer function from the estimated intercalation side current to the estimated film resistance. Note that θ is reset to zero at the start of each cycle. (b) is shows the normalized traces of estimated intercalation side current and estimated film resistance. When intercalation side current is near zero, the estimates of the film resistance tend to zero and are unreliable.

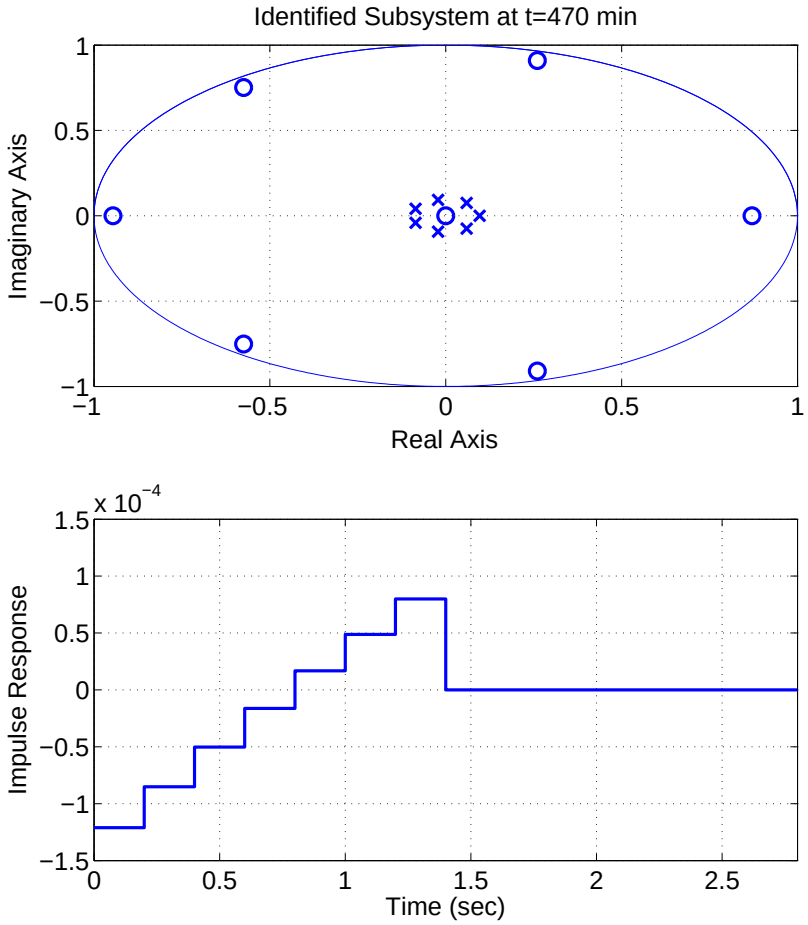


Figure 6.6: (a) is a pole-zero plot of the transfer function from intercalation side current and film resistance at $t = 470$ mins. Note that the transfer function is approximately a finite impulse response system. (b) shows the impulse response of this transfer function.

battery characteristics) and the measurements are noisy. The ultimate goal is to apply the subsystem identification method to experimental battery data.

CHAPTER VII

Application of Model Refinement to Semi-Parametric Identification of Wiener Systems

7.1 Introduction

Block-structured models are widely used for system identification [64, 97, 47]. These models provide useful information concerning the dynamic and static components of a system, and thus constitute grey-box models in which the block structure is ascribed physical meaning. The goal of system identification is to model the internal structure of each block from available data.

Among the most widely studied block-structured models are the Wiener [42, 46, 45, 98, 99, 100, 101, 43] and Hammerstein [42, 102, 103, 11, 43, 44] models. Each model structure involves a single linear dynamic block and a single nonlinear static block. For these two-block structures, the difficulty of the identification problem typically depends on *a priori* assumptions made about the components, for example, FIR-versus-IIR dynamics, and invertible-versus-noninvertible nonlinearities [99]. Furthermore, identification of Wiener systems is generally considered to be more challenging than identification of Hammerstein systems due to the fact that the input to the nonlinear block is available for Hammerstein systems but not for Wiener systems. In this chapter, we focus on Wiener systems.

The methods for identifying Wiener systems developed in [42, 45, 98, 101] assume that the nonlinear block is invertible. To overcome this requirement, nonparametric probabilistic methods are used in [47]. Alternatively, frequency-domain methods that apply multiple harmonic inputs are employed in [43, 46]. In [43], the multiple harmonic inputs are assigned random phase shifts, and a nonparametric model of the nonlinearity is obtained using the identified linear dynamic model, which is previously estimated in the frequency domain. In [46], the phase shift between the output of the linear dynamic block and the output is exploited in the frequency domain, for each harmonic input.

In this chapter we develop a novel technique for identifying single-input, single-output (SISO) Wiener systems. The proposed approach is semiparametric, which, as described in [47], refers to the fact that the nonlinear block is estimated nonparametrically, whereas the linear dynamics are identified parametrically. To do this, we consider a two-step procedure. In the first step, we apply a single harmonic input signal, and measure the output once the trajectory of the system reaches harmonic steady state. We then examine the output of the system (which is not harmonic due to the nonlinearity) relative to the input, and use the symmetry properties of these signals to estimate the nonharmonic phase shift. This estimate allows us to infer the phase shift of the unmeasured intermediate signal (that is, the output of the linear block) and thus reconstruct this signal up to an arbitrary amplitude. By plotting the output versus the reconstructed intermediate signal, we thus obtain a nonparametric approximation of the nonlinear block of the system.

The second step of the algorithm uses a sufficiently rich signal to estimate the linear dynamics of the system. Since we do not assume that the nonlinear block is invertible, we do not have an estimate of the output of the linear block. To overcome this difficulty, we apply retrospective cost optimization, which uses the available output signal (in this case, the output of the nonlinear block) to recursively update

the linear dynamics. This technique is inspired by retrospective-cost-based adaptive control [50, 53, 48], which is used for model updating in [54, 11, 11].

As alluded to above, the two-step identification algorithm described herein does not require invertibility of the nonlinear block as assumed in [42, 45, 98, 101]. In fact, we do not require that the nonlinear block be either one-to-one, onto, or continuous, nor do we assume as in [45] that any specific value of the nonlinearity be known.

The contents of the chapter are as follows. In Section 7.2 we define the Wiener identification problem. A method for nonparametric identification of the static nonlinearity using a single harmonic input is presented in Section 7.3, while a method for parametric identification of the linear time-invariant dynamics using retrospective cost optimization is reviewed in Section 7.4. These methods are demonstrated on several examples of increasing complexity in sections 7.5, 7.6, and 7.7. Concluding remarks are presented in Section 7.8. A preliminary version of the results of this chapter appears as [39].

7.2 Problem Formulation

Consider the block-structured Wiener model shown in Figure 7.1a, where $\mathcal{L}$ is the SISO discrete-time linear time-invariant dynamic system

$$x(k+1) = Ax(k) + Bu(k), \tag{7.1}$$

$$v(k) = Cx(k), \tag{7.2}$$

with input $u(k) \in \mathbb{R}$ and intermediate signal $v(k) \in \mathbb{R}$, where k is the sample index, and $y(k) \in \mathbb{R}$ is the output given by

$$y(k) = \mathcal{W}(v(k)), \tag{7.3}$$

where $\mathcal{W} : \mathbb{R} \longmapsto \mathbb{R}$ is the static nonlinearity. We assume that $\mathcal{L}$ is asymptotically stable and $\mathcal{W}$ is piecewise continuous. Note that we do not assume that $\mathcal{W}$ is invertible, one-to-one, continuous, or (as in [45]) $\mathcal{W}(0) = 0$. Also, we assume that $v(k)$ is not accessible, and that $x(0)$ is unknown and possibly nonzero.

Moreover, Figure 7.1b shows the scaled-domain modification $\mathcal{W}_\lambda(\nu) \stackrel{\triangle}{=} \mathcal{W}\left(\frac{\nu}{\lambda}\right)$ of $\mathcal{W}$, where λ is a nonzero real number. Therefore, $\mathcal{W}_\lambda(\lambda v) = \mathcal{W}(v)$. Each value of λ scales both the gain of $\mathcal{L}$ and the domain of $\mathcal{W}$. However, λ is not identifiable.

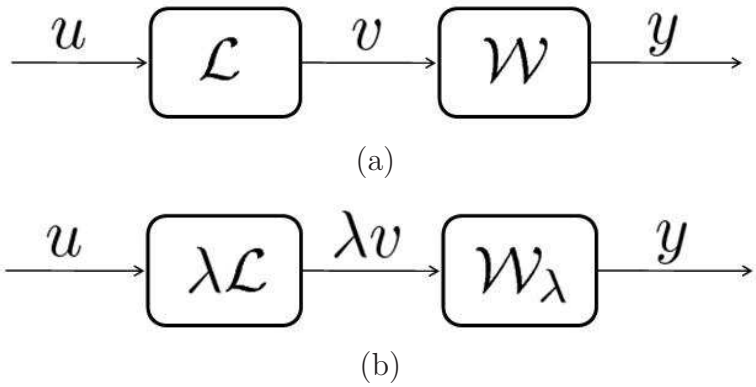


Figure 7.1: (a) Block-structured Wiener model, where u is the input, v is the intermediate signal, y is the output, $\mathcal{L}$ is a discrete-time linear time-invariant dynamic system, and $\mathcal{W}$ is a static nonlinearity. (b) An equivalent scaled model, where λ is a scaling factor and $\mathcal{W}_\lambda$ is a scaled-domain modification of $\mathcal{W}$ satisfying $\mathcal{W}_\lambda(\lambda v) = \mathcal{W}(v)$. The scaling factor λ is not identifiable.

7.3 Nonparametric Identification of the Static Nonlinearity

Consider the harmonic input signal

$$u(k) = A_0 \sin(\omega_0 k T_s) = A_0 \sin(\Omega_0 k), \tag{7.4}$$

where A_0 is the amplitude, ω_0 is the angular frequency in rad/sec, T_s is the sample period in sec/sample, and $\Omega_0 \stackrel{\triangle}{=} \omega_0 T_\mathrm{s}$ is the angular sample frequency in rad/sample. Since $\mathcal{L}$ is asymptotically stable, it follows that, for large values of k, the intermediate

signal v is given approximately by the harmonic steady-state signal

$$v(k) = |G(e^{\jmath\Omega_0})|A_0 \sin(\Omega_0 k + \angle G(e^{\jmath\Omega_0})), \tag{7.5}$$

where $|G(e^{\jmath\Omega_0})|$ and $\angle G(e^{\jmath\Omega_0})$ are, respectively, the magnitude and phase shift of the frequency response of $G(\mathbf{z}) = C(\mathbf{z}I - A)^{-1}B$ at the angular sample frequency Ω_0. Therefore,

$$y(k) = \mathcal{W}(|G(e^{\jmath\Omega_0})|A_0 \sin(\Omega_0 k + \angle G(e^{\jmath\Omega_0})). \tag{7.6}$$

Next, note that the continuous-time harmonic signal $\sin(\omega_0 t)$ is symmetric in the intervals $\left[0, \frac{1}{2}T_0\right]$ and $\left[\frac{1}{2}T_0, T_0\right]$ about the points $\frac{1}{4}T_0$ and $\frac{3}{4}T_0$, respectively, where $T_0 \triangleq \dfrac{2\pi}{\omega_0}$ is the period of the harmonic input. To preserve symmetry for the sampled signal (7.4) about the points $\frac{1}{4}T_0$ and $\frac{3}{4}T_0$, we assume that

$$\Omega_0 = \frac{\pi}{2m}, \tag{7.7}$$

where m is a positive integer. Thus $N_0 \triangleq 4m = \dfrac{T_0}{T_\mathrm{s}}$ is the period of the discrete-time input (7.4). With this choice of Ω_0, the sampled signal $u(k)$ is symmetric in the intervals $\left[0, \frac{1}{2}N_0\right]$ and $\left[\frac{1}{2}N_0, N_0\right]$ about the points $\frac{1}{4}N_0$ and $\frac{3}{4}N_0$, respectively. Furthermore, assuming that $q \triangleq \dfrac{\angle G(e^{\jmath\Omega_0})}{\Omega_0}$ is an integer, that is, $\dfrac{\angle G(e^{\jmath\Omega_0})}{\pi}$ is an integer, the intermediate signal $v(k)$, which is shifted relative to $u(k)$ due to $\angle G(e^{\jmath\Omega_0})$, is symmetric about $\frac{1}{4}N_0 + q$ in the interval $\left[q, \frac{1}{2}N_0 + q\right]$ and about $\frac{3}{4}N_0 + q$ in the interval $\left[\frac{1}{2}N_0 + q, N_0 + q\right]$. If q is not an integer, then $v(k)$ is only approximately symmetric.

Next, we note that the output signal y, which is not generally harmonic, possesses the same symmetry as v on the same intervals. By exploiting knowledge of this symmetry, we can identify the *nonharmonic phase shift* of y relative to u, and thus the phase shift of v relative to u. Since y is not sinusoidal, the nonharmonic phase

shift of y relative to u refers to the shifting of the symmetric portions of y relative to the symmetric portions of u. Knowledge of this nonharmonic phase shift allows us to determine v up to a constant multiple, specifically, v is a sinusoid that is shifted relative to u by a known number of samples.

To clarify the above discussion, we present two examples using $A_0 = 1$, $m = 18$ (so that $\Omega_0 = \pi/36$), and $G(\mathbf{z}) = \dfrac{0.0685}{\mathbf{z} - 0.9164}$. First, consider the polynomial nonlinearity $y = \mathcal{W}(v) = 0.6(v+1)^3 - 1$, which is neither even nor odd. Figure 7.2a illustrates the resulting signals $u(k)$, $v(k)$, and $y(k)$ in harmonic steady state. Note that u is symmetric about the discrete-time index δ in the interval $\left[\delta - \frac{1}{4}N_0, \delta + \frac{1}{4}N_0\right]$ and about $\delta + \frac{1}{2}N_0$ in the interval $\left[\delta + \frac{1}{4}N_0, \delta_2 + \frac{3}{4}N_0\right]$. Likewise, v is symmetric about the discrete-time index ε in the interval $\left[\varepsilon - \frac{1}{4}N_0, \varepsilon + \frac{1}{4}N_0\right]$ and about $\varepsilon + \frac{1}{2}N_0$ in the interval $\left[\varepsilon + \frac{1}{4}N_0, \varepsilon + \frac{3}{4}N_0\right]$. It thus follows that y is symmetric about ε in the interval $\left[\varepsilon - \frac{1}{4}N_0, \varepsilon + \frac{1}{4}N_0\right]$ and about $\varepsilon + \frac{1}{2}N_0$ in the interval $\left[\varepsilon + \frac{1}{4}N_0, \varepsilon + \frac{3}{4}N_0\right]$.

Second, we consider the even polynomial nonlinearity $y = \mathcal{W}(v) = v^2$. Figure 7.2b illustrates the resulting signals $u(k)$, $v(k)$, and $y(k)$ in harmonic steady state. The signals u and v are equal to the signals shown in Figure 7.2a. However, in addition to the two points of symmetry shown in Figure 7.2a, note that y has two additional points of symmetry, specifically, y is symmetric about $\varepsilon + \frac{1}{4}N_0$ in the interval $\left[\varepsilon, \varepsilon + \frac{1}{2}N_0\right]$ and about $\varepsilon + \frac{3}{4}N_0$ in the interval $\left[\varepsilon + \frac{1}{2}N_0, \varepsilon + N_0\right]$.

7.3.1 Symmetry Search Algorithm

We now present an algorithm to determine ε from y. We then use ε to estimate the nonharmonic phase shift of y relative to u. For convenience, we assume that the harmonic steady state begins at $k = 0$.

Consider the signal y shown in Figure 7.3, and let $n \geq 6m$ denote the width of the data window so that it includes at least one and a half periods. To encompass a complete signal period, we construct a sliding window with $N_0 + 1$ data points. The

window is divided into quarters as shown in Figure 7.3.

Next, for $k = 0, \ldots, n - N_0$, define

$$\beta_1(k) \triangleq \sum_{i=1}^{2m-1} |y(k+i-1) - y(k+2m-i+1)|, \tag{7.8}$$

which is the sum of the absolute difference in magnitude for each pair of candidate symmetric points in the first and second quarters about the point $k + \frac{1}{4}N_0$ for the sliding window starting at time step k. Likewise, for $k = 0, \ldots, n - N_0$, define

$$\beta_2(k) \triangleq \sum_{i=1}^{2m-1} |y(k+2m+i-1) - y(k+4m-i+1)|, \tag{7.9}$$

for each pair of candidate symmetric points in the third and fourth quarters about the point $k + \frac{3}{4}N_0$. The values of β_1 and β_2 quantify the symmetry error about the points $k + \frac{1}{4}N_0$ and $k + \frac{3}{4}N_0$, respectively, for each allowable value of k. Thus, using (7.8) and (7.9), we define the *symmetry error index*

$$\beta(k) \triangleq \beta_1(k) + \beta_2(k), \tag{7.10}$$

corresponding to the sliding window starting at point k, for $k = 0, \ldots, n - N_0$.

For $k = 0, \ldots, n-N_0$, let $k_0 < n-N_0$ be the minimizer of $\beta(k)$. We use knowledge of k_0 to determine the location of the points of symmetry ε and $\varepsilon + \frac{1}{2}N_0$ for the sliding window starting at point k_0. In particular, since k_0 is the starting point of the window that minimizes β and since ε and $\varepsilon + \frac{1}{2}N_0$ are, respectively, the quarter point and three quarter point of the same window, it follows that

$$\varepsilon = k_0 + \frac{1}{4}N_0, \tag{7.11}$$

$$\varepsilon + \frac{1}{2}N_0 = k_0 + \frac{3}{4}N_0. \tag{7.12}$$

Note that, in general, $\beta(k_0) \neq 0$. However if $\dfrac{\angle G(e^{\jmath\Omega_0})}{\pi}$ is an integer, then $\beta(k_0) = 0$, which indicates exact symmetry about $k_0 + \frac{1}{4}N_0$ in the interval $\left[k_0, k_0 + \frac{1}{2}N_0\right]$ and about $k_0 + \frac{3}{4}N_0$ in the interval $\left[k_0 + \frac{1}{2}N_0, k_0 + N_0\right]$.

To illustrate the symmetry search algorithm, we reconsider the example considered in Figures 7.2a and 7.3, where $y = \mathcal{W}(v) = 0.6(v+1)^3 - 1$. Note that $\mathcal{W}$ is not even. Figure 7.4a shows the values of β calculated for $y(k)$ on the interval $[k_0, k_0 + 2N_0]$. Since, in Figure 7.4a, the data window of y is selected to start at $k_0 = \varepsilon - \frac{1}{4}N_0$, the minimum values of $\beta(k)$ occur at k_0 and $k_0 + N_0$, where $k_0 + N_0$ is the start of the next period and, thus, need not be considered. Thus, using the unique minimizer k_0 of $\beta(k)$, it follows that the locations of the points of symmetry are given by (7.11) and (7.12).

Next, for the even nonlinearity $y = \mathcal{W}(v) = v^2$ considered in Figure 7.2b, Figure 7.4b shows the values of $\beta(k)$ calculated for $y(k)$ on the interval $[k_0, k_0 + 2N_0]$. In this case, the minimum values of $\beta(k)$ occur at k_0, $k_0 + \frac{1}{2}N_0$, and $k_0 + N_0$, where $k_0 + N_0$ is the start of the next period and, thus, need not be considered. Thus, using k_0, it follows that the locations of the points of symmetry are given by (7.11) and (7.12). Also, using $k_0 + \frac{1}{2}N_0$, we obtain two additional points of symmetry given by

$$\varepsilon + \frac{1}{4}N_0 = k_0 + \frac{1}{2}N_0, \tag{7.13}$$

$$\varepsilon + \frac{3}{4}N_0 = k_0 + N_0. \tag{7.14}$$

This ambiguity is due to the fact that ε and $\varepsilon + \frac{1}{2}N_0$ are the midpoints of two identical symmetric portions of y. Thus, the start of the data window within which the function has the symmetry properties illustrated in Figure 7.3 can be taken as either k_0 or $k_0 + \frac{1}{2}N_0$. Note that the second minimizer $k_0 + \frac{1}{2}N_0$ appears only for even nonlinearities.

7.3.2 Nonparametric Approximation of the Static Nonlinearity

Using δ, which is assumed to be known from the harmonic input u, and the estimate of ε obtained from y in Section 7.3.1, we now determine an estimate $\hat{\phi}$ of the nonharmonic phase shift of y relative to u by

$$\hat{\phi} \triangleq \Omega_0(\varepsilon - \delta), \tag{7.15}$$

which is an estimate of $\angle G(e^{\jmath\Omega_0})$. Moreover, define the virtual signal

$$\tilde{v}(k) \triangleq A_0 \sin(\Omega_0 k + \hat{\phi}), \tag{7.16}$$

which is an approximation of the intermediate signal v given by (7.5) divided by the constant $|G(e^{\jmath\Omega_0})|$. Note that, if $\hat{\phi} = \angle G(e^{\jmath\Omega_0})$, then $|G(e^{\jmath\Omega_0})|\tilde{v} = v$. Also, note that the amplitude of $\tilde{v}(k)$ is irrelevant due to the scaling factor λ shown in Figure 7.1b.

Using $\tilde{v}$ and y, the nonparametric estimate of $\mathcal{W}$ is given by

$$\hat{\mathcal{W}} \triangleq \{(\tilde{v}(k_0), y(k_0)), (\tilde{v}(k_0+1), y(k_0+1)), \ldots, (\tilde{v}(n), y(n))\}, \tag{7.17}$$

where each pair $(\tilde{v}(k), y(k))$, for $k = k_0, \ldots, n$, determines a value of the nonparametric estimate $\hat{\mathcal{W}}$ of $\mathcal{W}$.

Figure 7.4 shows that, depending on the type of nonlinearity, $\beta(k)$ has either one or two minima within each period. For a non-even polynomial nonlinearity, $\beta(k)$ has one minimum within each period. Therefore, the estimate of the nonharmonic phase shift has two candidate values, namely, $\hat{\phi}$ and $\hat{\phi} + \pi$. For an even nonlinearity, $\beta(k)$ has two minima within each period. Therefore, the estimate of the nonharmonic phase shift has four candidate values, namely, $\hat{\phi}$, $\hat{\phi} + \frac{\pi}{2}$, $\hat{\phi} + \pi$, and $\hat{\phi} + \frac{3\pi}{2}$. However, for the even case, $\hat{\phi}$ and $\hat{\phi} + \pi$ yield the same nonparametric model $\hat{\mathcal{W}}$, while $\hat{\phi} + \frac{\pi}{2}$ and $\hat{\phi} + \frac{3\pi}{2}$ yield the same $\hat{\mathcal{W}}$.

Therefore, for both non-even and even cases, there are two candidate nonparametric estimates of $\mathcal{W}$, both of which are constructed using (7.16) and (7.17). The correct nonparametric model will become apparent when identifying the dynamic block of the Wiener system.

7.4 Parametric Identification of the Linear Time-Invariant Dynamics

Using the nonparametric model $\hat{\mathcal{W}}$ of $\mathcal{W}$, we now identify a model of $\mathcal{L}$ given by $\hat{\mathcal{L}}$ using retrospective cost optimization (RCO) [11]. The RCO algorithm is presented in [11, 53, 54] together with guidelines for choosing its tuning parameters, namely, n_c, p, and α.

Consider the adaptive feedback architecture for $\hat{\mathcal{L}}$ shown in Figure 7.5, where $\hat{\mathcal{L}}_{\mathrm{m}}$ denotes the initial model with input $w \in \mathbb{R}$ and output $\hat{v} \in \mathbb{R}$, and where $\hat{\mathcal{L}}_{\Delta}$ denotes the feedback delta model with inputs u, $\hat{v} \in \mathbb{R}$ and output w.

The goal is to adaptively tune $\hat{\mathcal{L}}_{\Delta}$ so that the performance variable

$$z(k) \stackrel{\triangle}{=} y(k) - \hat{y}(k) \tag{7.18}$$

is minimized in the presence of the identification signal u. For simplicity, we choose $\hat{\mathcal{L}}_{\mathrm{m}}$ to be the one-step delay $1/\mathbf{z}$. Together, $\hat{\mathcal{L}}$ and $\hat{\mathcal{W}}$ comprise a *semiparametric model* of the Wiener system.

From Section 7.3.2, recall that there are two candidates for the nonparametric estimate of $\mathcal{W}$. Thus, we run RCO for each nonparametric estimate of $\mathcal{W}$ and obtain a corresponding parametric model of $\mathcal{L}$. Note that the performance variable z is calculated for both semiparametric models. We choose the semiparametric model whose performance variable has a smaller norm.

7.4.1 Retrospective Cost Optimization

We now review the RCO adaptive control algorithm and show how it is used to identify linear time-invariant dynamic systems using $\hat{\mathcal{W}}$. A detailed discussion of RCO and as well as the theoretical foundations of the algorithm are found in [53, 50, 49].

RCO depends on several parameters that are selected *a priori*. Specifically, n_c is the estimated plant order, $p \geq 1$ is the data window size used to estimate $\hat{\mathcal{L}}_\Delta$, and μ is the number of Markov parameters of $\hat{\mathcal{L}}_\Delta$. The methodology for choosing these parameters is as follows. n_c is overestimated, that is, chosen to be greater than the expected order of $\hat{\mathcal{L}}$. From Section 7.4, recall that we assume that the controller $\hat{\mathcal{L}}_\Delta$ is placed in feedback with a unit delay. Therefore, there is only one nonzero Markov parameter, so $\mu = 1$ in all example cases. The adaptive update law is based on a quadratic cost function, which involves a time-varying weighting parameter $\alpha(k) > 0$, referred to as the *learning rate* since it affects the convergence speed of the adaptive control algorithm. In [53], RCO is presented for MIMO systems, where l_u, l_v, l_w, and l_y denote the sizes of u, v, w, and y, respectively. However, in this chapter, we consider only the SISO system (7.1)-(7.3). For convenience, we keep the notation of [53] and set $l_u = l_v = l_w = l_y = 1$.

Let $\hat{\mathcal{L}}_{\mathrm{m}} \sim \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$ as given by (7.1), (7.2), where $x(k) \in \mathbb{R}^{l_x}$, $A \in \mathbb{R}^{l_x \times l_x}$, $B \in \mathbb{R}^{l_x \times 1}$, $C \in \mathbb{R}^{1 \times l_x}$. Since $\hat{\mathcal{L}}_{\mathrm{m}}$ is set as unit delay, it follows that $A = 0_{l_x \times l_x}$, $B = 1_{l_x \times 1}$, and $C = 1_{1 \times l_x}$, yielding

$$\hat{v}(k) = w(k-1),$$

where $w(k-1)$ is the output of $\hat{\mathcal{L}}_\Delta$, which was obtained using RCO in the previous iteration. Note that, to compute (7.18), $\hat{y}(k)$ is assumed to be known. To accomplish

that, we use the estimated intermediate signal $\hat{v}(k)$ with $\hat{\mathcal{W}}$ as follows. Note that, in general, $\hat{v}(k)$ is not in the set defined by (7.17). We thus suggest two methods by which this issue may be overcome. For simplicity, the first case is to use the closest value of $\tilde{v}(k)$ in the set (7.17) to $\hat{v}(k)$. Second, interpolation between the closest bounding values may be used. For convenience, henceforth, we use the first method.

Next, to compute $w(k)$ we use an exactly proper time-series controller of order n_{c} such that the control $w(k)$ is given by

$$w(k) = \sum_{i=1}^{n_{\mathrm{c}}} M_i(k) w(k-i) + \sum_{i=0}^{n_{\mathrm{c}}} N_i(k) \left[\begin{array}{c} \hat{v}(k-i) \\ u(k-i) \end{array} \right], \tag{7.19}$$

where $M_i \in \mathbb{R}^{l_w \times l_w}$, $i = 1, \ldots, n_{\mathrm{c}}$, and $N_i \in \mathbb{R}^{l_w \times (l_v + l_u)}$, $i = 0, \ldots, n_{\mathrm{c}}$, are given by an adaptive update law. Note that the ARX model given in (7.19) is a model of $\hat{\mathcal{L}}_{\Delta}$. The control can be expressed as

$$w(k) = \theta(k) \psi(k), \tag{7.20}$$

where

$$\theta(k) \triangleq \left[\begin{array}{cccccc} N_0(k) & \cdots & N_{n_{\mathrm{c}}}(k) & M_1(k) & \cdots & M_{n_{\mathrm{c}}}(k) \end{array} \right]$$

is the *controller parameter block matrix* and the *regressor vector* $\psi(k)$ is given by

$$\psi(k) \triangleq \begin{bmatrix} \hat{v}(k) \\ \vdots \\ \hat{v}(k-n_{\rm c}) \\ u(k) \\ \vdots \\ u(k-n_{\rm c}) \\ w(k-1) \\ \vdots \\ w(k-n_{\rm c}) \end{bmatrix} \in \mathbb{R}^{n_{\rm c}l_w+(n_{\rm c}+1)(l_v+l_u)}.$$

For positive integers p and μ, we define the *extended performance vector* $Z(k)$ and the *extended control vector* $W(k)$ by

$$Z(k) \triangleq \begin{bmatrix} z(k) \\ \vdots \\ z(k-p+1) \end{bmatrix}, \quad W(k) \triangleq \begin{bmatrix} w(k) \\ \vdots \\ w(k-p_{\rm c}+1) \end{bmatrix}, \tag{7.21}$$

where $p_{\rm c} \triangleq \mu + p$.

From (7.20), it follows that the extended control vector $W(k)$ can be written as

$$W(k) \triangleq \sum_{i=1}^{p_{\rm c}} L_i \theta(k-i+1)\psi(k-i+1), \tag{7.22}$$

where

$$L_i \triangleq \begin{bmatrix} 0_{(i-1)l_w \times l_w} \\ I_{l_w} \\ 0_{(p_{\rm c}-i)l_w \times l_w} \end{bmatrix} \in \mathbb{R}^{p_{\rm c}l_w \times l_w}. \tag{7.23}$$

We define the *surrogate performance vector* $\hat{Z}(\hat{\theta}(k),k)$ by

$$\hat{Z}(\hat{\theta}(k),k) \triangleq Z(k) - \bar{B}_{zw}\left(W(k) - \hat{W}(k)\right), \tag{7.24}$$

where

$$\hat{W}(k) \triangleq \sum_{i=1}^{p_c} L_i \hat{\theta}(k)\psi(k-i+1), \tag{7.25}$$

and $\hat{\theta}(k) \in \mathbb{R}^{l_w \times [n_c l_w + (n_c+1)(l_v+l_u)]}$ is the *surrogate controller parameter block matrix*. The block-Toeplitz *surrogate control matrix* $\bar{B}_{zw}$ is given by

$$\bar{B}_{zw} \triangleq \begin{bmatrix} 0_{l_z\times l_w} & \cdots & 0_{l_z\times l_w} & H_d & \cdots & H_\mu & 0_{l_z\times l_w} & \cdots & 0_{l_z\times l_w} \\ 0_{l_z\times l_w} & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0_{l_z\times l_w} \\ 0_{l_z\times l_w} & \cdots & 0_{l_z\times l_w} & 0_{l_z\times l_w} & \cdots & 0_{l_z\times l_w} & H_d & \cdots & H_\mu \end{bmatrix}.$$

where the *relative degree* d is the smallest positive integer i such that the ith Markov parameter $H_i = CA^{i-1}B$ of $\hat{\mathcal{L}}_{\mathrm{m}}$ is nonzero. The leading zeros in $\bar{B}_{zw}$ account for the nonzero relative degree d. The algorithm places no constraints on either the value of d or the rank of H_d or $\bar{B}_{zw}$. For the SISO case when $\hat{\mathcal{L}}_{\mathrm{m}}$ is a unit delay,

$$\bar{B}_{zw} = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}. \tag{7.26}$$

Furthermore, we define

$$D(k) \triangleq \sum_{i=1}^{n_c+\mu-1} \psi^{\mathrm{T}}(k-i+1) \otimes L_i, \tag{7.27}$$

$$f(k) \triangleq Z(k) - \bar{B}_{zw} W(k). \tag{7.28}$$

We now consider the cost function

$$J(\hat{\theta},k) \triangleq \hat{Z}^{\mathrm{T}}(\hat{\theta},k)R_1(k)\hat{Z}(\hat{\theta},k) + \mathrm{tr}\left[R_2(k)\left(\hat{\theta}-\theta(k)\right)^{\mathrm{T}} R_3(k)\left(\hat{\theta}-\theta(k)\right)\right], \quad (7.29)$$

where $R_1(k) \triangleq I_{pl_z}$, $R_2(k) \triangleq \alpha(k)I_{n_c(l_w+(l_v+l_u))}$, and $R_3(k) \triangleq I_{l_w}$. Note that the cost function is quadratic in the retrospective term $\hat{Z}$, while the second term penalizes the difference $\theta(k+1)-\theta(k)$; therefore, R_2 and R_3 can be used to control how much the controller parameters will change in a given step.

Substituting (7.24) and (7.25) into (7.29), J is written as the quadratic form

$$J(\hat{\theta},k) = c(k) + b^{\mathrm{T}}\mathrm{vec}\ \hat{\theta} + \left(\mathrm{vec}\ \hat{\theta}\right)^{\mathrm{T}} A(k)\mathrm{vec}\ \hat{\theta}, \quad (7.30)$$

where

$$A(k) = D^{\mathrm{T}}(k)D(k) + \alpha(k)I, \quad (7.31)$$

$$b(k) = 2D^{\mathrm{T}}(k)f(k) - 2\alpha(k)\mathrm{vec}\ \theta(k), \quad (7.32)$$

$$c(k) = f(k)^{\mathrm{T}}R_1(k)f(k) + \mathrm{tr}\left[R_2(k)\theta^{\mathrm{T}}(k)R_3(k)\theta(k)\right]. \quad (7.33)$$

Since $A(k)$ is positive definite, $J(\hat{\theta},k)$ has the strict global minimizer

$$\hat{\theta}(k) = \frac{1}{2}\mathrm{vec}^{-1}(A(k)^{-1}b(k)). \quad (7.34)$$

The controller gain update law is

$$\theta(k+1) = \hat{\theta}(k), \quad (7.35)$$

such that $w(k)$ is computed using (7.20). The key feature of the adaptive control algorithm (7.20) is the surrogate performance variable $Z(k)$ based on the difference

between the actual past control inputs $W(k)$ and the recomputed past control inputs based on the current control law $\hat{W}(k)$. The parameter α is chosen to be as small as possible while guaranteeing that $A(k)$ is positive definite.

7.5 Numerical Examples: Nominal Case

To demonstrate semiparametric model identification, we consider various static nonlinearities. For each example, we choose G to have poles $0.34 \pm 0.87\jmath, -0.3141 \pm 0.9\jmath, 0.05 \pm 0.3122\jmath, -0.6875$ and zeros $0.14 \pm 0.97\jmath, -0.12 \pm 0.62\jmath, -0.89$ with monic numerator and denominator. Also, $u(k)$ is chosen to be a realization of zero-mean Gaussian white noise with standard deviation $\sigma_u = 3.5$.

Note that A_0 should, in practice, be chosen to be greater than the expected operating range of the Wiener system. This guarantees that the inputs to the model can be interpolated from the nonparametric map. For the following examples we choose m to be much larger than required. Although we show in Section 7.7.2 that very little performance gain is attainted from choosing m large, it is visually easier to compare the identified nonparametric map to the true nonlinearity when using more data points. Finally, the parameter $\alpha(k)$ discussed in the previous section is chosen as a constant value for all examples. We choose varying values for $\alpha(k)$ to demonstrate that the final estimate of the Wiener system is not sensitive to this parameter.

Example 7.5.1. (Non-even Polynomial) Consider $\mathcal{W}$ defined by

$$y = \mathcal{W}(v) = v^3 + 4v + 7. \tag{7.36}$$

The parameters for nonparametric identification of $\mathcal{W}$ are $m = 500$ and $A_0 = 5$. Figure 7.6 compares the true and identified nonlinearities. The RCO parameters used to identify the linear dynamic system are set as $n_c = 9$, $p = 1$, and $\alpha = 1$. Figure 7.7 shows the frequency response of the true dynamic model G and the identified model

using RCO with the identified nonlinearity shown in Figure 7.6.

Example 7.5.2. (Even Polynomial) Consider $\mathcal{W}$ defined by

$$y(k) = \mathcal{W}(v) = 7v^4 + v^2. \tag{7.37}$$

The parameters for nonparametric identification of $\mathcal{W}$ are $m = 500$ and $A_0 = 5$. Figure 7.8 compares the true and identified nonlinearities. The RCO parameters used to identify the linear dynamic system are set as $n_c = 9$, $p = 1$, and $\alpha = 50$. Figure 7.9 shows the frequency response of G and the identified model using RCO with the identified nonlinearity shown in Figure 7.8.

Next, to illustrate the ambiguity discussed in Section 7.3.2, we select the incorrect nonharmonic phase shift, specifically, $\hat{\phi} + \frac{\pi}{2}$. Figure 7.10 shows a comparison of the true and identified nonlinearities. Note that the incorrect nonharmonic phase shift produces an erroneous nonparametric model of the nonlinearity. Figure 7.11 shows a frequency response comparison of G and the model identified using RCO with the identified nonlinearity shown in Figure 7.10.

To determine the appropriate phase shift $\hat{\phi}$ or $\hat{\phi} + \frac{\pi}{2}$, we examine the performance variable z given by (7.18), which provides insight into which candidate value yields the correct semiparametric model. The upper plot of Figure 7.12 shows the RCO performance variable z for the incorrect nonparametric model of $\mathcal{W}$, while the lower plot shows the performance variable for the correct nonparametric model of $\mathcal{W}$. The correct semiparametric model clearly outperforms the incorrect model.

7.6 Numerical Examples: Off-Nominal Cases

We now reconsider the Wiener system (7.1)-(7.3) with noise, as shown in Figure 7.13. The input $u(k)$ is a realization of zero-mean Gaussian white noise with standard deviation $\sigma_u = 3.5$, while $d_1(k) \in \mathbb{R}$ and $d_2(k) \in \mathbb{R}$ are unknown zero-mean Gaussian

white disturbances with standard deviations σ_{d_1} and σ_{d_2}, respectively. The output

$$y(k) = \mathcal{W}(v(k)) + d_3(k), \tag{7.38}$$

has standard deviation σ_y about its mean, and $d_3(k) \in \mathbb{R}$ is an unknown zero-mean Gaussian white disturbance with standard deviation σ_{d_3}. The disturbance signals $d_1(k)$, $d_2(k)$, and $d_3(k)$ are process, input, and output noise, respectively.

We now consider additional static nonlinearities, where, for each example, we choose G as in Section 7.5.

Example 7.6.1. (Deadzone) Consider $\mathcal{W}$ defined by

$$y = \mathcal{W}(v) = \begin{cases} 0, & \text{if } |v| \leq 0.17; \\ v, & \text{if } |v| > 0.17. \end{cases} \tag{7.39}$$

Furthermore, we consider process and output noise $\sigma_{d_1} = \frac{1}{15}\sigma_u$, $\sigma_{d_3} = \frac{1}{15}\sigma_y$ and $d_2 = 0$. For this problem, the parameters for nonparametric identification are $m = 250$ and $A_0 = 5$. In this example we also parameterize the estimated nonlinearity for comparison with the nonparametric estimate. The parametric model is a 25^{th} order polynomial. Figure 7.14 compares the true, nonparametric identified and parametric identified nonlinearities. The RCO parameters used to identify the linear dynamic system are set as $n_{\text{c}} = 9$, $p = 1$, and $\alpha = 10$. Figure 7.15 shows the frequency response of G and the identified model using RCO with the identified nonparametric model of the nonlinearity. Figure 7.16 shows the frequency response of G and the identified model using RCO with the identified parametric model of the nonlinearity. Figure 7.17 compares the output of the Wiener system $y(k)$ and the output of the estimated semiparametric Wiener model $\hat{y}(k)$, in response to a random input. Figure 7.18 compares the output of the Wiener system $y(k)$ and the output of the estimated parametric Wiener model $\hat{y}(k)$, in response to a random input.

Figure 7.19 is the difference between the error in the semiparametric Wiener model and the parametric Wiener model. Where the graph is negative, the semiparametric model has superior performance, and where the graph is positive the parametric model is superior.

Example 7.6.2. (Saturation) Consider $\mathcal{W}$ defined by

$$y = \mathcal{W}(v) = \begin{cases} 8.64(v + 0.23) - 3.98, & \text{if } 0.1 < v < 0.4; \\ 1.5, & \text{if } v \geq 0.4; \\ -1.2, & \text{if } v \leq 0.1. \end{cases} \tag{7.40}$$

Furthermore, we consider input noise $\sigma_{d_1} = \frac{1}{8}\sigma_u$ and $d_2 = d_3 = 0$. The parameters for nonparametric identification are $m = 150$ and $A_0 = 5$. Figure 7.20 compares the true and identified nonlinearities. The RCO parameters used to identify the linear dynamic system are set as $n_c = 9$, $p = 1$, and $\alpha = 1$. Figure 7.21 shows the frequency response of G and the identified model using RCO with the identified nonlinearity shown in Figure 7.20.

Example 7.6.3. (Switch function) Consider $\mathcal{W}$ defined by

$$y = \mathcal{W}(v) = \begin{cases} 0, & \text{if } |v| = 0; \\ 8.64v + \text{sgn}(v)4.5, & \text{if } 0 < |v| \leq 1.5. \end{cases} \tag{7.41}$$

Furthermore, we consider process, input, and output noise $\sigma_{d_1} = \frac{1}{15}\sigma_u$, $\sigma_{d_2} = \frac{1}{15}\sigma_w$, and $\sigma_{d_3} = \frac{1}{15}\sigma_y$. The parameters for nonparametric identification are $m = 100$ and $A_0 = 5$. Figure 7.22 compares the true and identified nonlinearities. The RCO parameters used to identify the linear dynamic system are set as $n_c = 9$, $p = 1$, and $\alpha = 1$. Figure 7.23 shows the frequency response of G and the identified model using RCO with the identified nonlinearity shown in Figure 7.22.

Example 7.6.4. (Stairs Function) Consider $\mathcal{W}$ defined by

$$y = \mathcal{W}(v) = \begin{cases} 0, & \text{if } |v| = 0; \\ \text{sgn}(v)1, & \text{if } 0 < |v| \leq 0.17; \\ \text{sgn}(v)3, & \text{if } 0.17 < |v| \leq 0.35; \\ \text{sgn}(v)4.5, & \text{if } 0.35 < |v| \leq 0.52; \\ \text{sgn}(v)6, & \text{if } 0.52 < |v|; \end{cases} \tag{7.42}$$

Furthermore, we consider process, input, and output noise $\sigma_{d_1} = \frac{1}{8}\sigma_u$, $\sigma_{d_2} = \frac{1}{8}\sigma_w$, and $\sigma_{d_3} = \frac{1}{8}\sigma_y$. The parameters for nonparametric identification are $m = 75$ and $A_0 = 5$. Figure 7.24 compares the true and identified nonlinearities. The RCO parameters used to identify the linear dynamic system are set as $n_c = 9$, $p = 1$, and $\alpha = 1$. Figure 7.25 is a frequency response comparison of G and the system identified using RCO with the identified nonlinearity shown in Figure 7.24. Figure 7.26 compares the output of the Wiener system $y(k)$ and the output of the estimated semiparametric Wiener model $\hat{y}(k)$, in response to a random input.

7.7 Numerical Examples: Error Metrics

We now investigate the effect of systematically decreasing the amount of available output data that is used to identify the linear block of the Wiener system. Moreover, we investigate the effect of decreasing m, which determines the number of points in the nonparametric model, and therefore affects the fidelity of $\hat{\mathcal{W}}$.

To quantify the accuracy of the identified semiparametric model, we compute the root-mean-square error (RMSE) for the first 15 Markov parameters of the true linear system and the identified linear system. The linear model is the same as in Sections 7.5 and 7.6, while $\mathcal{W}$ is given by (7.36).

7.7.1 Effect of Disturbances

To evaluate the effect of σ_{d_1}, σ_{d_2}, and σ_{d_3}, we decrease the number of available data points from 4000 to 10. For each case, we perform a 100-run Monte Carlo simulation with a signal-to-noise ratio of 10. We consider the effect of d_1, d_2, and d_3 individually, as well as the effect of all three noise signals, which may be uncorrelated or correlated. Furthermore we consider when d_1 and d_3 are correlated, and d_2 and d_3 are correlated.

Figure 7.27 demonstrates the increase in error for decreasing amounts of available data. Furthermore, we see that the cases with correlated disturbances yield similar results compared to the case with uncorrelated disturbances.

7.7.2 Nonparametric Model Accuracy

We now perform a Monte Carlo simulation to evaluate how m affects the accuracy of the identified linear system. Specifically, we vary m from 1 to 100. For each value of m we average the result over 100 simulations. We consider the nominal case, that is, without noise.

Figure 7.28 shows that RMSE generally decreases as m increases. Note that, for this example, only a slight decrease in RMSE is observed for $m \geq 20$.

7.8 Conclusions

In this chapter we develop a two-step method to identify semiparametric models for SISO discrete-time Wiener systems. We make only two assumptions about the system, namely, the linear dynamic block is assumed to be asymptotically stable, and the static nonlinearity is assumed to be piecewise continuous. Furthermore, this method requires identification signals with specific properties for each of the two steps, as discussed as follows.

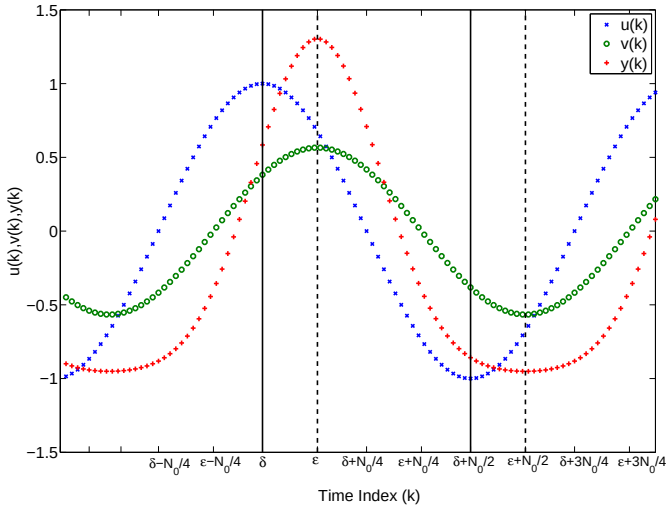
First, we choose a single harmonic input and measure the system output when

the state trajectory is in harmonic steady state. By exploiting symmetry properties of these signals, we approximate the nonharmonic phase shift and, therefore, estimate the intermediate signal. Using the estimate of the intermediate signal, a nonparametric model of the static nonlinearity is obtained.

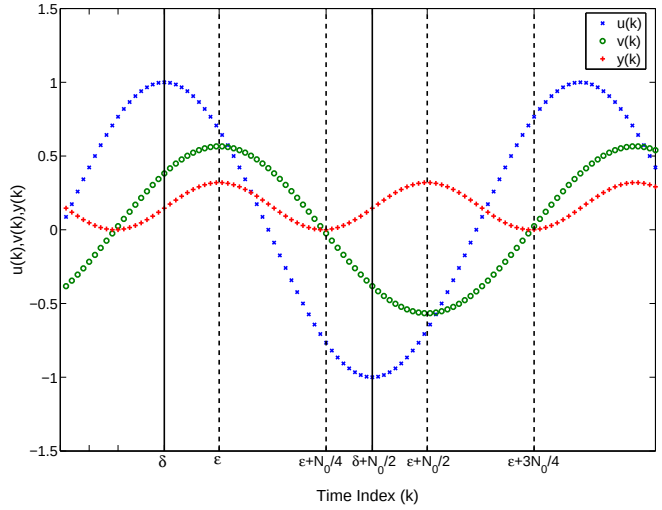
Second, using the identified nonparametric model, we use retrospective cost optimization to identify a parametric model of the dynamic system. As commonly assumed in the system identification literature, the identification signal for this step is assumed to be sufficiently persistent such that the dynamic linear system can be identified.

It is important to point out that the method investigated in this work does not require invertibility of the nonlinearity, which is a common assumption in Wiener identification. However, the cost of removing this assumption is the need for two steps, and the requirement that the signal for the first step be a single harmonic. Furthermore, the user must wait until the system has reached a steady state before useful data can be obtained. On the other hand, from Section 7.1, recall that there are methods based on multiple harmonic inputs in the literature. Finally, it should be noted that, although a nonparametric model of the nonlinearity was used in this discussion, the data which represents the nonparametric map could be parameterized.

The twos-step method presented in this chapter is effectively demonstrated on several examples of increasing complexity, including nonlinearities in the form of both even and non-even polynomials, deadzone, saturation, and discontinuity, and disturbances on the form of process, input, and output noise.



(a)



(b)

Figure 7.2: Illustration of the symmetry properties of the signals u, v, and y given by (7.4)-(7.6), respectively, for (a) the non-even polynomial nonlinearity $y = \mathcal{W}(v) = 0.6(v+1)^3 - 1$ and (b) the even polynomial nonlinearity $y = \mathcal{W}(v) = v^2$. The signals u and v are harmonic, whereas y is the output of the nonlinear block $\mathcal{W}$ and thus is not harmonic. Note that, for both cases, u is symmetric about δ in the interval $\left[\delta - \frac{1}{4}N_0, \delta + \frac{1}{4}N_0\right]$ and about $\delta + \frac{1}{2}N_0$ in the interval $\left[\delta + \frac{1}{4}N_0, \delta + \frac{3}{4}N_0\right]$, while v and y are symmetric about ε in the interval $\left[\varepsilon - \frac{1}{4}N_0, \varepsilon + \frac{1}{4}N_0\right]$ and about $\varepsilon + \frac{1}{2}N_0$ in the interval $\left[\varepsilon + \frac{1}{4}N_0, \varepsilon + \frac{3}{4}N_0\right]$. In addition, for the case of an even polynomial nonlinearity shown in (b), y is also symmetric about $\varepsilon + \frac{1}{4}N_0$ in the interval $\left[\varepsilon, \varepsilon + \frac{1}{2}N_0\right]$ and about $\varepsilon + \frac{3}{4}N_0$ in the interval $\left[\varepsilon + \frac{1}{2}N_0, \varepsilon + N_0\right]$.

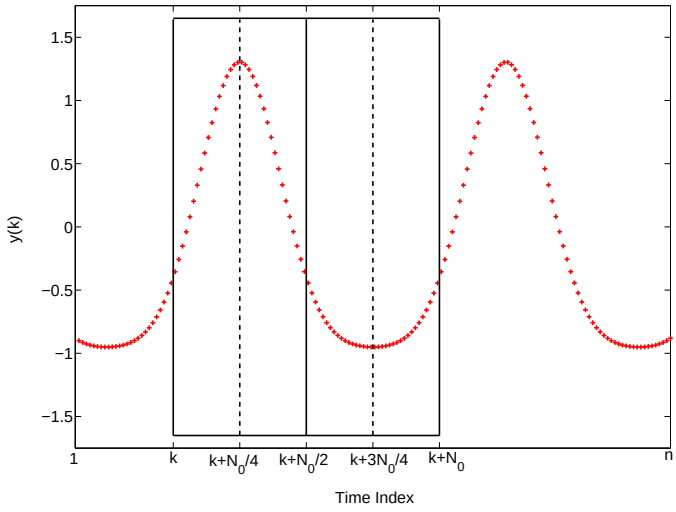
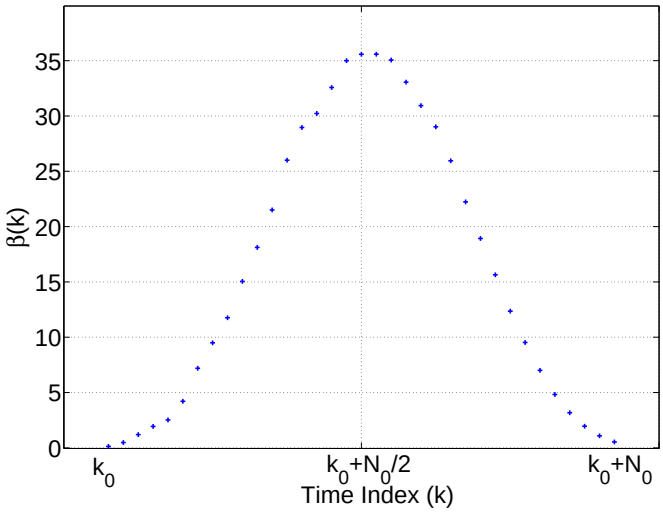
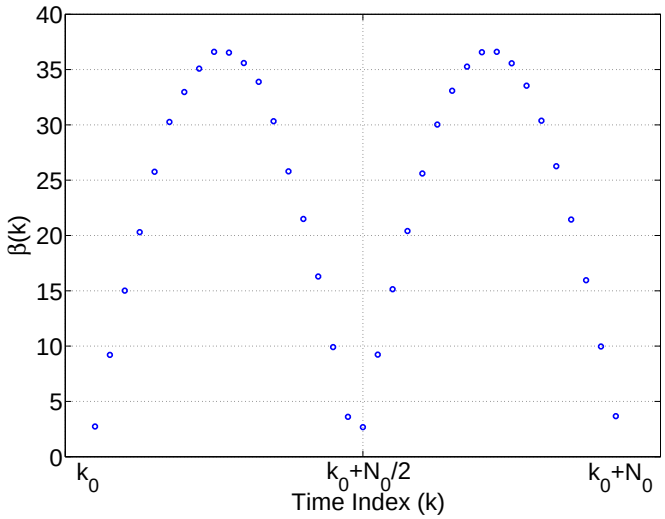


Figure 7.3: Illustration of the symmetry search algorithm. The solid line box comprises the sliding window of length $N_0 + 1$ starting at time k, while the dashed lines indicate the windowed points of symmetry.



(a)



(b)

Figure 7.4: Illustration of the symmetry error index $\beta(k)$ given by (7.8). The values of $\beta(k)$ are shown for two static nonlinearities, namely, (a) a non-even polynomial and (b) an even polynomial.

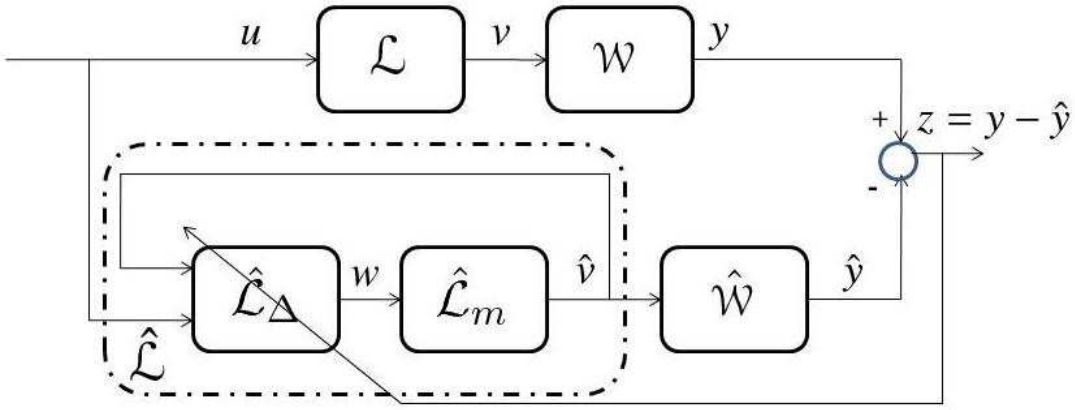


Figure 7.5: Identification architecture for Wiener models using retrospective cost optimization.

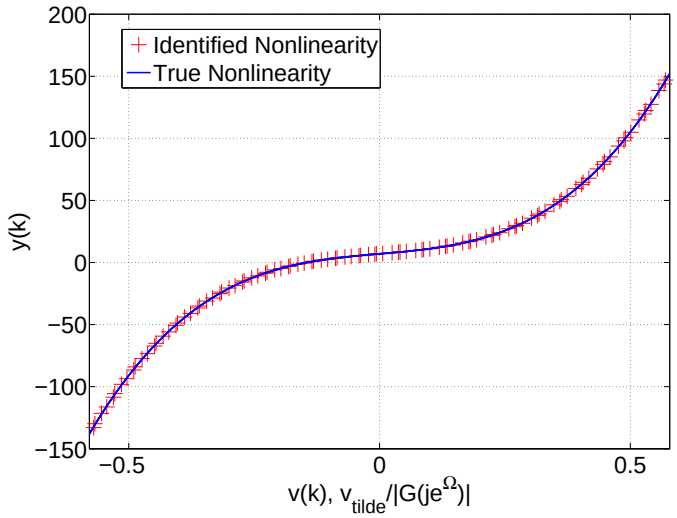


Figure 7.6: Identified nonlinearity versus true nonlinearity (7.36), where $m = 500$ and $A_0 = 5$ (Example 7.5.1). The argument of the identified nonlinearity is scaled by $\frac{1}{|G(e^{\jmath\Omega_0})|}$ to facilitate comparison with the true nonlinearity.

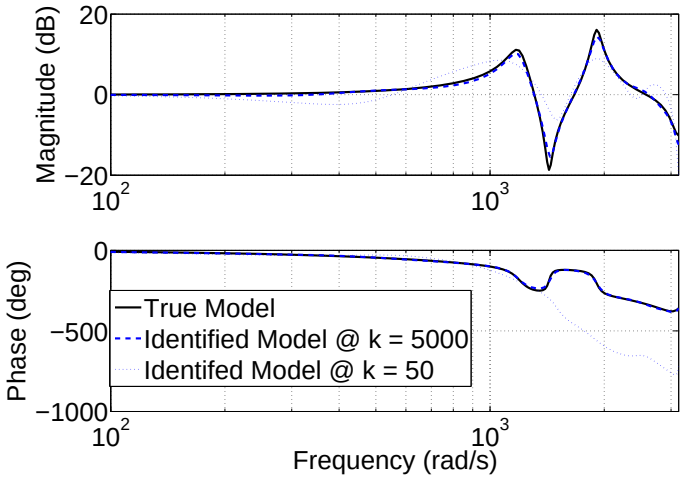


Figure 7.7: Frequency response comparison of the true G and the identified LTI system obtained using $\hat{\mathcal{W}}$ as an estimate of (7.36), where k is the number of data points used to determine the identified dynamic model. The RCO controller order is $n_c = 9$ with $p = 1$ and $\alpha = 1$ (Example 7.5.1).

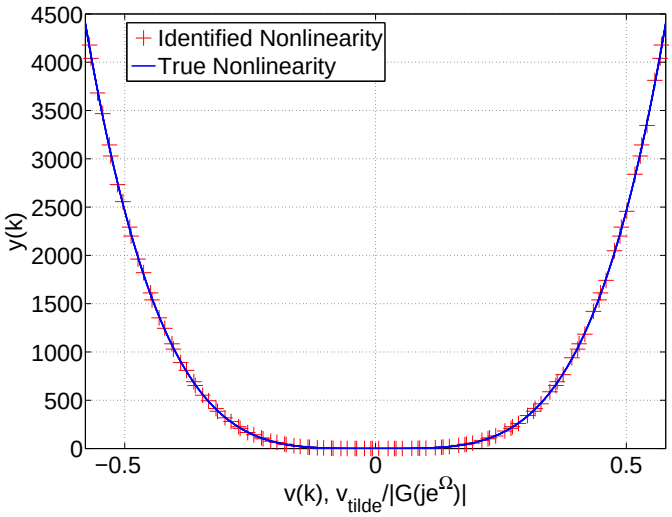


Figure 7.8: Identified nonlinearity versus true nonlinearity (7.37), where $m = 500$ and $A_0 = 5$ (Example 7.5.2).

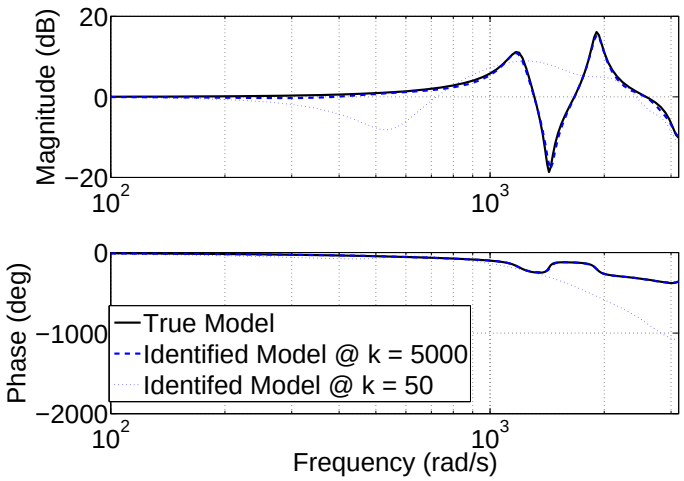


Figure 7.9: Frequency response comparison of the true G and the identified LTI system obtained using $\hat{\mathcal{W}}$ as an estimate of (7.37), where k is the number of data points used to determine the identified dynamic model. The RCO controller order is $n_c = 9$ with $p = 1$, and $\alpha = 50$ (Example 7.5.2).

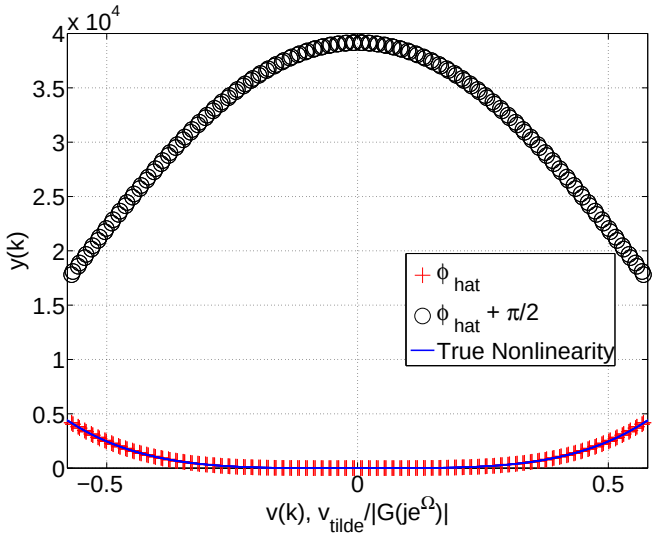


Figure 7.10: Identified nonlinearity versus true nonlinearity (7.37), where $m = 500$ and $A_0 = 5$ (Example 7.5.2). Both candidate values for the nonharmonic phase shift, namely, $\hat{\phi}$ and $\hat{\phi} + \frac{\pi}{2}$, are used to build the two candidate identified nonlinearities.

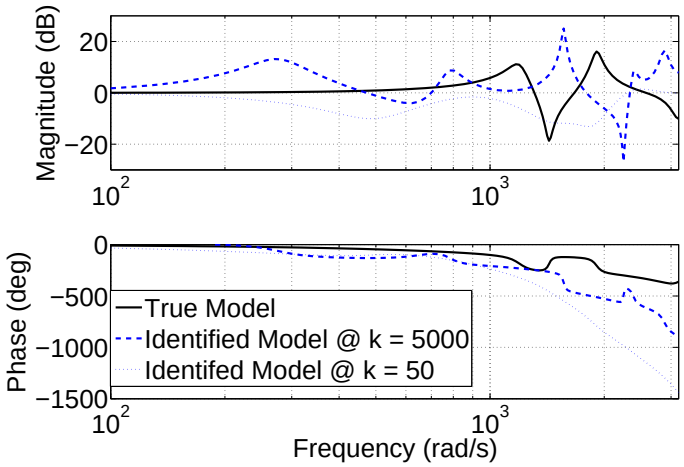


Figure 7.11: Frequency response comparison of the true G and the identified LTI system obtained using $\hat{\mathcal{W}}$ corresponding to the incorrect phase shift as an estimate of (7.37), where k is the number of data points used to determine the identified dynamic model. The RCO controller order is $n_c = 9$ with $p = 1$, and $\alpha = 50$ (Example 7.5.2).

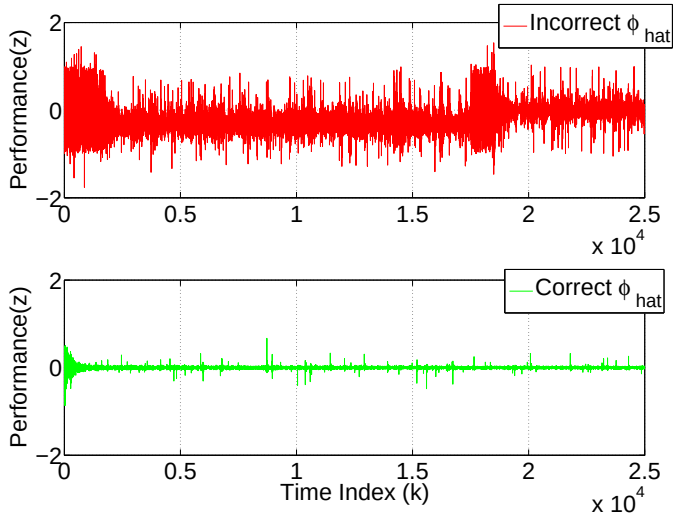


Figure 7.12: Retrospective optimization performance comparison for Example 7.5.2. The upper plot shows the performance variable z for the case in which the nonparametric model is generated using the incorrect candidate for the nonharmonic phase shift $\hat{\phi} + \frac{\pi}{2}$. The lower plot shows z for the case in which the correct candidate $\hat{\phi}$ is used.

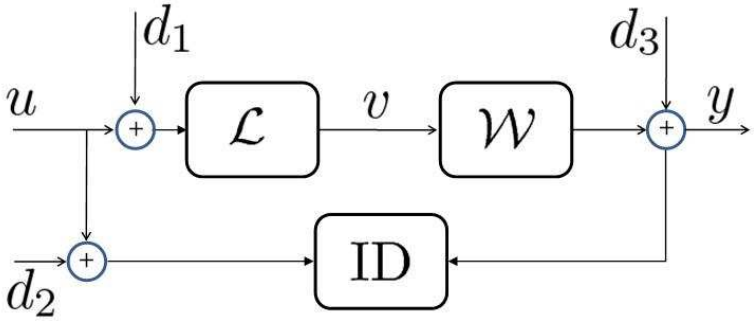


Figure 7.13: Block-structured Wiener model with process, input, and output noise, where d_1, d_2, and d_3 are unknown zero-mean Gaussian disturbances.

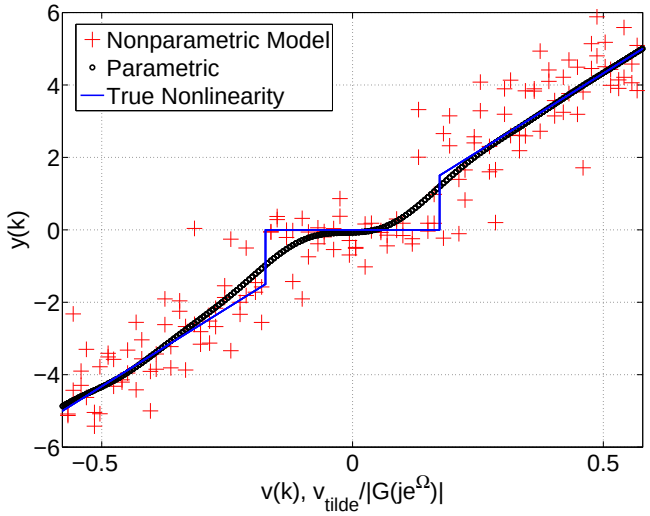


Figure 7.14: Identified nonlinearity versus true nonlinearity (7.39), where $m = 250$ and $A_0 = 5$. In this example, we also parameterize the estimated nonlinearity using a 25^{th} order polynomial. (Example 7.6.1).

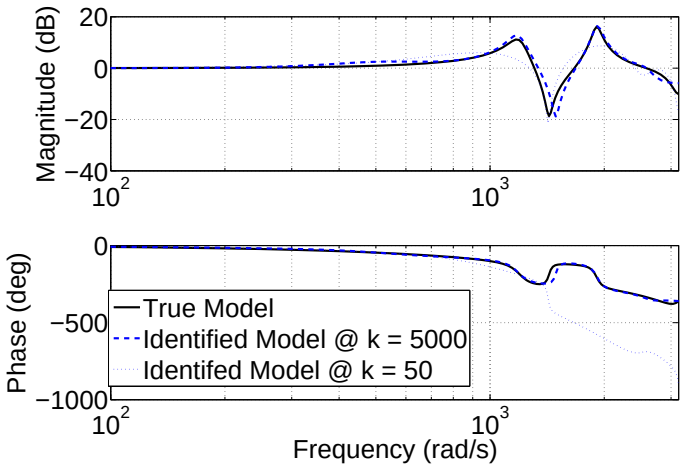


Figure 7.15: Frequency response comparison of the true G and the identified LTI system obtained using the nonparametric $\hat{\mathcal{W}}$ as an estimate of (7.39), where k is the number of data points used to determine the identified dynamic model. The RCO controller order is $n_c = 9$ with $p = 1$ and $\alpha = 10$. (Example 7.6.1).

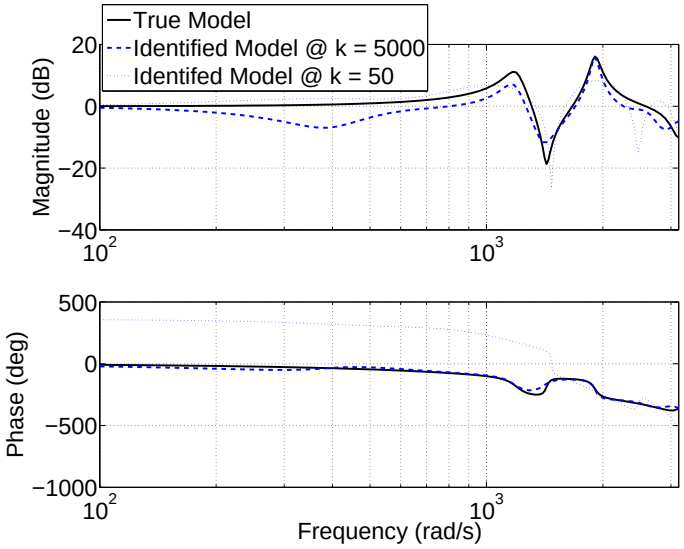


Figure 7.16: Frequency response comparison of the true G and the identified LTI system obtained using the parametric $\hat{\mathcal{W}}$ as an estimate of (7.39), where k is the number of data points used to determine the identified dynamic model. The RCO controller order is $n_c = 9$ with $p = 1$ and $\alpha = 10$ (Example 7.6.1).

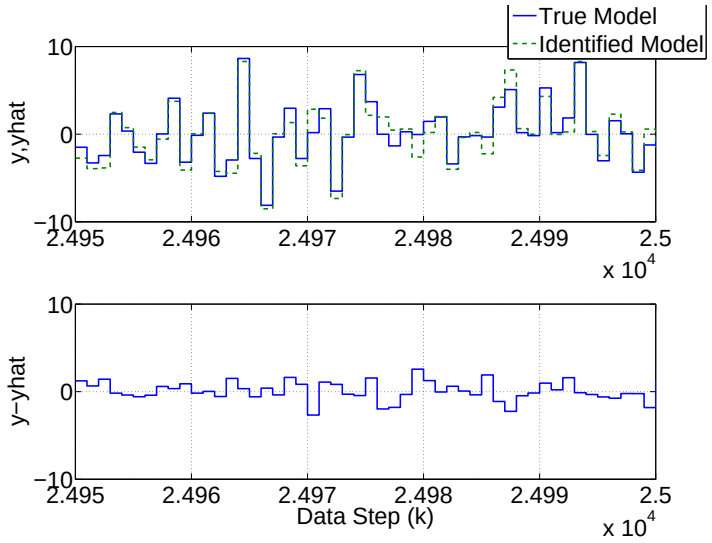


Figure 7.17: Performance comparison for Example 7.6.1 using the nonparametric estimate of the nonlinearity. The top plot is the output of the Wiener system $y(k)$, and the output of the estimated system $\hat{y}(k)$. The bottom plot is the performance $z(k)$.

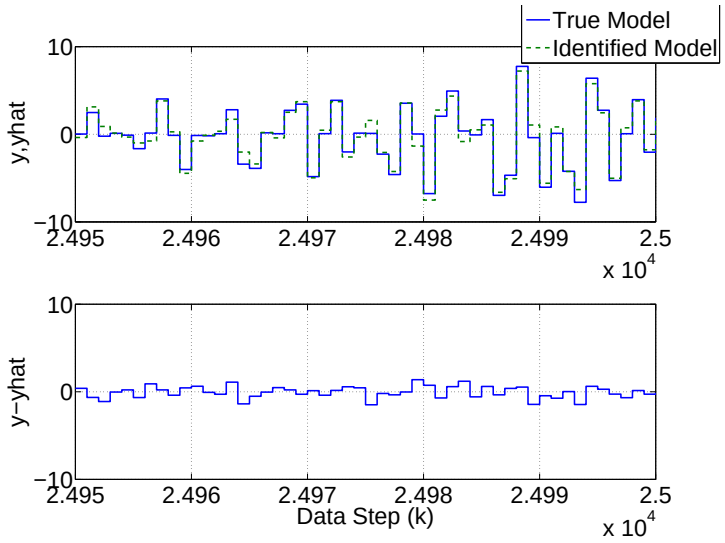


Figure 7.18: Performance comparison for Example 7.6.1 using the parametric estimate of the nonlinearity. The top plot is the output of the Wiener system $y(k)$, and the output of the estimated system $\hat{y}(k)$. The bottom plot is the performance $z(k)$.

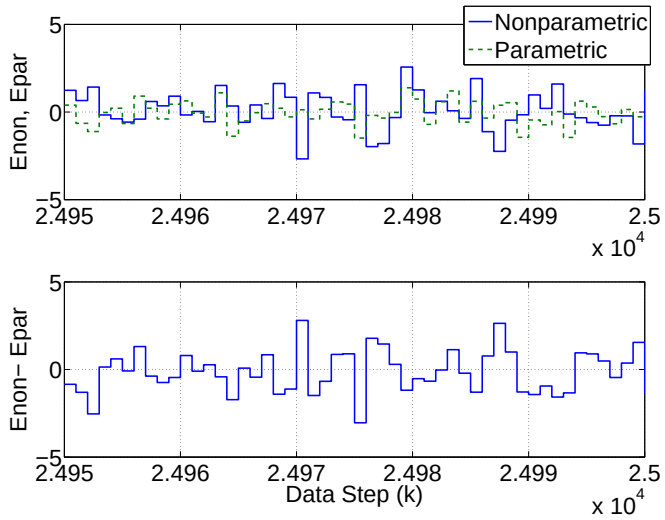


Figure 7.19: This plot is the difference between the error in the semiparametric Wiener model and the parametric Wiener model. Where the graph is negative, the semiparametric model has superior performance, and where the graph is positive the parametric model is superior.

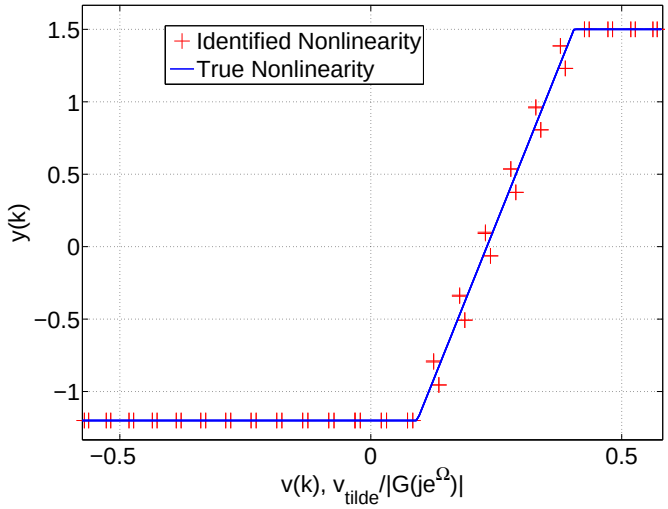


Figure 7.20: Identified nonlinearity versus true nonlinearity (7.40), where $m = 150$ and $A_0 = 5$ (Example 7.6.2).

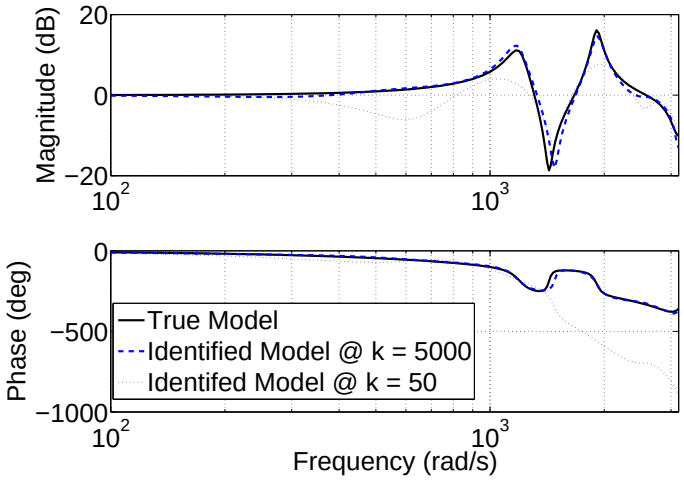


Figure 7.21: Frequency response comparison of the true G and the identified LTI system obtained using $\hat{\mathcal{W}}$ as an estimate of (7.40), where k is the number of data points used to determine the identified dynamic model. The RCO controller order is $n_c = 9$ with $p = 1$ and $\alpha = 1$ (Example 7.6.2).

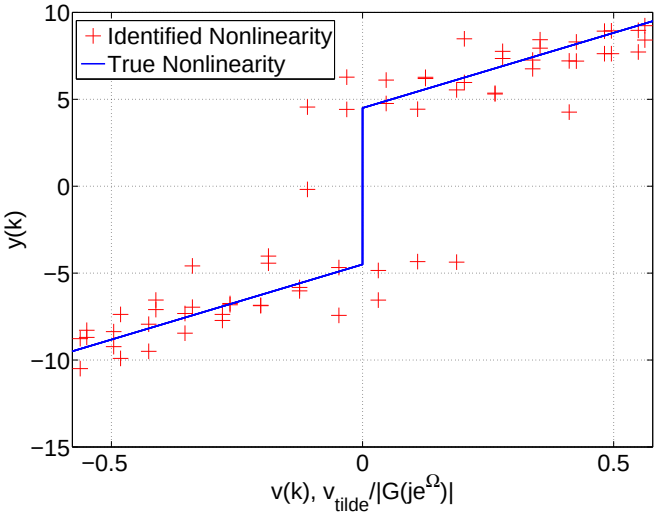


Figure 7.22: Identified nonlinearity versus true nonlinearity (7.41), where $m = 100$ and $A_0 = 5$ (Example 7.6.3).

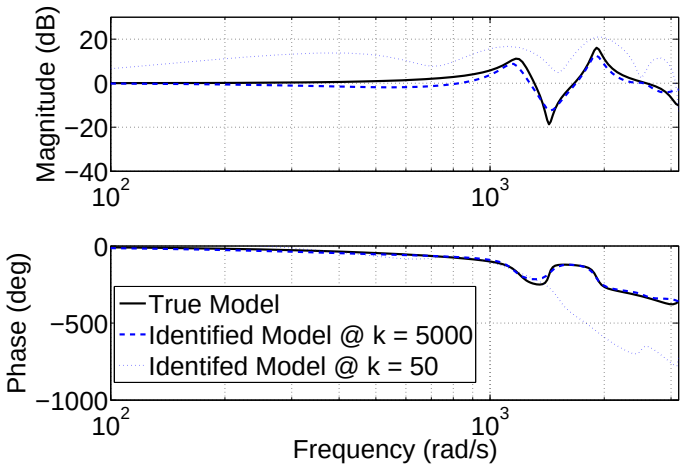


Figure 7.23: Frequency response comparison of the true G and the identified LTI system obtained using $\hat{\mathcal{W}}$ as an estimate of (7.41), where k is the number of data points used to determine the identified dynamic model. The RCO controller order is $n_c = 9$ with $p = 1$ and $\alpha = 1$ (Example 7.6.3).

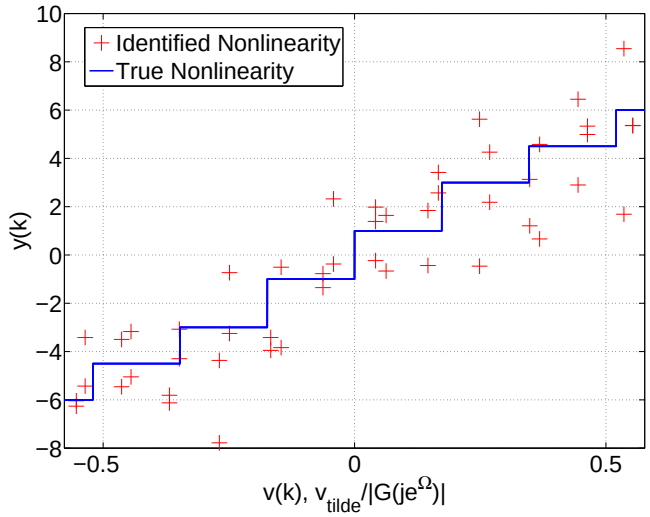


Figure 7.24: Identified nonlinearity versus true nonlinearity (7.42), where $m = 75$ and $A_0 = 5$ (Example 7.6.4).

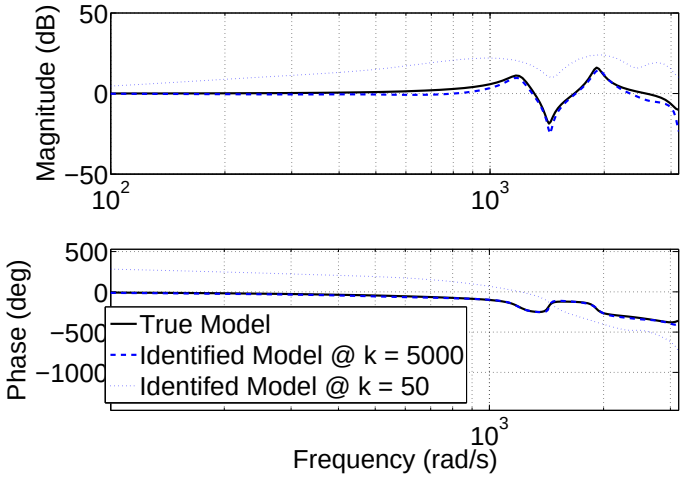


Figure 7.25: Frequency response comparison of the true G and the identified LTI system obtained using $\hat{\mathcal{W}}$ as an estimate of (7.42), where k is the number of data points used to determine the identified dynamic model. The RCO controller order is $n_c = 9$ with $p = 1$ and $\alpha = 1$ (Example 7.6.4).

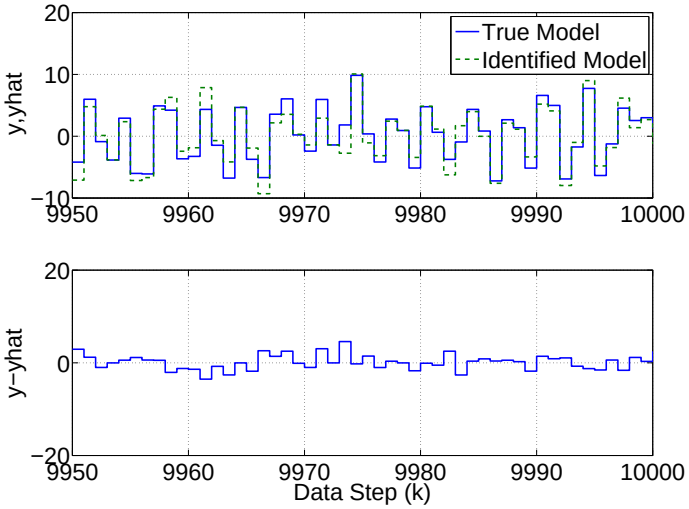


Figure 7.26: Performance comparison for Example 7.6.4 of the output of the Wiener system $y(k)$, and the output of the estimated system $\hat{y}(k)$. The bottom plot is the performance $z(k)$.

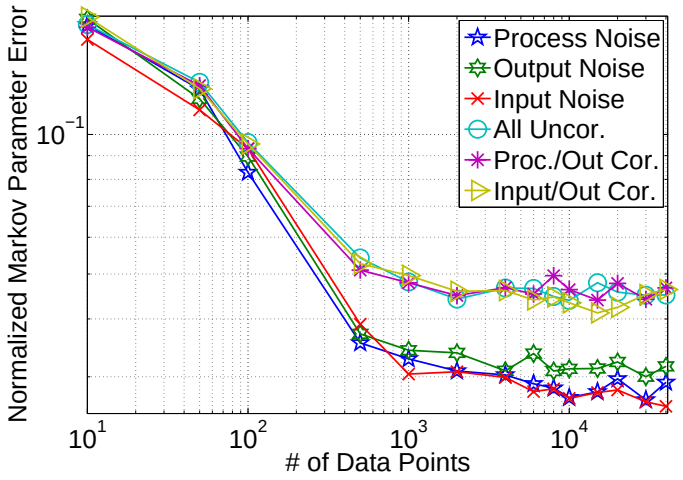


Figure 7.27: RMSE Markov parameter error versus number of data points. For each number of data points we perform a 100-run Monte Carlo simulation.

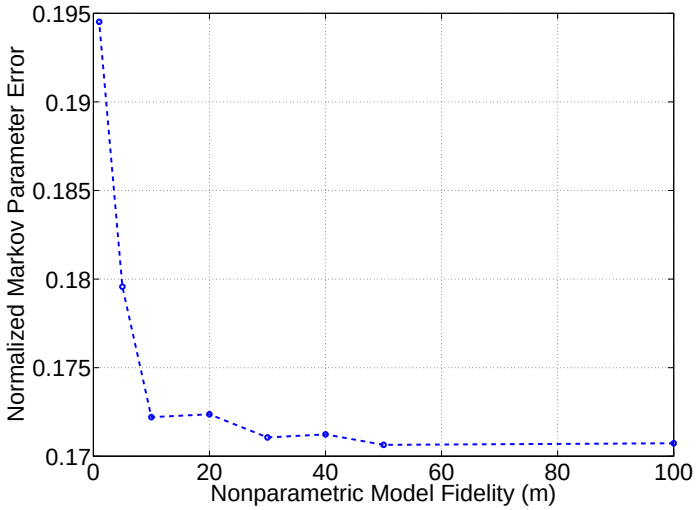


Figure 7.28: RMSE Markov parameter error for an increasing number of points in the nonparametric model. For each value of m, a 100-run Monte Carlo simulation is performed.

CHAPTER VIII

Adaptive State Estimation for Nonminimum-Phase Systems with Uncertain Harmonic Inputs

8.1 Introduction

The classical Kalman filter is the optimal state estimator for linear systems under white process and sensor noise with zero mean and finite second moments. Implementation of the optimal estimator under these idealized conditions depends on knowledge of the linear dynamics and noise covariances. When these assumptions are not satisfied, the accuracy of the Kalman filter can be degraded [85, 12, 15].

If the transfer function from the process noise to the measurements is minimum phase, the number of outputs equals the number of disturbances, and there is no sensor noise, then the minimum achievable estimation error is zero [104]. On the other hand, the presence of nonminimum-phase zeros increases the minimum achievable estimation error and thus, for harmonic disturbances, the Kalman filter does not give perfect state estimates [105, 106].

A more proactive approach is to implement an adaptive state estimator, where the goal is to identify the dynamics and noise statistics during system operation and use this information to tune the estimator on-line [14].

In addition to compensating for white process noise, the Kalman filter accommo-

dates the presence of a known, deterministic input. By injecting this signal into the estimator, the estimator experiences no loss of estimation accuracy relative to the case in which no deterministic input is present. This feature is essential when the Kalman filter is used in conjunction with the linear-quadratic regulator for constructing the full-order dynamic LQG controller.

In practice, however, the deterministic input may not be known exactly, and this error can viewed as a component of the process noise. However, this approach may be conservative and can lead to bias when the unknown input has a nonzero "mean" value. Consequently, a more direct approach is to extend the estimator to include an estimate of the unknown input [107, 108, 109, 110, 111]. Yet another approach is to constrain the gains of the estimator in order to guarantee that the state estimates are unbiased [112, 113, 114, 115].

In this chapter we consider state estimation for minimum- or nonminimum-phase systems in the presence of an unknown harmonic input. To address this problem we consider the estimator structure shown in Figure 8.1 with an auxiliary input $\hat{u}$, which is the output of an adaptive feedback system that is updated on-line. The signal $\hat{u}$ is estimated using a retrospective-cost-based input-reconstruction technique. In this way, the adaptive feedback system uses knowledge of the estimator residual to improve the accuracy of the state estimator by reconstructing the harmonic disturbance, thereby achieving perfect estimates in the minimum and nonminimum-phase cases. A related technique is used in [116].

The contents of this chapter are as follows. In Section 8.2 we describe the state estimation problem and construct a state estimator that uses an auxiliary input from an adaptive subsystem. In Section 8.3, we describe an input-reconstruction technique that constructs the auxiliary input by minimizing the residual error, that is, the difference between the measured output and the output of the estimator system.

Next, we numerically examine the adaptive state estimation error in comparison

to the optimal state estimator. In Section 8.5 we demonstrate the adaptive state estimator on linear numerical examples, and compare the results to the Kalman filter. In Section 8.6 we extend the method to nonlinear state estimation, and in Section 8.7 we demonstrate the method on nonlinear examples with comparisons to the extended and unscented Kalman filters.

8.2 Problem Formulation

Consider the linear-time-invariant system

$$x(k+1) = Ax(k) + Bu(k) + Bw(k), \tag{8.1}$$

$$y(k) = Cx(k), \tag{8.2}$$

where $x(k) \in \mathbb{R}^n$ is the unknown state, $u(k) \in \mathbb{R}^m$ is an unknown input, $w(k) \in \mathbb{R}^m$ is unknown zero-mean Guassian white noise, and $y(k) \in \mathbb{R}^p$ is the measured output, which is assumed to be bounded. The matrices $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, and $C \in \mathbb{R}^{p \times n}$ are known, and (A, C) is observable. Furthermore we assume that $u(k)$ is the output of a Lyapunov-stable, linear system.

In order to obtain an estimate $\hat{x}(k) \in \mathbb{R}^n$ of the state $x(k)$, we construct an adaptive state estimator (ASE) of the form

$$\hat{x}(k+1) = A\hat{x}(k) + B\hat{u}(k), \tag{8.3}$$

$$\hat{y}(k) = C\hat{x}(k), \tag{8.4}$$

$$z(k) = y(k) - \hat{y}(k), \tag{8.5}$$

where $\hat{y}(k) \in \mathbb{R}^p$ is the estimated output, $\hat{u}(k) \in \mathbb{R}^m$ is the estimator input, and $z(k) \in \mathbb{R}^p$ is the measured output error. Furthermore, $\hat{u}(k)$ is the output of the

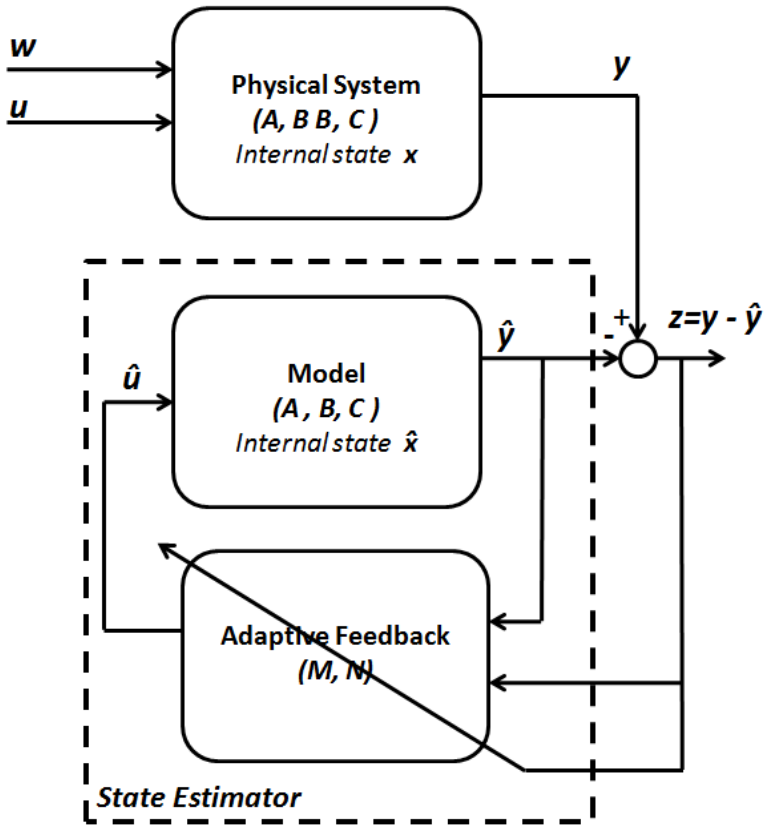


Figure 8.1: Adaptive State Estimator Architecture

strictly proper adaptive feedback system of order n_c, with input $z(k)$, given by

$$\hat{u}(k) = \sum_{i=1}^{n_c} M_i(k)\hat{u}(k-i) + \sum_{i=0}^{n_c} N_i(k)z(k-i), \tag{8.6}$$

where $M_i(k) \in \mathbb{R}^{m\times m}$, $i = 1, \ldots, n_c$, and $N_i(k) \in \mathbb{R}^{m\times p}$, $i = 0, \ldots, n_c$. The goal is to update $M_i(k)$ and $N_i(k)$ using the measured output error $z(k)$. Figure 8.1 shows the adaptive estimator structure.

8.3 State Estimation Using a Retrospective Surrogate Cost

For $i \geq 1$, define the Markov parameter H_i of (A, B, C) given by

$$H_i \triangleq CA^{i-1}B. \tag{8.7}$$

For example, $H_1 = CB$ and $H_2 = CAB$. Let r be a positive integer. Then, for all $k \geq r$,

$$\hat{x}(k) = A^r\hat{x}(k-r) + \sum_{i=1}^{r} A^{i-1}B\hat{u}(k-i), \tag{8.8}$$

and thus

$$z(k) = CA^r\hat{x}(k-r) - y(k) + \bar{H}\hat{\bar{U}}(k-1), \tag{8.9}$$

where

$$\bar{H} \triangleq \left[\begin{array}{ccc} H_1 & \cdots & H_r \end{array} \right] \in \mathbb{R}^{p \times rm}$$

and

$$\hat{\bar{U}}(k-1) \triangleq \left[\begin{array}{c} \hat{u}(k-1) \\ \vdots \\ \hat{u}(k-r) \end{array} \right].$$

Next, we rearrange the columns of $\bar{H}$ and the components of $\hat{\bar{U}}(k-1)$ and partition the resulting matrix and vector so that

$$\bar{H}\hat{\bar{U}}(k-1) = \mathcal{H}'\hat{U}'(k-1) + \mathcal{H}\hat{U}(k-1), \tag{8.10}$$

where $\mathcal{H}' \in \mathbb{R}^{p\times(rm-l_{\hat{U}})}$, $\mathcal{H} \in \mathbb{R}^{p\times l_{\hat{U}}}$, $\hat{U}'(k-1) \in \mathbb{R}^{rm-l_{\hat{U}}}$, and $\hat{U}(k-1) \in \mathbb{R}^{l_{\hat{U}}}$. Then, we can rewrite (8.9) as

$$z(k) = \mathcal{S}(k) + \mathcal{H}\hat{U}(k-1), \tag{8.11}$$

where

$$\mathcal{S}(k) \triangleq CA^r\hat{x}(k-r) - y(k) + \mathcal{H}'\hat{U}'(k-1). \tag{8.12}$$

For example, $\bar{H} = \left[\begin{array}{ccccc} H_1 & H_2 & H_3 & H_4 & H_5 \end{array}\right]$,

$$\mathcal{H}' = \left[\begin{array}{ccc} H_1 & H_2 & H_4 \end{array}\right], \quad \hat{U}'(k-1) = \left[\begin{array}{c} \hat{u}(k-1) \\ \hat{u}(k-2) \\ \hat{u}(k-4) \end{array}\right],$$

and

$$\mathcal{H} = \left[\begin{array}{cc} H_3 & H_5 \end{array}\right], \quad \hat{U}(k-1) = \left[\begin{array}{c} \hat{u}(k-3) \\ \hat{u}(k-5) \end{array}\right].$$

Note that the decomposition of $\bar{H}\hat{\bar{U}}(k-1)$ in (8.10) is not unique. Let s be a positive integer. Then for $i = 1, \ldots, s$, we replace $\mathcal{H}$, $\hat{U}(k-1)$, $\mathcal{H}'$, and $\hat{U}'(k-1)$ in (8.10) with $\mathcal{H}_j \in \mathbb{R}^{p\times l_{\hat{U}_j}}$, $\hat{U}_j(k-1) \in \mathbb{R}^{l_{\hat{U}_j}}$, $\mathcal{H}'_j \in \mathbb{R}^{p\times(rm-l_{\hat{U}_j})}$, and $\hat{U}'_j(k-1) \in \mathbb{R}^{rm-l_{\hat{U}_j}}$, respectively, such that (8.10) becomes

$$\bar{H}\hat{\bar{U}}(k-1) = \mathcal{H}'_j\hat{U}'_j(k-1) + \mathcal{H}_j\hat{U}_j(k-1). \tag{8.13}$$

Therefore, for $j = 1, \ldots, s$, we can rewrite (8.11) as

$$z(k) = \mathcal{S}_j(k) + \mathcal{H}_j \hat{U}_j(k-1), \tag{8.14}$$

where

$$\mathcal{S}_j(k) \triangleq CA^r \hat{x}(k-r) + y(k) + \mathcal{H}_j' \hat{U}_j'(k-1). \tag{8.15}$$

Next, let $0 \leq k_1 \leq k_2 \leq \cdots \leq k_s$. Replacing k by $k - k_j$ in (8.14) yields

$$z(k-k_j) = \mathcal{S}_j(k-k_j) + \mathcal{H}_j \hat{U}_j(k-k_j-1). \tag{8.16}$$

Now, by stacking $z(k-k_1), \ldots, z(k-k_s)$, we define the *extended performance*

$$Z(k) \triangleq \begin{bmatrix} z(k-k_1) \\ \vdots \\ z(k-k_s) \end{bmatrix} \in \mathbb{R}^{sp}. \tag{8.17}$$

Therefore,

$$Z(k) \triangleq \tilde{\mathcal{S}}(k) + \tilde{\mathcal{H}} \hat{\tilde{U}}(k-1), \tag{8.18}$$

where

$$\tilde{\mathcal{S}}(k) \triangleq \begin{bmatrix} \mathcal{S}_1(k-k_1) \\ \vdots \\ \mathcal{S}_s(k-k_s) \end{bmatrix} \in \mathbb{R}^{sp} \tag{8.19}$$

and $\hat{\tilde{U}}(k-1)$ has the form

$$\hat{\tilde{U}}(k-1) \triangleq \begin{bmatrix} \hat{u}(k-q_1) \\ \vdots \\ \hat{u}(k-q_g) \end{bmatrix} \in \mathbb{R}^{gm}, \tag{8.20}$$

where $k_1 \leq q_1 < q_2 < \cdots < q_g \leq k_s + r$. The vector $\hat{\tilde{U}}(k-1)$ is formed by stacking $\hat{U}_1(k-k_1-1), \ldots, \hat{U}_s(k-k_s-1)$ and removing copies of repeated components, and $\tilde{\mathcal{H}} \in \mathbb{R}^{sp \times gm}$ is constructed according to the structure of $\hat{\tilde{U}}(k-1)$. For example, with $s=2$, $k_1=0$, and $k_2=2$, stacking $\hat{U}_1(k-1) = \begin{bmatrix} \hat{u}(k-1) \\ \hat{u}(k-2) \end{bmatrix}$ and $\hat{U}_2(k-3) = \hat{u}(k-3)$ results in

$$\hat{\tilde{U}}(k-1) = \begin{bmatrix} \hat{u}(k-1) \\ \hat{u}(k-2) \\ \hat{u}(k-3) \end{bmatrix}, \quad \tilde{\mathcal{H}} = \begin{bmatrix} H_1 & H_2 & H_3 \\ 0 & H_1 & H_2 \\ 0 & 0 & H_3 \end{bmatrix}. \tag{8.21}$$

Note that $\tilde{\mathcal{H}}$ consists of the entries of $\mathcal{H}_1, \ldots, \mathcal{H}_s$ arranged according to the structure of $\tilde{U}(k-1)$.

Next, we define the *retrospective performance*

$$\hat{z}(k-k_j) \triangleq \mathcal{S}_j(k-k_j) + \mathcal{H}_j U_j^*(k-k_j-1), \tag{8.22}$$

where the past input estimates $\hat{U}_j(k-k_j-1)$ in (8.16) are replaced by the retrospectively optimized input estimates $U_j^*(k-k_j-1)$, which are determined below. In

analogy with (8.17), the *extended retrospective performance* is defined as

$$\hat{Z}(k) \triangleq \begin{bmatrix} \hat{z}(k-k_1) \\ \vdots \\ \hat{z}(k-k_s) \end{bmatrix} \in \mathbb{R}^{sp} \tag{8.23}$$

and thus is given by

$$\hat{Z}(k) = \tilde{\mathcal{S}}(k) + \tilde{\mathcal{H}}\tilde{U}^*(k-1), \tag{8.24}$$

where the components of $\tilde{U}^*(k-1) \in \mathbb{R}^{l_{\hat{U}}}$ are the components of $U_1^*(k-k_1-1), \ldots, U_s^*(k-k_s-1)$ ordered in the same way as the components of $\hat{\tilde{U}}(k-1)$. Subtracting (8.18) from (8.24) yields

$$\hat{Z}(k) = Z(k) - \tilde{\mathcal{H}}\hat{\tilde{U}}(k-1) + \tilde{\mathcal{H}}\tilde{U}^*(k-1). \tag{8.25}$$

Finally, we define the *retrospective cost function*

$$J(\tilde{U}^*(k-1), k) \triangleq \hat{Z}^{\mathrm{T}}(k)R_1(k)\hat{Z}(k) + \eta(k)\tilde{U}^{*\mathrm{T}}(k-1)R_2(k)\tilde{U}^*(k-1), \tag{8.26}$$

where $R_1(k) \in \mathbb{R}^{ps\times ps}$ is a positive-definite performance weighting, $R_2(k) \in \mathbb{R}^{gm\times gm}$ is a positive-definite input estimate weighting, and $\eta(k) \geq 0$ is a regularization weighting. The goal is to determine retrospective input estimates $\tilde{U}^*(k-1)$ that would have provided better performance than the estimated inputs $\hat{U}(k-1)$ that were applied to the system. The retrospectively optimized estimated input values $\tilde{U}^*(k-1)$ are then used to update the controller. Substituting (8.25) into (8.26) yields

$$J(\tilde{U}^*(k-1), k) = \tilde{U}^{*\mathrm{T}}(k-1)\mathcal{A}(k)\tilde{U}^*(k-1) + \tilde{U}^{*\mathrm{T}}(k-1)\mathcal{B}^{\mathrm{T}}(k) + \mathcal{C}(k), \tag{8.27}$$

where

$$\mathcal{A}(k) \triangleq \tilde{\mathcal{H}}^{\mathrm{T}} R_1(k) \tilde{\mathcal{H}} + \eta(k) R_2(k), \tag{8.28}$$

$$\mathcal{B}(k) \triangleq 2\tilde{\mathcal{H}}^{\mathrm{T}} R_1(k)[Z(k) - \tilde{\mathcal{H}}\hat{\tilde{U}}(k-1)], \tag{8.29}$$

$$\mathcal{C}(k) \triangleq Z^{\mathrm{T}}(k) R_1(k) Z(k) - 2Z^{\mathrm{T}}(k) R_1(k) \tilde{\mathcal{H}} \hat{\tilde{U}}(k-1) + \hat{\tilde{U}}^{\mathrm{T}}(k-1) \tilde{\mathcal{H}}^{\mathrm{T}} R_1(k) \tilde{\mathcal{H}} \hat{\tilde{U}}(k-1). \tag{8.30}$$

If either $\tilde{\mathcal{H}}$ has full column rank or $\eta(k) > 0$, then $\mathcal{A}(k)$ is positive definite. In this case, $J(\tilde{U}^*(k-1), k)$ has the unique global minimizer

$$\tilde{U}^*(k-1) = -\frac{1}{2} \mathcal{A}^{-1}(k) \mathcal{B}(k), \tag{8.31}$$

which is the retrospectively optimized estimated inputs.

The regularization weighting $\eta(k)$ can be used to bound the retrospectively optimized estimated inputs $\tilde{U}^*(k-1)$ and thus indirectly bound the estimated inputs $\hat{\tilde{U}}(k)$. For example, $\eta(k)$ may be performance based

$$\eta(k) = \eta_0(k) ||Z(k)||_2^2 \tag{8.32}$$

or error based

$$\eta(k) = \eta_0(k) ||\tilde{U}^*(k-2) - \hat{\tilde{U}}(k-2)||_2^2, \tag{8.33}$$

where $\eta_0(k) \geq 0$. Alternatively, the retrospectively optimized inputs can be bounded directly by using a saturation function, where $\eta(k) \equiv 0$ in (8.28) and (8.31) is replaced by

$$\tilde{U}^*(k-1) \triangleq \mathrm{sat}_{[a,b]}[-\frac{1}{2} \mathcal{A}^{-1}(k) \mathcal{B}(k)], \tag{8.34}$$

where $\text{sat}_{[a,b]}(\zeta)$ is the component-wise saturation function defined for scalar arguments by

$$\text{sat}_{[a,b]}(\zeta) \triangleq \begin{cases} b, & \text{if } \zeta \geq b, \\ \zeta, & \text{if } a < \zeta < b, \\ a & \text{if } \zeta \leq a, \end{cases} \tag{8.35}$$

where $a < b$ are the component-wise saturation levels.

8.4 Adaptive Feedback Update

The reconstructed input (8.6) can be expressed as

$$\hat{u}(k) = \theta(k)\phi(k-1), \tag{8.36}$$

where

$$\theta(k) \triangleq [M_1(k) \ \cdots \ M_{n_c}(k) \ N_1(k) \ \cdots \ N_{n_c}(k)] \in \mathbb{R}^{m \times n_c(m+p)} \tag{8.37}$$

and

$$\phi(k-1) \triangleq \begin{bmatrix} \hat{u}(k-1) \\ \vdots \\ \hat{u}(k-n_c) \\ y(k-1) \\ \vdots \\ y(k-n_c) \end{bmatrix} \in \mathbb{R}^{n_c(m+p)}. \tag{8.38}$$

Next, we define the recursive least squares (RLS) cost function

$$J_{\mathrm{R}}(\theta(k)) \triangleq \sum_{i=q_g+1}^{k} \lambda^{k-i} \|\phi^{\mathrm{T}}(i-q_g-1)\theta^{\mathrm{T}}(k) - u^{*\mathrm{T}}(i-q_g)\|^2 + \lambda^k(\theta(k)-\theta(0))P^{-1}(0)(\theta(k)-\theta(0))^{\mathrm{T}}, \tag{8.39}$$

where $\|\cdot\|$ is the Euclidean norm and, for some $\varepsilon \in (0,1)$, $\lambda(k) \in (\varepsilon, 1]$ is the forgetting factor, and $P(0) \in \mathbb{R}^{n_{\mathrm{c}}(m+p)\times n_{\mathrm{c}}(m+p)}$ is positive definite. Minimizing (8.39) yields

$$\begin{aligned}\theta^{\mathrm{T}}(k) \triangleq{}& \theta^{\mathrm{T}}(k-1) + \beta(k)P(k-1)\phi(k-q_g-1) \\ &\cdot [\phi^{\mathrm{T}}(k-q_g-1)P(k-1)\phi(k-q_g-1)+\lambda(k)]^{-1} \\ &\cdot [\theta(k-1)\phi(k-q_g-1) - u^*(k-q_g)]^{\mathrm{T}},\end{aligned} \tag{8.40}$$

where $\beta(k)$ is either 0 or 1. When $\beta(k) = 1$, the controller is allowed to adapt, whereas, when $\beta(k) = 0$, the adaptation is off. $P(k)$ is updated by

$$\begin{aligned}P(k) \triangleq{}& (1-\beta(k))P(k-1) + \beta(k)\lambda^{-1}(k)P(k-1) - \beta(k)\lambda^{-1}(k)P(k-1)\phi(k-q_g-1) \\ &\cdot [\phi^{\mathrm{T}}(k-q_g-1)P(k-1)\phi(k-q_g-1)+\lambda(k)]^{-1}\phi^{\mathrm{T}}(k-q_g-1)P(k-1).\end{aligned} \tag{8.41}$$

We initialize $P(0) = \gamma I$, where $\gamma > 0$. Furthermore, the updates (8.40) and (8.41) are based on the g^{th} component of $\tilde{U}^*(k-1)$. However any or all of the components of $\tilde{U}^*(k-1)$ may be used in the update of $\theta(k)$ and $P(k)$.

8.5 Linear Examples

In this section, we apply the adaptive state estimator to several linear examples and compare its performance with the Kalman filter (KF). Define the error metric

$$\varepsilon_k = \frac{1}{\ell} \sum_{i=k}^{k+l-1} ||e_i||, \tag{8.42}$$

where ℓ is the window size. For all examples in this section, $\ell = 2000$.

8.5.1 Example 1: Dual Spring-Mass-Damper System, Minimum-Phase

Consider the dual spring-mass-damper system shown in Figure 8.2. For $i = 1, 2$, let q_i be the position of i^{th} mass, let m_i be the mass of the i^{th} block, let k_i be the stiffness of the i^{th} spring, and let c_i be the damping coefficient of the i^{th} damper. Finally, let u be the force applied to the first block.

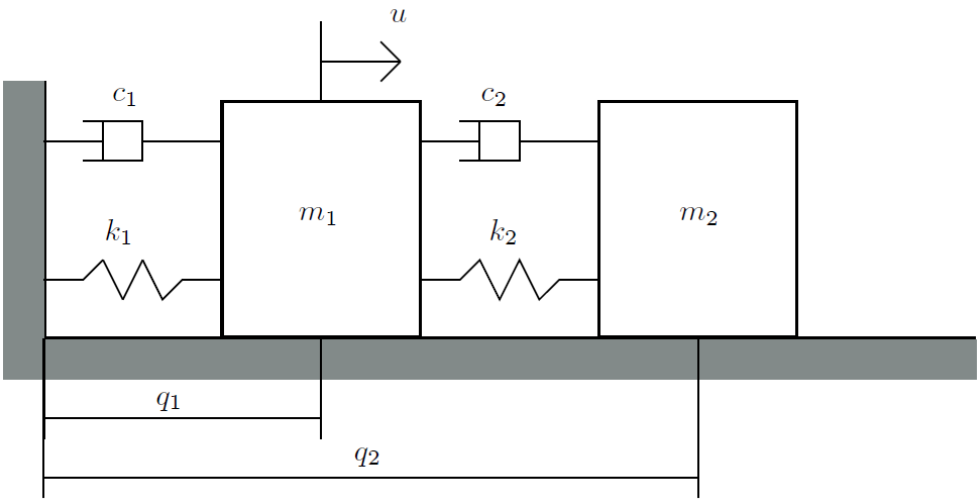


Figure 8.2: Dual spring-mass-damper system

The equations of motion of this system are

$$\dot{x} = A_c x + B_c u,$$

where

$$x = \begin{bmatrix} q_1 \\ \dot{q}_1 \\ q_2 \\ \dot{q}_2 \end{bmatrix}, A_c = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_1+k_2}{m_1} & -\frac{c_1+c_2}{m_1} & \frac{k_2}{m_1} & \frac{c_2}{m_1} \\ 0 & 0 & 0 & 1 \\ \frac{k_2}{m_2} & \frac{c_2}{m_2} & -\frac{k_2}{m_2} & -\frac{c_2}{m_2} \end{bmatrix}, \; B_c = \begin{bmatrix} 0 \\ \frac{1}{m_1} \\ 0 \\ 0 \end{bmatrix}. \tag{8.43}$$

We choose $m_1 = 5$, $m_2 = 4$, $k_1 = k_2 = 0.01$, $c_1 = 0.5$, and $c_2 = 0.05$. We discretize the system using

$$A = e^{A_c T_s} \; , \; B = A_c^{-1}(A - I)B_c, \tag{8.44}$$

where $T_s = 1$ is the sampling time. The output matrix is

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}, \tag{8.45}$$

which represents the position of the first mass. The zeros of the discretized system are -0.9680 and $0.9852 \pm 0.0687\jmath$. For the ASE, let $\eta(k) = 0$, $n_c = 3$, $P(0) = 1 \times 10^{15} I_{6\times 6}$, and $\tilde{\mathcal{H}} = CB$. For the Kalman filter, the noise covariance matrix is $Q = BB^{\mathrm{T}}$, and the initial error covariance is $I_{4\times 4}$. Finally, $u(k) = 20\sin(k)$. Figure 8.3 shows that ε_k converges to zero for both the KF and ASE.

8.5.2 Example 2: Dual Spring-Mass-Damper system, Nonminimum-Phase

For this example, we consider the system defined by (8.44) with the output matrix

$$C = \begin{bmatrix} -10 & 0 & 1 & 0 \end{bmatrix}, \tag{8.46}$$

which represents a difference in the positions of the two masses. The zeros of the discretized system are -0.9632, 1.1293, and 0.8767, and thus (A, B, C) is nonminimum phase. For the ASE, let $\eta(k) = 5 \times 10^{-5}$, $n_c = 5$, $P(0) = 10I_{10\times 10}$, and $\tilde{\mathcal{H}} = CAB$. For the Kalman filter, the noise covariance matrix is $Q = BB^{\mathrm{T}}$, and the initial error covariance is $I_{4\times 4}$. Finally, $u(k) = 20\sin(k)$. Figure 8.4 shows the performance ε_k of the ASE and KF.

8.5.3 Example 3: Nonminimum-Phase Linearized Planar Linkage

We consider the planar linkage system shown in Figure 8.5. Let p_1 be the point where the first link is attached to the horizontal plane, and let p_2 be the point where the two links are connected. Furthermore, for $i = 1, 2$, let q_i be the center of mass of the i^{th} link, let m_i be the mass of the i^{th} link, let c_i be the damping at the joint p_i, and let k_i be the stiffness of the joint p_i.

Next, let $\mathrm{F_A}$ be an inertial frame with the orthogonal unit vectors $(\hat{\imath}_{\mathrm{A}}, \hat{\jmath}_{\mathrm{A}}, \hat{k}_{\mathrm{A}})$, where $\hat{\imath}_{\mathrm{A}}$ and $\hat{\jmath}_{\mathrm{A}}$ lie in the plane of motion of the planar linkage system. For simplicity, we assume that the origin of $\mathrm{F_A}$ is located at p_1. In addition, for $i = 1, 2$, let $\mathrm{F_{B_i}}$ be a body-fixed frame attached to the i^{th} link. More specifically, $\mathrm{F_{B_i}}$ is a body-fixed frame that rotates as the i^{th} link rotates. For $i = 1, 2$, let $\mathrm{F_{B_i}}$ have orthogonal unit vectors $(\hat{\imath}_{\mathrm{B}_i}, \hat{\jmath}_{\mathrm{B}_i}, \hat{k}_{\mathrm{B}_i})$, where $\hat{\imath}_{\mathrm{B}_i}$ is in the direction from p_i to q_i, and $\hat{\jmath}_{\mathrm{B}_i}$ is orthogonal to $\hat{\imath}_{\mathrm{B}_i}$ in the plane of motion. Furthermore, u_1 is an external torque applied at p_1. Finally, for $i = 1, 2$, let θ_i be the angle from $\hat{\imath}_{\mathrm{A}}$ to $\hat{\imath}_{\mathrm{B}_i}$. All frames are right handed.

The equations of motion of the planar linkage system [117] are given by

$$\begin{aligned} u_1 &= (\frac{1}{3}m_1 l_1^2 + m_2 l_1^2)\ddot{\theta}_1 + \frac{1}{2}m_2 l_1 l_2 \sin(\theta_1 - \theta_2)\dot{\theta}_2^2 + \frac{1}{2}m_2 l_1 l_2 \cos(\theta_1 - \theta_2)\ddot{\theta}_2 \\ &\quad + (k_1 + k_2)\theta_1 - k_2\theta_2 + (c_1 + c_2)\dot{\theta}_1 - c_2\dot{\theta}_2, \\ 0 &= (\frac{1}{3}m_2 l_2^2)\ddot{\theta}_2 - \frac{1}{2}m_2 l_1 l_2 \sin(\theta_1 - \theta_2)\dot{\theta}_1^2 + \frac{1}{2}m_2 l_1 l_2 \cos(\theta_1 - \theta_2)\ddot{\theta}_1 \\ &\quad - k_2\theta_1 + k_2\theta_2 - c_2\dot{\theta}_1 + c_2\dot{\theta}_2, \end{aligned} \tag{8.47}$$

where $m_1 = 2$, $m_2 = 1$, $l_1 = 3$, $l_2 = 2$, $k_1 = 7$, $k_2 = 5$, $c_1 = 10$, and $c_2 = 1$. The output of the system

$$y = \theta_2, \tag{8.48}$$

is the angle θ_2, which represents the angle from $\hat{\imath}_{\mathrm{A}}$ to $\hat{\imath}_{\mathrm{B}_2}$. Linearizing and discretizing (8.47) with sampling time $T_s = 1$ yields

$$x(k+1) = Ax(k) + Bu(k),$$

where

$$x(k) = \begin{bmatrix} \theta_1(k) \\ \theta_2(k) \\ \dot{\theta}_1(k) \\ \dot{\theta}_2(k) \end{bmatrix}, A = \begin{bmatrix} 0.661 & 0.174 & 0.5517 & 0.1408 \\ 1.0993 & -0.0223 & 0.928 & 0.26 \\ -0.134 & -0.065 & 0.39 & 0.161 \\ 0.009 & -0.3 & 0.744 & -0.083 \end{bmatrix}, B = \begin{bmatrix} 0.024 \\ -0.011 \\ 0.028 \\ 0.042 \end{bmatrix}^T. \tag{8.49}$$

The zeros of the discretized system are 6.6598, 0.3219, and -0.2619, and thus (A, B, C) is nonminimum phase. For the ASE, let $\eta(k) = 5 \times 10^{-4}$, $n_c = 10$, $P(0) = 1 \times 10^6 I_{20\times 20}$, and $\tilde{\mathcal{H}} = CB$. For the Kalman filter, the noise covariance matrix is chosen to be $Q = BB^{\mathrm{T}}$, and the initial error covariance is $I_{4\times 4}$. Finally,

$u(k) = 20\sin(k)$. Figure 8.6 shows the performance ε_k of the ASE and KF.

Next, consider the linearized planar linkage system in Example 3 with the input $u(k) = 20\sin(k) + 5\sin(0.3k) + 80\sin(0.01k)$. For the ASE, $\eta(k) = 1 \times 10^{-5}$, $n_c = 7$, $P(0) = 1 \times 10^4 I_{14\times 14}$, and $\tilde{\mathcal{H}} = CAB$. For the Kalman filter, the noise covariance matrix is chosen to be $Q = BB^{\mathrm{T}}$, and the initial error covariance is $I_{4\times 4}$. For this example, Figure 8.7 shows the performance ε_k of the ASE and KF.

Next, consider the linearized planar linkage system in Example 3 with the input $u(k) = 10$. For the ASE, $\eta(k) = 2\times 10^{-3}$, $n_c = 1$, $P(0) = 1\times 10^3 I_{1\times 1}$, and $\tilde{\mathcal{H}} = CAB$. For the Kalman filter, the noise covariance matrix is chosen to be $Q = BB^{\mathrm{T}}$, and the initial error covariance is $I_{4\times 4}$. For this example, Figure 8.8 shows the performance ε_k of the ASE and KF.

8.5.4 Example 5: Linearized Planar Linkage with Process Noise

Consider the system

$$x(k+1) = Ax(k) + Bu(k) + \alpha Bw(k),$$

where $x(k)$, A, and B are given by (8.49). The output of this system is given by (8.48). We test the ASE and KF for $\alpha = 0$, $\alpha = 10^{-6}$, $\alpha = 10^{-4}$, and $\alpha = 10^{-2}$. For the ASE, $\eta(k) = 1 \times 10^{-5}$, $n_c = 5$, $P(0) = 1 \times 10^3 I_{10\times 10}$, and $\tilde{\mathcal{H}} = CA^2B$. For the KF, the noise covariance matrix is $Q = \alpha BB^{\mathrm{T}}$, and the initial error covariance is $I_{4\times 4}$. Finally, $u(k) = 20\sin(k)$. Figure 8.9 shows the performance ε_k of the ASE and KF. For the KF, the effect of α is negligible and thus we show ε_k of the KF for only $\alpha = 0$.

8.6 Nonlinear State Estimation

Consider the MIMO nonlinear time-invariant system,

$$x(k+1) = f(x(k)) + g(u(k), w(k)), \tag{8.50}$$

$$y(k) = h(x(k)), \tag{8.51}$$

where $x(k) \in \mathbb{R}^n$ is the state, $u(k) \in \mathbb{R}^p$ is an unknown input, $w(k) \in \mathbb{R}^m$ is unknown zero-mean Gaussian white noise, $y(k) \in \mathbb{R}^p$ is the measured output, which is assumed to be bounded, $f : \mathbb{R}^n \to \mathbb{R}^n$, $g : \mathbb{R}^m \to \mathbb{R}^n$, and $h : \mathbb{R}^n \to \mathbb{R}^p$ are known functions. Furthermore, we assume that $u(k)$ is the output of a Lyapunov-stable linear system.

In order to obtain estimates $\hat{x}(k) \in \mathbb{R}^n$ of the state $x(k)$, we construct a state estimator of the form

$$\hat{x}(k+1) = f(\hat{x}(k)) + g(\hat{u}(k)), \tag{8.52}$$

$$\hat{y}(k) = h(\hat{x}(k)), \tag{8.53}$$

where the estimated output is $\hat{y}(k) \in \mathbb{R}^p$, and $\hat{u}(k) \in \mathbb{R}^m$ is the estimator input given by (8.6). M_k and N_k are updated as in the linear case, where H_i is redefined as

$$H_i \triangleq H F^{i-1} G, \tag{8.54}$$

where x_{eq} is an equilibrium point and

$$F \triangleq \left.\frac{\partial f}{\partial \hat{x}}\right|_{x_{\text{eq}}}, \quad H \triangleq \left.\frac{\partial h}{\partial \hat{x}}\right|_{x_{\text{eq}}}, \quad G \triangleq \left.\frac{\partial g}{\partial \hat{x}}\right|_{x_{\text{eq}}}. \tag{8.55}$$

Unlike the extended Kalman filter, the adaptive state estimator does not require a linearization of f, g, h, at each step k, which is used by the extended Kalman filter to propagate the error covariance.

8.7 Nonlinear Examples

In this section, we compare the adaptive state estimator (ASE) with the extended Kalman filter (EKF) and the unscented Kalman filter (UKF). We consider the nonlinear planar linkage system and the Van der Pol oscillator. For all examples, the error metric is given by (8.42) with $\ell = 2000$.

8.7.1 Example 5: Nonlinear Planar Linkage

In this example, we consider the nonlinear planar linkage given by discretizing (8.47) with $T_s = 1$, and with the output matrix (8.48). For the ASE, let $\eta(k) = 5 \times 10^{-4}$, $n_c = 10$, $P(0) = 10I_{20\times 20}$, and $\tilde{\mathcal{H}} = HG$, where H and G are obtained by linearizing and discretizing the system about the origin. For the EKF and UKF, we set $Q = 100B_kB_k^{\mathrm{T}}$, where B_k is the input vector obtained by linearizing and discretizing the system about the current state estimate $\hat{x}(k)$. The initial error covariance matrix for the EKF and UKF is $I_{4\times 4}$. Furthermore, for the UKF, we use nine sigma points, and we set $\kappa = 0$, $\beta = 2$, and $\alpha = 0.1$. Finally, $u(k) = 20\sin(k)$. Figure 8.10 shows the state estimates of the ASE and UKF for the last fifty steps. Figure 8.11 shows the performance ε_k of the ASE, UKF, and EKF.

8.7.2 Example 6: Van der Pol Oscillator

We consider the Van der Pol oscillator

$$\ddot{q} - \mu(1 - q^2)\dot{q} + q = u,$$

where $\mu = 1$. The output of this system is $y = q$. We discretize this system with $T_s = 0.1$ to obtain

$$\begin{aligned} x_1(k+1) &= T_s x_2(k) + x_1(k), \\ x_2(k+1) &= T_s \mu (1 - x_1(k)^2) x_2(k) - T_s x_1(k) + T_s u(k) + x_2(k), \end{aligned}$$

where $x_1 = q$ and $x_2 = \dot{q}$. For the ASE, let $\eta(k) = 5 \times 10^{-4}$, $n_c = 2$, $P(0) = 100 I_{4\times 4}$, and $\tilde{\mathcal{H}} = HG$, where H and G are obtained by linearizing the discretized system about the origin. For the EKF and UKF, the noise covariance matrix is $Q = [0\ 10]^{\mathrm{T}}[0\ 10]$, and the initial error covariance matrix is $I_{2\times 2}$. Furthermore, for the UKF, we use five sigma points and we set $\kappa = 0$, $\alpha = 0.01$, and $\beta = 1$. Finally, $u(k) = 10\sin(k)$. Figure 8.12 shows the phase portrait for this example. Figure 8.13 shows the state estimates for the ASE and UKF. Figure 8.14 shows the performance ε_k of the ASE, EKF, and UKF.

Next, consider the Van der Pol oscillator in Example 6 with the alternative output

$$y = \dot{q}. \tag{8.56}$$

For the ASE, let $\eta(k) = 1 \times 10^{-4}$, $n_c = 3$, $P(0) = 100 I_{6\times 6}$, and $\tilde{\mathcal{H}} = HFG$, where H, F and G are obtained by linearizing the discretized system about the origin. For the EKF and UKF, the noise covariance matrix is $Q = [0\ 10]^{\mathrm{T}}[0\ 10]$, and the initial error covariance matrix is $I_{2\times 2}$. Furthermore, for the UKF, we use five sigma points and we set $\kappa = 0$, $\alpha = 0.01$, and $\beta = 1$. Finally, $u(k) = 10\sin(k)$. Figure 8.15 shows the state estimates for the ASE and UKF. Figure 8.14 shows the performance ε_k of the ASE, EKF, and UKF.

8.7.3 Example 7: Van der Pol Oscillator with Process Noise

Consider the discretized nonlinear Van der Pol oscillator

$$x_1(k+1) = T_s x_2(k) + x_1(k),$$
$$x_2(k+1) = T_s\mu(1 - x_1(k)^2)x_2(k) - T_s x_1(k) + x_2(k) + T_s u(k) + \gamma T_s w(k),$$

where $\mu = 1$ and $T_s = 0.1$. The output of this system is $y = x_1$. Furthermore, $u(k) = 20\sin(k)$ and $w(k)$ is the realization of a zero mean Gaussian white noise process with unit variance. We test the ASE for $\gamma = 0$, $\gamma = 10^{-2}$, and $\gamma = 1$. We let $\eta(k) = 5 \times 10^{-4}$, $n_c = 2$, $P(0) = 100I_{4\times 4}$, $\tilde{\mathcal{H}} = HG$, where H and G are obtained by linearizing the discretized system about the origin. For the UKF, we choose the same parameters as in Example 6. Figure 8.17 shows the performance ε_k of the ASE for several values of γ. For the UKF, the effect of γ is negligible and thus we show ε_k of the KF only for $\gamma = 0$.

8.8 Conclusions

In this chapter we demonstrated a method for obtaining state estimates for minimum- and nonminimum-phase systems in the presence of harmonic process noise. First we constructed an estimator based on the known system model. At each step k we reconstruct the signal $u(k)$, called $\hat{u}^*(k)$, which minimizes the residual error $y(k) - \hat{y}(k)$. We then estimate a feedback system with input $z(k)$ and output $\hat{u}(k)$. Using the signal $\hat{u}(k)$ as the input to the estimator, we obtain estimates $\hat{x}$ of the system state $x(k)$.

We demonstrated the method on several linear examples including minimum and nonminimum-phase systems. In the minimum-phase case, the adaptive input reconstruction filter and the Kalman filter asymptotically reach zero state-estimation-error. In the nonminimum-phase case, the Kalman filter reaches a finite lower bound, on

the state-estimation-error. The adaptive input reconstruction filter outperforms the Kalman filter in this case.

Finally, we extended the method to nonlinear state estimation. We compare the adaptive input reconstruction filter to the extended Kalman filter and the unscented Kalman filter. We note that the adaptive input reconstruction filter does not require knowledge of the process noise covariance or linearizations of the model about each state estimate.

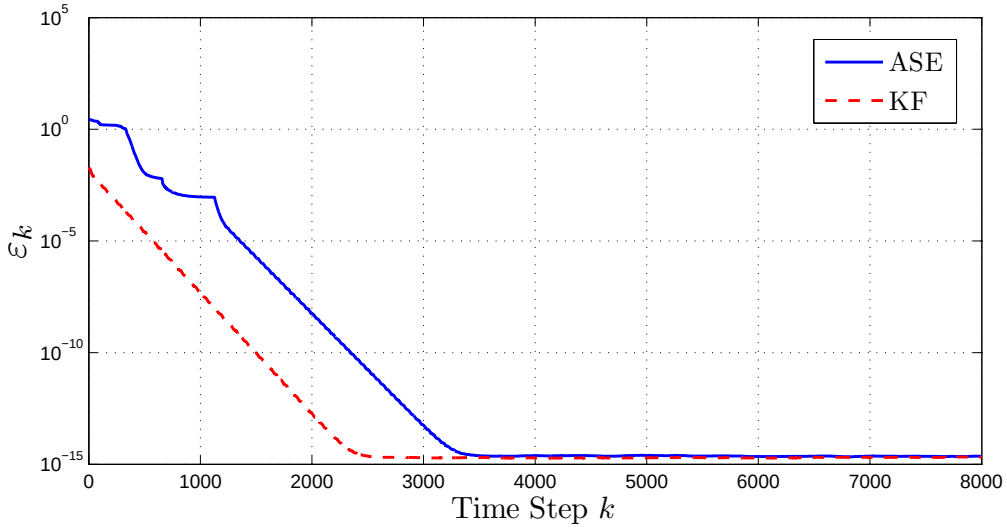


Figure 8.3: Comparison of the performance ε_k of the ASE and KF for the minimum-phase spring-mass-damper system.

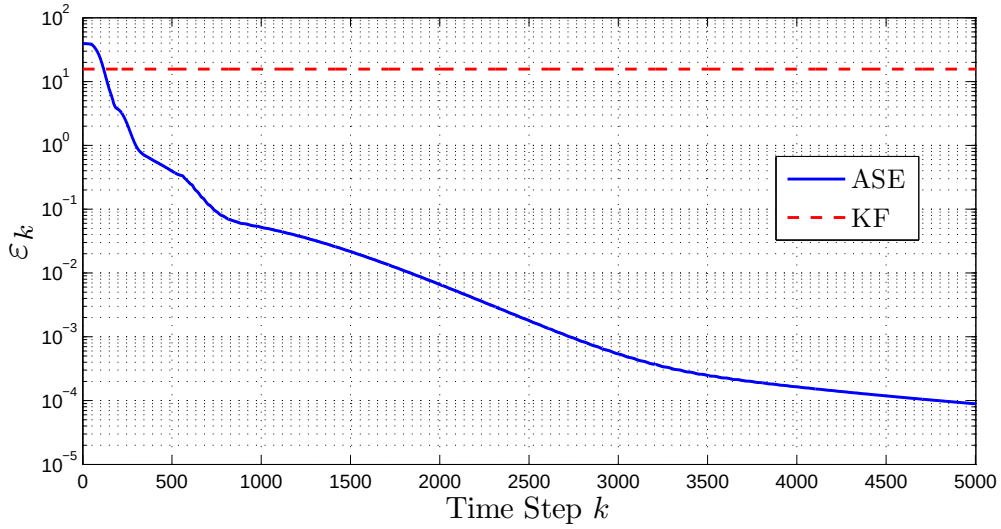


Figure 8.4: Comparison of the performance ε_k of the ASE and KF for the nonminimum-phase spring-mass-damper system.

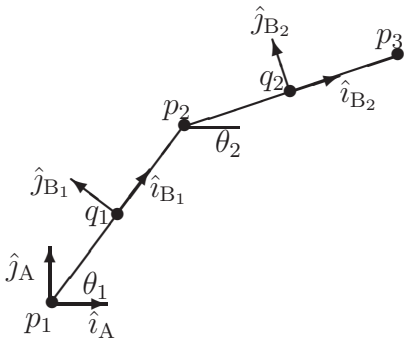


Figure 8.5: Planar linkage system. All motion is in the horizontal plane.

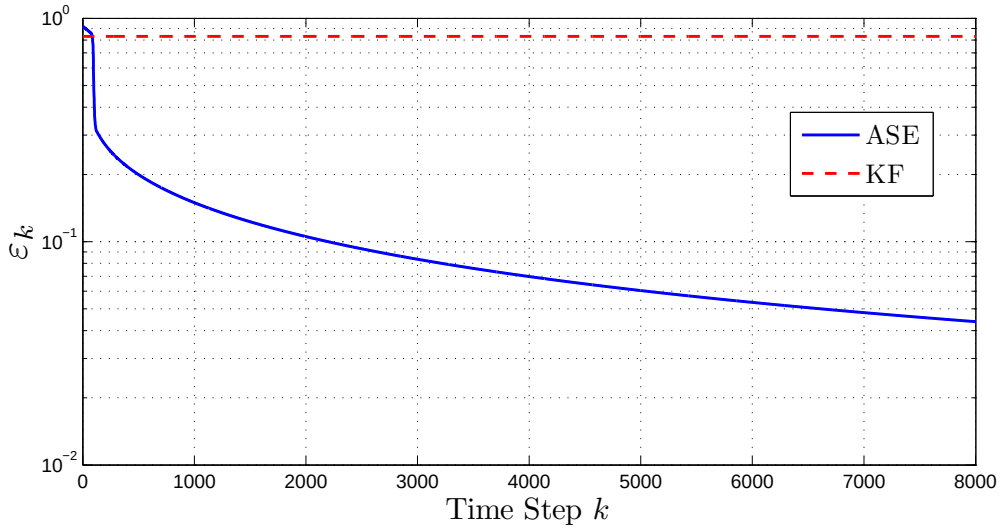


Figure 8.6: Comparison of the performance ε_k of the ASE and KF for the nonminimum-phase linearized planar linkage.

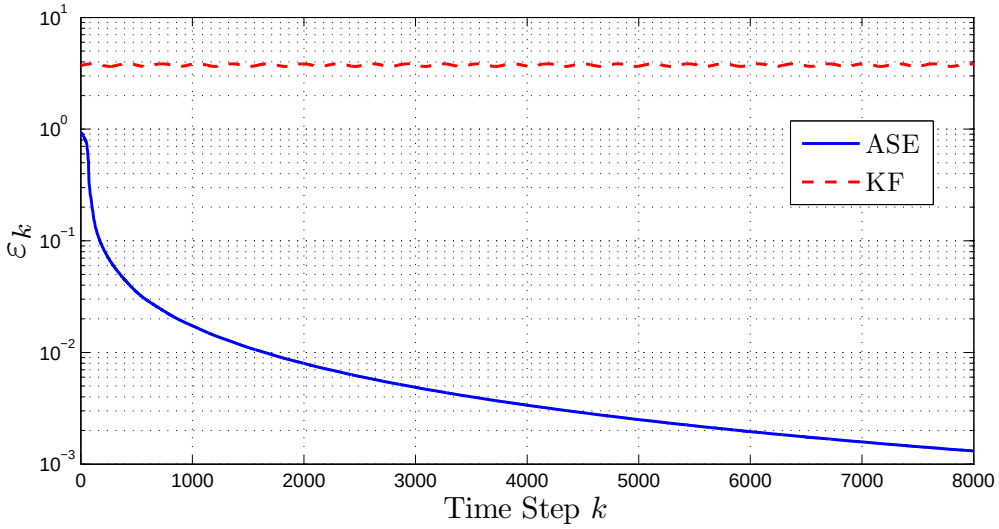


Figure 8.7: Comparison of the performance ε_k of the ASE and KF for the nonminimum-phase linearized planar linkage with a multi-harmonic input.

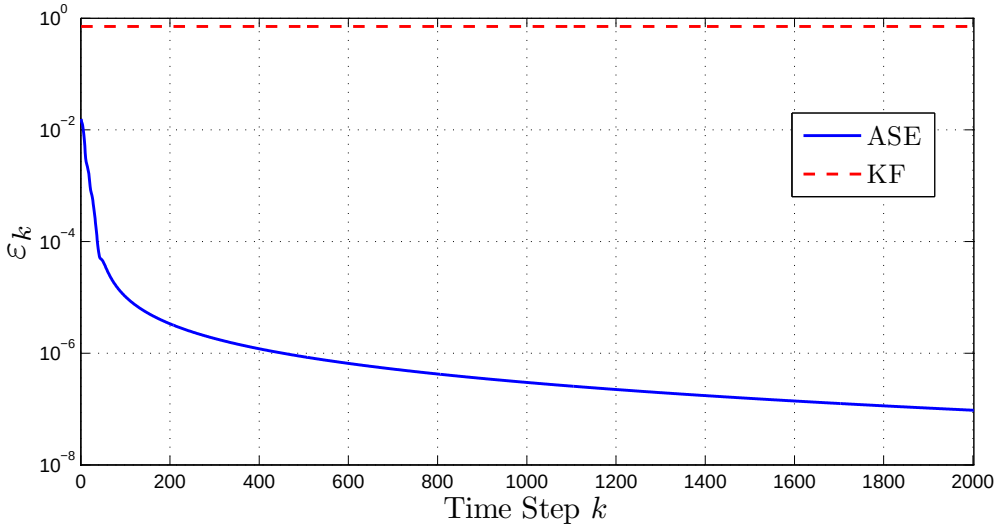


Figure 8.8: Comparison of the performance ε_k of the ASE and KF for the nonminimum-phase linearized planar linkage with a constant input.

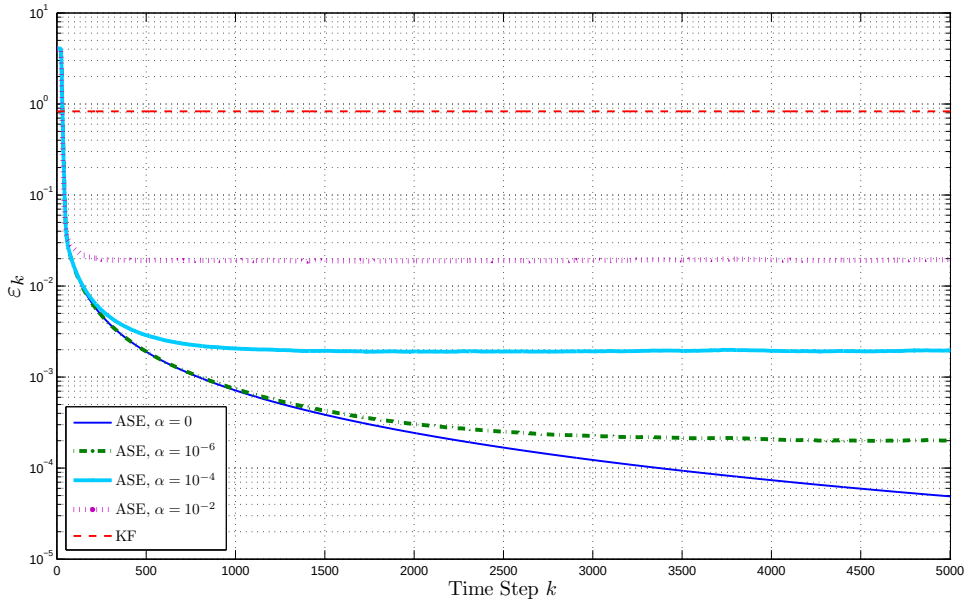


Figure 8.9: Comparison of the performance ε_k of the ASE and KF for the nonminimum-phase linearized planar linkage with process noise.

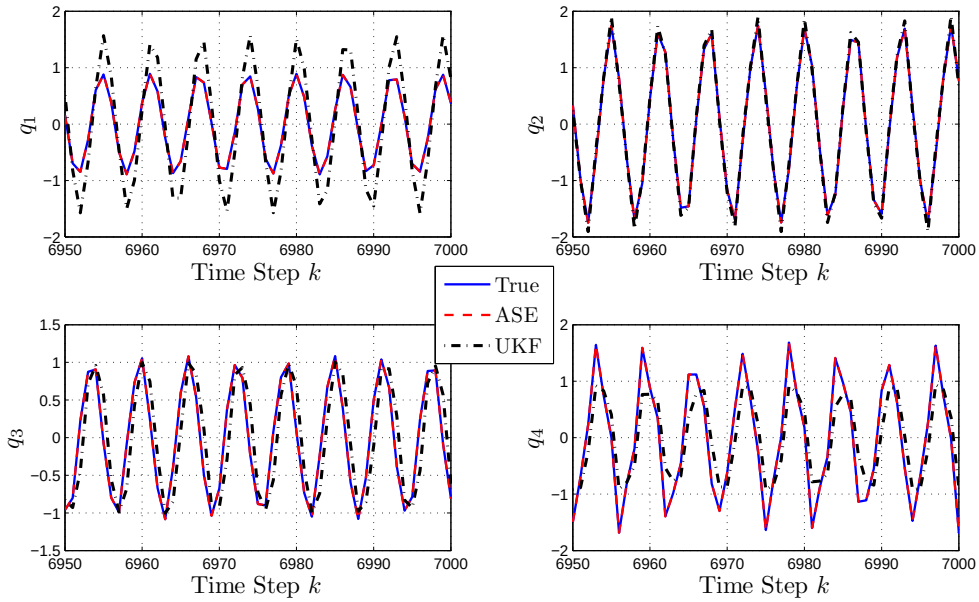


Figure 8.10: State estimates for the nonlinear planar linkage.

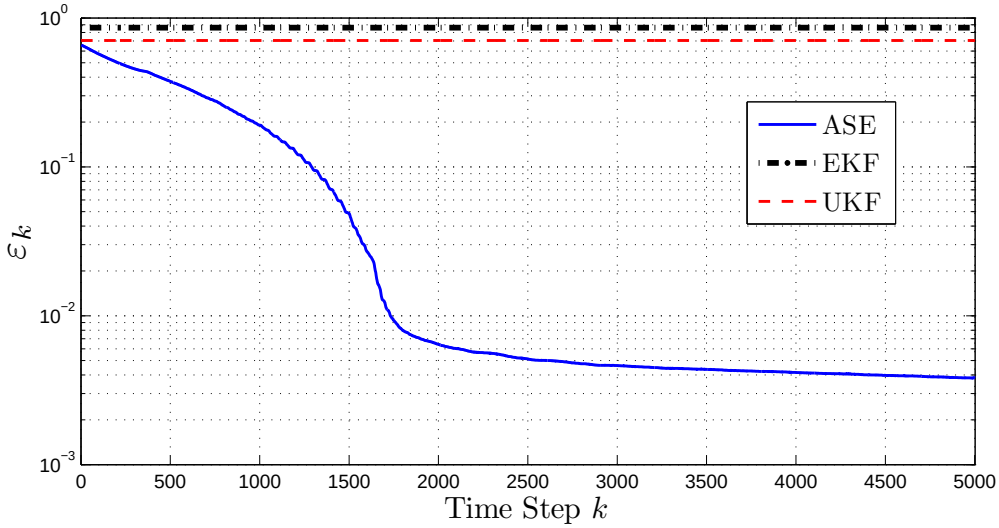


Figure 8.11: Comparison of the performance ε_k of the ASE, EKF, and UKF for the nonlinear planar linkage.

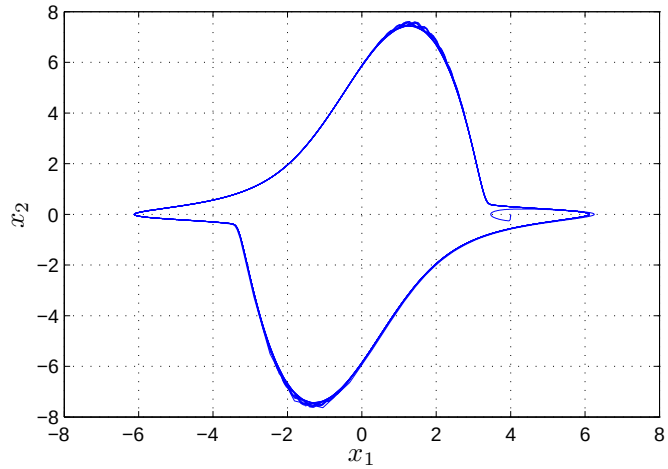


Figure 8.12: Phase portrait of the Van der Pol oscillator.

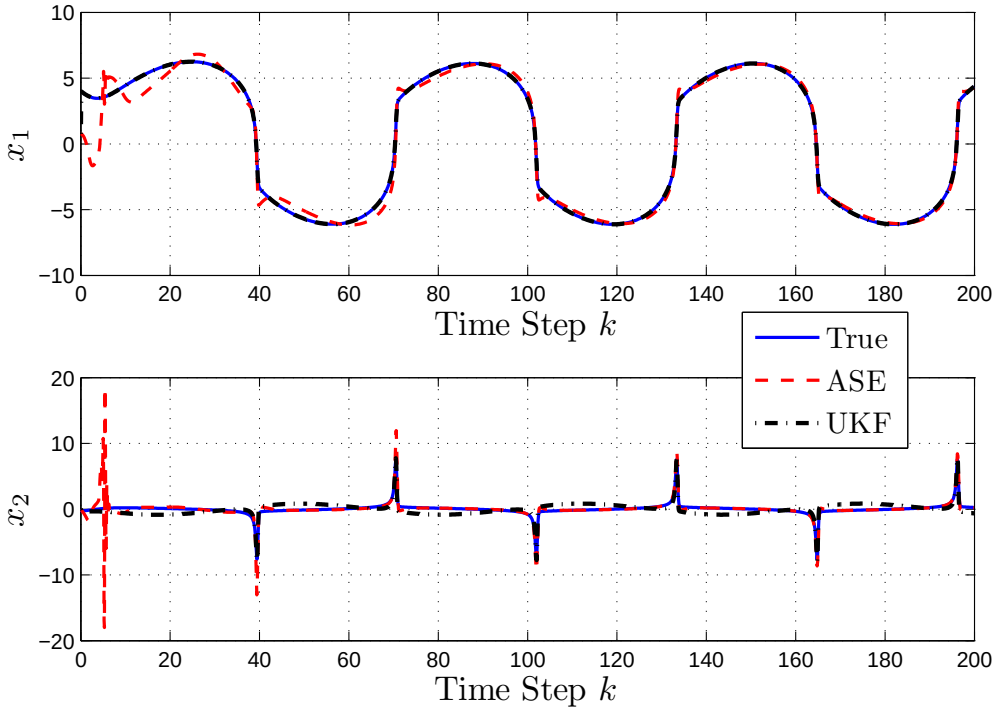


Figure 8.13: State estimates for the ASE and UKF.

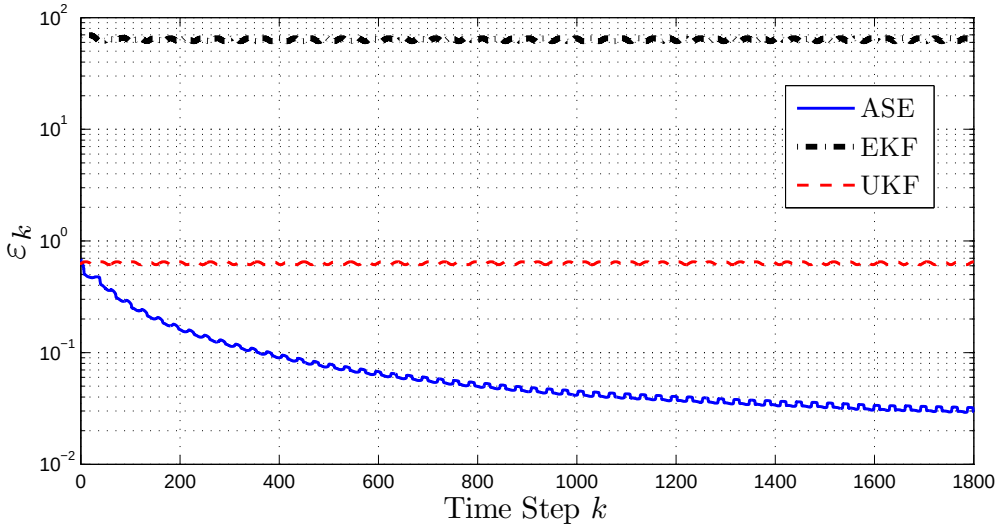


Figure 8.14: Comparison of the performance ε_k of the ASE, EKF, and UKF for the Van der Pol oscillator.

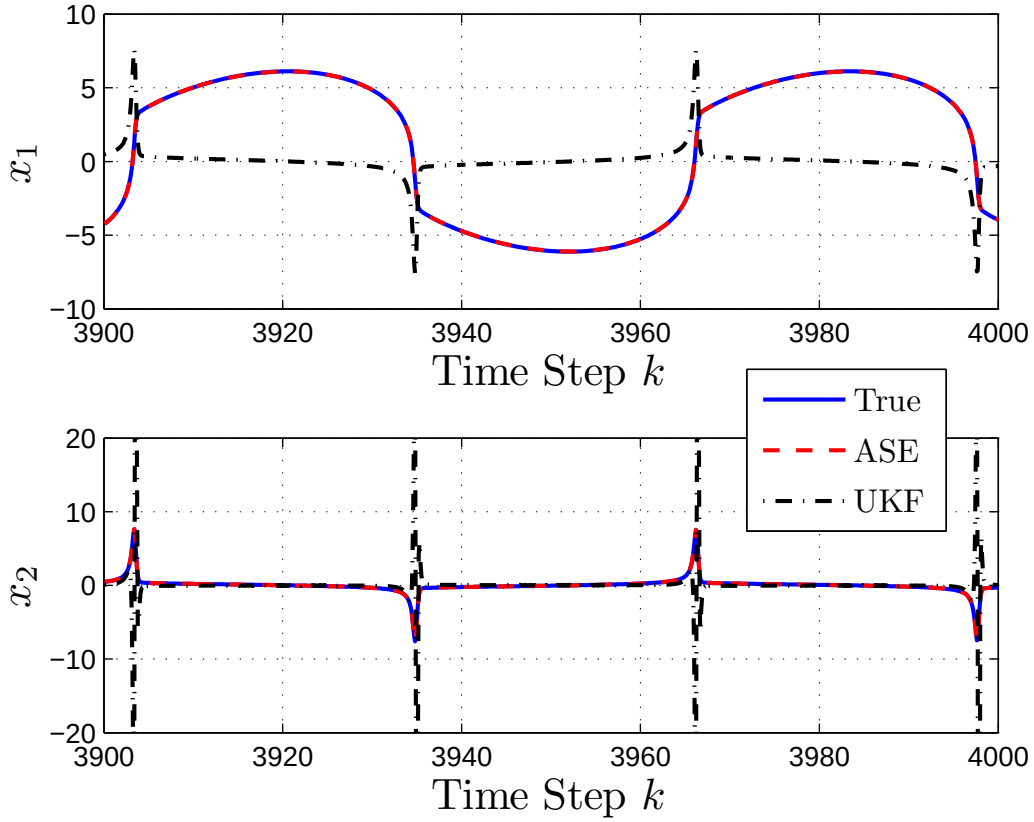


Figure 8.15: State estimates for the ASE and UKF.

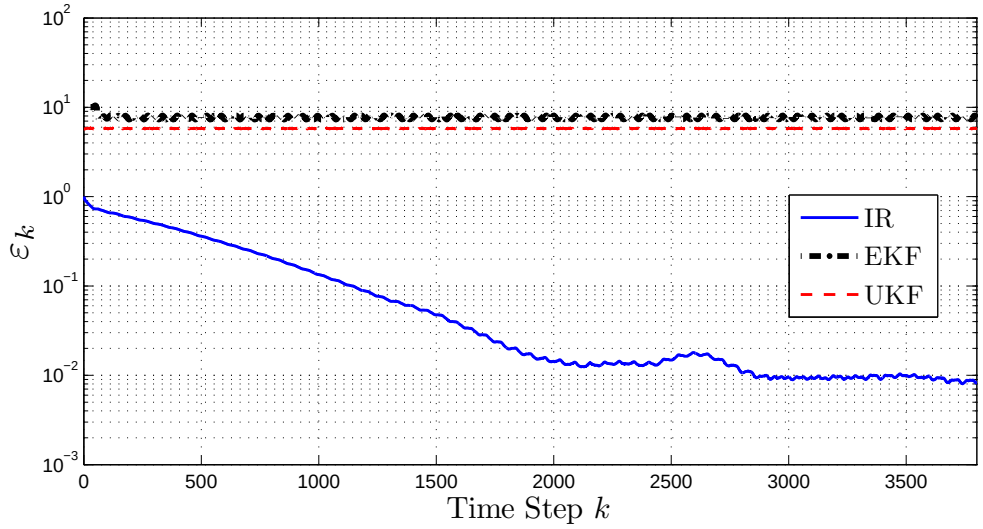


Figure 8.16: Comparison of the performance ε_k of the ASE, EKF, and UKF for the Van der Pol oscillator.

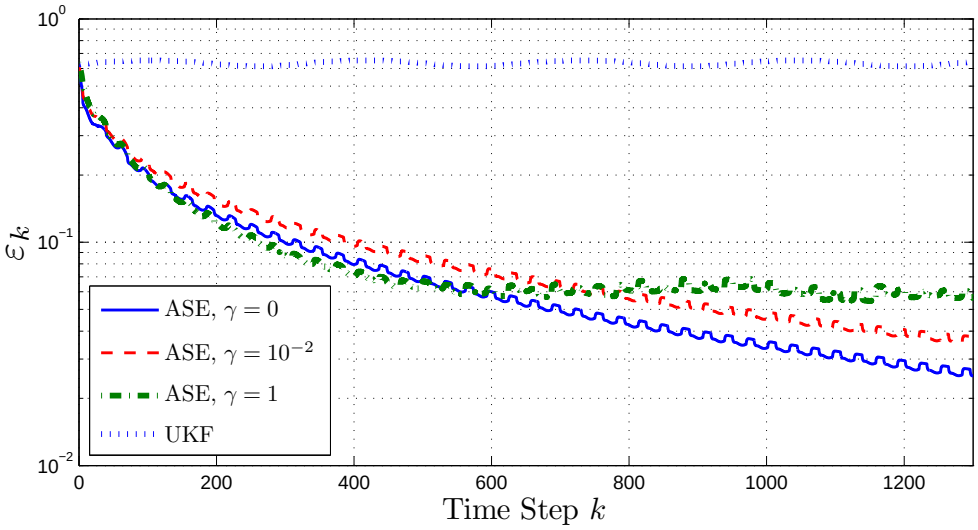


Figure 8.17: Comparison of the performance ε_k of the ASE for the Van der Pol oscillator with process noise.

CHAPTER IX

Adaptive Control

9.1 Introduction

One of the ironies of feedback control is that the underlying motivation for feedback is uncertainty, yet uncertain dynamics can degrade performance and render the closed-loop system unstable. Although robust control can mitigate this difficulty by trading performance for stability, unpredictable changes that occur during operation may render prior uncertainty bounds unreliable.

Unlike robust control, an adaptive controller is self-tuned during operation. This tuning accounts for the actual—and possibly changing—dynamics of the system as well as the nature of the external signals, such as commands and disturbances. Adaptive control may also be required for systems that are difficult to model due to either unknown physics or the inability to perform sufficiently accurate identification. Adaptive control may depend on prior modeling information, such as bounds on the model order and parameters, or it may entail explicit on-line identification. These approaches are known, respectively, as direct and indirect adaptive control.

Whether the adaptive controller is direct or indirect, it is desirable to develop algorithms that require the least amount of modeling information. In the most extreme case, LQG control, which is nonadaptive, requires a complete and exact model of the plant dynamics as well as a complete and exact model of the disturbance and sensor-

noise statistics and spectrum. Although robust control techniques increase tolerance to modeling errors, these methods require a nominal model as well as bounds on all uncertain parameters. These bounds can be determined by prior analysis or identification, but may become invalid during operation if changes occur, and, as noted above, they necessitate tradeoffs between stability and performance.

In adaptive control, the controller is tuned to the actual plant during operation. However, this ability comes at a cost. Adaptive control algorithms may require restrictive assumptions, such as full-state feedback, positive realness, minimum-phase zeros, matched uncertainty, and matched disturbances, as well as information on the sign of the high-frequency gain, relative degree, and zero locations [118, 119, 120, 121]. In particular, the starting point for this chapter is retrospective cost adaptive control (RCAC) [48, 50, 49, 58]. This direct adaptive control approach is applicable to MIMO (output feedback) plants that are possibly unstable and nonminimum phase (NMP) with uncertain command and disturbance spectra. The modeling information required by RCAC in [48, 50, 49, 58] is the first nonzero Markov parameter and locations of the NMP zeros, if any. Alternatively, a collection of Markov parameters can be used as long as a sufficient number is available to capture the locations of the NMP zeros.

This chapter extends prior RCAC results by developing a modification of RCAC that does not require knowledge of the locations of the NMP zeros if the system is either asymptotically stable or minimum phase. Instead, this extension requires knowledge of a limited number of Markov parameters; typically only one Markov parameter is sufficient. We analyze the algorithm by viewing it as a virtual input reconstruction technique [122], where the past control inputs for an idealized system are retrospectively optimized and subsequently used to update the controller parameters of the real system. The error dynamics of the reconstruction process may be unstable if the open-loop system has NMP zeros; however, if the open-loop system

is also asymptotically stable, then, by weighting the reconstructed controls, the error dynamics can be stabilized.

The algorithm developed in this chapter is demonstrated on both SISO and MIMO examples. In all cases, the number of Markov parameters that are used is not sufficient to determine the NMP zeros of the system. Consequently, these examples demonstrate the ability to control open-loop asymptotically stable NMP systems with unknown NMP zeros.

9.2 Problem Formulation

Consider the MIMO discrete-time system

$$x(k+1) = Ax(k) + Bu(k) + D_1 w(k), \tag{9.1}$$

$$y(k) = Cx(k) + D_2 w(k), \tag{9.2}$$

$$z(k) = E_1 x(k) + E_0 w(k), \tag{9.3}$$

where $x(k) \in \mathbb{R}^n$, $y(k) \in \mathbb{R}^{l_y}$, $z(k) \in \mathbb{R}^{l_z}$, $u(k) \in \mathbb{R}^{l_u}$, $w(k) \in \mathbb{R}^{l_w}$, $k \geq 0$, (A, B) is controllable, and (A, C) and (A, E) are observable. Although the state $x(k)$ is used for analysis, the algorithm described in this chapter does not rely on a state space representation. The goal is to develop an adaptive output feedback controller that minimizes the performance variable z in the presence of the exogenous signal w with minimal modeling information about the dynamics and w. A block diagram for (9.1)-(9.3) is shown in Figure 9.1, where

$$G(\mathbf{q}) = [G_{zw}(\mathbf{q}) \quad G_{zu}(\mathbf{q})], \tag{9.4}$$

$$G_{zw}(\mathbf{q}) \stackrel{\triangle}{=} E_1(\mathbf{q}I - A)^{-1}D_1 + E_0, \tag{9.5}$$

$$G_{zu}(\mathbf{q}) \stackrel{\triangle}{=} E_1(\mathbf{q}I - A)^{-1}B, \tag{9.6}$$

and

$$z(k) = G_{zw}(\mathbf{q})w(k) + G_{zu}(\mathbf{q})u(k), \tag{9.7}$$

where $\mathbf{q}$ is the forward-shift operator. The system (9.1)–(9.3) can represent a sampled-data application arising from a continuous-time system.

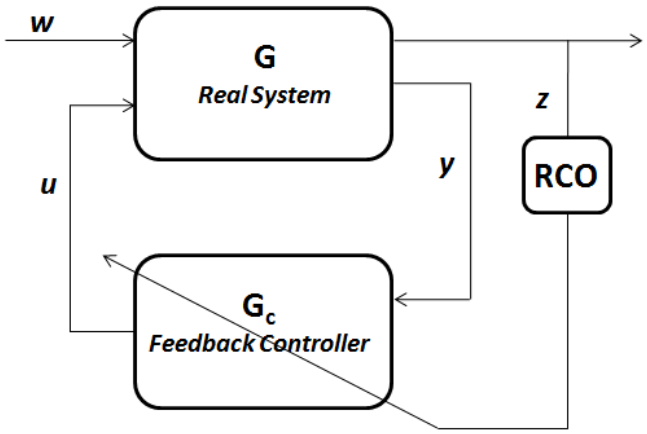


Figure 9.1: Disturbance-rejection and command-following architecture.

Note that w can represent either a command signal to be followed, an external disturbance to be rejected, or both. In particular, if $D_1 = 0$ and $E_0 \neq 0$, then the objective is to have the output $E_1 x$ follow the command signal $-E_0 w$. On the other hand, if $D_1 \neq 0$ and $E_0 = 0$, then the objective is to reject the disturbance w from the performance measurement $E_1 x$. Furthermore, if $D_1 = [\hat{D}_1 \quad 0]$, $E_0 = [0 \quad \hat{E}_0]$, and $w(k) = [w_1(k)^{\mathrm{T}} \quad w_2(k)^{\mathrm{T}}]^{\mathrm{T}}$, then the objective is to have $E_1 x$ follow the command $-\hat{E}_0 w_2$ while rejecting the disturbance w_1. Finally, if D_1 and E_0 are empty matrices, then the objective is output stabilization, that is, convergence of z to zero.

9.3 Retrospective Cost

For $i \geq 1$, define the Markov parameter of G_{zu} given by

$$H_i \triangleq E_1 A^{i-1} B. \tag{9.8}$$

For example, $H_1 = E_1 B$ and $H_2 = E_1 AB$. Let r be a positive integer. Then, for all $k \geq r$,

$$x(k) = A^r x(k-r) + \sum_{i=1}^{r} A^{i-1} B u(k-i) + \sum_{i=1}^{r} A^{i-1} D_1 w(k-i), \tag{9.9}$$

and thus

$$z(k) = E_1 A^r x(k-r) + \sum_{i=1}^{r} E_1 A^{i-1} D_1 w(k-i) + E_0 w(k) + \bar{H} \bar{U}(k-1), \tag{9.10}$$

where

$$\bar{H} \triangleq \left[\begin{array}{ccc} H_1 & \cdots & H_r \end{array} \right] \in \mathbb{R}^{l_z \times r l_u}$$

and

$$\bar{U}(k-1) \triangleq \left[\begin{array}{c} u(k-1) \\ \vdots \\ u(k-r) \end{array} \right].$$

Next, we partition the columns of $\bar{H}$ and the components of $\bar{U}(k-1)$ such that

$$\bar{H} \bar{U}(k-1) = \mathcal{H}' U'(k-1) + \mathcal{H} U(k-1), \tag{9.11}$$

where $\mathcal{H}' \in \mathbb{R}^{l_z \times (r l_u - l_U)}$, $\mathcal{H} \in \mathbb{R}^{l_z \times l_U}$, $U'(k-1) \in \mathbb{R}^{r l_u - l_U}$, and $U(k-1) \in \mathbb{R}^{l_U}$. As

shown below, only the Markov parameters in $\mathcal{H}$ are needed for the update law.

Then, we can rewrite (9.10) as

$$z(k) = \mathcal{S}(k) + \mathcal{H}U(k-1), \tag{9.12}$$

where

$$\mathcal{S}(k) \triangleq E_1 A^r x(k-r) + \sum_{i=1}^{r} E_1 A^{i-1} D_1 w(k-i) + E_0 w(k) + \mathcal{H}' U'(k-1). \tag{9.13}$$

For example, $\bar{H} = \left[\begin{array}{ccccc} H_1 & H_2 & H_3 & H_4 & H_5 \end{array} \right]$,

$$\mathcal{H}' = \left[\begin{array}{ccc} H_1 & H_2 & H_4 \end{array} \right], \quad U'(k-1) = \left[\begin{array}{c} u(k-1) \\ u(k-2) \\ u(k-4) \end{array} \right],$$

and

$$\mathcal{H} = \left[\begin{array}{cc} H_3 & H_5 \end{array} \right], \quad U(k-1) = \left[\begin{array}{c} u(k-3) \\ u(k-5) \end{array} \right].$$

Note that the decomposition of $\bar{H}\bar{U}(k-1)$ in (9.11) is not unique. Let s be a positive integer. Then, for $j = 1, \ldots, s$, we replace $\mathcal{H}$, $U(k-1)$, $\mathcal{H}'$, and $U'(k-1)$ in (9.11) with $\mathcal{H}_j \in \mathbb{R}^{l_z \times l_{U_j}}$, $U_j(k-1) \in \mathbb{R}^{l_{U_j}}$, $\mathcal{H}'_j \in \mathbb{R}^{l_z \times (rl_u - l_{U_j})}$, and $U'_j(k-1) \in \mathbb{R}^{rl_u - l_{U_j}}$, respectively, such that (9.11) becomes

$$\bar{H}\bar{U}(k-1) = \mathcal{H}'_j U'_j(k-1) + \mathcal{H}_j U_j(k-1). \tag{9.14}$$

Therefore, for $j = 1, \ldots, s$, we can rewrite (9.12) as

$$z(k) = \mathcal{S}_j(k) + \mathcal{H}_j U_j(k-1), \tag{9.15}$$

where

$$\mathcal{S}_j(k) \stackrel{\triangle}{=} E_1 A^r x(k-r) + \sum_{i=1}^{r} E_1 A^{i-1} D_1 w(k-i) + E_0 w(k) + \mathcal{H}_j' U_j'(k-1). \tag{9.16}$$

Next, let $0 \leq k_1 \leq k_2 \leq \cdots \leq k_s$. Replacing k by $k - k_j$ in (9.15) yields

$$z(k-k_j) = \mathcal{S}_j(k-k_j) + \mathcal{H}_j U_j(k-k_j-1). \tag{9.17}$$

Now, by stacking $z(k-k_1), \ldots, z(k-k_s)$, we define the *extended performance*

$$Z(k) \stackrel{\triangle}{=} \begin{bmatrix} z(k-k_1) \\ \vdots \\ z(k-k_s) \end{bmatrix} \in \mathbb{R}^{sl_z}. \tag{9.18}$$

Therefore,

$$Z(k) \stackrel{\triangle}{=} \tilde{\mathcal{S}}(k) + \tilde{\mathcal{H}} \tilde{U}(k-1), \tag{9.19}$$

where

$$\tilde{\mathcal{S}}(k) \stackrel{\triangle}{=} \begin{bmatrix} \mathcal{S}_1(k-k_1) \\ \vdots \\ \mathcal{S}_s(k-k_s) \end{bmatrix} \in \mathbb{R}^{sl_z} \tag{9.20}$$

and $\tilde{U}(k-1)$ has the form

$$\tilde{U}(k-1) \triangleq \begin{bmatrix} u(k-q_1) \\ \vdots \\ u(k-q_g) \end{bmatrix} \in \mathbb{R}^{gl_u}, \tag{9.21}$$

where $k_1 \leq q_1 < q_2 < \cdots < q_g \leq k_s + r$. The vector $\tilde{U}(k-1)$ is formed by stacking $U_1(k-k_1-1), \ldots, U_s(k-k_s-1)$ and removing copies of repeated components, and $\tilde{\mathcal{H}} \in \mathbb{R}^{sl_z \times gl_u}$ is constructed according to the structure of $\tilde{U}(k-1)$.

For example, with $s = 2$, $k_1 = 0$, and $k_2 = 2$, stacking $U_1(k-1) = \begin{bmatrix} u(k-1) \\ u(k-2) \end{bmatrix}$ and $U_2(k-3) = u(k-3)$ results in

$$\tilde{U}(k-1) = \begin{bmatrix} u(k-1) \\ u(k-2) \\ u(k-3) \end{bmatrix}, \quad \tilde{\mathcal{H}} = \begin{bmatrix} H_1 & H_2 & H_3 \\ 0 & H_1 & H_2 \\ 0 & 0 & H_3 \end{bmatrix}. \tag{9.22}$$

Note that $\tilde{\mathcal{H}}$ consists of the entries of $\mathcal{H}_1, \ldots, \mathcal{H}_s$ arranged according to the structure of $\tilde{U}(k-1)$.

Next, we define the *retrospective performance*

$$\hat{z}(k-k_j) \triangleq \mathcal{S}_j(k-k_j) + \mathcal{H}_j \hat{U}_j(k-k_j-1), \tag{9.23}$$

where the past controls $U_j(k-k_j-1)$ in (9.17) are replaced by the retrospective controls $\hat{U}_j(k-k_j-1)$, which are determined below. In analogy with (9.18), the

extended retrospective performance is defined as

$$\hat{Z}(k) \triangleq \begin{bmatrix} \hat{z}(k-k_1) \\ \vdots \\ \hat{z}(k-k_s) \end{bmatrix} \in \mathbb{R}^{sl_z} \tag{9.24}$$

and thus is given by

$$\hat{Z}(k) = \tilde{\mathcal{S}}(k) + \tilde{\mathcal{H}}\hat{\tilde{U}}(k-1), \tag{9.25}$$

where the components of $\hat{\tilde{U}}(k-1) \in \mathbb{R}^{l_{\tilde{U}}}$ are the components of $\hat{U}_1(k-k_1-1), \ldots, \hat{U}_s(k-k_s-1)$ ordered in the same way as the components of $\tilde{U}(k-1)$ in (9.19). Subtracting (9.19) from (9.25) yields

$$\hat{Z}(k) = Z(k) - \tilde{\mathcal{H}}\tilde{U}(k-1) + \tilde{\mathcal{H}}\hat{\tilde{U}}(k-1). \tag{9.26}$$

Finally, we define the *retrospective cost function*

$$J(\hat{\tilde{U}}(k-1), k) \triangleq \hat{Z}^{\mathrm{T}}(k)R_1(k)\hat{Z}(k) + \hat{\tilde{U}}^{\mathrm{T}}(k-1)R_2(k)\hat{\tilde{U}}(k-1), \tag{9.27}$$

where $R_1(k) \in \mathbb{R}^{l_z s \times l_z s}$ is the positive-definite performance weighting and $R_2(k) \in \mathbb{R}^{gl_u \times gl_u}$ is the positive-semidefinite control weighting. The goal is to determine retrospective controls $\hat{\tilde{U}}(k-1)$ that would have provided better performance than the controls $U(k)$ that were applied to the system. The retrospectively optimized controls $\hat{\tilde{U}}(k-1)$ are then used to update the controller. Substituting (9.26) into (9.27) yields

$$J(\hat{\tilde{U}}(k-1), k) = \hat{\tilde{U}}(k-1)^{\mathrm{T}}\mathcal{A}(k)\hat{\tilde{U}}(k-1) + \hat{\tilde{U}}^{\mathrm{T}}(k-1)\mathcal{B}^{\mathrm{T}}(k) + \mathcal{C}(k), \tag{9.28}$$

where

$$\mathcal{A}(k) \triangleq \tilde{\mathcal{H}}^{\mathrm{T}} R_1(k) \tilde{\mathcal{H}} + R_2(k), \tag{9.29}$$

$$\mathcal{B}(k) \triangleq 2\tilde{\mathcal{H}}^{\mathrm{T}} R_1(k)[Z(k) - \tilde{\mathcal{H}}\tilde{U}(k-1)], \tag{9.30}$$

$$\mathcal{C}(k) \triangleq Z^{\mathrm{T}}(k) R_1(k) Z(k) - 2Z^{\mathrm{T}}(k) R_1(k) \tilde{\mathcal{H}} \tilde{U}(k-1) + \tilde{U}^{\mathrm{T}}(k-1) \tilde{\mathcal{H}}^{\mathrm{T}} R_1(k) \tilde{\mathcal{H}} \tilde{U}(k-1). \tag{9.31}$$

If either $\tilde{\mathcal{H}}$ has full column rank or $R_2(k)$ is positive definite, then $\mathcal{A}(k)$ is positive definite. In this case, $J(\hat{\tilde{U}}(k-1), k)$ has the unique global minimizer

$$\hat{\tilde{U}}(k-1) = -\frac{1}{2} \mathcal{A}^{-1}(k) \mathcal{B}(k), \tag{9.32}$$

which are the retrospectively optimized controls. In Section 9.5 we show that (9.32) is the solution to a virtual input reconstruction problem.

For convenience we write $R_2(k) = \eta(k) R_{20}$, where R_{20} is positive definite and $\eta(k) \geq 0$. The scaling $\eta(k)$ can be used to bound the retrospectively optimized controls $\hat{\tilde{U}}(k-1)$ and thus indirectly bound the magnitude of the next control $u(k)$. We choose $\eta(k)$ to have the property that $\eta(k) \to 0$ as $k \to \infty$. For example, $\eta(k)$ may depend on k such as

$$\eta(k) = \frac{\eta_0}{k^a}, \tag{9.33}$$

where $\eta_0 \geq 0$, and $a \geq 1$ is an integer. Furthermore, $\eta(k)$ may be performance based [122, 38]

$$\eta(k) = \eta_0(k) ||Z(k)||_2^2 \tag{9.34}$$

or control-error based

$$\eta(k) = \eta_0(k)||\hat{\tilde{U}}(k-2) - \tilde{U}(k-2)||_2^2. \tag{9.35}$$

Alternatively, $\hat{\tilde{U}}(k-1)$ can be bounded directly by setting $R_2(k) = 0$ in (9.29) and replacing (9.32) by

$$\hat{\tilde{U}}(k-1) \stackrel{\triangle}{=} \mathrm{sat}_{[a,b]}[-\frac{1}{2}\mathcal{A}^{-1}(k)\mathcal{B}(k)], \tag{9.36}$$

where $\mathrm{sat}_{[a,b]}(\zeta)$ is the component-wise saturation function defined for scalar arguments by

$$\mathrm{sat}_{[a,b]}(\zeta) \stackrel{\triangle}{=} \begin{cases} b, & \text{if } \zeta \geq b, \\ \zeta, & \text{if } a < \zeta < b, \\ a & \text{if } \zeta \leq a, \end{cases} \tag{9.37}$$

where $a < b$ are the component-wise saturation levels.

9.4 Controller Construction

The control $u(k)$ is given by the strictly proper time-series controller of order n_c given by

$$u(k) = \sum_{i=1}^{n_c} M_i(k)u(k-i) + \sum_{i=1}^{n_c} N_i(k)y(k-i), \tag{9.38}$$

where, for all $i = 1, \ldots, n_c$, $M_i(k) \in \mathbb{R}^{l_u \times l_u}$ and $N_i(k) \in \mathbb{R}^{l_u \times l_y}$. The control (9.38) can be expressed as

$$u(k) = \theta(k)\phi(k-1), \tag{9.39}$$

where

$$\theta(k) \triangleq [M_1(k) \ \cdots \ M_{n_c}(k) \ N_1(k) \ \cdots \ N_{n_c}(k)] \in \mathbb{R}^{l_u \times n_c(l_u+l_y)} \tag{9.40}$$

and

$$\phi(k-1) \triangleq \begin{bmatrix} u(k-1) \\ \vdots \\ u(k-n_c) \\ y(k-1) \\ \vdots \\ y(k-n_c) \end{bmatrix} \in \mathbb{R}^{n_c(l_u+l_y)}. \tag{9.41}$$

9.4.1 Batch Least Squares Update of $\theta(k)$

Define the regressor matrix

$$\Phi_p(k-q_g-1) \triangleq \begin{bmatrix} L_{u,p}(k-q_g-1) & L_{y,p}(k-q_g-1) \end{bmatrix} \in \mathbb{R}^{p \times n_c(l_u+l_y)}, \tag{9.42}$$

where $p \geq 1$ is the data window size,

$$L_{y,p}(k-q_g-1) \triangleq \begin{bmatrix} y(k-q_g-1) & \cdots & y(k-n_c-q_g) \\ \vdots & \ddots & \vdots \\ y(k-q_g-p-1) & \cdots & y(k-n_c-q_g-p) \end{bmatrix} \tag{9.43}$$

and

$$L_{u,p}(k-q_g-1) \triangleq \begin{bmatrix} u(k-q_g-1) & \cdots & u(k-n_c-q_g) \\ \vdots & \ddots & \vdots \\ u(k-q_g-p-1) & \cdots & u(k-n_c-q_g-p) \end{bmatrix}. \tag{9.44}$$

Next, consider the quadratic cost

$$J_{\mathrm{B}}(\theta(k)) \triangleq \|\Phi_p(k-q_g-1)\theta^{\mathrm{T}}(k) - \Psi_p(k-q_g)\|^2 + \alpha(k)\mathrm{tr}[\theta(k)\theta^{\mathrm{T}}(k)], \tag{9.45}$$

where $\alpha(k) > 0$, and

$$\Psi_p(k-q_g) \triangleq \begin{bmatrix} \hat{u}^{\mathrm{T}}(k-q_g) \\ \vdots \\ \hat{u}^{\mathrm{T}}(k-q_g+1-p) \end{bmatrix} \in \mathbb{R}^{p\times l_u}. \tag{9.46}$$

Minimizing (9.45) yields the controller update

$$\theta^{\mathrm{T}}(k) = [\Phi_p^{\mathrm{T}}(k-q_g-1)\Phi_p(k-q_g-1) + \alpha(k)I]^{-1}\Phi_p^{\mathrm{T}}(k-q_g-1)\Psi_p(k-q_g). \tag{9.47}$$

Note that, since $\Psi_p(k-q_g)$ contains $\hat{u}(k-q_g)$, the update (9.47) is based on the g^{th} component of $\hat{\tilde{U}}(k-1)$. However any or all of the components of $\hat{\tilde{U}}(k-1)$ can be used to update $\theta(k)$.

9.4.2 Recursive Least Squares Update of $\theta(k)$

Using $\Psi_1(k-d) = \hat{u}^{\mathrm{T}}(k-q_g)$, we define the cumulative cost function

$$\begin{aligned} J_{\mathrm{R}}(\theta(k)) \triangleq & \sum_{i=q_g+1}^{k} \lambda^{k-i}\|\phi^{\mathrm{T}}(i-q_g-1)\theta^{\mathrm{T}}(k) - \hat{u}^{\mathrm{T}}(i-q_g)\|^2 \\ & + \lambda^k(\theta(k)-\theta(0))P^{-1}(0)(\theta(k)-\theta(0))^{\mathrm{T}}, \end{aligned} \tag{9.48}$$

where $\|\cdot\|$ is the Euclidean norm, for some $\varepsilon \in (0,1)$, $\lambda(k) \in (\varepsilon, 1]$ is the forgetting factor, and $P(0) \in \mathbb{R}^{n_c(l_u+l_y)\times n_c(l_u+l_y)}$ is positive definite. Minimizing (9.48) yields

$$\begin{aligned}\theta^{\mathrm{T}}(k) \triangleq\ & \theta^{\mathrm{T}}(k-1) + \beta(k)P(k-1)\phi(k-q_g-1) \\ & \cdot [\phi^{\mathrm{T}}(k-q_g-1)P(k-1)\phi(k-q_g-1) + \lambda(k)]^{-1} \\ & \cdot [\theta(k-1)\phi(k-q_g-1) - \hat{u}(k-q_g)]^{\mathrm{T}}, \end{aligned} \tag{9.49}$$

where $\beta(k)$ is either 0 or 1. When $\beta(k) = 1$, the controller is allowed to adapt, whereas, when $\beta(k) = 0$, the adaptation is off. $P(k)$ is updated by

$$\begin{aligned}P(k) \triangleq\ & (1-\beta(k))P(k-1) + \beta(k)\lambda^{-1}(k)P(k-1) - \beta(k)\lambda^{-1}(k)P(k-1) \\ & \cdot \phi(k-q_g-1)[\phi^{\mathrm{T}}(k-q_g-1)P(k-1)\phi(k-q_g-1) + \lambda(k)]^{-1} \\ & \cdot \phi^{\mathrm{T}}(k-q_g-1)P(k-1). \end{aligned} \tag{9.50}$$

We initialize $P(0) = \gamma I$, where $\gamma > 0$.

9.5 Convergence Analysis

In this section we consider three systems, namely, the *real system*, with performance $z(k)$ and state $x(k)$; the *retrospective system*, with performance $\hat{z}(k)$ and state $\hat{x}(k)$; and, finally, the *ideal system*, with performance $z^*(k)$ and state $x^*(k)$. The goal is to determine conditions under which the state of the real system tends to the state of the retrospective system. We also provide conditions under which the state of the retrospective system tends to the state of the ideal system. Under these conditions, the performance $z(k)$ tends to zero and the state $x(k)$ of the real system is bounded.

9.5.1 Sufficient Conditions for $z(k) - \hat{z}(k) \to 0$ as $k \to \infty$

Consider the *retrospective system*

$$\hat{x}(k) = A^r x(k-r) + \sum_{i=1}^{r} A^{i-1} D_1 w(k-i) + \mathcal{K}' U'(k-1) + \mathcal{K}\hat{U}(k-1), \tag{9.51}$$

$$\begin{aligned}\hat{z}(k) &= E_1 \hat{x}(k) + E_0 w(k), \\ &= E_1 A^r x(k-r) + \sum_{i=1}^{r} E_1 A^{i-1} D_1 w(k-i) + E_0 w(k) \\ &+ \mathcal{H}' U'(k-1) + \mathcal{H}\hat{U}(k-1),\end{aligned} \tag{9.52}$$

where

$$U'(k-1) \triangleq \begin{bmatrix} u(k-\sigma_1') \\ \vdots \\ u(k-\sigma_{rl_u-l_U}') \end{bmatrix}, \quad \hat{U}(k-1) \triangleq \begin{bmatrix} \hat{u}(k-\sigma_1) \\ \vdots \\ \hat{u}(k-\sigma_{l_U}) \end{bmatrix},$$

where the sets $\{\sigma_1', \ldots, \sigma_{rl_u-l_U}'\}$ and $\{\sigma_1, \ldots, \sigma_{l_U}\}$ are a partition of $\{1, \ldots, r\}$, and $\mathcal{K} \in \mathbb{R}^{n \times l_U}$ and $\mathcal{K}' \in \mathbb{R}^{n \times (rl_u - l_U)}$ satisfy $\mathcal{H} = E_1\mathcal{K}$ and $\mathcal{H}' = E_1\mathcal{K}'$, respectively.

Next, consider the *extended retrospective system*

$$\hat{X}(k+1) = \tilde{A}X(k) + \tilde{B}\hat{\tilde{U}}(k) + \tilde{B}'\tilde{U}'(k) + \tilde{D}_1 W(k), \tag{9.53}$$

$$\hat{Z}(k) = \tilde{E}_1 \hat{X}(k) + \tilde{E}_0 W(k), \tag{9.54}$$

where the state $X(k)$ is an input to the extended retrospective system,

$$\hat{X}(k) \triangleq \begin{bmatrix} \hat{x}(k) \\ \vdots \\ \hat{x}(k-k_s-r+1) \end{bmatrix} \in \mathbb{R}^{n(k_s+r)}, \quad X(k) \triangleq \begin{bmatrix} x(k) \\ \vdots \\ x(k-k_s-r+1) \end{bmatrix} \in \mathbb{R}^{n(k_s+r)},$$
$$W(k) \triangleq \begin{bmatrix} w(k) \\ \vdots \\ w(k-k_s-r+1) \end{bmatrix} \in \mathbb{R}^{l_w(k_s+r)}, \quad \tilde{U}'(k) \triangleq \begin{bmatrix} u(k-q_1'+1) \\ \vdots \\ u(k-q_{k_s+r-g}'+1) \end{bmatrix} \in \mathbb{R}^{l_u(k_s+r-g)}, \tag{9.55}$$

$\tilde{A} \triangleq I_{(k_s+r)} \otimes A \in \mathbb{R}^{n(k_s+r)\times n(k_s+r)}$, $\tilde{D}_1 \triangleq I_{(k_s+r)} \otimes D_1 \in \mathbb{R}^{n(k_s+r)\times l_w(k_s+r)}$, and $\otimes$ is the Kronecker product. $\tilde{E}_0 \in \mathbb{R}^{sl_z \times l_w(k_s+r)}$ is the matrix with block entries E_0 and $0_{l_z \times l_w}$ that satisfies

$$\begin{bmatrix} w(k-k_1) \\ \vdots \\ w(k-k_s) \end{bmatrix} = \tilde{E}_0 \begin{bmatrix} w(k) \\ \vdots \\ w(k-k_s-r+1) \end{bmatrix}. \tag{9.56}$$

$\tilde{E}_1 \in \mathbb{R}^{sl_z \times n(k_s+r)}$ is the matrix with block entries E_1 and $0_{l_z \times n}$ that satisfies

$$\begin{bmatrix} x(k-k_1) \\ \vdots \\ x(k-k_s) \end{bmatrix} = \tilde{E}_1 \begin{bmatrix} x(k) \\ \vdots \\ x(k-k_s-r+1) \end{bmatrix}. \tag{9.57}$$

$\tilde{B} \in \mathbb{R}^{n(k_s+r)\times gl_u}$ and $\tilde{B}' \in \mathbb{R}^{n(k_s+r)\times(k_s+r-g)l_u}$ are the matrices with block entries B and $0_{n\times l_u}$ that satisfy

$$\tilde{B}\tilde{U}(k-1) + \tilde{B}'\tilde{U}'(k-1) = \begin{bmatrix} Bu(k-1) \\ \vdots \\ Bu(k-k_s-r) \end{bmatrix} \in \mathbb{R}^{n(k_s+r)}. \tag{9.58}$$

Note that the sets $\{q_1, \ldots, q_g\}$ and $\{q'_1, \ldots, q'_{k_s+r-g}\}$ are a partition of $\{1, \ldots, k_s+r\}$.

Assumption 9.5.1. *Assume that* $\theta(k-1)\phi(k-q_g-1)-\hat{u}(k-q_g) \to 0$ *as* $k \to \infty$.

Assumption 9.5.1 allows us to assume that the fit between $u(k)$ and $\hat{u}(k)$ is successful, and allows to focus on the subsequent behavior of the retrospective system.

The next result provides a sufficient condition under which $x(k) - \hat{x}(k) \to 0$ as $k \to \infty$.

Fact 9.5.1. Let $\hat{\hat{U}}(k-1)$ be given by (9.32) and let $\hat{x}(k)$ be given by (9.51), and let $\theta(k)$ be updated by (9.49) and (9.50). If Assumption 9.5.1 is satisfied, then $x(k) - \hat{x}(k) \to 0$ as $k \to \infty$, and thus $z(k) - \hat{z}(k) \to 0$ as $k \to \infty$.

Proof. First, we write (9.9) as

$$x(k) = A^r x(k-r) + \sum_{i=1}^{r} A^{i-1} D_1 w(k-i) + \mathcal{K}' U'(k-1) + \mathcal{K} U(k-1). \tag{9.59}$$

Subtracting (9.51) from (9.59) yields

$$x(k)-\hat{x}(k) = \mathcal{K}[U(k-1) - \hat{U}(k-1)]. \tag{9.60}$$

Furthermore, using (9.39) we rewrite the components of $U(k-1)$ as

$$U(k-1) = \begin{bmatrix} \theta(k-\sigma_1)\phi(k-\sigma_1-1) \\ \vdots \\ \theta(k-\sigma_{l_U})\phi(k-\sigma_{l_U}-1) \end{bmatrix}. \tag{9.61}$$

Therefore, (9.60) becomes

$$x(k) - \hat{x}(k) = \mathcal{K} \begin{bmatrix} \theta(k-\sigma_1-1)\phi(k-\sigma_1-1) - \hat{u}(k-\sigma_1) \\ \vdots \\ \theta(k-\sigma_{l_U}-1)\phi(k-\sigma_{l_U}-1) - \hat{u}(k-\sigma_{l_U}) \end{bmatrix} + O(k), \tag{9.62}$$

where

$$O(k) \stackrel{\triangle}{=} \mathcal{K} \begin{bmatrix} \theta(k-\sigma_1) - \theta(k-\sigma_1-1)\phi(k-\sigma_1-1) \\ \vdots \\ \theta(k-\sigma_{l_U}) - \theta(k-\sigma_1-1)\phi(k-\sigma_{l_U}-1) \end{bmatrix}. \tag{9.63}$$

Since $\theta(k-1)\phi(k-q_g-1) - \hat{u}(k-q_g) \to 0$ as $k \to \infty$, it follows from (9.49) that $\theta(k) - \theta(k-1) \to 0$ as $k \to \infty$, which implies that $O(k) \to 0$ as $k \to \infty$. Therefore, $x(k) - \hat{x}(k) \to 0$ as $k \to \infty$, which implies $z(k) - \hat{z}(k) = E_1[x(k) - \hat{x}(k)] \to 0$ as $k \to \infty$. □

The following result gives conditions under which $\hat{Z}(k) \equiv 0$.

Fact 9.5.2. Assume that $\tilde{\mathcal{H}}$ is nonsingular, $R_1(k) \equiv I$, $R_2(k) \equiv 0$, and define $\hat{\tilde{U}}(k-1)$ by (9.32). Then $\hat{Z}(k) \equiv 0$.

Proof. Since $Z(k)$ is in the range of $\tilde{\mathcal{H}}$, there exists $Q(k) \in \mathbb{R}^{sl_{\tilde{u}}}$ such that $Z(k) = \tilde{\mathcal{H}}Q(k)$. Substituting (9.32) into (9.26) yields

$$\begin{aligned} \hat{Z}(k) &= Z(k) + \tilde{\mathcal{H}}(\tilde{\mathcal{H}}^{\mathrm{T}}\tilde{\mathcal{H}})^{-1}\tilde{\mathcal{H}}^{\mathrm{T}}(-Z(k) + \tilde{\mathcal{H}}\tilde{U}(k-1)) - \tilde{\mathcal{H}}\tilde{U}(k-1) \\ &= Z(k) - \tilde{\mathcal{H}}(\tilde{\mathcal{H}}^{\mathrm{T}}\tilde{\mathcal{H}})^{-1}\tilde{\mathcal{H}}^{\mathrm{T}}Z(k) \\ &= \tilde{\mathcal{H}}Q(k) - \tilde{\mathcal{H}}(\tilde{\mathcal{H}}^{\mathrm{T}}\tilde{\mathcal{H}})^{-1}\tilde{\mathcal{H}}^{\mathrm{T}}\tilde{\mathcal{H}}Q(k) = 0. \end{aligned}$$ □

Fact 9.5.3. Assume that $\tilde{\mathcal{H}}$ is nonsingular, $R_1(k) \equiv I$, and $R_2 \equiv 0$. Furthermore, assume that Assumption 9.5.1 is satisfied, let $\hat{x}(k)$ be given by (9.51), let $\theta(k)$ be updated by (9.49) and (9.50), and let $\hat{\tilde{U}}(k-1)$ be given by (9.32). Then $Z(k) \to 0$ as $k \to \infty$.

Proof. It follows from Fact 9.5.1 that $z(k) - \hat{z}(k) \to 0$ as $k \to \infty$, and thus $Z(k) - \hat{Z}(k) \to 0$ as $k \to \infty$. It follows from Fact 9.5.2 that $\hat{Z}(k) \equiv 0$. Hence $Z(k) \to 0$, as $k \to \infty$. □

In view of Fact 9.5.1, we assume henceforth that k is sufficiently large that the difference between $\hat{x}(k)$, $\hat{u}(k)$, $\hat{y}(k)$, and $\hat{z}(k)$ and $x(k)$, $u(k)$, $y(k)$, and $z(k)$, respectively, is negligible. The following analysis focuses on the subsequent behavior of $\hat{x}(k)$, $\hat{u}(k)$, and $\hat{z}(k)$ in the case where $\eta(k) \equiv 0$ and $R_1(k) \equiv I$.

9.5.2 Boundedness of the State

The *ideal system* is defined by

$$x^*(k) = A^r x^*(k-r) + \sum_{i=1}^{r} A^{i-1} D_1 w(k-i) + \mathcal{K}' U'(k-1) + \mathcal{K} U^*(k-1), \tag{9.64}$$

$$\begin{aligned} z^*(k) &= E_1 A^r x^*(k-r) + \sum_{i=1}^{r} E_1 A^{i-1} D_1 w(k-i) + E_0 w(k) \\ &\quad + \mathcal{H}' U'(k-1) + \mathcal{H} U^*(k-1), \end{aligned} \tag{9.65}$$

where $x^*(k)$ is the state of the ideal system and $U^*(k-1)$ is defined analogously to $U(k-1)$, with $u(k)$ replaced by $u^*(k)$, where

$$u^*(k) = \theta^* \phi^*(k-1), \tag{9.66}$$

$$\phi^*(k-1) \triangleq \begin{bmatrix} u^*(k-1) \\ \vdots \\ u^*(k-n_{\mathrm{c}}) \\ y^*(k-1) \\ \vdots \\ y^*(k-n_{\mathrm{c}}) \end{bmatrix}. \tag{9.67}$$

The ideal controller θ^* is assumed to yield the ideal performance

$$z^*(k) \equiv 0. \tag{9.68}$$

Adding and subtracting $E_1 A^r \hat{x}(k-r)$ to and from (9.65) yields

$$z^*(k) = \mathcal{S}(k) + E_1 A^r e(k-r) + \mathcal{H} U^*(k-1), \tag{9.69}$$

where $\mathcal{S}(k)$ is defined by (9.13) with $x(k)$ replaced by $\hat{x}(k)$, and $e(k) \triangleq x^*(k) - \hat{x}(k)$.

The *extended ideal system* is given by

$$X^*(k+1) = \tilde{A} X^*(k) + \tilde{B} \tilde{U}^*(k) + \tilde{B}' \tilde{U}'(k) + \tilde{D}_1 W(k), \tag{9.70}$$

$$Z^*(k) = \tilde{\mathcal{S}}(k) + \tilde{E}_1 \tilde{A}^r E(k-1) + \tilde{\mathcal{H}} \tilde{U}^*(k-1) = 0, \tag{9.71}$$

where $X^*(k+1)$ and $Z^*(k)$ are defined in the same way as $X(k+1)$ and $Z(k)$, $E(k) \triangleq X^*(k) - \hat{X}(k)$, and

$$\tilde{U}^*(k) \triangleq [I_{l_{\tilde{U}}} \otimes \theta^*] \tilde{\phi}^*(k-1), \tag{9.72}$$

$$\tilde{\phi}^*(k) \triangleq \begin{bmatrix} \phi^*(k-q_1) \\ \vdots \\ \phi^*(k-q_{l_{\tilde{U}}}) \end{bmatrix}. \tag{9.73}$$

Fact 9.5.4. Assume $\tilde{\mathcal{H}}$ is nonsingular, $R_1(k) \equiv I$, and $R_2(k) \equiv 0$. Furthermore, assume that Assumption 9.5.1 is satisfied, let $\hat{x}(k)$ be given by (9.51), let $\theta(k)$ be updated by (9.49) and (9.50), and let $\hat{\tilde{U}}(k-1)$ be given by (9.32). If $\tilde{A} - \tilde{B} \tilde{\mathcal{H}}^\dagger \tilde{E}_1 \tilde{A}^r$ is asymptotically stable, then $x(k) - x^*(k) \to 0$ as $k \to \infty$, and thus, $z(k) = z(k) - z^*(k) \to 0$ as $k \to \infty$.

Proof. Subtracting (9.25) from (9.71) and solving for $\hat{\tilde{U}}(k-1)$ yields

$$\hat{\tilde{U}}(k-1) = \tilde{\mathcal{H}}^{\dagger}[\tilde{E}_1\tilde{A}^r E(k-1) + \tilde{\mathcal{H}}\tilde{U}^*(k-1) + \hat{Z}(k)], \tag{9.74}$$

where $\tilde{\mathcal{H}}^{\dagger}\tilde{\mathcal{H}} = I_{l_{\tilde{U}}}$. By Fact 9.5.2, $\hat{Z}(k) \equiv 0$ and therefore (9.74) implies

$$\hat{\tilde{U}}(k-1) = \tilde{\mathcal{H}}^{\dagger}\tilde{E}_1\tilde{A}^r E(k-1) + \tilde{U}^*(k-1). \tag{9.75}$$

Subtracting (9.53) from (9.70), and using (9.75) yields the error dynamics

$$E(k) = (\tilde{A} - \tilde{B}\tilde{\mathcal{H}}^{\dagger}\tilde{E}_1\tilde{A}^r)E(k-1). \qquad \square$$

The relationship between the poles of $\tilde{A} - \tilde{B}\tilde{\mathcal{H}}^{\dagger}\tilde{E}_1\tilde{A}^r$ and the transmission zeros of the real open-loop system is discussed in Fact 9.5.6.

9.5.3 Analysis of the case $r = s = 1$

Consider the case $\eta(k) \equiv 0$, $r = s = 1$, so that, $\tilde{\mathcal{H}}' = 0_{l_z \times l_u}$, and $\tilde{\mathcal{H}} = H_1 = E_1 B$. Furthermore, assume that H_1 is square and invertible. Substituting (9.10) in (9.32) yields

$$\hat{u}(k-1) = -H_1^{-1}[E_1 A x(k-1) + E_1 D_1 w(k-1)], \tag{9.76}$$

which yields the closed-loop retrospective system

$$\hat{x}(k) = (A - BH_1^{-1}E_1 A)x(k-1) + (D_1 - BH_1^{-1}E_1 D_1)w(k-1), \tag{9.77}$$

$$\hat{z}(k) = E_1\hat{x}(k). \tag{9.78}$$

Furthermore, substituting (9.77) into (9.78) yields $\hat{z}(k) \equiv 0$, while the error dynamics are

$$E(k+1) = (A - BH_1^{-1}E_1A)E(k). \tag{9.79}$$

The following result shows that the stability of (9.79) depends on the transmission zeros of (A, B, E_1).

Fact 9.5.5. Assume that $l_z = l_u$ and H_1 is nonsingular. Then $A - BH_1^{-1}E_1A$ is asymptotically stable if and only if (A, B, E_1) is minimum phase.

Proof. It follows from (9.6) that

$$\mathbf{q}G_{zu}(\mathbf{q}) = E_1A(\mathbf{q}I - A)^{-1}B + H_1 \tag{9.80}$$

$$= \frac{1}{\alpha(\mathbf{q})}\beta(\mathbf{q}), \tag{9.81}$$

where $\alpha(\mathbf{q}) \in \mathbb{R}[\mathbf{q}]$ and $\beta(\mathbf{q}) \in \mathbb{R}^{l_z \times l_z}[\mathbf{q}]$ are defined by

$$\begin{aligned}
\alpha(\mathbf{q}) &\stackrel{\triangle}{=} \det(\mathbf{q}I - A) \\
&= \mathbf{q}^n + \alpha_1\mathbf{q}^{n-1} + \cdots + \alpha_{n-1}\mathbf{q} + \alpha_n, \qquad (9.82) \\
\beta(\mathbf{q}) &\stackrel{\triangle}{=} E_1A\mathrm{adj}(\mathbf{q}I - A) + \alpha(\mathbf{q})H_1 \\
&= \mathbf{q}^n\beta_0 + \mathbf{q}^{n-1}\beta_1 + \cdots + \mathbf{q}^2\beta_{n-2} + \mathbf{q}\beta_{n-1}. \qquad (9.83)
\end{aligned}$$

Furthermore, note that

$$\det\ \mathbf{q}G_{zu}(\mathbf{q}) = \frac{\det\ \beta(\mathbf{q})}{\alpha^{l_z}(\mathbf{q})}. \tag{9.84}$$

Since H_1 is nonsingular, it follows that $\mathbf{q}G_{zu}(\mathbf{q})$ has full normal rank, det β(q) is not the zero polynomial, and the transmission zeros of $\mathbf{q}G_{zu}(\mathbf{q})$ are the roots of det $\beta(\mathbf{q})$.

Hence (9.81) implies that

$$\mathbf{q}^{-1}G_{zu}^{-1}(\mathbf{q}) = \frac{\alpha(\mathbf{q})}{\det\ \beta(\mathbf{q})}\text{adj}\ \beta(\mathbf{q}). \tag{9.85}$$

From (9.77), it follows that $\det\ \beta(\mathbf{q}) = \det\ (\mathbf{q}I - A + BH_1^{-1}E_1A)$. Therefore, the eigenvalues of $A - BH_1^{-1}E_1A$ are the roots of $\det\ \beta(\mathbf{q})$. Consequently, (A, B, E_1) is minimum phase if and only if $\mathbf{q}^{-1}G_{zu}^{-1}(\mathbf{q})$ is asymptotically stable. □

Fact 9.5.6. Assume that $l_z = l_u < n$ and H_1 is nonsingular. If $(A-BH_1^{-1}E_1A, E_1)$ is detectable, then (A, B, E_1) is minimum phase.

Proof. To prove necessity, define

$$\mathcal{O} \triangleq \begin{bmatrix} E_1 \\ E_1(A - BH_1^{-1}E_1A) \\ \vdots \\ E_1(A - BH_1^{-1}E_1A)^{n-1} \end{bmatrix}, \tag{9.86}$$

which is the observability matrix of $(A - BH_1^{-1}E_1A, BH_1^{-1}, E_1)$. Since $H_1 = E_1B$, it follows that $E_1(A - BH_1^{-1}E_1A) = 0$. Therefore,

$$\mathcal{O} \triangleq \begin{bmatrix} E_1 \\ 0_{l_z\times n} \\ \vdots \\ 0_{l_z\times n} \end{bmatrix}, \tag{9.87}$$

and thus, rank $\mathcal{O} = l_z$. Since (A, B, E_1) is minimum phase it follows from Fact 9.5.5 that $A - BH_1^{-1}E_1A$ is asymptotically stable, and thus $(A - BH_1^{-1}E_1A, E_1)$ is detectable. □

9.6 Convergence Analysis with Control Weighting

Unlike Section 9.5.9.5.2, where $\eta(k) \equiv 0$, we now assume that $\eta(k) \geq 0$. In this case, with $\hat{\tilde{U}}(k-1)$ given by (9.32), it follows that $\hat{Z}(k)$ is given by

$$\hat{Z}(k) = Z(k) - \tilde{\mathcal{H}}\tilde{U}(k-1) + \tilde{\mathcal{H}}[\tilde{\mathcal{H}}^{\mathrm{T}}R_1(k)\tilde{\mathcal{H}} + \eta(k)R_{20}]^{-1}\tilde{\mathcal{H}}^{\mathrm{T}}R_1(k)[-Z(k) + \tilde{\mathcal{H}}\tilde{U}(k-1)]. \quad (9.88)$$

Furthermore in view of Fact 9.5.1, we assume henceforth that k is sufficiently large that the difference between $\hat{x}(k)$, $\hat{u}(k)$, $\hat{y}(k)$, and $\hat{z}(k)$ and $x(k)$, $u(k)$, $y(k)$, and $z(k)$, respectively, is negligible. The following analysis focuses on the subsequent behavior of $\hat{x}(k)$ and $\hat{z}(k)$ under the assumption that $\eta(k) \geq 0$.

Substituting (9.32) into (9.53) and replacing $X(k-1)$ in (9.53) with $\hat{X}(k-1)$ yields

$$\begin{aligned}\hat{X}(k) &= \tilde{A}\hat{X}(k-1) + \tilde{B}[\tilde{\mathcal{H}}^{\mathrm{T}}R_1(k-1)\tilde{\mathcal{H}} \\ &\quad + \eta(k-1)R_{20}(k-1)]^{-1}\tilde{\mathcal{H}}^{\mathrm{T}}R_1(k-1)[-\hat{Z}(k) + \tilde{\mathcal{H}}\hat{\tilde{U}}(k-1)] \\ &\quad + \tilde{B}'\hat{\tilde{U}}'(k-1) + \tilde{D}_1W(k-1), \quad (9.89)\\ \hat{Z}(k) &= \tilde{E}_1\hat{X}(k) + \tilde{E}_0W(k). \quad (9.90)\end{aligned}$$

Next, we write the extended retrospective performance as

$$\hat{Z}(k) = \tilde{E}_1\tilde{A}^r\hat{X}(k-1) + \tilde{\mathcal{H}}\hat{\tilde{U}}(k-1) + \tilde{\mathcal{H}}'\hat{\tilde{U}}'(k-1) + \tilde{D}\tilde{A}^rW(k-1). \quad (9.91)$$

Substituting (9.91) into (9.89) yields

$$\begin{aligned}\hat{X}(k) &= \tilde{A}_{\mathrm{VIR}}(k-1)\hat{X}(k-1) \\ &\quad + [\tilde{D}_1 - \tilde{B}[\tilde{\mathcal{H}}^{\mathrm{T}}R_1(k-1)\tilde{\mathcal{H}} + \eta(k-1)R_{20}(k-1)]^{-1}\tilde{\mathcal{H}}^{\mathrm{T}}R_1(k-1)\tilde{D}\tilde{A}^r]W(k-1) \\ &\quad + [\tilde{B} - \tilde{B}[\tilde{\mathcal{H}}^{\mathrm{T}}R_1(k-1)\tilde{\mathcal{H}} + \eta(k-1)R_{20}(k-1)]^{-1}\tilde{\mathcal{H}}^{\mathrm{T}}R_1(k-1)\tilde{\mathcal{H}}']\hat{\tilde{U}}'(k-1), \quad (9.92)\end{aligned}$$

where

$$\tilde{A}_{\mathrm{VIR}}(k) \triangleq \tilde{A} - \tilde{B}[\tilde{\mathcal{H}}^{\mathrm{T}} R_1(k) \tilde{\mathcal{H}} + \eta(k) R_{20}(k)]^{-1} \tilde{\mathcal{H}}^{\mathrm{T}} R_1(k) \tilde{E}_1 \tilde{A}^r. \tag{9.93}$$

Proposition IX.1. *Assume that $\tilde{A}$ is asymptotically stable. Then there exists $\bar{\eta}(k)$ such that, for all $\eta(k) \geq \bar{\eta}(k)$, the matrix $\tilde{A}_{\mathrm{VIR}}(k)$ is asymptotically stable.*

Proof. Since $\tilde{A}$ is asymptotically stable it follows that $\varepsilon \triangleq 1 - \rho(\tilde{A}) \in (0, 1)$, where ρ denotes spectral radius. Next, since $\lim_{\eta(k) \to \infty} \tilde{A}_{\mathrm{VIR}}(k) = \tilde{A}$, it follows that $\lim_{\eta(k) \to \infty} \rho(\tilde{A}_{\mathrm{VIR}}(k)) = \rho(\tilde{A})$. Hence there exists $\bar{\eta}(k)$ such that, for all $\eta(k) > \bar{\eta}(k)$, $\left| \rho(\tilde{A}_{\mathrm{VIR}}(k)) - \rho(\tilde{A}) \right| < \varepsilon$, and thus $\rho(\tilde{A}_{\mathrm{VIR}}(k)) < \varepsilon + \rho(\tilde{A}) = \varepsilon + 1 - \varepsilon = 1$. □

Next, we set $\eta(k) = 0$ in (9.93) and define

$$\bar{\tilde{A}}_{\mathrm{VIR}}(k) \triangleq \tilde{A} - \tilde{B}[\tilde{\mathcal{H}}^{\mathrm{T}} R_1(k) \tilde{\mathcal{H}}]^{-1} \tilde{\mathcal{H}}^{\mathrm{T}} R_1(k) \tilde{E}_1 \tilde{A}^r. \tag{9.94}$$

Proposition IX.2. *Assume that $\tilde{\mathcal{H}}$ has full row rank and $\bar{\tilde{A}}_{\mathrm{VIR}}(k)$ is asymptotically stable. Then there exists $\bar{\eta}(k)$ such that, for all $\eta(k) \in (0, \bar{\eta}(k))$, the matrix $\tilde{A}_{\mathrm{VIR}}(k)$ is asymptotically stable.*

Proof. Since $\bar{\tilde{A}}_{\mathrm{VIR}}(k)$ is asymptotically stable, it follows that $\varepsilon \triangleq 1 - \rho(\bar{\tilde{A}}_{\mathrm{VIR}}(k)) \in (0, 1)$. Next, since $\lim_{\eta(k) \to 0} \tilde{A}_{\mathrm{VIR}}(k) = \bar{\tilde{A}}_{\mathrm{VIR}}(k)$ it follows that $\lim_{\eta(k) \to 0} \rho(\tilde{A}_{\mathrm{VIR}}(k)) = \rho(\bar{\tilde{A}}_{\mathrm{VIR}}(k))$. Hence there exists $\bar{\eta}(k)$ such that, for all $\eta(k) \in (0, \bar{\eta}(k))$, $\left| \rho(\tilde{A}_{\mathrm{VIR}}(k)) - \rho(\bar{\tilde{A}}_{\mathrm{VIR}}(k)) \right| < \varepsilon$, and thus $\rho(\tilde{A}_{\mathrm{VIR}}(k)) < \varepsilon + \rho(\bar{\tilde{A}}_{\mathrm{VIR}}(k)) = \varepsilon + 1 - \varepsilon = 1$. □

It follows from Fact 9.5.5 that, if (A, B, E_1) is minimum phase, $R_1 = I$, and $\tilde{\mathcal{H}} = H_1$ is nonsingular, then $\bar{\tilde{A}}_{\mathrm{VIR}}(k)$ is asymptotically stable. Note that if $\eta(k)$ is nonzero, then the asymptotic stability of $\tilde{A}_{\mathrm{VIR}}(k)$ for all k does not imply that the state $\hat{X}(k)$ in (9.92) is bounded. This motivates the frequency-domain analysis in Section 9.7.

9.6.1 Augmented Retrospectively Optimized Controls

If (A, B, E_1) is square and NMP, then $\tilde{A}_{\text{VIR}}(k)$ with $\tilde{\mathcal{H}} = H_1$, may be unstable for all $\eta(k)$. However, we can use knowledge of the NMP zero locations to filter the retrospectively optimized controls, $\hat{\tilde{U}}(k-1)$ given by (9.32).

Consider the filter

$$f(\mathbf{q}) = \frac{1}{\mathbf{q}^l} d(\mathbf{q}), \tag{9.95}$$

where $d(\mathbf{q}) \in \mathbb{R}[\mathbf{q}]^{l_u \times l_u}$ and

$$d(\mathbf{q}) \stackrel{\triangle}{=} \mathbf{q}^l d_0 + \mathbf{q}^{l-1} d_1 + \ldots + \mathbf{q} d_{l-1}, \tag{9.96}$$

where $d(\mathbf{q})$ and l are chosen such that the transmission zeros of $f(\mathbf{q})$ contain the unstable eigenvalues of $\tilde{A} - \tilde{B}\tilde{\mathcal{H}}^{\dagger}\tilde{E}_1\tilde{A}^r$, for example, the coefficients of $d(\mathbf{q})$ may be Markov parameters.

Next we filter the retrospectively optimized controls $\hat{\tilde{U}}(k-1)$, which yields

$$\hat{\tilde{U}}_{\text{F}}(k-1) = [I_{g\times g} \otimes f(\mathbf{q})]\hat{\tilde{U}}(k-1). \tag{9.97}$$

The filtered retrospectively optimized controls $\hat{\tilde{U}}_{\text{F}}(k-1)$ are used in place of $\hat{\tilde{U}}(k-1)$.

9.7 Frequency-Domain Conditions for Convergence

Propositions IX.1 and IX.2 gave sufficient conditions on $\eta(k)$ for when (9.93) is asymptotically stable. However, if $\eta(k)$ is nonzero, then guaranteeing that $\tilde{A}_{\text{VIR}}(k)$ is stable for all k does not guarantee that $z(k)$ tends to zero.

In this section, we give frequency-domain conditions on $\tilde{\mathcal{H}}$ and $\hat{\tilde{U}}(k-1)$ such that assuming (9.93) is stable for some $\eta(k)$, then $z(k)$ will tend to zero. First

we introduce $G_{\text{FIR}}(\mathbf{q})$, which is an FIR filter whose numerator coefficients are the Markov parameters of $G_{zu}(\mathbf{q})$ that comprise $\tilde{\mathcal{H}}$. The structure of $G_{\text{FIR}}(\mathbf{q})$ depends on the components of $\hat{\tilde{U}}(k-1)$ that are used to update $\theta(k)$ and the structure of $\tilde{\mathcal{H}}$. For example, in the SISO case assume that $\tilde{\mathcal{H}} = [H_3 \ H_2 \ H_1]^{\mathrm{T}}$ and $g = q_g = 3$, which results in $\hat{\tilde{U}}(k-1) = \hat{u}(k-q_g)$. Then

$$G_{\text{FIR}}(\mathbf{q}) = \frac{H_1\mathbf{q}^2 + H_2\mathbf{q} + H_3}{\mathbf{q}^3}. \tag{9.98}$$

Note that $G_{\text{FIR}}(\mathbf{q})$ is only used an analysis too, $G_{\text{FIR}}(\mathbf{q})$ is not used in the actual implementation of the algorithm.

Furthermore, let the external signal $w(k)$ be a sinusoid whose frequency is Θ. For this analysis we assume that $\beta(k) = 1$, only when the state of the system reaches harmonic steady state, and $\beta(k) = 0$ otherwise. Specifically, we update the controller coefficients only when the state of the system reaches harmonic steady state and let

$$\nu = \beta(0) + \cdots + \beta(k), \tag{9.99}$$

where ν is the number of controller updates. In practice, however, we update the controller coefficients at every step k.

Proposition IX.3. *Let $\hat{\tilde{U}}(k-1)$ be given by* (9.32)*. Assume that Assumption 9.5.1 is satisfied, and that $R_2(k)$ is chosen such that $R_2(k) \to 0$, as $\nu \to \infty$, which from Fact 9.5.2 implies that $\hat{z}(k) \to 0$, as $\nu \to \infty$ and let $l_u = l_z = 1$. Next, assume that the frequency Θ does not coincide with any zeros of G_{FIR}. Furthermore, assume that* (9.92) *reaches harmonic steady state for all ν, that then if*

$$\left| 1 - \frac{G_{zu}(e^{\jmath\Theta})}{G_{\text{FIR}}(e^{\jmath\Theta})} \right| < 1, \tag{9.100}$$

then $z(k) \to 0$ as $k \to \infty$.

In harmonic steady state we have

$$z_\nu = G_{zw}(e^{\jmath\Theta})w + G_{zu}(e^{\jmath\Theta})\hat{u}_\nu + G_{zu}(e^{\jmath\Theta})g_\nu, \tag{9.101}$$

where z_ν,w, g_ν are phasors, and $g_\nu \stackrel{\triangle}{=} u_\nu - \hat{u}_\nu$. In view of the assumption that $u(k)-\hat{u}(k) \to 0$ as $k \to \infty$, we assume that g_ν is negligible and omitted for simplicity.

Next, the retrospective cost in harmonic steady state is,

$$\hat{z}_\nu \stackrel{\triangle}{=} z_{\nu-1} - G_{\mathrm{FIR}}(e^{\jmath\Theta})\hat{u}_{\nu-1} + G_{\mathrm{FIR}}(e^{\jmath\Theta})\hat{u}_\nu, \tag{9.102}$$

$$\hat{z}_\nu = G_{zw}(e^{\jmath\Theta})w + G_{zu}(e^{\jmath\Theta})\hat{u}_{\nu-1} - G_{\mathrm{FIR}}(e^{\jmath\Theta})\hat{u}_{\nu-1} + G_{\mathrm{FIR}}(e^{\jmath\Theta})\hat{u}_\nu, \tag{9.103}$$

$$\hat{z}_\nu = G_{zw}(e^{\jmath\Theta})w + [G_{zu}(e^{\jmath\Theta}) - G_{\mathrm{FIR}}(e^{\jmath\Theta})]\hat{u}_{\nu-1} + G_{\mathrm{FIR}}(e^{\jmath\Theta})\hat{u}_\nu. \tag{9.104}$$

Solving (9.104) for $\hat{u}_\nu$ yields,

$$\hat{u}_\nu = G_{\mathrm{FIR}}^{-1}(e^{\jmath\Theta})\left[\hat{z}_\nu - G_{zw}(e^{\jmath\Theta})w - [G_{zu}(e^{\jmath\Theta}) - G_{\mathrm{FIR}}(e^{\jmath\Theta})]\hat{u}_{\nu-1}\right]. \tag{9.105}$$

Substituting (9.105) into (9.101) yields,

$$z_\nu = [1 - G_{zu}(e^{\jmath\Theta})G_{\mathrm{FIR}}^{-1}(e^{\jmath\Theta})][G_{zw}(e^{\jmath\Theta})w + G_{zu}(e^{\jmath\Theta})\hat{u}_{\nu-1}] + G_{zu}(e^{\jmath\Theta})G_{\mathrm{FIR}}^{-1}(e^{\jmath\Theta})\hat{z}_\nu. \tag{9.106}$$

Using this process we write z_ν in terms of $\hat{u}_0$ as

$$\begin{aligned} z_\nu = {} & [1 - G_{zu}(e^{\jmath\Theta})G_{\mathrm{FIR}}^{-1}(e^{\jmath\Theta})]^\nu[G_{zw}(e^{\jmath\Theta})w + G_{zu}(e^{\jmath\Theta})\hat{u}_0] \\ & + \sum_{i=0}^{\nu}[1 - G_{zu}(e^{\jmath\Theta})G_{\mathrm{FIR}}^{-1}(e^{\jmath\Theta})]^i[G_{zu}(e^{\jmath\Theta})G_{\mathrm{FIR}}^{-1}(e^{\jmath\Theta})]\hat{z}_{\nu-i}. \end{aligned} \tag{9.107}$$

It follows from (9.107) that

$$|z_\nu| \leq \left|1 - G_{zu}(e^{\jmath\Theta})G_{\text{FIR}}^{-1}(e^{\jmath\Theta})\right|^\nu \left|G_{zw}(e^{\jmath\Theta})w + G_{zu}(e^{\jmath\Theta})\hat{u}_0\right| + \left|\sum_{i=0}^{\nu}[1 - G_{zu}(e^{\jmath\Theta})G_{\text{FIR}}^{-1}(e^{\jmath\Theta})]^i[G_{zu}(e^{\jmath\Theta})G_{\text{FIR}}^{-1}(e^{\jmath\Theta})]\hat{z}_{\nu-i}\right|. \quad (9.108)$$

Therefore, since $\left|1 - \frac{G_{zu}(e^{\jmath\Theta})}{G_{\text{FIR}}(e^{\jmath\Theta})}\right| < 1$, it follows that $\left|1 - \frac{G_{zu}(e^{\jmath\Theta})}{G_{\text{FIR}}(e^{\jmath\Theta})}\right|^\nu \to 0$ as $\nu \to \infty$, then $|z_\nu| \to 0$ as $\nu \to \infty$.

Condition (9.100) has a simple geometric interpretation, namely, $G_{\text{FIR}}(e^{\jmath\Theta})$ must lie in a half plane that contains $G_{zu}(e^{\jmath\Theta})$ and whose boundary is perpendicular to $|G_{zu}(e^{\jmath\Theta})|$ and passes through $\frac{1}{2}|G_{zu}(e^{\jmath\Theta})|$. Figure 9.2 illustrates the region of admissible $G_{\text{FIR}}(e^{\jmath\Theta})$ for a given $|G_{zu}(e^{\jmath\Theta})|$ and frequency.

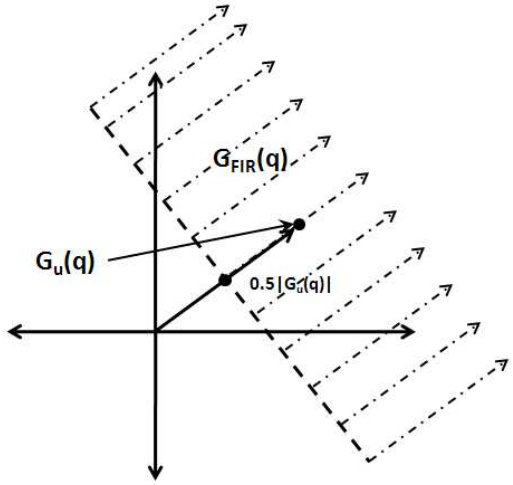


Figure 9.2: The shaded region on the complex plane illustrates the region of admissible $G_{\text{FIR}}(e^{\jmath\Theta})$ for a given $|G_{zu}(e^{\jmath\Theta})|$ and frequency Θ as determined by (9.100). The admissible region is a half plane.

From Proposition IX.3, we note that if Θ coincides with a zero of G_{FIR}, then the system remains at its open-loop performance value. Furthermore, this analysis can be done for multiple frequencies.

9.8 Numerical Examples

For all numerical examples in this chapter we use the recursive least squares update (9.49) and (9.50). Furthermore, we consider only the disturbance rejection problem, where $D_1 \neq 0$, $D_2 = 0$, and $E_0 = 0$. For the examples in this chapter we choose $\eta(k) = \frac{\eta_0}{k}$, where η_0 is a nonnegative number.

9.8.1 SISO Examples

Example 9.8.1. (SISO MP) Consider the system

$$A = \begin{bmatrix} 1.7 & -1.2 & 0.7 \\ 1 & 0 & 0 \\ 0 & 0.5 & 0 \end{bmatrix}, B = \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix}, \tag{9.109}$$

$$D_1 = \begin{bmatrix} 0.9794 \\ -0.2656 \\ -0.5484 \end{bmatrix}, C = E_1 = \begin{bmatrix} 0.5 \\ -0.65 \\ 0.4 \end{bmatrix}^{\mathrm{T}}, \tag{9.110}$$

which is minimum-phase and stable. The goal is to reject the disturbance $w(k) = \sin(\frac{\pi}{5}k)$. We choose $\tilde{\mathcal{H}} = H_1$, $n_c = 5$, $\gamma = 1$ and $\eta_0 = 0$. Figure 9.3 shows the adaptive controller in closed-loop with the plant. The performance is reduced to zero with knowledge of just one Markov parameter. Furthermore, we are able to choose $\eta_0 = 0$, since the plant is minimum phase.

Example 9.8.2. (SISO NMP) Consider the system (9.109) and (9.110), where C and E_1 are replaced by

$$C = E_1 = \begin{bmatrix} 0.5 & -1.25 & 1 \end{bmatrix}, \tag{9.111}$$

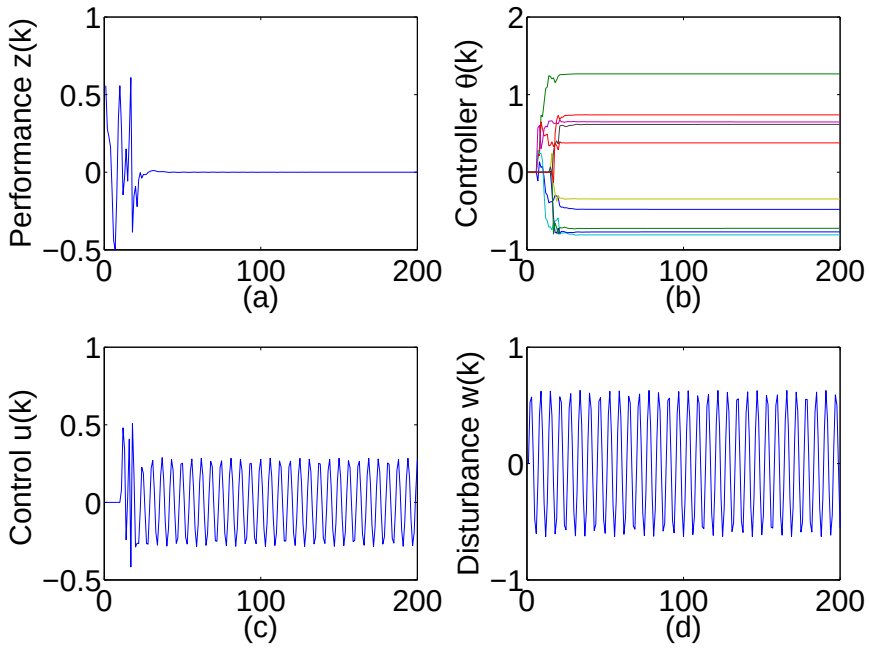


Figure 9.3: For this example, the plant is SISO and minimum phase. We choose $\tilde{\mathcal{H}} = H_1$ and $\eta_0 = 0$. (a) shows the performance $z(k)$, (b) shows the controller parameters $\theta(k)$, (c) shows the control signal $u(k)$, and (d) shows the disturbance $w(k)$.

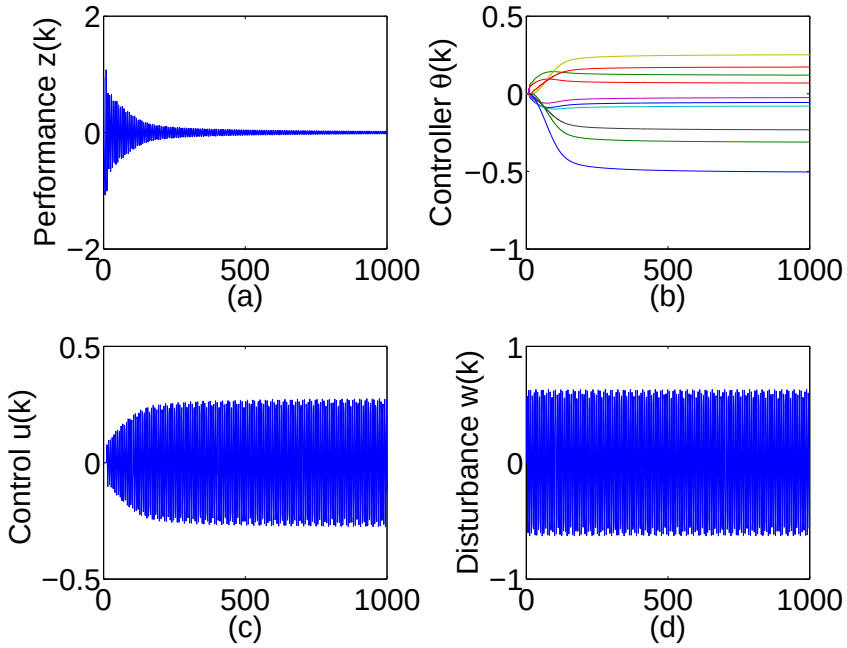


Figure 9.4: For this example, the plant is SISO and nonminimum phase. We choose $\tilde{\mathcal{H}} = H_1 = 1$, and $\eta_0 = 75$. (a) shows the performance $z(k)$, (b) shows the controller parameters $\theta(k)$, (c) shows the control signal $u(k)$, and (d) shows the disturbance $w(k)$.

which makes the system nonminimum phase, with a zero at 2. The goal is to reject the disturbance $w(k) = \sin(\frac{\pi}{5}k)$. We choose $\tilde{\mathcal{H}} = H_1 = 1$, $n_c = 5$, $\eta_0 = 75$, and $\gamma = 1$. Figure 9.4 shows the adaptive filter in closed loop with the nonminimum-phase system. Note that the controller does not have any knowledge of the nonminimum-phase zero.

Example 9.8.3. (SISO NMP) Next, we consider the same plant and disturbance as Example 9.8.2. Furthermore we choose the controller parameters as in 9.8.2. However, in this case we assume that we have knowledge of only the 4^{th} Markov parameter, so that $\tilde{\mathcal{H}} = H_4 = -1.3420$. Figure 9.5 shows the closed-loop performance using knowledge of just the 4^{th} Markov parameter. There is no discernible performance gain or degradation when using H_4 as opposed to H_1.

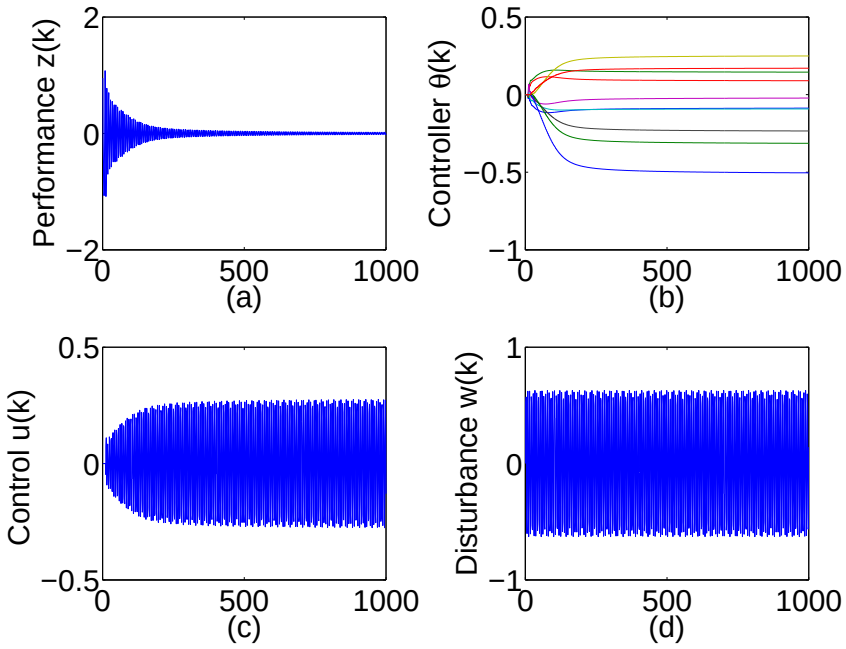


Figure 9.5: For this example, the plant is SISO and nonminimum phase. We choose $\tilde{\mathcal{H}} = -1.3420$ and $\eta_0 = 75$. (a) shows the performance $z(k)$, (b) shows the controller parameters $\theta(k)$, (c) shows the control signal $u(k)$, and (d) shows the disturbance $w(k)$.

9.8.2 MIMO Examples

Example 9.8.4. (2×2 NMP) Consider the asymptotically stable system

$$A = \begin{bmatrix} -0.3 & 0.3 & 0.56 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.6 & -0.1 & 0.4 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & 0 \end{bmatrix}, \tag{9.112}$$

$$B = \begin{bmatrix} 2 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 4 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \; D_1 = \begin{bmatrix} 0.9794 \\ -0.2656 \\ -0.5484 \\ 0.0963 \\ -1.3807 \\ -0.7284 \end{bmatrix}, \tag{9.113}$$

$$C = E_1 = \begin{bmatrix} 0 & 0.5 & 0 & 0.25 & -0.25 & -1 \\ 0 & 0 & 1 & 0 & 0.25 & -1 \end{bmatrix}, \tag{9.114}$$

which has a nonminimum-phase transmission zero at 2. The goal is to reject the disturbance $w(k) = \sin(\frac{\pi}{5}k)$. We choose $\tilde{\mathcal{H}} = [H_3^{\mathrm{T}} \quad H_2^{\mathrm{T}} \quad H_1^{\mathrm{T}}]^{\mathrm{T}}$, $n_c = 5$, $\eta_0 = 75$, and $\gamma = 1$.

Figure 9.6 shows closed-loop performance with knowledge of the 2^{nd} Markov parameter. Note in this case that the first nonzero Markov parameter $E_1 B = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$

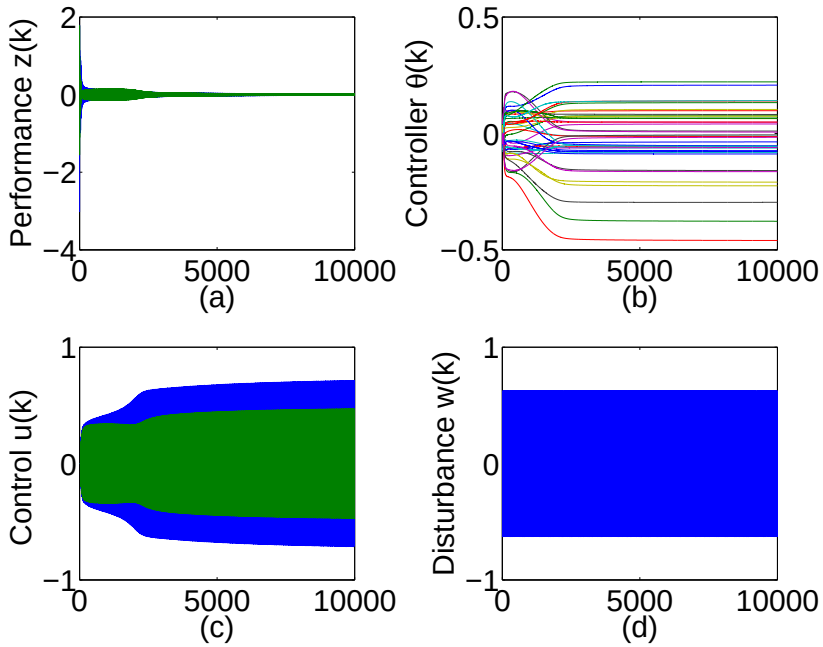


Figure 9.6: For this example, the plant is 2×2 MIMO, and is nonminimum phase. We choose $\tilde{\mathcal{H}} = [H_3^{\mathrm{T}} \quad H_2^{\mathrm{T}} \quad H_1^{\mathrm{T}}]^{\mathrm{T}}$ and $\eta_0 = 75$. (a) shows the performance $z(k)$, (b) shows the controller parameters $\theta(k)$, (c) shows the control signal $u(k)$, and (d) shows the disturbance $w(k)$.

is not left invertible, and thus another Markov parameter must be used in $\tilde{\mathcal{H}}$, specifically, we used $H_2 = E_1AB$.

Example 9.8.5. (1×2 NMP) Consider the asymptotically stable system

$$A = \begin{bmatrix} 0 & 0.4 & 0 & 0 \\ 0.5 & 0 & 0 & 0 \\ 0 & 00 & -0.1 & 0.4 \\ 0 & 0 & 0.5 & 0 \end{bmatrix}, B = \begin{bmatrix} 2 & 0 \\ 0 & 0 \\ 0 & 2 \\ 0 & 0 \end{bmatrix}, \tag{9.115}$$

$$D_1 = \begin{bmatrix} 0.9794 \\ -0.2656 \\ -0.5484 \\ 0.0963 \end{bmatrix}, C = E_1 = \begin{bmatrix} 0.5 \\ -1.5 \\ 0.5 \\ -1.5 \end{bmatrix}^{\mathrm{T}}, \tag{9.116}$$

which has a nonminimum-phase transmission zero at 1.5. The goal is to reject the disturbance $w(k) = \sin(\frac{\pi}{5}k)$. We choose $\tilde{\mathcal{H}} = [H_2^{\mathrm{T}}\ H_1^{\mathrm{T}}]^{\mathrm{T}} = \begin{bmatrix} -1.5 & -1.6 \\ 1 & 1 \end{bmatrix}$, $n_c = 5$, $\gamma = 1$, and $\eta_0 = 0$. Figure 9.7 shows the closed-loop performance with knowledge of the 1st and 2nd Markov parameters. In this case, we note that any single Markov parameter is not left invertible. Therefore, wide systems require more than one Markov parameter to ensure that $\tilde{\mathcal{H}}$ is invertible.

9.9 Conclusions

In this chapter we extended the RCAC adaptive control algorithm and investigated its ability to adaptively control systems without knowledge of the nonminimum-phase zeros, if any. Conditions for the stability of the error system were examined in the unregularized and regularized versions of the algorithm. Furthermore, the algorithm was demonstrated on SISO and MIMO examples, starting with a minimum-phase

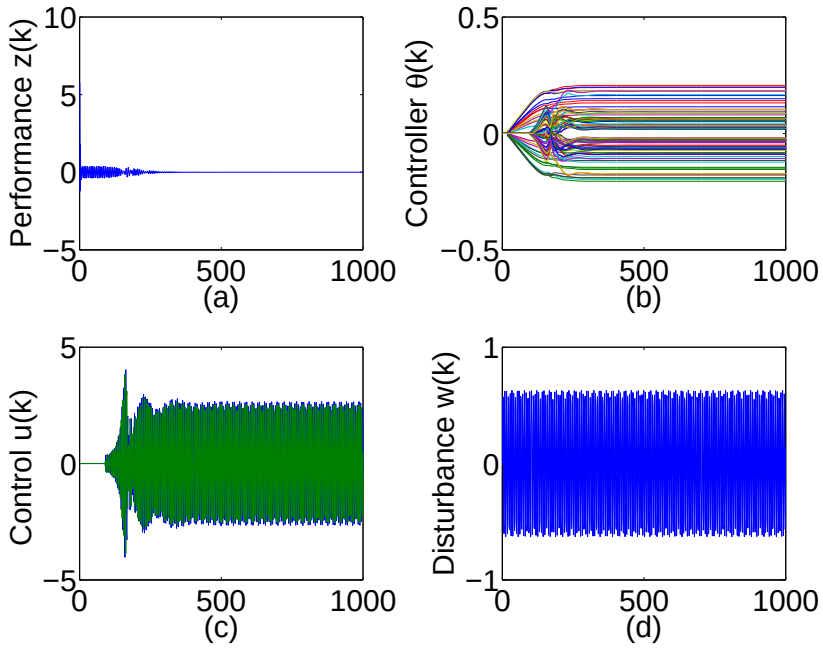


Figure 9.7: For this example, the plant is nonminimum phase 1x2 MIMO. We choose $\tilde{\mathcal{H}} = [H_2^{\mathrm{T}} \; H_1^{\mathrm{T}}]^{\mathrm{T}}$ and $\eta_0 = 0$. (a) shows the performance $z(k)$, (b) shows the controller parameters $\theta(k)$, (c) shows the control signal $u(k)$, and (d) shows the disturbance $w(k)$.

system, and working up to the nonminimum-phase case, including non-square systems. Furthermore, we demonstrated that the algorithm can utilize various combinations of Markov parameters, for example, the 2^{nd} and 6^{th}, Markov parameters. In all cases, the number of Markov parameters that are used is not sufficient to determine the nonminimum-phase zeros of the system. Consequently, these examples demonstrate the ability to control MIMO nonminimum-phase systems with unknown nonminimum-phase zeros. In this chapter we assumed that the Markov parameters that are used in RCAC are exactly known. However, in practice, these parameters are uncertain due to modeling errors.

CHAPTER X

Application of Adaptive Control to a Seeker-Guided 2D Missile with Unmodeled Aerodynamics

10.1 Introduction

Many difficulties are encountered when applying adaptive control to a tail-controlled missile system. One of the issues encountered is nonminimum-phase zeros in the linearizations about trim setpoints; these zeros are are due to tail actuation and sensor suites located ahead of the center of gravity. Nonminimum-phase zeros might also appear in the digitized system once the continuous-time plant is sampled during flight. Therefore, standard techniques for minimum-phase systems [123, 124] are not well-suited for missile control. Additionally, actuator delays, aerodynamic effects, and structural flexibility introduce significant complexity to the problem. Furthermore, as described in [33], tail-controlled missiles require fast response times in order to achieve acceleration commands generated by guidance control laws.

In previous work [33], cumulative retrospective cost adaptive control (RCAC) [58, 59], was implemented on a 2D missile as an outer-loop with the three-loop-autopilot, found in [125], as the inner loop. The cumulative retrospective cost adaptive control algorithm requires knowledge of nonminimum-phase zero locations. Since

the linearized missile dynamics in closed loop with the three-loop autopilot has a nonminimum-phase zero, this information was scheduled in order to implement the adaptive controller.

The present work differs from the work in [33] in several ways. First, the inner-loop is eliminated, specifically, the missile is controlled only by the retrospective cost adaptive control (RCAC) algorithm. Secondly, we do not schedule the nonminimum-phase zero location. Recent extensions to the RCAC algorithm eliminate the requirement for knowledge of the nonminumum-phase zero locations, as described in [122, 37], specifically, the algorithm development requires a limited number of Markov parameters. Furthermore, in the SISO case, the algorithm has been shown to be robust to the magnitude of the Markov parameter, meaning that sign information usually suffices. Therefore, we do not schedule the estimate of the required Markov parameter; a single value is used for all trim conditions. It should be noted that this parameter does not depend on any aerodynamic modeling information specific to the 2D missile. Details about the required modeling information are discussed in [37]. Furthermore, to reduce oscillation of the missile body during the flight, we include rate saturation in the retrospective cost function.

In the present work, we design a set of scenarios and compare the performance of the retrospective cost adaptive controller to the three-loop-autopilot performance. We examine the nominal scenario, in which, the autopilot has exact knowledge of the missile's aerodynamic coefficients and dynamics. The adaptive controller tuning parameters are tuned to yield the best performance under the nominal scenario. Next, we perturb the aerodynamic coefficients using an affine linear transformation (shift and scale factor). In these scenarios, the autopilot remains tuned for the nominal scenario; the adaptive controller tuning parameters are also unchanged. Finally, we examine the effect of noisy sensors on the performance of both controllers, specifically, noise is added to the body angle sensor. In each scenario presented, a Monte Carlo

simulation is used to determine the median miss distance, where the variation in simulations is due to the changing initial conditions of the target.

This Chapter is organized as follows. In Section 10.2, we give a general problem formulation and define the tactical objective. In Section 10.3, we present the nonlinear missile dynamics and actuator dynamics. In Section 10.4, a brief construction of the RCAC algorithm applied to the 2D missile problem is outlined. The generalized algorithm setup is given in Chapter IX. In Sections 10.5 to Section 10.8, we present the simulation setups and results of the Monte Carlo simulations. Finally, in Section 10.9 we provide conclusions.

Nomenclature

m	=	Missile mass
I_{yy}	=	Missile Inertia
g	=	Acceleration due to gravity
X,Z	=	Inertial position in the X,Z plane
CG	=	center of gravity
U	=	Inertial velocity component along the body frame x-axis of the (CG)
W	=	Inertial velocity component along the body frame z-axis of the CG
V_m	=	Magnitude of missile velocity
M	=	Mach number
α	=	Angle of attack
θ	=	Pitch angle
q	=	Body angle
ρ	=	Air density
S_{ref}	=	Reference area
d_{ref}	=	Reference length
C_x	=	Aerodynamic force coefficient along the body frame x-axis
C_z	=	Aerodynamic force coefficient along the body frame z-axis
C_m	=	Aerodynamic moment coefficient along the body y-axis at the CG
$\bar{q}$	=	Dynamic pressure
δ_p	=	Tail fin angle
T	=	Thrust along the body frame x-axis

10.2 Problem Formulation

Consider the target dynamics

$$\dot{v}(t) = f_t(v(t)), \tag{10.1}$$

$$v_p(t) = [I_2 \ 0_{2\times 2}]v(t), \tag{10.2}$$

where $v(t) \in \mathbb{R}^4$ is the inertial position and velocity of the target at time t, and $v_p(t) \in \mathbb{R}^2$ is the inertial position of the target. The boundary conditions are

$$v(0) \neq 0, \tag{10.3}$$

$$v(T_t) \neq 0, \tag{10.4}$$

where T_t is the final time of the target flight.

Next consider the missile dynamics

$$\dot{x}(t) = f_m(x(t)) + g(u(t)), \tag{10.5}$$

$$x_p(t) = [I_2 \ 0_{2\times 2}]x(t), \tag{10.6}$$

where $x(t) \in \mathbb{R}^4$ is the inertial position and velocity of the missile at time t and $x_p(t) \in \mathbb{R}^2$ is the inertial position of the missile. The goal is to determine the control $u(t)$, that minimizes $|v_p(t) - x_p(t)|$ at some time $t \leq T_t$.

10.3 Nonlinear Missile Model

Consider the nonlinear three-degree-of-freedom model

$$\dot{X} = U\cos\theta + W\sin\theta, \tag{10.7}$$

$$\dot{Z} = -U\sin\theta + W\cos\theta, \tag{10.8}$$

$$\dot{\theta} = q, \tag{10.9}$$

$$\dot{U} = \frac{1}{m}T + \frac{\bar{q}S_{\text{ref}}}{m}C_x(\alpha, M) - qW - g\sin\theta, \tag{10.10}$$

$$\dot{W} = \frac{\bar{q}S_{\text{ref}}}{m}C_z(\alpha, M, \delta_p) + qU + g\cos\theta, \tag{10.11}$$

$$\dot{q} = \frac{\bar{q}S_{\text{ref}}d_{\text{ref}}}{I_{yy}}C_m(\alpha, M, \delta_p, q), \tag{10.12}$$

where the dynamic pressure is given by $\bar{q} = \frac{1}{2}\rho V_m^2$ and angle of attack is defined as $\alpha = \text{atan}(W/U)$. Measurements of the body rate $\dot{q}$ and lateral missile acceleration are assumed to take place at the inertial measurement unit (IMU) location. The lateral acceleration at the IMU is related to that measured at the center of gravity through the equation

$$A_{z,\text{IMU}} = A_{z,\text{CG}} - \dot{q}x_{\text{IMU}}, \tag{10.13}$$

where x_{IMU} is the distance from the CG to IMU. Here we assume that the missile IMU is forward of the CG location. The nonlinear model in equations (5)-(11) is linearized about trim points in the region of operation for the purpose of autopilot controller design. Additionally, the second-order fin actuator model

$$\delta_p(s) = \frac{\omega_a^2}{s^2 + 2\lambda_a\omega_a s + \omega_a^2}u(s), \tag{10.14}$$

is used, where $u(s)$ is the fin actuator command, ω_a and λ_a represent the natural frequency and damping ratio, respectively, of the actuator dynamics.

10.3.1 Three-Loop Autopilot

We compare the adaptive controller with the three-loop auto-pilot [125], which is given by

$$u(s) = K_q q(s) + \frac{1}{s}(K_\theta q(s) + K_I[K_{ss}A_{z,\text{cmd}} - A_{z,\text{IMU}}]), \tag{10.15}$$

where K_q, K_θ, K_I, K_{ss} are control gains, and $A_{z,\text{cmd}}$ is the lateral acceleration command for the IMU location provided by the missile's guidance system. A proportional navigation guidance law generates the acceleration. The control gains K_q, K_θ, K_I, and K_{ss} are determined by modeling and analysis, and scheduled based on angle of attack and Mach number, therefore the autopilot is actually many separate controls, which are switched depending on trim condition. Note that in practice this continuous-time controller is digitized using sample and hold operations.

10.4 Overview of Application of Retrospective Cost to the 2D Missile Problem

In this Section we given a brief overview of the retrospective cost adaptive controller for the 2D nonlinear missile model. Note that RCAC does not require digitization since, it is developed in discrete time.

10.4.1 Controller Construction

The control $u(k)$ is given by the strictly proper time-series controller of order n_c given by

$$u(k) = \sum_{i=1}^{n_\text{c}} M_i(k)u(k-i) + \sum_{i=1}^{n_\text{c}} N_i(k)y(k-i), \tag{10.16}$$

where, for all $i = 1, \ldots, n_{\mathrm{c}}$, $M_i(k) \in \mathbb{R}^{l_u \times l_u}$ and $N_i(k) \in \mathbb{R}^{l_u \times l_y}$. The control (10.16) can be expressed as

$$u(k) = \theta(k)\phi(k-1), \tag{10.17}$$

where

$$\theta(k) \triangleq [M_1(k) \ \cdots \ M_{n_{\mathrm{c}}}(k) \ N_1(k) \ \cdots \ N_{n_{\mathrm{c}}}(k)] \in \mathbb{R}^{l_u \times n_{\mathrm{c}}(l_u + l_y)} \tag{10.18}$$

and

$$\phi(k-1) \triangleq \begin{bmatrix} u(k-1) \\ \vdots \\ u(k-n_{\mathrm{c}}) \\ y(k-1) \\ \vdots \\ y(k-n_{\mathrm{c}}) \end{bmatrix} \in \mathbb{R}^{n_{\mathrm{c}}(l_u + l_y)}, \tag{10.19}$$

where $y(k) = z(k)$, for all k.

10.4.2 Recursive Least Squares Update of $\theta(k)$

We define the cumulative cost function

$$\begin{aligned} J_{\mathrm{R}}(\theta(k)) \triangleq & \sum_{i=q_g+1}^{k} \lambda^{k-i} \| \phi^{\mathrm{T}}(i - q_g - 1)\theta^{\mathrm{T}}(k) - \hat{u}^{\mathrm{T}}(i - q_g) \|^2 \\ & + \lambda^k (\theta(k) - \theta(0)) P^{-1}(0) (\theta(k) - \theta(0))^{\mathrm{T}}, \end{aligned} \tag{10.20}$$

where $\| \cdot \|$ is the Euclidean norm and, for some $\varepsilon \in (0, 1)$, $\lambda(k) \in (\varepsilon, 1]$ is the forgetting factor and $\hat{u}(k)$ are the retrospectively optimized controls discussed in Section 10.4.3.

Minimizing (10.20) yields

$$\begin{aligned}\theta^{\mathrm{T}}(k) \triangleq\ & \theta^{\mathrm{T}}(k-1) + \beta(k)P(k-1)\phi(k-q_g-1) \\ & \cdot [\phi^{\mathrm{T}}(k-q_g-1)P(k-1)\phi(k-q_g-1) + \lambda(k)]^{-1} \\ & \cdot [\theta(k-1)\phi(k-q_g-1) - \hat{u}(k-q_g)]^{\mathrm{T}}, \end{aligned} \tag{10.21}$$

where $\beta(k)$ is either 0 or 1. When $\beta(k) = 1$, the controller is allowed to adapt, whereas, when $\beta(k) = 0$, the adaptation is off. The error covariance is updated by

$$\begin{aligned}P(k) \triangleq\ & (1-\beta(k))P(k-1) + \beta(k)\lambda^{-1}(k)P(k-1) - \beta(k)\lambda^{-1}(k)P(k-1) \\ & \cdot \phi(k-q_g-1)[\phi^{\mathrm{T}}(k-q_g-1)P(k-1)\phi(k-q_g-1) + \lambda(k)]^{-1} \\ & \cdot \phi^{\mathrm{T}}(k-q_g-1)P(k-1). \end{aligned} \tag{10.22}$$

We initialize the error covariance matrix as $P(0) = \gamma I$, where $\gamma > 0$.

10.4.3 Retrospectively Optimized Controls

We solve for $\hat{\bar{U}}(k)$, which has $\hat{u}(k-q_g)$ as a component, by minimizing the *retrospective cost function*

$$\begin{aligned}J(\hat{\bar{U}}(k-1), k) \triangleq\ & \hat{Z}^{\mathrm{T}}(k)R_1(k)\hat{Z}(k) + \eta(k)\hat{\bar{U}}^{\mathrm{T}}(k-1)R_2(k)\hat{\bar{U}}(k-1) \\ & + \mu(k)[\hat{\bar{U}}(k-1) + \hat{\bar{U}}(k-2)]^{\mathrm{T}}[\hat{\bar{U}}(k-1) + \hat{\bar{U}}(k-2)], \end{aligned} \tag{10.23}$$

where $R_1(k) \in \mathbb{R}^{l_z s \times l_z s}$ is a positive-definite performance weighting, $R_2(k) \in \mathbb{R}^{gl_u \times gl_u}$ is a positive-definite control weighting, $\eta(k) \geq 0$ is a regularization weighting, and $\mu(k) \geq 0$ is a control rate penalty. The goal is to determine retrospective controls $\hat{\bar{U}}(k-1)$ that would have provided better performance than the controls $U(k)$ that were applied to the system. The retrospectively optimized control values $\hat{\bar{U}}(k-1)$ are

then used to update the controller. Note that $\hat{Z}(k)$, $Z(k)$, $\tilde{\mathcal{H}}$, $\tilde{U}(k-1)$, and $\hat{\tilde{U}}(k-1)$ are developed in Chapter IX.

Furthermore, $Z(k)$ is formed by stacking past performance variables $z(k)$, which for the 2D missile problem is defined as

$$z(k) \triangleq M[A_{z,\text{cmd}} - A_{z,\text{IMU}}]. \tag{10.24}$$

Next, substituting (9.26) into (10.23) yields

$$J(\hat{\tilde{U}}(k-1),k) = \hat{\tilde{U}}(k-1)^{\mathrm{T}}\mathcal{A}(k)\hat{\tilde{U}}(k-1) + \hat{\tilde{U}}^{\mathrm{T}}(k-1)\mathcal{B}^{\mathrm{T}}(k) + \mathcal{C}(k), \tag{10.25}$$

where

$$\mathcal{A}(k) \triangleq \tilde{\mathcal{H}}^{\mathrm{T}}R_1(k)\tilde{\mathcal{H}} + \eta(k)R_2(k) + \mu I, \tag{10.26}$$

$$\mathcal{B}(k) \triangleq 2\tilde{\mathcal{H}}^{\mathrm{T}}R_1(k)[Z(k) - \tilde{\mathcal{H}}\tilde{U}(k-1)] + 2\mu\hat{\tilde{U}}(k-2), \tag{10.27}$$

$$\begin{aligned}\mathcal{C}(k) \triangleq{}& Z^{\mathrm{T}}(k)R_1(k)Z(k) - 2Z^{\mathrm{T}}(k)R_1(k)\tilde{\mathcal{H}}\tilde{U}(k-1) + \tilde{U}^{\mathrm{T}}(k-1)\tilde{\mathcal{H}}^{\mathrm{T}}R_1(k)\tilde{\mathcal{H}}\tilde{U}(k-1)\\ &+ \hat{\tilde{U}}^{\mathrm{T}}(k-2)\hat{\tilde{U}}(k-2).\end{aligned} \tag{10.28}$$

If either $\tilde{\mathcal{H}}$ has full column rank or $\eta(k) > 0$, then $\mathcal{A}(k)$ is positive definite. In this case, $J(\hat{\tilde{U}}(k-1),k)$ has the unique global minimizer

$$\hat{\tilde{U}}(k-1) = -\frac{1}{2}\mathcal{A}^{-1}(k)\mathcal{B}(k), \tag{10.29}$$

which is the optimized retrospective control.

10.5 Case 1 - Nominal Conditions

For each scenario, a Monte Carlo simulation of 50 samples is run. The initial conditions of the target are varied for each of the runs. Table 10.4 gives the mean,

Variable	Mean	Distribution	Standard Deviation
X - position	4500 [m]	Normal	500 [m]
Z - position	3248 [m]	Normal	100 [m]
Magnitude of target velocity	328 [m/s]	Normal	50 [m/s]
Body angle of target	π [rad]	Normal	0.1 [rad]

Table 10.1: For each scenario the Monte Carlo simulation uses the initial conditions and distributions given in this table.

Scenario	Autopilot Median Miss [m]	Adaptive Control Median Miss [m]
Nominal	0.12	0.19

Table 10.2: Nominal median miss distance for the autopilot and adaptively controlled missile.

distribution function, and standard deviation for each of the initial condition variables.

For all scenarios, the retrospective cost adaptive controller parameters are chosen as $n_c = 5$, $\mathcal{H} = \hat{H}_1 = 1$, $\theta(0) = 0$, $\eta(k) = 0.01z^2(k)$, $\mu = 500$, $\gamma = 1 \times 10^{-5}$, $R_1 = 1$, and $R_2 = 1$. Note that $\hat{H}_1$ is an estimate of the first nonzero Markov parameter H_1 of the 2D missile. Although H_1 is a function of the trim condition, we choose $\hat{H}_1 = 1$, for all trim conditions. Furthermore, the adaptive controller is initialized at zero for every run of the Monte Carlo simulation. No baseline controller is used in conjunction with the adaptive controller.

For the nominal scenario the Monte Carlo simulation yields a median miss distance for the autopilot of 0.12 meters, and for the adaptive control the median miss distance is 0.19 meters.

10.5.1 Mitigation of Oscillatory Trajectories for Nominal Aerodynamics

For the nominal scenario we examine a single run of the Monte Carlo simulation to evaluate the presence of oscillatory trajectories, which are undesirable. Figure 10.1 shows the trajectory of the missile and target for a single run. The normal acceleration and fin control are smooth, and the resulting miss distance is less than 1 meter.

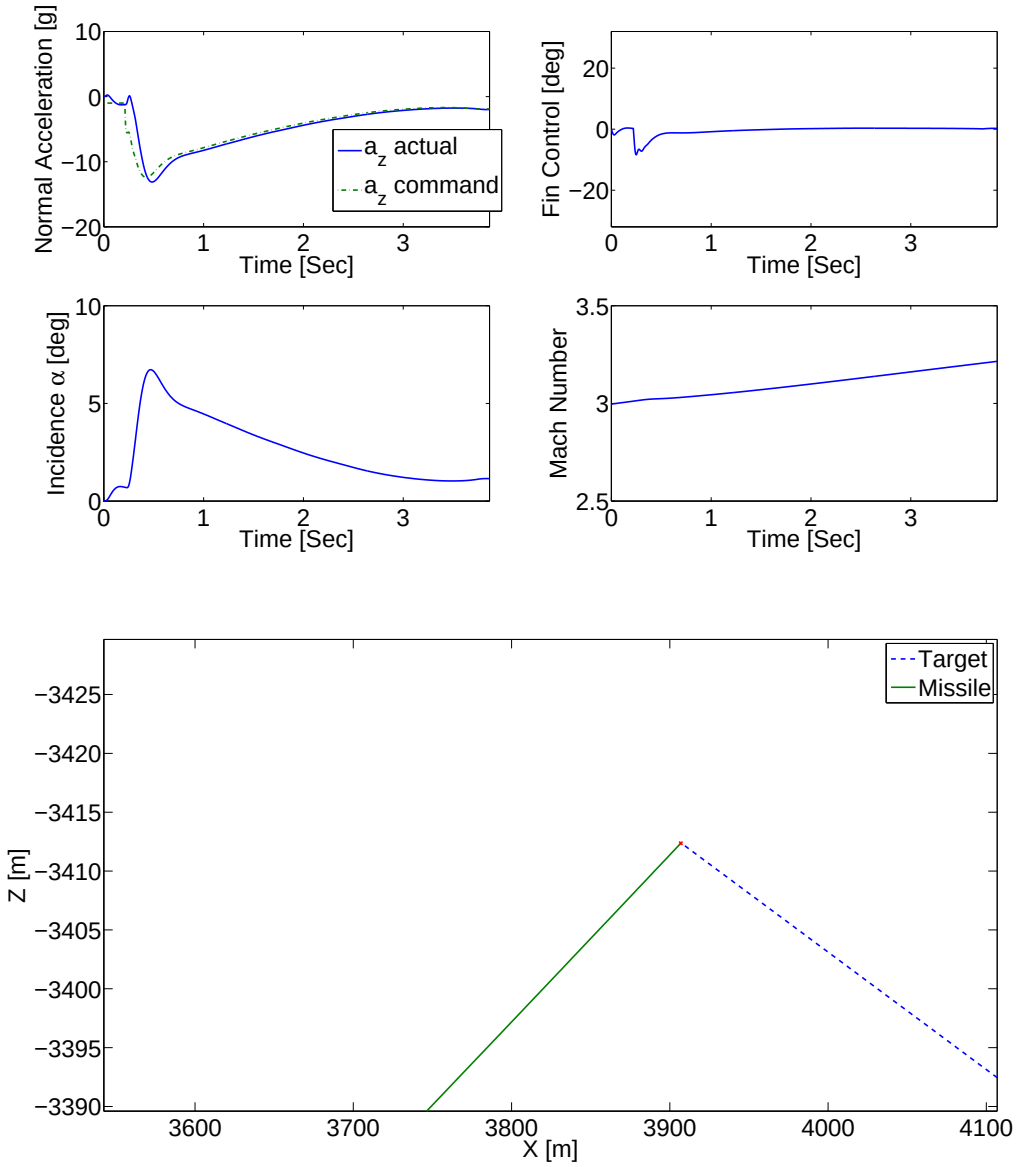


Figure 10.1: Autopilot controlled missile with nominal aerodynamic coefficients.

Next, for this single example, apply RCAC with $\mu(k) = 0$ for all $k \geq 0$. Figure 10.2 shows the trajectory of the missile and target for a single run. In this case, the normal acceleration and fin control contain high frequency content, which causes the missile to oscillate about its center of gravity resulting in a miss distance greater than 10 meters. Figure 10.3 shows the adaptive controller gains $\theta(k)$. Note the sharp jump in the control gains at approximately 0.8 seconds, approximately the time at which

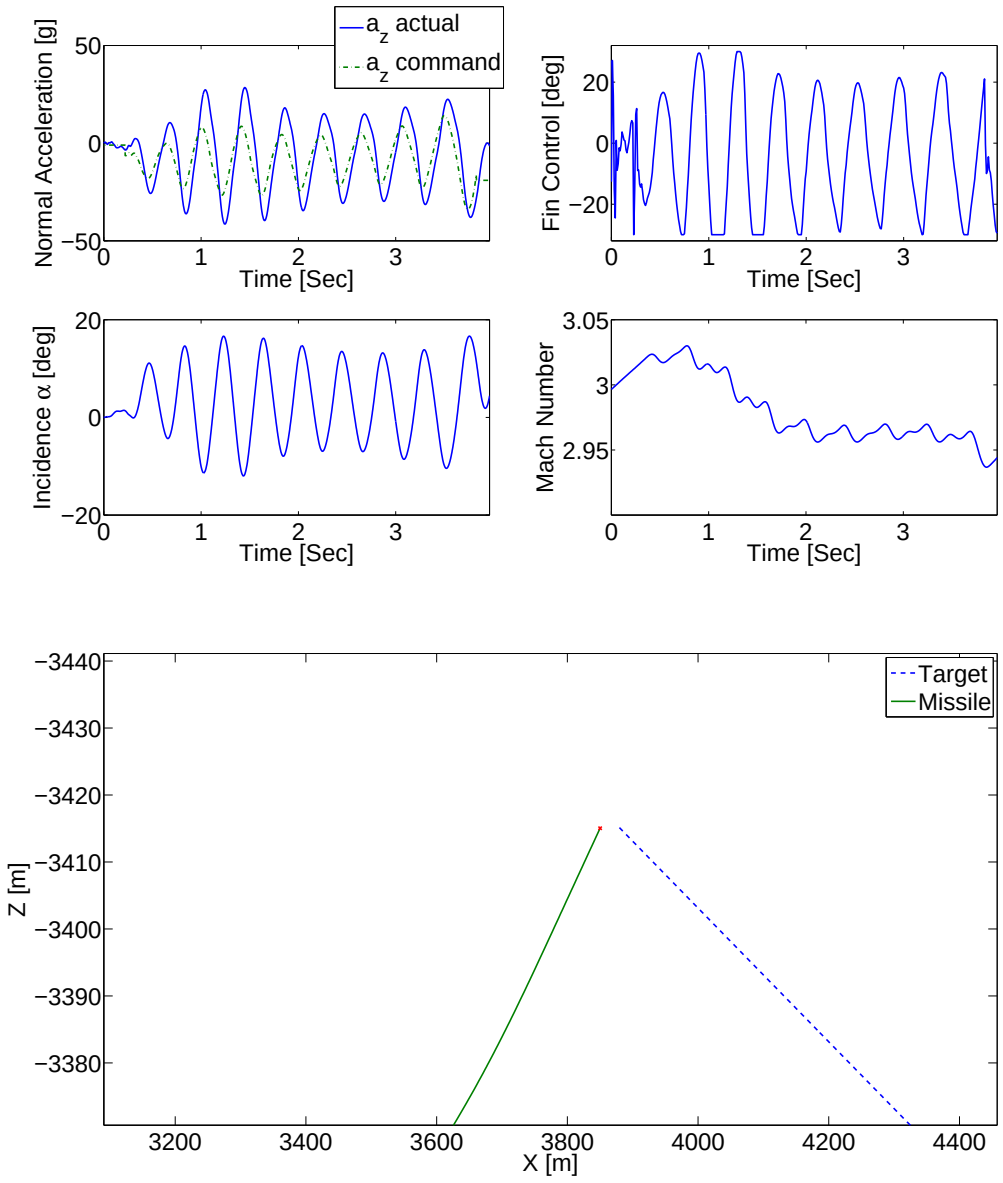


Figure 10.2: RCAC controlled missile with nominal aerodynamic coefficients. In this scenario, the oscillation penalty is removed from the adaptive control cost function.

the seeker locks onto the target.

To eliminate the high frequency content in the missile trajectory, we now choose $\mu(k) = 500$. Figure 10.4 shows the trajectory of the missile and target for same run as Figure 10.2. Note that the normal acceleration and fin control no longer contain high frequency content, this results in a smoother flight path, and a miss distance of

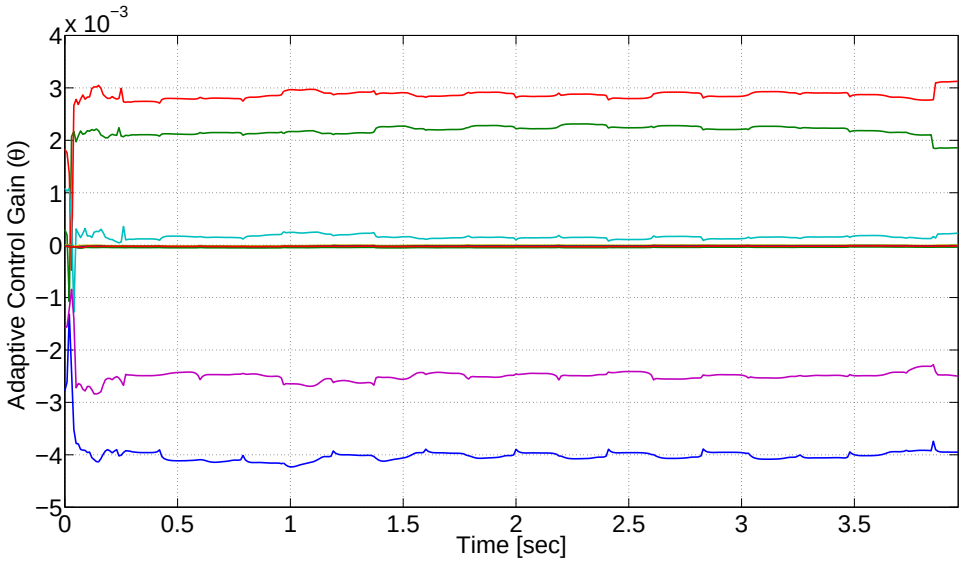


Figure 10.3: Adaptive control gains $\theta(k)$, where the missile has nominal aerodynamic coefficients. In this simulation, the RCAC cost function does not include an oscillation penalty term.

less than 1 meter. Furthermore, as shown in Figure 10.5, the oscillation in the control gains is greatly reduced.

10.6 Case 2 - Aerodynamic Force Coefficient

In this set of scenarios we compare the median miss distance of the autopilot and the adaptively controlled missile when the aerodynamic force coefficient $C_z(\alpha, M, \delta_p)$, is modified by an affine linear function. Table 10.4 lists the scenarios and affine functions used to modify $C_z(\alpha, M, \delta_p)$. For this set of scenarios, the autopilot has a median miss distance ranging from 0.1 meters to 11.29 meters. The adaptively controlled missile has a median distance ranging from 0.17 meters to 35.77 meters. Note that the adaptive controller parameters and gains are not scheduled as in the autopilot setup.

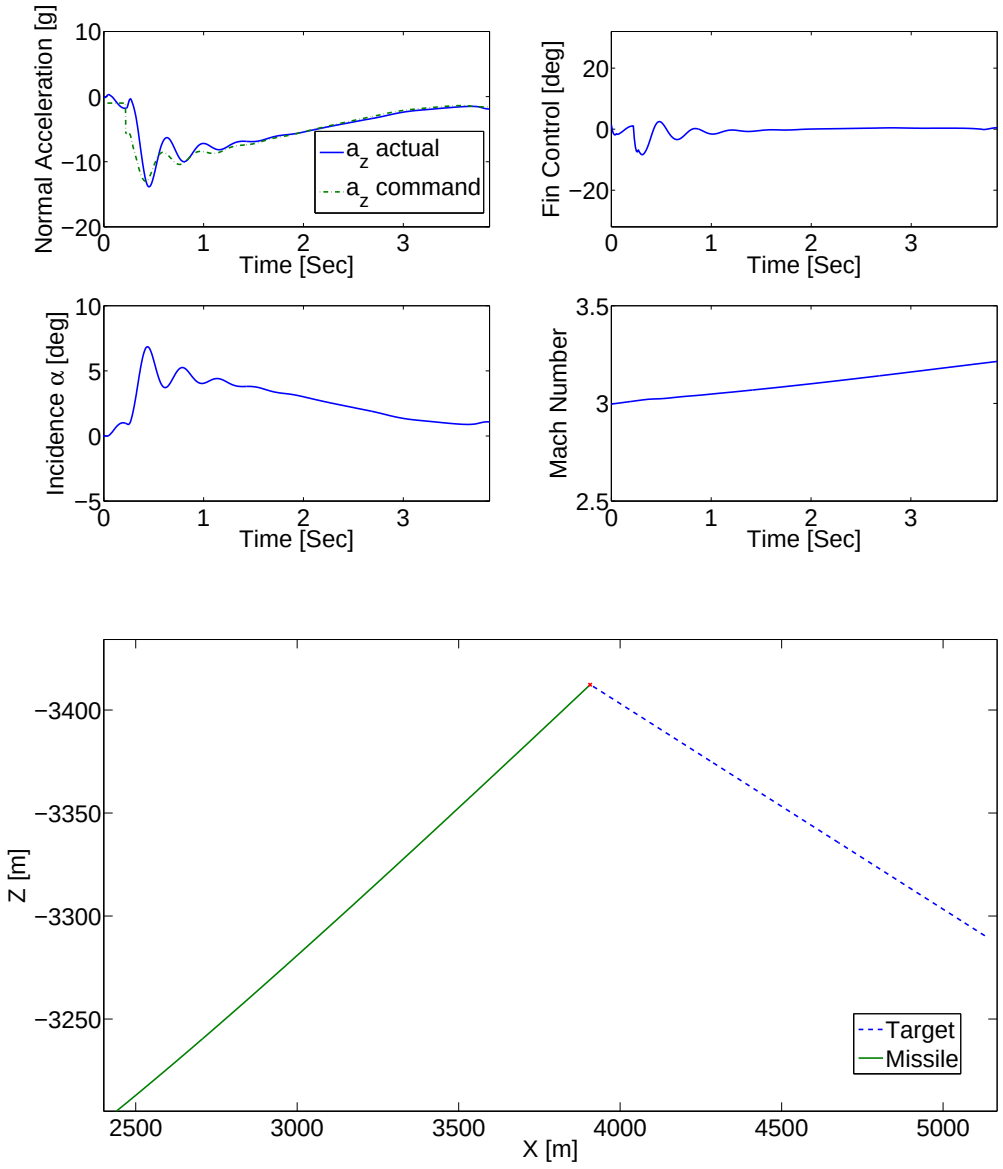


Figure 10.4: RCAC controlled missile with nominal aerodynamic coefficients. Furthermore, a penalty is added to the adaptive control cost function to penalize oscillatory trajectories.

10.6.1 Mitigation of Oscillatory Trajectories for Off-Nominal Aerodynamics

For Scenario 3, we examine a single run of the Monte Carlo simulation to evaluate the presence of oscillatory trajectories. Figure 10.6 shows the autopilot controlled

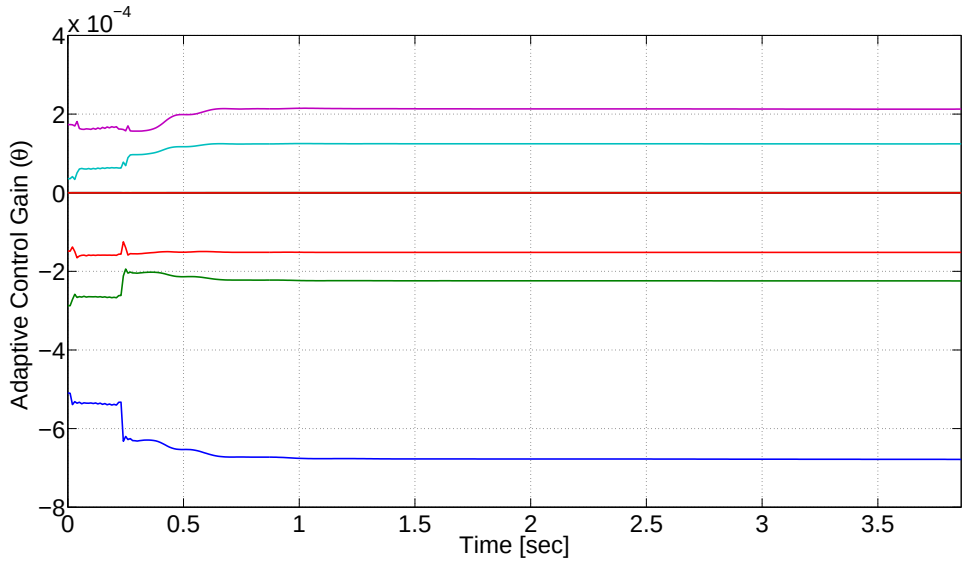


Figure 10.5: Adaptive control gains $\theta(k)$, where the missile has nominal aerodynamic coefficients. Furthermore, a penalty is added to the adaptive control cost function to penalize oscillatory trajectories.

missile trajectory and the target for a single run. Note that in this scenario the normal acceleration and fin control oscillate at high frequency. This behavior creates a miss distance of 8.9 meters.

Choosing $\mu = 500$, the trajectory of the adaptively controlled missile is smoothed out over the portion of the flight before the seeker locks onto the target, as shown in Figure 10.7. Furthermore, the adaptive control gains shown in Figure 10.8 converge to constant values. The resulting miss distance is 1.1 meters.

Scenario	Autopilot Median Miss [m]	Adaptive Control Median Miss [m]	Aerodynamic Transformation
1	0.1	0.17	$3C_z(\alpha, M, \delta_p)$
2	2.74	0.89	$3C_z(\alpha, M, \delta_p) + 3$
3	8.9	1.1	$5C_z(\alpha, M, \delta_p) + 5$
4	0.35	4.67	$0.75C_z(\alpha, M, \delta_p)$
5	0.65	5.2	$0.75C_z(\alpha, M, \delta_p)$ - 0.75
6	11.29	35.77	$0.5C_z(\alpha, M, \delta_p)$ -1

Table 10.3: Scenario and results for off-nominal aerodynamic force coefficients, specifically, the force coefficient $C_z(\alpha, M, \delta_p)$.

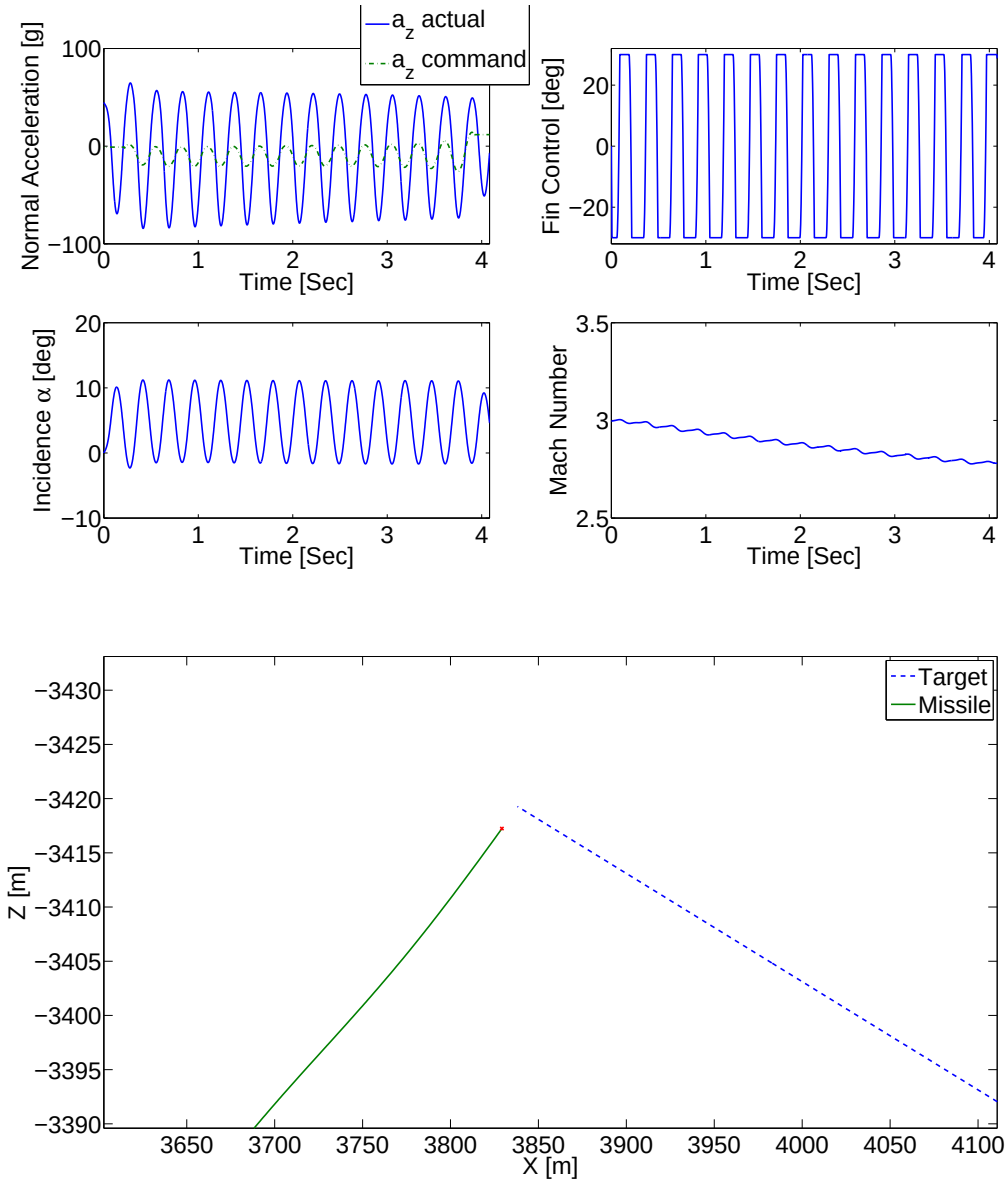


Figure 10.6: Autopilot controlled missile with off-nominal aerodynamic coefficients. In this case $C_z(\alpha, M, \delta_p)$ is perturbed by an affine linear transformation.

10.7 Case 3 - Off-Nominal Aerodynamic Moment Coefficient

In this set of scenarios we compare the median miss distance of the autopilot and the adaptively controlled missile when the aerodynamic moment coefficient $C_z(\alpha, M, \delta_p)$ is modified by an affine linear function. Table 10.4 lists the scenarios and affine functions used to modify $C_m(\alpha, M, \delta_p)$. For this set of scenarios, we demon-

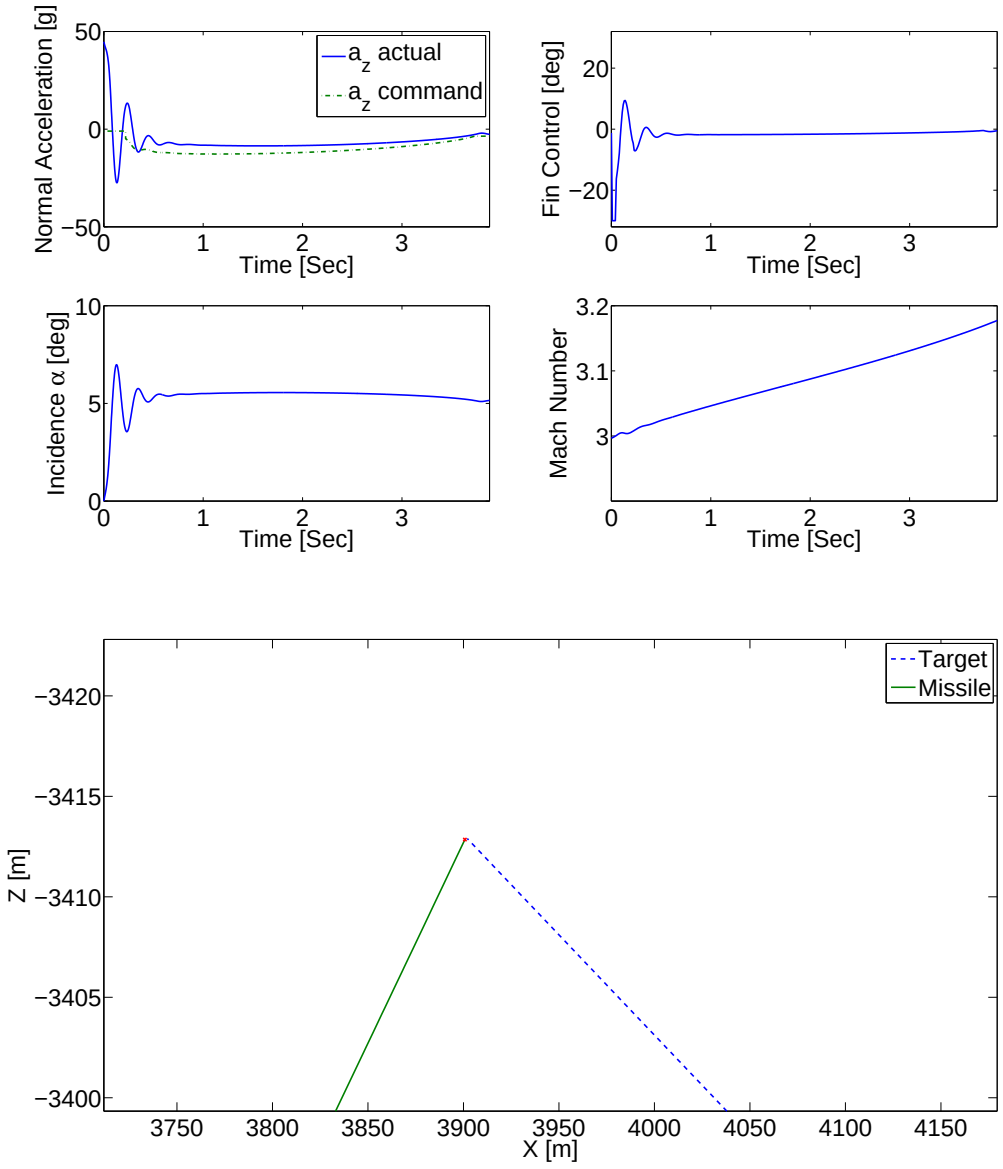


Figure 10.7: RCAC controlled missile with off-nominal aerodynamic coefficients. In this case $C_z(\alpha, M, \delta_p)$ is perturbed by an affine linear transformation. Furthermore, a penalty is added to the RCAC cost function to penalize oscillatory trajectories.

strate that both the autopilot and adaptive controller are robust to changes in the aerodynamic moment coefficient, the median miss distances are all under 1 meter.

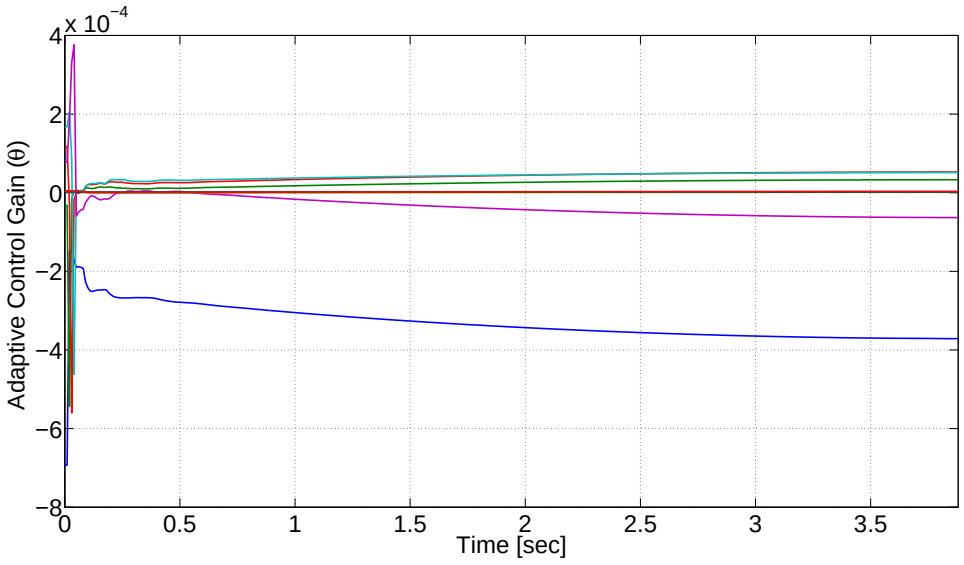


Figure 10.8: Adaptive control gains $\theta(k)$, where $C_z(\alpha, M, \delta_p)$ is perturbed by an affine linear transformation. In this scenario, the cost function includes an oscillation penalty.

10.8 Case 4 - Noisy Body Angle Sensor

We now evaluate the autopilot and adaptive controller performance when noise is added to the body angle sensor q. Note that the autopilot uses the body angle directly for control. However, RCAC uses only the acceleration command-following error for feedback, although the seeker uses the body angle to generate the acceleration. Therefore, the adaptive controller is corrupted indirectly by sensor noise. Figure

Scenario	Autopilot Median Miss [m]	Adaptive Control Median Miss [m]	Aerodynamic Transformation
1	0.08	0.31	$0.5C_m(\alpha, M, \delta_p, q)$
2	0.08	0.61	$0.5C_m(\alpha, M, \delta_p, q)$ -0.5
3	0.07	0.24	$0.5C_m(\alpha, M, \delta_p, q)$ +0.5
4	0.09	0.2	$0.25C_m(\alpha, M, \delta_p, q)$
5	0.07	0.44	$0.25C_m(\alpha, M, \delta_p, q)$+1
6	0.06	0.16	$0.25C_m(\alpha, M, \delta_p, q)$ -1

Table 10.4: Scenario and results for off-nominal aerodynamic coefficients, specifically, in the moment coefficient $C_m(\alpha, M, \delta_p, q)$

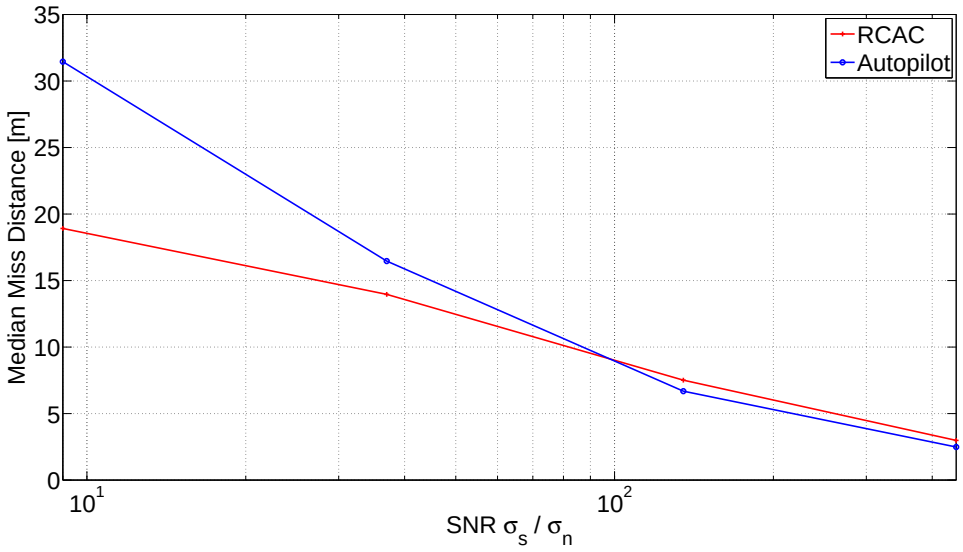


Figure 10.9: Comparison between RCAC controlled missile and autopilot controlled missile under nominal flight conditions. In this scenario we consider noise on the sensing of q. For each point, a Monte Carlo run of 50 is used to determine the median miss distance.

10.9 shows the median miss distance in meters, for the both the autopilot and the adaptive controller for varying signal to noise ratios. Note that the performance of both controllers degrades as the noise increases, however, the adaptive controller degrades more slowly than the autopilot. The adaptive controller yields a smaller median miss distance at a signal to noise ratio of 400.

10.9 Conclusions

In this Chapter we extended the results presented in [33] by using recent developments in retrospective cost adaptive control [122, 37]. In previous work, the adaptive controller was used in an outer loop around the autopilot, since RCAC alone failed to stabilize the missile for most tactical trajectories. The extensions in RCAC, which eliminate the need for knowledge of nonminimum-phase zero locations, allow us to eliminate the autopilot in the the inner loop. We demonstrated that the adaptive

controller yields performance that is comparable to the gain scheduled three-loop autopilot for the nominal case. Furthermore, we demonstrated that in certain off-nominal scenarios, the adaptive controller outperforms the three-loop autopilot. In all cases, the tuning of both the autopilot and adaptive controller is fixed, that is, the tuning parameters are not modified for each scenario.

CHAPTER XI

Application of Adaptive Control to Proportional Integral Derivative Problems

11.1 Introduction

Because of the ease of tuning and implementation, PID control remains one of the most popular and widely used methods in control engineering [126, 127, 128, 129]. In applications involving multiple PID loops operating on MIMO systems with uncertain dynamics, there remains a need for methods that can reliably tune multiple PID loops online based on identified models. Self-tuning and adaptive PID control methods provide a viable approach to this problem [130, 131, 132].

In this chapter we develop a novel approach to digital adaptive MIMO PID control for sampled-data systems. The approach that we adopt is based on retrospective-cost adaptive control (RCAC) developed in [50, 49, 58, 59, 60, 37, 57]. This approach applies to stabilization, command-following, and disturbance-rejection problems for SISO and MIMO plants that are possibly nonminimum phase. From an identification point of view, RCAC requires knowledge of Markov parameters of the plant. The number of required Markov parameters and their accuracy is plant dependent, but typically only a single Markov parameter is needed for SISO systems. Identification of Markov parameters is discussed in [133], while robustness of RCAC to uncertainty

in the Markov parameters is discussed in [38].

For the case of PID control for MIMO systems that are possibly open-loop unstable and nonminimum phase, we develop a variation of RCAC that enforces a digital PID controller structure. This structure assumes direct feedthrough for proportional control, nonrepeated poles at $z = 1$ for integral control, and backward differences for derivative control.

In practice, actuator saturation gives rise to integrator windup, and various techniques have been developed to address this phenomenon [134, 135, 136]. Within the context of RCAC, actuator saturation is addressed by using the actual control input in the regressor step used to reconstruct the past input that optimizes a surrogate cost function, which is the basis of retrospective cost optimization [137].

The contents of this chapter are as follows in Section 11.2 we outline a problem formulation for the adaptive PID controller architecture. In Section 11.3 we present the retrospective cost adaptive control algorithm, with a model structure constrained to yield the coefficients of a PID controller. Finally, in Section 11.4 we demonstrate the method on several of examples of increasing complexity, with saturation, with constant disturbances, and multi-input, multi-output plants.

11.2 Problem Formulation

Consider the MIMO discrete-time system

$$x(k+1) = Ax(k) + B\mathrm{sat}_{[a,b]}[u(k)] + D_1 w(k), \tag{11.1}$$

$$y(k) = Cx(k) + D_2 w(k), \tag{11.2}$$

$$z(k) = E_1 x(k) - r(k), \tag{11.3}$$

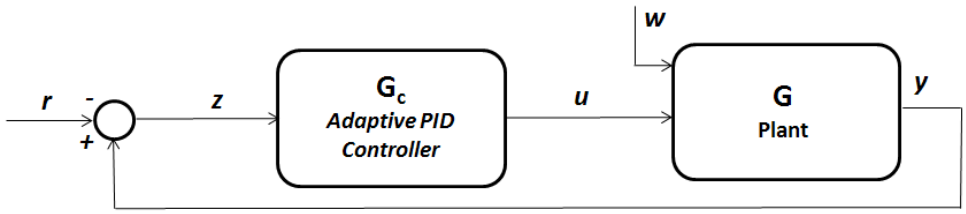


Figure 11.1: Adaptive PID controller architecture

where $x(k) \in \mathbb{R}^n$, $u(k) \in \mathbb{R}^{l_u}$, $y(k) \in \mathbb{R}^{l_y}$, $z(k) \in \mathbb{R}^{l_z}$, $w(k) \in \mathbb{R}^{l_w}$, $r(k) \in \mathbb{R}^{l_w}$, $a \in \mathbb{R}^{l_u}$, $b \in \mathbb{R}^{l_u}$, and $k \geq 0$. Furthermore, $\mathrm{sat}_{[i,j]}[\eta] : \mathbb{R}^{l_u} \to \mathbb{R}^{l_u}$ is the saturation function

$$\mathrm{sat}_{[k,j]}[\eta] \triangleq \begin{cases} j_i, & \text{if } \eta_i \geq j_i, \text{ for } i = 1, \cdots, l_u, \\ u_i{}^-(k), & \text{if } k_i < \eta_i < j_i, \text{ for } i = 1, \cdots, l_u, \\ k_i & \text{if } \eta_i \leq k_i \text{ for } i = 1, \cdots, l_u, \end{cases} \tag{11.4}$$

and the subscript i denotes the i^{th} entry of a vector.

The goal is to develop an adaptive output feedback controller that minimizes the performance variable z in the presence of the disturbance signal w and command r with minimal modeling information about the plant dynamics and w. The block diagram for (11.1)-(11.3) is shown in Figure 11.1. Furthermore, the controller is constrained to have the structure,

$$u(k) = \mathrm{sat}_{[c,d]} \left[\left(K_{\mathrm{P}}(k) + K_{\mathrm{I}}(k) \frac{\mathbf{q}}{\mathbf{q}-1} + K_{\mathrm{D}}(k) \frac{\mathbf{q}-1}{\mathbf{q}} \right) z(k) \right], \tag{11.5}$$

where $c \in \mathbb{R}^{l_u}$ and $d \in \mathbb{R}^{l_u}$ are user specified control authority limitations, where $a_i \leq c_i$ and $b_i \leq d_i$, for $i = 1, \ldots, l_u$, and the gains $K_{\mathrm{P}}(k) \in \mathbb{R}^{l_u \times l_z}$, $K_{\mathrm{I}}(k) \in \mathbb{R}^{l_u \times l_z}$, and $K_{\mathrm{D}}(k) \in \mathbb{R}^{l_u \times l_z}$ are updated by the adaptive algorithm in the next section.

11.3 Retrospective Surrogate Cost

For $i \geq 1$, define the Markov parameters of (A, B, E_1) as

$$H_i \stackrel{\triangle}{=} E_1 A^{i-1} B. \tag{11.6}$$

For example, $H_1 = E_1 B$ and $H_2 = E_1 AB$. Let r be a positive integer. Then, for all $k \geq r$,

$$x(k) = A^r x(k-r) + \sum_{i=1}^{r} A^{i-1} B \mathrm{sat}_{[a,b]}[u(k-i)] + \sum_{i=1}^{r} A^{i-1} D_1 w(k-i), \tag{11.7}$$

and thus

$$z(k) = E_1 A^r x(k-r) + \sum_{i=1}^{r} E_1 A^{i-1} D_1 w(k-i) + E_0 w(k) + \bar{H}\bar{U}(k-1), \tag{11.8}$$

where

$$\bar{H} \stackrel{\triangle}{=} \left[\begin{array}{ccc} H_1 & \cdots & H_r \end{array} \right] \in \mathbb{R}^{l_z \times r l_u}$$

and

$$\bar{U}(k-1) \stackrel{\triangle}{=} \left[\begin{array}{c} \mathrm{sat}_{[a,b]}[u(k-1)] \\ \vdots \\ \mathrm{sat}_{[a,b]}[u(k-r)] \end{array} \right].$$

Next, we rearrange the columns of $\bar{H}$ and the components of $\bar{U}(k-1)$ and partition the resulting matrix and vector so that

$$\bar{H}\bar{U}(k-1) = \mathcal{H}' U'(k-1) + \mathcal{H} U(k-1), \tag{11.9}$$

where $\mathcal{H}' \in \mathbb{R}^{l_z \times (rl_u - l_U)}$, $\mathcal{H} \in \mathbb{R}^{l_z \times l_U}$, $U'(k-1) \in \mathbb{R}^{rl_u - l_U}$, and $U(k-1) \in \mathbb{R}^{l_U}$. Then, we can rewrite (11.8) as

$$z(k) = \mathcal{S}(k) + \mathcal{H}U(k-1), \tag{11.10}$$

where

$$\mathcal{S}(k) \triangleq E_1 A^r x(k-r) + \sum_{i=1}^{r} E_1 A^{i-1} D_1 w(k-i) + E_0 w(k) + \mathcal{H}'U'(k-1). \tag{11.11}$$

Next, for $j = 1, \ldots, s$, we rewrite (11.10) with a delay of k_j time steps, where $0 \le k_1 \le k_2 \le \cdots \le k_s$, in the form

$$z(k-k_j) = \mathcal{S}_j(k-k_j) + \mathcal{H}_j U_j(k-k_j-1), \tag{11.12}$$

where (11.11) becomes

$$\mathcal{S}_j(k-k_j) \triangleq E_1 A^r x(k-k_j-r) + \sum_{i=1}^{r} E_1 A^{i-1} D_1 w(k-k_j-i) + E_0 w(k-k_j) + \mathcal{H}'_j U'_j(k-k_j-1)$$

and (11.9) becomes

$$\bar{H}\bar{U}(k-k_j-1) = \mathcal{H}'_j U'_j(k-k_j-1) + \mathcal{H}_j U_j(k-k_j-1), \tag{11.13}$$

where $\mathcal{H}'_j \in \mathbb{R}^{l_z \times (rl_u - l_{U_j})}$, $\mathcal{H}_j \in \mathbb{R}^{l_z \times l_{U_j}}$, $U'_j(k-k_j-1) \in \mathbb{R}^{rl_u - l_{U_j}}$, and $U_j(k-k_j-1) \in \mathbb{R}^{l_{U_j}}$. Now, by stacking $z(k-k_1), \ldots, z(k-k_s)$, we define the *extended performance*

$$Z(k) \triangleq \begin{bmatrix} z^{\mathrm{T}}(k-k_1) & \cdots & z^{\mathrm{T}}(k-k_s) \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{sl_z}. \tag{11.14}$$

Therefore,

$$Z(k) \triangleq \tilde{\mathcal{S}}(k) + \tilde{\mathcal{H}}\tilde{U}(k-1), \tag{11.15}$$

where

$$\tilde{\mathcal{S}}(k) \triangleq \left[\begin{array}{ccc} \mathcal{S}_1{}^{\mathrm{T}}(k-k_1) & \cdots & \mathcal{S}_s{}^{\mathrm{T}}(k-k_s) \end{array} \right]^{\mathrm{T}} \in \mathbb{R}^{sl_z}, \tag{11.16}$$

$\tilde{U}(k-1)$ has the form

$$\tilde{U}(k-1) \triangleq \left[\begin{array}{ccc} u^{\mathrm{T}}(k-q_1) & \cdots & u^{\mathrm{T}}(k-q_{l_{\tilde{U}}}) \end{array} \right]^{\mathrm{T}} \in \mathbb{R}^{l_{\tilde{U}}}, \tag{11.17}$$

where, for $i = 1, \ldots, l_{\tilde{U}}$, $k_1 \leq q_i \leq k_s + r$, and $\tilde{\mathcal{H}} \in \mathbb{R}^{sl_z \times l_{\tilde{U}}}$ is constructed according to the structure of $\tilde{U}(k-1)$. The vector $\tilde{U}(k-1)$ is formed by stacking $U_1(k-k_1-1), \ldots, U_s(k-k_s-1)$ and removing copies of repeated components.

Next, we define the *surrogate performance*

$$\hat{z}(k-k_j) \triangleq \mathcal{S}_j(k-k_j) + \mathcal{H}_j\hat{U}_j(k-k_j-1), \tag{11.18}$$

where the past controls $U_j(k-k_j-1)$ in (11.12) are replaced by the surrogate controls $\hat{U}_j(k-k_j-1)$. In analogy with (11.14), the *extended surrogate performance* for (11.18) is defined as

$$\hat{Z}(k) \triangleq \left[\begin{array}{ccc} \hat{z}^{\mathrm{T}}(k-k_1) & \cdots & \hat{z}^{\mathrm{T}}(k-k_s) \end{array} \right]^{\mathrm{T}} \in \mathbb{R}^{sl_z} \tag{11.19}$$

and thus is given by

$$\hat{Z}(k) = \tilde{\mathcal{S}}(k) + \tilde{\mathcal{H}}\hat{\tilde{U}}(k-1), \tag{11.20}$$

where the components of $\hat{\tilde{U}}(k-1) \in \mathbb{R}^{l_{\tilde{U}}}$ are the components of $\hat{U}_1(k-k_1-1), \ldots, \hat{U}_s(k-k_s-1)$ ordered in the same way as the components of $\tilde{U}(k-1)$. Subtracting (11.15) from (11.20) yields

$$\hat{Z}(k) = Z(k) - \tilde{\mathcal{H}}\tilde{U}(k-1) + \tilde{\mathcal{H}}\hat{\tilde{U}}(k-1). \tag{11.21}$$

Finally, we define the *regularized retrospective cost function*

$$\bar{J}(\hat{\tilde{U}}(k-1), k) \triangleq \hat{Z}^{\mathrm{T}}(k)R(k)\hat{Z}(k) + \eta(k)\hat{\tilde{U}}^{\mathrm{T}}(k-1)\hat{\tilde{U}}(k-1), \tag{11.22}$$

where $R(k) \in \mathbb{R}^{l_z s \times l_z s}$ is a positive-definite performance weighting and $\eta(k) \geq 0$. The goal is to determine refined controls $\hat{\tilde{U}}(k-1)$ that would have provided better performance than the controls $U(k)$ that were applied to the system. The refined control values $\hat{\tilde{U}}(k-1)$ are subsequently used to update the controller.

Substituting (11.21) into (11.22) yields

$$\bar{J}(\hat{\tilde{U}}(k-1), k) = \hat{\tilde{U}}(k-1)^{\mathrm{T}}\mathcal{A}(k)\hat{\tilde{U}}(k-1) + \hat{\tilde{U}}^{\mathrm{T}}(k-1)\mathcal{B}^{\mathrm{T}}(k) + \mathcal{C}(k), \tag{11.23}$$

where

$$\mathcal{A}(k) \triangleq \tilde{\mathcal{H}}^{\mathrm{T}}R(k)\tilde{\mathcal{H}} + \eta(k)I_{l_{\tilde{U}}}, \tag{11.24}$$

$$\mathcal{B}(k) \triangleq 2\tilde{\mathcal{H}}^{\mathrm{T}}R(k)[Z(k) - \tilde{\mathcal{H}}\tilde{U}(k-1)], \tag{11.25}$$

$$\begin{aligned}\mathcal{C}(k) \triangleq{}& Z^{\mathrm{T}}(k)R(k)Z(k) - 2Z^{\mathrm{T}}(k)R(k)\tilde{\mathcal{H}}\tilde{U}(k-1) \\ &+ \tilde{U}^{\mathrm{T}}(k-1)\tilde{\mathcal{H}}^{\mathrm{T}}R(k)\tilde{\mathcal{H}}\tilde{U}(k-1).\end{aligned} \tag{11.26}$$

If either $\tilde{\mathcal{H}}$ has full column rank or $\eta(k) > 0$, then $\mathcal{A}(k)$ is positive definite. In this

case, $\bar{J}(\hat{\tilde{U}}(k-1), k)$ has the unique global minimizer

$$\hat{\tilde{U}}(k-1) = -\frac{1}{2}\mathcal{A}^{-1}(k)\mathcal{B}(k). \tag{11.27}$$

11.3.1 Controller Construction

Rearranging (11.5), the control $u(k)$ is given by the exactly proper time-series controller of order 3 given by

$$u(k) = \text{sat}_{[c,d]}[\text{sat}_{[a,b]}[u(k-1)] + N_1(k)z(k) + N_2(k)z(k-1) + N_3(k)z(k-2)], \tag{11.28}$$

where, for all $i = 1, \ldots, 3$, $N_i(k) \in \mathbb{R}^{l_u \times l_z}$. Note that the gains in (11.5) and (11.28) are related by

$$N_1 = K_{\mathrm{P}} + K_{\mathrm{I}} + K_{\mathrm{D}}, \tag{11.29}$$

$$N_2 = -K_{\mathrm{P}} - 2K_{\mathrm{D}}, \tag{11.30}$$

$$N_3 = K_{\mathrm{D}}. \tag{11.31}$$

The control (11.28) can be expressed as

$$u(k) = \text{sat}_{[c,d]}\left[\theta(k)\phi(k-1)\right], \tag{11.32}$$

where

$$\theta(k) \stackrel{\triangle}{=} [N_1(k) \ \cdots \ N_3(k)] \in \mathbb{R}^{l_u \times 3l_z} \tag{11.33}$$

and

$$\phi(k-1) \triangleq \begin{bmatrix} z^{\mathrm{T}}(k) & \cdots & z^{\mathrm{T}}(k-2) \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{3l_z}. \tag{11.34}$$

Next let d be a positive integer such that $\tilde{U}(k-1)$ contains $u(k-d)$. We define the cumulative cost function

$$\begin{aligned} J_{\mathrm{R}}(\theta(k)) \triangleq & \sum_{i=d+1}^{k} \lambda^{k-i} \| \phi^{\mathrm{T}}(i-d-1)\theta^{\mathrm{T}}(k) - \hat{u}^{\mathrm{T}}(i-d) + \mathrm{sat}_{[a,b]}[u^{\mathrm{T}}(i-d)] \|^2 \\ & + \lambda^k (\theta(k) - \theta(0)) P^{-1}(0) (\theta(k) - \theta(0))^{\mathrm{T}}, \end{aligned} \tag{11.35}$$

where $\|\cdot\|$ is the Euclidean norm, and $\lambda(k) \in (0,1]$ is the forgetting factor. Minimizing (11.35) yields

$$\begin{aligned} \theta^{\mathrm{T}}(k) \triangleq & \ \theta^{\mathrm{T}}(k-1) + \beta(k) P(k-1)\phi(k-d-1) \\ & \cdot [\phi^{\mathrm{T}}(k-d-1)P(k-1)\phi(k-d-1) + \lambda(k)]^{-1} \\ & \cdot [\phi^{\mathrm{T}}(k-d-1)\theta^{\mathrm{T}}(k-1) - \hat{u}^{\mathrm{T}}(k-d) \\ & + \mathrm{sat}_{[a,b]}[u^{\mathrm{T}}(k-d)]], \end{aligned} \tag{11.36}$$

where $\beta(k)$ is either 0 or 1. When $\beta(k)$ is 1, the controller is allowed to adapt, whereas, when $\beta(k)$ is 0, the controller adaptation is off. The error covariance is updated by

$$\begin{aligned} P(k) \triangleq & (1 - \beta(k))P(k-1) + \beta(k)\lambda^{-1}(k)P(k-1) \\ & - \beta(k)\lambda^{-1}(k)P(k-1)\phi(k-d-1) \\ & \cdot [\phi^{\mathrm{T}}(k-d-1)P(k-1)\phi(k-d-1) + \lambda(k)]^{-1} \\ & \cdot \phi^{\mathrm{T}}(k-d-1)P(k-1). \end{aligned} \tag{11.37}$$

We initialize the error covariance matrix as $P(0) = \gamma I$, where $\gamma > 0$.

11.4 Examples

For illustration, we compare the performance of RCAC with a static PID controller with gains $K_P = -1$, $K_I = -1$, $K_D = -1$. In examples where these gains cause the closed loop to be unstable, we omit the comparison. We set $\eta(k) = 0$ for all of the following examples.

11.4.1 Integrator

Let $G(\mathbf{q}) = \frac{\mathbf{1}}{\mathbf{1-q}}$, where the goal is to track a pulse with amplitude 5 and mean 5. For this example there are no actuator constraints, therefore, $a = -\infty$ and $b = \infty$. Furthermore, there are no disturbances, that is, $w(k) = 0$.

We choose the RCAC tuning parameters as $P(0) = 1$, $\tilde{\mathcal{H}} = H_1$, $c = -0.5$, and $d = 0.5$. Despite no actuator limitations we limit the control authority to improve the transient response.

Figure 11.2(a) compares the reference signal $r(k)$ with the system output $y(k)$, 11.2(b) is the unsaturated control signal $\theta(k)\phi(k-1)$, 11.2(c) shows the saturated control signal $\text{sat}_{[a,b]}[u(k)]$, 11.2(d) are the control gains $\theta(k)$. In this example, the adaptive PID controller is able to track the setpoints. Note that after the first setpoint the controller gains converge to constant values for the remainder of the pulse train.

11.4.2 Integrator with windup

Let $G(\mathbf{q}) = \frac{\mathbf{1}}{\mathbf{1-q}}$, where the goal is to track a pulse with amplitude 5 and mean 5. in the presence of actuator limitations $a = -0.1$ and $b = 0.1$. Furthermore, there are no disturbances, that is, $w(k) = 0$.

We choose the RCAC tuning parameters as $P(0) = 1$, $\tilde{\mathcal{H}} = H_1$, $c = -0.1$, and $d = 0.1$.

Figure 11.3(a) compares the reference signal $r(k)$ with the system output $y(k)$, 11.2(b) is the unsaturated control signal $\theta(k)\phi(k-1)$, 11.3(c) shows the saturated

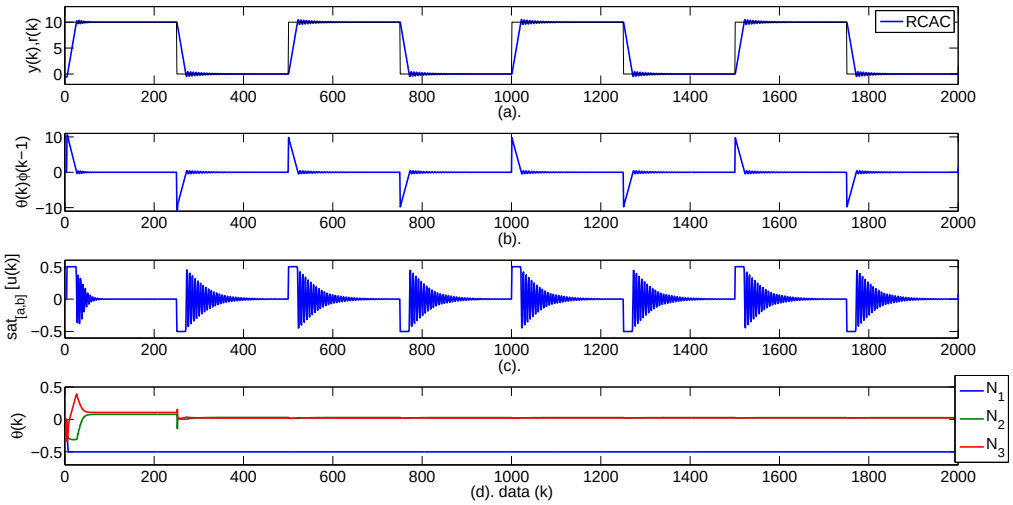


Figure 11.2: Example 1: (a) compares the reference signal $r(k)$ with the system output $y(k)$, (b) is the unsaturated control signal $\theta(k)\phi(k-1)$, (c) shows the saturated control signal $\text{sat}_{[a,b]}[u(k)]$, and (d) shows the control gains $\theta(k)$.

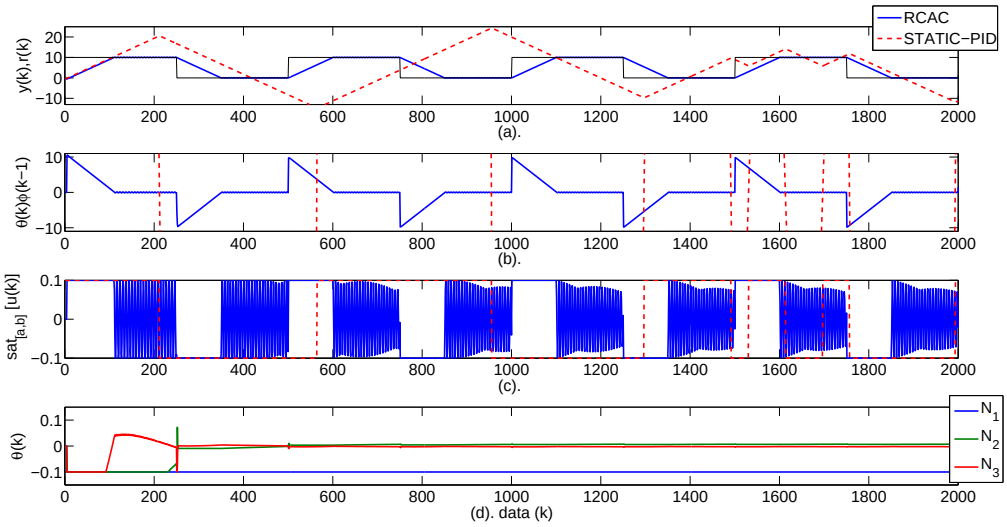


Figure 11.3: Example 2: (a) compares the reference signal $r(k)$ with the system output $y(k)$, (b) is the unsaturated control signal $\theta(k)\phi(k-1)$, (c) shows the saturated control signal $\text{sat}_{[a,b]}[u(k)]$, and (d) shows the control gains $\theta(k)$.

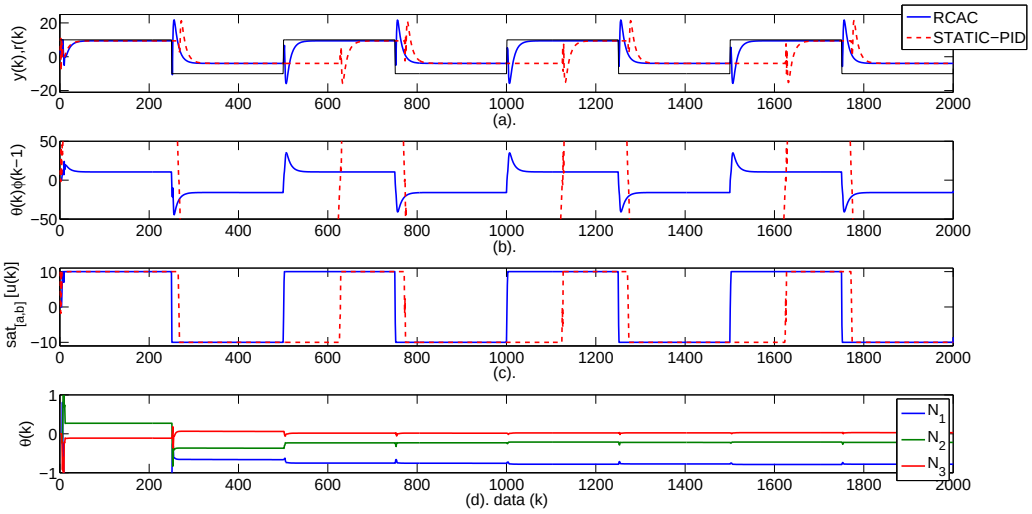


Figure 11.4: Example 3: (a) compares the reference signal $r(k)$ with the system output $y(k)$, (b) is the unsaturated control signal $\theta(k)\phi(k-1)$, (c) shows the saturated control signal $\text{sat}_{[a,b]}[u(k)]$, and (d) shows the control gains $\theta(k)$.

control signal $\text{sat}_{[a,b]}[u(k)]$, and 11.3(d) shows the control gains $\theta(k)$. In this example, the adaptive PID controller is able to track the pulse despite the actuator saturation. The static PID controller with unity gains exhibits integrator windup, resulting in large control demands (on the order of 10^3). We note that the adaptive controller gains converge after the first pulse.

11.4.3 Integrator with constant disturbance

Let $G(\mathbf{q}) = \frac{\mathbf{1}}{\mathbf{1}-\mathbf{q}}$, where the goal is to track a pulse with amplitude 5 and mean 5. Furthermore, let $w(k) = -10$, that is, the goal is to track the setpoint in the presence of a constant disturbance. We assume that there are no actuator limitations in this case. The static PID controller with unity gains is unstable and therefore is omitted.

We choose the RCAC tuning parameters as $P(0) = 0.01$, $\tilde{\mathcal{H}} = H_1$, $c = -\infty$, and $d = \infty$.

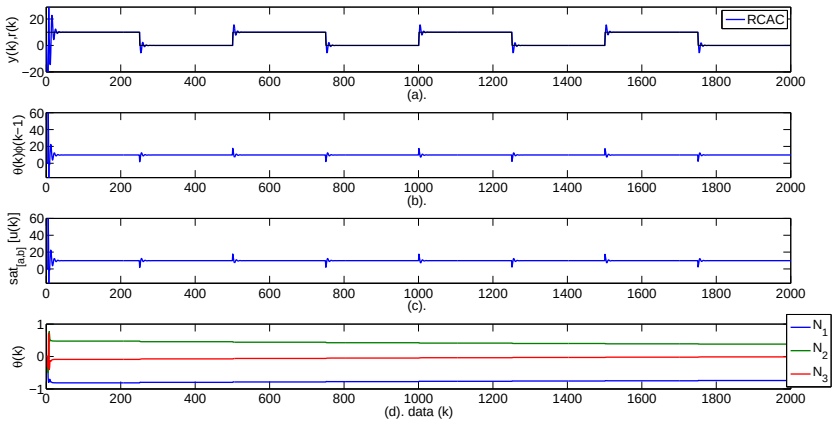


Figure 11.5: Example 4: (a) compares the reference signal $r(k)$ with the system output $y(k)$, (b) is the unsaturated control signal $\theta(k)\phi(k-1)$, (c) shows the saturated control signal $\text{sat}_{[a,b]}[u(k)]$, and (d) shows the control gains $\theta(k)$.

Figure 11.4(a) compares the reference signal $r(k)$ with the system output $y(k)$, 11.4 (b)is the unsaturated control signal $\theta(k)\phi(k-1)$, 11.4(c) shows the saturated control signal $\text{sat}_{[a,b]}[u(k)]$, and 11.4(d) shows the control gains $\theta(k)$. In this example, the adaptive PID controller is able to track the pulse despite the presence of a constant disturbance.

11.4.4 Nonminimum-phase system with constant disturbance

Let $G(\mathbf{q}) = \frac{(\mathbf{q}-\mathbf{1.1})(\mathbf{q}-\mathbf{1.3})}{(\mathbf{q}-\mathbf{0.1})(\mathbf{q}-\mathbf{0.5})(\mathbf{q}-\mathbf{0.9})}$, where the goal is to track a pulse with amplitude 10 and mean 0. Furthermore, let $w(k) = -5$, that is, the goal is to track the setpoint in the presence of a constant disturbance. Furthermore, the actuator saturates at $a = -10$ and $b = 10$.

We choose the RCAC tuning parameters as $P(0) = 1$, $\tilde{\mathcal{H}} = H_1$, $c = -10$, and $d = 10$. Figure 11.5(a) compares the reference signal $r(k)$ with the system output $y(k)$, 11.5 (b)is the unsaturated control signal $\theta(k)\phi(k-1)$, 11.5(c) shows the saturated control signal $\text{sat}_{[a,b]}[u(k)]$, and 11.5(d) shows the control gains $\theta(k)$. In this example,

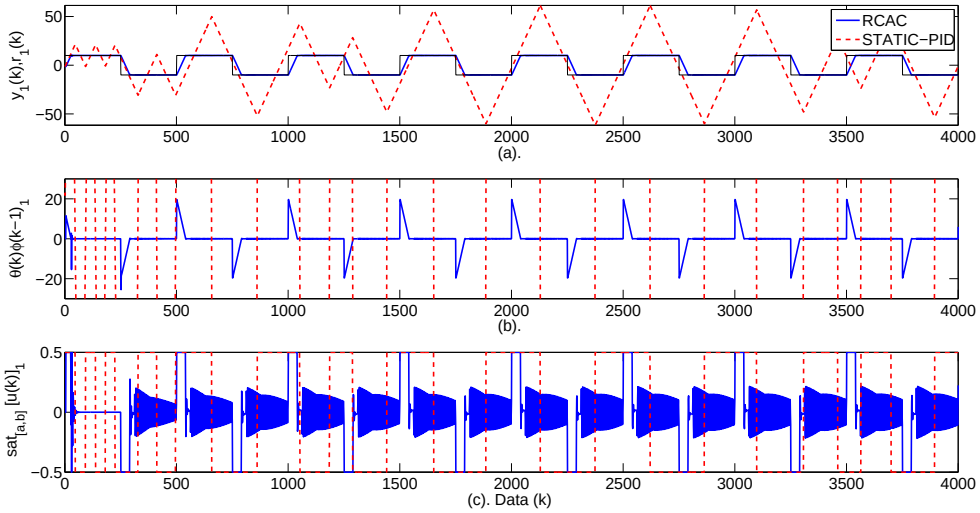


Figure 11.6: Example 5:(a) compares the first channel reference signal $r_1(k)$ with the first system output $y_1(k)$, (b) is the unsaturated channel one control signal $\theta(k)\phi(k-1)_1$, and (c) shows the saturated channel one control signal $\text{sat}_{[a,b]}[u(k)]_1$.

the adaptive PID controller is able to track the pulse despite the disturbance, control authority limitations and nonminimum phase zeros. The static PID controller exhibits integrator windup, resulting in large control demands (on the order of 500).

11.4.5 MIMO integrator with windup

Let $G(\mathbf{q}) = \frac{\mathbf{1}}{\mathbf{1-q}}\mathbf{I_2}$, where the goal is to track a pulse on each output channel. The reference signal for the first channel has amplitude 10 and mean 0. The second channel reference has amplitude 15 and mean 0. We do not consider any disturbances, that is, $w(k) = 0$. Furthermore, the actuator saturates at $a = [-0.5 \ -0.5]$ and $b = [0.5 \ 0.5]$.

We choose the RCAC tuning parameters as $P(0) = 1$, $\tilde{\mathcal{H}} = H_1$, $c = [-0.5 \ -0.5]$, and $d = [0.5 \ 0.5]$. Figure 11.6(a) compares the reference signal for the first channel $r_1(k)$ with the first channel system output $y_1(k)$, 11.6 (b) is the unsaturated control signal from the first channel $\theta(k)\phi(k-1)_1$, and 11.3(c) shows the saturated

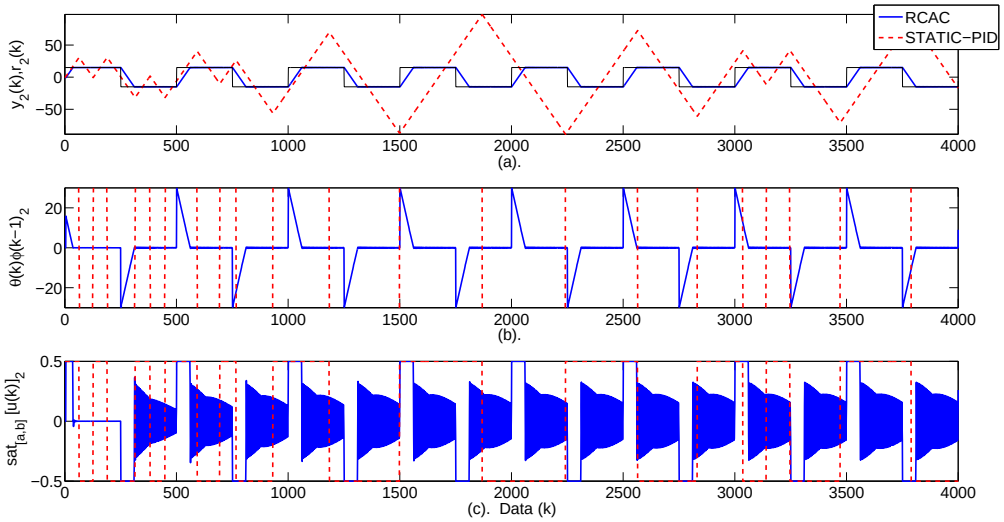


Figure 11.7: Example 5: (a) compares the second channel reference signal $r_2(k)$ with the second system output $y_2(k)$, (b) is the unsaturated channel two control signal $\theta(k)\phi(k-1)_2$, and (c) shows the saturated channel two control signal $\text{sat}_{[a,b]}[u(k)]_2$.

first channel control signal $\text{sat}_{[a,b]}[u(k)]_1$. In this example, the adaptive PID controller is able to track the setpoints despite the actuator limitations. The static PID controller exhibits windup, which results in large control demands (on the order of 10^4). Figure 11.7(a) compares the reference signal for the second channel $r_2(k)$ with the second channel system output $y_2(k)$, 11.7 (b) is the unsaturated first channel control signal $\theta(k)\phi(k-1)_2$, and 11.7(c) shows the saturated first channel control signal $\text{sat}_{[a,b]}[u(k)]_2$. In this example, the adaptive PID controller is able to track the pulse despite the actuator limitations. The static PID controller exhibits windup, which results in large control demands (on the order of 10^4). Figure 11.8 shows the adaptive PID controller gains. Note that the adaptive PID controller is a two-input two-output PID controller, note that the off diagonal elements of the adaptive controller are nonzero.

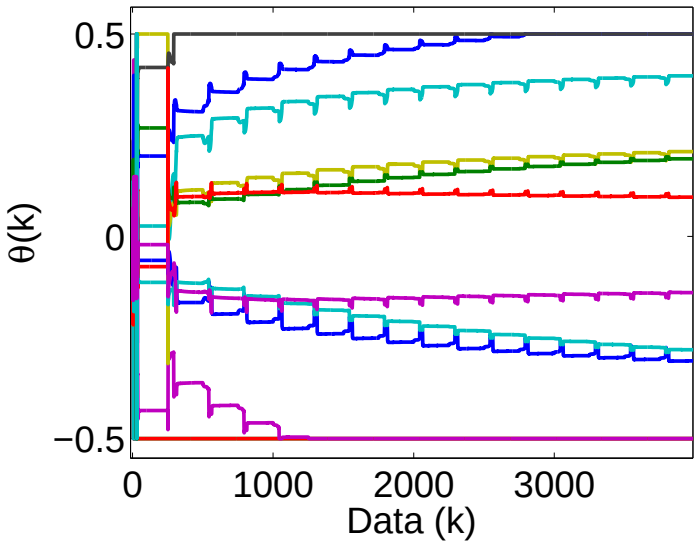


Figure 11.8: Example 5: the coefficients of the multi-input-multi-output PID controller.

11.4.6 MIMO coupled channels

Let

$$G(\mathbf{q}) = \begin{bmatrix} \frac{(\mathbf{q}-0.1)(\mathbf{q}-0.3)}{d} & \frac{(\mathbf{q}-0.5)(\mathbf{q}+0.2)}{d} \\ \frac{(\mathbf{q}-0.7)}{d} & \frac{(\mathbf{q}-0.2\pm 0.3j)}{d} \end{bmatrix}, \tag{11.38}$$

where $d = (\mathbf{q} - 0.5)(\mathbf{q} + 0.5 \pm 0.5)$. The goal is to track a pulse on each output channel. The reference signal on for the first channel has amplitude 10 and mean 0, and the second channel has amplitude 7.5 and mean 2.5, and actuator constraints $a = [-30 \ -30]$ and $b = [30 \ 30]$. Furthermore, let $w_1(k) = 1$, and $w_2(k) = 3$, that is, each channel has a constant disturbance. The static PID controller with unity gains is unstable and is omitted.

We choose the RCAC tuning parameters as $P(0) = 1$, $\tilde{\mathcal{H}} = H_1$, $c = [-30 \ -30]$, and $d = [30 \ 30]$. Figure 11.9(a) compares the reference signal for the first channel, $r_1(k)$, with the first channel system output $y_1(k)$, 11.9 (b) is the unsaturated first channel control signal, $\theta(k)\phi(k-1)_1$, and 11.9(c) shows the saturated first channel

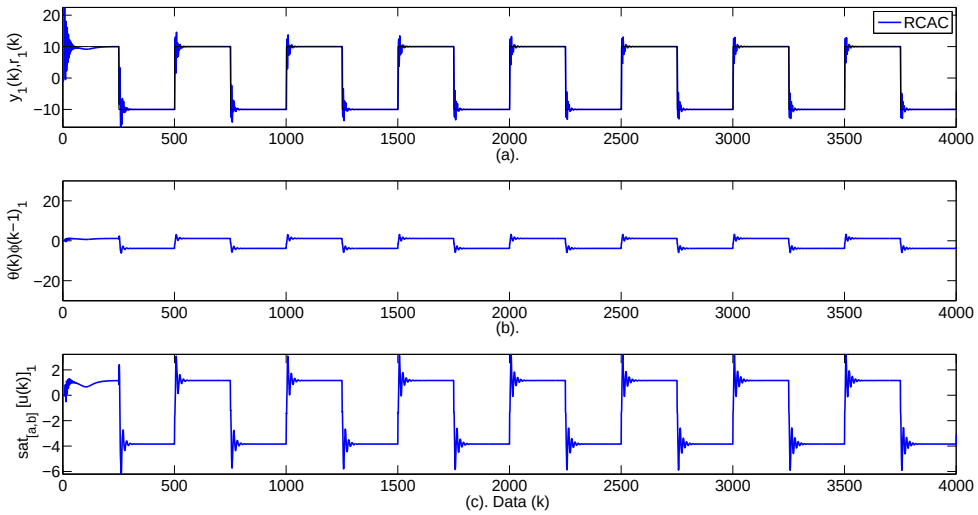


Figure 11.9: Example 6: (a) compares the first channel reference signal $r_1(k)$ with the first system output $y_1(k)$, (b) is the unsaturated channel one control signal $\theta(k)\phi(k-1)_1$, and (c) shows the saturated channel one control signal $\text{sat}_{[a,b]}[u(k)]_1$

control signal $\text{sat}_{[a,b]}[u(k)]_1$. In this example, the adaptive PID controller is able to track the pulse despite the actuator limitations and constant disturbance. Figure 11.10(a) compares the reference signal for the second channel, $r_2(k)$ with the second channel system output, $y_2(k)$, 11.10 (b) is the unsaturated first channel control signal, $\theta(k)\phi(k-1)_2$, and 11.10(c) shows the saturated first channel control signal $\text{sat}_{[a,b]}[u(k)]_2$. In this example, the adaptive PID controller is able to track the pulse despite the actuator limitations and in the presence of the constant disturbance. Figure 11.11 shows the adaptive PID controller gains. In this case, the adaptive PID controller is a two-input-two-output PID controller, note that the off diagonal elements of the adaptive controller are nonzero.

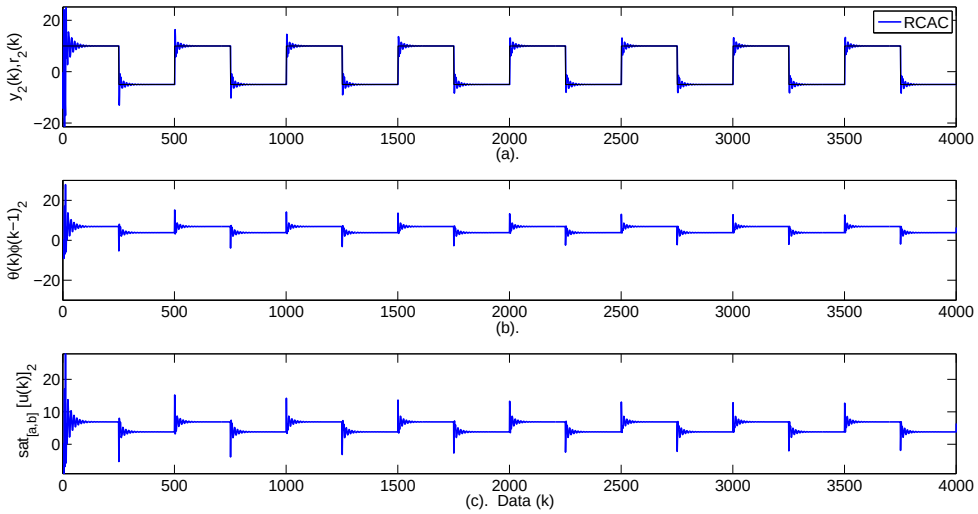


Figure 11.10: Example 6: (a) compares the second channel reference signal $r_2(k)$ with the second system output $y_2(k)$, (b) is the unsaturated channel two control signal $\theta(k)\phi(k-1)_2$, and (c) shows the saturated channel two control signal $\text{sat}_{[a,b]}[u(k)]_2$.

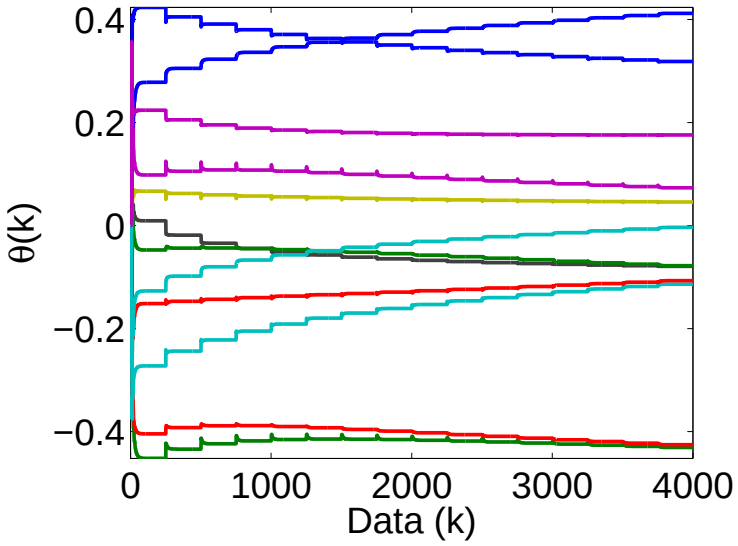


Figure 11.11: Example 6: the coefficients of the multi-input-multi-output PID controller.

11.5 Conclusions

We presented a framework for an adaptive proportional-integrator-derivative (PID) controller, which is applicable to problems that require setpoint tracking and disturbance rejection. The method presented accounts for actuator saturation, constant disturbances, and is applicable to MIMO plants. We update the coefficients of the PID controller using the retrospective cost adaptive control update law, which is reformulated to enforce a PID controller structure. We demonstrated the method on several examples, including cases conducive to integrator windup, constant disturbances, and MIMO plants.

CHAPTER XII

Conclusions and Future Work

12.1 Conclusions

In this dissertation we propose a technique for asymptotic input reconstruction. In Chapter I, we outlined the difficulties encountered in asymptotic input reconstruction. These include the requirement for analytical modeling information, and the possible instability of the input reconstruction error dynamics due to nonminimum-phase zeros. Furthermore, we use input reconstruction as a link between problems in system identification, state estimation and adaptive control.

In Chapter II, we proposed a frequency based technique for asymptotic input reconstruction in the presence of nonminimum-phase zeros under the assumption that the plant is known and the input is harmonic. To summarize, we invert the plant at specific frequencies, and fit a finite-impulse-response (FIR) model through the data at those frequencies. The resulting FIR plant is then used in place of the actual plant inverse, which is unstable if the open-loop plant is nonminimum-phase.

A drawback of this method is the need for complete modeling information of the plant. Another drawback is the requirement for knowledge of the input frequencies. The input frequency information can be detected in a finite fourier transform of the output, but this requires additional computation and logic. Finally, errors due to the difference between the FIR plant and the plant inverse at the input frequencies

add to the input reconstruction error; since there is no feedback mechanism, this error is persistent. In the second part of Chapter II, we introduce an adaptive input reconstruction technique. This technique does not require knowledge of the input frequencies, or an analytical model, provided criteria on the required Markov parameters are met. The adaptive input reconstruction technique introduced in Chapter II, provides the foundation for the remaining chapters addressing model refinement, state estimation, and adaptive control, all formulated as input reconstruction problems. Furthermore, an error analysis and frequency domain argument for successful input reconstruction using the adaptive method are presented in Chapter IX.

In Chapter III, we discuss and demonstrate semiparametric identification of Hammerstein systems. We take advantage of the fact that the linear component of Hammerstein systems can be semi-consistently identified using the input and output signals. Using this model, we utilize the input reconstruction technique discussed in Chapter I to reconstruct the intermediate signal between the static nonlinearity and the linear system. Finally, a nonparametric model of the nonlinearity is constructed using the input and estimated intermediate signal.

In Chapter IV, we discuss model refinement, and in doing so we show the difference between system emulation and subsystem identification. The main difference is that in subsystem identification, the estimated subsystem is an estimate of the unknown subsystem in the physical system. In system emulation, the estimated subsystem simply corrects the closed loop model such that it behaves like the physical system in an input-output sense. In the latter case, the identified subsystem has no physical meaning. Furthermore, we argue that model refinement, in any form, is the estimation of an inaccessible signal; in this case we wish to do system identification, but cannot since we do not know the input to the system. We then reformulate the adaptive input reconstruction technique to use an initial model to reconstruct the inaccessible signal. We apply the method to several linear examples to illustrate the technique.

In Chapters V and VI we apply the methods developed in Chapter IV to nonlinear model refinement. In Chapter V, we identify static parameters and dynamic physical processes in the atmosphere using the space weather model, GITM (Global Ionosphere Thermosphere Model), as the initial model with simulated data gathered on orbit. In Chapter VI, we simulate the process of nondestructively estimating the health of a L_i ion battery using a DFN (Doyle Fuller Newman) battery model as the initial model with simulated battery output data to reconstruct the internal film growth.

In Chapter VII, we discuss and demonstrate semiparametric identification of Wiener systems. We take of advantage of the fact that harmonic signals passed through a static nonlinearity maintain their signal symmetry in the time-domain. This allows us to identify a nonparametric model of the nonlinearity, assuming we can pass a signal with a single harmonic through the system. Using the nonparametric model, we then estimate the intermediate signal between the static nonlinearity and the linear system using input reconstruction.

In Chapter IX and XI, we demonstrate the connection between asymptotic input reconstruction and adaptive control. We analyze the stability of the adaptive input reconstruction method formulated as an adaptive control problem. Furthermore, we provide frequency domain criteria on the Markov parameters used to implement the adaptive input reconstruction technique. The technique developed in this dissertation uses an infinite-impulse-response (IIR) model structure. In Chapter XI, we restrict the model structure to a proportional, integral, derivative (PID) model structure to demonstrate, the flexibility of the algorithm, as well as how a PID controller might be tuned automatically in applications where a typical PID controller would have to be tuned manually.

12.2 Proposed Future Work

This dissertation brings together a number of techniques under the common framework of asymptotic input reconstruction. We propose a short summary of possible future directions in each of the following areas.

Throughout the dissertation, we highlight one of the main issues in asymptotic input reconstruction: nonminimum-phase zeros. However, in practice this is only true in the square case, that is, the system has the same number of inputs as outputs. We demonstrate in the dissertation that in the square case, the poles of the input reconstruction error dynamics contain the transmission zeros of the open loop system. However, in the non-square cases, it is not obvious what the poles of the input reconstruction error dynamics are. Future research might focus on gaining insight into the physical meaning of these numbers.

Many of the applications presented in this dissertation involve nonlinear systems. However, we do not provide motivation for why a technique developed for linear systems can be successfully applied to nonlinear systems. From the stability analysis we can justify scaling the required modeling information without destabilizing the input reconstruction algorithm. This results in robustness of the algorithm; as long as the Markov parameters do not change sign, the frequency domain criteria and error dynamics stability will not be violated. Some work has been done to verify this in simulation. Suppose we were to linearize a nonlinear system and compute the Markov parameters at each time-step, we would speculate that as long as the sign of the Markov parameter does not change, a static value may be used in the algorithm. If the sign of the Markov parameter changes, then the Markov parameter used in the algorithm may need to be scheduled. Much work remains to be done to guarantee successful application of the adaptive input reconstruction algorithm on nonlinear systems.

The work on identification of Hammerstein systems uses a standard model inverse

as the input reconstruction technique. This work was completed prior to the development of the adaptive input reconstruction method. Therefore, error in the linear system identification and nonparametric identification of the static nonlinearity does not improve as more data is given to the algorithm. Revisiting this problem with the updated techniques may yield improved results.

In model refinement, the results presented are based on a 1D GITM simulation. Specifically, we estimate parameters and physical process in a 1D, 400 km high column of the atmosphere above the Earth's surface. This work is currently being extended to 3D GITM.

Recently, the adaptive input reconstruction technique was used to estimate F10.7, which is a measurement of solar intensity, and the primary driver of the GITM model. The performance of the adaptive method should be compared to that of other estimation algorithms, specifically, DART (Data Assimilation Research Testbed). While the adaptive method uses a single ensemble, methods such as DART use many ensembles to estimate parameters such as F10.7. A study study comparison may demonstrate possible drawbacks, if any, of using multiple ensembles, which is a computational expensive process.

The work carried out regarding health monitoring of L_i ion batteries is a demonstration of concept. To estimate the unknown film growth we used input reconstruction through a nonlinear model. Validation of our predictions using experimental results in collaboration with the Department of Mechanical Engineering, is feasible. This application is especially challenging since the observation map, or C matrix in the linear sense, changes with time. Depending on the cycle, the output was either voltage or current. Additionally, the film growth was unidentifiable during certain phases of the cycle. Our results reflect the unidentifiability of the film resistance, since the estimates drop off to zero during these intervals. This required us to reset the input reconstruction at each cycle. This application introduces a number of compli-

cating factors, which require investigation, such as, asymptotic input reconstruction for time-varying systems, and identifiability of inaccessible signals in general.

In the state estimation application, we do not consider the effect of sensor noise on the state estimates. Work by others on input reconstruction based state estimators involves formulating algorithms based on a Kalman filter, which allows knowledge of noise covariances to be used to improve state estimates. While these state estimators are limited by nonminimum-phase zeros, we may be able to reformulate the adaptive input reconstruction in a similar fashion to also take advantage of known noise covariances.

Finally, in the adaptive control application, we provide guidelines for stability of the algorithm. However, as with most adaptive methods, these guidelines only provide asymptotic properties. While we develop "rules of thumb" for transient behavior by tuning specific parameters, there are no guarantees on the magnitude of transients. Adaptive control is unique in this sense; model refitment and state estimation are unaffected by large transients since they are software based. Control, on the other hand, involves putting real signals into physical systems, in this case, the presence of large transients may result in severe consequences. Thus, more research needs to be conducted regarding the stability of the proposed algorithm.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] D. J. Pawlowski and A. J. Ridley, "Quantifying the effect of thermospheric parameterization in a global model," *J. Atmos. and Sol.-Terr. Phys.*, vol. 71, pp. 2017–2026, 2009.

[2] R. Pintelon and J. Schoukens, *System Identification: A Frequency Domain Approach*. Piscataway, NJ: IEEE Press, 2001.

[3] L. Ljung, *System Identifftication: Theory for the User*, 2nd edition ed. Prentice Hall, 1999.

[4] J. N. Juang, *Applied System Identification*. Upper Saddle River, NJ: Prentice-Hall, 1993.

[5] T. Katayama, *Subspace Methods for System Identification*. Englewood Cliffs, New Jersey: Springer, 2005.

[6] C. Minas and D. Inman, "Matching finite element models to modal data," *J. Vibration Acoust.*, vol. 112, pp. 84–92, October 1990.

[7] J. B. Carvalho, B. N. Datta, W. Lin, and C. Wang, "Symmetry preserving eigenvalue embedding in finite-element model updating of vibrating structures," *J. Sound Vibration*, vol. 290, pp. 839–864, March 2006.

[8] M. I. Friswell and J. E. Mottershead, *Finite Element Model Updating in Structural Dynamics*. Dordrecht: Kluwer, 1995.

[9] S. Mijanovic, G. E. Stewart, G. A. Dumont, and M. S. Davies, "A controller perturbation technique for transferring closed-loop stability between systems," *Automatica*, vol. 39, pp. 1783–1791, 2003.

[10] S. O. R. Moheimani, "Model correction for sampled-data models of structures," *J. Guidance, Contr., and Dynamics*, vol. 24, pp. 634–637, March 2001.

[11] H. J. Palanthandalam-Madapusi, E. L. Renk, and D. S. Bernstein, "Data-Based Model Refinement for Linear and Hammerstein Systems Using Subspace Identification and Adaptive Disturbance Rejection," in *Proc. Conf. Contr. Appl.*, ser. Toronto, Canada, August 2005, pp. 1630–1635.

[12] L. Xie and Y. C. Soh, "Robust Kalman filtering for uncertain systems," *Systems and Control Letters*, vol. 22, pp. 123–129, February 1994.

[13] G. Chen and C. K. Chui, "A modified adaptive Kalman filter for real-time applications," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 27, pp. 149–154, January 1991.

[14] R. K. Mehra, "On the identification of variances and adaptive Kalman filtering," *IEEE Transactions on Automatic Control*, vol. AC-15, pp. 175–184, 1970.

[15] I. R. Petersen and A. V. Savkin, *Robust Kalman Filtering for Signals and Systems with Large Uncertainties*. Springer, 1999.

[16] L. M. Silverman, "Inversion of multivariable linear systems," *IEEE Trans. Autom. Control*, vol. AC-14, pp. 270–276, March 1969.

[17] M. K. Sain and J. L. Massey, "Invertibility of linear time-invariant dynamical systems," *IEEE Trans. Autom. Control*, vol. AC-14, pp. 141–149, February 1969.

[18] A. S. Willsky, "On the invertibility of linear systems," *IEEE Trans. Autom. Control*, vol. AC-19, pp. 272–274, March 1974.

[19] P. J. Moylan, "Stable inversion of linear systems," *IEEE Trans. Autom. Control*, vol. 22, pp. 74–78, January 1977.

[20] T. Floquet and J. P. Barbot, "State and unknown input estimation for linear discrete-time systems," *Automatica*, vol. 42, pp. 1883–1889, 2006.

[21] M. Corless and J. Tu, "State and input estimation for a class of uncertain systems," *Automatica*, vol. 34, pp. 757–764, June 1998.

[22] E. Zattoni, G. Marro, and D. S. Bernstein, "A Markov-parameter-based Method for Online Reconstruction of Unknown State and Input in Discrete-Time Systems," in *Proc. Conf. Dec. Contr.*, ser. Atlanta, GA, December 2010, pp. 6022–6027.

[23] G. Marro, D. Prattichizzo, and E. Zattoni, "Convolution profiles for right-inversion of multivariable nonminimum phase discrete-time systems," *Automatica*, vol. 38, pp. 1695–1703, October 2002.

[24] Y. Xiong and M. Saif, "Unknown disturbance inputs estimation based on a state functional observer design," *Automatica*, vol. 39, pp. 1389–1398, 2003.

[25] H. Palanthandalam-Madapusi and D. S. Bernstein, "A Subspace Algorithm for Simultaneous Identification and Input Reconstruction," *Int. J. Adaptive Contr. Sig. Proc.*, vol. 23, pp. 1053–1069, 2009.

[26] G. Marro and E. Zattoni, "Unknown-state, unknown-input reconstruction in discrete-time nonminimum-phase systems: geometric methods," *Automatica*, vol. 46, May 2010.

[27] S. Kirtikar, H. Palanthandalam-Madapusi, E. Zattoni, and D. S. Bernstein, "l-Delay Input and Initial-State Reconstruction for Discrete-Time Linear Systems," *Circ. Sys. Sig. Processing*, vol. 30, pp. 233–262, 2011.

[28] A. M. D'Amato, K. S. Mitchell, B. O. S. Teixeira, and D. S. Bernstein, "Semiparametric Identification of Hammerstein Systems Using Input Reconstruction and a Single Harmonic Input," in *Proc. Conf. Dec. Contr.*, ser. Atlanta, GA, December 2010, pp. 6365–6370.

[29] A. M. Morozov, A. M. D'Amato, A. A. Ali, A. J. Ridley, and D. S. Bernstein, "Retrospective-Cost-Based Model Refinement for System Emulation and Subsystem Identification," in *Proc. Conf. Dec. Contr.*, ser. Orlando, FL, December 2011.

[30] A. M. D'Amato, A. J. Ridley, and D. S. Bernstein, "Retrospective-cost-based adaptive model refinement for the ionosphere and thermosphere," *Statistical Analysis and Data Mining*, vol. 4, pp. 446–458, 2011.

[31] A. M. D'Amato, B. O. S. Teixeira, and D. S. Bernstein, "Semiparametric Identification of Wiener Systems Using a Single Harmonic Input and Retrospective Cost Optimization," in *Proc. Amer. Contr. Conf.*, ser. Baltimore, MD, July 2010, pp. 4498–4503.

[32] A. M. D'Amato, J. C. Springmann, A. Ali, J. W. Cutler, A. J. Ridley, and D. S. Bernstein, "Adaptive State Estimation for Uncertain Systems with Uncertain Noise Spectra," in *Proc. AIAA Guidance, Navigation and Control Conference*, ser. Portland, OR, August 2010, pp. AIAA–2010–7577.

[33] R. J. Fuentes, J. B. Hoagg, B. J. Anderton, A. M. D'Amato, and D. S. Bernstein, "Investigation of Cumulative Retrospective Cost Adaptive Control for Missile Application," in *Proc. AIAA Guidance, Navigation and Control Conference*, Toronto, ON, August 2010, pp. AIAA–2010–7577.

[34] M. Hou and R. J. Patton, "Input observability and input reconstruction," *Automatica*, vol. 34, pp. 789–794, June 1998.

[35] J. Ghosh and B. Paden, "A Pseudoinverse-Based Iterative Learning Control," *IEEE Trans. Automatic Control*, vol. 47, pp. 831–837, May 2002.

[36] ——, "Iterative Learning Control for Nonlinear Nonminimum Phase Plants," *Trans. ASME*, vol. 123, pp. 21–30, 2001.

[37] A. M. D'Amato, E. D. Sumer, and D. S. Bernstein, "Frequency-Domain Stability Analysis of Retrospective-Cost Adaptive Control for Systems with Unknown

Nonminimum-Phase Zeros," in *Proc. Conf. Dec. Contr.*, ser. Orlando, FL, December 2011.

[38] E. D. Sumer, A. M. D'Amato, A. M. Morozov, J. B. Hoagg, and D. S. Bernstein, "Robustness of Retrospective Cost Adaptive Control to Markov-Parameter Uncertainty," in *Proc. Conf. Dec. Contr.*, ser. Orlando, FL, December 2011.

[39] A. M. D'Amato, B. O. S. Teixeira, and D. S. Bernstein, "Semiparametric Identification of Wiener Systems Using a Single Harmonic Input and Retrospective Cost Optimization," *IET Contr. Theory Appl.*, vol. 5, pp. 594–605, 2011.

[40] A. M. D'Amato, J. S. Springmann, A. Ali, J. W. Cutler, A. J. Ridley, and D. S. Bernstein, "Adaptive State Estimation for Uncertain Systems with Uncertain Noise Spectra,," in *Proc. AIAA Guidance, Navigation and Control Conference*, ser. Portland, OR, August 2011, pp. AIAA–2010–7577.

[41] A. M. D'Amato, A. J. Ridley, and D. S. Bernstein, "Retrospective-cost-based adaptive model refinement for the ionosphere and thermosphere," *Statistical Analysis and Data Mining*, vol. 4, pp. 446–458, 2011.

[42] L. A. Aguirre, M. C. S. Coelho, and M. V. Corra, "Decoupling the linear and nonlinear parts in Hammerstein model identification," *Automatica*, vol. 40, pp. 671–676, April 2004.

[43] P. Crama and J. Schoukens, "Initial Estimates of Wiener and Hammerstein Systems Using Multisine Excitation," *IEEE Trans. Instrum. Meas.*, vol. 50, pp. 1791–1795, June 2001.

[44] J. Wang, Q. Zhang, and L. Ljung, "Revisiting Hammerstein System Identification Through the Two-Stage Algorithm for Bilinear Parameter Estimation," *IEEE Trans. Instrum. Meas.*, vol. 45, pp. 2627–2633, November 2009.

[45] E. W. Bai and J. Reyland, "Towards Identification of Wiener Systems with the Least Amount of *a Priori* Information: IIR cases," *Automatica*, vol. 45, pp. 956–964, 2009.

[46] E. W. Bai, " Frequency Domain Identification of Wiener Models," *Automatica*, vol. 39, pp. 1521–1530, September 2003.

[47] W. Greblicki and M. Pawlak, *System Identification: A Frequency Domain Approach*. Cambridge University Press, 2008.

[48] R. Venugopal and D. S. Bernstein, "Adaptive Disturbance Rejection Using ARMARKOV System Representations," *IEEE Trans. Contr. Sys. Tech.*, vol. 8, pp. 257–269, 2000.

[49] M. A. Santillo and D. S. Bernstein, "Adaptive Control Based on Retrospective Cost Optimization," *AIAA J. Guid. Contr. Dyn.*, vol. 33, pp. 289–304, 2010.

[50] J. B. Hoagg, M. A. Santillo, and D. S. Bernstein, "Discrete-Time Adaptive Command Following and Disturbance Rejection for Minimum Phase Systems with Unknown Exogenous Dynamics," *IEEE Trans. Autom. Contr.*, vol. 53, pp. 912–928, 2008.

[51] J. N. Juang, *System Identification: Theory for the User*, 2nd edition ed. Prentice Hall, 1999.

[52] A. A. Ali, A. M. D'Amato, M. S. Holzel, S. L. Kukreja, and D. S. Bernstein, "Consistent identification of hammerstein systems using an ersatz nonlinearity," in *Proc. Amer. Contr. Conf.*, San Francisco, CA, June 2011, pp. 1242–1246.

[53] M. A. Santillo, A. M. D'Amato, and D. S. Bernstein, "System Identification Using a Retrospective Correction Filter for Adaptive Feedback Model Updating," in *Proc. Amer. Contr. Conf.*, ser. St. Louis, MO, June 2009, pp. 4392–4397.

[54] A. M. D'Amato and D. S. Bernstein, "Linear Fractional Transformation Identification Using Retrospective Cost Optimization," in *Proc. SYSID*, ser. Saint-Malo, France, July 2009, pp. 450–455.

[55] A. M. D'Amato, B. J. Arritt, J. A. Banik, E. V. Ardelean, and D. S. Bernstein, "Structural Health Determination and Model Refinement for a Deployable Composite Boom," in *AIAA SDM Conf.*, ser. Palm Springs, CA, April 2009, pp. AIAA–2009–2373.

[56] A. M. D'Amato, A. R. Wu, K. S. Mitchell, S. L. Kukreja, and D. S. Bernstein, "Damage Localization for Structural Health Monitoring Using Retrospective Cost Model Refinement," in *AIAA SDM Conf.*, ser. Orlando, FL, April 2010, pp. AIAA–2010–2628.

[57] A. M. D'Amato, E. D. Sumer, and D. S. Bernstein, "Retrospective cost adaptive control for systems with unknown nonminimum-phase zeros," in *AIAA Guid. Nav. Contr. Conf.*, Portland, OR, August 2011, AIAA-2009-2373.

[58] J. B. Hoagg and D. S. Bernstein, "Retrospective Cost Adaptive Control for Nonminimum-Phase Discrete-Time Systems Part 1: The Ideal Controller and Error System, Part 2: The Adaptive Controller and Stability Analysis," in *Proc. Conf. Dec. Contr.*, ser. Atlanta, GA, December 2010, pp. 893–904.

[59] ——, "Retrospective Cost Model Reference Adaptive Control for Nonminimum-Phase Discrete-Time Systems, Part 2: The Adaptive Controller and Stability Analysis," in *Proc. Conf. Dec. Contr.*, ser. San Francisco, CA, June 2011, pp. 2927–2932.

[60] ——, "Retrospective Cost Model Reference Adaptive Control for Nonminimum-Phase Discrete-Time Systems, Part 1: The Ideal Controller and Error System," in *Proc. Conf. Dec. Contr.*, ser. San Francisco, CA, June 2011, pp. 2933–2938.

[61] P. V. Overschee and B. D. Moor, *Subspace Identification for Linear Systems: Theory, Implementation, Applications*. Kluwer, 1996.

[62] J. S. Bendat, *Nonlinear Systems Techniques and Applications*. Wiley, 1989.

[63] J. Sjberg, Q. Zhang, L. Ljung, A. Benveniste, B. Deylon, P. Y. Glorennec, H. Hjalmarsson, and A. Juditsky, "Nonlinear Black-Box Modeling in System Identification: A Unified Overview," *Automatica*, vol. 31, pp. 1691–1724, December 1995.

[64] R. Haber and L. Keviczky, *Nonlinear System Identification–Input-Output Modeling Approach Vol. 1: Nonlinear System Parameter Identification*. Kluwer Academic Publishers, 1999.

[65] O. Nelles, *Nonlinear System Identification*. Springer, 2001.

[66] H. Palanthandalam-Madapusi, D. S. Bernstein, and A. J. Ridley, "Space Weather Forecasting: Identifying Periodically Switching Block-structured Models to Predict Magnetic-field Fluctuations," *IEEE Contr. Sys. Mag.*, vol. 27, pp. 109–123, October 2007.

[67] F. D'Andrea and R. Vautard, "Reducing systematic errors by empirically correcting model errors," *Tellus*, vol. 52A, pp. 21–41, March 2000.

[68] T. DelSole and A. Y. Hou, "Empirical correction of a dynamical model. Part I: Fundamental issues," *Monthly Weather Rev.*, vol. 127, pp. 2533–2545, November 2001.

[69] C. M. Danforth, E. Kalnay, and T. Miyoshi, "Estimating and correcting global weather model error," *Monthly Weather Rev.*, vol. 135, pp. 281–299, February 2007.

[70] J. B. Hoagg and D. S. Bernstein, "Cumulative Retrospective Cost Adaptive Control with RLS-Based Optimization," in *Proc. Amer. Contr. Conf.*, ser. Baltimore, MD, June 2010, pp. 4016–4021.

[71] A. J. Ridley, Y. Deng, and G. Tòth, "The global ionosphere-thermosphere model," *J. Atmos. Sol-Terr. Phys.*, vol. 68, p. 839, 2006.

[72] M. S. Holzel, M. A. Santillo, J. B. Hoagg, and D. S. Bernstein, "Adaptive Control of the NASA Generic Transport Model Using Retrospective Cost Optimization," in *Proc. AIAA Guid. Nav. Contr. Conf.*, ser. Chicago, IL, August 2009, pp. AIAA–2009–5616.

[73] M. S. Fledderjohn, Y. C. Cho, J. B. Hoagg, M. A. Santillo, W. Shyy, and D. S. Bernstein, "Retrospective Cost Adaptive Flow Control Using a Dielectric Barrier Discharge Actuator," in *Proc. AIAA Guid. Nav. Contr. Conf.*, ser. Chicago, IL, August 2009, pp. AIAA–2009–5857.

[74] M. A. Santillo, M. S. Holzel, J. B. Hoagg, and D. S. Bernstein, "Adaptive Control Using Retrospective Cost Optimization with RLS-Based Estimation for Concurrent Markov-Parameter Updating," in *Proc. Conf. Dec. Contr.*, ser. Shanghai, China, December 2009, pp. 3466–3471.

[75] H. Sane and D. S. Bernstein, "Active Noise Control Using an Acoustic Servovalve," in *Proc. Conf. Dec. Contr.*, ser. Philadelphia, PA, June 1998, pp. 2621–2625.

[76] S. L. Lacy, R. Venugopal, and D. S. Bernstein, "ARMARKOV Adaptive Control of Self-Excited Oscillations of a Ducted Flame," in *Proc. Conf. Dec. Contr.*, ser. Tampa, FL, December 1998, pp. 4527–4528.

[77] ——, "Disturbance Rejection Using Self-Tuning ARMARKOV Adaptive Control with Simultaneous Identification," *IEEE Trans. on Control Systems Technology*, vol. 1, pp. 101–106.

[78] E. Yigit and A. J. Ridley, "Effects of high-latitude thermosphere heating at various scale sizes simulated by a nonhydrostatic global thermosphere-ionosphere model," *J. Atmos. and Sol.-Terr. Phys.*, vol. 73, pp. 592–600, 2010.

[79] R. G. Roble, E. C. Ridley, A. D. Richmond, and R. E. Dickinson, "A coupled thermosphere/ionosphere general circulation model," *Geophys. Res. Lett.*, vol. 15, p. 1325, 1988.

[80] T. J. Fuller-Rowell and D. Rees, "A three-dimensional, time-dependent, global model of the thermosphere," *J. Atmos. Sci.*, vol. 37, p. 2545, 1980.

[81] Y. Deng, A. D. Richmond, A. J. Ridley, and H. L. Liu, "Assessment of the nonhydrostatic effect on the upper atmosphere using a general circulation model (gcm)," *Geophys. Res. Lett.*, vol. 35, 2008.

[82] C. Jablonowski, M. Herzog, P. J. E., R. C. Oehmke, Q. F. Stout, B. V. Leer, and K. G. Powell, "Block-structured Adaptive Grids on the Sphere: Advection Experiments,," *Monthly Weather Review*, vol. 134, pp. 3691–3713, 2006.

[83] A. M. D'Amato, A. J. Ridley, and D. S. Bernstein, "Adaptive Model Refinement for the Ionosphere and Thermosphere," in *NASA CIDU*, ser. Mountain View, CA, October 2010.

[84] A. M. D'Amato, S. L. Kukreja, and D. S. Bernstein, "Data-Based Model Refinement Using Retrospective Cost optimization," in *Proc. AIAA Guidance, Navigation and Control Conference*, ser. Toronto, ON, August 2010, pp. AIAA–2010–812 786.

[85] D. S. Bernstein and W. M. Haddad, "Steady-State Kalman Filtering with an H_∞ Error Bound," *Systems and Control Letters*, vol. 12, pp. 9–16, 1989.

[86] R. W. Schunk and A. F. Nagy, *Ionospheres*. Cambridge Press, 2000.

[87] C. A. Barth, K. D. Mankoff, S. M. Bailey, and S. C. Solomon, "Global observations of nitric oxide in the thermosphere," *J. Geophys. Res*, vol. 108, p. 1027, January 2003.

[88] D. R. Marsh, S. C. Solomon, and A. E. Reynolds, "Empirical model of nitric oxide in the lower thermosphere," *J. Geophys. Res*, vol. 109, p. A07301, January 2004.

[89] J. Wang, P. Liu, J. Hicks-Garner, E. Sherman, S. Soukiazian, M. Verbrugge, H. Tataria, J. Musser, and P. Finamore, "Cycle-life model for graphite-LiFePO4 cells," *J. Power Sources*, vol. 196, pp. A260–A271, 2011.

[90] G. Plett, "Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs; Part 1. Background," *J. Power Sources*, vol. 134, pp. 252–261, June 2004.

[91] ——, "Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs; Part 2. Modeling and identification," *J. Power Sources*, vol. 134, pp. 262–2276, June 2004.

[92] ——, "Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs; Part 3. State and parameter estimation," *J. Power Sources*, vol. 134, pp. 277–292, June 2004.

[93] T. Fuller, M. Doyle, and J. Newman, "Modeling of Galvanostaic Charge and Discharge of the Lithium/Polymer/Insertion Cell," *J. Electrochemical Society*, vol. 140, pp. 1526–1533, June 1993.

[94] ——, "Simulation and Optimization of the Dual Lithium Ion Insertion Cell," *J. Electrochemical Society*, vol. 141, pp. 1–10, January 1994.

[95] P. Ramadass, B. Haran, P. M. Gomadam, R. White, and B. N. Popov, "Development of First Principles Capacity Fade Model for Li-Ion Cells," *J. Electrochemical Society*, vol. 151, no. 2, pp. A196–A203, January 2004.

[96] J. Forman, S. Moura, J. L. Stein, and H. K. Fathy, "Genetic Parameter Identification of the Doyle-Fuller-Newman Model from Experimental Cycling of a LiFePO4 Battery," *American Control Conference*, 2011.

[97] S. A. Billings, "Identification of Non-linear Systems - A Survey," *Automatica*, vol. 127, pp. 272–285, June 1980.

[98] A. Hagenblad, L. Ljung, and A. Wills, "Maximum Likelihood Identification of Wiener Models," *Automatica*, vol. 44, pp. 2697–2705, November 2008.

[99] S. Lacy and D. S. Bernstein, "Identification of FIR Wiener Systems with Unknown, Non-invertible, Polynomial Nonlinearities," *Int. J. Control*, vol. 76, pp. 1500–1507, 2003.

[100] S. Lacy, R. S. Erwin, and D. S. Bernstein, "Identification of Wiener Systems with Known Noninvertible Nonlinearities," *Journal of Dynamic Systems, Measurement, and Control*, vol. 123, pp. 566–571, 2001.

[101] S. Ttterman and H. T. Toivonen, "Support Vector Method for Identification of Wiener Models," *J. Process Contr.*, vol. 19, pp. 1174–1181, 2009.

[102] E. W. Bai, "Decoupling the linear and nonlinear parts in Hammerstein model identification," *Automatica*, vol. 40, pp. 671–676, April 2004.

[103] F. Ding, T. Shi, and T. W. Chen, "Auxiliary model-based least-squares identification methods for Hammerstein output-error systems," *Syst. & Contr. Let.*, vol. 56, pp. 373–380, May 2007.

[104] H. Kwakernaak and R. Sivan, "The Maximally Achievable Accuracy of Linear Optimal Regulators and Linear Optimal Filters," *IEEE Transactions on Automatic Control*, vol. AC-17, pp. 79–86, January 1972.

[105] W. Lan, B. M. Chen, and Z. Ding, "Adaptive estimation and rejection of unknown disturbances through measurement feedback for a class of nonminimum phase nonlinear MIMO systems," *International Conference on Control and Automation*, vol. 2, pp. 685–690, 2005.

[106] Z. Ding, "Adaptive estimation and rejection of unknown sinusoidal disturbances in a class of non-minimum-phase nonlinear systems," *IEE Proceedings - Control Theory and Applications*, vol. 153, pp. 379–386, 2005.

[107] E. Wang, S. H. Wang and P. Dorato, "Observing the states of systems with unmeasurable disturbance," *IEEE Transactions on Automatic Control*, vol. AC-16, pp. 716–717, 1975.

[108] S. Bhattacharyya, "Observer design for linear systems with unknown inputs," *IEEE Transactions on Automatic Control*, vol. 23, pp. 483–484, 1978.

[109] Y. Guan and M. Saif, "A novel approach to the design of unknown input observers," *IEEE Transactions on Automatic Control*, vol. 36, pp. 632–635, 1991.

[110] H. J. Palanthandalam-Madapusi and D. S. Bernstein, "Unbiased minimum-variance filtering for input reconstruction," in *IEEE American Control Conference*, ser. New York, NY, July 2007, pp. 5712–5717.

[111] H. J. Palanthandalam-Madapusi and K. E. Fitch, "Unbiased Minimum-Variance Filtering for Delayed Input Reconstruction," in *IEEE American Control Conference*, ser. San Francisco, CA, July 2011, pp. 4859–4860.

[112] P. K. Kitanidis, "Unbiased minimum-variance linear state estimation," *Automatica*, vol. 23, pp. 775–778, 1986.

[113] M. Darouach and M. Zasadzinski, "Unbiased minimum variance estimation for systems with unknown exogenous inputs," *Automatica*, vol. 33, pp. 717–719, 1997.

[114] C. S. Hsieh, "Robust two-stage Kalman filters for systems with unknown inputs," *IEEE Transactions on Automatic Control*, vol. 45, pp. 2374–2378, December 2000.

[115] S. Gillijns and B. D. Moor, "Unbiased minimum-variance input and state estimation for linear discrete-time systems," *Automatica*, vol. 43, 2007.

[116] I. Yaesh and U. Shaked, "Simplified adaptive estimation," *IEEE Transactions on Automatic Control*, vol. 57, pp. 49–55, January 2008.

[117] J. B. Morozov, A. V. Hoagg and D. S. Bernstein, "Retrospective Adaptive Control of a Planar Multilink Arm with Nonminimum-Phase Zeros," in *Proc. Conf. Dec. Contr.*, ser. Atlanta, GA, December 2010, pp. 3706–3711.

[118] E. W. Bai and S. S. Sastry, "Persistency of excitation, sufficient richness and parameter convergence in discrete-time adaptive control," *Sys. Contr. Lett.*, vol. 6, pp. 153–163, 1985.

[119] D. S. Bayard, "Stable direct adaptive periodic control using only plant order knowledge," *Int. J. Adaptive Contr. Signal Processing*, vol. 10, pp. 551–570, 1996.

[120] B. D. O. Anderson, "Topical Problems of Adaptive Control," in *Proc. European Contr. Conf.*, ser. Atlanta, GA, July 2007, pp. 4997–4998.

[121] K. J. Åström and B. Wittenmark, *Adaptive Control*, 2nd edition ed. Reading, MA: Addison-Wesley, 1995.

[122] A. M. D'Amato and D. S. Bernstein, "Adaptive Forward-Propagation Input Reconstruction for Nonminimum-Phase Systems," in *Proc. Amer. Contr. Conf.*, Montreal, QC, June 2012, p. submitted.

[123] K. Narendra and A. Annaswamy, *Stable Adaptive Systems*. Prentice-Hall, 1989.

[124] K. Åström and B. Wittenmark, *Adaptive Control*. Addison-Wesley, 1995.

[125] C. Mracek and D. Ridgely, "Missile Longitudinal Autopilots: Connections between Optimal Control and Classical Topologies," in *Proc. AIAA Guidance, Navigation and Control Conference*, San Francisco, CA, August 2005, pp. AIAA 2005–6381.

[126] C. Knospe, "PID Control: Introduction to the Special Section," *IEEE Contr. Sys. Mag.*, vol. 26, January 2006.

[127] K. J. Astrom and T. Hagglund, *Advanced PID Control*. ISA, 2005.

[128] A. Visioli, *Practical PID Control*. Springer, 2010.

[129] M. A. Johnson and M. H. Moradi, *PID Control: New Identification and Design Methods*. Springer, 2005.

[130] E. Poulin, A. Pomerleau, A. Desbiens, and D. Hodouin, "Development and Evaluation of an Auto-tuning and Adaptive PID Controller," *Automatica*, vol. 32, pp. 71–82, January 1995.

[131] V. Bobal, J. Bohm, J. Fessi, and J. Machacek, *Digital Self-tuning Controllers*. Springer, 2005.

[132] C. C. Yu, *Autotuning of PID Controllers: A Relay Feedback Approach*. Springer, 2010.

[133] M. S. Fledderjohn, M. S. Holzel, H. Palanthandalam-Madapusi, R. J. Fuentes, and D. S. Bernstein, "A Comparison of Least Squares Algorithms for Estimating Markov Parameters," in *Proc. Amer. Contr. Conf.*, Baltimore, MD, June 2010, pp. 3735–3740.

[134] A. H. Glattfelder and W. Schaufelberger, *Control Systems with Input and Output Constraints*. Springer, 2003.

[135] L. Zaccarian and A. R. Teel, *Modern Anti-windup Synthesis: Control Augmentation for Actuator Saturation*. Princeton, 2011.

[136] S. Tarbouriech, G. Garcia, J. M. G. da Silva, and I. Queinnec, *Stability and Stabilization for Linear Systems with Saturating Actuators*. Springer, 2011.

[137] B. C. Coffer, J. B. Hoagg, and D. S. Bernstein, "Cumulative Retrospective Cost Adaptive Control with Amplitude and Rate Saturation," in *Proc. Amer. Contr. Conf.*, San Francisco, CA, June 2011, pp. 2344–2349.