MODEL FREE SUBSPACE BASED $\mathcal{H}_\infty$ CONTROL

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL ENGINEERING
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Bruce R. Woodley
January 2001

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Jonathan P. How
(Principal Adviser)

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Stephen P. Boyd

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Antony C. Fraser-Smith

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Stephen M. Rock

Approved for the University Committee on Graduate Studies:

Abstract

Plant knowledge is essential to the development of feedback control laws for dynamic systems. In cases where the plant is difficult or expensive to model from first principles, experimental data are often used to obtain plant knowledge. There are two major approaches for control design incorporating experimental data: model identification/model based control design, and model free (direct) control design. This work addresses the direct control design problem.

The general model free control design problem requires the engineer to collect experimental data, choose a performance objective, and choose a noise and/or uncertainty model. With these design choices, it is then possible to calculate a control law that optimizes expected future performance. Recently, there has been significant interest in developing a direct control design methodology that explicitly accounts for the uncertainty present in the experimental data, thereby producing a more reliable and automated control design technique.

This research exploits subspace prediction methods in order to develop a novel direct control design technique which explicitly allows the inclusion of plant uncertainty. The control design technique is known as model free subspace based $\mathcal{H}_\infty$ control. The new control law can be viewed as a method of "predictive control" similar to the $\mathcal{H}_2$ based "generalized predictive control" (GPC), or the new control law can be viewed as an extension of model free subspace based linear quadratic Gaussian (LQG) control. The ability to easily include plant uncertainty differentiates the model free subspace based $\mathcal{H}_\infty$ technique.

A computationally efficient method of updating model free subspace based controllers is derived, thereby enabling on-line adaptation. This implementation method is particularly effective because the computational effort required to incorporate new data is invariant with respect to the total amount of data collected.

The $\mathcal{H}_\infty$ design technique is demonstrated through a number of laboratory experiments

utilizing a flexible structure. High performance control is experimentally demonstrated for a non-collocated control problem using a very short "identification" data set. In simulation, the adaptive model free subspace based $\mathcal{H}_\infty$ technique is found to rapidly develop an excellent control law after just a few seconds of system operation.

To Annie Nicholls,
who saw my dream of the stars,
encouraged it,
and still expects me to realize it.

Acknowledgments

I would first like to thank my primary advisor, Prof. Jonathan How, for his excellent technical guidance. His patience while watching me wander down all the "blind alleys" was greatly appreciated. A special thank you must also go to Dr. Robert Kosut, who volunteered his time to play an important role in helping me develop my ideas, and who always brought great wisdom and enthusiasm to my work.

Thank you to Prof. Cannon for inviting me into the Aerospace Robotics Lab (ARL), and giving me my start on the helicopter project. That work, which unfortunately did not become part of my thesis, did however give me exposure to the aerospace world: a broadening of my education that I greatly value. Thank you to Prof. Rock who drove the "helicopter team" onto bigger and greater challenges, and who has taught me much about managing technical teams and about engineering in the real world.

Thank you also to the remaining members of my committee, Prof. Stephen Boyd and Prof. Antony Fraser-Smith for their time, good humor, and excellent teaching. I would like to acknowledge the support of Prof. Bernard Widrow who gave me my start in the Ph.D. program here at Stanford: his help came at a critical time for me, and is much appreciated.

Life has run smoothly because of the great efforts of Jane Lintott - thank you! For Godwin Zhang, who has been electrical designer, builder, and rapid bug fixer extraordinaire, I would like to quote Godwin's Law: "90% of electrical problems are caused by lack of power, the rest are caused by too much power!". Thank you to Aldo Rossi, who has taught me much about machining, and even more about people, life, and how to live with honor and good humor. Thank you to Gad Shelef for all his beautiful work on the helicopter. The helicopter would have never flown without Dr. Steve Morris's excellent piloting and maintenance.

Thank you to Brad Betts for his great help in our early years together at Stanford, especially during the Ph.D. qualifying exams. To Alex (Luebke) Stand, for the good times together. Thanks to Chad Jennings for the camping and moral support. Late nights will

not be the same without the great debates and laughter of Adrian Butscher.

Many students in the lab have been extremely helpful throughout my time at Stanford, including Gordon Hunt for all his work on software drivers, and Andrew Conway, Eric Frew, Hank Jones, and Ed LeMaster for their work on the helicopter. Thank you to all the past and present students in the ARL who have made it a great place to work: Chris Clarke, Tobé Corazzini, Steve Fleischer, Andreas Huster, Steve Ims, Bob Kindel, Kortney Leabourne, Rick Marks, Tim McLain, Dave Miles, Eric Miles, Jorge Moraleda, Denny Morse, Mel Ni, Jeff Ota, Gerardo Pardo-Castellote, Chad Partridge, Eric Prigge, Jason Rife, Andrew Robertson, Jeff Russakow, Stef Sonck-Thiebaut, H.D. Stevens, Howard Wang, Ed Wilson, and Kurt Zimmerman.

There are many non-ARL people who have helped me through conversations about this work, including Haitham Hindi, Gokhan Inalhan, Miguel Lobo, Tom Paré, and Bijan Sayyar-Rodsari.

Thank you to my immediate family Bob and Ruth Woodley, and Douglas Woodley, who have taught me much about commitment, persistence, and "do-it-right-or-not-at-all" as a mantra by which to live.

Finally, thank you to Heidi Schubert who has been many different things to this process, including lab mate and colleague, supporter, critic, advisor, coach, cheerleader, camping partner, roommate, and above all, my very best friend.

Contents

List of Figures

Chapter 1

Introduction

This dissertation presents a new method of control synthesis which combines the functions of traditional system identification with that of control design, enabling synthesis of $\mathcal{H}_\infty$ optimal controllers in a single "model free" process. The method utilizes ideas from the recently developed field of subspace identification in order to reduce vast amounts of experimental data to a much smaller "subspace predictor", which is then applied to develop a control law. The technique is referred to as "model free" because at no time in the process is an explicit model of the plant formulated. In addition, an efficient method of recursively updating the subspace predictor is developed, thereby allowing on-line adaptation of the controller as new experimental data are collected.

1.1 Motivation

In order to effectively control a dynamic system (plant), it is essential to have a good understanding of the plant's input-output behavior. This understanding may be explicitly represented by a plant model, or might be implicitly contained in an experimental data set, a control law that has been tuned based on experiment, or some other structure. Internal stability issues may also require that the plant's internal dynamics be well understood.

Typical methods of gaining this understanding include analytical modeling from first principles (such as Newton's laws of motion), direct measurement of various plant parameters (such as mass), model identification from input-output data, model tuning from input-output data, and control law adaptation from input-output data during closed loop

	Plant Model	No Plant Model
Off-line	Model Based Design	Direct Control Design
On-line	Indirect Adaptive	Direct Adaptive

Table 1.1: Four techniques of using experimental data

operation. The process usually requires many iterations among one or more of these techniques. In addition, techniques such as plant model verification, closed loop analysis, and injection of engineering insight (or conjecture!) may be required to converge upon an acceptable control design.

There are many types of systems where experimental data are particularly valuable in obtaining knowledge of plant behavior. Examples include cases where the plant is difficult or expensive to model, where the plant is time-varying, or where the plant is well modeled but certain parameters must be determined experimentally. Examples of difficult or expensive plants to model include arc furnaces [35, 42] and helicopter rotors [37]. Examples of time-varying plants include engines which wear with time, and satellites which change temperature in low earth orbit [23]. The torsional pendulum that is used in Chapter 6 is an example of a plant that has features that can be well modeled from first principles, yet requires experimental data in order to obtain appropriate model parameters. Methods of using experimental data can roughly be divided into four categories, as shown in Table 1.1. The techniques are distinguished by whether they operate "on-line" or "off-line", and whether a plant model is explicitly used to perform the control design.

Plant model identification is perhaps the most popular method of using experimental data in the control design process. The engineer usually performs a number of experiments, and then uses the experimental data in conjunction with various optimization techniques to form a model of the plant. The plant model is then used with one of the well known model based control design techniques to synthesize a control law. Typical identification techniques include the classical prediction error (PE), auto regressive with exogenous input (ARX), auto regressive moving average with exogenous input (ARMAX), output error (OE), and Box Jenkins [25, 28] techniques. More recently, subspace techniques such as eigensystem realization analysis (ERA) [20] and numerical algorithms for subspace state space system identification (N4SID) [39] have gained popularity. The control design literature is vast, and includes simple proportional integral derivative (PID) as well as more advanced modern and

post modern techniques *e.g.* linear quadratic Gaussian (LQG) [15], and μ-synthesis [44].

When model based design is used "on-line", it is usually referred to as indirect adaptive control [3]. The process typically begins by assuming a nominal plant model. As new experimental data are collected, the outputs are compared to the outputs predicted by the nominal plant model, producing a nominal error. The gradient of the error with respect to the plant parameters is used to modify the plant parameters to improve the plant model. Periodically, the control law is updated using the most recently developed plant model as the basis for control synthesis. An example of this approach is model reference adaptive control [3].

The "no plant model" column in Table 1.1 is somewhat of a misnomer: in some sense, a data set can be considered an empirical plant model, thus any simplified representation of the data set is also a plant model. The defining feature of model free techniques is that a single integrated procedure derives the control law directly from experimental data and a performance specification. If the technique has a sufficiently low computational burden, it is generally straightforward to implement the "no plant model" design technique on-line. This produces a "direct adaptive" control technique where the controller attempts to improves its performance in response to newly available experimental data. Examples of direct control techniques include model free subspace based LQG control [11, 12], adaptive inverse control [41], LMS [40], and FxLMS and its alternatives [34]. The next two subsections describe the features of model free control design and outline conditions under which it might be advantageous to apply model free techniques.

1.1.1 Features Of Model Free Techniques

The most important feature of model free control design techniques is the close coupling of the "plant identification" and the "control design" steps. In traditional model based control design, the development of a plant model requires great simplifications of the experimental data set in order to obtain a plant model. In the model free technique, much more of the experimental information is retained throughout the control design process. If the controller is then simplified, the simplifications that are made are with respect to the controller's input-output relationship, rather than with respect to the plant input-output relationship. The simplifications of the controller are made with respect to what is important to the control law, rather than what is important to the plant model [1, 2, 9].

Closer coupling of the identification and control design process should lead to increased

automation of the control design process, however, this conjecture can only be confirmed by the experiences of control engineers who are able to try both model based and model free techniques in the field. The increased automation is expected to result from the removal of the intermediate design steps, thereby requiring the engineer to make fewer arbitrary choices of parameters during the design process. An additional advantage is realized in the iterative process between designing "identification" experiments and performing closed loop tests: the model free control design process naturally provides the engineer with an immediate estimate of closed loop performance.

Due to the increased automation, model free techniques are easily implemented as part of an adaptive framework. It is believed that model free techniques will be of great utility when solving adaptive control problems.

1.1.2 When One Might Use Model Free Techniques

Control problems with certain attributes are likely to receive the most benefit from model free techniques. These attributes include:

- Experimental data are plentiful, are representative of the important system dynamics, and are inexpensive to obtain.
- Iteration between design and closed loop experiment is possible.
- Some aspects of the system are difficult to quantify by analytic modeling, *e.g.* time-varying nonlinearities.
- The plant has many inputs and many outputs, such that modeling each input-output relationship might be prohibitively tedious.

The adaptive methodologies are of course applicable to cases where the plant is time-varying. In many time-varying problems, the control law adapts to compensate for changing system parameters, such as temperature and drag. However, if the plant *structure* is changing, such as addition of new modes, a model free technique may be better suited than a model based adaptive controller in which the model structure is determined *a priori*.

1.1.3 Model Free Robust Control

One of the difficult problems that remains in the adaptive control community is developing control laws that achieve good closed loop performance when the plant is only partially or

inaccurately represented by the experimental data. This problem has motivated much of the recent work in model free and model based *robust* control techniques. Recent solutions include direct controller *unfalsification* methods [32, 33] and uncertainty model unfalsification methods [22]. The limited success and computational complexity of the unfalsification methods motivated the development of the new model free robust control technique presented in this dissertation.

1.2 Problem Statement And Solution Methodology

A precise statement of the model free robust control problem is as follows: given plant input-output data (u, y), a performance objective J, and an admissible noise and/or uncertainty model, the objective is to find a feedback control law that optimizes the J for future time. Many design choices must be made in order to formulate a solution to this problem. They include:

- Selecting a method of extrapolation of plant behavior into the future.
- Developing a noise and/or uncertainty model.
- Selecting an appropriate performance objective, taking into account the uncertainty model.
- Ensuring that a computationally efficient method of updating the control law exists.

With these issues in mind, the following design choices were made for this thesis.

- **Performance objective:** A finite horizon $\mathcal{H}_\infty$ objective was selected as it can easily account for plant uncertainty in the design specification via the small gain theorem. In the $\mathcal{H}_\infty$ output feedback formulation, the reference signal is considered to be a disturbance thus the control law is designed for the worst case reference signal that could possibly drive the system. The design technique also allows the engineer to specify other "worst case" disturbances that the control law must be able to reject.
- **Extrapolation method and noise model:** The subspace predictor was selected to extrapolate past input-output data in order to predict the future plant outputs. This relatively recent technique has been very successful in producing good results on

various model identification test cases [39]. The subspace predictor implicitly assumes a stochastic noise model in predicting plant outputs from past input-output data. As is shown in Chapter 4, the model free subspace based $\mathcal{H}_\infty$ design technique retains the Kalman filter like qualities of subspace based identification, yet provides the best possible control in the case of "worst case" disturbances that are not considered by the subspace predictor.

- **Uncertainty model:** Application of the robust design technique developed in this thesis requires the engineer to select an uncertainty model when choosing the performance objective.

These design choices result in what has been termed model free subspace based $\mathcal{H}_\infty$ control. In this form, model free subspace based $\mathcal{H}_\infty$ control is a member of a class of controllers known as *predictive control* [10, 16, 26]. Predictive control refers to any technique that employs the following steps:

1. A *predictor* is used to determine the plant input that will optimize a specified cost function over a future time horizon.
2. The first time step of the control is implemented. The plant output at this time step is recorded.
3. The new input-output data are added to the predictor, and steps 1–3 are repeated.

These steps are collectively known as a "receding horizon" implementation. A popular form of prediction control is the so-called "generalized prediction control" (GPC) which employs a quadratic cost function and an "auto-regressive integrated moving average with exogenous input" (ARIMAX) predictor [3, 7, 8, 10]. Predictive control has been applied with $\mathcal{H}_\infty$ optimal predictors and $\mathcal{H}_2$ cost functions [19], $\mathcal{H}_\infty$ optimal predictors and $\mathcal{H}_\infty$ control costs [43], and mixed $\mathcal{H}_2/\mathcal{H}_\infty$ minimax predictors [38]. Subspace predictors have previously been applied to the predictive control problem with quadratic costs [11], resulting in model free subspace based LQG control. One major contribution of this thesis is that it extends model free *subspace based* predictive control to include $\mathcal{H}_\infty$ cost functions.

1.2.1 Prediction Control Issues

Several issues are often raised in the discussion of predictive control: how does one choose the length of the prediction horizon, what stability guarantees exist, and why is only one

time step of the control implemented before updating the predictor?

Unfortunately, good answers to the first two questions do not exist at this time. Intuitively, one would expect that the longer the prediction horizon, the greater the probability that the closed loop system is stable. This conjecture is supported by the experiments and the design example in Chapter 5. It is sometimes possible to derive sufficient conditions for a *model based* receding horizon control law to be closed loop stable when the model is an exact representation of the plant, *e.g.* the LQG laws in [3]. However, when real experimental data are used, it is very difficult to find conditions under which closed loop stability can be guaranteed [3]. The choice of control horizon length is thus a design parameter which must be selected by the engineer.

The issue of implementing only one time step of the control law before updating the predictor can be understood by comparing this strategy to the strategy of implementing two control time steps before updating the predictor. If two time steps are implemented before the predictor is updated, then the control law is effectively ignoring input-output data that are available after the first time step. In a system with stationary noises, a controller that uses the newest data to estimate the present system state will outperform an "ignorant" controller that relies exclusively on older data.

1.2.2 Comparison To Subspace Identification/$\mathcal{H}_\infty$ Design

An obvious question that arises when evaluating the model free subspace based $\mathcal{H}_\infty$ control technique is how does the technique differ from using a subspace identification technique (such as N4SID) to develop a linear time-invariant (LTI) plant model, then using the plant model to perform a model based $\mathcal{H}_\infty$ control design? Several difference are apparent.

A subspace identification technique such as N4SID has, as its first step, the development of a subspace predictor identical to the one used in this thesis. The technique then extracts an *infinite horizon* plant model from the subspace predictor. The approximation being made by the N4SID method is that the Markov parameters of the finite horizon subspace predictor can be effectively used to extract an infinite horizon LTI plant model.

Furthermore, the N4SID method extracts a *low order* LTI model from the subspace predictor. There are many more degrees of freedom in the subspace predictor than the number of degrees of freedom in the LTI system that is used to approximate it. The N4SID method permits the use of a Hankel singular value plot to aid the engineer in selecting an appropriate model order for the LTI system. The typical difficulty with this reduction

process is determining how many Hankel singular values should be retained: the choice is often not obvious due to the blurring between plant dynamics and system noise. This *ad hoc* process can lead to the elimination of large amount of information that might be significant.

The model free subspace based technique retains all of the information in the subspace predictor. It does not attempt to extrapolate plant input-output beyond the time horizon that the subspace predictor is designed to predict plant input-output behavior. There is no need to choose a model order for the extraction of an LTI model from the subspace predictor.

The control design for the model free subspace based $\mathcal{H}_\infty$ controller is based on a finite horizon cost function: the model based control design typically uses an infinite horizon cost. In the model free case, it is assumed that the prediction horizon is long enough to produce a stable closed loop system. In the model based control design, it is assumed that the infinite horizon plant model is a sufficiently accurate approximation of the subspace predictor *and* that the prediction horizon is long enough to produce a stable closed loop system.

Thus the fundamental differences between these two control design techniques are the types of approximations that are being made. In the model free technique, it is directly assumed that the finite horizon predictor contains sufficient information to capture the behavior of the system in the closed loop. In the model based approach, large amounts of information in the subspace predictor are discarded in order to obtain a low order infinite horizon model. The low order model is necessary in order to make the infinite horizon $\mathcal{H}_\infty$ control design tractable. The close coupling between experimental data and control design in the model free technique has the potential to significantly improve the end-to-end performance of the "data to controller" design process.

1.3 Contributions

The following contributions to the control literature are presented in this dissertation.

1. A novel method of designing a mixed subspace/$\mathcal{H}_\infty$ control law directly from experimental data is developed. Since a plant model is not explicitly formed, the technique is known as model free subspace based $\mathcal{H}_\infty$ control. This technique is also applicable to model free subspace based robust $\mathcal{H}_\infty$ control design.

2. A computationally efficient algorithm for adaptively implementing the model free subspace based $\mathcal{H}_\infty$ control design is developed. This algorithm is also applicable to the adaptive implementation of the model free subspace based linear quadratic Gaussian (LQG) control technique.

3. It is shown that the receding horizon implementation of the model free subspace based $\mathcal{H}_\infty$ control law is equivalent to implementing a discrete time LTI dynamic feedback controller. This observation greatly simplifies controller implementation, enables the engineer to use the familiar tools of LTI analysis, and permits the engineer to use familiar model order reduction tools to simplify the controller complexity. These observations are also applicable to the model free subspace based LQG control law. This observation is nearly identical to one that has been made for GPC controllers [3].

4. Key properties of the model free subspace based $\mathcal{H}_\infty$ control law are analyzed, demonstrating that the control law has the limiting behavior that one would expect from such a control law. Specifically:

   (a) As the design parameter $\gamma \to \infty$, the model free subspace based $\mathcal{H}_\infty$ control law converges to the model free subspace based LQG control law.

   (b) As the past data length becomes very large, the model free subspace based $\mathcal{H}_\infty$ control law behaves like a Kalman filter coupled to a full information finite horizon $\mathcal{H}_\infty$ controller.

5. The model free subspace based $\mathcal{H}_\infty$ control design technique is demonstrated experimentally on a torsional spring/mass system. The good performance of the model free subspace based $\mathcal{H}_\infty$ control design technique on this challenging control problem serves as a "proof of concept" for its future deployment in real world applications. Furthermore, the adaptive technique is demonstrated in simulation using a noisy LTI model of the same torsional spring/mass system. The control law begins with a very poor knowledge of the spring/mass system, and uses the input-output information obtained during closed loop operation in order to obtain a satisfactory control law.

6. Using the model unfalsification concept, a convex method of developing LTI uncertainty models directly from the subspace predictor is derived. This result extends

the ARX uncertainty model unfalsification technique [21]. Although the convex technique is not computationally feasible at this time, this work establishes a connection between the ARX uncertainty model and the subspace uncertainty model. Should an efficient method of computing the ARX uncertainty model be developed, then the method would be immediately applicable to the subspace uncertainty model.

1.4 Thesis Outline

Chapter 2 reviews the ideas of subspace prediction developed in [39]. A computationally efficient algorithm for determining the optimal subspace predictor is also developed in Chapter 2. Chapter 3 derives the model free subspace based $\mathcal{H}_\infty$ optimal controller. Both the strictly causal controller (where the reference signal is not known at the present time) and the causal controller (where the reference signal is known at the present time) are derived. Robust $\mathcal{H}_\infty$ control laws, which take into account multiplicative or additive uncertainties, are also derived.

Chapter 4 analyzes the limiting cases of the model free subspace based $\mathcal{H}_\infty$ controller, which provides some insight as to the behavior of the new control design technique. Also included is a short derivation of the model free subspace based LQG control law. Chapter 5 details both the batch and adaptive implementation of the control design technique. All of the details in Chapter 5 are also applicable to the LQG control design technique. A design example illustrates the use of the model free subspace based $\mathcal{H}_\infty$ technique, and demonstrates the trades that the engineer must make when using this design tool.

Chapter 6 presents experimental results which demonstrate the model free subspace based $\mathcal{H}_\infty$ control design technique on a physical system. Chapter 7 describes the use of the model unfalsification concept in order to derive an LTI uncertainty model from the experimental data. This uncertainty model can be used in conjunction with the model free subspace based robust $\mathcal{H}_\infty$ control design technique to produce a robust control design directly from data. Unfortunately, this technique remains too computationally expensive for practical application at this time. Chapter 8 summarizes the thesis, and provides suggestions for future work.

Chapter 2

Subspace Prediction

The subspace system identification methods developed in [39] have recently gained much popularity for identification of linear time-invariant (LTI) systems. The technique is employed in a two step process: first, the best least squares subspace predictor is derived from available experimental data; second, the predictor is used to derive a state space model of the dynamic system. Each of these steps significantly reduces the volume of information used to represent the plant input-output behavior. The formation of the subspace predictor serves two purposes: it simultaneously reduces the effects of noise in the measured data, and it establishes a method of extrapolating future plant input-output behavior from past input-output data. Since the subspace predictor can be shown to be equivalent to a very high order LTI plant model, the derivation of the state space model from the predictor is similar to a plant model order reduction.

The key idea established in [11, 12] is that LQG control design can be performed directly from the subspace predictor, avoiding the "model formation" step. This thesis extends this concept to include $\mathcal{H}_\infty$ performance specifications. In both cases, all of the information that remains after the development of the subspace predictor is used to derive a control law. This chapter describes the subspace predictor, various methods of computing the predictor from experimental data, and in the case of an LTI plant, the relationship between the predictor and the true plant.

2.1 Least Squares Subspace Predictor

This section presents a review of the subspace predictor concept [39]. The development of the predictor begins with experimental input-output data. Consider input-output data of length n from a plant with m inputs ($u_k \in \mathbb{R}^m$) and l outputs ($y_k \in \mathbb{R}^l$)

$$\left(\begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{n-1} \end{bmatrix}, \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{n-1} \end{bmatrix} \right)$$

The engineer then chooses a prediction horizon, i. i should be chosen to be larger than the expected order of the plant (if the plant is LTI), and is usually chosen to be 2 or 3 times larger than the expected plant order [39]. The data set is then broken into j prediction problems, where $j = n - 2i + 1$. Usually there is a relatively large amount of data, so that $j \gg i$. The goal is to find a single predictor that simultaneously optimizes (in the least squares sense) the j prediction problems. Figure 2.1 illustrates two of these prediction problems, with $i = 4$. In the first problem, the input data $\{u_0, \cdots, u_{2i-1}\}$ plus the output data $\{y_0, \cdots, y_{i-1}\}$ are used to predict $\{y_i, \cdots, y_{2i-1}\}$. The second prediction problem uses $\{u_1, \cdots, u_{2i}\}$ and $\{y_1, \cdots, y_i\}$ to predict $\{y_{i+1}, \cdots, y_{2i}\}$. Sliding the prediction windows to the right through the length n data set results in $j = n - 2i + 1$ prediction problems.

The j prediction problems can be formalized as follows. Define the block Hankel matrices formed from the data

$$U_p \triangleq \begin{bmatrix} u_0 & u_1 & \cdots & u_{j-1} \\ u_1 & u_2 & \cdots & u_j \\ \vdots & \vdots & \cdots & \vdots \\ u_{i-1} & u_i & \cdots & u_{i+j-2} \end{bmatrix} \in \mathbb{R}^{im \times j} \tag{2.1}$$

$$U_f \triangleq \begin{bmatrix} u_i & u_{i+1} & \cdots & u_{i+j-1} \\ u_{i+1} & u_{i+2} & \cdots & u_{i+j} \\ \vdots & \vdots & \cdots & \vdots \\ u_{2i-1} & u_{2i} & \cdots & u_{2i+j-2} \end{bmatrix} \in \mathbb{R}^{im \times j} \tag{2.2}$$

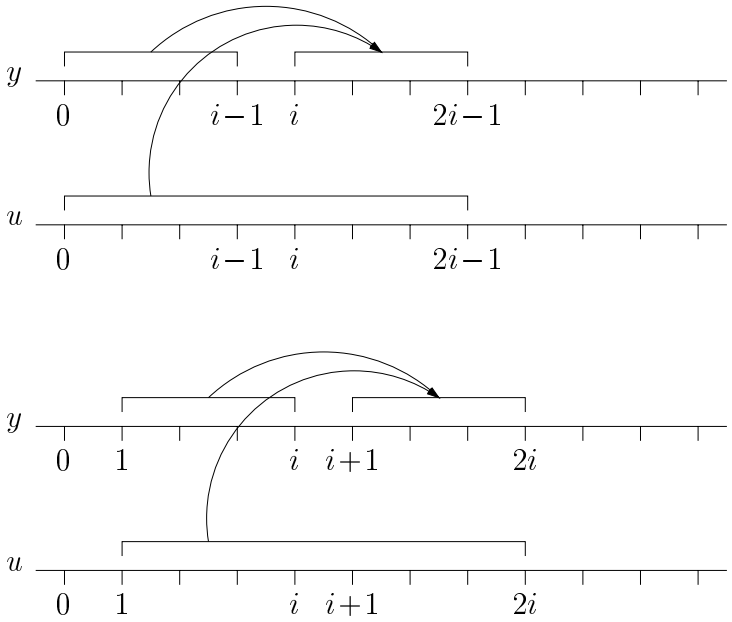

Figure 2.1: Subspace predictor illustration, $i = 4$

Two of the j prediction problems are shown. The predictor is determined by choosing L_w and L_u to simultaneously optimize the predictions indicated by the arrows.

$$Y_p \triangleq \begin{bmatrix} y_0 & y_1 & \cdots & y_{j-1} \\ y_1 & y_2 & \cdots & y_j \\ \vdots & \vdots & \cdots & \vdots \\ y_{i-1} & y_i & \cdots & y_{i+j-2} \end{bmatrix} \in \mathbb{R}^{il \times j} \tag{2.3}$$

$$Y_f \triangleq \begin{bmatrix} y_i & y_{i+1} & \cdots & y_{i+j-1} \\ y_{i+1} & y_{i+2} & \cdots & y_{i+j} \\ \vdots & \vdots & \cdots & \vdots \\ y_{2i-1} & y_{2i} & \cdots & y_{2i+j-2} \end{bmatrix} \in \mathbb{R}^{il \times j} \tag{2.4}$$

The subscripts "p" and "f" can be thought of as representing "past" and "future" time. Also define

$$W_p \triangleq \begin{bmatrix} U_p \\ Y_p \end{bmatrix}$$

so that W_p represents all the "past" data.

The problem of obtaining the best linear least squares predictor of Y_f, given W_p and U_f can be written as a Frobenius norm minimization

$$\min_{L_w, L_u} \left\| Y_f - \begin{bmatrix} L_w & L_u \end{bmatrix} \begin{bmatrix} W_p \\ U_f \end{bmatrix} \right\|_F^2 \tag{2.5}$$

It is shown in [39] that the solution to this problem is given by the orthogonal projection, (represented by the symbol /) of the row space of Y_f into the row space spanned by W_p and U_f. It is this geometry that inspires the name "subspace predictor". The orthogonal projection solution to the problem (2.5) is given by

$$\begin{aligned} \widehat{Y}_f &= Y_f \Big/ \begin{bmatrix} W_p \\ U_f \end{bmatrix} \\ &\triangleq Y_f \begin{bmatrix} W_p \\ U_f \end{bmatrix}^T \left(\begin{bmatrix} W_p \\ U_f \end{bmatrix} \begin{bmatrix} W_p \\ U_f \end{bmatrix}^T \right)^\dagger \begin{bmatrix} W_p \\ U_f \end{bmatrix} \end{aligned}$$

where † denotes the Moore-Penrose or pseudoinverse. Thus

$$\begin{bmatrix} L_w & L_u \end{bmatrix} = Y_f \begin{bmatrix} W_p \\ U_f \end{bmatrix}^T \left(\begin{bmatrix} W_p \\ U_f \end{bmatrix} \begin{bmatrix} W_p \\ U_f \end{bmatrix}^T \right)^\dagger$$

where

$$L_w \in \mathbb{R}^{il \times i(l+m)} \qquad L_u \in \mathbb{R}^{il \times im}$$

Consider k to be the present time index. Given past experimental data

$$\left(\begin{bmatrix} u_{k-i} \\ \vdots \\ u_{k-1} \end{bmatrix}, \begin{bmatrix} y_{k-i} \\ \vdots \\ y_{k-1} \end{bmatrix} \right)$$

and future inputs

$$\begin{bmatrix} u_k \\ \vdots \\ u_{k+i-1} \end{bmatrix}$$

L_w and L_u can be used to form an estimate of future outputs, namely

$$\begin{bmatrix} \widehat{y}_k \\ \vdots \\ \widehat{y}_{k+i-1} \end{bmatrix} = L_w \begin{bmatrix} u_{k-i} \\ \vdots \\ u_{k-1} \\ y_{k-i} \\ \vdots \\ y_{k-1} \end{bmatrix} + L_u \begin{bmatrix} u_k \\ \vdots \\ u_{k+i-1} \end{bmatrix} \tag{2.6}$$

Equation (2.6) will be used extensively throughout this thesis in order to extrapolate future plant input-output behavior from past input-output data.

2.2 Calculating L_w, L_u, QR Method

The QR decomposition forms the basis of a computationally efficient and numerically reliable method of finding L_w and L_u. First calculate the QR decomposition

$$\begin{bmatrix} W_p \\ U_f \\ Y_f \end{bmatrix} = R^T Q^T = \begin{bmatrix} R_{11} & 0 & 0 \\ R_{21} & R_{22} & 0 \\ R_{31} & R_{32} & R_{33} \end{bmatrix} Q^T \tag{2.7}$$

then

$$\begin{bmatrix} L_w & L_u \end{bmatrix} = \begin{bmatrix} R_{31} & R_{32} \end{bmatrix} \begin{bmatrix} R_{11} & 0 \\ R_{21} & R_{22} \end{bmatrix}^{\dagger}$$

The pseudoinverse is usually calculated using the singular value decomposition (SVD). Compute the SVD

$$\begin{bmatrix} R_{11} & 0 \\ R_{21} & R_{22} \end{bmatrix} = U \Sigma V^T$$

with U and V orthogonal, and the matrix of singular values

$$\Sigma = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_{i(l+2m)} \end{bmatrix}$$

with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{i(l+2m)} \geq 0$. In order to compute the matrix inverse without encountering singularities, the directions of U and V associated with σ's that are close to zero are discarded. Thus choose $q \leq i(l+2m)$ such that $\sigma_r > tol \; \forall \; 1 \leq r \leq q$, where tol is some very small positive tolerance. Let U_q and V_q represent the matrices formed from the first q columns of the matrices U and V respectively. Then

$$\begin{bmatrix} R_{11} & 0 \\ R_{21} & R_{22} \end{bmatrix}^{\dagger} \triangleq V_q \begin{bmatrix} \sigma_1^{-1} & 0 & \cdots & 0 \\ 0 & \sigma_2^{-1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_q^{-1} \end{bmatrix} U_q^T$$

and

$$\begin{bmatrix} L_w & L_u \end{bmatrix} = \begin{bmatrix} R_{31} & R_{32} \end{bmatrix} V_q \begin{bmatrix} \sigma_1^{-1} & 0 & \cdots & 0 \\ 0 & \sigma_2^{-1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_q^{-1} \end{bmatrix} U_q^T$$

The QR decomposition has computational complexity $O(i^2 j)$, while the SVD algorithm has complexity $O(i^3)$, producing overall computational complexity $O(i^2 j + i^3)$ [17]. The storage requirement for the QR/SVD algorithm is $O(ij)$.

2.3 A Fast Method Of Calculating L_w, L_u

The QR method of calculating L_w and L_u is computationally inefficient for the typical case where $j \gg i$. As an example of typical i and j, the experiment in Section 6.2 uses $i = 64$ and $j = 2048$. Much effort goes into computing the very large matrix $Q^T \in \mathbb{R}^{2i(l+m)\times j}$, which is then not used in the calculation of $\begin{bmatrix} L_w & L_u \end{bmatrix}$. A more efficient computational method exists based on the Cholesky factorization. Let

$$A \triangleq \begin{bmatrix} W_p \\ U_f \\ Y_f \end{bmatrix} \in \mathbb{R}^{2i(l+m)\times j}$$

then using (2.7)

$$AA^T = R^T Q^T Q R = R^T R \in \mathbb{R}^{2i(l+m)\times 2i(l+m)}$$

thus

$$R = chol(AA^T)$$

where $chol(AA^T)$ is the Cholesky factorization of AA^T. Since AA^T is much smaller than A when $j \gg i$, the Cholesky factorization of AA^T can take significantly fewer computations to perform that the QR decomposition of A. The Cholesky factorization is in fact $O(i^3)$ [17], however, one must account for the computation of AA^T.

The "brute force" method of computing AA^T is $O(i^2 j)$. Fortunately, the Hankel structure of A can be exploited to drastically reduce the computational effort. Let

$$U \triangleq \begin{bmatrix} U_p \\ U_f \end{bmatrix} \qquad Y \triangleq \begin{bmatrix} Y_p \\ Y_f \end{bmatrix} \qquad B \triangleq \begin{bmatrix} U_p \\ U_f \\ Y_p \\ Y_f \end{bmatrix} = \begin{bmatrix} U \\ Y \end{bmatrix}$$

Referring to (2.1) – (2.4), it is clear that both $U \in \mathbb{R}^{2im \times j}$ and $Y \in \mathbb{R}^{2il \times j}$ have Hankel structure. By inspection, A is a permutation of B, and thus AA^T is a permutation of BB^T: computing

$$BB^T = \begin{bmatrix} UU^T & UY^T \\ YU^T & YY^T \end{bmatrix}$$

is equivalent to computing AA^T. Since BB^T is symmetric, it is sufficient to compute only the block upper triangle of UU^T and YY^T, and to compute only UY^T.

To find UU^T, first compute the first block row of UU^T. This process is $O(ij)$. The block Hankel structure of U can be exploited in order to compute successive entries of the upper triangle of UU^T via the recursion

$$[UU^T]_{r,s} = [UU^T]_{r-1,s-1} - [U]_{r-1,1}[U]_{s-1,1}^T + [U]_{r,j}[U]_{s,j}^T \qquad (2.8)$$
$$\forall \quad 2i \geq s \geq r \geq 2$$

where $[\bullet]_{r,s}$ represents the block in the r^{th} block row and the s^{th} block column of $\bullet$. The intuitive interpretation of (2.8) is that each new entry of UU^T can be written as a function of the entry above and to the left of itself. Also note that many of the outer products on the right-hand side of (2.8) have already been performed in the computation of entries in the first block row of UU^T, and that these outer products are themselves symmetric, further

	QR	Cholesky
Floating Point Operations	3.2×10^8	3.4×10^5
Memory (Bytes)	1.08×10^6	1.4×10^4

Table 2.1: Example computation of R, QR vs. Cholesky method

simplifying the recursion.

The computation of the remaining $i(2i-1)$ unique block entries of UU^T is thus $O(i^2)$, and the overall computation of UU^T is $O(ij+i^2)$. An identical procedure applies to the computation of YY^T. Similarly, UY^T is found by computing the first block row *and the first block column* of UY^T. This process is also $O(ij)$. Exploiting the block Hankel structure of U and Y^T, successive entries can be computed via the recursion

$$[UY^T]_{r,s} = [UY^T]_{r-1,s-1} - [U]_{r-1,1}[Y]^T_{s-1,1} + [U]_{r,j}[Y]^T_{s,j} \qquad (2.9)$$
$$\forall \quad 2i \geq r \geq 2 \qquad \forall \quad 2i \geq s \geq 2$$

which is identical to (2.8), except that the ranges of r and s have been expanded. Thus the computation of BB^T is $O(ij+i^2)$, and it follows that the computation of AA^T is $O(ij+i^2)$.

Combining the above results, the Cholesky factorization/SVD method has an overall computational complexity $O(ij+i^3)$, which is less that the QR/SVD algorithm, which is $O(i^2j+i^3)$. In the typical case where $j \gg i$, this difference is significant.

An additional advantage of the Cholesky method is the reduced storage requirements. The recursive method of computing BB^T does not require the complete formation of U or Y. Since any block $i \times j$ Hankel matrix can be represented by a block vector of length $i+j-1$, the storage requirements of the Cholesky method is $O(i^2+j)$, which is significantly less than the QR storage requirement, which is $O(ij)$ [17].

Table 2.1 compares the resources required for a computation of R via the QR and Cholesky methods, where $i = 20$ and $j = 1000$. The computations are performed using Matlab, thus the statistics should be regarded as Matlab specific. Clearly the Cholesky method, which exploits the Hankel structure of the problem, reduces the resources required to find R.

2.3.1 Fast Updating And Downdating Of L_w, L_u

The Cholesky method also allows the rapid addition of new experimental data and the rapid removal of old experimental data. The computational details associated with performing these updates and downdates are quite complicated: an excellent discussion can be found in [17]. An important feature of updating and downdating the Cholesky factorization of AA^T is that the storage requirement is $O(i^2)$, *which is independent of* j. The independence is especially powerful if j is increasing as new experimental data are collected.

Consider new input-output data (u_n, y_n). This new data will permit an additional column to be appended to A forming $A_{new} = \begin{bmatrix} A & a_{new} \end{bmatrix}$. Then

$$\begin{aligned} R_{new} &= chol(A_{new}A_{new}^T) = chol(AA^T + a_{new}a_{new}^T) \\ &\triangleq cholupdate(R, a_{new}) \end{aligned}$$

where *cholupdate* is the rank-1 update of the Cholesky factorization. The rank-1 update can be performed extremely quickly, and has computational complexity $O(i^2)$. The storage requirement to store the "old" data points in order to form a_{new} is $O(i)$, while the storage requirement to retain R is $O(i^2)$.

It is also possible to remove "old" experimental data using a rank-1 Cholesky downdate

$$\begin{aligned} R_{new} &= chol(AA^T - a_{old}a_{old}^T) \\ &\triangleq choldowndate(R, a_{old}) \end{aligned}$$

This technique also has computational complexity $O(i^2)$ [17]. Combining the update and downdate techniques enables "sliding window" adaptation, where at each time step, the oldest experimental data are removed from R, and new data are added. The sliding window procedure requires the storage of experimental data over the whole "sliding window" in order to enable the removal of experimental data.

In both the strictly updating and the sliding window adaptive techniques, the matrix R plays the role of maintaining a summary of all past information in order to update the predictor as new experimental data becomes available. At each time step, new data are used to update R, then L_w and L_u are computed from R using the SVD.

2.4 Relation To The Exact Plant Model

If the plant is linear time-invariant, the plant input is persistently exciting, the process noise ζ_k and sensor noise η_k are white, and the data length $j \to \infty$, it can be shown [11, 12, 39] that the matrices L_w and L_u become closely related to the exact plant model

$$\begin{aligned} x_{k+1} &= A_p x_k + B_p u_k + \zeta_k \\ y_k &= C_p x_k + D_p u_k + \eta_k \end{aligned}$$

where ζ_k and η_k are zero-mean, stationary, ergodic, white noise sequences with covariance

$$E\left[\begin{bmatrix} \zeta_r \\ \eta_r \end{bmatrix} \begin{bmatrix} \zeta_s^T & \eta_s^T \end{bmatrix}\right] = \begin{bmatrix} Q_p & S_p \\ S_p^T & R_p \end{bmatrix} \delta_{rs}$$

where δ_{rs} is the discrete time impulse function. Note that $A_p \in \mathbb{R}^{n_p \times n_p}$, $B_p \in \mathbb{R}^{n_p \times m}$, $C_p \in \mathbb{R}^{l \times n_p}$, $D_p \in \mathbb{R}^{l \times m}$, $\zeta_k \in \mathbb{R}^{n_p}$, and $\eta_k \in \mathbb{R}^l$. Under these conditions, the matrix L_u converges to a Toeplitz matrix (H_p) formed from the Markov parameters of the plant.

$$j \to \infty \quad \Rightarrow \quad L_u \to H_p \triangleq \begin{bmatrix} D_p & 0 & 0 & \cdots & 0 \\ C_p B_p & D_p & 0 & \cdots & 0 \\ C_p A_p B_p & C_p B_p & D_p & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ C_p A_p^{i-2} B_p & C_p A_p^{i-3} B_p & C_p A_p^{i-4} B_p & \cdots & D_p \end{bmatrix} \tag{2.10}$$

Consider the product $L_w W_k$ where

$$W_k \triangleq \begin{bmatrix} \begin{bmatrix} u_{k-i} & u_{k-i+1} & \cdots & u_{k-i+j-1} \\ u_{k-i+1} & u_{k-i+2} & \cdots & u_{k-i+j} \\ \vdots & \vdots & \cdots & \vdots \\ u_{k-1} & u_k & \cdots & u_{k+j-2} \end{bmatrix} \\ \begin{bmatrix} y_{k-i} & y_{k-i+1} & \cdots & y_{k-i+j-1} \\ y_{k-i+1} & y_{k-i+2} & \cdots & y_{k-i+j} \\ \vdots & \vdots & \cdots & \vdots \\ y_{k-1} & y_k & \cdots & y_{k+j-2} \end{bmatrix} \end{bmatrix} \tag{2.11}$$

Then under the same conditions, the product $L_w W_k$ converges to the product of the plant extended observability matrix (Γ_p) and a matrix of state estimates

$$j \to \infty \quad \Rightarrow \quad L_w W_k \to \Gamma_p \widehat{X} = \begin{bmatrix} C_p \\ C_p A_p \\ \vdots \\ C_p A_p^{i-1} \end{bmatrix} \begin{bmatrix} \widehat{x}_k & \widehat{x}_{k+1} & \cdots & \widehat{x}_{k+j-1} \end{bmatrix} \tag{2.12}$$

The state estimates $[\ \widehat{x}_k \quad \widehat{x}_{k+1} \quad \cdots \quad \widehat{x}_{k+j-1}\ ]$ are equivalent to those produced by j independent Kalman filters, each acting on a column of W_k. A more complete discussion can be found in [39]. The underlying idea is that the subspace technique simultaneously recovers the true plant model and Kalman estimates of the plant states. This result will be used in Chapter 4 to relate the model free subspace based $\mathcal{H}_\infty$ control law to the model based $\mathcal{H}_\infty$ control law.

2.5 Summary

The important results of this chapter are:

- A fast method of calculating the subspace predictor L_w, L_u from experimental data;
- A fast method of updating L_w, L_u when new experimental data are available, and/or when old experimental data must be removed;
- An expression for the optimal subspace estimate of the vector y (2.6); and
- Expressions describing the convergence properties of the subspace predictor as the amount of experimental data tends toward infinity (2.10), (2.12).

Chapter 3

Model Free Subspace Based $\mathcal{H}_\infty$ Control

This chapter utilizes the subspace predictor of Chapter 2 to derive several novel $\mathcal{H}_\infty$-optimal output feedback controllers. All control laws utilize finite horizon cost functions. The term "model free" is somewhat of a misnomer in that L_w and L_u can be considered to form a high order plant model, however the terminology is retained to be consistent with previous literature, for example [11, 12]. The important distinction is that traditional plant models are not used.

Section 3.1 reviews the concept of mixed sensitivity $\mathcal{H}_\infty$ loop shaping design. Section 3.2 describes the main result of this thesis, the derivation of a strictly causal, model free, subspace based, level-γ central $\mathcal{H}_\infty$ controller. Section 3.3 derives a similar control law, where the assumption of strict causality of the control law is relaxed, *i.e.* the present reference signal (r_k) is known. Finally, Section 3.4 describes the control law when a multiplicative or additive uncertainty is included in the performance specification.

3.1 Output Feedback Mixed Sensitivity Cost Function

Consider the discrete time output feedback structure shown in Figure 3.1. An $\mathcal{H}_\infty$ mixed sensitivity criteria is commonly used to specify the desired minimum control performance

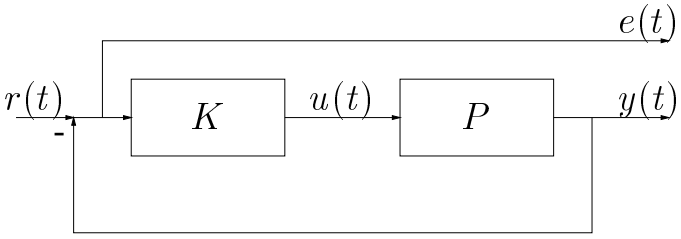

Figure 3.1: Output feedback block diagram

and desired maximum control usage. One such specification is

$$\left\| \begin{bmatrix} W_1 S \\ W_2 Q \end{bmatrix} \right\|_\infty \leq \gamma \tag{3.1}$$

where $S = (I + PK)^{-1} = G_{er}$, $Q = K(I + PK)^{-1} = G_{ur}$, and W_1 and W_2 are weighting functions chosen by the engineer [27]. Small S up to a desired cutoff frequency corresponds to y tracking r well at frequencies below the cutoff frequency. Limiting the magnitude of Q, especially at high frequencies, limits the control effort used. The selection of W_1 and W_2 specifies these desires: W_1 is often chosen to be large at low frequency and small at high frequency, while W_2 is typically chosen to be small at low frequency, and large at high frequency. In order for the problem to be well posed, W_1 and W_2 must have finite gain over all frequency. As a consequence, W_1 and W_2 must have relative McMillan degree 0, or equivalently, full-rank direct feedthrough terms, and must have at least as many output channels as input channels: typically, W_1 and W_2 are square. It is important to note that for each gain bound γ, there is a (possibly empty) set of control laws that satisfy (3.1).

An equivalent time domain discrete time expression for the specification (3.1) also exists. Let

$$z \triangleq \begin{bmatrix} z_{w_1} \\ z_{w_2} \end{bmatrix} = \begin{bmatrix} w_1 * e \\ w_2 * u \end{bmatrix} = \begin{bmatrix} w_1 * (r - y) \\ w_2 * u \end{bmatrix} \tag{3.2}$$

where w_1 and w_2 are the respective discrete impulse responses of the discrete time weighting functions W_1 and W_2. With (3.2), (3.1) can be written

$$\sup_r J_\infty(\gamma) \leq 0$$

where

$$J_\infty(\gamma) \triangleq \sum_{t=0}^{\infty} \left(z_t^T z_t - \gamma^2 r_t^T r_t \right)$$

and the system is assumed to be at rest at $t = 0$ [44]. It is important that the inequality be satisfied when the system is at rest at $t = 0$ so that the measured energy in z_t is strictly due to system excitation by the inputs r_t, and is not due to energy stored in the system prior to $t = 0$. The equivalent finite horizon problem is

$$\sup_r J(\gamma) \leq 0 \tag{3.3}$$

where

$$J(\gamma) \triangleq \sum_{t=0}^{i-1} \left(z_t^T z_t - \gamma^2 r_t^T r_t \right) \tag{3.4}$$

and the system is again assumed to be at rest at $t = 0$. Note that the length of the horizon (i) can be chosen arbitrarily. In this thesis, the horizon (i) is identical to the prediction horizon in Section 2.1 (i), so that (2.6) can be used to calculate $J(\gamma)$. Finally, the central (or minimum entropy, or minimax) finite horizon $\mathcal{H}_\infty$ controller is the controller that satisfies

$$\min_u \sup_r J(\gamma) \leq 0 \tag{3.5}$$

whenever the system is at rest at $t = 0$ [44].

3.2 Subspace Based Finite Horizon $\mathcal{H}_\infty$ Control

Figure 3.2 shows the generalized plant that results when the subspace predictor (2.6) is coupled to the weighting functions W_1 and W_2 to represent the mixed sensitivity cost (3.1). The block $L_w w_p$ represents the response of the predictor due to past excitation, while the L_u block represents response due to the inputs u. The level-γ $\mathcal{H}_\infty$ control design problem is to choose a control u such that the finite horizon $\mathcal{H}_\infty$ gain from r to z is at most magnitude γ. This section derives the condition on γ that ensures that the problem is feasible, and computes the central solution for this $\mathcal{H}_\infty$ control problem. The result is summarized in the following theorem.

Theorem 3.1

If measurements of the plant input (u), plant output (y), and reference (r) are available for

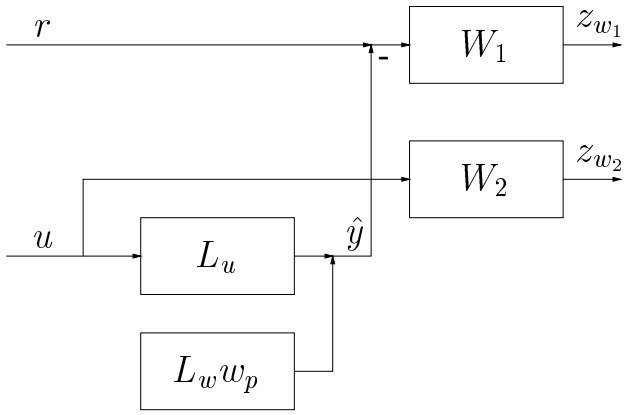

Figure 3.2: Generalized plant for non-robust $\mathcal{H}_\infty$ control design

times $\{k-i, \cdots, k-2, k-1\}$*, then the strictly causal, finite horizon, model free, subspace based, level-*γ*, central* $\mathcal{H}_\infty$ *control for times* $\{k, \cdots, k+i-1\}$ *is*

$$u_{opt} = -(L_u^T \widetilde{Q_1} L_u + Q_2)^{-1} \begin{bmatrix} \left(L_u^T \widetilde{Q_1} L_w\right)^T \\ \left(-L_u^T(\gamma^{-2}\widetilde{Q_1} + I)H_1^T \Gamma_1\right)^T \\ \left(H_2^T \Gamma_2\right)^T \end{bmatrix}^T \begin{bmatrix} w_p \\ x_{w_1} \\ x_{w_2} \end{bmatrix}_k \tag{3.6}$$

$$\widetilde{Q_1} \triangleq (Q_1^{-1} - \gamma^{-2} I)^{-1} \tag{3.7}$$

provided that

$$\gamma > \gamma_{\min} \triangleq \sqrt{\overline{\lambda}\left[(Q_1^{-1} + L_u Q_2^{-1} L_u^T)^{-1}\right]} \tag{3.8}$$

where

- u_{opt} *is the vector of optimal future plant inputs at times* $\{k, \cdots, k+i-1\}$

$$u_{opt} \triangleq \begin{bmatrix} u_k \\ \vdots \\ u_{k+i-1} \end{bmatrix} \tag{3.9}$$

- *The discrete LTI weighting filters* W_1 *and* W_2 *have minimal state space representation*

$$\begin{aligned} (x_{w_1})_{k+1} &= A_{w_1}(x_{w_1})_k + B_{w_1}(r_k - y_k) \\ (z_{w_1})_k &= C_{w_1}(x_{w_1})_k + D_{w_1}(r_k - y_k) \end{aligned}$$

$$\begin{aligned}(x_{w_2})_{k+1} &= A_{w_2}(x_{w_2})_k + B_{w_2}u_k \\ (z_{w_2})_k &= C_{w_2}(x_{w_2})_k + D_{w_2}u_k\end{aligned}$$

with $A_{w_1} \in \mathbb{R}^{n_{w_1} \times n_{w_1}}$, $D_{w_1} \in \mathbb{R}^{l_{w_1} \times l}$, $A_{w_2} \in \mathbb{R}^{n_{w_2} \times n_{w_2}}$, *and* $D_{w_2} \in \mathbb{R}^{l_{w_2} \times m}$

- H_1 *and* H_2 *are lower triangular Toeplitz matrices formed from the impulse responses (Markov parameters) of the discrete weighting filters* W_1 *and* W_2

$$H_1 \triangleq \begin{bmatrix} D_{w_1} & 0 & 0 & \cdots & 0 \\ C_{w_1}B_{w_1} & D_{w_1} & 0 & \cdots & 0 \\ C_{w_1}A_{w_1}B_{w_1} & C_{w_1}B_{w_1} & D_{w_1} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ C_{w_1}A_{w_1}^{i-2}B_{w_1} & C_{w_1}A_{w_1}^{i-3}B_{w_1} & C_{w_1}A_{w_1}^{i-4}B_{w_1} & \cdots & D_{w_1} \end{bmatrix} \tag{3.10}$$

$$H_2 \triangleq \begin{bmatrix} D_{w_2} & 0 & 0 & \cdots & 0 \\ C_{w_2}B_{w_2} & D_{w_2} & 0 & \cdots & 0 \\ C_{w_2}A_{w_2}B_{w_2} & C_{w_2}B_{w_2} & D_{w_2} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ C_{w_2}A_{w_2}^{i-2}B_{w_2} & C_{w_2}A_{w_2}^{i-3}B_{w_2} & C_{w_2}A_{w_2}^{i-4}B_{w_2} & \cdots & D_{w_2} \end{bmatrix} \tag{3.11}$$

- $Q_1 \triangleq H_1^T H_1$, $Q_2 \triangleq H_2^T H_2$

- Γ_1 *and* Γ_2 *are the extended observability matrices formed from the impulse responses of the weighting filters* W_1 *and* W_2

$$\Gamma_1 \triangleq \begin{bmatrix} C_{w_1} \\ C_{w_1}A_{w_1} \\ \vdots \\ C_{w_1}A_{w_1}^{i-1} \end{bmatrix} \qquad \Gamma_2 \triangleq \begin{bmatrix} C_{w_2} \\ C_{w_2}A_{w_2} \\ \vdots \\ C_{w_2}A_{w_2}^{i-1} \end{bmatrix} \tag{3.12}$$

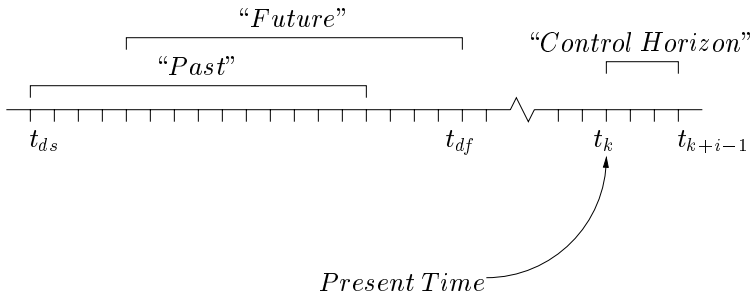

Figure 3.3: Time line for $\mathcal{H}_\infty$ control design

- $(w_p)_k$ *is a vector made up of past plant inputs and outputs*

$$(w_p)_k \triangleq \begin{bmatrix} u_{k-i} \\ \vdots \\ u_{k-1} \\ y_{k-i} \\ \vdots \\ y_{k-1} \end{bmatrix} \tag{3.13}$$

□

Despite the apparent complexity of the control law (3.6), (3.7), close examination reveals that the control law is simply a matrix operating on the data $(w_p)_k$ and states of the weighting functions $(x_{w_1})_k$, $(x_{w_2})_k$. The achieveable performance (3.8) is a function of experimental data through L_u, and a function of the weighting functions W_1 and W_2 through Q_1 and Q_2. Figure 3.3 clarifies the time line for Theorem 3.1. Data used to derive the subspace predictor are collected from t_{ds} (data start) to t_{df} (data finish), the present time is t_k, and the control horizon is from t_k to t_{k+i-1}.

The following mathematical results will be used in the proof of Theorem 3.1 [36].

Lemma 3.1

If A_1^{-1}, A_3^{-1}, *and* $\begin{bmatrix} A_1 & A_2 \\ A_2^T & A_3 \end{bmatrix}^{-1}$ *exist, then*

$$\begin{bmatrix} A_1 & A_2 \\ A_2^T & A_3 \end{bmatrix}^{-1} = \begin{bmatrix} (A_1 - A_2A_3^{-1}A_2^T)^{-1} & -(A_1 - A_2A_3^{-1}A_2^T)^{-1}A_2A_3^{-1} \\ -(A_3 - A_2^TA_1^{-1}A_2)^{-1}A_2^TA_1^{-1} & (A_3 - A_2^TA_1^{-1}A_2)^{-1} \end{bmatrix} \quad (3.14)$$

□

Lemma 3.2

If A^{-1}, C^{-1}, and $(A^{-1} + B^TC^{-1}B)^{-1}$ exist, then

$$A - AB^T(BAB^T + C)^{-1}BA = (A^{-1} + B^TC^{-1}B)^{-1} \quad (3.15)$$

$$AB^T(BAB^T + C)^{-1} = (A^{-1} + B^TC^{-1}B)^{-1}B^TC^{-1} \quad (3.16)$$

□

Proof of Theorem 3.1

Let k be the present time, and define

$$z_{w_1} \triangleq \begin{bmatrix} (z_{w_1})_k \\ \vdots \\ (z_{w_1})_{k+i-1} \end{bmatrix} \in \mathbb{R}^{il_{w_1}} \quad r \triangleq \begin{bmatrix} r_k \\ \vdots \\ r_{k+i-1} \end{bmatrix} \in \mathbb{R}^{il} \quad u \triangleq \begin{bmatrix} u_k \\ \vdots \\ u_{k+i-1} \end{bmatrix} \in \mathbb{R}^{im}$$

$$z_{w_2} \triangleq \begin{bmatrix} (z_{w_2})_k \\ \vdots \\ (z_{w_2})_{k+i-1} \end{bmatrix} \in \mathbb{R}^{il_{w_2}} \quad y \triangleq \begin{bmatrix} y_k \\ \vdots \\ y_{k+i-1} \end{bmatrix} \in \mathbb{R}^{il}$$

then (3.5) can be written in vector form

$$\min_u \sup_r J(\gamma) \leq 0 \quad (3.17)$$

where

$$J(\gamma) = z_{w_1}^Tz_{w_1} + z_{w_2}^Tz_{w_2} - \gamma^2r^Tr \quad (3.18)$$

and the system is assumed to be at rest at time k. Let $w_{1nr} \in \mathbb{R}^{il_{w_1}}$ and $w_{2nr} \in \mathbb{R}^{il_{w_2}}$ be the vectors of the natural responses of W_1 and W_2 due to $(x_{w_1})_k$ and $(x_{w_2})_k$ respectively.

Then

$$\begin{bmatrix} z_{w_1} \\ z_{w_2} \end{bmatrix} = \begin{bmatrix} H_1(r-y) + w_{1nr} \\ H_2 u + w_{2nr} \end{bmatrix}$$

Using the definitions of Γ_1 and Γ_2 (3.12), w_{1nr} and w_{2nr} are

$$\begin{bmatrix} w_{1nr} \\ w_{2nr} \end{bmatrix} = \begin{bmatrix} \Gamma_1(x_{w_1})_k \\ \Gamma_2(x_{w_2})_k \end{bmatrix}$$

thus

$$\begin{bmatrix} z_{w_1} \\ z_{w_2} \end{bmatrix} = \begin{bmatrix} H_1(r-y) + \Gamma_1(x_{w_1})_k \\ H_2 u + \Gamma_2(x_{w_2})_k \end{bmatrix} \tag{3.19}$$

Using (3.13), the strictly causal estimate of y (2.6) from Section 2.1 can also be written in vector form

$$\widehat{y} = L_w(w_p)_k + L_u u \tag{3.20}$$

Substituting the estimate of y (3.20) into (3.19) results in

$$\begin{bmatrix} z_{w_1} \\ z_{w_2} \end{bmatrix} = \begin{bmatrix} H_1 & -H_1 L_u & -H_1 L_w & \Gamma_1 & 0 \\ 0 & H_2 & 0 & 0 & \Gamma_2 \end{bmatrix} x \tag{3.21}$$

where

$$x \triangleq \begin{bmatrix} r \\ u \\ (w_p)_k \\ (x_{w_1})_k \\ (x_{w_2})_k \end{bmatrix}$$

Substituting (3.21) into (3.17) and (3.18) produces the objective

$$\min_u \sup_r x^T \begin{bmatrix} Q_1 - \gamma^2 I & -Q_1 L_u & -Q_1 L_w & H_1^T \Gamma_1 & 0 \\ -L_u^T Q_1 & L_u^T Q_1 L_u + Q_2 & L_u^T Q_1 L_w & -L_u^T H_1^T \Gamma_1 & H_2^T \Gamma_2 \\ -L_w^T Q_1 & L_w^T Q_1 L_u & L_w^T Q_1 L_w & -L_w^T H_1^T \Gamma_1 & 0 \\ \Gamma_1^T H_1 & -\Gamma_1^T H_1 L_u & -\Gamma_1^T H_1 L_w & \Gamma_1^T \Gamma_1 & 0 \\ 0 & \Gamma_2^T H_2 & 0 & 0 & \Gamma_2^T \Gamma_2 \end{bmatrix} x \leq 0 \tag{3.22}$$

whenever the system is at rest at time k.

The necessary condition for simultaneously maximizing with respect to r and minimizing with respect to u (a stationary point) is found by differentiating the left-hand side of (3.22) with respect to r and u, and setting the derivative to zero [6] *i.e.*

$$\frac{\partial J(\gamma)}{\partial \begin{bmatrix} r \\ u \end{bmatrix}} = 0 \tag{3.23}$$

or

$$2 \begin{bmatrix} Q_1 - \gamma^2 I & -Q_1 L_u & -Q_1 L_w & H_1^T \Gamma_1 & 0 \\ -L_u^T Q_1 & L_u^T Q_1 L_u + Q_2 & L_u^T Q_1 L_w & -L_u^T H_1^T \Gamma_1 & H_2^T \Gamma_2 \end{bmatrix} \begin{bmatrix} r \\ u \\ (w_p)_k \\ (x_{w_1})_k \\ (x_{w_2})_k \end{bmatrix} = 0$$

which implies

$$\begin{bmatrix} Q_1 - \gamma^2 I & -Q_1 L_u \\ -L_u^T Q_1 & L_u^T Q_1 L_u + Q_2 \end{bmatrix} \begin{bmatrix} r \\ u \end{bmatrix} = \begin{bmatrix} Q_1 L_w & -H_1^T \Gamma_1 & 0 \\ -L_u^T Q_1 L_w & L_u^T H_1^T \Gamma_1 & -H_2^T \Gamma_2 \end{bmatrix} \begin{bmatrix} w_p \\ x_{w_1} \\ x_{w_2} \end{bmatrix}_k \tag{3.24}$$

The sufficient condition for the optimization requires that the Hessian of the left-hand side of (3.22) satisfy the saddle condition for a maximum in r and minimum in u [6], *i.e. im* of the eigenvalues of the Hessian must be positive and *il* of the eigenvalues of the Hessian must be negative. The Hessian is

$$H_{hess} \triangleq \frac{\partial^2 J(\gamma)}{\partial \begin{bmatrix} r \\ u \end{bmatrix}^2} = \begin{bmatrix} Q_1 - \gamma^2 I & -Q_1 L_u \\ -L_u^T Q_1 & L_u^T Q_1 L_u + Q_2 \end{bmatrix}$$

Also let

$$\begin{bmatrix} A_1 & A_2 \\ A_2^T & A_3 \end{bmatrix} \triangleq \begin{bmatrix} Q_1 - \gamma^2 I & -Q_1 L_u \\ -L_u^T Q_1 & L_u^T Q_1 L_u + Q_2 \end{bmatrix} = H_{hess}$$

Note that $Q_1 = Q_1^T$ and $Q_2 = Q_2^T$ are guaranteed to be positive definite (as opposed to positive semi-definite) since the relative degree condition on W_1 and W_2 implies that D_1

and D_2 are full rank. The positivity of Q_1 and Q_2 implies $A_3 > 0$, which implies A_3^{-1} exists. Thus the Hessian can be written using the Schur decomposition

$$H_{hess} = \begin{bmatrix} I & A_2 A_3^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} A_1 - A_2 A_3^{-1} A_2^T & 0 \\ 0 & A_3 \end{bmatrix} \begin{bmatrix} I & 0 \\ A_3^{-1} A_2^T & I \end{bmatrix} \tag{3.25}$$

Let

$$d \triangleq \begin{bmatrix} I & 0 \\ A_3^{-1} A_2^T & I \end{bmatrix} \begin{bmatrix} r \\ u \end{bmatrix} = \begin{bmatrix} r \\ A_3^{-1} A_2^T r + u \end{bmatrix} \tag{3.26}$$

then second derivative of the cost in the $\begin{bmatrix} r^T & u^T \end{bmatrix}^T$ direction can be rewritten as

$$H_{hess} = d^T \begin{bmatrix} A_1 - A_2 A_3^{-1} A_2^T & 0 \\ 0 & A_3 \end{bmatrix} d \tag{3.27}$$

The condition of being at a saddle point can be viewed equally well in the d coordinates. Thus the saddle condition required to simultaneously maximize over r and minimize over u is that im of the eigenvalues of

$$\begin{bmatrix} A_1 - A_2 A_3^{-1} A_2^T & 0 \\ 0 & A_3 \end{bmatrix} \tag{3.28}$$

must be positive, and the remaining il eigenvalues must be negative. Since $A_3 > 0$, and $A_3 \in \mathbb{R}^{im \times im}$, A_3 provides the required im positive eigenvalues. As a result, the saddle condition reduces to

$$A_1 - A_2 A_3^{-1} A_2^T < 0 \tag{3.29}$$

or

$$Q_1 - Q_1 L_u (L_u^T Q_1 L_u + Q_2)^{-1} L_u^T Q_1 < \gamma^2 I \tag{3.30}$$

Using (3.15), (3.30) is

$$(Q_1^{-1} + L_u Q_2^{-1} L_u^T)^{-1} < \gamma^2 I$$

or finally

$$\gamma > \sqrt{\overline{\lambda}\left[(Q_1^{-1} + L_u Q_2^{-1} L_u^T)^{-1}\right]} \equiv \gamma_{\min}$$

The condition for the saddle (3.29) also establishes that the matrix on the left-hand side of (3.24) is invertible. Thus the worst case r and the optimum u are given by

$$\begin{bmatrix} r_{wc} \\ u_{opt} \end{bmatrix} = \begin{bmatrix} Q_1 - \gamma^2 I & -Q_1 L_u \\ -L_u^T Q_1 & L_u^T Q_1 L_u + Q_2 \end{bmatrix}^{-1} \begin{bmatrix} Q_1 L_w & -H_1^T \Gamma_1 & 0 \\ -L_u^T Q_1 L_w & L_u^T H_1^T \Gamma_1 & -H_2^T \Gamma_2 \end{bmatrix} \begin{bmatrix} w_p \\ x_{w_1} \\ x_{w_2} \end{bmatrix}_k \tag{3.31}$$

Left multiplying (3.14) in Lemma 3.1 by $\begin{bmatrix} 0 & I \end{bmatrix}$ produces the relation

$$\begin{bmatrix} 0 & I \end{bmatrix} \begin{bmatrix} A_1 & A_2 \\ A_2^T & A_3 \end{bmatrix}^{-1} = (A_3 - A_2^T A_1^{-1} A_2)^{-1} \begin{bmatrix} -A_2^T A_1^{-1} & I \end{bmatrix} \tag{3.32}$$

Using (3.32), the second row of (3.31) can be written as

$$\begin{aligned} u_{opt} \;=\; & \left(L_u^T Q_1 L_u + Q_2 - L_u^T Q_1 (Q_1 - \gamma^2 I)^{-1} Q_1 L_u\right)^{-1} \cdot \\ & \begin{bmatrix} L_u^T Q_1 (Q_1 - \gamma^2 I)^{-1} & I \end{bmatrix} \begin{bmatrix} Q_1 L_w & -H_1^T \Gamma_1 & 0 \\ -L_u^T Q_1 L_w & L_u^T H_1^T \Gamma_1 & -H_2^T \Gamma_2 \end{bmatrix} \begin{bmatrix} w_p \\ x_{w_1} \\ x_{w_2} \end{bmatrix}_k \end{aligned}$$

or

$$\begin{aligned} u_{opt} \;=\; & \left(L_u^T \left(Q_1 - Q_1 (Q_1 - \gamma^2 I)^{-1} Q_1\right) L_u + Q_2\right)^{-1} \cdot \\ & \begin{bmatrix} \left(-L_u^T \left(Q_1 - Q_1 (Q_1 - \gamma^2 I)^{-1} Q_1\right) L_w\right)^T \\ \left(L_u^T \left(-Q_1 (Q_1 - \gamma^2 I)^{-1} + I\right) H_1^T \Gamma_1\right)^T \\ \left(-H_2^T \Gamma_2\right)^T \end{bmatrix}^T \begin{bmatrix} w_p \\ x_{w_1} \\ x_{w_2} \end{bmatrix}_k \end{aligned} \tag{3.33}$$

Using (3.15)

$$Q_1 - Q_1 (Q_1 - \gamma^2 I)^{-1} Q_1 = (Q_1^{-1} - \gamma^{-2} I)^{-1} = \widetilde{Q_1} \tag{3.34}$$

Using (3.16)

$$-Q_1 (Q_1 - \gamma^2 I)^{-1} = (Q_1^{-1} - \gamma^{-2} I)^{-1} \gamma^{-2} I = \gamma^{-2} \widetilde{Q_1} \tag{3.35}$$

Substituting (3.34) and (3.35) into the control law (3.33) produces

$$\begin{aligned} u_{opt} &= -(L_u^T\widetilde{Q_1}L_u + Q_2)^{-1} \begin{bmatrix} \left(L_u^T\widetilde{Q_1}L_w\right)^T \\ \left(-L_u^T(\gamma^{-2}\widetilde{Q_1} + I)H_1^T\Gamma_1\right)^T \\ \left(H_2^T\Gamma_2\right)^T \end{bmatrix}^T \begin{bmatrix} w_p \\ x_{w_1} \\ x_{w_2} \end{bmatrix}_k \\ \widetilde{Q_1} &= (Q_1^{-1} - \gamma^{-2}I)^{-1} \end{aligned}$$

provided $\widetilde{Q_1}$ exists. In general $(Q_1^{-1} - \gamma^{-2}I)$ could have both positive and negative eigenvalues, and for some values of γ, $(Q_1^{-1} - \gamma^{-2}I)$ could be singular. In this case, (3.31) must be used to calculate u_{opt}.

All that remains is to determine if the inequality on the right-hand side of (3.22) is satisfied. Setting all system states at time k to zero, *i.e.* $(w_p)_k = 0$, $(x_{w_1})_k = 0$, and $(x_{w_2})_k = 0$, (3.22) becomes

$$\begin{bmatrix} r_{wc} \\ u_{opt} \end{bmatrix}^T \begin{bmatrix} Q_1 - \gamma^2 I & -Q_1L_u \\ -L_u^TQ_1 & L_u^TQ_1L_u + Q_2 \end{bmatrix} \begin{bmatrix} r_{wc} \\ u_{opt} \end{bmatrix} \leq 0 \tag{3.36}$$

and (3.31) reduces to

$$\begin{bmatrix} r_{wc} \\ u_{opt} \end{bmatrix} = 0$$

thus (3.36) is satisfied. □

3.3 Model Free Subspace Based $\mathcal{H}_\infty$ Control With r_k Known

In the previous section, it is assumed that r_k is unknown when $\{u_k, \cdots, u_{k+i-1}\}$ are calculated. In this section, the requirement of a strictly causal controller is removed: it is assumed that r_k is known, *i.e.* the desired trajectory at time k is available before time k. The resulting control law (3.38) – (3.40), is only slightly different from the strictly causal control law (3.6), (3.7). The advantage of (3.38) – (3.40) is that a slightly better performance is achieved, as is shown in Theorem 3.3 at the end of this section. The cost to achieve this slightly better performance is that the desired reference r_k needs to be known prior to implementing the control at time k: implementation issues usually govern whether the strictly causal or the causal control law can be used.

Referring to (3.10), let

$$\begin{bmatrix} H_{11} & H_{12} \end{bmatrix} \stackrel{\triangle}{=} H_1 \qquad (3.37)$$
$$H_{11} \in \mathbb{R}^{il_{w_1} \times l}$$
$$H_{12} \in \mathbb{R}^{il_{w_1} \times (i-1)l}$$

that is, H_{11} is the first block column of H_1, and H_{12} contains the remaining columns of H_1. For the purpose of this section, let $r \in \mathbb{R}^{(i-1)l}$, and $Q_{12} \stackrel{\triangle}{=} H_{12}^T H_{12}$. Note that $Q_{12} > 0$. Then (3.21) can be written

$$\begin{bmatrix} z_{w_1} \\ z_{w_2} \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & -H_1 L_u & -H_1 L_w & \Gamma_1 & 0 \\ 0 & 0 & H_2 & 0 & 0 & \Gamma_2 \end{bmatrix} x$$

with x broken into six subvectors

$$x = \begin{bmatrix} r_k \\ r \\ u \\ (w_p)_k \\ (x_{w_1})_k \\ (x_{w_2})_k \end{bmatrix}$$

These definitions and are used in Theorem 3.2 and its proof.

Theorem 3.2

If measurements of the plant input (u) and plant output (y) are available for times $\{k-i, \cdots, k-2, k-1\}$*, and measurements of the reference (r) are available for times* $\{k-i, \cdots, k-1, k\}$*, then the finite horizon, model free, subspace based, level-*γ *central* $\mathcal{H}_\infty$ *control for times* $\{k, \cdots, k+i-1\}$ *is*

$$u_{opt} = (L_u^T \widetilde{Q_{12}} L_u + Q_2)^{-1} \begin{bmatrix} \left(L_u^T H_1^T (I - H_{12} \overline{Q_{12}} H_{12}^T) H_{11}\right)^T \\ \left(-L_u^T (Q_1 - H_1^T H_{12} \overline{Q_{12}} H_{12}^T H_1) L_w\right)^T \\ \left(L_u^T H_1^T (I - H_{12} \overline{Q_{12}} H_{12}^T) \Gamma_1\right)^T \\ \left(-H_2^T \Gamma_2\right)^T \end{bmatrix}^T \begin{bmatrix} r_k \\ (w_p)_k \\ (x_{w_1})_k \\ (x_{w_2})_k \end{bmatrix} \qquad (3.38)$$

$$\widetilde{Q_{12}} \triangleq Q_1 - Q_{12}(Q_{12} - \gamma^2 I)^{-1} Q_{12} \tag{3.39}$$

$$\overline{Q_{12}} \triangleq (Q_{12} - \gamma^2 I)^{-1} \tag{3.40}$$

provided that

$$\gamma > \gamma_{\min_c} \triangleq \sqrt{\overline{\lambda}\left[H_{12}^T (I + H_1 L_u Q_2^{-1} L_u^T H_1^T)^{-1} H_{12}\right]} \tag{3.41}$$

□

Outline of Proof of Theorem 3.2

The proof of Theorem 3.2 is nearly identical to the proof of Theorem 3.1. Only some key equations will be written. The objective is given by

$$\min_u \sup_r x^T M x \leq 0$$

where

$$M \triangleq \begin{bmatrix} H_{11}^T H_{11} - \gamma^2 & H_{11}^T H_{12} & -H_{11}^T H_1 L_u & -H_{11}^T H_1 L_w & H_{11}^T \Gamma_1 & 0 \\ H_{12}^T H_{11} & Q_{12} - \gamma^2 I & -H_{12}^T H_1 L_u & -H_{12}^T H_1 L_w & H_{12}^T \Gamma_1 & 0 \\ -L_u^T H_1^T H_{11} & -L_u^T H_1^T H_{12} & L_u^T Q_1 L_u + Q_2 & L_u^T Q_1 L_w & -L_u^T H_1^T \Gamma_1 & H_2^T \Gamma_2 \\ -L_w H_1^T H_{11} & -L_w^T H_1^T H_{12} & L_w^T Q_1 L_u & L_w^T Q_1 L_w & -L_w^T H_1^T \Gamma_1 & 0 \\ \Gamma_1^T H_{11} & \Gamma_1^T H_1 & -\Gamma_1^T H_1 L_u & -\Gamma_1^T H_1 L_w & \Gamma_1^T \Gamma_1 & 0 \\ 0 & 0 & \Gamma_2^T H_2 & 0 & 0 & \Gamma_2^T \Gamma_2 \end{bmatrix}$$

The necessary condition (3.23) implies

$$\begin{bmatrix} Q_{12} - \gamma^2 I & -H_{12}^T H_1 L_u \\ -L_u^T H_1^T H_{12} & L_u^T Q_1 L_u + Q_2 \end{bmatrix} \begin{bmatrix} r \\ u \end{bmatrix} =$$

$$\begin{bmatrix} -H_{12}^T H_{11} & H_{12}^T H_1 L_w & -H_{12}^T \Gamma_1 & 0 \\ L_u^T H_1^T H_{11} & -L_u^T Q_1 L_w & L_u^T H_1^T \Gamma_1 & -H_2^T \Gamma_2 \end{bmatrix} \begin{bmatrix} r_k \\ (w_p)_k \\ (x_{w_1})_k \\ (x_{w_2})_k \end{bmatrix}$$

The sufficient condition, given by a saddle condition argument identical to that in the proof of Theorem 3.1 is

$$Q_{12} - \gamma^2 I - H_{12}^T H_1 L_u (L_u^T Q_1 L_u + Q_2)^{-1} L_u^T H_1^T H_{12} < 0$$

or

$$\begin{aligned} \gamma &> \sqrt{\overline{\lambda}\left[Q_{12} - H_{12}^T H_1 L_u (L_u^T Q_1 L_u + Q_2)^{-1} L_u^T H_1^T H_{12}\right]} \\ &= \sqrt{\overline{\lambda}\left[H_{12}^T (I - H_1 L_u (L_u^T H_1^T H_1 L_u + Q_2)^{-1} L_u^T H_1^T) H_{12}\right]} \end{aligned} \tag{3.42}$$

Using (3.15), (3.42) can be written

$$\gamma > \sqrt{\overline{\lambda}\left[H_{12}^T (I + H_1 L_u Q_2^{-1} L_u^T H_1^T)^{-1} H_{12}\right]} \equiv \gamma_{\min_c}$$

The remainder of the proof follows the proof of Theorem 3.1. □

Theorem 3.3 compares the best achieveable performance of the strictly causal control law to the best achieveable performance of the causal control law. As expected, the control law that uses more information can achieve better performance.

Theorem 3.3

The best achieveable performance of the causal control law (3.41), is better than or equal to the best achievable performance of the strictly causal control law (3.8), i.e.

$$\gamma_{\min} \geq \gamma_{\min_c}$$

□

Proof of Theorem 3.3

The best achieveable performance of the strictly causal control law (3.8) can be expanded using (3.15)

$$\begin{aligned} \gamma_{\min} &= \sqrt{\overline{\lambda}\left[(Q_1^{-1} + L_u Q_2^{-1} L_u^T)^{-1}\right]} \\ &= \sqrt{\overline{\lambda}\left[Q_1 - Q_1 L_u (L_u^T Q_1 L_u + Q_2)^{-1} L_u^T Q_1\right]} \end{aligned} \tag{3.43}$$

Since $Q_1 = H_1^T H_1$, (3.43) can be written

$$\gamma_{\min} = \sqrt{\overline{\lambda}\left[H_1^T (I - H_1 L_u (L_u^T H_1^T H_1 L_u + Q_2)^{-1} L_u^T H_1^T) H_1\right]} \tag{3.44}$$

Using (3.15), (3.44) is

$$\gamma_{\min} = \sqrt{\overline{\lambda}\left[H_1^T(I + H_1 L_u Q_2^{-1} L_u^T H_1^T)^{-1} H_1\right]}$$

and with (3.37)

$$\begin{aligned}
\gamma_{\min} &= \sqrt{\overline{\lambda}\left[\begin{bmatrix} H_{11}^T \\ H_{12}^T \end{bmatrix}(I + H_1 L_u Q_2^{-1} L_u^T H_1^T)^{-1}\begin{bmatrix} H_{11} & H_{12} \end{bmatrix}\right]} \\
&= \sqrt{\max_{\|v\|=1}\begin{bmatrix} v_1^T & v_2^T \end{bmatrix}\begin{bmatrix} H_{11}^T \\ H_{12}^T \end{bmatrix}(I + H_1 L_u Q_2^{-1} L_u^T H_1^T)^{-1}\begin{bmatrix} H_{11} & H_{12} \end{bmatrix}\begin{bmatrix} v_1 \\ v_2 \end{bmatrix}} \\
&= \sqrt{\max_{\|v\|=1}(v_1^T H_{11}^T + v_2^T H_{12}^T)(I + H_1 L_u Q_2^{-1} L_u^T H_1^T)^{-1}(H_{11} v_1 + H_{12} v_2)}
\end{aligned}$$

where $v \stackrel{\triangle}{=} \begin{bmatrix} v_1^T & v_2^T \end{bmatrix}^T$, and v_1 and v_2 are appropriately sized vectors. Setting $v_1 = 0$ results in

$$\begin{aligned}
\gamma_{\min} &= \sqrt{\max_{\|v\|=1}(v_1^T H_{11}^T + v_2^T H_{12}^T)(I + H_1 L_u Q_2^{-1} L_u^T H_1^T)^{-1}(H_{11} v_1 + H_{12} v_2)} \\
&\geq \sqrt{\max_{\|v_2\|=1}\left[v_2^T H_{12}^T (I + H_1 L_u Q_2^{-1} L_u^T H_1^T)^{-1} H_{12} v_2\right]} \\
&= \sqrt{\overline{\lambda}\left[H_{12}^T (I + H_1 L_u Q_2^{-1} L_u^T H_1^T)^{-1} H_{12}\right]} \\
&\equiv \gamma_{\min_c}
\end{aligned}$$

thus

$$\gamma_{\min} \geq \gamma_{\min_c}$$

□

3.4 Robust Subspace Based Finite Horizon $\mathcal{H}_\infty$ Control

The derivations of the subspace based $\mathcal{H}_\infty$ control laws in the presence of multiplicative or additive uncertainties are nearly identical to the derivation of Theorem 3.1. To avoid repetition, only the outlines of the proofs of each result will be presented.

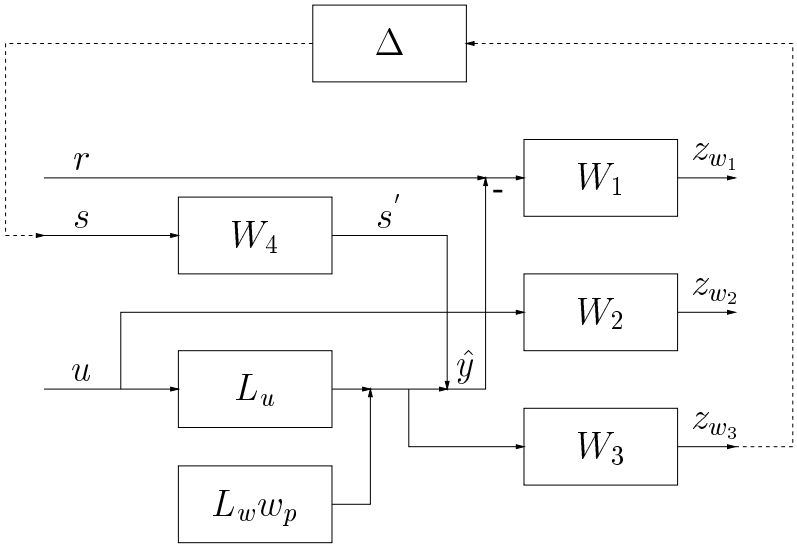

Figure 3.4: Generalized plant for robust $\mathcal{H}_\infty$ control design: multiplicative uncertainty

3.4.1 Multiplicative Uncertainty

Figure 3.4 shows the generalized plant of Figure 3.2 with the addition of a frequency weighted multiplicative uncertainty Δ. The weighting functions W_3 and W_4 allow the engineer to shape the uncertainty so as to capture the uncertainty's frequency dependence. Δ has l_{w_3} inputs and m_{w_4} outputs. The predictor now takes the form $\widehat{y} = (I + W_4(q)\Delta(q)W_3(q))(L_w w_p + L_u u)$, where q is the forward shift operator.

The robust level-γ $\mathcal{H}_\infty$ control design problem is to choose a control u such that the finite horizon $\mathcal{H}_\infty$ gain from $\left[\begin{array}{cc} r^T & s^T \end{array}\right]^T$ to z is at most magnitude γ. Theorem 3.4 states the central solution to this problem.

Theorem 3.4

If measurements of the plant input (u), plant output (y), and reference (r) are available for times $\{k-i, \cdots, k-2, k-1\}$, then the robust, strictly causal, finite horizon, model free, subspace based, level-γ central $\mathcal{H}_\infty$ control for times $\{k, \cdots, k+i-1\}$ is given by

$$\left[\begin{array}{c} r_{wc} \\ s_{wc} \\ u_{opt} \end{array}\right] = \left[\begin{array}{ccc} Q_1 - \gamma^2 I & -Q_1 H_4 & -Q_1 L_u \\ -H_4^T Q_1 & H_4^T Q_1 H_4 - \gamma^2 I & H_4^T Q_1 L_u \\ -L_u^T Q_1 & L_u^T Q_1 H_4 & L_u^T (Q_1 + Q_3) L_u + Q_2 \end{array}\right]^{-1} \cdot$$

$$\begin{bmatrix} Q_1 L_w & -H_1^T \Gamma_1 & 0 & 0 & Q_1 \Gamma_4 \\ -H_4^T Q_1 L_w & H_4^T H_1^T \Gamma_1 & 0 & 0 & -H_4^T Q_1 \Gamma_4 \\ -L_u^T (Q_1 + Q_3) L_w & L_u^T H_1^T \Gamma_1 & -H_2^T \Gamma_2 & -L_u^T H_3^T \Gamma_3 & -L_u^T Q_1 \Gamma_4 \end{bmatrix} \begin{bmatrix} w_p \\ x_{w_1} \\ x_{w_2} \\ x_{w_3} \\ x_{w_4} \end{bmatrix}_k \tag{3.45}$$

provided that

$$\gamma > \gamma_{\min_m} \triangleq \sqrt{\overline{\lambda} \left[(I + Q_{4,outer})(Q_1^{-1} + L_u (Q_2 + L_u^T Q_3 L_u)^{-1} L_u^T)^{-1} \right]} \tag{3.46}$$

where

- *The discrete LTI weighting filters W_3 and W_4 have minimal state space representation*

$$\begin{aligned} (x_{w_3})_{k+1} &= A_{w_3} (x_{w_3})_k + B_{w_3} y_k \\ (z_{w_3})_k &= C_{w_3} (x_{w_3})_k + D_{w_3} y_k \\ (x_{w_4})_{k+1} &= A_{w_4} (x_{w_4})_k + B_{w_4} s_k \\ s_k^{'} &= C_{w_4} (x_{w_4})_k + D_{w_4} s_k \end{aligned}$$

  with $A_{w_3} \in \mathbb{R}^{n_{w_3} \times n_{w_3}}$, $D_{w_3} \in \mathbb{R}^{l_{w_3} \times l}$, $A_{w_4} \in \mathbb{R}^{n_{w_4} \times n_{w_4}}$, and $D_{w_4} \in \mathbb{R}^{l \times m_{w_4}}$

- *H_3 and H_4 are lower triangular Toeplitz matrices formed from the impulse responses (Markov parameters) of the discrete weighting filters W_3 and W_4*

$$H_3 \triangleq \begin{bmatrix} D_{w_3} & 0 & 0 & \cdots & 0 \\ C_{w_3} B_{w_3} & D_{w_3} & 0 & \cdots & 0 \\ C_{w_3} A_{w_3} B_{w_3} & C_{w_3} B_{w_3} & D_{w_3} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ C_{w_3} A_{w_3}^{i-2} B_{w_3} & C_{w_3} A_{w_3}^{i-3} B_{w_3} & C_{w_3} A_{w_3}^{i-4} B_{w_3} & \cdots & D_{w_3} \end{bmatrix}$$

$$H_4 \triangleq \begin{bmatrix} D_{w_4} & 0 & 0 & \cdots & 0 \\ C_{w_4} B_{w_4} & D_{w_4} & 0 & \cdots & 0 \\ C_{w_4} A_{w_4} B_{w_4} & C_{w_4} B_{w_4} & D_{w_4} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ C_{w_4} A_{w_4}^{i-2} B_{w_4} & C_{w_4} A_{w_4}^{i-3} B_{w_4} & C_{w_4} A_{w_4}^{i-4} B_{w_4} & \cdots & D_{w_4} \end{bmatrix}$$

- $Q_3 \triangleq H_3^T H_3 > 0$, $Q_{4,outer} \triangleq H_4 H_4^T \geq 0$
- Γ_3 *and* Γ_4 *are the extended observability matrices formed from the impulse responses of the weighting filters* W_3 *and* W_4

$$\Gamma_3 \triangleq \begin{bmatrix} C_{w_3} \\ C_{w_3} A_{w_3} \\ \vdots \\ C_{w_3} A_{w_3}^{i-1} \end{bmatrix} \qquad \Gamma_4 \triangleq \begin{bmatrix} C_{w_4} \\ C_{w_4} A_{w_4} \\ \vdots \\ C_{w_4} A_{w_4}^{i-1} \end{bmatrix}$$

□

Lemma 3.3, also known as the Schur complement, will be used in the proof of Theorem 3.4 [4].

Lemma 3.3

If

$$X \triangleq \begin{bmatrix} A & B \\ B^T & C \end{bmatrix} = X^T \qquad C > 0$$

then

$$X > 0 \qquad \Leftrightarrow \qquad A - BC^{-1}B^T > 0 \tag{3.47}$$

Similarly, if $A > 0$ *then*

$$X > 0 \qquad \Leftrightarrow \qquad C - B^T A^{-1} B > 0 \tag{3.48}$$

□

Outline of Proof of Theorem 3.4

Using arguments identical to those used in the proof of Theorem 3.1, the performance variables are

$$\begin{bmatrix} z_{w_1} \\ z_{w_2} \\ z_{w_3} \end{bmatrix} = \begin{bmatrix} H_1 & -H_1 H_4 & -H_1 L_u & -H_1 L_w & \Gamma_1 & 0 & 0 & -H_1 \Gamma_4 \\ 0 & 0 & H_2 & 0 & 0 & \Gamma_2 & 0 & 0 \\ 0 & 0 & H_3 L_u & H_3 L_w & 0 & 0 & \Gamma_3 & 0 \end{bmatrix} x \tag{3.49}$$

where

$$x \triangleq \begin{bmatrix} r \\ s \\ u \\ (w_p)_k \\ (x_{w_1})_k \\ (x_{w_2})_k \\ (x_{w_3})_k \\ (x_{w_4})_k \end{bmatrix} \tag{3.50}$$

The objective (3.5) is

$$\min_u \sup_{r,s} x^T \begin{bmatrix} M_1 & M_2 \end{bmatrix} x \leq 0 \tag{3.51}$$

where

$$M_1 \triangleq \begin{bmatrix} Q_1 - \gamma^2 I & -Q_1 H_4 & -Q_1 L_u \\ -H_4^T Q_1 & H_4^T Q_1 H_4 - \gamma^2 I & H_4^T Q_1 L_u \\ -L_u^T Q_1 & L_u^T Q_1 H_4 & L_u^T (Q_1 + Q_3) L_u + Q_2 \\ -L_w^T Q_1 & L_w^T Q_1 H_4 & L_w^T (Q_1 + Q_3) L_u \\ \Gamma_1^T H_1 & -\Gamma_1^T H_1 H_4 & -\Gamma_1^T H_1 L_u \\ 0 & 0 & \Gamma_2^T H_2 \\ 0 & 0 & \Gamma_3^T H_3 L_u \\ -\Gamma_4^T Q_1 & \Gamma_4^T Q_1 H_4 & \Gamma_4^T Q_1 L_u \end{bmatrix}$$

$$M_2 \triangleq \begin{bmatrix} -Q_1 L_w & H_1^T \Gamma_1 & 0 & 0 & -Q_1 \Gamma_4 \\ H_4^T Q_1 L_w & -H_4^T H_1^T \Gamma_1 & 0 & 0 & H_4^T Q_1 \Gamma_4 \\ L_u^T (Q_1 + Q_3) L_w & -L_u^T H_1^T \Gamma_1 & H_2^T \Gamma_2 & L_u^T H_3^T \Gamma_3 & L_u^T Q_1 \Gamma_4 \\ L_w^T (Q_1 + Q_3) L_w & -L_w^T H_1^T \Gamma_1 & 0 & L_w^T H_3^T \Gamma_3 & L_w^T Q_1 \Gamma_4 \\ -\Gamma_1^T H_1 L_w & \Gamma_1^T \Gamma_1 & 0 & 0 & -\Gamma_1^T H_1 \Gamma_4 \\ 0 & 0 & \Gamma_2^T \Gamma_2 & 0 & 0 \\ \Gamma_3^T H_3 L_w & 0 & 0 & \Gamma_3^T \Gamma_3 & 0 \\ \Gamma_4^T Q_1 L_w & -\Gamma_4^T H_1^T \Gamma_1 & 0 & 0 & \Gamma_4^T Q_1 \Gamma_4 \end{bmatrix}$$

when the system is at rest at time k.

The necessary condition (3.23), *i.e.* taking the derivative and setting it equal to zero,

results in the control law

$$\begin{bmatrix} r_{wc} \\ s_{wc} \\ u_{opt} \end{bmatrix} = \begin{bmatrix} Q_1 - \gamma^2 I & -Q_1 H_4 & -Q_1 L_u \\ -H_4^T Q_1 & H_4^T Q_1 H_4 - \gamma^2 I & H_4^T Q_1 L_u \\ -L_u^T Q_1 & L_u^T Q_1 H_4 & L_u^T (Q_1 + Q_3) L_u + Q_2 \end{bmatrix}^{-1} \cdot$$

$$\begin{bmatrix} Q_1 L_w & -H_1^T \Gamma_1 & 0 & 0 & Q_1 \Gamma_4 \\ -H_4^T Q_1 L_w & H_4^T H_1^T \Gamma_1 & 0 & 0 & -H_4^T Q_1 \Gamma_4 \\ -L_u^T (Q_1 + Q_3) L_w & L_u^T H_1^T \Gamma_1 & -H_2^T \Gamma_2 & -L_u^T H_3^T \Gamma_3 & -L_u^T Q_1 \Gamma_4 \end{bmatrix} \begin{bmatrix} w_p \\ x_{w_1} \\ x_{w_2} \\ x_{w_3} \\ x_{w_4} \end{bmatrix}_k$$

The sufficient (saddle) condition is that

$$H_{hess} \triangleq \begin{bmatrix} Q_1 - \gamma^2 I & -Q_1 H_4 & -Q_1 L_u \\ -H_4^T Q_1 & H_4^T Q_1 H_4 - \gamma^2 I & H_4^T Q_1 L_u \\ -L_u^T Q_1 & L_u^T Q_1 H_4 & L_u^T (Q_1 + Q_3) L_u + Q_2 \end{bmatrix} \tag{3.52}$$

must have im positive eigenvalues while the remaining $i(l + m_{w_4})$ eigenvalues must be negative. The im positive eigenvalues are associated with the minimization in (3.51), while the $i(l+m_{w_4})$ negative eigenvalues are associated with the supremum in (3.51). Recognizing that $L_u^T(Q_1+Q_3)L_u+Q_2$ is positive, (3.52) can be broken up using the Schur decomposition of Section 3.2 with

$$\begin{aligned} A_1 &\triangleq \begin{bmatrix} Q_1 - \gamma^2 I & -Q_1 H_4 \\ -H_4^T Q_1 & H_4^T Q_1 H_4 - \gamma^2 I \end{bmatrix} \\ A_2 &\triangleq \begin{bmatrix} -Q_1 L_u \\ H_4^T Q_1 L_u \end{bmatrix} \\ A_3 &\triangleq L_u^T (Q_1 + Q_3) L_u + Q_2 \end{aligned}$$

Using the change of coordinates argument (3.25) – (3.28), and again noting that $A_3 > 0$ and $A_3 \in \mathbb{R}^{im \times im}$, the eigenvalue condition on (3.52) requires (3.29) or

$$-A_1 + A_2 A_3^{-1} A_2^T > 0$$

which can be written

$$\begin{bmatrix} \gamma^2 I - X & XH_4 \\ H_4^T X & \gamma^2 I - H_4^T X H_4 \end{bmatrix} > 0 \tag{3.53}$$

where

$$\begin{aligned} X &\triangleq Q_1 - Q_1 L_u (L_u^T (Q_1 + Q_3) L_u + Q_2)^{-1} L_u^T Q_1 \\ &= Q_1 - Q_1 L_u (L_u^T Q_1 L_u + (Q_2 + L_u^T Q_3 L_u))^{-1} L_u^T Q_1 \end{aligned} \tag{3.54}$$

Using (3.15), *i.e.* Lemma 3.2, (3.54) is

$$X = (Q_1^{-1} + L_u (Q_2 + L_u^T Q_3 L_u)^{-1} L_u^T)^{-1} \tag{3.55}$$

By inspection, (3.55) implies $X = X^T > 0$. Referring to (3.53), it is clear that

$$\gamma^2 I - H_4^T X H_4 > 0 \tag{3.56}$$

is necessary for (3.53) to be satisfied. Thus it is possible to take the Schur complement (3.47) of (3.53), resulting in

$$\gamma^2 I - X - XH_4 (\gamma^2 I - H_4^T X H_4)^{-1} H_4^T X > 0$$

or

$$X - XH_4 \left(H_4^T X H_4 + (-\gamma^2 I) \right)^{-1} H_4^T X < \gamma^2 I \tag{3.57}$$

Applying (3.15) to (3.57) results in

$$\left(X^{-1} - \gamma^{-2} H_4 H_4^T \right)^{-1} < \gamma^2 I$$

or

$$\left(\gamma^2 X^{-1} - Q_{4,outer} \right)^{-1} < I \tag{3.58}$$

Since $X > 0$, the Schur complement (3.47) of (3.56) is

$$\begin{bmatrix} \gamma^2 I & H_4^T \\ H_4 & X^{-1} \end{bmatrix} > 0 \tag{3.59}$$

Since $\gamma^2 I > 0$, the Schur complement (3.48) of (3.59) is

$$X^{-1} - \gamma^{-2} H_4 H_4^T > 0$$

or

$$\gamma^2 X^{-1} - Q_{4,outer} > 0 \tag{3.60}$$

The inequality (3.60) implies (3.58) can be written as

$$\begin{aligned} I &< \gamma^2 X^{-1} - Q_{4,outer} \\ (I + Q_{4,outer})X &< \gamma^2 I \\ \sqrt{\overline{\lambda}[(I + Q_{4,outer})X]} &< \gamma \end{aligned}$$

or finally

$$\gamma > \sqrt{\overline{\lambda}\left[(I + Q_{4,outer})(Q_1^{-1} + L_u(Q_2 + L_u^T Q_3 L_u)^{-1} L_u^T)^{-1}\right]} \equiv \gamma_{\min_m}$$

The remainder of the proof follows from the proof of Theorem 3.1. □

It is expected that the robust design should produce a lower performance control law than the non-robust design, as the role of the uncertainty in Figure 3.4 is to trade the maximum achieveable performance in return for robustness. Theorem 3.5 formalizes this notion.

Theorem 3.5
If $\gamma_{\min_m}$ is defined as in (3.46), then

$$\gamma_{\min_m} > \gamma_{\min}$$

where $\gamma_{\min}$ is associated with the non-robust design (3.8). □

Proof of Theorem 3.5
Since

$$Q_2 + L_u^T Q_3 L_u > Q_2 > 0$$

then

$$\begin{aligned} Q_2^{-1} &> (Q_2 + L_u^T Q_3 L_u)^{-1} \\ L_u Q_2^{-1} L_u^T &> L_u (Q_2 + L_u^T Q_3 L_u)^{-1} L_u^T \\ Q_1^{-1} + L_u Q_2^{-1} L_u^T &> Q_1^{-1} + L_u (Q_2 + L_u^T Q_3 L_u)^{-1} L_u^T \\ (Q_1^{-1} + L_u (Q_2 + L_u^T Q_3 L_u)^{-1} L_u^T)^{-1} &> (Q_1^{-1} + L_u Q_2^{-1} L_u^T)^{-1} \end{aligned}$$

thus

$$\begin{aligned} \gamma_{\min_m} &= \sqrt{\overline{\lambda}\left[(I + Q_{4,outer})(Q_1^{-1} + L_u (Q_2 + L_u^T Q_3 L_u)^{-1} L_u^T)^{-1}\right]} \\ &> \sqrt{\overline{\lambda}\left[(I + Q_{4,outer})(Q_1^{-1} + L_u Q_2^{-1} L_u^T)^{-1}\right]} \\ &\geq \sqrt{\overline{\lambda}\left[(Q_1^{-1} + L_u Q_2^{-1} L_u^T)^{-1}\right]} \\ &\equiv \gamma_{\min} \end{aligned}$$

that is, $\gamma_{\min_m} > \gamma_{\min}$. □

Finally, Theorem 3.6 relates the robust control law (multiplicative uncertainty) to the non-robust control law.

Theorem 3.6

If the weighting functions associated with the uncertainty block (W_3 and W_4) are set to zero, then the robust $\mathcal{H}_\infty$ control law (multiplicative uncertainty) reduces to the non-robust $\mathcal{H}_\infty$ control law. □

Proof of Theorem 3.6

If $W_3 = 0$, then $H_3 = 0$ and $\Gamma_3 = 0$; if $W_4 = 0$, then $H_4 = 0$ and $\Gamma_4 = 0$. Then the

performance variables (3.49) are

$$\begin{bmatrix} z_{w_1} \\ z_{w_2} \\ z_{w_3} \end{bmatrix} = \begin{bmatrix} H_1 & 0 & -H_1L_u & -H_1L_w & \Gamma_1 & 0 & 0 & 0 \\ 0 & 0 & H_2 & 0 & 0 & \Gamma_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r \\ s \\ u \\ (w_p)_k \\ (x_{w_1})_k \\ (x_{w_2})_k \\ (x_{w_3})_k \\ (x_{w_4})_k \end{bmatrix}$$

$$= \begin{bmatrix} H_1 & -H_1L_u & -H_1L_w & \Gamma_1 & 0 \\ 0 & H_2 & 0 & 0 & \Gamma_2 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r \\ u \\ (w_p)_k \\ (x_{w_1})_k \\ (x_{w_2})_k \end{bmatrix} \tag{3.61}$$

Note that (3.61) is nearly identical to the expression for the performance variables in the non-robust case (3.21). If

$$\underline{x} \stackrel{\triangle}{=} \begin{bmatrix} r \\ u \\ (w_p)_k \\ (x_{w_1})_k \\ (x_{w_2})_k \end{bmatrix}$$

then the robust performance specification (3.51) is given by

$$\min_u \sup_{r,s} \underline{x}^T \begin{bmatrix} Q_1 - \gamma^2 I & -Q_1L_u & -Q_1L_w & H_1^T\Gamma_1 & 0 \\ -L_u^TQ_1 & L_u^TQ_1L_u + Q_2 & L_u^TQ_1L_w & -L_u^TH_1^T\Gamma_1 & H_2^T\Gamma_2 \\ -L_w^TQ_1 & L_w^TQ_1L_u & L_w^TQ_1L_w & -L_w^TH_1^T\Gamma_1 & 0 \\ \Gamma_1^TH_1 & -\Gamma_1^TH_1L_u & -\Gamma_1^TH_1L_w & \Gamma_1^T\Gamma_1 & 0 \\ 0 & \Gamma_2^TH_2 & 0 & 0 & \Gamma_2^T\Gamma_2 \end{bmatrix} \underline{x} - \gamma^2 s^T s \leq 0 \tag{3.62}$$

when the system is at rest at time k. Taking the supremum over s, it is clear that the optimum choice for s is $s = 0$: any other choice would result in a decrease of the left-hand side of (3.62). With $s = 0$, (3.62) is identical to the non-robust control objective (3.22).

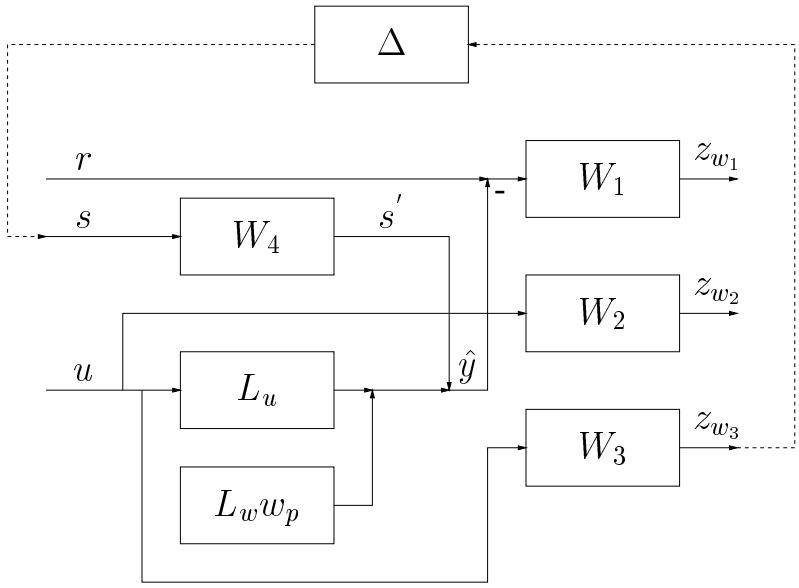

Figure 3.5: Generalized plant for robust $\mathcal{H}_\infty$ control design: additive uncertainty

Similarly when $W_3 = 0$ and $W_4 = 0$, (3.46) is

$$\begin{aligned}\gamma_{\min_m} &= \sqrt{\overline{\lambda}\left[(I + Q_{4,outer})(Q_1^{-1} + L_u(Q_2 + L_u^T Q_3 L_u)^{-1} L_u^T)^{-1}\right]} \\ &= \sqrt{\overline{\lambda}\left[(Q_1^{-1} + L_u Q_2^{-1} L_u^T)^{-1}\right]} \\ &\equiv \gamma_{\min}\end{aligned}$$

It follows that the robust control law of Theorem 3.4 collapses to the non-robust control law of Theorem 3.1 when $W_3 = 0$ and $W_4 = 0$. □

3.4.2 Additive Uncertainty

Figure 3.5 shows the generalized plant of Figure 3.2, modified to include an additive uncertainty Δ. In this case, the predictor takes the form $\widehat{y} = L_w w_p + L_u u + W_4(q)\Delta(q)W_3(q)u$. The robust level-$\gamma$ $\mathcal{H}_\infty$ control design problem is to choose a control u such that the finite horizon $\mathcal{H}_\infty$ gain from $\begin{bmatrix} r^T & s^T \end{bmatrix}^T$ to z is at most magnitude γ. Theorem 3.7 states the central solution to the problem.

Theorem 3.7

Assume the generalized plant of Figure 3.5. If measurements of the plant input (u), plant

output (y), and reference (r) are available for times $\{k-i, \cdots, k-2, k-1\}$*, then the robust, strictly causal, finite horizon, model free, subspace based, level-*γ*, central* $\mathcal{H}_\infty$ *control for times* $\{k, \cdots, k+i-1\}$ *is given by*

$$\begin{bmatrix} r_{wc} \\ s_{wc} \\ u_{opt} \end{bmatrix} = \begin{bmatrix} Q_1 - \gamma^2 I & -Q_1 H_4 & -Q_1 L_u \\ -H_4^T Q_1 & H_4^T Q_1 H_4 - \gamma^2 I & H_4^T Q_1 L_u \\ -L_u^T Q_1 & L_u^T Q_1 H_4 & L_u^T Q_1 L_u + Q_2 + Q_3 \end{bmatrix}^{-1} \cdot$$

$$\begin{bmatrix} Q_1 L_w & -H_1^T \Gamma_1 & 0 & 0 & Q_1 \Gamma_4 \\ -H_4^T Q_1 L_w & H_4^T H_1^T \Gamma_1 & 0 & 0 & -H_4^T Q_1 \Gamma_4 \\ -L_u^T Q_1 L_w & L_u^T H_1^T \Gamma_1 & -H_2^T \Gamma_2 & -H_3^T \Gamma_3 & -L_u^T Q_1 \Gamma_4 \end{bmatrix} \begin{bmatrix} w_p \\ x_{w_1} \\ x_{w_2} \\ x_{w_3} \\ x_{w_4} \end{bmatrix}_k \tag{3.63}$$

provided that

$$\gamma > \gamma_{\min_a} \triangleq \sqrt{\overline{\lambda}\left[(I + Q_{4,outer})(Q_1^{-1} + L_u(Q_2+Q_3)^{-1}L_u^T)^{-1}\right]} \tag{3.64}$$

□

Note that the state space system corresponding to the weighting function W_3 must have dimensions $D_{w_3} \in \mathbb{R}^{l_{w_3} \times m}$. Again, $\gamma_{\min_a} > \gamma_{\min}$ where $\gamma_{\min}$ is associated with the non-robust design (3.8). Also, the robust control law with additive uncertainty reduces to the non-robust control law when $W_3 = 0$ and $W_4 = 0$.

Outline of Proof of Theorem 3.7

The proof of Theorem 3.7 is identical to the proof of Theorem 3.4, with the substitution

$$\begin{bmatrix} z_{w_1} \\ z_{w_2} \\ z_{w_3} \end{bmatrix} = \begin{bmatrix} H_1 & -H_1 H_4 & -H_1 L_u & -H_1 L_w & \Gamma_1 & 0 & 0 & -H_1 \Gamma_4 \\ 0 & 0 & H_2 & 0 & 0 & \Gamma_2 & 0 & 0 \\ 0 & 0 & H_3 & 0 & 0 & 0 & \Gamma_3 & 0 \end{bmatrix} x$$

for the performance variables.

In the additive case

$$X \triangleq (Q_1^{-1} + L_u(Q_2 + Q_3)^{-1}L_u^T)^{-1}$$

□

Theorem 3.7 will be used in Chapter 7, as part of a proposed method of designing robust controllers directly from experimental data.

3.5 Summary

The important results of this chapter are:

- The model free subspace based $\mathcal{H}_\infty$ control law (3.6) – (3.8);
- The non-strictly causal model free subspace based $\mathcal{H}_\infty$ control law (3.38) – (3.41);
- The robust model free subspace based $\mathcal{H}_\infty$ control law with multiplicative uncertainty (3.45), (3.46); and
- The robust model free subspace based $\mathcal{H}_\infty$ control law with additive uncertainty (3.63), (3.64).

Chapter 4

Limiting Cases

With any new control technique, it is important to test various limiting cases in order to ensure that the technique produces answers that are consistent with existing theory. In this chapter, the subspace based $\mathcal{H}_\infty$ controller of Section 3.2 is evaluated in two limiting cases. As the design parameter $\gamma \to \infty$, the $\mathcal{H}_\infty$ controller is shown to recover the model free, subspace based, linear quadratic Gaussian (LQG) control technique of [11]. Furthermore, as the amount of experimental data becomes large ($j \to \infty$), the subspace predictor is shown to recover the Kalman filter properties normally associated with subspace techniques [39]: the subspace based $\mathcal{H}_\infty$ controller behaves like a Kalman filter estimating the plant state, with a model based full information $\mathcal{H}_\infty$ controller utilizing the plant state to produce the control u.

4.1 Subspace Based $\mathcal{H}_\infty$ Controller As $\gamma \to \infty$

As $\gamma \to \infty$, the central finite horizon $\mathcal{H}_\infty$ specification (3.5) converges to an LQ cost [44]. Consider the expression

$$\lim_{\gamma\to\infty} \min_u \sup_r J = \lim_{\gamma\to\infty} \min_u \sup_r \sum_{t=0}^{i-1} \left(z_t^T z_t - \gamma^2 r_t^T r_t \right) \tag{4.1}$$

Informally, as γ becomes very large, the term $-\gamma^2 r_t^T r_t$ begins to dominate the cost function. Thus the optimum choice for the disturbance (r) in order to maximize the cost is to choose

$r_t = 0 \; \forall \; t \in \{0, \cdots, i-1\}$. With this choice, (4.1) becomes

$$\lim_{\gamma \to \infty} \min_u \sup_r J = \min_u \sum_{t=0}^{i-1} z_t^T z_t = \min_u \sum_{t=0}^{i-1} \|(w_1 * y)_t\|_2^2 + \|(w_2 * u)_t\|_2^2$$

which is an LQ cost.

This informal argument suggests the following question: does the subspace $\mathcal{H}_\infty$ controller recover the subspace LQG controller as $\gamma \to \infty$? The answer is in fact, yes. Section 4.1.1 derives the subspace LQG result of [11], while section 4.1.2 compares the subspace based $\mathcal{H}_\infty$ control law to the subspace based LQG control law, demonstrating that the subspace based LQG controller is a special case of the subspace based $\mathcal{H}_\infty$ controller.

4.1.1 Subspace Based Finite Horizon LQG Regulator

The following theorem was first recognized in [11].

Theorem 4.1

If measurements of the plant input (u) and plant output (y) are available for times $\{k-i, \cdots, k-2, k-1\}$*, then the strictly causal, finite horizon, model free, subspace based LQG control for times* $\{k, \cdots, k+i-1\}$ *is*

$$u_{opt} = -(L_u^T Q_{diag} L_u + R_{diag})^{-1} L_u^T Q_{diag} L_w (w_p)_k \tag{4.2}$$

where

$$Q_{diag} \triangleq \begin{bmatrix} Q & 0 & \cdots & 0 \\ 0 & Q & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & Q \end{bmatrix} \in \mathbb{R}^{il \times il} \qquad R_{diag} \triangleq \begin{bmatrix} R & 0 & \cdots & 0 \\ 0 & R & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & R \end{bmatrix} \in \mathbb{R}^{im \times im}$$

$$Q \in \mathbb{R}^{l \times l} \qquad R \in \mathbb{R}^{m \times m}$$

and $Q = Q^T > 0$, $R = R^T > 0$. □

Proof of Theorem 4.1

Consider the finite horizon LQ objective function where the present time is given by k and the future horizon is length i

$$\min_u J = \min_u \sum_{t=k}^{k+i-1} \left(y_t^T Q y_t + u_t^T R u_t \right) \tag{4.3}$$

Note that the system has m inputs ($u_t \in \mathbb{R}^m$) and l outputs ($y_t \in \mathbb{R}^l$). In vector form, (4.3) is

$$\min_u J = \min_u \left(y^T Q_{diag} y + u^T R_{diag} u \right)$$

where $u \in \mathbb{R}^{im}$, $y \in \mathbb{R}^{il}$. Using the strictly causal estimate of y (2.6) from Section 2.1, that is

$$\widehat{y} = L_w (w_p)_k + L_u u$$

the cost J is

$$J = (L_w (w_p)_k + L_u u)^T Q_{diag} (L_w (w_p)_k + L_u u) + u^T R_{diag} u$$

The necessary and sufficient conditions for minimizing J are [6]

$$\frac{\partial J}{\partial u} = 2(L_w (w_p)_k + L_u u)^T Q_{diag} L_u + 2u^T R_{diag} = 0 \tag{4.4}$$

$$\frac{\partial^2 J}{\partial u^2} = 2L_u^T Q_{diag} L_u + 2R_{diag} > 0 \tag{4.5}$$

Since Q_{diag} and R_{diag} are positive, the sufficient condition (4.5) is always satisfied, thus a global minimum exists. The necessary condition (4.4) can be rewritten

$$L_u^T Q L_u u + R_{diag} u = -L_u^T Q_{diag} L_w (w_p)_k$$

which can be solved for u, producing the control law

$$u_{opt} = -(L_u^T Q_{diag} L_u + R_{diag})^{-1} L_u^T Q_{diag} L_w (w_p)_k \tag{4.6}$$

Since $L_u^T Q_{diag} L_u + R_{diag} > 0$ the inverse in (4.6) always exists. □

4.1.2 Subspace Based $\mathcal{H}_\infty$ Control Law As $\gamma \to \infty$

The LQG objective (4.3) is a non-frequency weighted cost function. In order to compare the subspace based $\mathcal{H}_\infty$ control law with the subspace based LQG control law, the $\mathcal{H}_\infty$ weighting functions W_1 and W_2 must be chosen to be frequency independent. This choice is used to formulate Theorem 4.2.

Theorem 4.2

If the weighting functions W_1 and W_2 are chosen to be constants, then as $\gamma \to \infty$, the strictly causal, finite horizon, model free, subspace based, level-γ central $\mathcal{H}_\infty$ control law converges to the finite horizon, model free, subspace based LQG control law. □

Proof of Theorem 4.2

Consider the subspace based $\mathcal{H}_\infty$ control law (3.6)

$$u_{opt} = -(L_u^T \widetilde{Q_1} L_u + Q_2)^{-1} \begin{bmatrix} \left(L_u^T \widetilde{Q_1} L_w\right)^T \\ \left(-L_u^T(\gamma^{-2}\widetilde{Q_1} + I)H_1^T \Gamma_1\right)^T \\ \left(H_2^T \Gamma_2\right)^T \end{bmatrix}^T \begin{bmatrix} w_p \\ x_{w_1} \\ x_{w_2} \end{bmatrix}_k \tag{4.7}$$

$$\widetilde{Q_1} = (Q_1^{-1} - \gamma^{-2} I)^{-1} \tag{4.8}$$

If W_1 and W_2 are constants, then

$$A_1 = 0 \quad B_1 = 0 \quad C_1 = 0$$
$$A_2 = 0 \quad B_2 = 0 \quad C_2 = 0$$

and thus

$$\Gamma_1 = 0 \qquad H_1 = \begin{bmatrix} D_1 & 0 & \cdots & 0 \\ 0 & D_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & D_1 \end{bmatrix} \qquad Q_1 = \begin{bmatrix} D_1^T D_1 & 0 & \cdots & 0 \\ 0 & D_1^T D_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & D_1^T D_1 \end{bmatrix}$$

$$\Gamma_2 = 0 \qquad H_2 = \begin{bmatrix} D_2 & 0 & \cdots & 0 \\ 0 & D_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & D_2 \end{bmatrix} \qquad Q_2 = \begin{bmatrix} D_2^T D_2 & 0 & \cdots & 0 \\ 0 & D_2^T D_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & D_2^T D_2 \end{bmatrix}$$

With this selection of W_1 and W_2, the $\mathcal{H}_\infty$ control law (4.7), (4.8) becomes

$$\begin{aligned} u_{opt} &= -(L_u^T \widetilde{Q_1} L_u + Q_2)^{-1} L_u^T \widetilde{Q_1} L_w (w_p)_k && (4.9) \\ \widetilde{Q_1} &= (Q_1^{-1} - \gamma^{-2} I)^{-1} && (4.10) \end{aligned}$$

Referring to (4.10), as $\gamma \to \infty$

$$\widetilde{Q_1} \to Q_1$$

thus the $\mathcal{H}_\infty$ control law in (4.9) becomes

$$u_{opt} = -(L_u^T Q_1 L_u + Q_2)^{-1} L_u^T Q_1 L_w (w_p)_k \tag{4.11}$$

which is identical to the LQG control law (4.2), with $Q_1 = Q_{diag}$ and $Q_2 = R_{diag}$ or equivalently $D_1^T D_1 = Q$ and $D_2^T D_2 = R$. □

Note that the $\mathcal{H}_\infty$ control law (4.11) does not depend on measured values of r. The lack of dependence can be explained by the following argument. Since the controller is strictly causal, the control law only has access to $\{r_{k-1}, r_{k-2}, \cdots\}$. However, the performance channel z_{w_1} is not dependent on $\{r_{k-1}, r_{k-2}, \cdots\}$ because

$$(z_{w_1})_k = D_1(r_k - y_k)$$

and r_k is independent of $\{r_{k-1}, r_{k-2}, \cdots\}$. The information contained in $\{r_{k-1}, r_{k-2}, \cdots\}$ is not relevant to the controller's efforts to minimize z_{w_1}, and thus measured values of r do not couple into the control law.

4.2 Convergence To Model Based Finite Horizon $\mathcal{H}_\infty$ Control

Recall from Section 2.4, that if the true plant is given by

$$\begin{aligned} x_{k+1} &= A_p x_k + B_p u_k + \zeta_k \\ y_k &= C_p x_k + D_p u_k + \eta_k \end{aligned}$$

where ζ_k and η_k are zero-mean, stationary, ergodic, white noise sequences with covariance

$$E\left[\begin{bmatrix} \zeta_r \\ \eta_r \end{bmatrix}\begin{bmatrix} \zeta_s^T & \eta_s^T \end{bmatrix}\right] = \begin{bmatrix} Q_p & S_p \\ S_p^T & R_p \end{bmatrix}\delta_{rs}$$

and if the data length $j \to \infty$, then the matrices $L_w W_k$ and L_u approach the limits

$$L_w W_k \rightarrow \Gamma_p \widehat{X} \tag{4.12}$$

$$L_u \rightarrow H_p \tag{4.13}$$

Referring to (2.11) and (2.12), the first column of (4.12) can be written

$$L_w (w_p)_k \rightarrow \Gamma_p \widehat{x}_k \tag{4.14}$$

If these limiting results are applied to the subspace based $\mathcal{H}_\infty$ control law, a connection between the model based finite horizon $\mathcal{H}_\infty$ control law and the subspace based $\mathcal{H}_\infty$ control law can be derived. Theorem 4.3 summarizes this result.

Theorem 4.3
When the subspace predictor has the properties (4.13) and (4.14), the $\mathcal{H}_\infty$ control law (4.7), (4.8) is identical to a strictly causal, finite horizon, model based, full information $\mathcal{H}_\infty$ control law acting on a Kalman estimate of the generalize plant state. □

The model based control law [18] is stated in Section 4.2.1. The proof of Theorem 4.3 is given in Section 4.2.2.

4.2.1 Model Based $\mathcal{H}_\infty$ Control

The equations below describe the model based $\mathcal{H}_\infty$ control law, and are algebraic simplifications of those derived in [18]. The generalized plant for the model based full information problem is

$$\begin{bmatrix} x_{k+1} \\ z_k \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}\begin{bmatrix} x_k \\ r_k \\ u_k \end{bmatrix}$$

where

$$A \triangleq \begin{bmatrix} A_p & 0 & 0 \\ -B_1C_p & A_1 & 0 \\ 0 & 0 & A_2 \end{bmatrix} \qquad B \triangleq \begin{bmatrix} 0 & B_p \\ B_1 & -B_1D_p \\ 0 & B_2 \end{bmatrix}$$

$$C \triangleq \begin{bmatrix} -D_1C_p & C_1 & 0 \\ 0 & 0 & C_2 \end{bmatrix} \qquad D \triangleq \begin{bmatrix} D_1 & -D_1D_p \\ 0 & D_2 \end{bmatrix}$$

Also, let

$$R_\gamma \triangleq \begin{bmatrix} \gamma^2 I_l & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{(l+m)\times(l+m)} \tag{4.15}$$

where I_l is the $l \times l$ identity matrix. Define $\{X_{k+i-1}, \cdots, X_k\}$ through the backwards recursion

$$X_{k+i-1} \triangleq 0 \tag{4.16}$$

$$\begin{aligned} X_{t-1} \triangleq\ & A^TX_tA + C^TC - \\ & (A^TX_tB + C^TD)\left(B^TX_tB + D^TD - R_\gamma\right)^{-1}(A^TX_tB + C^TD)^T \\ & \forall\, t \in \{k+i-1, \cdots, k\} \end{aligned} \tag{4.17}$$

Then the worst case disturbances $\{r_k, \cdots, r_{k+i-1}\}$ and the minimum cost controls $\{u_k, \cdots, u_{k+i-1}\}$ can be found from the initial plant state x_k, and the forward recursion

$$\begin{bmatrix} r_t \\ u_t \end{bmatrix} = -\left(B^TX_tB + D^TD - R_\gamma\right)^{-1}(A^TX_tB + C^TD)^Tx_t \tag{4.18}$$

$$x_{t+1} = Ax_t + B\begin{bmatrix} r_t \\ u_t \end{bmatrix} \qquad \forall\, t \in \{k, \cdots, k+i-1\} \tag{4.19}$$

4.2.2 Proof Of Theorem 4.3

The proof of Theorem 4.3 utilizes Theorem 4.4, which was first established in [11]. Theorem 4.4 demonstrates a relation between the subspace based LQG controller, and the model based, finite horizon LQG control law. For a proof of Theorem 4.4, see [11].

Theorem 4.4

Consider the LQG control law (4.6) with the substitutions (4.13) and (4.14)

$$u_{opt} = -(H_p^T Q_{diag} H_p + R_{diag})^{-1} H_p^T Q_{diag} \Gamma_p x_k \tag{4.20}$$

Also consider the finite horizon, discrete time, model based, LQR control law, which is calculated as follows. Define $\{P_{k+i-1}, \cdots, P_k\}$ *through the backwards recursion*

$$\begin{aligned} P_{k+i-1} &\triangleq C_p^T Q C_p & (4.21)\\ P_{t-1} &\triangleq A_p^T P_t A_p + C_p^T Q C_p - (A_p^T P_t B_p + C_p^T Q D_p) \cdot \\ &\quad \left(B_p^T P_t B_p + D_p^T Q D_p + R\right)^{-1} (A_p^T P_t B_p + C_p^T Q D_p)^T & (4.22)\\ &\quad \forall\, t \in \{k+i-1, \cdots, k\} \end{aligned}$$

Then the minimum cost LQR control $\{u_k, \cdots, u_{k+i-1}\}$ *is calculated from the initial plant state* x_k*, and by the forward recursion*

$$\begin{aligned} u_t &= -\left(B_p^T P_t B_p + D_p^T Q D_p + R\right)^{-1} (A_p^T P_t B_p + C_p^T Q D_p)^T x_t & (4.23)\\ x_{t+1} &= A_p x_t + B_p u_t \qquad \forall\, t \in \{k, \cdots, k+i-1\} & (4.24) \end{aligned}$$

Under the conditions that allow the substitutions (4.13) and (4.14), the two control laws are identical i.e.

$$u_{opt} = \begin{bmatrix} u_k \\ u_{k+1} \\ \vdots \\ u_{k+i-1} \end{bmatrix}$$

The control is equivalent to a finite horizon, model based, LQR control law acting on a Kalman estimate of the plant state. □

Note that a key difference between the LQG Riccati equations (4.21), (4.22) and the $\mathcal{H}_\infty$ Riccati equations (4.16), (4.17) is that in (4.22) R is positive, and in (4.17) the term $-R_\gamma$ is negative. In order to prove Theorem 4.3, the expression for the subspace based $\mathcal{H}_\infty$ control law will be manipulated into a form that is identical to the subspace based LQG control law. Theorem 4.4 will then be invoked, thereby proving Theorem 4.3.

Proof of Theorem 4.3

Recall the expression for the performance variable in the subspace based design (3.21) is

$$\begin{bmatrix} z_{w_1} \\ z_{w_2} \end{bmatrix} = \begin{bmatrix} H_1 & -H_1L_u & -H_1L_w & \Gamma_1 & 0 \\ 0 & H_2 & 0 & 0 & \Gamma_2 \end{bmatrix} \begin{bmatrix} r \\ u \\ (w_p)_k \\ (x_{w_1})_k \\ (x_{w_2})_k \end{bmatrix} \tag{4.25}$$

Substituting $\Gamma_p\widehat{x}_k$ for $L_w(w_p)_k$ and H_p for L_u, as per (4.14) and (4.13) respectively, the performance variables (4.25) become

$$\begin{bmatrix} z_{w_1} \\ z_{w_2} \end{bmatrix} = \begin{bmatrix} H_1 & -H_1H_p & -H_1\Gamma_p & \Gamma_1 & 0 \\ 0 & H_2 & 0 & 0 & \Gamma_2 \end{bmatrix} \begin{bmatrix} r \\ u \\ \widehat{x}_k \\ (x_{w_1})_k \\ (x_{w_2})_k \end{bmatrix} \tag{4.26}$$

The next step is the "interlacing" of (4.26). Let

$$w_t \triangleq \begin{bmatrix} r_t \\ u_t \end{bmatrix} \qquad z_t \triangleq \begin{bmatrix} (z_{w_1})_t \\ (z_{w_2})_t \end{bmatrix} \qquad \forall\, t \in \{k, \cdots, k+i-1\}$$

and let

$$w \triangleq \begin{bmatrix} w_k \\ \vdots \\ w_{k+i-1} \end{bmatrix} \in \mathbb{R}^{i(l+m)} \qquad z \triangleq \begin{bmatrix} z_k \\ \vdots \\ z_{k+i-1} \end{bmatrix} \in \mathbb{R}^{i(l_{w_1}+l_{w_2})}$$

$$x \triangleq \begin{bmatrix} \widehat{x}_k \\ (x_{w_1})_k \\ (x_{w_2})_k \end{bmatrix} \in \mathbb{R}^{n_p+n_{w_1}+n_{w_2}}$$

Let $\{\bullet\}_{p:q,r:s}$ be the submatrix of $\bullet$ formed by the entries in rows $\{p, p+1, \cdots, q\}$ and the entries in columns $\{r, r+1, \cdots, s\}$. Define

$$H \triangleq \begin{bmatrix} \widetilde{H}_{11} & \widetilde{H}_{12} & \cdots & \widetilde{H}_{1i} \\ \widetilde{H}_{21} & \widetilde{H}_{22} & \cdots & \widetilde{H}_{2i} \\ \vdots & \vdots & \ddots & \vdots \\ \widetilde{H}_{i1} & \widetilde{H}_{i2} & \cdots & \widetilde{H}_{ii} \end{bmatrix}$$

$$\Gamma \triangleq \begin{bmatrix} \{-H_1\Gamma_p\}_{1:l_{w_1},1:n_p} & \{\Gamma_1\}_{1:l_{w_1},1:n_{w_1}} & 0 \\ 0 & 0 & \{\Gamma_2\}_{1:l_{w_2},1:n_{w_2}} \\ \{-H_1\Gamma_p\}_{l_{w_1}+1:2l_{w_1},1:n_p} & \{\Gamma_1\}_{l_{w_1}+1:2l_{w_1},1:n_{w_1}} & 0 \\ 0 & 0 & \{\Gamma_2\}_{l_{w_2}+1:2l_{w_2},1:n_{w_2}} \\ \vdots & \vdots & \vdots \\ \{-H_1\Gamma_p\}_{(i-1)l_{w_1}+1:il_{w_1},1:n_p} & \{\Gamma_1\}_{(i-1)l_{w_1}+1:il_{w_1},1:n_{w_1}} & 0 \\ 0 & 0 & \{\Gamma_2\}_{(i-1)l_{w_2}+1:il_{w_2},1:n_{w_2}} \end{bmatrix}$$

where

$$\widetilde{H}_{11} \triangleq \begin{bmatrix} \{H_1\}_{1:l_{w_1},1:l} & \{-H_1H_p\}_{1:l_{w_1},1:m} \\ 0 & \{H_2\}_{1:l_{w_2},1:m} \end{bmatrix}$$

$$\widetilde{H}_{12} \triangleq \begin{bmatrix} \{H_1\}_{1:l_{w_1},l+1:2l} & \{-H_1H_p\}_{1:l_{w_1},m+1:2m} \\ 0 & \{H_2\}_{1:l_{w_2},m+1:2m} \end{bmatrix}$$

$$\widetilde{H}_{1i} \triangleq \begin{bmatrix} \{H_1\}_{1:l_{w_1},(i-1)l+1:il} & \{-H_1H_p\}_{1:l_{w_1},(i-1)m+1:im} \\ 0 & \{H_2\}_{1:l_{w_2},(i-1)m+1:im} \end{bmatrix}$$

$$\widetilde{H}_{21} \triangleq \begin{bmatrix} \{H_1\}_{l_{w_1}+1:2l_{w_1},1:l} & \{-H_1H_p\}_{l_{w_1}+1:2l_{w_1},1:m} \\ 0 & \{H_2\}_{l_{w_2}+1:2l_{w_2},1:m} \end{bmatrix}$$

$$\widetilde{H}_{22} \triangleq \begin{bmatrix} \{H_1\}_{l_{w_1}+1:2l_{w_1},l+1:2l} & \{-H_1H_p\}_{l_{w_1}+1:2l_{w_1},m+1:2m} \\ 0 & \{H_2\}_{l_{w_2}+1:2l_{w_2},m+1:2m} \end{bmatrix}$$

$$\widetilde{H}_{2i} \triangleq \begin{bmatrix} \{H_1\}_{l_{w_1}+1:2l_{w_1},(i-1)l+1:il} & \{-H_1H_p\}_{l_{w_1}+1:2l_{w_1},(i-1)m+1:im} \\ 0 & \{H_2\}_{l_{w_2}+1:2l_{w_2},(i-1)m+1:im} \end{bmatrix}$$

$$\widetilde{H}_{i1} \triangleq \begin{bmatrix} \{H_1\}_{(i-1)l_{w_1}+1:il_{w_1},1:l} & \{-H_1H_p\}_{(i-1)l_{w_1}+1:il_{w_1},1:m} \\ 0 & \{H_2\}_{(i-1)l_{w_2}+1:il_{w_2},1:m} \end{bmatrix}$$

$$\widetilde{H}_{i2} \triangleq \begin{bmatrix} \{H_1\}_{(i-1)l_{w_1}+1:il_{w_1},l+1:2l} & \{-H_1H_p\}_{(i-1)l_{w_1}+1:il_{w_1},m+1:2m} \\ 0 & \{H_2\}_{(i-1)l_{w_2}+1:il_{w_2},m+1:2m} \end{bmatrix}$$

$$\widetilde{H}_{ii} \triangleq \begin{bmatrix} \{H_1\}_{(i-1)l_{w_1}+1:il_{w_1},(i-1)l+1:il} & \{-H_1H_p\}_{(i-1)l_{w_1}+1:il_{w_1},(i-1)m+1:im} \\ 0 & \{H_2\}_{(i-1)l_{w_2}+1:il_{w_2},(i-1)m+1:im} \end{bmatrix}$$

Then the "interlaced" version of (4.26) is

$$z = \begin{bmatrix} H & \Gamma \end{bmatrix} \begin{bmatrix} w \\ x \end{bmatrix} \tag{4.27}$$

Define

$$R_{\gamma d} \triangleq \begin{bmatrix} R_\gamma & 0 & \cdots & 0 \\ 0 & R_\gamma & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & R_\gamma \end{bmatrix} \in \mathbb{R}^{i(l+m)\times i(l+m)} \tag{4.28}$$

where R_γ is defined by (4.15). Using (4.27) and (4.28), equation (3.18) is

$$J(\gamma) = \begin{bmatrix} w \\ x \end{bmatrix}^T \begin{bmatrix} H^TH & H^T\Gamma \\ \Gamma^TH & \Gamma^T\Gamma \end{bmatrix} \begin{bmatrix} w \\ x \end{bmatrix} - w^TR_{\gamma d}w$$

With this $J(\gamma)$, the necessary condition for the minimax optimization is

$$\begin{bmatrix} H^TH - R_{\gamma d} & H^T\Gamma \end{bmatrix} \begin{bmatrix} w \\ x \end{bmatrix} = 0$$

which implies that the control law and worst case disturbance (embedded in w) is

$$w = \left(H^TH - R_{\gamma d}\right)^{-1} H^T\Gamma x \tag{4.29}$$

Equation (4.29) is identical to the subspace based LQG control law (4.20), with the substitutions

$$\begin{aligned} u_{opt} &= w \\ H_p &= H \\ Q_{diag} &= I \end{aligned}$$

$$\begin{aligned} R_{diag} &= -R_{\gamma d} \\ \Gamma_p &= \Gamma \\ x_k &= x \end{aligned}$$

Furthermore (4.16) – (4.19) are identical to (4.21) – (4.24) with the substitutions

$$\begin{aligned} P_{k+i-1} &= 0 \\ A_p &= A \\ B_p &= B \\ C_p &= C \\ D_p &= D \\ Q &= I \\ R &= -R_\gamma \\ P_t &= X_t \quad \forall \quad t \in \{k, \cdots, k+i-1\} \\ u_t &= \begin{bmatrix} r_t \\ u_t \end{bmatrix} \quad \forall \quad t \in \{k, \cdots, k+i-1\} \end{aligned}$$

Thus with these substitutions, it follows from Theorem 4.4 that (4.16) – (4.19) result in identical $w^T = \begin{bmatrix} r_t^T & u_t^T \end{bmatrix}$ as calculated by (4.29). Since (4.29) is a permutation of (4.7), (4.8), then (4.16) – (4.19) result in identical r and u as that calculated in (4.7), (4.8). □

4.3 Summary

The important results of this chapter are:

- As $\gamma \to \infty$, the model free subspace based $\mathcal{H}_\infty$ control law converges to the model free subspace based LQG control law; and
- If the true plant is LTI, as $j \to \infty$ the model free subspace based $\mathcal{H}_\infty$ controller behaves like a Kalman filter estimating the plant state, with a model based full information $\mathcal{H}_\infty$ controller utilizing the plant state to produce the control u.

Chapter 5

Controller Implementation

This chapter considers the issues associated with the implementation of the subspace based $\mathcal{H}_\infty$ controllers derived in Chapter 3. Both off-line (batch) and on-line (adaptive) implementations will be considered. Although the discussion will focus on the model free controller of Section 3.2, the methods developed are applicable to the other $\mathcal{H}_\infty$ control laws derived in Chapter 3, as well as to the subspace based LQG control law derived in Chapter 4.

A design example employing the model free $\mathcal{H}_\infty$ controller of Section 3.2 is used to provide some insight into the use of the various design parameters available to the engineer. The variation of the control law with respect to the engineer's selection of i, j, γ, W_1, W_2 are illustrated in this simple design example.

5.1 Receding Horizon Implementation

A finite horizon controller is most often implemented with a "receding horizon" procedure. Consider a future horizon of length i. At the present time k, the optimal control is calculated for times $\{k, k+1, \cdots, k+i-1\}$. The control at time k is implemented, and data y_k and r_k are collected. The "horizon" is shifted one time step into the future, and the procedure is repeated.

The following algorithm lists all the steps required to apply the receding horizon procedure to the implementation of the model free subspace based $\mathcal{H}_\infty$ control problem.

1. Collect experimental data: form U_p, U_f, Y_p, Y_f
2. Compute L_w, L_u

3. Select (possibly many) weighting functions W_1, W_2
4. Compute $\gamma_{\min}$ for each set of weighting functions: finalize the choice of W_1, W_2, and choose $\gamma > \gamma_{\min}$
5. Initialize the weighting function states $(x_{w_1})_k$, $(x_{w_2})_k$
6. Form $(w_p)_k$
7. Calculate the optimal control u_{opt} for times $\{k, k+1, \cdots, k+i-1\}$

$$u_{opt} = f\left(W_1, W_2, L_w, L_u, \gamma, \begin{bmatrix} w_p \\ x_{w_1} \\ x_{w_2} \end{bmatrix}_k\right)$$

   using (3.6), (3.7)
8. Implement the first time step of u_{opt} (time k)
9. Take measurements r_k and y_k
10. Update weighting filter states

$$\begin{aligned} x_{w_1} &:= A_{w_1} x_{w_1} + B_{w_1}(r_k - y_k) \\ x_{w_2} &:= A_{w_2} x_{w_2} + B_{w_2} u_k \end{aligned}$$

11. $k := k+1$, go to step 6

A few comments on the steps of the implementation:

1. The design of the excitation signal requires some care. The signal must sufficiently excite the plant so that all important aspects of the plant behavior appear in the plant's output. If the plant is indeed LTI, then one sufficient condition for the design of the input signal is that the input be persistently exciting of order $2mi$, that is

$$\begin{bmatrix} U_p \\ U_f \end{bmatrix} \begin{bmatrix} U_p \\ U_f \end{bmatrix}^T$$

should be rank $2mi$ [39]. Further discussion of appropriate excitation signals can be found in [25, 28].

2. The computation of L_w and L_u can be performed in a batch process, or can be performed recursively. The memory requirement for recursive calculation can be significantly less than for a batch process.

3. It is essential that W_1 and W_2 have relative McMillan degree zero.

4. It is usually good practice not to choose γ very close to $\gamma_{\min}$, as the effective controller gains are very large: the formula for calculating u_{opt} is close to singular near $\gamma_{\min}$. As a rule of thumb, choose $\gamma > (1.1)\gamma_{\min}$.

5. The initial states of the weighting functions can be set to 0 if input, output, and reference data immediately prior to time k are not available. If the data are available, it is possible to find the initial states of the weighting filters by initializing the weighting filter states to 0, and then propagating the filters forward to the present time using the available data.

6. If the input-output data immediately prior to time k are not known during the first pass through this algorithm, the initial $(w_p)_k$ can be set to 0.

7. The majority of the matrix operations necessary to compute u_{opt} can be performed prior to this step.

8. Steps 8 – 11 are straightforward.

Note that the control law is strictly causal: u_k is not a function of r_k nor y_k. The measurements of r_k and y_k couple into future u_{opt} through the state of the performance weighting function (x_{w_1}) in step 10.

5.2 Simplified Receding Horizon Implementation

The important observation made in this section is that steps 5 – 11 of the receding horizon implementation in Section 5.1 can be expressed as a linear time-invariant (LTI) discrete time system. Steps 1 – 4 remain, and contain the design steps necessary to develop the

controller. Consider the expression for the non-robust $\mathcal{H}_\infty$ control law (3.6), (3.7),

$$u_{opt} = -(L_u^T \widetilde{Q_1} L_u + Q_2)^{-1} \begin{bmatrix} \left(L_u^T \widetilde{Q_1} L_w\right)^T \\ \left(-L_u^T(\gamma^{-2}\widetilde{Q_1} + I)H_1^T \Gamma_1\right)^T \\ \left(H_2^T \Gamma_2\right)^T \end{bmatrix}^T \begin{bmatrix} w_p \\ x_{w_1} \\ x_{w_2} \end{bmatrix}_k \tag{5.1}$$

$$\widetilde{Q_1} = (Q_1^{-1} - \gamma^{-2} I)^{-1} \tag{5.2}$$

Let

$$\begin{aligned} \begin{bmatrix} k_1 & k_2 \end{bmatrix} &= \left\{-(L_u^T \widetilde{Q_1} L_u + Q_2)^{-1} L_u^T \widetilde{Q_1} L_w\right\}_{1:m,:} \\ k_3 &= \left\{(L_u^T \widetilde{Q_1} L_u + Q_2)^{-1} L_u^T (\gamma^{-2}\widetilde{Q_1} + I) H_1^T \Gamma_1\right\}_{1:m,:} \\ k_4 &= \left\{-(L_u^T \widetilde{Q_1} L_u + Q_2)^{-1} H_2^T \Gamma_2\right\}_{1:m,:} \end{aligned}$$

where $\{\bullet\}_{1:m,:}$ means extract the first m rows of the matrix $\bullet$, and $k_1 \in \mathbb{R}^{m \times im}$, $k_2 \in \mathbb{R}^{m \times il}$, $k_3 \in \mathbb{R}^{m \times n_{w_1}}$, $k_4 \in \mathbb{R}^{m \times n_{w_2}}$. If available, initialize the vector

$$\begin{bmatrix} u_p \\ y_p \\ x_{w_1} \\ x_{w_2} \end{bmatrix}_k$$

based on input, output, and reference data, otherwise initialize the vector to 0. Then steps 5 – 11 of the receding horizon implementation can be performed by the discrete time LTI system

$$\begin{bmatrix} u_p \\ y_p \\ x_{w_1} \\ x_{w_2} \end{bmatrix}_{k+1} = \begin{bmatrix} \begin{bmatrix} S_m \\ k_1 \end{bmatrix} & \begin{bmatrix} 0 \\ k_2 \end{bmatrix} & \begin{bmatrix} 0 \\ k_3 \end{bmatrix} & \begin{bmatrix} 0 \\ k_4 \end{bmatrix} \\ 0 & \begin{bmatrix} S_l \\ 0 \end{bmatrix} & 0 & 0 \\ 0 & 0 & A_{w_1} & 0 \\ B_{w_2} k_1 & B_{w_2} k_2 & B_{w_2} k_3 & A_{w_2} + B_{w_2} k_4 \end{bmatrix} \begin{bmatrix} u_p \\ y_p \\ x_{w_1} \\ x_{w_2} \end{bmatrix}_k +$$

$$\begin{bmatrix} 0 & 0 \\ 0 & \begin{bmatrix} 0 \\ I_l \end{bmatrix} \\ B_{w_1} & -B_{w_1} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} r \\ y \end{bmatrix}_k \tag{5.3}$$

$$u_k = \begin{bmatrix} k_1 & k_2 & k_3 & k_4 \end{bmatrix} \begin{bmatrix} u_p \\ y_p \\ x_{w_1} \\ x_{w_2} \end{bmatrix}_k \tag{5.4}$$

where I_m and I_l are $m \times m$ and $l \times l$ identity matrices respectively, and

$$S_m = \begin{bmatrix} 0 & I_m & 0 & \cdots & 0 \\ 0 & 0 & I_m & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \cdots & I_m \end{bmatrix} \in \mathbb{R}^{(i-1)m \times im}$$

$$S_l = \begin{bmatrix} 0 & I_l & 0 & \cdots & 0 \\ 0 & 0 & I_l & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \cdots & I_l \end{bmatrix} \in \mathbb{R}^{(i-1)l \times il}$$

Note that the control law (5.3), (5.4) is an $i(m+l) + n_{w_1} + n_{w_2}$ order LTI controller.

Expressing the subspace based controller as an LTI system enables the use of well known LTI analysis tools for evaluation of the control law. If an approximate plant model is available, approximations of gain and phase margin may be obtained from Nyquist or Bode plots. If the system is SISO and an approximate plant model is available, root loci can be constructed. Expressing the controller as an LTI system also greatly simplifies any coding required for implementing or simulating the overall closed loop system.

Section 5.4 uses the LTI form of the subspace based control law to examine the pole and zero locations of several control laws, while Section 6.2 uses Bode plots to directly compare two control laws. Approximate plant models are not used in these two sections.

5.2.1 Summary Of Batch Design Procedure

The control design procedure can be viewed as a "black box" requiring the following steps by the engineer:

1. Collect experimental data
2. Select i, j. Select (possibly many) weighting functions W_1, W_2
3. Compute $\gamma_{\min}$ for each set of weighting functions: finalize the choice of W_1, W_2, and choose $\gamma > \gamma_{\min}$
4. Compute and implement the final $i(m+l) + n_{w_1} + n_{w_2}$ order LTI control law (5.3), (5.4)

The design example in Section 5.4 illustrates the effects of each of the engineer selected parameters (W_1, W_2, i, j and γ) in the context of a simple control design problem.

5.3 Adaptive Implementation

The receding horizon implementation of Section 5.2 also simplifies the adaptive implementation of the subspace based $\mathcal{H}_\infty$ control law. The block diagram in Figure 5.1 outlines the information flow among the three major components: the plant P, the controller K, and the adaptive algorithm *Adapt*. The K block is identical to the state space system (5.3), (5.4) with the exception that the matrices $k_1, \cdots, k_4$ are frequently updated by the *Adapt* block. Because of this updating, K is a linear time-varying (LTV) system. K's system state is composed of past input-output data and internal states of the weighting filters, thus the matrices $k_1, \cdots, k_4$ can be modified by the *Adapt* block without transforming the controller's state. The *Adapt* block in Figure 5.1 performs several simple operations:

1. The best estimates of L_w, L_u are computed by the update procedure described in Chapter 2
2. The latest estimate of L_w, L_u is used to compute the new estimate of best achieveable performance $\gamma_{\min}$
3. $\gamma > \gamma_{\min}$ is selected

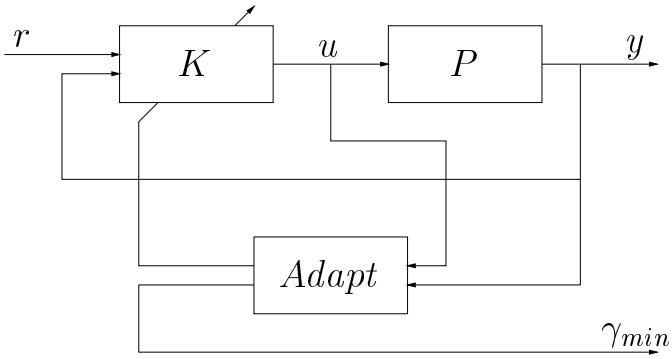

Figure 5.1: Adaptive implementation block diagram

4. The new control gains $k_1, \cdots, k_4$ are calculated via (5.1), (5.2) and passed onto the controller K

Since the K block and the *Adapt* block can operate independently, different computers could be used for each task. The data link between the two computers need only be fast enough to satisfy the minimum desired update latency.

The processes performed by the *Adapt* block leave several design choices available to the engineer. The prediction horizon i, and the weighting filters W_1, W_2 must of course be selected. j, the size of the window used to compute L_w, L_u must be selected if the "sliding window" procedures described in Chapter 2 are used. Computation power may be a key factor in determining the frequency at which a new controller will be computed. If the computational resources do not permit adaptation at every time step, it is possible to perform n Cholesky updates before computing L_w, L_u, and the new control gains.

The policy used to select γ requires some engineering judgment. Many policies are possible, such as $\gamma = \gamma_{\min} + \alpha$ or $\gamma = (1 + \alpha)\gamma_{\min}$ where $\alpha > 0$. It is possible to have other "outer loops" such as changing the design weighting filters W_1 and W_2 based on the measured $\gamma_{\min}$. One approach to outer loop W_1, W_2 design is to use the so-called "wind surfer" adaptation policies [24]. The general idea is to begin by implementing a low bandwidth controller. If the requested performance is achieved, the controller bandwidth is progressively increased until the desired bandwidth is achieved. If at any step, the requested bandwidth not achieved, the controller bandwidth is reduced, thereby allowing the plant to remain under control, while potentially useful information is gathered. The adaptive system continually advances toward, and retreats from the final control bandwidth goal.

5.3.1 Summary Of Adaptive Design Procedure

The adaptive design procedure can be viewed as a "black box" requiring the following steps by the engineer:

1. Select i (the prediction horizon), and possibly j (the sliding window size)
2. Select weighting functions W_1, W_2
3. Select a γ policy, such as $\gamma = (1.1)\gamma_{\min}$
4. Implement the adaptive controller as in Figure 5.1

5.4 Design Example

The effects of the engineer's selections of W_1, W_2, i, j, and γ are illustrated through a series of simulations using the simple plant

$$P(s) \stackrel{\triangle}{=} \frac{1}{s(s+1)} \tag{5.5}$$

Plants similar to (5.5) are often used to approximate actuators deployed within a servo control loop: the widespread use of such plant models motivates the study of (5.5). Since the model free subspace based $\mathcal{H}_\infty$ design technique is a discrete time method, (5.5) is discretized using a sample frequency of 10 Hz: well above the plant's cutoff frequency near 0.3 Hz. The zero order hold discretization of (5.5) is

$$P_d(z) \stackrel{\triangle}{=} \frac{0.004837z + 0.004679}{z^2 - 1.905z + 0.9048} \tag{5.6}$$

In order to better illustrate the intuition behind selecting W_1 and W_2, these weighting functions are designed in the continuous domain, and then discretized using the Tustin (or bilinear) transform [15].

This section does not consider the effects of changes in the excitation signals used to obtain the experimental data: the topic is well covered in standard texts on system identification, such as [25]. In order to eliminate variation of results due to changes in "experimental" data, a single data set, obtained from a noisy simulation of (5.6) is used for all control designs. The data set is generated by exciting (5.6) with a pseudo-random binary

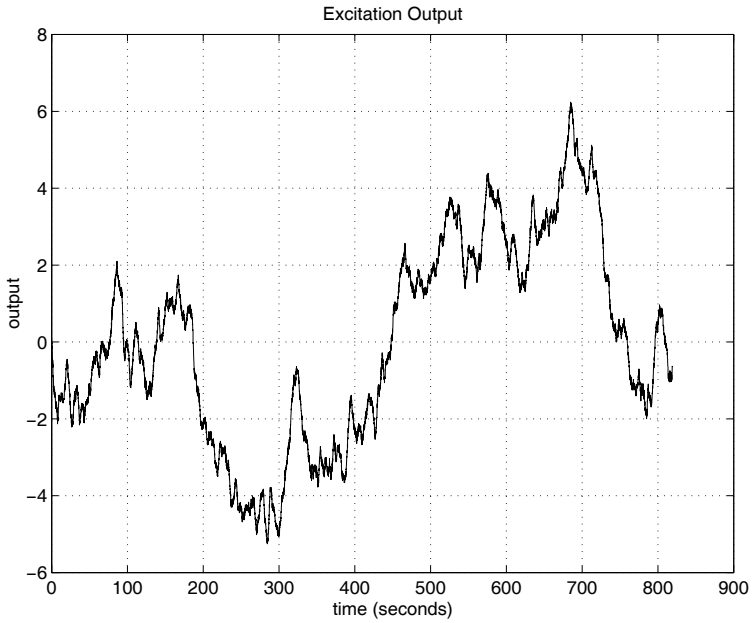

Figure 5.2: Output data used for design examples

sequence of unity power. Both white Gaussian process noise ($\sigma = 0.05$) and white Gaussian sensor noise ($\sigma = 0.05$) are injected into this simulation. These noises can also be included in the closed loop simulations without significantly changing the performance, however, in order to better illustrate the true effects of the various selections made by the designer, the closed loop simulations are performed without the addition of noise. Figure 5.2 shows the output of the input-output data used throughout this section. The input data are omitted because the input is simply a sequence of $+1$'s and -1's.

The next four subsections consider the nominal control design

$$i = 20 \qquad j = 1000 \qquad \gamma = (1.1)\gamma_{\min}$$
$$W_1 = 100\left(\tfrac{0.01}{5}\right)\tfrac{s+5}{s+0.01} \qquad W_2 = 0.05\left(\tfrac{10}{0.5}\right)\tfrac{s+0.5}{s+10}$$

and investigate changes in the closed loop behavior that result from changes in each of the five design parameters. In most cases, the rule $\gamma = (1.1)\gamma_{\min}$ is used to select γ. For the nominal design, $\gamma_{\min} = 0.880$.

	W_1	W_2	$\gamma_{\min}$	$\gamma = (1.1)\gamma_{\min}$
Design 1	W_{11}	W_{21}	0.880	0.968
Design 2	W_{12}	W_{21}	1.493	1.642
Design 3	W_{12}	W_{22}	1.347	1.482

Table 5.1: Weighting functions and $\gamma_{\min}$, γ associated with various W_1, W_2 designs

5.4.1 Effects Of W_1, W_2

Performance specification by loop shaping is an established design methodology in the control literature [5], thus this section will only provide a cursory illustration of the tradeoffs possible with loop shaping functions in the context of model free subspace based $\mathcal{H}_\infty$ control. A secondary consideration when designing the dynamics of W_1 and W_2 is that the controller order $(i(m+l)+n_{w_1}+n_{w_2})$ is a function of the number of modes in the weighting functions.

Four weighting functions are used in this section

$$W_{11} \triangleq 100\left(\tfrac{0.01}{5}\right)\tfrac{s+5}{s+0.01} \quad W_{21} \triangleq 0.05\left(\tfrac{10}{0.5}\right)\tfrac{s+0.5}{s+10}$$
$$W_{12} \triangleq 100\left(\tfrac{0.02}{10}\right)\tfrac{s+10}{s+0.02} \quad W_{22} \triangleq 0.05\left(\tfrac{20}{1}\right)\tfrac{s+1}{s+20}$$

Table 5.1 shows each design case studied in this subsection. Also included in Table 5.1 are the $\gamma_{\min}$ and γ associated with each design. Design 1 is the nominal design, Design 2 increases the desired closed loop tracking bandwidth without changing the allowed control usage, while Design 3 requests the same tracking bandwidth as Design 2 but increases the allowed control usage. Figure 5.3 shows the inverses of these weighting functions after conversion to discrete time.

As expected, $\gamma_{\min}$ for Design 2 is significantly worse than $\gamma_{\min}$ for Design 1 because greater tracking performance is requested without allowing greater control usage. The controller is unable to meet the performance at the nominal performance level ($\gamma_{\min} = 0.880$), which is reflected in the higher $\gamma_{\min}$. The step response of Design 2 is faster yet more poorly damped than the step response of Design 1, as shown in Figure 5.4.

Design 3 requests the same tracking performance as Design 2, however Design 3 permits greater control usage than Design 1. The $\gamma_{\min}$ associated with Design 3 is less than the $\gamma_{\min}$ associated with Design 2, which is reflected in the faster step response of Design 3.

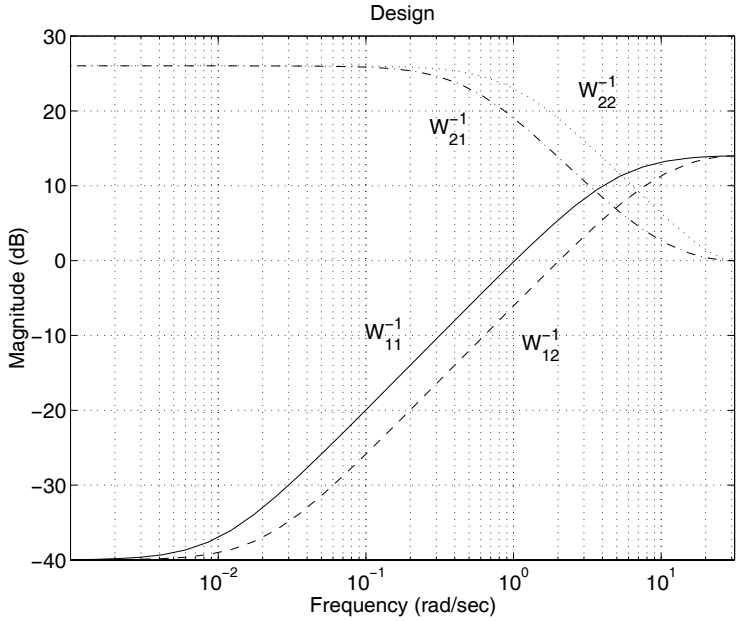

Figure 5.3: Discrete time weighting functions used in the design example

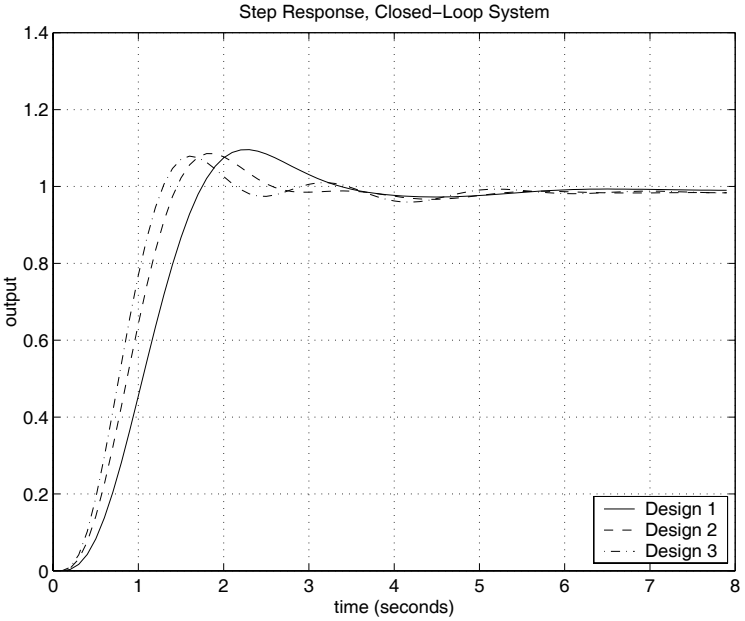

Figure 5.4: Step responses for various W_1, W_2, designs

	$\gamma_{\min}$	$\gamma = (1.1)\gamma_{\min}$
$i = 300$	0.919	1.011
$i = 80$	0.966	1.062
$i = 25$	0.941	1.035
$i = 20$	0.880	0.968
$i = 13$	0.832	0.915
$i = 10$	0.756	0.831
$i = 8$	0.655	0.721

Table 5.2: $\gamma_{\min}$, γ for various choices of i

5.4.2 Effect Of i

This subsection illustrates the changes in the controller and the closed loop performance that result from changes in the design parameter i. Compared to other design variables, this design parameter also has the single greatest effect on controller order ($i(m+l)+n_{w_1}+n_{w_2}$). The engineer's ability to vary the number of plant outputs (l) or inputs (m), or vary the number of states in either weighting function (n_{w_1} or n_{w_2}) is usually quite limited, whereas i can be varied over quite a wide range. Thus choice of i is primarily a trade between control performance and the controller order.

Table 5.2 shows the $\gamma_{\min}$ associated with each of the seven values of i considered in this subsection. Note that although $j = 1000$ for each of the test cases, $n = 2i + j - 1$, thus each test case uses a slightly different amount of experimental data. This variation in the data used might account for some small fluctuations in $\gamma_{\min}$, however the variations in data volume are small compared to the overall data set size for all but the largest values of i.

The general trend is that $\gamma_{\min}$ becomes smaller as i is decreased. Figure 5.5 shows the step responses for the $i = 80$ through $i = 8$ test cases, with $\gamma/\gamma_{\min}$ constant. In addition, Figure 5.6 shows the step responses with constant $\gamma = 1.1$. In both cases, the $i = 300$ case is omitted, as its performance is nearly identical to the $i = 80$ case.

Shortening the prediction horizon, *i.e.* decreasing i, requires the control law to decrease how far into the future it projects the cost of the present system state and the cost of all possible control actions over that future horizon. Thus the more short sighted the controller is, the more optimistic the controller can be about future performance. This explains the decrease in $\gamma_{\min}$ as i decreases. Figure 5.5 and Figure 5.6 show that as the prediction horizon is shortened, the closed loop performance degrades until the system eventually goes unstable. Both figures are needed to argue this conclusion, because if only constant $\gamma/\gamma_{\min}$

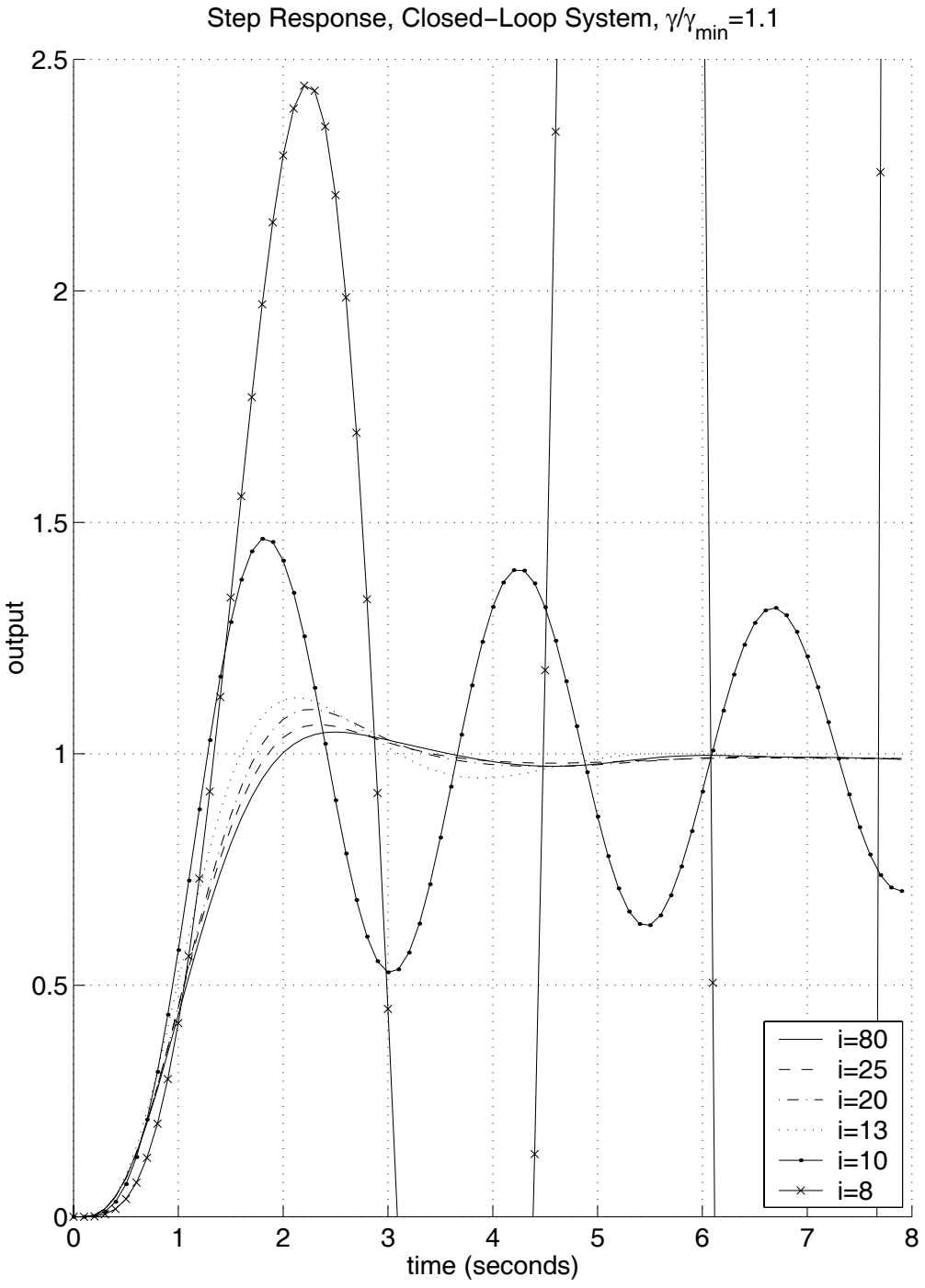

Figure 5.5: Step responses for various choices of i, constant $\gamma/\gamma_{\min}$

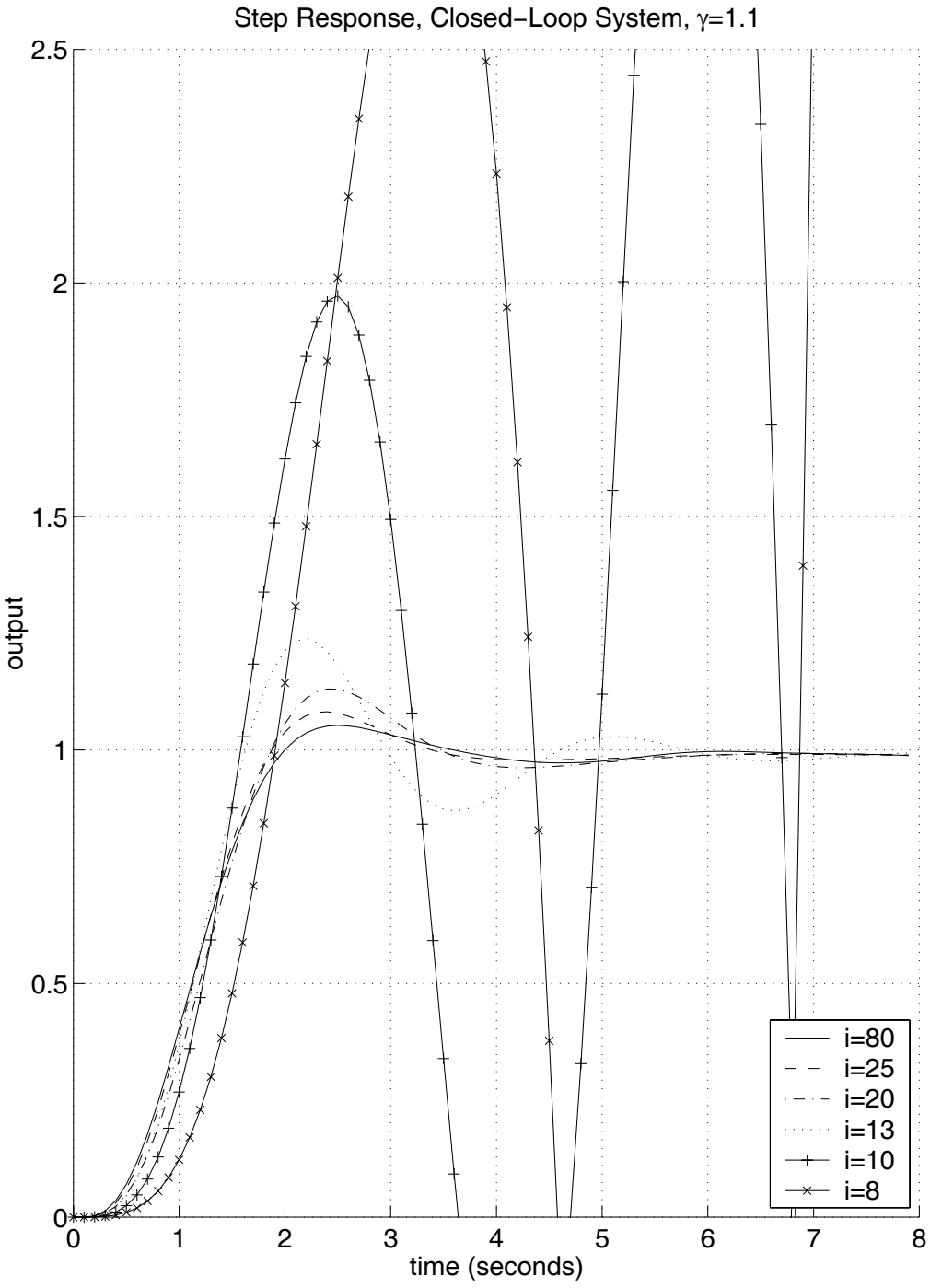

Figure 5.6: Step responses for various choices of i, constant γ

	$\gamma_{\min}$	$\gamma = (1.1)\gamma_{\min}$
$i = 300$	N/A	N/A
$i = 80$	0.968	1.065
$i = 25$	0.941	1.036
$i = 20$	0.880	0.968
$i = 13$	0.831	0.915
$i = 10$	0.756	0.831
$i = 8$	0.655	0.720

Table 5.3: $\gamma_{\min}$, γ for various choices of i, constant n

were used, one could argue that the performance degradation is a result of inappropriately low values of γ due to optimistically low estimates of $\gamma_{\min}$. The smallest i that produces good performance is approximately $i = 20$, which corresponds to a prediction time of 2 seconds: roughly the rise time of the closed loop system.

Table 5.3, Figure 5.7, and Figure 5.8 show the results when varying i with n fixed at the nominal design condition $n = 1039$. Of course j is then selected according to the relation $j = n + 1 - 2i$. This example represents what might be considered the more typical use of the batch design technique, where the engineer is given a fixed volume of experimental data and asked to design a control law. In the $i = 300$ case, the problem of finding a predictor is underdetermined thus a design is not possible. The conclusions that can be drawn from the constant n results are identical to the conclusions reached when j is constant.

Figure 5.9 shows the loci of poles and zeros for each of the seven design cases. Comparing the loci, it is apparent that as i is increased, two circular patterns of poles and zeros become more defined, move closer to each other, and move closer to the unit circle. When the poles and zeros become sufficiently close, it is possible to remove them using a model order reduction technique at the price of a small change in controller performance [18].

The approach of computing a very high order controller then proceding with a model order reduction may be difficult to use when the experimental data includes time delays, nonlinearities, or noise. Figure 5.10 illustrates the loci for $i = 300$ juxtaposed with the loci for $i = 300$ when the experimental data are taken from a *noise free* simulation. When noisy experimental data are used, the poles and zeros are much further from each other than when noise free experimental data are used. In some cases where real world experimental data are used, it may be impractical to choose i sufficiently large such that controller order reduction may be performed without adversely effecting closed loop performance.

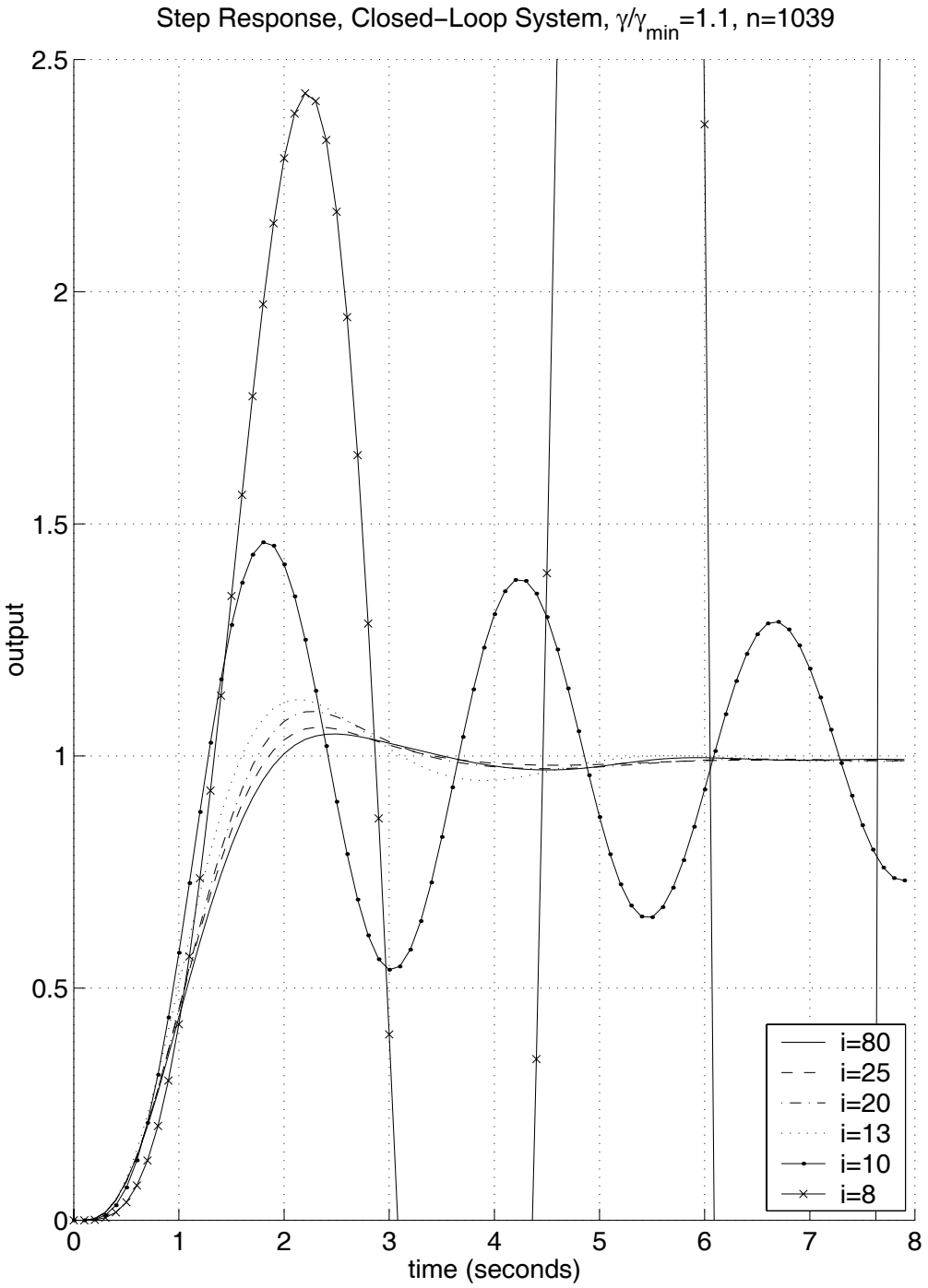

Figure 5.7: Step responses for various choices of i, constant $\gamma/\gamma_{\min}$, constant n

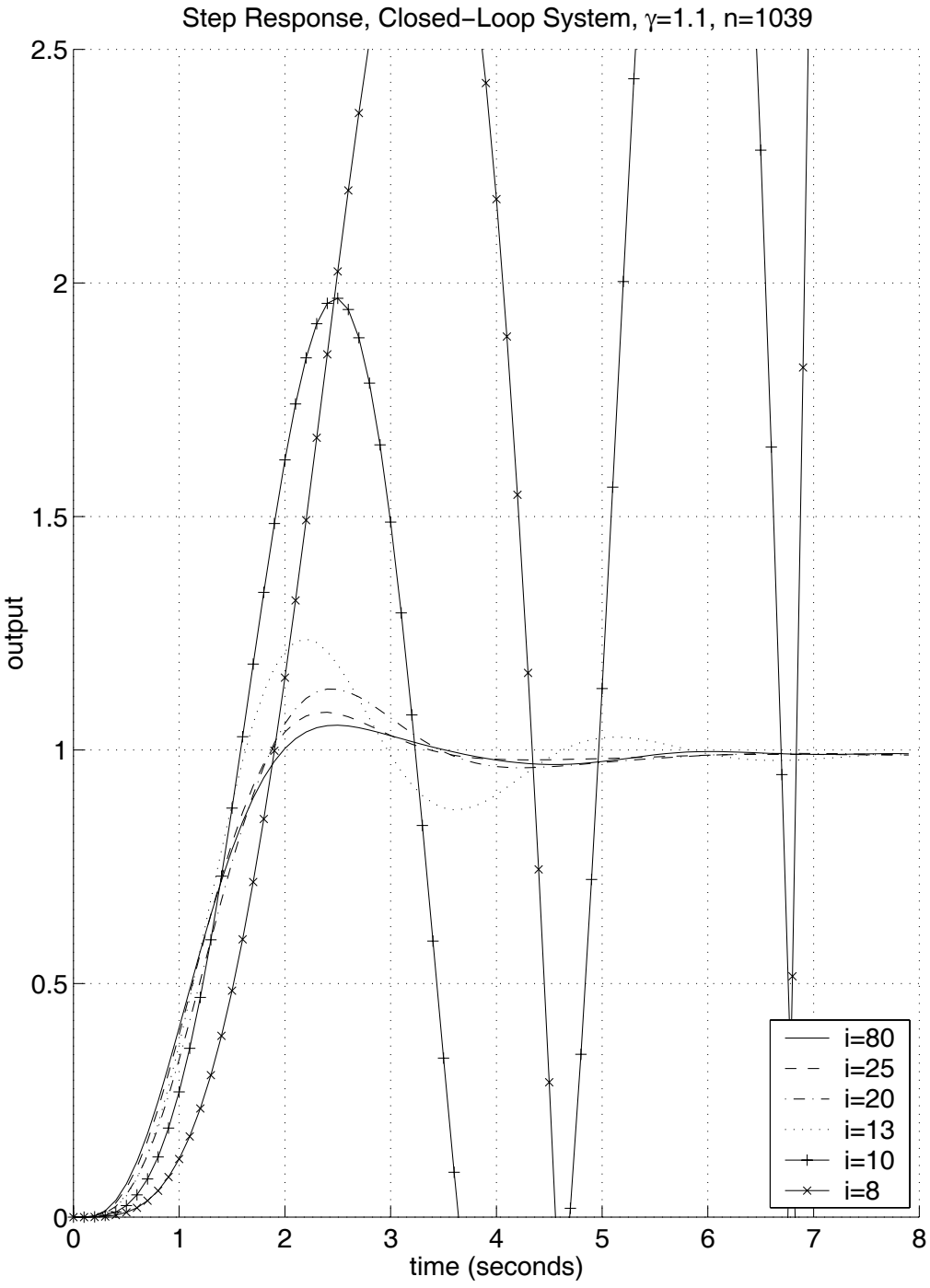

Figure 5.8: Step responses for various choices of i, constant γ, constant n

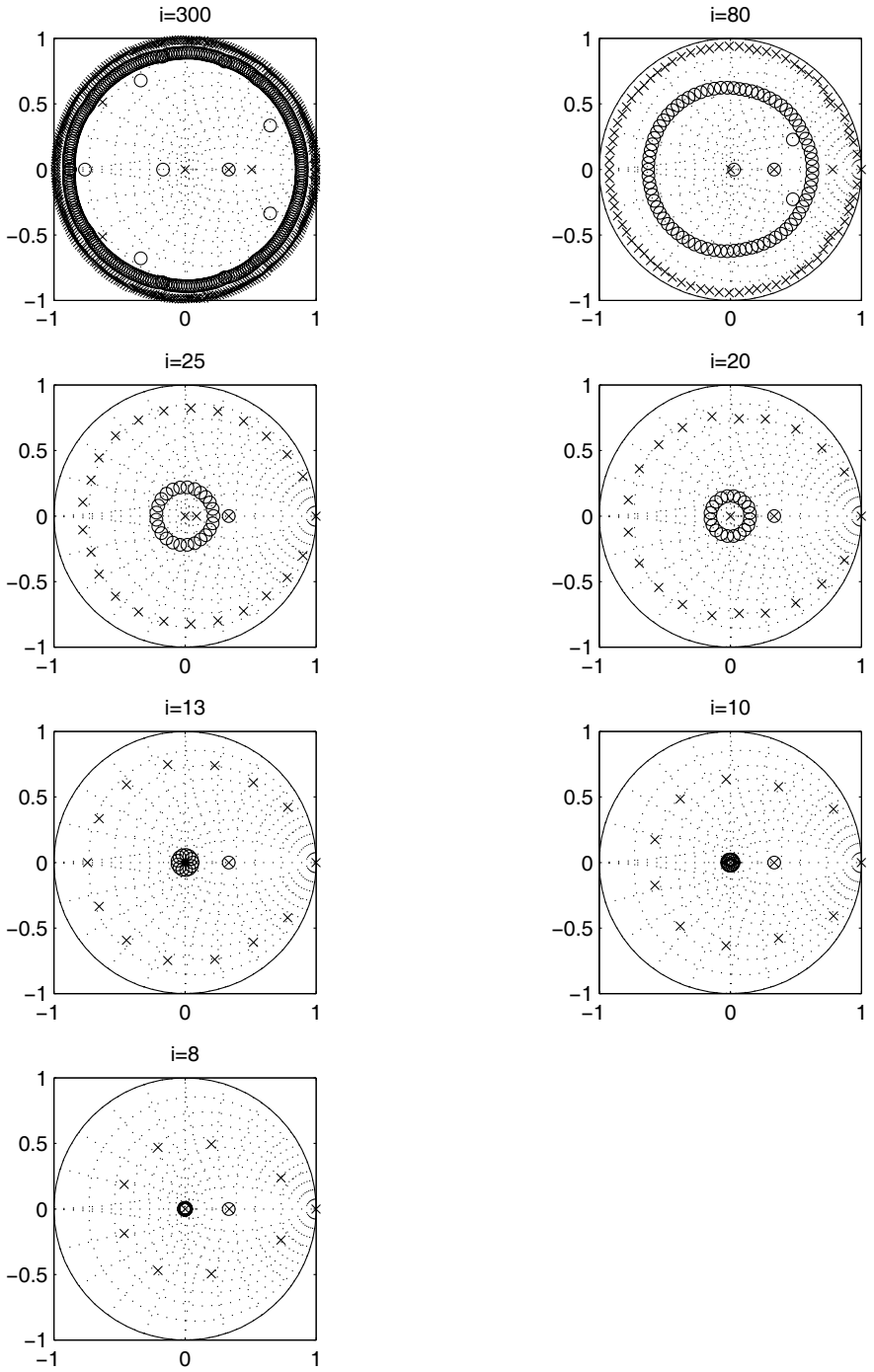

Figure 5.9: Pole and zero locations for various choices of i

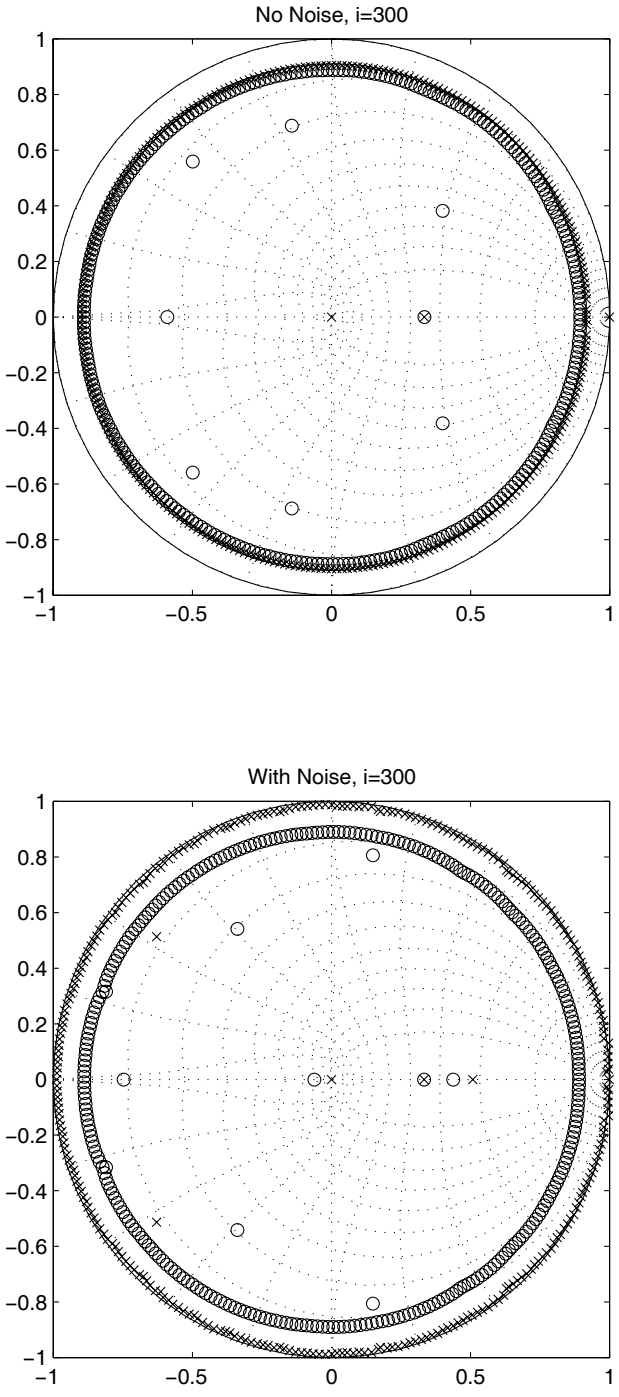

Figure 5.10: Effect of noise on the convergence of pole and zero location $i = 300$

	$\gamma_{\min}$	$\gamma = (1.1)\gamma_{\min}$
$j = 8000$	0.869	0.956
$j = 1000$	0.880	0.968
$j = 100$	0.905	0.995
$j = 75$	0.882	0.970
$j = 60$	1.026	1.129

Table 5.4: $\gamma_{\min}$, γ for various choices of j

5.4.3 Effect Of j

The changes in the controller due to changes in the design parameter j are straightforward to quantify. Since $n = 2i + j - 1$, changes to the amount of experimental data are directly proportional to changes in j. As shown in Table 5.4, $\gamma_{\min}$ does not appreciably change as j changes. The step responses shown in Figure 5.11 show that as j decreases, the closed loop performance degrades, as would be expected with a reduction in the volume of experimental data available for control design. In many practical situations, the engineer will first select or be given experimental data of length n. The selection of i will determine j via the relation $j = n - 2i + 1$. The case of trading i and j with fixed n was studied in Subsection 5.4.2.

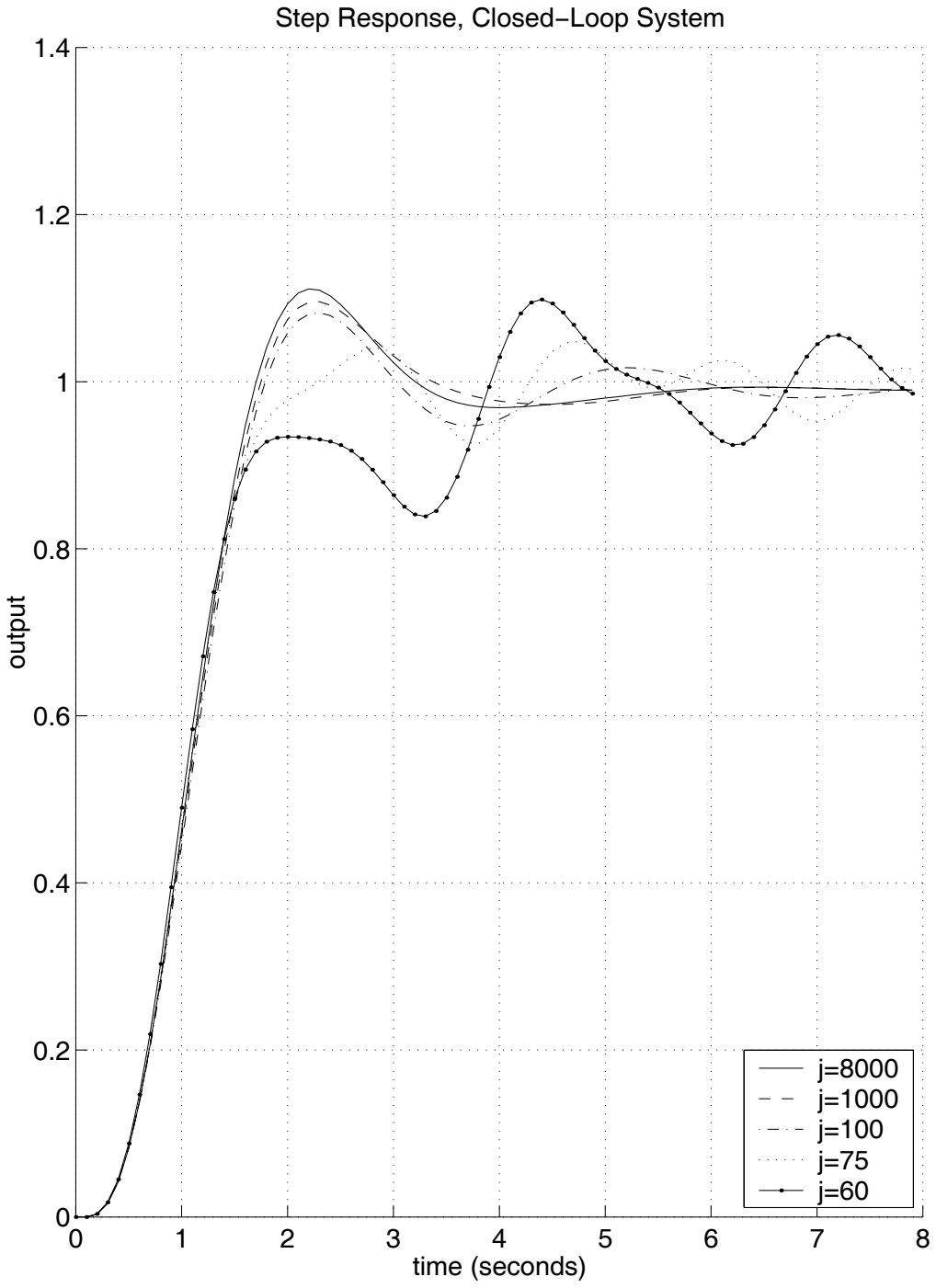

Figure 5.11: Step responses for various choices of j

	γ
$\gamma/\gamma_{\min} = 10^{10}$	8.80×10^9
$\gamma/\gamma_{\min} = 2$	1.760
$\gamma/\gamma_{\min} = 1.5$	1.320
$\gamma/\gamma_{\min} = 1.1$	0.968
$\gamma/\gamma_{\min} = 1.0268$	0.903

Table 5.5: $\gamma/\gamma_{\min}$, γ used to study performance with respect to choice of γ

5.4.4 Effect Of γ

Table 5.5 shows various values of $\gamma/\gamma_{\min}$ used to study the performance variations with respect to choice of γ. Figure 5.12 shows the closed loop response associated with each value of $\gamma/\gamma_{\min}$. As γ is decreased, the control becomes progressively more aggressive, until about 3% above $\gamma_{\min}$, at which point the closed loop response goes unstable. Smaller values of γ result in extremely violent instability. A general rule of thumb is to choose $\gamma = (1.1)\gamma_{\min}$ in order to preserve some margin in the control design. One may choose $\gamma > (1.1)\gamma_{\min}$ if an additional design margins are desired.

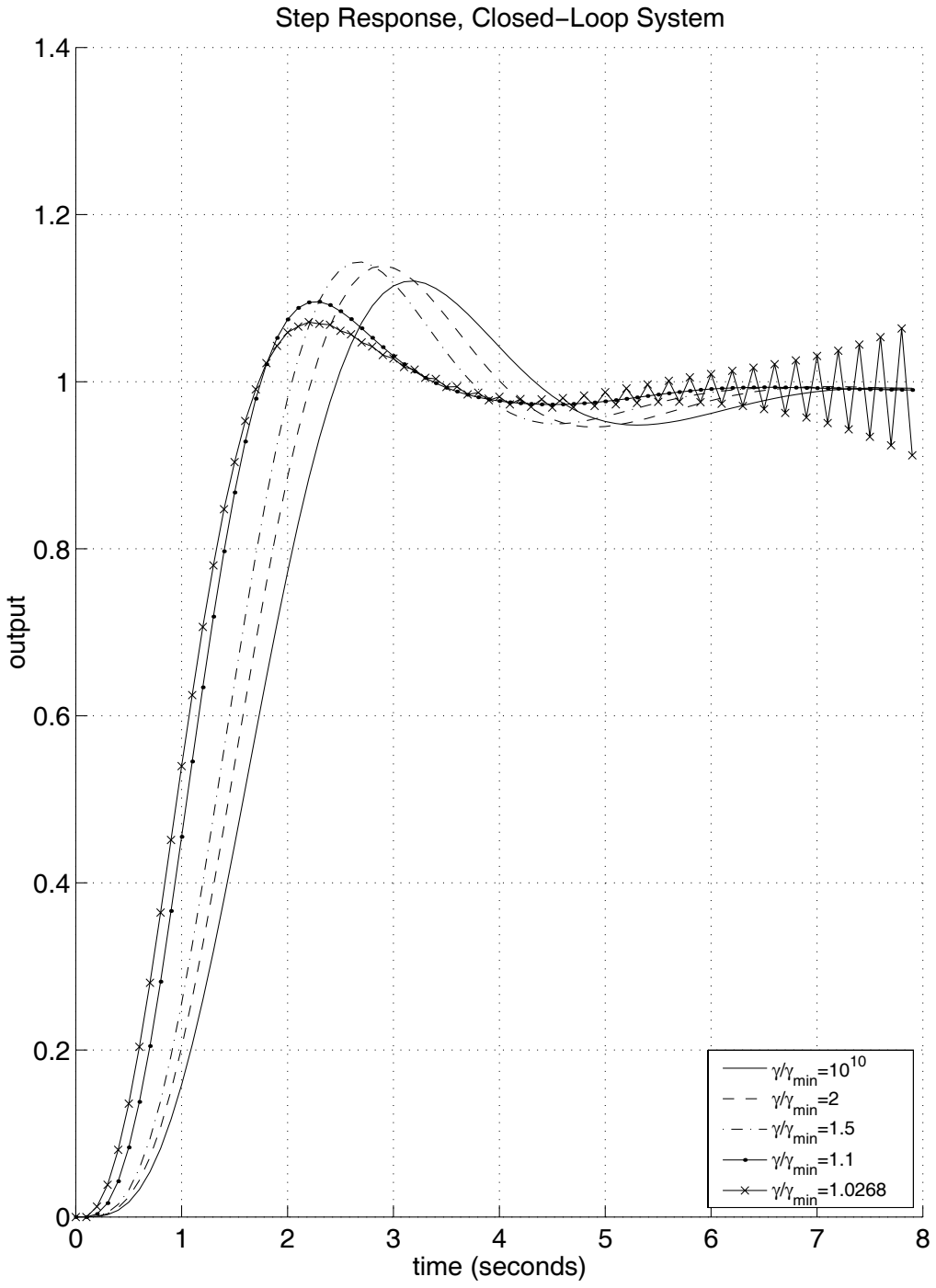

Figure 5.12: Step responses for various choices of γ

5.5 Summary

The important results of this chapter are:

- The simplified receding horizon implementation (5.3), (5.4);
- The algorithm for batch implementation of the model free subspace based $\mathcal{H}_\infty$ control law as summarized in Section 5.2.1;
- The algorithm for adaptive implementation of the model free subspace based $\mathcal{H}_\infty$ control law as summarized in Section 5.3.1; and
- The choices of the design parameters W_1, W_2, i, j, and γ have the following effects:
  - The weighting functions W_1, W_2 control the closed loop behavior in a fashion common to model based loop shaping functions;
  - The prediction horizon i is critical in the trade between controller order and performance. Larger i generally produces better performance;
  - The parameter j is set by i and the amount of experimental data that are available.
  - The parameter γ selects, to some extent, the aggressiveness of the control law. Larger γ generally produces more conservative control laws.

Chapter 6

Experiments

In this chapter, the model free subspace based $\mathcal{H}_\infty$ control technique is applied to solve a number of control problems. Several laboratory experiments have been conducted in order to demonstrate the off-line technique on a real physical system. Due to computational considerations, the on-line technique is demonstrated via simulation, using a plant model that resembles the experimental system used for the off-line demonstrations.

6.1 Experimental Hardware

Figure 6.1 shows a schematic and photo of the experimental hardware, manufactured by Educational Control Products [29]. It consists of three rotational inertias, linked by two torsional springs. The position of each inertia is sensed with a shaft encoder (16000 divisions per revolution), producing angles y_1, y_2, y_3. A single actuator provides a torque u at inertia y_1. The torque command u (-10 volts to $+10$ volts) is converted to a motor drive current by a "black box" controller. The commutation for the DC brushless motor is rather poorly designed, thus several time-varying nonlinearities (deadband, stiction, backlash) are present.

Assuming that the input u is in fact a commanded torque, it is possible to derive transfer functions from the input u to each of the three outputs [29]. Control of the structure using only y_1 is referred to as the single input single output (SISO) collocated control problem, because the sensing and actuation take place at the same mass element. Similarly, controlling the structure using only a combination of y_2 and y_3 is referred to as the non-collocated control problem. Clearly, there are seven unique combinations of sensors that can be used to control the structure.

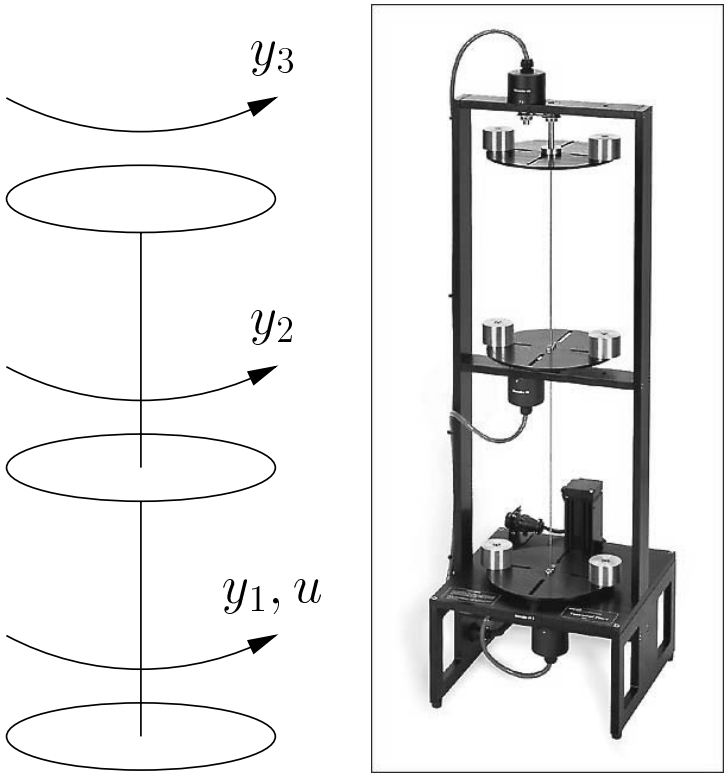

Figure 6.1: Schematic and photo of the three disk system

Models for each of the three SISO transfer functions are tabulated below. The numerical values for the parameters were derived by many students who have studied this experimental system as part of their curriculum at Stanford University. The sensor units are radians, and the input units are volts. The transfer functions are

$$
\begin{aligned}
P_{y_1,u} &= \frac{k(s^2+2\zeta_{z_1}\omega_{n_{z_1}}s+\omega_{n_{z_1}}^2)(s^2+2\zeta_{z_2}\omega_{n_{z_2}}s+\omega_{n_{z_2}}^2)}{s(s+\alpha)(s^2+2\zeta_{p_1}\omega_{n_{p_1}}s+\omega_{n_{p_1}}^2)(s^2+2\zeta_{p_2}\omega_{n_{p_2}}s+\omega_{n_{p_2}}^2)} \\
P_{y_2,u} &= \frac{k(\omega_{n_{z_2}}^2)(s^2+2\zeta_{z_1}\omega_{n_{z_1}}s+\omega_{n_{z_1}}^2)}{s(s+\alpha)(s^2+2\zeta_{p_1}\omega_{n_{p_1}}s+\omega_{n_{p_1}}^2)(s^2+2\zeta_{p_2}\omega_{n_{p_2}}s+\omega_{n_{p_2}}^2)} \\
P_{y_3,u} &= \frac{k(\omega_{n_{z_1}}^2)(\omega_{n_{z_2}}^2)}{s(s+\alpha)(s^2+2\zeta_{p_1}\omega_{n_{p_1}}s+\omega_{n_{p_1}}^2)(s^2+2\zeta_{p_2}\omega_{n_{p_2}}s+\omega_{n_{p_2}}^2)}
\end{aligned}
$$

with $k = 24.26$, $\alpha = 0.35$, $\zeta_{z_1} = 0.026$, $\omega_{n_{z_1}} = 10.65$, $\zeta_{z_2} = 0.011$, $\omega_{n_{z_2}} = 42.00$, $\zeta_{p_1} = 0.024$, $\omega_{n_{p_1}} = 15.85$, $\zeta_{p_2} = 0.010$, $\omega_{n_{p_2}} = 43.50$. This flexible structure experiment is well studied [29, 31]. Of the SISO problems, control using only y_3 is a much more difficult problem than

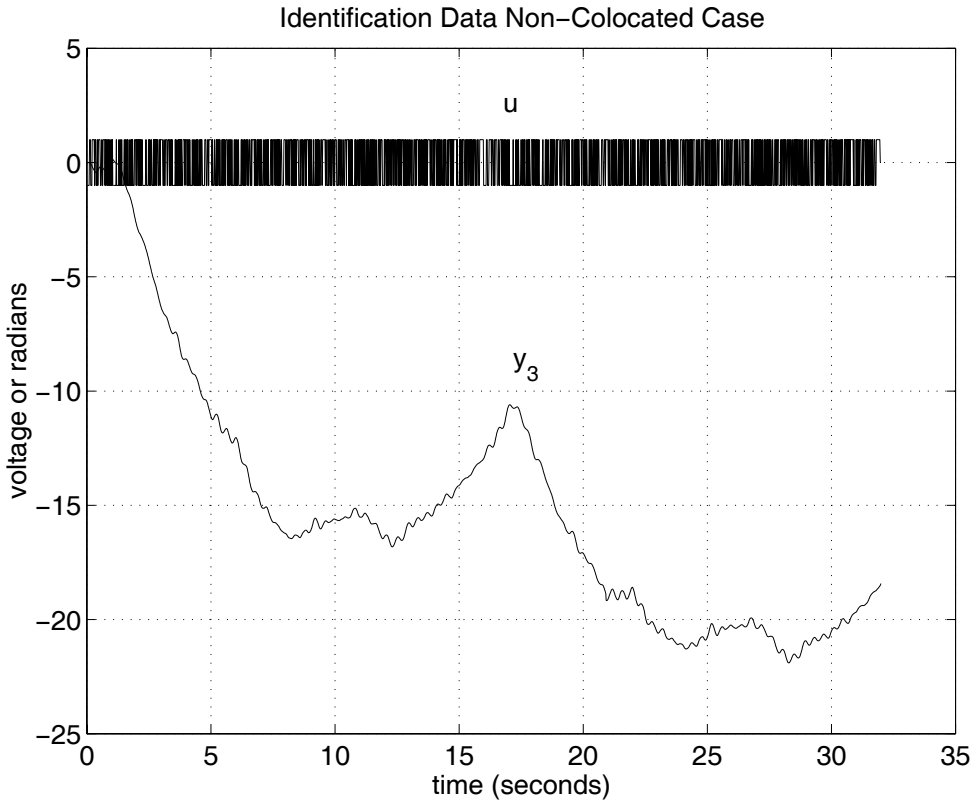

Figure 6.2: Identification data

Two cycles of a 16 *second pseudo random binary sequence drive the plant. The first resonant mode of the plant is visible in the* y_3 *time data.*

control using only y_1 [15].

6.2 Non-Collocated Off-Line Control Design

The following example demonstates the capability of the batch $\mathcal{H}_\infty$ control design technique. Figure 6.2 shows two cycles of a 16 second pseudo-random binary sequence used to excite the plant, and the resulting plant output y_3. Data were collected for 32 seconds at a sampling frequency of 64 Hz.

The weighting functions for the control design were selected in continuous time, and discretized using the Tustin transformation. The discrete time Bode plots of the inverse of the weighting functions are shown in Figure 6.3. The Bode plot of the best model of the

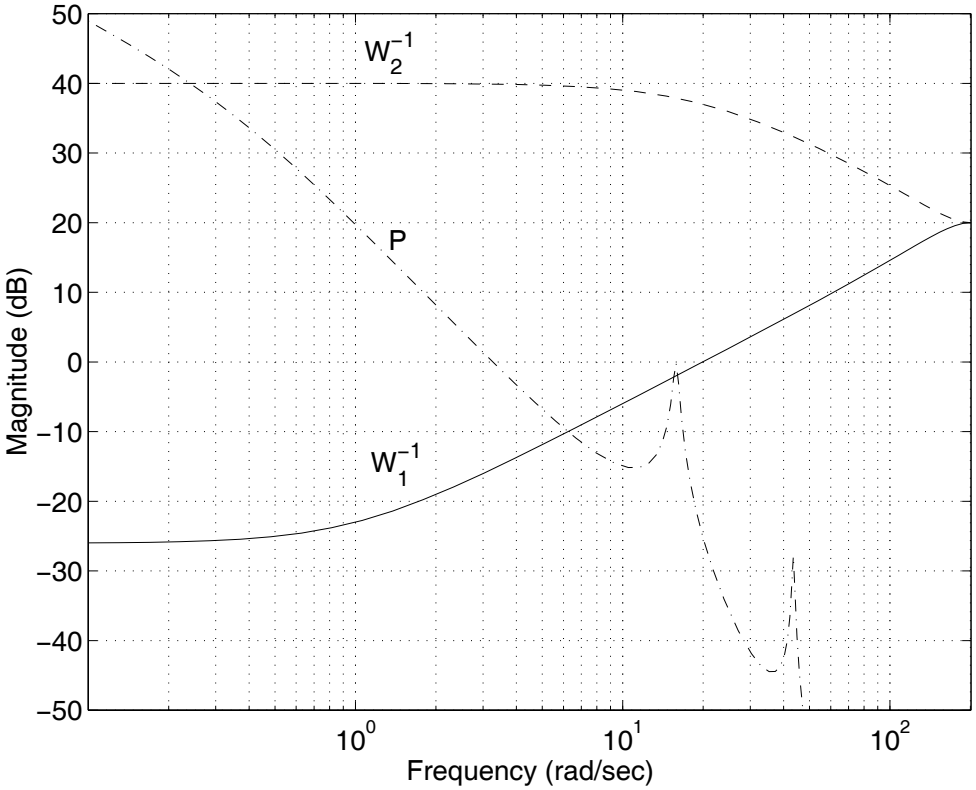

Figure 6.3: Control specification

To satisfy this performance specification, $\|S\|$ must be less than $\|W_1\|^{-1}$ over all frequency.

plant is also displayed in Figure 6.3. The important feature of Figure 6.3 is that W_1 specifies a bandwidth of nearly 20 rad/s, which is past the first resonant mode of the nominal plant (16 rad/s). It is also important to note that although the excitation signal in Figure 6.2 is very rich, the data set is quite short ($n = 2048$).

With the data set of Figure 6.2, $i = 64$, and the control specification of Figure 6.3, $\gamma_{\min} = 1.7$ was computed. Selecting $\gamma = 2.3 = (1.35)\gamma_{\min}$ in order to provide a margin of robustness beyond the usual $\gamma = (1.1)\gamma_{\min}$, the model free subspace based $\mathcal{H}_\infty$ controller was implemented. The experimental step response is shown in Figure 6.4. Note a rise time of 0.11 seconds, ringing at 3.5 Hz (22 rad/s), and a steady state error of 18%.

The observed rise time, ringing and steady state errors compare well to the values expected if the plant is approximated by a second order LTI system [14]. Figure 6.3 indicates

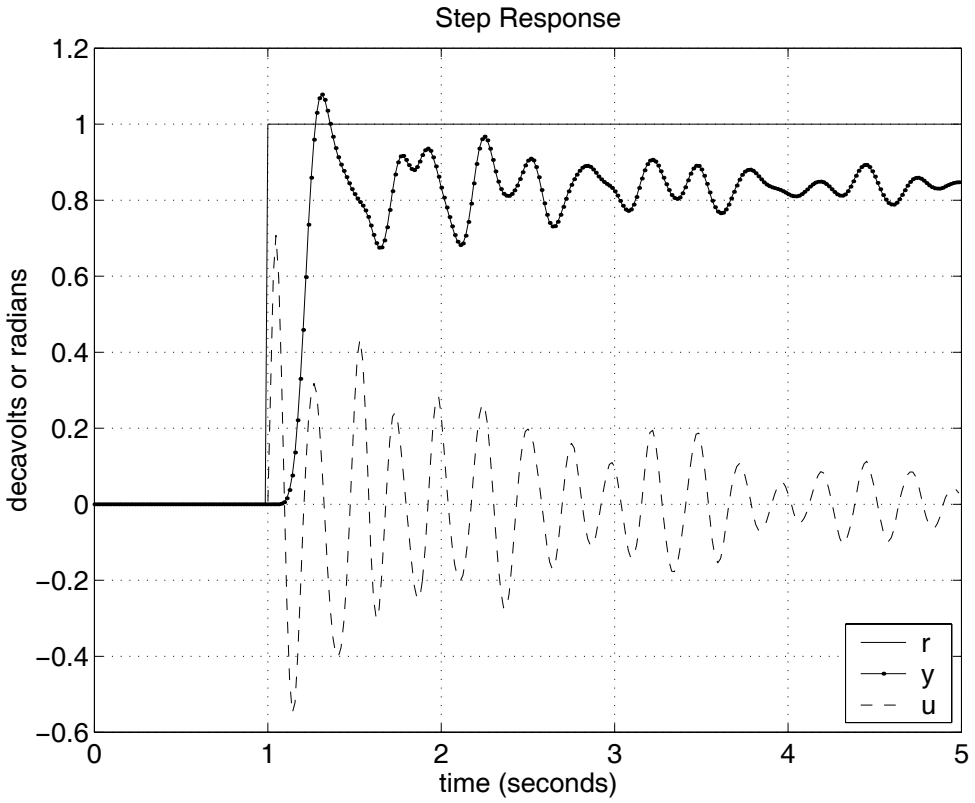

Figure 6.4: Experimental step response

The rise time of this step response is consistent with the design bandwidth. The residual ringing is expected based on the performance specification.

a design cutoff frequency of 3.2 Hz (20 rad/s): this implies $t_{rise} \simeq \frac{1.8}{\omega_n} = \frac{1.8}{20} = 0.09$ seconds, which is close to the observed 0.11 seconds. The ringing at approximately 3.5 Hz (22 rad/s) is also consistent with the design cutoff frequency of 3.2 Hz (20 rad/s). The choice of $\gamma = 2.3$ with a design of 26 dB of rejection at DC implies an open loop DC gain of $\frac{1}{2.3}10^{26/20} = 8.7$, which in turn implies an expected steady state error of $\frac{1}{1+8.7} = 10\%$. The observed steady state error is 18%.

In order to better evaluate the step response of Figure 6.4, a discrete, infinite horizon *model based* $\mathcal{H}_\infty$ controller was designed using the zero order hold discretization of the plant model $P_{y_3,u}$, the weighting filters of Figure 6.3, and $\gamma = (1.35)\gamma_{\min} = (1.35)(2.24) = 3.0$. Figure 6.5 shows the simulated step response (y_{ideal}), generated from a noise free closed

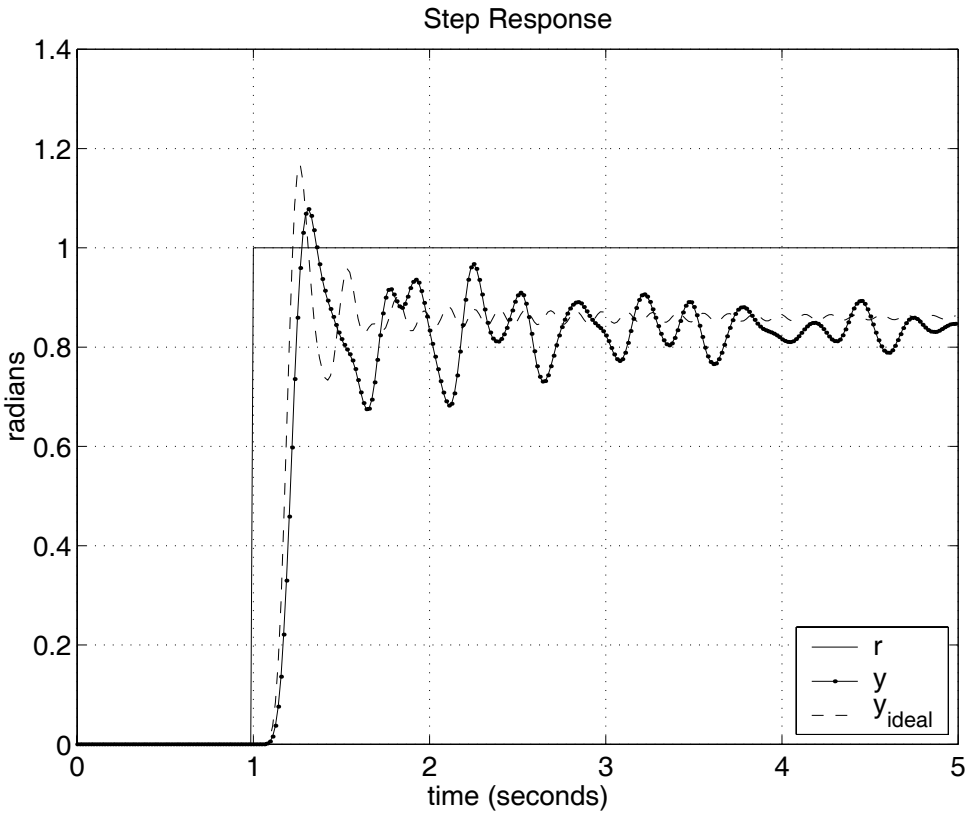

Figure 6.5: Comparison to ideal simulation

The similarity between the ideal step response and the experimental step response indicate that the experiment produces similar response to the best that could be expected with complete plant knowledge.

loop simulation of the plant with the model based $\mathcal{H}_\infty$ control law. The key point is that y_{ideal} is the best response that any controller can be expected to achieve with this performance specification, because the control design and simulation are performed with perfect plant knowledge. Comparing the experimental data y to the ideal response indicates that the actual performance achieved by the finite horizon subspace based controller using experimental data is similar to the best that could be expected.

The "ideal" design of Figure 6.5 suggests one more obvious experiment: how well would the infinite horizon model based $\mathcal{H}_\infty$ control design perform on the real system? The answer is: quite poorly. The unstable step response of this closed loop experiment appears in Figure

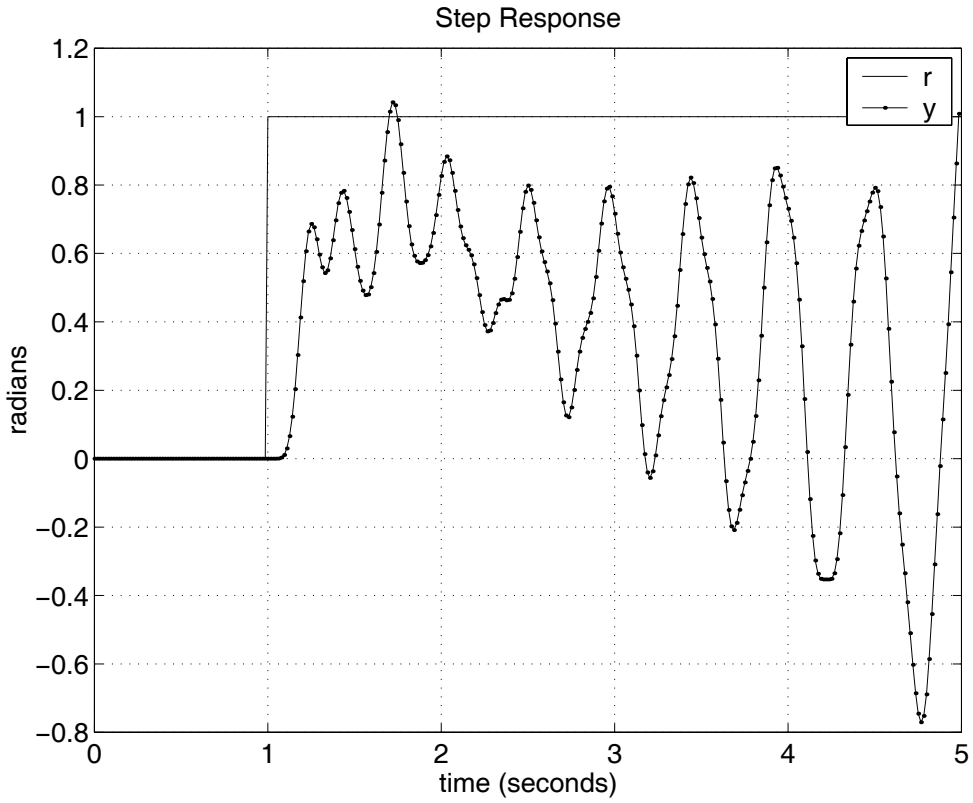

Figure 6.6: Model based control experiment

The model based control design produces an unstable step response when tested on the experimental system.

6.6, showing large oscillations at approximately 13 rad/s. Poor DC response is also evident in the drastic undershoot of the initial step response.

Clearly the plant model used to design the infinite horizon control law is not an accurate model of the real dynamics of the system. Although it is not possible to directly determine what portion of the plant model is at fault, it is possible to directly compare the successful model free control law with the unsuccessful model based control law. Figure 6.7 shows these two control laws. The subspace based controller (K_{sub}) has significantly greater low frequency gain than the model based controller (K_{model}). The notches in the controllers near 13 rad/s are in different locations, and are thus the likely source of the unstable closed loop performance of the model based control law.

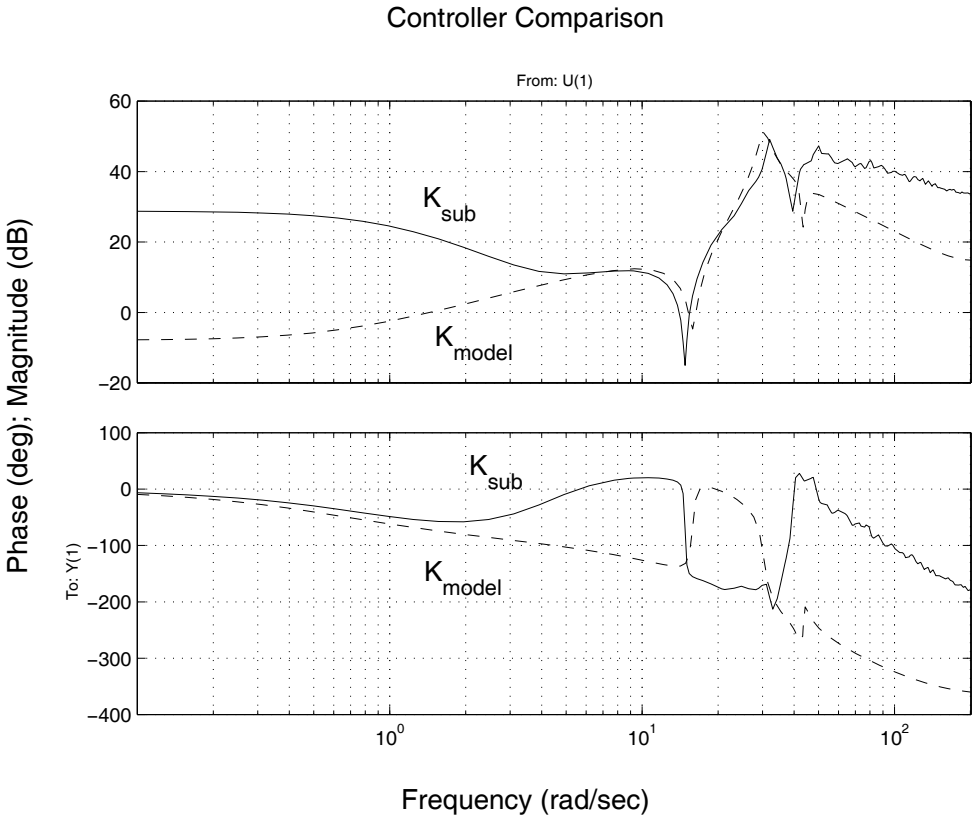

Figure 6.7: Comparison of subspace based and model based controllers

The subspace based controller (K_{sub}) and the model based controller (K_{model}) differ significantly in the location of the notch near 13 *rad/s. This is the likely source of the unstable closed loop performance of the model based control law.*

6.3 Adaptive Simulation

This section demonstates the on-line model free subspace based $\mathcal{H}_\infty$ algorithm, especially its ability to control a dynamic system with very little *a priori* information. Unfortunately, presently available computational power limits the evaluation of the adaptive algorithm to simulated plants. Throughout this section, the collocated SISO plant of Figure 6.1 is simulated with zero-mean white Gaussian sensor noise, $\sigma = 0.01$.

Throughout this simulation, the weighting functions of Figure 6.8 are used. The policy for selecting γ is $\gamma = \gamma_{\min} + 0.6$, which for most values of $\gamma_{\min}$ is somewhat greater than the minimum rule of thumb of $\gamma = (1.1)\gamma_{\min}$. During simulation, the extra margin was found to increase the robustness of the design, thereby causing the simulation to have smaller excursions prior to controller convergence. The sample rate is 64 Hz, and $i = 64$. In order to initialize R (as defined in Section 2.2), the plant is excited with 6 seconds ($n = 384$) of zero-mean white Gaussian noise, $\sigma = 1$. The resulting input-output data are then used to form square U_p (2.1), U_f (2.2), Y_p (2.3), Y_f (2.4) from which the initial R is calculated.

Every 50 time steps, L_w, L_u are updated with 50 new (u, y_1) pairs, and new k_1, k_2, k_3, k_4 are computed and introduced into the controller K. Figure 6.9 shows the adaptive simulation starting at $t = 6$ seconds, *i.e.* immediately after the initialization data. The reference signal is a series of step commands. The adaptive system begins with very poor performance, then quickly improves to achieve good tracking control.

As was done for the off-line design example, the signal y_{ideal} is calculated and plotted for the model based infinite horizon $\mathcal{H}_\infty$ controller utilizing the weighting functions of Figure 6.8. In addition, Figure 6.10 plots the difference between y (the model free adaptive simulation) and y_{ideal}. As the adaptive system converges, its performance closely resembles that achieveable with perfect plant knowledge.

Figure 6.11 shows $\gamma_{\min}$ as a function of time for the simulation of Figure 6.9. As the controller obtains more experimental data, the best attainable performance changes: note that there is no reason to expect that it should vary monotonically. After $t = 15$ seconds, $\gamma_{\min}$ settles towards a steady state value, close to the $\gamma_{\min} = 2.25$ computed by the *infinite horizon* model based design. Thus the $\gamma_{\min}$ determined by the adaptive algorithm is a reasonable estimate of the true achievable closed loop performance.

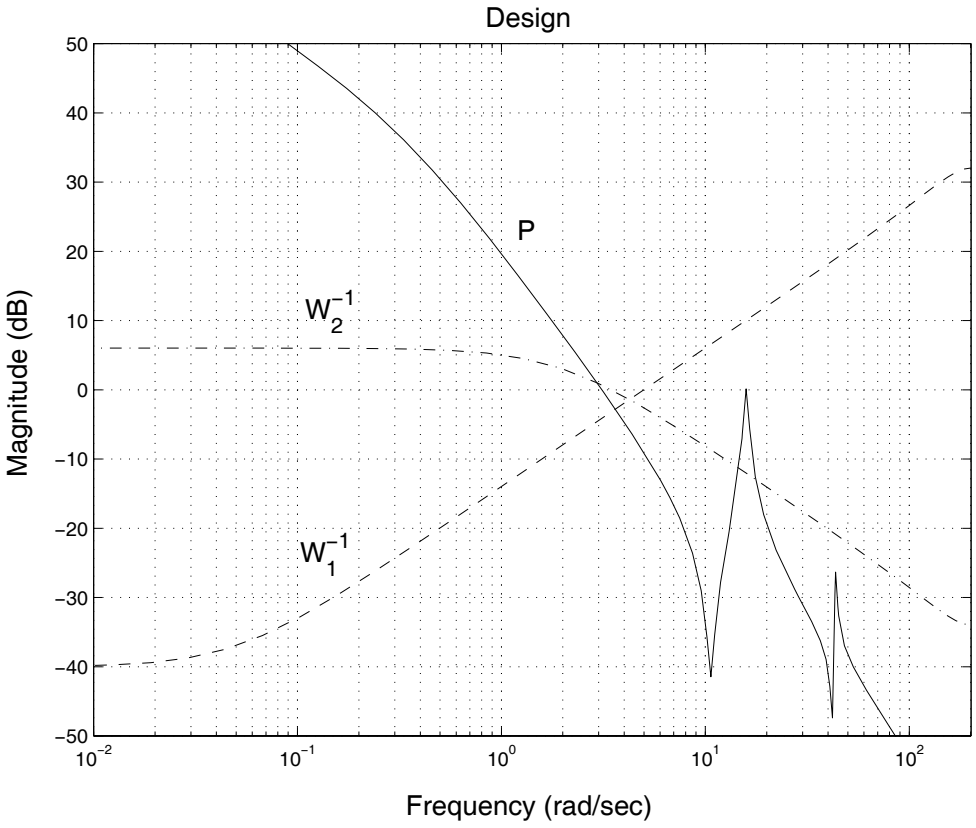

Figure 6.8: Design for adaptive simulation

To satisfy this performance specification, $\|S\|$ must be less than $\|W_1\|^{-1}$ over all frequency.

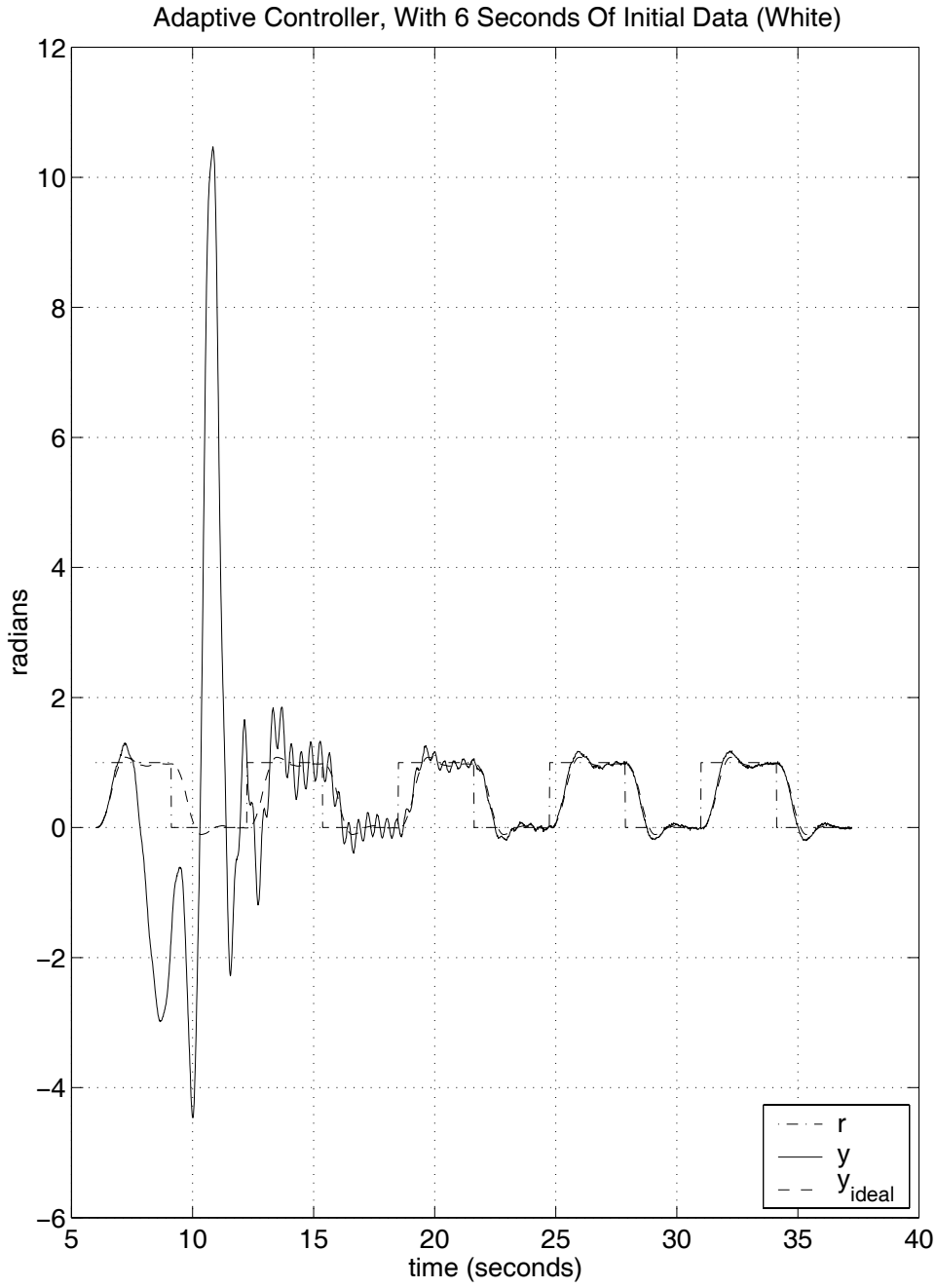

Figure 6.9: Adaptive simulation

The adaptive controller converges quickly, producing performance that closely resembles that achieveable with perfect plant knowledge.

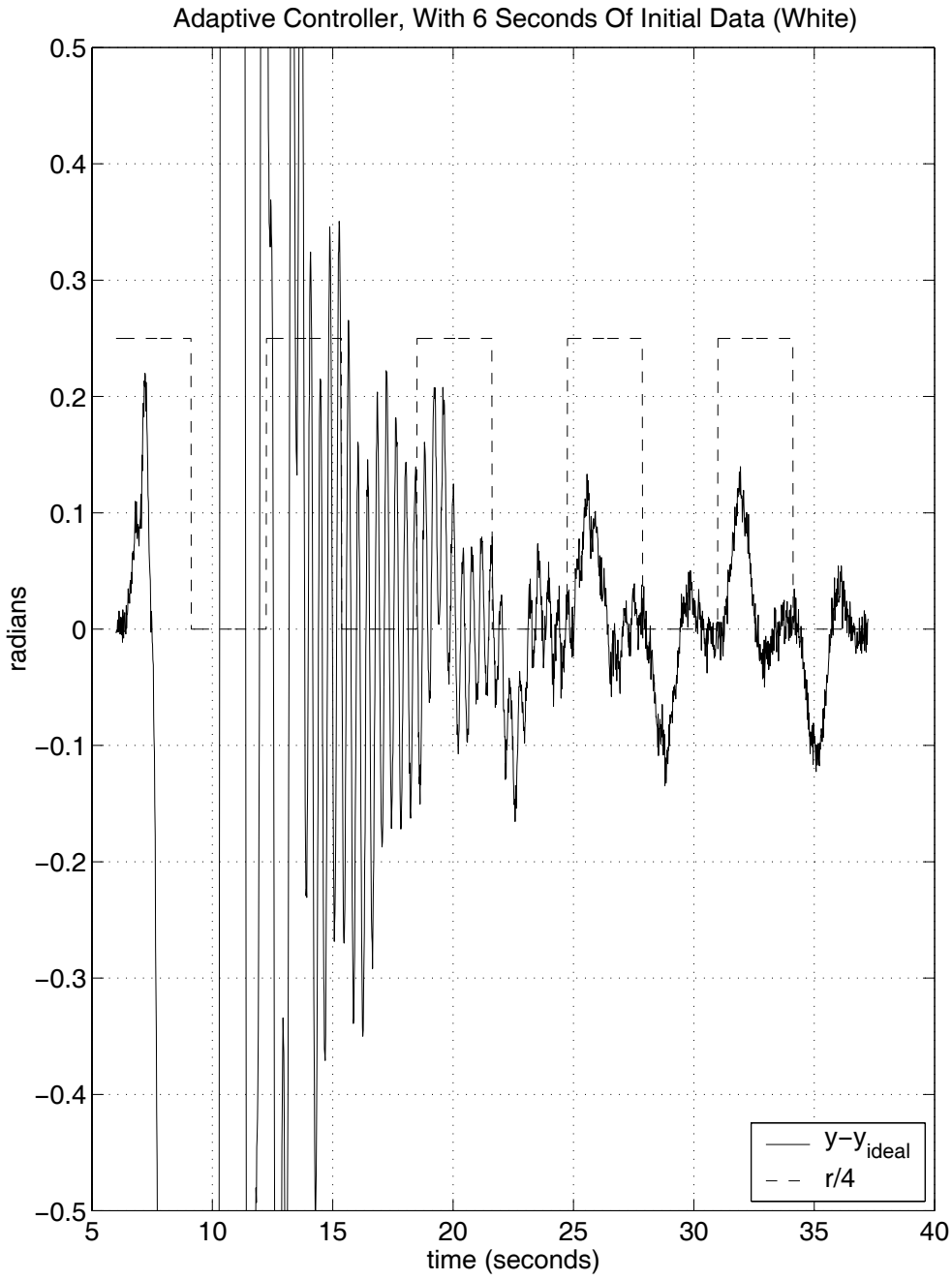

Figure 6.10: Adaptive simulation error

The difference between the adaptive system performance and the ideal system performance quickly becomes small as the adaptive system converges. The reference signal has been scaled by 1/4.

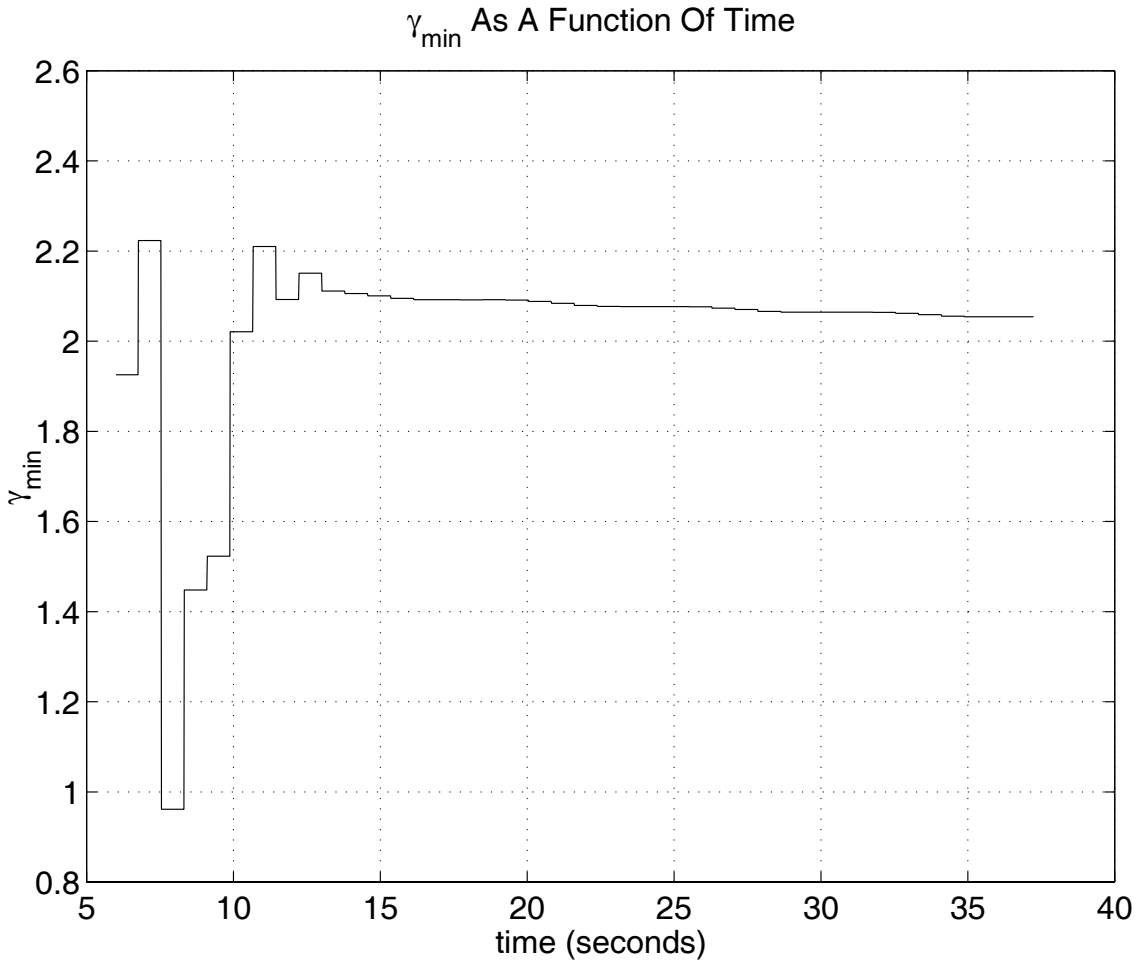

Figure 6.11: $\gamma_{\min}$ during adaptive simulation

During adaptation, $\gamma_{\min}$ fluctuates in a non-monotonic manner, eventually converging to a steady state value close to that predicted by the infinite horizon model based control design.

6.4 Summary

The high performance non-collocated control of a lightly damped flexible structure is demonstrated using the batch model free subspace based $\mathcal{H}_\infty$ control technique. The observed experimental performance is close to the best performance that can be obtained in simulation, demonstrating the capabilities of the new control design technique. A simulation in which a lightly damped collocated flexible structure is controlled using the adaptive model free subspace based $\mathcal{H}_\infty$ technique demonstrates the adaptive technique's capability to rapidly converge to an appropriate control law. Again, performance is close to the best performance that can be obtained in simulation.

Chapter 7

Model Unfalsification

In order to use one of the model free subspace based *robust* control laws of Chapter 3, the engineer must select an appropriate uncertainty model based on intuition, plant analysis, or other such technique [18, 44]. If it were possible to efficiently obtain an appropriate uncertainty model during the process of forming the subspace predictor, a more automated robust design technique would be enabled.

In this chapter, the theory of model unfalsification [32] is applied to the problem of deriving an uncertainty model for the model free subspace based robust control technique employing an additive uncertainty. Unfortunately, this method of automatically obtaining an uncertainty model is too computationally expensive to demonstrate in simulation. The subspace uncertainty model unfalsification technique developed in this chapter is similar to ARX uncertainty model unfalsification [21], which also suffers from computational complexity issues. If the computation required to solve the ARX uncertainty model unfalsification problem were greatly reduced, then the connection between ARX uncertainty model unfalsification and subspace uncertainty model unfalsification demonstrated in this chapter will likely enable automated model free subspace based robust control.

Section 7.1 reviews the theory of unfalsification of LTI models. Section 7.2 reviews the application of these ideas to the ARX problem. Section 7.3 applies the uncertainty model unfalsification idea to the subspace prediction problem.

7.1 LTI Uncertainty Model Unfalsification

Science operates on the principle of forming many hypotheses about the way the world works, and then performing many experiments to test each hypothesis. If a single experiment is performed to test a single hypothesis, and the experiment supports the hypothesis, then all that can be said is that the experiment *does not falsify* the hypothesis. Humanity never has conclusive proof that the hypothesis is true: we can only build up vast amounts of evidence (through experiment) that the hypothesis has not been proven false [13]. Experiments falsify or *unfalsify* scientific hypotheses. This idea can be summarized by the following definition.

Definition [32]
A physical model is said to be unfalsified by an experiment if the model could have produced the experimental data. □

The notion of unfalsification can be applied to the observation of plant input-output data [30]. In this chapter, the hypothesis that is to be tested is "could the input-output data have been produced by an LTI transfer function with $\mathcal{H}_\infty$ induced gain less than γ?". Theorem 7.1 provides the exact condition for the existence of input-output data that could have been produced by a level-γ $\mathcal{H}_\infty$ norm bounded LTI system [30].

Theorem 7.1
Consider SISO input-output data of length n

$$\left(\begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{n-1} \end{bmatrix}, \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{n-1} \end{bmatrix} \right)$$

Let $\star$ represent unknown future data. Then there exists an LTI operator $\Delta(q)$ such that

$\|\Delta\|_\infty \leq \gamma$ *and*

$$\begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{n-1} \\ \star \\ \vdots \end{bmatrix} = \Delta(q) \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{n-1} \\ \star \\ \vdots \end{bmatrix}$$

if and only if

$$\mathcal{T}(y)^T \mathcal{T}(y) \leq \gamma^2 \mathcal{T}(u)^T \mathcal{T}(u) \tag{7.1}$$

where u, y, $\mathcal{T}(u)$ *and* $\mathcal{T}(y)$ *are defined by*

$$u \stackrel{\triangle}{=} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{n-1} \end{bmatrix} \qquad y \stackrel{\triangle}{=} \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{n-1} \end{bmatrix}$$

$$\mathcal{T}(u) \stackrel{\triangle}{=} \begin{bmatrix} u_0 & 0 & \cdots & 0 \\ u_1 & u_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ u_{n-1} & u_{n-2} & \cdots & u_0 \end{bmatrix} \qquad \mathcal{T}(y) \stackrel{\triangle}{=} \begin{bmatrix} y_0 & 0 & \cdots & 0 \\ y_1 & y_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ y_{n-1} & y_{n-2} & \cdots & y_0 \end{bmatrix}$$

and q *is the forward shift operator.* □

7.2 ARX Uncertainty Model

The application of Theorem 7.1 to the derivation of an uncertainty/noise model from experimental input-output data was first developed in [21, 22]. In this section, Theorem 7.1 is applied to the ARX plant identification problem: the resulting formulation simultaneously recovers a SISO LTI plant model and an LTI uncertainty model by performing a linear matrix inequality (LMI) constrained optimization [21]. First, the classic ARX formulation will be reviewed [25].

7.2.1 Classical ARX

Consider the ARX plant structure

$$\begin{aligned} A(q)y &= B(q)u + e \\ A(q) &= 1 + a_1 q^{-1} + \cdots + a_{n_a} q^{-n_a} \\ B(q) &= b_1 q^{-1} + \cdots + b_{n_b} q^{-n_b} \end{aligned}$$

where $[u, y]$ are the input-output experimental data, q is the forward shift operator, and the error signal e represents all the mismatch between the model and the actual experimental data. In the ARX problem, it is assumed that the error is due to "noise". In order to best fit the polynomials $A(q)$ and $B(q)$ to the data, θ is selected to minimize the noise energy, that is

$$\theta = \arg\min_{\theta} \sum_{k=0}^{n-1} \|e_k(\theta)\|_2^2 = \arg\min_{\theta} e(\theta)^T e(\theta) \tag{7.2}$$

where

$$\begin{aligned} e(\theta) &\triangleq \Phi\theta + Y \\ \Phi &\triangleq \begin{bmatrix} y_{-1} & \cdots & y_{-n_a} & -u_{-1} & \cdots & -u_{-n_b} \\ y_0 & \cdots & y_{-n_a+1} & -u_0 & \cdots & -u_{-n_b+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ y_{n-2} & \cdots & y_{n-n_a-1} & -u_{n-2} & \cdots & -u_{n-n_b-1} \end{bmatrix} \\ \theta &\triangleq \begin{bmatrix} a_1 \\ \vdots \\ a_{n_a} \\ b_1 \\ \vdots \\ b_{n_b} \end{bmatrix} \qquad Y \triangleq \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{n-1} \end{bmatrix} \end{aligned}$$

Note that (7.2) is a least squares problem. Either the system is assumed to be at rest, *i.e.* $y_k = 0 \; \forall \; k \in \{-1, \cdots - n_a\}$ and $u_k = 0 \; \forall \; k \in \{-1, \cdots, -n_b\}$, or it is assumed that the data prior to time 0 is known *a priori*. The solution to the ARX problem (7.2) is given by

$$\theta = -(\Phi^T \Phi)^{-1} \Phi^T Y$$

7.2.2 ARX Unfalsified Uncertainty Model

Within the ARX framework, it is possible to assume that the error (e) is not only due to noise ξ, but also due to "plant uncertainty" [21] *i.e.*

$$A(q)y - B(q)u = e \stackrel{\triangle}{=} \xi + \Delta(q)u \tag{7.3}$$

This formulation leaves the engineer with a trade: how much of the mismatch (e) should be attributed to noise, and how much should be attributed to plant uncertainty? One such method for making this trade is to choose a maximum permissable noise power

$$\|\xi\|_{rms} \leq \sigma$$

and then use the experimental data to compute the minimum sized uncertainty that is necessary to unfalsify the error model (7.3) with respect to the experimental data. By invoking Theorem 7.1, the minimum δ_{unf} that satisfies

$$\|\Delta\|_\infty \leq \delta_{unf}$$

and unfalsifies the experimental data is given by

$$\delta_{unf} = \min_{\delta,\theta,\xi} \delta \tag{7.4}$$

subject to

$$\mathcal{T}(e(\theta) - \xi)^T \mathcal{T}(e(\theta) - \xi) \leq \delta^2 \mathcal{T}(u)^T \mathcal{T}(u)$$

Problem (7.4) can be written as a linear matrix inequality (LMI) optimization [4]. The program is

$$\delta_{unf} = \min_{\delta,\theta,\xi} \delta$$

subject to

$$0 \leq \begin{bmatrix} \sigma & \frac{1}{\sqrt{n}}\xi^T \\ \frac{1}{\sqrt{n}}\xi & \sigma I \end{bmatrix} \qquad 0 \leq \begin{bmatrix} \delta\mathcal{T}(u)^T\mathcal{T}(u) & \mathcal{T}(e(\theta) - \xi)^T \\ \mathcal{T}(e(\theta) - \xi) & \delta I \end{bmatrix}$$

where $\xi \in \mathbb{R}^n$. For even moderate data sets, this is a large problem, as there are $n+n_a+n_b+1$ variables, and the constraints are size $(n+1) \times (n+1)$ and size $(2n) \times (2n)$.

7.3 Subspace Uncertainty Model Unfalsification

A similar LMI optimization can be formulated for the subspace prediction problem of Section 2.1. The error model for the subspace problem (2.5) is

$$Y_f = \begin{bmatrix} L_w & L_u \end{bmatrix} \begin{bmatrix} W_p \\ U_f \end{bmatrix} + E$$

where Y_f, W_p, and U_f are as defined in Section 2.1, and E is a matrix of noises. If the error model is modified to include plant uncertainty, then

$$Y_f - \begin{bmatrix} L_w & L_u \end{bmatrix} \begin{bmatrix} W_p \\ U_f \end{bmatrix} = E = \Xi + \overline{\Delta}(q)U_f$$

where Ξ is a matrix of noises, and $\overline{\Delta}$ represents a set of j LTI operators that each operate on one column of the matrix U_f.

If the engineer specifies σ, the RMS noise power in Ξ, that is

$$\frac{1}{ilj} \|\Xi\|_F^2 \leq \sigma^2$$

then the smallest plant uncertainty that can unfalsify the experimental data is given by

$$\delta_{unf} = \min_{\delta, L_w, L_u, \Xi} \delta \tag{7.5}$$

subject to

$$\begin{aligned} 0 &\leq \begin{bmatrix} \sigma & \frac{1}{\sqrt{ilj}} vec(\Xi)^T \\ \frac{1}{\sqrt{ilj}} vec(\Xi) & \sigma I \end{bmatrix} \\ 0 &\leq \begin{bmatrix} \delta \mathcal{T}((U_f)_{:,k})^T \mathcal{T}((U_f)_{:,k}) & \mathcal{T}((E(L_w, L_u) - \Xi)_{:,k})^T \\ \mathcal{T}((E(L_w, L_u) - \Xi)_{:,k}) & \delta I \end{bmatrix} \qquad \forall\, k \in \{1, \cdots, j\} \end{aligned}$$

where

$$vec(\Xi) = \begin{bmatrix} \Xi_{:,1} \\ \Xi_{:,2} \\ \vdots \\ \Xi_{:,j} \end{bmatrix}$$

Note that this problem is extremely large, with $(il)(il+2im+j)+1$ variables, one constraint size $(ilj+1) \times (ilj+1)$, and j constraints size $(i+il) \times (i+il)$.

7.4 Discussion

The method of developing an unfalsified model proposed in Section 7.3 is computationally expensive. Consider the simple example of a SISO plant with $i = 100$, $j = 1000$: the problem has 130001 variables, and it would take 20.04×10^9 real numbers to write out the constraints. This method of obtaining uncertainty models directly from data will not be computationally realistic in the forseeable future.

However, if it were possible to solve (7.5) for moderately large problems, the following steps might be used as the basis of a robust adaptive system.

1. Collect experimental data
2. Select W_1, W_2, σ
3. Compute δ by solving (7.5); compute L_w, L_u, $\gamma_{\min_a}$ using Theorem 3.7 with $W_3 = \frac{1}{\delta} I$, $W_4 = I$
4. Select $\gamma > \gamma_{\min_a}$
5. Implement the robust controller using Theorem 3.7

It is hoped that future researchers may be able to more efficiently solve (7.5) or a similar problem, so that appropriate uncertainty models may be derived directly from experimental data.

Chapter 8

Conclusions

This chapter summarizes the work in this thesis, and provides suggestions for future researchers.

8.1 Summary

This thesis presents a new control design technique that enables the synthesis of central γ-optimal $\mathcal{H}_\infty$ controllers directly from experimental data. The process uses a subspace predictor in order to extrapolate future plant outputs from past experimental data and future plant inputs.

The new model free subspace based control design technique is able to solve many MIMO $\mathcal{H}_\infty$ problems, including mixed sensitivity cost functions, mixed sensitivity cost functions with multiplicative uncertainty, and mixed sensitivity cost functions with additive uncertainty. In the limit as $\gamma \to \infty$, the design technique recovers the model free subspace based LQG control design technique. If the experimental data are collected from a truly linear time-invariant (LTI) plant, and if the amount of experimental data becomes very large, it is possible to show that the model free subspace based $\mathcal{H}_\infty$ controller is equivalent to a Kalman filter estimating the plant state, with a full information finite horizon $\mathcal{H}_\infty$ controller operating on the estimated plant state.

A simplified method of implementing the receding horizon versions of model free subspace based controllers is developed. A fast method of updating the controllers upon the collection of new experimental data is also developed, enabling adaptive model free subspace based control. It is possible to closely control the "windowing" of the experimental

data, so that the controller may not only "learn" from new experimental data, but it may also "forget" old experimental data. A trade study is performed on the design technique using a simple control problem.

A non-collocated flexible structure laboratory experiment is used to demonstrate the off-line $\mathcal{H}_\infty$ control design technique. In the example high performance control design, the model free subspace based technique out performs the model based control design technique. A simulated collocated flexible structure is used to demonstrate the on-line adaptive technique. It is shown that the adaptive technique quickly converges to an approximation of the ideal performance that could be expected if perfect plant knowledge were available.

Finally, using plant unfalsification techniques, a method of determining appropriate additive uncertainty models for robust model free subspace based $\mathcal{H}_\infty$ controllers is proposed. It is computationally expensive and is thus unlikely to be useful in the near future.

8.2 Future Work

This research provides several opportunities for possible future research and development. The method of calculating the subspace predictors might be made more efficient by better exploitation of the Hankel structure of the problem. It might also be possible to extend the technique to other mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control problems. As is the case with model based robust $\mathcal{H}_\infty$ design, the model free robust $\mathcal{H}_\infty$ solution is often conservative due to the off diagonal terms in the unstructured uncertainty block Δ. Extending the formulation to a structured uncertainty, enabling μ-synthesis for model free subspace based control may help alleviate this conservativeness. Developing a computationally feasible method of calculating the uncertainty Δ from experimental data may greatly improve the ability to automate on-line model free robust control design.

Bibliography

[1] Pascale Bendotti, Benoit Codrons, Clement-Marc Falinower, and Michel Gevers. Control oriented low order modelling of a complex pressurized water reactor plant: A comparison between open loop and closed loop methods. *Proceedings of the IEEE Conference on Decision and Control*, 3:3390–3395, 1998.

[2] Dennis S. Bernstein and David C. Hyland. Optimal projection equations for finite-dimensional fixed-order dynamic compensation of infinite-dimensional systems. *SIAM Journal on Control and Optimization*, 24(1):122–151, January 1986.

[3] R.B. Bitmead, M. Gevers, and V. Wertz. *Adaptive Optimal Control - The Thinking Man's GPC*. Prentice Hall International, 1990.

[4] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. SIAM Studies in Applied Mathematics, 1994.

[5] Stephen P. Boyd and Craig H. Barratt. *Linear Controller Design, Limits of Performance*. Prentice Hall, 1991.

[6] Arthur E. Bryson and Yu-Chi Ho. *Applied Optimal Control*. Hemisphere Publishing Corporation, 1975.

[7] D.W. Clarke, C. Mohtadi, and P.S. Tuffs. Generalized predictive control - part i. the basic algorithm. *Automatica*, 23(2):137–148, March 1987.

[8] D.W. Clarke, C. Mohtadi, and P.S. Tuffs. Generalized predictive control - part ii. extensions and interpretations. *Automatica*, 23(2):149–160, March 1987.

[9] Benoit Codrons, Pascale Bendotti, Clement-Marc Falinower, and Michel Gevers. Comparison between model reduction and controller reduction:application to a pressurized

water reactor nuclear plant. *Proceedings of the IEEE Conference on Decision and Control*, 5:4625–4630, 1999.

[10] Hüseyin Demircioğlu and Ercan Karasu. Generalized predictive control. *IEEE Control Systems Magazine*, 20(5):36–47, October 2000.

[11] W. Favoreel, B. De Moor, M. Gevers, and P. van Overschee. Model-free subspace-based lqg-design. Technical report, Katholieke Universiteit Leuven, 1998. ftp://ftp.esat.kuleuven.ac.be/pub/SISTA/favoreel/reports/report98-34I.ps.gz.

[12] W. Favoreel, B. De Moor, and P. van Overschee. Model-free subspace-based lqg-design. *Proceedings of the American Control Conference*, pages 3372–3376, June 1999.

[13] R.P. Feynman, R.B. Leighton, and M. Sands. *The Feynman Lectures on Physics.* Addison-Wesley, 1963.

[14] Gene F. Franklin, J. David Powell, and Abbas Emami-Naeini. *Feedback Control of Dynamic Systems.* Addison-Wesley Publishing Company, 1994.

[15] Gene F. Franklin, J. David Powell, and Michael L. Workman. *Digital Control of Dynamic Systems.* Addison-Wesley Publishing Company, 1990.

[16] Carlos E. Garcia, David M. Prett, and Manfred Morari. Model predictive control: Theory and practice - a survey. *Automatica*, 25(3):335–348, May 1989.

[17] Gene H. Golub and Charles F. Van Loan. *Matrix Computations.* The Johns Hopkins University Press, 1996.

[18] Michael Green and David J.N. Limebeer. *Linear Robust Control.* Prentice Hall, 1995.

[19] M.J. Grimble. Multi-step H_∞ generalized predictive control. *Dynamics and Control*, 8(4):303–339, October 1998.

[20] J.N. Juang. *Applied System Identification.* PTR Prentice-Hall, 1994.

[21] R.L. Kosut. Uncertainty model unfalsification: A system identification paradigm compatible with robust control design. *Proceedings of the IEEE Conference on Decision and Control*, December 1995.

[22] R.L. Kosut. Iterative adaptive robust control via uncertainty model unfalsification. *Proceedings of the International Federation of Automatic Control*, June 1996.

[23] Wiley J. Larson and James R. Wertz Editors. *Space Mission Analysis and Design*. Microcosm, Inc. Kluwer Academic Publishers, 1992.

[24] W.S. Lee, B.D.O Anderson, I.M.Y. Mareels, and R.L. Kosut. On some key issues in the windsurfer approach to robust adaptive control. *Automatica*, 31(11):1619–1636, 1995.

[25] L. Ljung. *System Identification: Theory for the User*. Prentice-Hall, 1987.

[26] P. Lundstrom, J.H. Lee, M. Morari, and S. Skogestad. Limitations of dynamic matrix control. *Computers and Chemical Engineering*, 19(4):409–421, April 1995.

[27] J.M. Maciejowski. *Multivariable Feedback Design*. Addison-Wesley, 1989.

[28] J.P. Norton. *Introduction to Identification*. Academic Press, 1986.

[29] Thomas R. Parks. *Manual for Model 205/205a Torsional Control*. Educational Control Products, 1999.

[30] K. Polla, P. Khargonnekar, A. Tikku, J. Krause, and K. Nagpal. A time-domain approach to model validation. *IEEE Transactions on Automatic Control*, 39(5):951–959, May 1994.

[31] Dan E. Rosenthal. *Experiments in Control of Flexible Structures with Uncertain Parameters*. PhD thesis, Stanford University, 1984.

[32] M.G. Safonov and T.C. Tsao. The unfalsified control concept: A direct path from experiment to controller. *Proceedings of the Conference on Feedback Control, Nonlinear Systems, and Complexity*, May 1994.

[33] M.G. Safonov and T.C. Tsao. The unfalsified control concept and learning. *IEEE Transactions On Automatic Control*, 42(6):843–847, June 1997.

[34] B. Sayyar-Rodsari, J.P. How, Babak Hassibi, and A. Carrier. An H_∞-optimal alternative to the fxlms algorithm. *Proceedings of the American Control Conference*, pages 1116–1121, June 1998.

[35] W.E. Staib and R.B. Staib. A neural network electrode positioning optimization system for the electric arc furnace. *International Joint Conference on Neural Networks*, 111:1–9, 1992.

[36] Robert F. Stengel. *Optimal Control And Estimation.* Dover Publications, Inc., second edition, 1994.

[37] Mark B. Tischler, Joseph T. Driscoll, Mavis G. Cauffman, and Cynthia J. Freedman. Study of bearingless main rotor dynamics from frequency-response wind tunnel test data. *American Helicopter Society Aeromechanics Specialists Conference*, January 1994.

[38] Johnson Tse, Joseph Bentsman, and Norman Miller. Properties of the self-tuning minimax predictive control (mpc). *Proceedings of the 1993 American Control Conference*, pages 1721–1725, 1993.

[39] Peter van Overschee and Bart De Moor. *Subspace Identification for Linear Systems.* Kluwer Academic Publishers, 1996.

[40] Bernard Widrow and Samuel D. Stearns. *Adaptive Signal Processing.* Prentice-Hall, 1985.

[41] Bernard Widrow and E. Walach. *Adaptive Inverse Control.* Prentice-Hall, 1994.

[42] E. Wilson. Adaptive profile optimization for the electric arc furnace. *Steel Technology International*, pages 140–145, 1997.

[43] Haipeng Zhao and Joseph Bentsman. Multivariable h_∞ predictive control based on minimax predictor. *Proceedings of the IEEE Conference on Decision and Control*, 4:3699–3705, 1999.

[44] Kemin Zhou, John C. Doyle, and Keith Glover. *Robust and Optimal Control.* Prentice Hall, 1996.