Robust Discrete Estimation of the Space Shuttle Main Engine

by

Jonathan Andrew Jensen

B.S., Mechanical Engineering,
United States Air Force Academy
(1994)

SUBMITTED TO THE DEPARTMENT OF AERONAUTICS AND ASTRONAUTICS IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 1996

Signature of Author_____
Department of Aeronautics and Astronautics
May 10, 1996

Certified by_____
Roger M. Hain
Technical Staff, C. S. Draper Laboratory
Thesis Supervisor

Certified by_____
Wallace E. Vander Velde
Professor of Aeronautics and Astronautics, Emeritus
Thesis Supervisor

Accepted by_____
Professor Harold Y. Wachman
Chairman, Department Graduate Committee

Robust Discrete Estimation of the Space Shuttle Main Engine

by

Jonathan Andrew Jensen

B.S., Mechanical Engineering,
United States Air Force Academy
(1994)

Submitted to the Department of Aeronautics and Astronautics on May 10, 1996 in partial fulfillment of the requirements for the degree of Master of Science

Abstract

This thesis applies recently developed robust H_∞, or game-theoretic, estimation algorithms to the Space Shuttle Main Engine (SSME). The objective is to process noisy, inaccurate sensor data in order to obtain estimates of pressure in the main combustion chamber and the oxygen to fuel mixture ratio. Each of the estimators are based on discrete-time, state-space models of the SSME, and employ varying levels of robustness when solving the H_∞ estimation problem. Two general problems are examined. First, H_∞ minimax estimators are derived for the case where the plant dynamics are accurately known, but the noise statistics are uncertain. The effects of various noise inputs are explored. Next, robust H_∞ estimators are designed when plant, sensor, and noise uncertainties are present. It is shown that the performance of the normally optimal Kalman filter degrades considerably in the presence of model uncertainty. By contrast, the robust H_∞ estimators perform well for the entire range of plant, sensor, and noise models considered.

Thesis Supervisor:	Roger M. Hain
Title:	Member of the Technical Staff, C. S. Draper Laboratory
Thesis Supervisor:	Wallace E. Vander Velde
Title:	Professor of Aeronautics and Astronautics, Emeritus

Acknowledgments

Naturally there are many who deserve credit for their assistance in this endeavor. To start, I would like to thank Draper Laboratory for giving me this opportunity. Without their resources this thesis would not have been possible. There are several individuals of Draper who deserve recognition for their help, insightful questions, patience, encouragement, and knowledge, including Neil Adams and Brent Appleby. Rami Mangoubi explained his theory, offered suggestions on how to improve my results, and devoted time to careful and detailed proofreading. A special thanks to Roger Hain, who, as my supervisor, not only challenged me technically, but also taught me much in how to listen, how to think, and how to respect others. In addition to the Draper employees, I would like to thank Professor Vander Velde for his careful reading of this thesis and prudent suggestions.

Next I would like to thank some of my peers--my neighbor Jim Dyess, who helped keep MIT in perspective, and Larry McGovern, who repeatedly answered many of my ignorant questions. Scott Carter, my cohort in "general disregard for following procedure," provided a much needed link to the Air Force and inspired me to excellence. I guess we were the off year.

And lastly, thank you is simply not enough for my wife, Katharine, who provided support, encouragement, and love for me as no other could.

This thesis was prepared at The Charles Stark Draper Laboratory, Inc., under Indepent Research and Development CSR Project Number 709.

Table of Contents

List of Figures

List of Tables

1.0 Introduction

For many applications, including control systems, fault detection, and model prediction, it is necessary that states of interest be accurately known for valid results. Unfortunately, there are multiple reasons why obtaining this knowledge can present challenges. For example, sensors may not be available to measure the state, or if one is available, it may be too noisy to trust the measurements it does provide. Observers provide the necessary link to the states by using the system model and the available state measurements to estimate the values of all the system's states.

Estimation theory has progressed greatly since Norbert Wiener developed the first model based estimator at MIT shortly after World War II [22]. Wiener's approach to the filtering problem, which assumes linearity, is to minimize the mean-squared error between the signal and its estimate when given known auto- and cross-correlation functions for the signal and the noise. This was a radical departure from the circuit design filters of the time [2].

An important breakthrough occurred in the early 1960s when Kalman developed the state-space equivalent to Wiener's frequency domain filter [14]. Unlike the unwieldy Wiener filter, the Kalman filter is especially suited to the time domain and digital computer implementation, an increasingly powerful resource as computer speed increases. Like the Wiener filter, the Kalman filter minimizes the mean-squared error. In fact, the Kalman filter is the optimal estimator in the least-squares sense for linear systems with Gaussian disturbances. Ever since Kalman developed his filter for optimal estimation, it has been the most widely used tool for estimating the states of a dynamic system in aerospace applications, such as guidance and navigation and trajectory determination, and has spread to other engineering disciplines as well. Its impact is not limited to engineering, however; it has been applied in such diverse fields as economics, geology, and agriculture [21]. Additionally, the Kalman filter has been the basis for many variations, improving its results for different classes of problems such as smoothing [10], numerical stability [17], and singular measurements [4]. A modification of the Kalman filter, the extended Kalman filter, is also used for obtaining suboptimal filters for nonlinear systems [11].

For many applications, though, two major assumptions relied on by the Kalman filter may be inappropriate. First, the Kalman filter assumes perfect knowledge of the plant dynamics, leaving no room for modeling errors, unmodeled dynamics, etc. Second, it assumes that input disturbances into the system are known well enough that they can be

reduced to being white in nature. For these applications, including the ones examined in this thesis, these are not necessarily the best, or even valid, assumptions. Thus, the need for estimators that are robust to both plant and noise model uncertainties arises. Appleby derived a game-theoretic, continuous-time estimator that is robust to a class of plant and noise model uncertainties for linear systems at steady state [1]. Mangoubi derived both discrete-time and continuous-time estimators that can be applied to linear, time-varying or time-invariant systems defined over a finite or infinite horizon with an arbitrary initial condition [15].

Robust estimation also improves the performance of vehicle health management (VHM) systems. In order to reduce life-cycle costs, future aerospace vehicles will depend on automatic VHM systems to observe the vehicle's subsystems. Not only is VHM expected to identify any off-nominal performance that occurs, but possibly even reconfigure on-line to correct for degraded or failed performance. Off-line, VHM recorded data may warn of impending failure, help with maintenance scheduling, or highlight possible problems that have not yet surfaced [20]. If the filters used by the VHM systems are not robust to model uncertainty, the performance will suffer, as either the false alarm threshold will be unacceptably high, or failures may not be detected promptly. A false alarm could even result in unnecessarily aborting a mission, and a failure that is not detected soon enough may cause a disaster. Even in the absence of disastrous consequences, a poor VHM performance would have the net effect of increasing, rather than decreasing, associated costs. Therefore, developing estimators that are robust to plant model uncertainty allows the monitoring of unanticipated conditions that lead to off-nominal plant operation while decreasing the probability of triggering a false alarm [15]. Estimators robust to noise error extend the range of useful information from a sensor, even though the sensor may have itself experienced some degradation. This estimator would also protect against incorporation of highly inaccurate measurements if the sensor does fail completely. Additionally, the ability of an estimator to be robust to sensor failure would potentially allow a different design approach to VHM systems, allowing the use of a greater number of lower quality, but lower cost, sensors, as opposed to a small number of sensors that are designed at great cost to be highly reliable.

The objective of this thesis is to apply the theory from Mangoubi's work to a validated model for the Space Shuttle Main Engine (SSME) [6] by building robust estimators and demonstrating their improved performance over a wider range of operation when compared to corresponding Kalman filters. The thesis organization follows.

Chapter Two gives a description of the discrete-time state space models for the SSME at various power levels derived in [6]. A sample control input and corresponding outputs for different power levels are replicated from [6]. It also details how the noise models, both process and sensor, are derived for this thesis.

Chapter Three formulates the estimation problem, followed by the derivation of the Kalman filter. The Kalman filter is then applied to the SSME models from Chapter 2, showing advantages and drawbacks to the Kalman filter. Finally, the structure of the H_∞ estimator, and its reliance on the small gain theorem, is presented.

Chapter Four introduces the H_∞/minimax filter solution for discrete-time derived by Mangoubi for known plant models but unknown disturbances. An H_∞/minimax estimator for the SSME is compared to the Kalman filter for white noise and a "worst-case" noise. The effects of estimator's parameter, γ, are also demonstrated, and it is shown that the Kalman filter is a special case of the minimax estimator.

Chapter Five is devoted to the H_∞ estimation problem for the case where both plant and noise model uncertainties are present. Two separate cases of engine operation are considered. First, it is assumed that there is no change in the control input, and the valves remain in the same position. For this case, the uncertainty is modeled parametrically, and the Kalman filter is compared to the H_∞ estimator for different variations. In the second case, the control signal is allowed to alter the valve openings. In the control input case the Kalman filter and the H_∞ filter are again compared.

Finally, Chapter Six summarizes the results from the preceding chapters, and offers some recommendations for further work.

2.0 Model Identification

Proper estimator design relies on prior knowledge of the system. Only after obtaining models for the plant and for the disturbances can one formulate an estimator and evaluate its performance. In order to build this foundation, Section 2.1 gives a brief description of the SSME and Section 2.2 describes the state space models for the system. Since no noise model is included in the system descriptions in [6], noise models are generated in Section 2.3.

2.1 Space Shuttle Main Engine

Three Space Shuttle Main Engines comprise the primary propulsion system for the space shuttle orbiter, but are only used during launch. During usage the engines are operated at various power levels, identified as percentages of rated thrust levels. After reaching some predetermined speed the main engines are turned off and the fuel tank is discarded [13]. After reaching orbit, attitude correction is done with a system of 44 smaller jets composing the Reaction Control System (RCS). Many of these smaller jets are also used during re-entry of the orbiter before atmospheric dynamic pressures fully enable the use of aerosurfaces, making these jets unnecessary. A recent study of estimating the RCS for vehicle health monitoring during re-entry using aerosurface commands and space shuttle orbiter dynamic responses is found in [20].

Each of the engines uses liquid oxygen and liquid hydrogen from the external fuel tank that is attached to the underside of the orbiter during its ascent. Figure 2.1 shows a schematic of the SSME. To summarize the main fuel and oxidizer flows, the liquids first enter their respective low pressure turbopumps, and, upon exiting, proceed directly to the two high pressure turbopumps. The output from each high pressure turbopump is divided and used for many different purposes. Some of the fuel flow from the high pressure fuel turbopump, controlled by the main fuel valve (MFV), is channeled into cooling the main combustion chamber and the nozzle, and returns to drive the low pressure fuel turbopump. The rest of the fuel is allowed to bypass the cooling flows via the chamber coolant valve (CCV), and is injected into the preburners. Similarly, some of the oxidizer is sent to the preburners through the fuel preburner oxidizer valve (FPOV) and the oxidizer preburner oxidizer valve (OPOV). The preburners, the location of the first stage of combustion, produce heated hydrogen-rich gas that drives the high pressure turbopumps on its way to the main combustion chamber. The rest of the oxidizer from the high pressure oxidizer

turbopump is controlled by the main oxidizer valve (MOV), which regulates the amount of oxidizer in the main combustion chamber [13]. The present control system of the SSME uses these five valves (FPOV, OPOV, MFV, MOV, CCV) as inputs to regulate the mixture ratio (MR) and chamber pressure (P_c) in the main combustion chamber. These values are closely associated with the final thrust produced by the SSME [6].

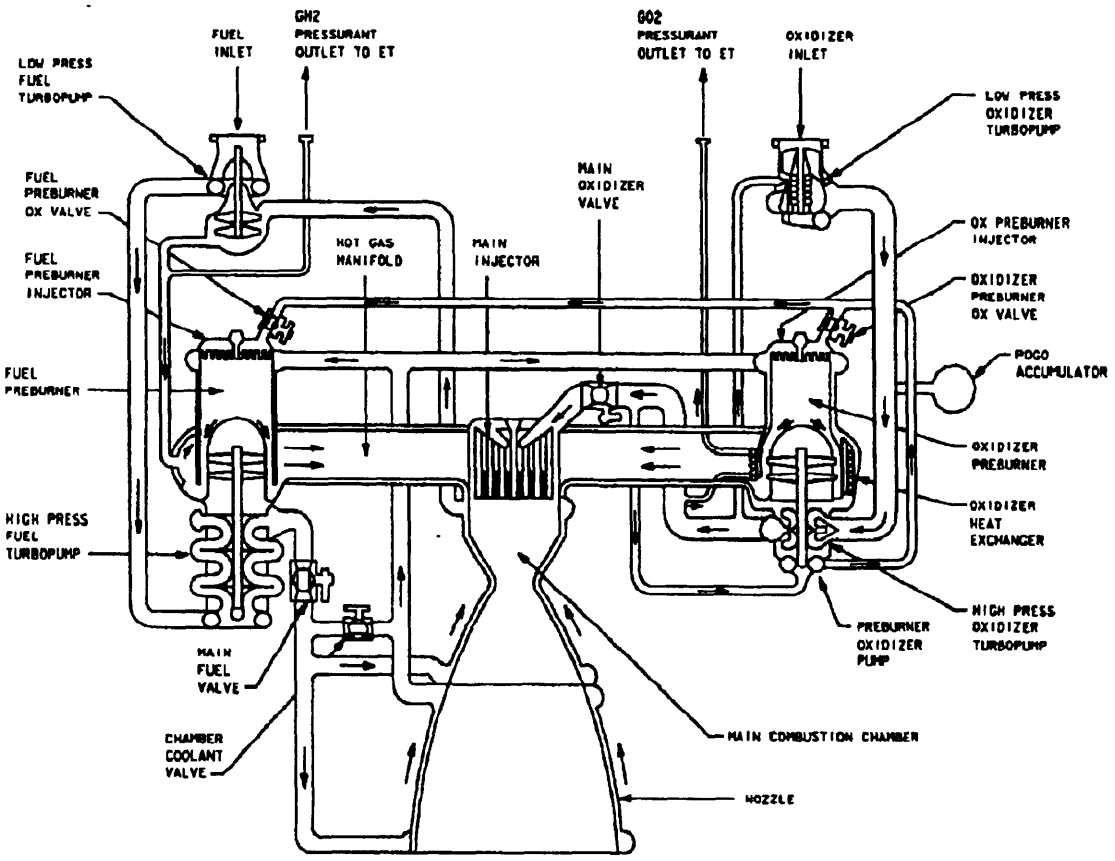

Figure 2.1: Schematic of the Space Shuttle Main Engine [13]

Figure 2.2 shows a typical thrust profile for an SSME. The time span between 30 seconds and 90 seconds when engine power levels drop from 100% to 65% rated power level (rpl) is known as the "thrust bucket." During this time period the orbiter passes through its highest dynamic pressure. In order to combat the increased structural loading due to the dynamic pressure, defined by $\frac{1}{2}\rho V^2$, where ρ is atmospheric density and V is shuttle velocity, the shuttle reduces its engine power level. This reduces the gain in velocity until the density decreases enough through altitude gains in order to safely return the power level to 104% of rated capacity for approximately the next 400 seconds. During

this and other periods of constant power level, only small corrective control inputs are needed to maintain power levels. This is in contrast to the stages where heavy control inputs are necessary to rapidly change the engine's power level, such as the thrust bucket and when the engines are throttled back in preparation for shutdown.

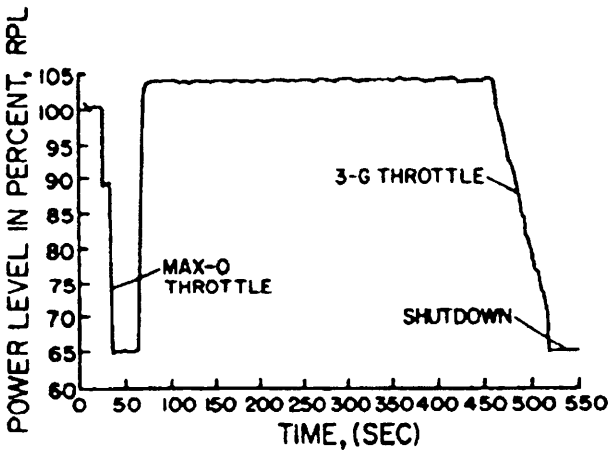

Figure 2.2: Thrust Profile for SSME Mission [18]

2.2 Description of System Matrices

The description of the SSME shows a complex system, so it is important to develop a model which, while allowing proper estimator design, is simple enough to implement. The Rocketdyne Division of Rockwell International has developed a full nonlinear model that completely describes the SSME [19], but the model's size and complexity become serious disadvantages for design work. Instead of using Rocketdyne's nonlinear model, the models used in this thesis are the open-loop space shuttle main engine models from the work completed in [6]. As a simplification, the authors in [7] show that during the "main stage of operation" the rotary motions of the CCV ($ß_{CCV}$), the MOV ($ß_{MOV}$), and the MFV ($ß_{MFV}$) are essentially decoupled from the outputs. The main stage of operation is the period following the initial start-up procedures of the SSMEs until shutdown, and includes the entire mission profile shown in Figure 2.2 (start-up procedures precede the curve in the figure). This leaves the rotary motion of the two remaining valves, $ß_{FPOV}$ and $ß_{OPOV}$, as the input.

In their derivation, Duyar et al. identified the steps taken to achieve their results. In order to obtain equations approximating the response of the full nonlinear SSME model, the nonlinear model response is identified based on its output to a specific input signal.

First, a driving signal of pseudorandom binary sequences (PRBS) is selected, with a separate PRBS used for each input channel. These signals have a clock time of 0.04 seconds, which corresponds to a maximum frequency of 78.5 rad/sec before aliasing occurs. Since the models are derived from these signals, the validity of the models also is limited to frequencies less than 78.5 rad/sec.

Next, the simplified models were derived. In their paper, Duyar et al. describe in detail the methodology they used to convert the nonlinear, continuous-time system

$$\begin{aligned} \dot{x}(t) &= f[x(t), u(t)] \\ y(t) &= g[x(t)] \end{aligned} \tag{2.2-1}$$

to the discrete-time, linear time-invariant models given by

$$\begin{aligned} \delta x_{k+1} &= A\delta x_k + B\delta u_k \\ \delta y_k &= C\delta x_k \end{aligned} \tag{2.2-2}$$

The discrete-time state-space matrices for each of these power levels are in Table 2.1.

Power Level	A	B	C
70% Power	0 0 -0.2553 0.0416 0 0 0.5044 -0.1108 1 0 1.0218 -0.1130 0 1 -0.7902 0.5812	0.3923 0.0958 1.0558 -0.2652 0.2584 0.0788 0.5902 -0.2697	0 0 1 0 0 0 0 1
80% Power	0 0 -0.1576 0.0257 0 0 0.2587 -0.0545 1 0 0.8759 -0.0666 0 1 -0.5157 0.4766	0.4068 0.1564 0.9631 -0.6740 0.2574 0.0384 0.6046 -0.2307	0 0 1 0 0 0 0 1
90% Power	0 0 -0.1206 0.0155 0 0 0.1434 -0.0275 1 0 0.8117 -0.0478 0 1 -0.4183 0.4226	0.4223 0.1546 0.9217 -0.7302 0.2518 0.0210 0.6288 -0.1779	0 0 1 0 0 0 0 1
100% Power	0 0 -0.1279 0.0212 0 0 0.1994 -0.0508 1 0 0.7816 -0.0565 0 1 -0.5253 0.4750	0.3534 0.0475 0.6831 -0.6446 0.2007 -0.0181 0.5043 -0.1387	0 0 1 0 0 0 0 1
110% Power	0 0 -0.0840 0.0026 0 0 0.1082 -0.0123 1 0 0.7280 0.0181 0 1 -0.4164 0.3330	0.2416 0.0786 0.5112 -0.5750 0.1456 -0.0243 0.3593 -0.0780	0 0 1 0 0 0 0 1

Table 2.1: Discrete-time state-space matrices at varying power levels

As previously mentioned, the deviation in control inputs, δu_k, in Eq. (2.2-2) are the rotary motions of the fuel preburner oxidizer valve and the oxidizer preburner oxidizer valve, given by $ß_{FPOV}$ and $ß_{OPOV}$. The ouput vector, δy_k, has the elements chamber pressure, P_C, and the mixture ratio, MR. The first and second states are mathematical states, and do not carry direct physical analogs [9]. The authors repeat the model identification for five different power levels, and each of the models is valid around the corresponding equilibrium point. For the remainder of the thesis, the δ is dropped from the state-space equations.

The simplified SSME models in Table 2.1 were used in MATLAB as the plant models for this thesis. While the models are designed to mimic the high-frequency PRBSs, Duyar et al. also use a low frequency series of step and ramp inputs (see Figure 2.3) to demonstrate the validity of their linearized models. This series of step and ramp inputs was also selected as the control input, u_k, for the MATLAB models. Given this input, a sample output for the 80% power level model is given in Figure 2.4. This output agrees closely with the output in [6], indicating that the MATLAB models were indeed the same as the validated models.

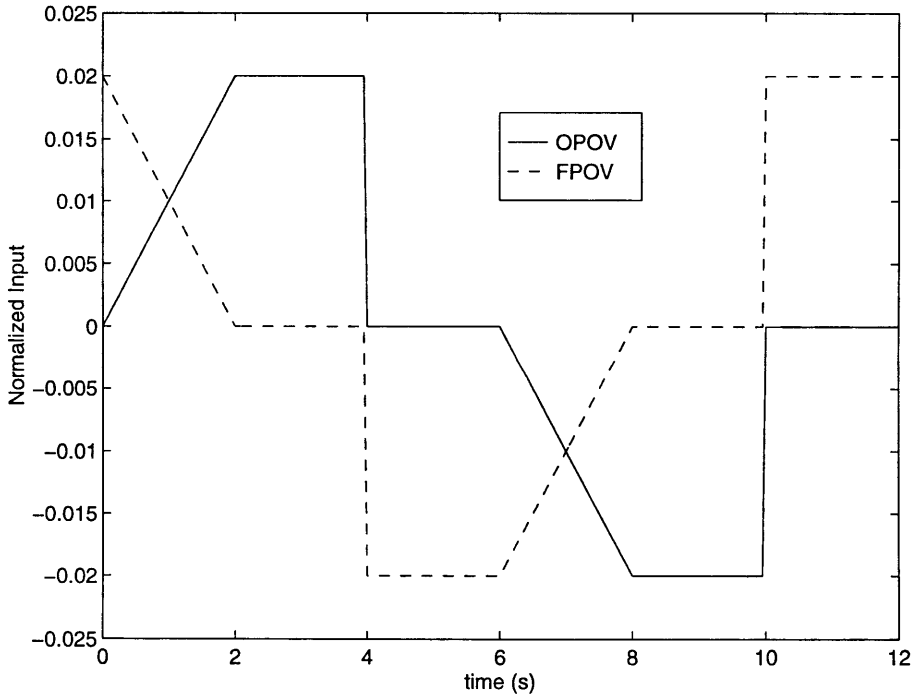

Figure 2.3: Normalized rotary valve input to open-loop system

While Figure 2.4 only shows the response for the 80% power level model, all of the responses for each power level follow the same general shape. It is important to note that both the inputs and the outputs are already normalized. The normalization is accomplished according to:

$$\delta x_k^{ij} = \frac{x_k^j - x_{ss}^{ij}}{x_{ss}^{ij}} \tag{2.2-3}$$

where i is the operating point, j is the component of the state vector, k is the time step, ss denotes the steady state value, and, for this equation, x denotes the actual state values (not the deviations). This normalization equation becomes important later during frequency response analysis. Since the outputs are normalized to the same order of magnitude, the singular values will not be distorted by different units (psi for P_c and unitless for MR).

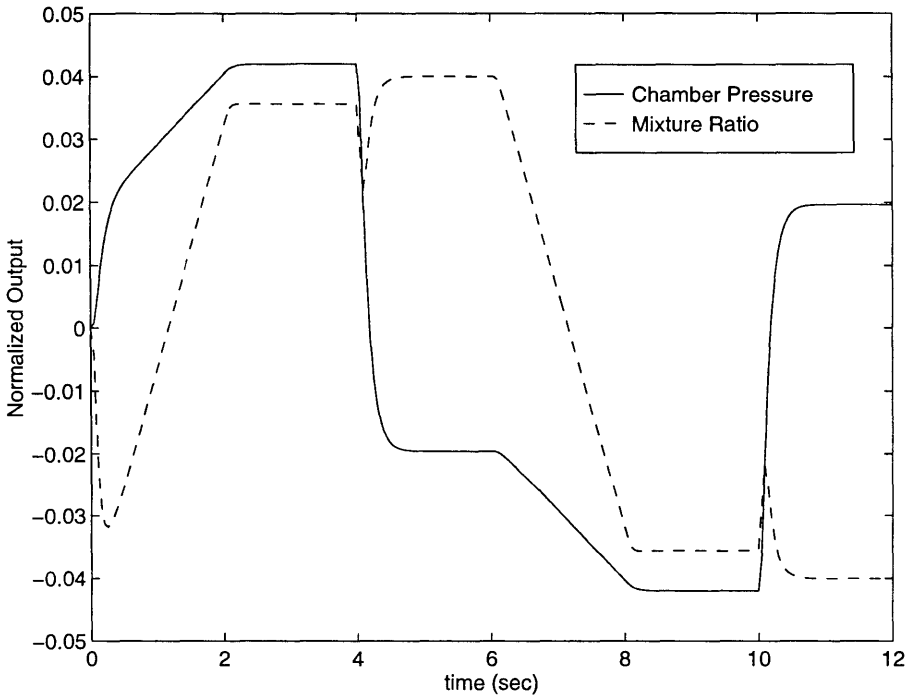

Figure 2.4: Normalized open-loop output for the 80% power level model

2.3 Noise Models

The models identified in [6] are for the open-loop system, and do not account for any noise. In general, this is not a realistic situation. To rectify this drawback a more typical system is defined as

$$x_{k+1} = Ax_k + Bu_k + Gd_k$$
$$y_k = Cx_k + Du_k + Ed_k \tag{2.3-1}$$

where d_k is some input noise vector. In addition to control input, the states are now affected by process noise (Gd_k). Process noise has many different sources and can manifest itself in many different ways. Some examples of phenomena that can be modeled as process noise are unmodeled dynamics, modeling errors, input disturbances (e.g., waves against a ship or air gusts on an airplane). In the case of the SSME, a constant pressure is assumed in the fuel supply. This is not the case, however. As the liquid hydrogen and oxygen boil, the pressure increases, and the gases must be vented to maintain the constant pressure. This appears as disturbances to the plant [16]. In addition to process noise, the measurements are subject to some sensor noise (Ed_k). Sensor noise stems from the inability of the sensor to accurately measure the states. Sensor noise can be from a faulty sensor, one that is improperly located, or from a sensor whose performance degrades with time or environmental conditions, as is the case in the SSME [16]. In order to design and compare estimators it is crucial to determine a set of values for these noises. Since noise models are not present in [6], their derivation is part of this thesis work.

To develop a noise model for use with the MATLAB SSME models, the first assumption made was to divide the noise vector at each step, d_k, into separate process and measurement noises such that

$$d_k = \begin{bmatrix} w_k \\ v_k \end{bmatrix} \tag{2.3-2}$$

where w_k is the portion of d_k related to process noise and v_k is likewise associated with the measurement noise. Next, an assumption was made to model the noise vector, d_k, as a Gaussian disturbance, i.e.,

$$d_k \sim N(0, \sqrt{I}) \tag{2.3-3}$$

and the process $\{d_k\}$ is white with unit intensity according to

$$E[d_k d_l] = I\delta_{kl} \tag{2.3-4}$$

where δ_{kl} is the Dirac delta function

$$\delta_{kl} = \begin{cases} 1 & k = l \\ 0 & k \neq l \end{cases} \tag{2.3-5}$$

Furthermore, it is assumed that the process noise and the sensor noise are uncorrelated, resulting in

$$GE^T = 0 \tag{2.3-6}$$

The states and the noises at the same time step, k, are also uncorrelated. This allowed for further detailing of each type of noise.

2.3.1 Process noise

It was initially decided to only give process noise to the same channels as the control inputs, defining the process noise matrix, G, as some scalar multiple of B. Since d_k is comprised of both process and sensor noise, this translated into G equal to some matrix $[\alpha B \ 0]$. The process noise corresponds to some degree of uncertainty in the actuators, or more specifically in this application, there is some Gaussian jitter associated with the two control valves.

After determining the nature of the process noise and how it enters the system, the magnitude of G, as defined by α, next became an issue. Since the process noise is associated with the control input, its intensity was a function of the peak values of the inputs (which never exceed ±0.02). By defining two standard deviations of the process noise to be 10% of the peak values for the two normalized inputs, the process noise has a 95% chance of being within ±0.002 of the control signals. Thus, the standard deviation of the process noise is 0.001. By setting the constant for multiplying B equal to the standard deviation (i.e., $\alpha = 0.001$) the process noise now meets the requirements, and its intensity was equal to GG^T. Figure 2.5 shows the output states without any noise compared to the states with process noise added.

2.3.2 Sensor noise

After defining the process noise, the next step was to determine the sensor noise, Ed_k. Again, with no noise model to work with, several assumptions were made. First, the measurement noise for each channel was assumed to be uncorrellated with other channels (as well as the process noise), allowing the basis for the sensor noise matrix, E, to be a diagonal matrix. Since d_k is comprised of both process and sensor noise, this translated into E equal to some matrix $[0 \ \beta]$, where β is a diagonal matrix that multiplies v_k. Two different sensor models were then developed, corresponding to "good" and "bad" sensors. These sensors were purely arbitrary, based on engineering judgment of what "good" and "bad" meant in comparison to the actual state values.

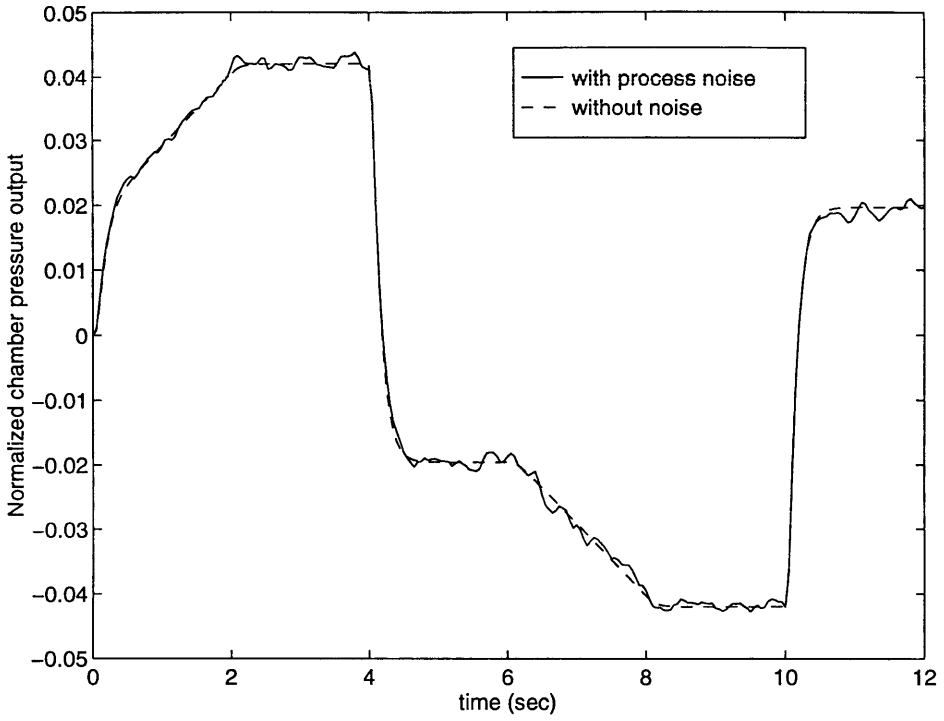

Figure 2.5: Output at 80% power level with and without process noise

The good sensor E matrix is defined as two times the identity matrix. In order to keep the sensor noise and the process noise on the same order of magnitude, E is also multiplied by $\alpha = 0.001$. The bad sensor is developed identically except that it is five times the identity instead of two times the identity. The resulting sensors are therefore

$$E_{good} = \begin{bmatrix} 0 & 0 & 0.002 & 0 \\ 0 & 0 & 0 & 0.002 \end{bmatrix}$$
$$E_{bad} = \begin{bmatrix} 0 & 0 & 0.005 & 0 \\ 0 & 0 & 0 & 0.005 \end{bmatrix} \qquad (2.3\text{-}7)$$

Both sensors have a measurement noise intensity of EE^T. Figure 2.6 shows the same output (normalized chamber pressure) as Figure 2.5, but now additional curves show the effect of the good sensor and the bad sensor. Similarly, Figure 2.7 shows the effect of the noise on the mixture ratio output.

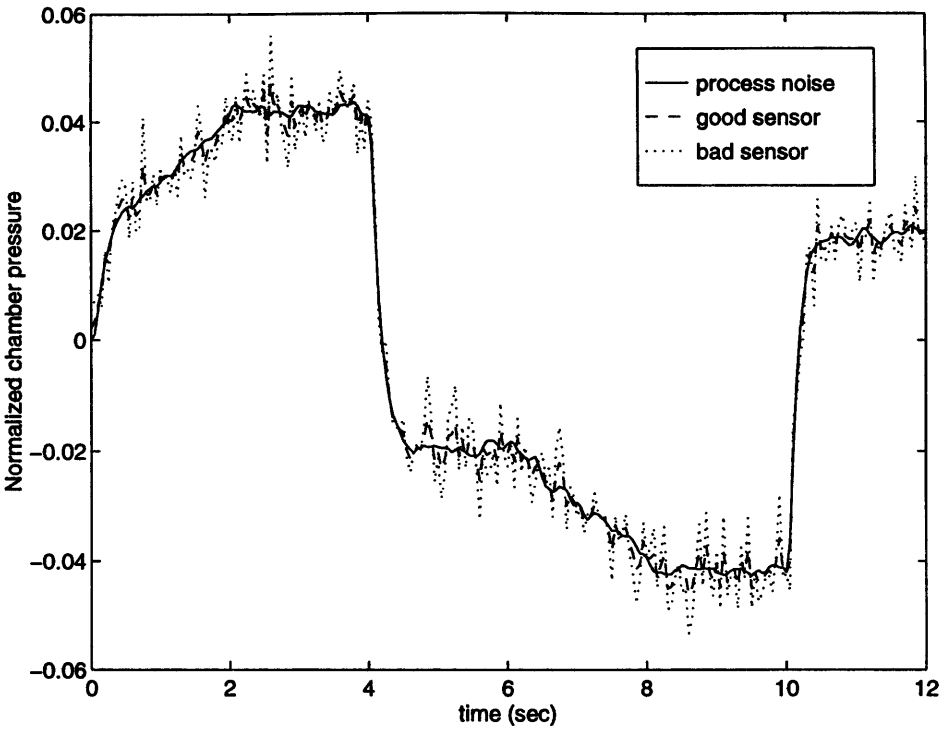

Figure 2.6: Chamber pressure with noises added

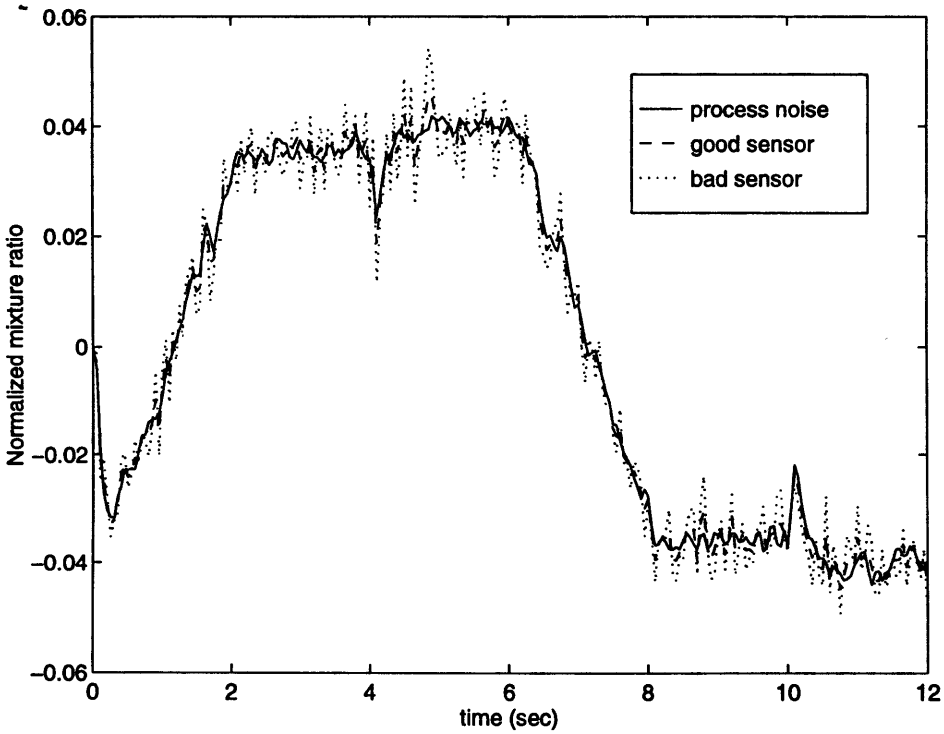

Figure 2.7: Mixture ratio with noises added

3.0 Problem Formulation

The linear models described in the previous chapter can now be used to derive the desired filter. In Section 3.1 a derivation of the Kalman filter as a solution to its associated estimation problem is presented with results showing its optimality for perfect knowledge of the plant and noise models as well as its limitations when given a perturbed system. Section 3.2 develops the framework for the H_∞ estimator as an alternative to the Kalman filter.

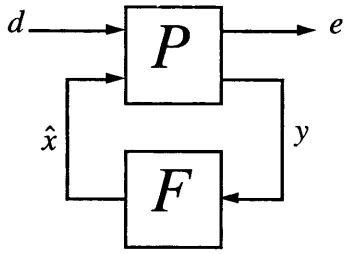

Figure 3.1: Block diagram of estimation problem

In general, the estimators have the structure in Figure 3.1. The plant, $\boldsymbol{P}$, is given some input disturbance vector at time k, d_k, and possibly also some control input, u_k. The filter, $\boldsymbol{F}$, uses the output of the plant, y_k, (and knowledge of the control input if applicable) to generate state estimates, $\hat{x}_k$, by incorporating those measurements into some knowledge of the plant. An additional output is the error vector, e_k, which is a weighted difference between the actual states and the estimates according to

$$\begin{aligned} e_k &\equiv M(x_k - \hat{x}_k) \\ &= M\tilde{x}_k \end{aligned} \tag{3.0-1}$$

where the error observation matrix, M, is an arbitrary matrix. M can be adjusted to select only certain states, a combination of states, or even to weigh certain states more heavily than others. This error vector can be thought of as a performance vector. It is important to note that the estimate can be obtained at two different points in time associated with the time step, k. It can be determined before the measurements at time k, y_k, are incorporated (*a priori*), or it can be determined after incorporating y_k (*a posteriori*). In this thesis all equations are given in *a priori* form unless otherwise noted with a "super +" indicating that the equation is *a posteriori*.

3.1 Kalman Filter

The Kalman filter is perhaps the best known estimator for states of linear dynamic systems, spanning both time and disciplines in its applications. It is a recursive (meaning that there is no need to store past data) filter that is the optimal estimator in a least-squares error sense for linear Gaussian systems where plant dynamics and disturbance statistics are known. This means that the mean-squared error is minimized over some time period. Put another way, the Kalman filter minimizes the expected value of the estimation error 2-norm for at finite time interval [1, N]

$$E\left[\|e\|_2^2\right] \equiv E\left[\sum_{k=1}^{N} e_k^T e_k\right] \tag{3.1-1}$$[1]

which can be reformulated as

$$\begin{aligned} E\left[\sum_{k=1}^{N} trace\left\{e_k e_k^T\right\}\right] &= \sum_{k=1}^{N} trace\left\{E\left[e_k e_k^T\right]\right\} \\ &= \sum_{k=1}^{N} trace\left\{cov\left[e_k\right]\right\} \end{aligned} \tag{3.1-2}$$

where the error vector is defined above and *cov* denotes the covariance matrix of e_k. When converted into the frequency domain, the steady-state Kalman filter also minimizes the H_2-norm of the transfer function from the disturbance to error [1].

3.1.1 Derivation of Kalman filter

This derivation of the Kalman filter does not include any control signal, u_k, although it is easy to show how the Kalman filter accounts for it.

The equation below represents the system associated with Figure 3.1,

$$\left[\begin{array}{c} x_{k+1} \\ \hline e_k \\ y_k \end{array}\right] = \left[\begin{array}{c|cc} A_k & G_k & 0 \\ \hline M_k & 0 & -M_k \\ C_k & E_k & 0 \end{array}\right] \left[\begin{array}{c} x_k \\ \hline d_k \\ \hat{x}_k \end{array}\right] \tag{3.1-3}$$

with the system and noise matrices defined in Chapter 2. At each time step the actual unweighted estimation error is the difference between the state and the state estimate,

[1]Transposes of vectors and matrices will be denoted by a superscript T. The "prime" symbol will be reserved to denote some alteration to the original matrix.

$$\tilde{x}_k = x_k - \hat{x}_k \tag{3.1-4}$$

Assuming that the estimator is unbiased, the associated error covariance is defined as

$$P_k = E\left[\tilde{x}_k \tilde{x}_k^T\right] \tag{3.1-5}$$

The *a posteriori* filter uses an additional measurement to improve the estimate according to the form

$$\begin{aligned} \hat{x}_k^+ &= \hat{x}_k + K_k(y_k - \hat{y}_k) \\ &= \hat{x}_k + K_k(y_k - C_k\hat{x}_k) \end{aligned} \tag{3.1-6}$$

where K_k is some undetermined gain. The updated error equation from Eq. (3.1-4) and Eq. (3.1-6) becomes

$$\begin{aligned} \tilde{x}_k^+ &= x_k - \hat{x}_k - K_k(y_k - C_k\hat{x}_k) \\ &= (I - K_kC_k)\tilde{x}_k - K_kE_kd_k \end{aligned} \tag{3.1-7}$$

This updated error can then be used to solve for the updated error covariance (recognizing that *a priori* error and noise are uncorrelated)

$$\begin{aligned} P_k^+ &= E\left[\tilde{x}_k^+ \tilde{x}_k^{+T}\right] \\ &= E\left[\left\{(I - K_kC_k)\tilde{x}_k - K_kE_kd_k\right\}\left\{(I - K_kC_k)\tilde{x}_k - K_kE_kd_k\right\}^T\right] \\ &= (I - K_kC_k)P_k(I - K_kC_k)^T + K_kE_kE_k^TK_k^T \end{aligned} \tag{3.1-8}$$

The covariance is now quadratic in K. Since the trace of the covariance matrix represents the total squared error it is the value to be minimized. By setting the derivative of the trace of P_k^+ with respect to K to zero the optimal K can be determined (that is, the gain that minimizes the squared error). Using the property in [11] that

$$\frac{\partial\left[trace\left(ABA^T\right)\right]}{\partial A} = 2AB \tag{3.1-9}$$

this derivative is

$$\frac{\partial\left(trace\left(P_k^+\right)\right)}{\partial K_k} = -2(I - K_kC_k)P_kC_k^T + 2K_kE_kE_k^T \tag{3.1-10}$$

Setting Eq. (3.1-10) equal to zero and solving for K_k yields

$$K_k = P_k C_k^T \left(C_k P_k C_k^T + E_k E_k^T \right)^{-1} \tag{3.1-11}$$

This is known as the Kalman gain matrix and is the gain that minimizes the squared error of the estimates. Substituting this K_k back into Eq. (3.1-8) reduces to

$$\begin{aligned} P_k^+ &= (I - K_k C_k) P_k (I - K_k C_k)^T + K_k E_k E_k^T K_k^T \\ &= (I - K_k C_k) P_k \end{aligned} \tag{3.1-12}$$

Since the second term, $K_k C_k P_k$, is positive semidefinite, a measurement will at the *worst* cause no change in the error covariance (which would happen, for instance, when $E_k \to \infty$ so that K_k = 0). Since this would be extremely unlikely , the measurement normally reduces the error covariance. This is intuitive--the measurement should add some knowledge of the system which in turn the filter uses to improve its estimate.

After updating the current time step with measurements, the estimate is propagated to the next time step, k+1, according to

$$\hat{x}_{k+1} = A_k \hat{x}_k^+ \tag{3.1-13}$$

which leads to the new updated estimation error

$$\tilde{x}_{k+1} = x_{k+1} - \hat{x}_{k+1} \tag{3.1-14}$$

Substituting Eq. (3.1-3) and Eq. (3.1-13) into Eq. (3.1-14) yields the error dynamics

$$\begin{aligned} \tilde{x}_{k+1} &= A_k x_k + G_k d_k - A_k \hat{x}_k \\ &= A_k \tilde{x}_k + G_k d_k \end{aligned} \tag{3.1-15}$$

Noise is not included in the estimator due to its zero-mean unbiased nature. The associated error covariance matrix becomes

$$\begin{aligned} P_{k+1} &= E\left[\tilde{x}_{k+1} \tilde{x}_{k+1}^T \right] \\ &= E\left[(A_k \tilde{x}_k + G_k d_k)(A_k \tilde{x}_k + G_k d_k)^T \right] \\ &= A_k P_k A^T + G_k G_k^T \end{aligned} \tag{3.1-16}$$

The Kalman filter is comprised of Equations (3.1-6), (3.1-11), (3.1-12), (3.1-13), and (3.1-16).

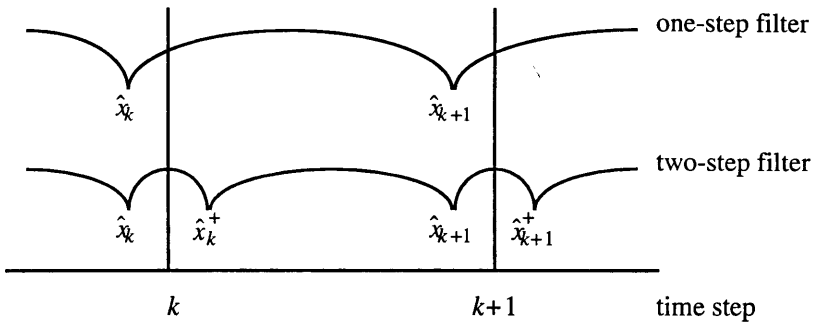

Figure 3.2: Two-step Kalman filter vs. one-step Kalman filter

This two-step process of updating and propagating can be merged to form a Kalman filter with only one step for each time step. Figure 3.2 shows the conceptual difference between a one-step filter and a two-step filter. By combining Eqs. (3.1-6) and (3.1-13) the one-step filter equation is

$$\begin{aligned}\hat{x}_{k+1} &= A_k\left(\hat{x}_k + K_k\left(y_k - C_k\hat{x}_k\right)\right) \\ &= \left(A_k - A_k K_k C_k\right)\hat{x}_k + A_k K_k y_k\end{aligned} \tag{3.1-17}$$

Likewise, by substituting Eq. (3.1-6) into Eq. (3.1-15), a one-step error equation can be obtained

$$\begin{aligned}\tilde{x}_{k+1} &= A_k x_k + G_k d_k - A_k\left[\hat{x}_k + K_k\left(y_k - C_k\hat{x}_k\right)\right] \\ &= \left(A_k - A_k K_k C_k\right)\tilde{x}_k + \left(G_k - A_k K_k E_k\right)d_k\end{aligned} \tag{3.1-18}$$

Incorporating the matrix A and the Kalman gain matrix into a new Kalman gain matrix, K_k', gives

$$\begin{aligned}\hat{x}_{k+1} &= \left(A_k - K_k' C_k\right)\hat{x}_k + K_k' y_k \\ \tilde{x}_{k+1} &= \left(A_k - K_k' C_k\right)\tilde{x}_k + \left(G_k - K_k' E_k\right)d_k\end{aligned} \tag{3.1-19}$$

This gives a one-step error covariance equation as

$$\begin{aligned}P_{k+1} &= E\left[\left\{\left(A_k - K_k' C_k\right)\tilde{x}_k + \left(G_k - K_k' E_k\right)d_k\right\}\left\{\left(A_k - K_k' C_k\right)\tilde{x}_k + \left(G_k - K_k' E_k\right)d_k\right\}^T\right] \\ &= \left(A_k - K_k' C_k\right)P_k\left(A_k - K_k' C_k\right)^T + \left(G_k - K_k' E_k\right)\left(G_k - K_k' E_k\right)^T \\ &= \tilde{A}_k P_k \tilde{A}_k^T + \tilde{G}_k \tilde{G}_k^T\end{aligned} \tag{3.1-20}$$

The one-step Kalman filter is formed by Eqs. (3.1-19) and (3.1-20), and should give the same results as its underlying two-step relative.

3.1.2 Application of Kalman filter

The Kalman filter is now applied to the SSME models obtained in Chapter 2. Since the matrices from Chapter 2 are time invariant, the subscript k can be dropped for the matrices (although it must be retained for the vectors). When the models are fully known the state and error can be combined to form an augmented system of the form in Eq. (3.1-21). It can be clearly seen that there is no interaction between the states and the estimation error.

$$\begin{bmatrix} x_{k+1} \\ \tilde{x}_{k+1} \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & A-K'C \end{bmatrix} \begin{bmatrix} x_k \\ \tilde{x}_k \end{bmatrix} + \begin{bmatrix} B & G \\ 0 & G-K'E \end{bmatrix} \begin{bmatrix} u_k \\ d_k \end{bmatrix} \tag{3.1-21}$$

Using the 110% power level as the nominal system (now with some control signal), the results of its corresponding Kalman filter can be seen in Figure 3.3. Even in the case of a Kalman filter designed for the bad sensor the estimation error is minimal as it relies on an accurate model. So when the models are known, the Kalman filter does an excellent job of estimating the states. Table 3.1 summarizes the squared error over the 12 second time span for each state for two different Kalman filters, one for the good sensor model and one for the bad sensor model.

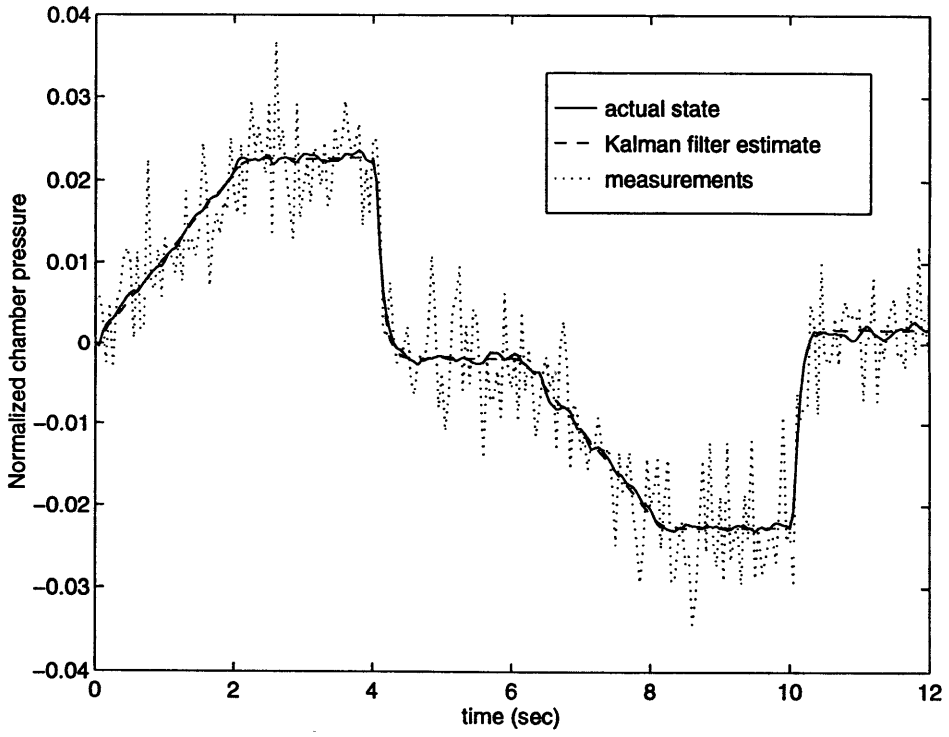

Figure 3.3: Kalman filter for the 110% power level model

	State	Squared error
Kalman Filter designed for good sensor model	Chamber Pressure	0.0483×10^{-3}
	Mixture Ratio	0.2034×10^{-3}
Kalman Filter designed for bad sensor model	Chamber Pressure	0.0514×10^{-3}
	Mixture Ratio	0.2117×10^{-3}

Table 3.1: Squared estimation error of Kalman filter at 110% power level

The Kalman filter's limitations become apparent, however, when the system it is trying to estimate is not the one for which it is designed. The Kalman filter, still believing it is estimating the nominal system, does very poorly when the actual system is some perturbed system. The augmented dynamics, now with a perturbation in A, are shown in Eq. (3.1-22).

$$\begin{bmatrix} x_{k+1} \\ \tilde{x}_{k+1} \end{bmatrix} = \begin{bmatrix} A+\Delta A & 0 \\ \Delta A & A-K'C \end{bmatrix} \begin{bmatrix} x_k \\ \tilde{x}_k \end{bmatrix} + \begin{bmatrix} B & G \\ 0 & G-K'E \end{bmatrix} \begin{bmatrix} u_k \\ d_k \end{bmatrix} \tag{3.1-22}$$

Even if ΔA is small, the Kalman filter is designed for A, and will not be the optimal estimator for the perturbed system. The equations for the Kalman filter in the previous section provide no method for incorporating the ΔA in the lower left corner in the augmented matrix. This is when the Kalman filter's limitations become apparent. The error increases in magnitude as more matrices are perturbed (i.e., some ΔB, ΔG, ΔC, etc. exists in addition to ΔA) and the perturbations for each matrix grow.

Figures 3.4 and 3.5 show the performance of the Kalman filter designed for the 110% power level model with a good sensor when the actual system is at 90% power level with a good sensor (thus, according the matrix definitions of Chapter 2, incurring some ΔA, ΔB, ΔG, and ΔD). Table 3.2 displays the squared error. Compared to the error in Table 3.1 the error is much larger (which is also readily seen in both figures). The Kalman filter can never adjust to the perturbed system, especially to the ΔB term, and is often closer to the nominal plant (for which the Kalman filter was designed) than to the output from the actual perturbed system. It is errors such as these that the H_∞ estimator hopes to address by building in robustness to perturbations.

	State	Squared error
Perturbed Kalman Filter with good sensor	Chamber Pressure	0.0437
	Mixture Ratio	0.0458

Table 3.2: Squared estimation error of Kalman filter designed for 110% at 90%

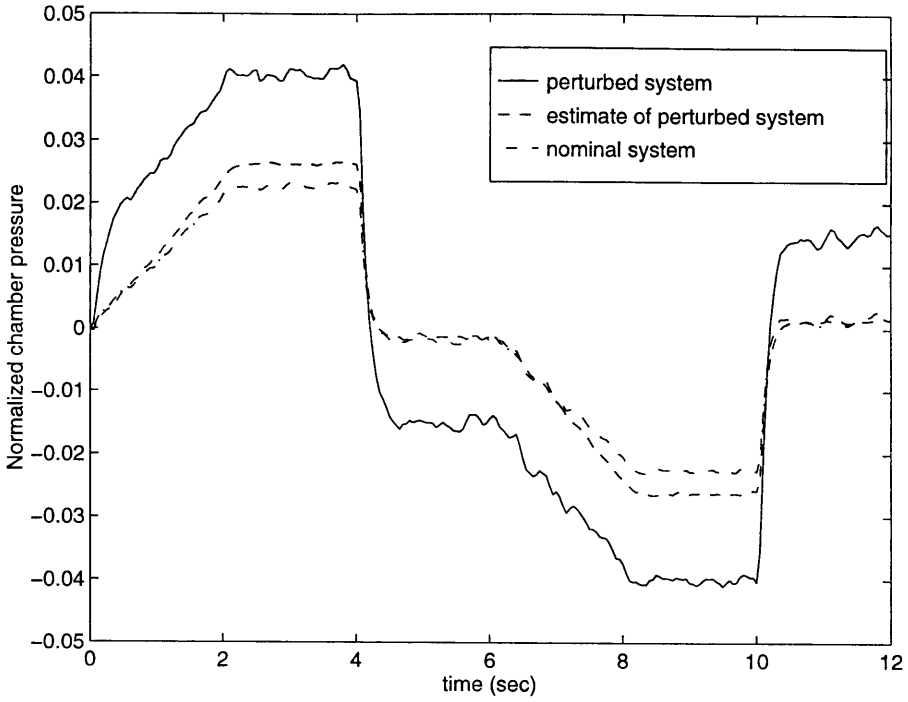

Figure 3.4: Kalman filter estimates of chamber pressure for a perturbed system

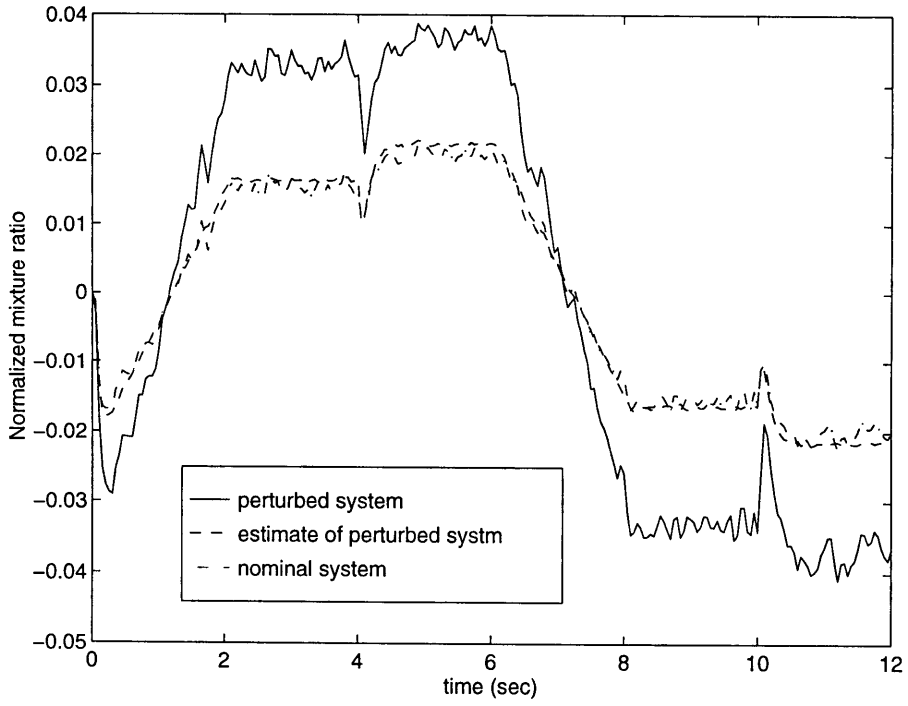

Figure 3.5: Kalman filter estimates of mixture ratio for a perturbed system

3.2 Small Gain Theorem and H_∞ estimation

An alternative to Kalman filters is to use H_∞ estimation, a form of robust estimation that allows for uncertainties in the models. Figure 3.6 shows the robust estimation problem. Now the Δ-block represents the uncertainty in the models, with ε and η representing the interation between the uncertainty and the plant, P. The goal is to minimize some induced norm for the transfer function, G, that is from the the disturbance, d, to the error, e, for all possible perturbations in Δ.

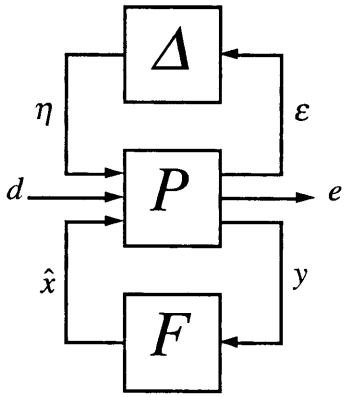

Figure 3.6: Block diagram of robust estimation problem

Estimators concerned with the H_∞ norm rely on the small gain theorem [3] for their performance guarantees. The small gain theorem states that a stable open-loop system will have a bounded induced norm if its feedback loop is composed entirely of operators whose induced norm are each less than one. The definition of an induced norm (a vector norm extended to matrices but still defined in terms of vector norms) is

$$\|A\|_{ip} = \sup_{x \neq 0} \frac{\|Ax\|_p}{\|x\|_p} \tag{3.2-1}$$

where A is a matrix, x is a vector, and p is the "order" of the norm. Since the plant is a linear, time-invariant system, the norm of interest in this thesis is the ∞-norm, or

$$\begin{aligned} \|G\|_\infty &\equiv \sup_\omega \sigma_{\max}\left(G(j\omega)\right) \\ &= \sup_{d \neq 0} \frac{\|e\|_2}{\|d\|_2} \end{aligned} \tag{3.2-2}$$

The estimator problem over a finite time interval $[1, N]$ now becomes that of finding estimates $\hat{x}_k, k = 1, 2, \ldots N$ such that

$$\frac{\|e\|_\infty}{\|d\|_\infty} < \gamma \quad \forall\, d \tag{3.2-3}$$

It is important to note that even though the system used in this thesis is time-invariant the theory that follows is applicable to time-varying systems with arbitrary initial conditions.

Compared to the Kalman filter, H_∞ estimation is still in its infancy. The first state-space solution to the H_∞ control problem was introduced in 1984 [12]. This theory was computationally difficult, and required higher order compensators. Further advancements were made by 1989 when Doyle, Glover et al. achieved a state-space solution that required two Riccati equations [5]. Previous to 1995, the H_∞ estimation problem had only been developed for linear time-invariant systems at steady state [1]. Mangoubi, however, was able to derive filters that are robust to general classes of model uncertainties for both discrete and continuous-time applications [15]. The filters can be developed for time-invariant or time-varying systems over a finite or infinite-time horizon. The generality of the filters in [15] also includes arbitrary initial conditions.

4.0 Noise Model Uncertainty

In the previous chapter, the strengths of the Kalman filter when the noise and plant models are known were highlighted. The weaknesses of the Kalman filter were also shown for improper knowledge of the plant model. In this chapter, the plant is assumed to be known perfectly, but the noise model is uncertain. This allows for the derivation of an estimator that is designed to minimize the error when worst-case noise is present. Other than the noise having a bound on its 2-norm, there are no assumptions in the derivation.

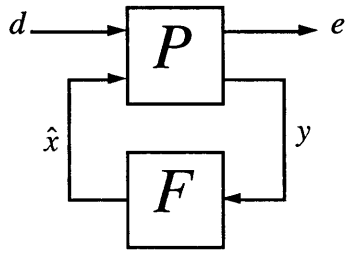

Figure 4.1: Block diagram of estimation problem

4.1 Derivation of minimax estimator

As in Chapter 3, the given system is defined in Figure 4.1 and Equation (4.1-1).

$$\begin{bmatrix} x_{k+1} \\ \hline e_k \\ y_k \end{bmatrix} = \left[\begin{array}{c|cc} A_k & G_k & 0 \\ \hline M_k & 0 & -M_k \\ C_k & E_k & 0 \end{array}\right] \begin{bmatrix} x_k \\ \hline d_k \\ \hat{x}_k \end{bmatrix} \tag{4.1-1}$$

Although the equations and results for this chapter include a control input, u, this equation can be easily modified to include u as well. The new vectors for including a control signal are in Eq.(4.1-2).

$$r_k = \begin{bmatrix} u_k \\ d_k \end{bmatrix}$$
$$y_k' = \begin{bmatrix} y_k \\ u_k \end{bmatrix} \tag{4.1-2}$$

This translates into the altered system matrices of

$$B_k' \leftarrow \begin{bmatrix} B_k & G_k \end{bmatrix}$$
$$C_k' \leftarrow \begin{bmatrix} C_k \\ 0 \end{bmatrix} \tag{4.1-3}$$
$$D_k' \leftarrow \begin{bmatrix} D_k & E_k \\ I & 0 \end{bmatrix}$$

In general, an H_∞ estimator seeks to find an estimate, $\hat{x}_k$, such that

$$\sup_{d \neq 0} \frac{\|e\|_2}{\|d\|_2} < \gamma \tag{4.1-4}$$

where

$$e_k = M_k\left(x_k - \hat{x}_k\right) \tag{4.1-5}$$

and d_k is the disturbance input vector. This is equivalent to

$$\frac{\|e\|_2^2}{\|d\|_2^2} < \gamma^2 \quad \forall\, d \neq 0 \tag{4.1-6}$$

A full derivation of the minimax H_∞ estimator that also includes the initial conditions in the norm constraints is in [15]. A cursory explanation of the solution to the problem is given here. By rearranging Eq. (4.1-6) and defining the performance criterion, J, as

$$J = \|e\|_2^2 - \gamma^2 \|d\|_2^2 < 0 \quad \forall\, d \neq 0 \tag{4.1-7}$$

a game-theoretic formulation of the problem can be defined as

$$\min_{\hat{x}} \max_{d} \; J \tag{4.1-8}$$

subject to the dynamic constraints in Eq. (4.1-1). Furthermore, it is shown in [15] that if the Riccati equation

$$\begin{aligned} P_{k+1} &= \left(A_k - K_k C_k\right) H_k^{-1} \left(A_k - K_k C_k\right)^T + \left(G_k - K_k E_k\right)\left(G_k - K_k E_k\right)^T \\ &= \tilde{A}_k H_k^{-1} \tilde{A}_k^T + \tilde{G}_k \tilde{G}_k^T \end{aligned} \tag{4.1-9}$$

where

$$H_k = P_k^{-1} - \frac{1}{\gamma^2} M_k^T M_k$$
$$K_k = \left[G_k E_k^T + A_k H_k^{-1} C_k^T\right]\left[E_k E_k^T + C_k H_k^{-1} C_k^T\right]^{-1} \tag{4.1-10}$$

has a solution such that H_k is positive definite, then an estimator exists that satisfies

$$J^* < 0 \tag{4.1-11}$$

which means that the desired bound of Eq. (4.1-4) is satisfied. This estimator follows the form

$$\hat{x}_{k+1} = (A_k - K_k C_k)\hat{x}_k + K_k y_k \tag{4.1-12}$$

This has the same form as the one-step Kalman filter in Eq. (3.1-19). This leads to the error dynamics

$$\begin{aligned} \tilde{x}_{k+1} &= (A_k - K_k C_k)\tilde{x}_k + (G_k - K_k E_k)d_k \\ &= \tilde{A}_k \tilde{x}_k + \tilde{G}_k d_k \\ e_k &= M_k \tilde{x}_k \end{aligned} \tag{4.1-13}$$

which, again, have the same form as the one-step Kalman filter in Eq. (3.1-19). By earlier definitions of G_k and E_k, the noise is uncorrelated (i.e., $G_k E_k^T = 0$), simplifying the gain in Eq. (4.1-10) to the form in Eq. (3.1-11). The theory is not constrained to only uncorrelated noise, though. The above filter gives the *a priori* estimate; the *a posteriori* estimator can be found in [15]. This H_∞ solution also follows the property of

$$\|\tilde{G}\|_\infty < \gamma \tag{4.1-14}$$

By decreasing γ in Eq. (4.1-10) to as small a value as possible before H_k violates the positive semi-definite bound, the H_∞ minimax estimator is obtained. This estimator has the smallest possible H_∞ norm, but at an increased H_2 cost (indeed, as γ approaches the minimum γ, the H_2 cost continues to increase).

On the other hand, as $\gamma \to \infty$ the second term in Equation (4.1-10) approaches zero. When this happens $H_k^{-1} \to P_k$, which results in the solution to the Riccati equation (4.1-9) being the true error covariance matrix and the gain in Equation (4.1-10) approaching the optimal gain. In other words, the H_∞ estimator with γ at infinity becomes the Kalman filter. This allows for γ to be thought of as a design parameter. By selecting γ_{min}, the H_∞ norm is minimized (but at a great H_2 cost), while letting $\gamma = \infty$ optimizes the H_2 norm but doesn't allow for any disturbance model error.

There is another design parameter besides γ, though. While

$$e_k = M_k \tilde{x}_k \tag{4.1-15}$$

for both the Kalman filter and the H_∞ estimator, M does not affect the Kalman filter gain matrix while it does for the H_∞ estimator via Eq. (4.1-10). This allows the design engineer to select an M that emphasizes certain states of interest, although this emphasis does come at a cost in terms of the estimation error of the other states.

4.2 Application of minimax estimator

The example used in this section is for the SSME at the 90% power level model with the observation matrix, M, set to the identity. In order to keep B and G on the same order of magnitude (and thus avoid numerical problems) the input signal was multiplied by 1000. Since the system is linear, by subsequently dividing the output signal by 1000 a true relationship is found. The γ_{min} for the system is 2.0310×10^{-3}. The singular values for the transfer function $\boldsymbol{G}$ can be seen in Figure 4.2. The plot clearly shows that as γ increases the minimax estimator approaches the Kalman filter. By $100\gamma_{min}$ in this case the Kalman filter solution has been reached. Also clear in the graph is that the H_∞-norm of the minimax estimator is lower than the Kalman filter's H_∞-norm, but the H_2-norm of the Kalman filter is much lower than the minimax estimator (the minimax H_2-norm approaches infinity since the logarithmic scale gives the illusion of understating the area under the curve at high frequencies). True to the minimax theory, the H_∞ optimal solution gives

$$\begin{aligned} \|\boldsymbol{G}\|_\infty &= 2.03096 \times 10^{-3} < 2.0310 \times 10^{-3} \\ &= -53.85 \text{ dB} \end{aligned} \tag{4.2-1}$$

For this example, the graph of Figure 4.2 suggests that a possible compromise between the H_2 and the H_∞ lines is to choose $\gamma = 1.1\gamma_{min}$. As the plot shows, the H_∞ performance provides some additional protection for unknown disturbances compared to the Kalman filter while the H_2 performance is fairly close to that of the Kalman filter. The effectiveness of the observers becomes apparent when the system is subjected to some noise.

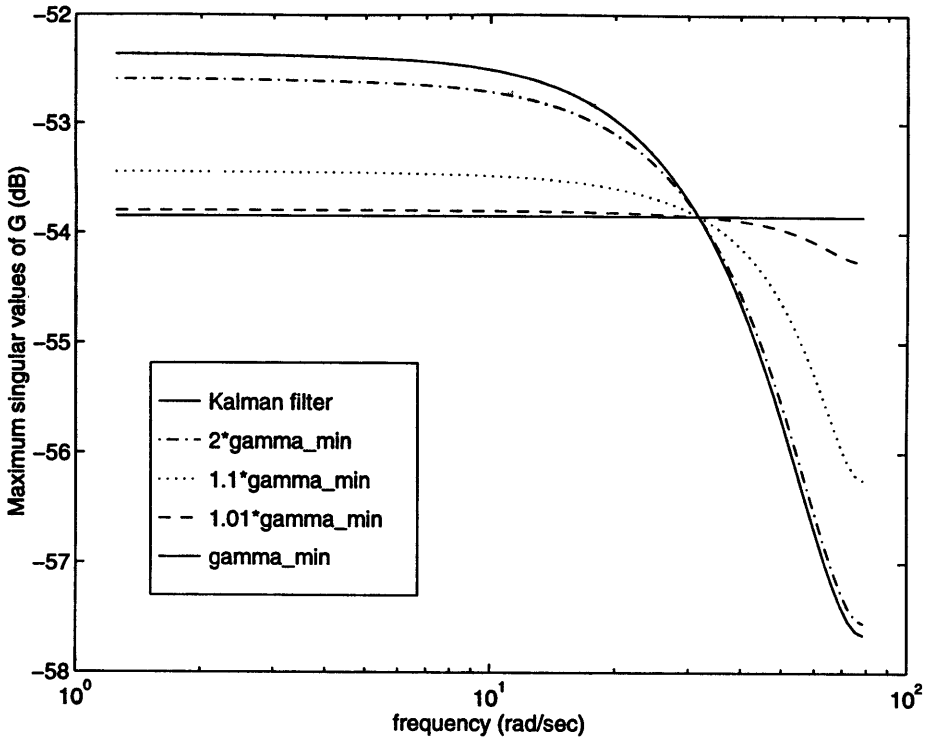

Figure 4.2: Singular values of G with increasing γ

4.2.1 White noise

The first simulation of the H_∞ minimax estimator involved giving the system white noise. The squared error for the 5 second run is summarized in Table 4.1.

	State	Squared error
Kalman filter	Chamber Pressure	0.0518×10^{-3}
	Mixture Ratio	0.2277×10^{-3}
H_∞ estimator for γ_{min}	Chamber Pressure	0.0821×10^{-3}
	Mixture Ratio	0.2900×10^{-3}

Table 4.1: Squared estimation error given white noise

As expected, the Kalman filter has less squared error than the minimax estimator. Since the noise input is white and the models are known perfectly, the Kalman filter is the optimal estimator in this situation. The minimax estimator is not designed for a white noise input, however. It is designed to protect against the worst possible noise at each time step. The value of this protection is readily apparent in the next section.

4.2.2 Worst-case noise

When the noise is not pure white noise the results are dramatically different. When the system is given some noise input that is in the worst-case direction, the minimax estimator is clearly superior. It is shown in [15] that the worst-case noise can be defined as

$$d_k^* = \tilde{B}_k^T P_{k+1}^{-1} \tilde{x}_{k+1} \tag{4.2-2}$$

The noise depends on future information, which is clearly not yet available. Instead it was assumed that

$$d_k = \tilde{B}^T P^{-1} \tilde{x}'_{k+1} \tag{4.2-3}$$

where

$$\tilde{x}'_{k+1} = \tilde{A}\tilde{x}_k + \tilde{B}d_{k-1} \tag{4.2-4}$$

would sufficiently simulate the disturbance in Eq. (4.2-2). While this noise did approximate the worst-case noise sufficiently to show that the minimax was better able to handle the disturbance, the true worst-case noise would have produced even larger errors for both estimators. The approximation of the worst-case noise when given an initial error (i.e., $\tilde{x}_0 \neq 0$) can be seen in Figure 4.3.

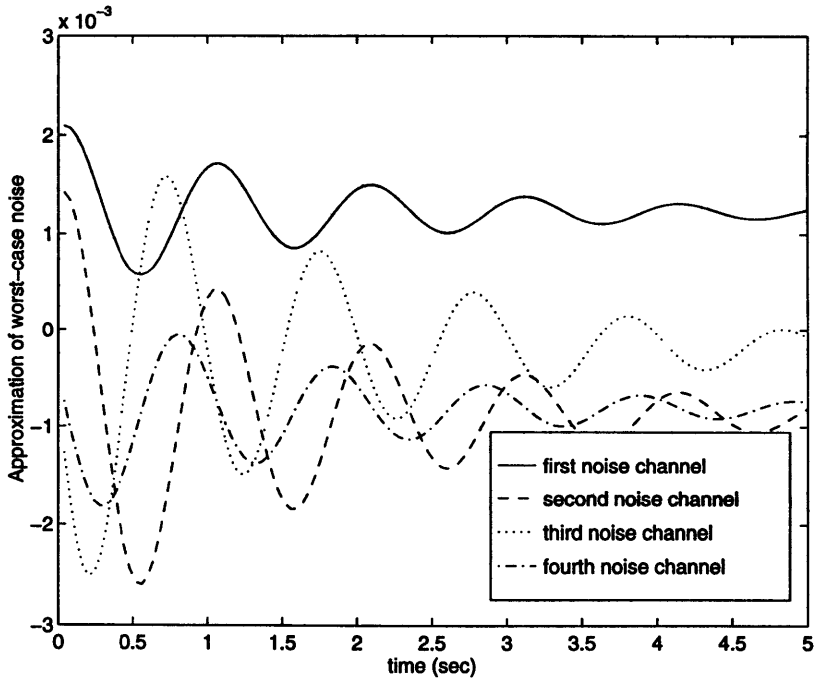

Figure 4.3: Simulated worst-case noise input

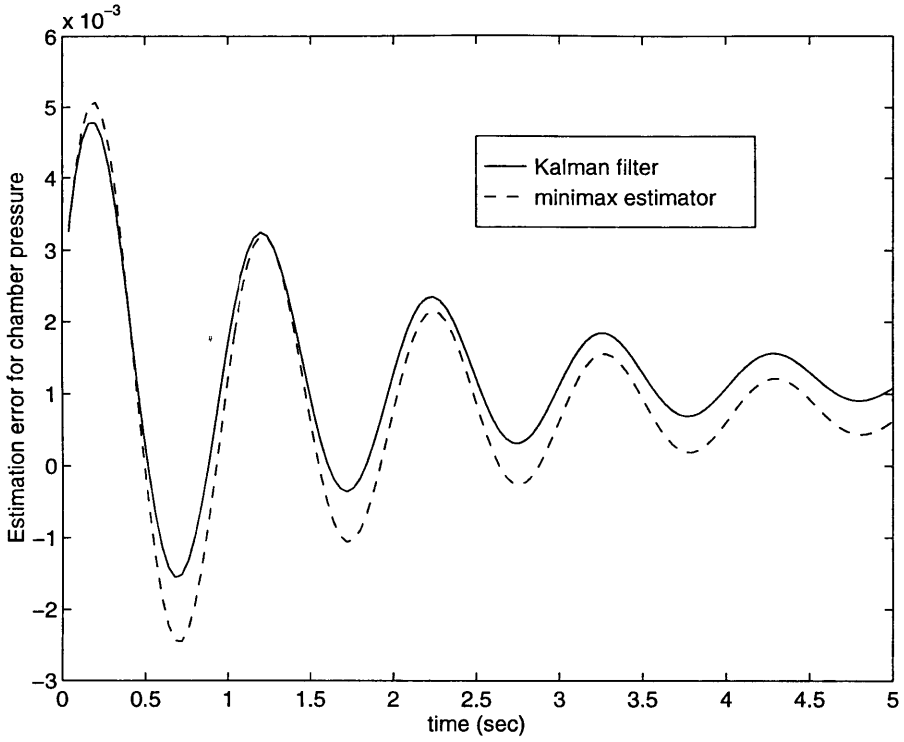

Figure 4.4: Effects of noise in worst case direction on chamber pressure

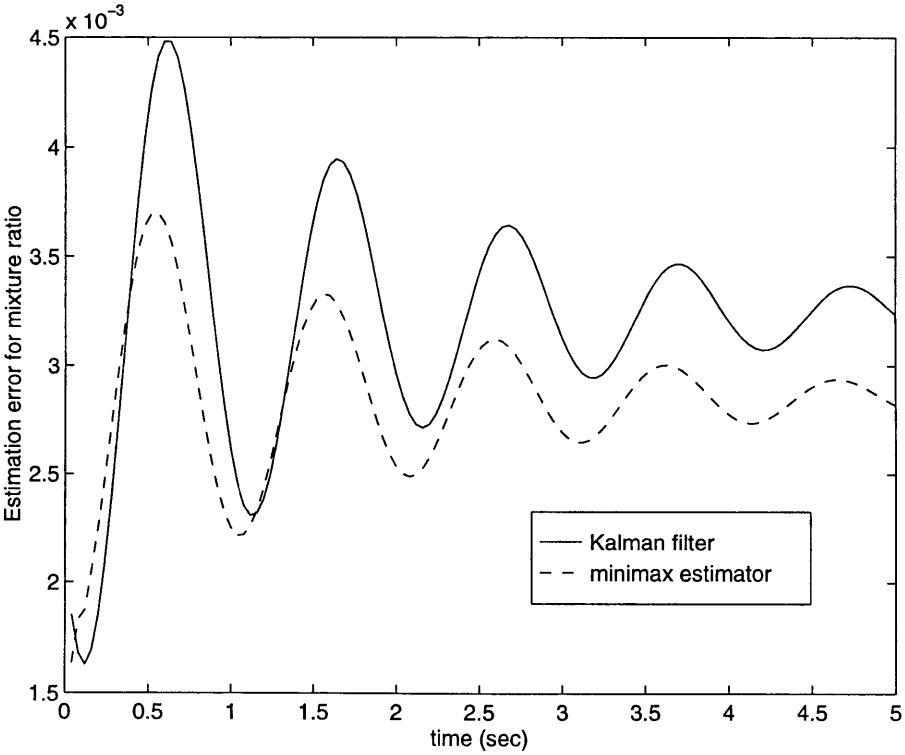

Figure 4.5: Effects of noise in worst case direction on mixture ratio

The effects of this noise on the estimation error can be seen in Figures 4.4 and 4.5, and the squared estimation error over the 5 second simulation is summarized in Table 4.2. In both Figure 4.4 and Figure 4.5, the estimators are converging to a biased value. However, the Kalman filter's bias is greater for both states than the bias present in the minimax estimator. Thus, the sum of the estimation error for the Kalman filter will approach infinity much faster than for the minimax filter.

	State	Squared error
Kalman filter	Chamber Pressure	0.3967×10^{-3}
	Mixture Ratio	1.3257×10^{-3}
Minimax estimator	Chamber Pressure	0.3699×10^{-3}
	Mixture Ratio	1.0260×10^{-3}

Table 4.2: Squared estimation error given "worst-case" noise

While the minimax estimator is clearly superior for the worst-case noise, this noise's dependence on the future renders it physically impossible. However, systems may approach this noise, and the minimax estimator would demonstrate its value.

4.2.3 Sinusoidal input

In this section a new noise was created based on a sinusoidal input with a frequency of 10 rad/sec. Each channel's phase was allowed to vary by up to $\pi/2$ from each other. This is a more realistic noise since it does not depend on future events. Instead, it can represent real noise in the system dynamics (for example, the FPOV and OPOV oscillating at 10 rad/sec). The results can be seen in Figures 4.6 and 4.7, and the squared error over the 5 second simulation is shown in Table 4.3. Even though the minimax is specifically designed for the worst-case noise, the results clearly show that its estimation error for both states is less than the Kalman filter. The Kalman filter, designed to minimize error when the system is given white noise, cannot handle the colored noise without additional modeling (such as a shaping filter) as well as the minimax estimator.

	State	Squared error
Kalman filter	Chamber Pressure	0.0207×10^{-3}
	Mixture Ratio	0.3696×10^{-3}
Minimax estimator	Chamber Pressure	0.0177×10^{-3}
	Mixture Ratio	0.2606×10^{-3}

Table 4.3: Squared estimation error given 10 rad/sec noise input

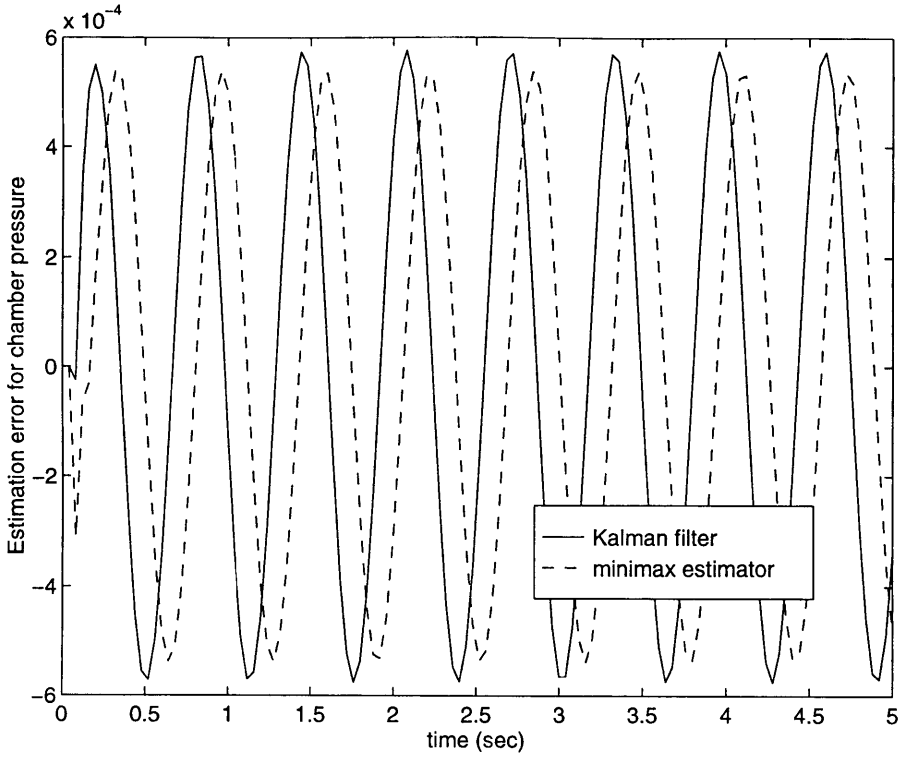

Figure 4.6: Effects of sinusoidal noise on chamber pressure

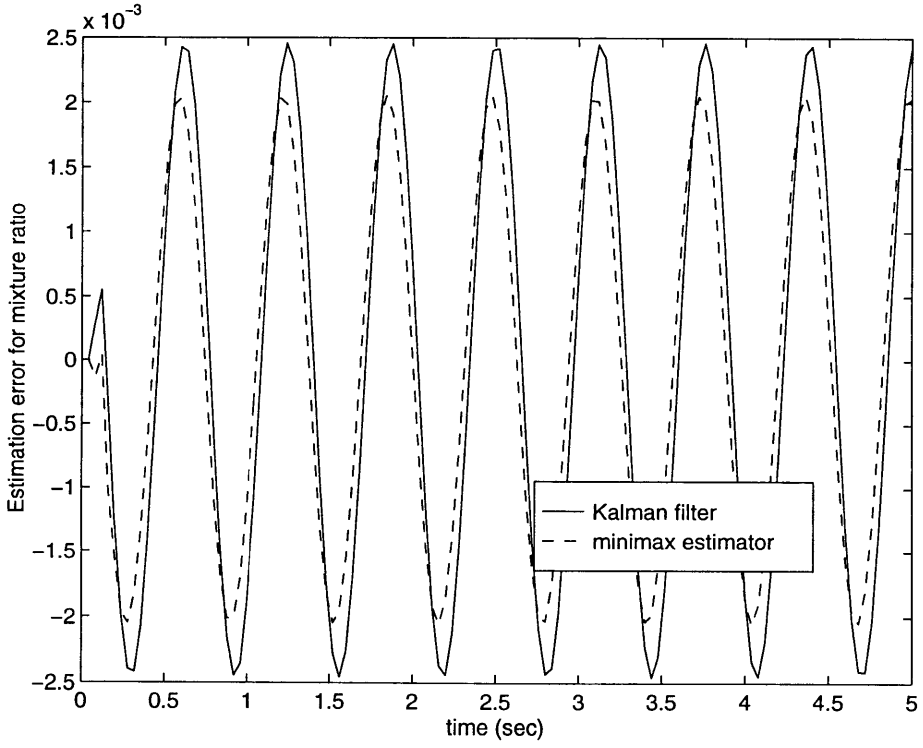

Figure 4.7: Effects of sinusoidal noise on mixture ratio

4.3 Summary

When the noise model is known well the Kalman filter is the optimal estimator in the least-squares error sense. However, noise models are not always known completely. An estimator that minimizes the ∞-norm of the transfer function from the worst-case disturbance to the error can be derived, and when this worst-case noise is given to this estimator and the Kalman filter, the H_∞ minimax estimator is superior. The disturbance does not need to be in the worst-case direction for the minimax estimator to show its worth, however. Using the minimax estimator on an unexpected sinusoidal input produces less estimation error than the Kalman filter designed for the nominal white noise. There are trade-offs associated with using the H_∞ minimax estimator, though. It's H_2 cost dramatically increases compared to the Kalman filter. Thus, a compromise between H_2 and H_∞ performance can be found by varying γ.

5.0 Plant Model Uncertainty

In the previous chapter, noise model uncertainty was examined. An estimator was designed that is able to handle the noise in the worst-case direction better than the Kalman filter is. While robustness to noise is important, robustness to plant model errors is even more important. Very rarely are physical plant models known completely, rarer still are they linear; even if the knowledge is complete, often the dynamics are truncated due to the model's large size. In this chapter, the uncertainty shall be expanded from just the noise model to the plant model as well. Figure 5.1 shows the new form with the Δ-block now encasing all of the uncertainty.

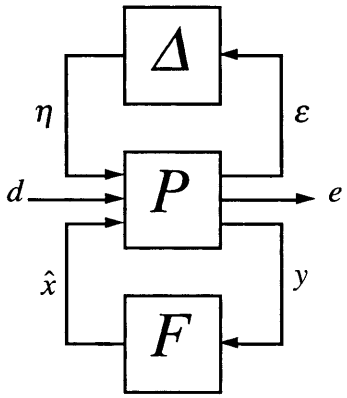

Figure 5.1: Estimation block diagram with uncertainty

This open-loop system corresponds to the state-space model

$$\begin{bmatrix} x_{k+1} \\ \varepsilon_k \\ e_k \\ y_k \end{bmatrix} = \left[\begin{array}{c|ccc} A_k & Q_k & G_k & 0 \\ \hline S_k & 0 & L_k & 0 \\ M_k & 0 & 0 & -M_k \\ C_k & R_k & E_k & 0 \end{array}\right] \begin{bmatrix} x_k \\ \eta_k \\ d_k \\ \hat{x}_k \end{bmatrix} \tag{5.0-1}$$

As in Chapter 4, the system can be modified to include a control input, u_k, by following the format found in Eqs. (4.1-2) and (4.1-3). Section 5.1 derives the robust filter, followed by application of this theory to systems with only noise input (Section 5.2) and systems with both noise and control inputs (Section 5.3).

5.1 Derivation of robust filter

Once again, the complete derivation of the robust filter is found in [15]. The robust estimator seeks to bound the H_∞-norm of the transfer function, $\boldsymbol{G}$, from the disturbance, d, to the estimation error, e. This is to take place for all possible model perturbations, Δ, that have a bounded induced 2-norm. Based on this, a performance criterion, J_1, can be defined.

$$\begin{aligned} \sup_{(d,\eta)\neq 0} J_1 &\equiv \frac{\|e\|_2^2}{\|d\|_2^2} < \gamma^2 \\ \forall\, \Delta \ni \|\Delta\|_2^2 &\equiv \sup_{\varepsilon\neq 0} \frac{\|\eta\|_2^2}{\|\varepsilon\|_2^2} < \frac{1}{\gamma^2} \end{aligned} \tag{5.1-1}$$

Now the approach changes slightly from Chapter 4. The output from the perturbation, η, is treated as an additional input signal, and is grouped together with the disturbance, d. The combined exogenous signal is

$$d_k' = \begin{bmatrix} \eta_k \\ d_k \end{bmatrix} \tag{5.1-2}$$

Likewise, the input to the perturbation, ε, is treated as an additional error signal. This allows for the merger of the system matrices according to

$$\begin{aligned} B_k &= \begin{bmatrix} Q_k & G_k \end{bmatrix} \\ D_k &= \begin{bmatrix} R_k & E_k \end{bmatrix} \\ T_k &= \begin{bmatrix} 0 & L_k \end{bmatrix} \end{aligned} \tag{5.1-3}$$

The performance criterion for this augmented system is defined as

$$\sup_{(d,\eta)\neq 0} J_2 < \gamma^2 \tag{5.1-4}$$

where

$$J_2 \equiv \sup_{(d,\eta)\neq 0} \frac{\|e\|_2^2 + \|\varepsilon\|_2^2}{\|d\|_2^2 + \|\eta\|_2^2} \tag{5.1-5}$$

Eq. (5.1-5) is the condition for robust performance (small gain condition). Although initial conditions are not included in Eq. (5.1-5), the full derivation in [15] does account for them.

Converting Eq. (5.1-5) into a game-theoretic problem and defining a new performance criterion yields

$$\min_{\hat{x}} \max_{d,\eta} J_3 \tag{5.1-6}$$

where

$$J_3 \equiv \|e\|_2^2 + \|\varepsilon\|_2^2 - \gamma^2\left(\|d\|_2^2 + \|\eta\|_2^2\right) \tag{5.1-7}$$

subject to Eq. (5.0-1) and Eq. (5.1-3).

The solution to this robust estimation problem requires two Riccati equations to be solved [15]. Provided the first Riccati equation

$$\begin{aligned} X_k &= A_k^T X_{k+1} A_k + S_k' S_k + \frac{1}{\gamma^2} F_k Z_k^{-1} F_k^{-1} \\ X_N &= 0 \end{aligned} \tag{5.1-8}$$

has a solution such that $X_k > 0$, where

$$\begin{aligned} F_k &\equiv S_k^T T_k + A_k^T X_{k+1} B_k \\ Z_k &\equiv I - \frac{1}{\gamma^2}\left(T_k^T T_k + B_k^T X_{k+1} B_k\right) \end{aligned} \tag{5.1-9}$$

then it is possible to transform Eq. (5.0-1) and Eq. (5.1-3) into the new system

$$\begin{aligned} x_{k+1} &= \overline{A}_k x_k + \overline{B}_k \overline{d}_k \\ y_k &= \overline{C}_k x_k + \overline{D}_k \overline{d}_k \end{aligned} \tag{5.1-10}$$

with

$$\begin{aligned} \overline{A}_k &= A_k + \frac{1}{\gamma^2} B_k Z_k^{-1} F_k^T \\ \overline{B}_k &= B_k Z_k^{-1/2} \\ \overline{C}_k &= C_k + \frac{1}{\gamma^2} D_k Z_k^{-1} F_k^T \\ \overline{D}_k &= D_k Z_k^{-1/2} \end{aligned} \tag{5.1-11}$$

where the new disturbance is defined by

$$d_k^* \equiv \frac{1}{\gamma^2} Z_k^{-1/2} F_k^T x_k$$
$$\bar{d}_k \equiv Z_k^{1/2} d_k - d_k^* \tag{5.1-12}$$

The problem of optimizing is exactly the same as the one treated in the previous chapter. This problem, therefore, has a solution provided the second Riccati equation

$$\bar{P}_{k+1} = \left(\bar{A}_k - \bar{K}_k \bar{C}_k\right) \bar{H}_k^{-1} \left(\bar{A}_k - \bar{K}_k \bar{C}_k\right)^T + \left(\bar{B}_k - \bar{K}_k \bar{D}_k\right)\left(\bar{B}_k - \bar{K}_k \bar{D}_k\right)^T$$
$$\bar{H}_k = \bar{P}_k^{-1} - \frac{1}{\gamma^2} M_k^T M_k \tag{5.1-13}$$

has a solution such that $\bar{H}_k > 0$. Thus, the solution also takes on the familiar form of

$$\hat{x}_{k+1} = \left(\bar{A}_k - \bar{K}_k \bar{C}_k\right) \hat{x}_k + \bar{K}_k y_k \tag{5.1-14}$$

with the gain

$$\bar{K}_k^* = \left[\bar{B}_k \bar{D}_k^T + \bar{A}_k \bar{H}_k^{-1} \bar{C}_k^T\right]\left[\bar{D}_k \bar{D}_k^T + \bar{C}_k \bar{H}_k^{-1} \bar{C}_k^T\right]^{-1} \tag{5.1-15}$$

The γ in Eq. (5.1-8) and the γ in (5.1-13) are the same. While this initially appears to be a limitation, the second term in the equation for $\bar{H}_k$ can still be altered by varying M_k during the design of the estimator. After the estimator design is complete, a new M_k can be chosen for the analytical comparison between the Kalman filter and the robust estimator.

5.2 Application without a control signal variation

This section is deals with the case with no change in the control input, *u*--instead the disturbance, *d*, is the only true exogenous signal. Also, once again, the SSME models are time-invariant, thus the subscript *k*'s may be dropped from all of the matrices in the above derivation.

In order to use the robust estimator, it is necessary to develop some model of the uncertainty for the estimator to be designed around. In this section all of the uncertainty is modeled parametrically. A description of how to achieve the desired matrices needed to complete Eq. (5.0-1) through parametric uncertainty is given in Appendix A. The parametric uncertainty for the SSME models could be achieved many different ways, but in this analysis two primary modeling methods are used. One was to take the difference between two models and use that as the uncertainty for each matrix; for example, $\Delta A = A_{pert} - A_{nom}$, where the nominal system is the 110% power level model and the perturbed

system is, say, the 90% power level model. The other method is to add some fixed percentage of the matrix back to the nominal case to achieve the perturbed case, such as $A_{pert} = 1.10A_{nom}$. This creates a ΔA of 10% of A_{nom}. Sensor uncertainty is always modeled as either $\Delta E = E_{bad} - E_{good}$ or $\Delta E = 0$. Since C has no dynamics imbedded within it (it simply draws the values of the last two states) it was never given any uncertainty. The nominal base system for each case (in both this section and the following section) is the 110% power level model; the sensor quality for the nominal system is allowed to change for the different cases.

The results of this section are given mainly as statistical analysis. One can readily see the differences in filter behavior by examining responses to the white noise input with this analysis. While time simulation data would offer the same conclusions, the filter differences are more difficult to observe from time history plots. In order to compare the robust filter and the Kalman filter the steady-state error covariances from the robust filter and the perturbed Kalman filter must be calculated. This is accomplished using Lyapunov's equation on the augmented dynamics. By creating the augmented system of states and estimates

$$\begin{aligned}\begin{bmatrix} x_{k+1} \\ \hat{x}_{k+1} \end{bmatrix} &= \begin{bmatrix} A & 0 \\ \overline{K}C & \overline{A} - \overline{K}\overline{C} \end{bmatrix} \begin{bmatrix} x_k \\ \hat{x}_k \end{bmatrix} + \begin{bmatrix} B \\ \overline{K}\overline{D} \end{bmatrix} d_k \\ &= A_{aug} x_{k_{aug}} + B_{aug} d_{k\,aug}\end{aligned} \tag{5.2-1}$$

the estimator error can now be defined as

$$\begin{aligned} e_k &= \begin{bmatrix} M & -M \end{bmatrix} \begin{bmatrix} x_k \\ \hat{x}_k \end{bmatrix} \\ &= C_{aug} x_{k_{aug}} \end{aligned} \tag{5.2-2}$$

By using these augmented matrices in Lyapunov's equation the true estimation error covariance matrix is

$$\begin{aligned} Y &= A_{aug} Y A_{aug}^T + B_{aug} B_{aug}^T \\ P_{\tilde{x}} &= C_{aug} Y C_{aug}^T \end{aligned} \tag{5.2-3}$$

For each example in this section the data is organized into tables comparing the Kalman filter and the robust filter.

5.2.1 Uncertainty in plant model

The first application of the robust theory is made only to uncertainties in the A and G matrices (corresponding to uncertainty in the plant dynamics and system input models) with a good sensor. As in Chapter 4, this disturbance signal is premultiplied by 1000 and the outputs are divided by 1000 to keep the matrices' order of magnitude equivalent. The design uncertainty bound for the A and G matrices is 15% (i.e., $\Delta A = .15A_{nom}$). Since γ is set to 1 x 10^6, the choice for M needed to affect Eq. (5.1-13) is also very large. In this case, improved results were obtained by setting

$$M = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & & 2 \end{bmatrix} \times \left(3 \times 10^8\right) \tag{5.2-4}$$

After obtaining the filter gain, $\overline{K}$, the dynamics of the estimator are fully determined. Changing M to the identity matrix for analysis does not affect the estimator's performance, but it does allow more meaningful comparisons to be made between the Kalman filter and the robust filter, as well as comparisons of performance between different robust estimators. Thus, M is set to I for analysis.

	State	Kalman Filter	Robust Filter	Optimal Filter
70% Power Level	Chamber Pressure	0.0816 x 10^{-5}	0.0751 x 10^{-5}	0.0723 x 10^{-5}
	Mixture Ratio	0.2180 x 10^{-5}	0.2068 x 10^{-5}	0.2036 x 10^{-5}
80% Power Level	Chamber Pressure	0.0755 x 10^{-5}	0.0709 x 10^{-5}	0.0691 x 10^{-5}
	Mixture Ratio	0.2431 x 10^{-5}	0.2297 x 10^{-5}	0.2252 x 10^{-5}
90% Power Level	Chamber Pressure	0.0681 x 10^{-5}	0.0640 x 10^{-5}	0.0633 x 10^{-5}
	Mixture Ratio	0.2321 x 10^{-5}	0.2211 x 10^{-5}	0.2186 x 10^{-5}
100% Power Level	Chamber Pressure	0.0376 x 10^{-5}	0.0375 x 10^{-5}	0.0370 x 10^{-5}
	Mixture Ratio	0.1491 x 10^{-5}	0.1459 x 10^{-5}	0.1454 x 10^{-5}
110% Power Level	Chamber Pressure	0.0225 x 10^{-5}	0.0242 x 10^{-5}	0.0225 x 10^{-5}
	Mixture Ratio	0.0836 x 10^{-5}	0.0874 x 10^{-5}	0.0836 x 10^{-5}

Table 5.1: Variances of error for perturbed plant model

In Table 5.1, the variances from the diagonal of the error covariance matrix, $P_{\tilde{x}}$, for the different estimators are displayed. The column entitled "Kalman Filter" gives results for the case where the Kalman filter designed for the nominal case (110% power level model) is used at all power levels. The robust filter column is the robust H_∞ filter

designed for the nominal case with some uncertainty bound and then applied to the perturbed systems. In the last column, the optimal filter is defined as the Kalman filter for the *perturbed* system. This optimal filter represents the theoretical limit of performance, that is, the least-squares error, for the perturbed system--no filter will be able to outperform it.

Using the standard deviations from these variances, calculated by $\sqrt{diag(P_{\tilde{x}})}$, the following percent difference table was assembled by calculating the percent difference of the standard deviation of the errors according to

$$\%\ difference = \left(\frac{\sigma_{model} - \sigma_{model}^{opt}}{\sigma_{model}^{opt}} \right) \times 100 \qquad (5.2\text{-}5)$$

where *opt* denotes the optimal filter defined above and *model* denotes the particular power level of the perturbed system. All percent difference tables in this chapter were calculated using the same methodology.

	State	Kalman Filter	Robust Filter
70% Power Level	Chamber Pressure	6.2972	1.9759
	Mixture Ratio	3.4795	0.7765
80% Power Level	Chamber Pressure	4.5297	1.2656
	Mixture Ratio	3.9119	1.0067
90% Power Level	Chamber Pressure	3.6626	0.5291
	Mixture Ratio	3.0582	0.5797
100% Power Level	Chamber Pressure	0.8399	0.6551
	Mixture Ratio	1.2631	0.1620
110% Power Level	Chamber Pressure	0	3.6953
	Mixture Ratio	0	2.2269

Table 5.2: Percent difference from optimal of standard deviation of error for perturbed plant model

Table 5.2 shows the results for two filters. Since the nominal system is 110% power level, the Kalman filter at 110% is the optimal Kalman filter (it has perfect model knowledge for the plant and the noise). Therefore, its 0% error is expected. The robust filter sacrifices 2% and 3% for the two states at 110% though, from the uncertainty in the filter design. It more than makes up for that small error at other power levels, though. At each power level other than the 110% model the Kalman filter's performance is worse than the robust H_∞ filter. The results are most apparent when larger perturbations appear in A

and G, such as at the 70% power level. There the robust filter estimation error is less than a third of the size of the Kalman filter's error.

5.2.2 Uncertainty in plant and sensor error models

Now, in addition to uncertainty in A and G, some uncertainty is added to the sensor error model, changing E. The sensor uncertainty is designed around the bad sensor, and the plant model uncertainty is 5% off of the nominal 110% power level model. The Kalman filter is designed for the good sensor. As in the previous sub-section, γremains at 1×10^6, and robust performance improvs by changing M for the filter design. In this case the design M is

$$M = I \times \left(2 \times 10^8\right) \tag{5.2-6}$$

Similar to the last section, M is again reset to identity for the analysis.

	State	Kalman Filter	Robust Filter
70% Power Level	Chamber Pressure	1.7802	0.3398
	Mixture Ratio	0.8499	0.1400
80% Power Level	Chamber Pressure	2.4987	0.5544
	Mixture Ratio	0.6185	0.1124
90% Power Level	Chamber Pressure	2.5985	0.5544
	Mixture Ratio	0.7662	0.1007
100% Power Level	Chamber Pressure	8.0531	3.2068
	Mixture Ratio	2.5423	0.7374
110% Power Level	Chamber Pressure	17.5125	8.4851
	Mixture Ratio	7.8384	3.3665

Table 5.3: Percent difference from optimal of standard deviation of error for perturbed plant and sensor models

The first simulation was performed with the bad sensor model. Table 5.3 shows a dramatic degradation at the 110% power level model for the Kalman filter with a perturbed sensor error model. The standard deviation of error of the chamber pressure estimate is now more than 17% larger than the optimal Kalman filter's standard deviation of error for the perturbed system. The robust filter shows the value of adding the uncertainty--not only is it dramatically better at 110%, but at each power level it remains better than the Kalman filter designed for the nominal system. By presenting the above data in the form of the percent difference table an interesting trend emerges. The Kalman filter error appears to actually decrease as the plant gets farther away from the nominal system. While this is true

for the percent difference, the absolute variance data shows that the Kalman filter does indeed get worse as the plant progresses beyond the nominal plant. However, the variances of the optimal filter for the off-nominal power levels increase dramatically. Although it is a different case, this trend can be seen in the last column in Table 5.1. Thus, the percent difference for the nominal system is greater since the variances of the optimal system are much smaller, making any difference comparatively much larger.

This improved performance comes at a cost, however. The next table (Table 5.4) shows the performance of the same filters, but now the perturbed systems have a *good* sensor instead of the bad sensor for Table 5.3. Because the system now has only a perturbed A and G with a good sensor, the Kalman filter results are exactly the same as those of Table 5.2. The robust filter, on the other hand, is still designed for uncertainty in the sensor error model. Thus, its performance degrades from the results listed in Table 5.3.

	State	Kalman Filter	Robust Filter
70% Power Level	Chamber Pressure	6.2972	8.0394
	Mixture Ratio	3.4795	4.7211
80% Power Level	Chamber Pressure	4.5297	5.9325
	Mixture Ratio	3.9119	5.1757
90% Power Level	Chamber Pressure	3.6626	5.3030
	Mixture Ratio	3.0582	4.2236
100% Power Level	Chamber Pressure	0.8399	2.1460
	Mixture Ratio	1.2631	2.1941
110% Power Level	Chamber Pressure	0	0.5035
	Mixture Ratio	0	0.2098

Table 5.4: Percent difference from optimal of standard deviation of error for perturbed plant dynamics (robust filter designed for perturbed plant and sensor error models)

5.2.3 Uncertainty in sensor error model

The final example in this section is for perturbations in the sensor error model alone. The plant model is unchanged from the nominal 110% power level--both filters are designed using an accurate plant model. Perfect sensors are impossible, and often precise knowledge of the sensor model is very difficult to obtain. Thus, the addition of robustness to sensor error is certain to be a useful enhancement in an estimator's design. For this first

simulation, the Kalman filter is designed for the nominal case with the good sensor, and the robust estimator is designed with the uncertainty of the bad sensor error model taken into account. The error weighting matrix, M, is defined as the identity for both the design and analysis in this example.

The Kalman filter results are identical to the equivalent models in Table 5.3 and Table 5.4. In contrast to the models used to generate the data in Tables 5.3 and 5.4, the robust estimator in this case did not take any model uncertainty into account and the estimation error weighting matrix, M, also is different. Hence, the robust estimator's results have changed from those examples. For the perturbed case, the robust filter has gained an additional 3% for chamber pressure error from the Kalman filter. The error when the sensor model is correct is still less than 1% (although it is now slightly worse than the case for Table 5.4). This allows significant room for the robust filter to be used as the estimator for a system where the noise model is uncertain.

110% Power Level	Kalman Filter	Robust Filter
Chamber Pressure	17.5125	5.6375
Mixture Ratio	7.8384	2.2884

Table 5.5: Percent difference from optimal of standard deviation of error for bad sensor

110% Power Level	Kalman Filter	Robust Filter
Chamber Pressure	0	0.8655
Mixture Ratio	0	0.3503

Table 5.6: Percent difference from optimal of standard deviation of error for nominal sensor

These results are already dramatic. An even more dramatic result is seen if the sensor is not only of poor quality, but it has "failed," becoming in effect a white noise generator. A new sensor model was developed for a severely degraded sensor according to

$$E_{degraded} = \begin{bmatrix} 0 & 0 & 0.020 & 0 \\ 0 & 0 & 0 & 0.020 \end{bmatrix} \tag{5.2-3}$$

The sensor now has a variance of 100 times that of the good sensor. A robust filter was designed around the nominal good sensor with the uncertainty of this degraded sensor, while the Kalman filter still only used the nominal good sensor. The results of the simulation with the failed sensor are in Table 5.7.

110% Power Level	Kalman Filter	Robust Filter
Chamber Pressure	211.6694	19.7098
Mixture Ratio	117.2075	7.6359

Table 5.7 Percent difference from optimal of standard deviation of error for degraded sensor

Now the chamber pressure has a standard deviation of over 200% worse than the optimal Kalman filter for the perturbed system. This is compared to the robust filter's error of less than 20%. The results are equally impressive for the mixture ratio. Once again, though, this dramatic improvement in performance must come at some cost. Table 5.8 shows the results when the sensor is still the good sensor. Again, the Kalman filter, having perfect knowledge of the plant and noise models, is the optimal estimator. The robust estimator only sacrifices 4% in chamber pressure from the optimal estimator, though. For an improvement of almost 200% for when the sensor has failed, this cost seems minimal to bear.

110% Power Level	Kalman Filter	Robust Filter
Chamber Pressure	0	3.9835
Mixture Ratio	0	1.5048

Table 5.8 Percent difference from optimal of standard deviation of error for nominal sensor

Quality sensors are expensive. By using robust filters, lower quality sensors can be used with acceptable loss of performance. Also, sensors degrade with time. Estimators that are robust to sensor perturbations allow for the sensors to degrade over time without sacrificing much in the quality of the estimate. This can also be a significant cost savings.

5.3 Application with control signal variation

The preceding section only allowed for disturbances as input signals. In this section, the control input, u, will also be included.

5.3.1 Uncertainty in sensor error model

The first example shown is the same case as in Section 5.2.3. The sensor error models used there are the same as the ones used in this section. Since there is no perturbation in B, the Kalman filter is able to properly account for the control signal. Thus, the data presented in the tables from Section 5.2.3 are unchanged. The effects of the degraded sensor can easily be seen in a time plot, though. Figure 5.2 shows the state

estimates for chamber pressure with the degraded sensor model for both the Kalman filter and the robust estimator. In order to better distinguish the different curves, the plot has been enlarged in the region of 4-6 seconds.

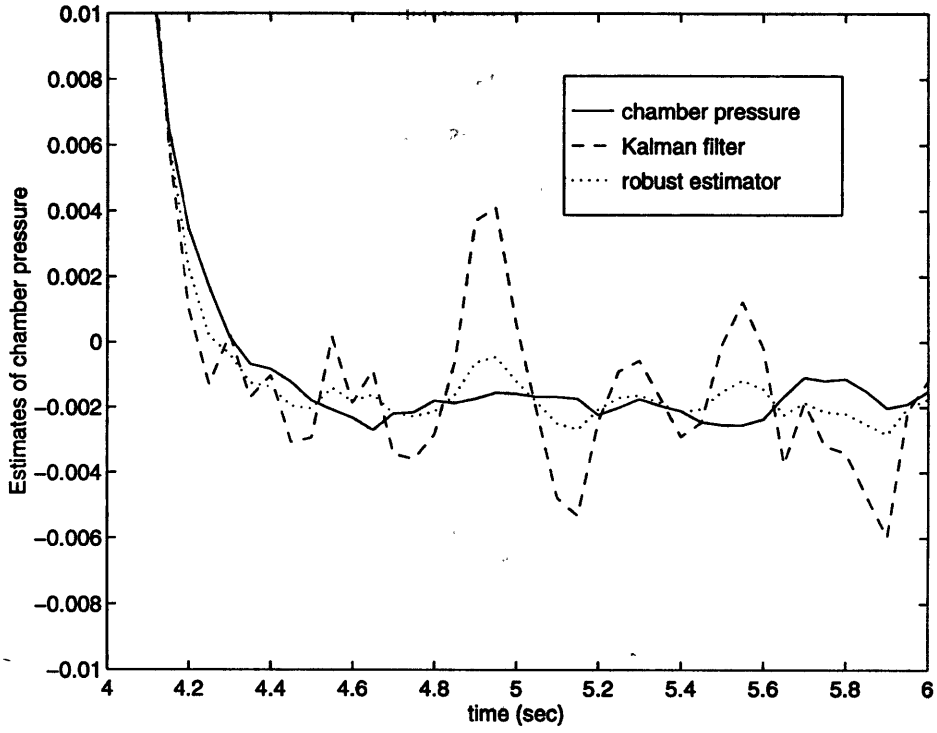

Figure 5.2: Estimators with degraded sensor model

	State	Squared error
Kalman filter	Chamber Pressure	0.6480×10^{-3}
	Mixture Ratio	1.0635×10^{-3}
Robust estimator	Chamber Pressure	0.0820×10^{-3}
	Mixture Ratio	0.2416×10^{-3}

Table 5.9: Squared estimation error for degraded sensor model

The squared estimation error for the entire 12 second simulation (not just from 4-6 seconds) for the two estimators is compared in Table 5.9. The numbers from the table mirror what is readily apparent in the plot--the robust estimator performs much better when the sensor has degraded than the Kalman filter designed for the nominal good sensor. As usual, this comes at a cost. Figure 5.3 and Table 5.10 show the results for when the sensor is nominal (i.e., it is the good sensor).

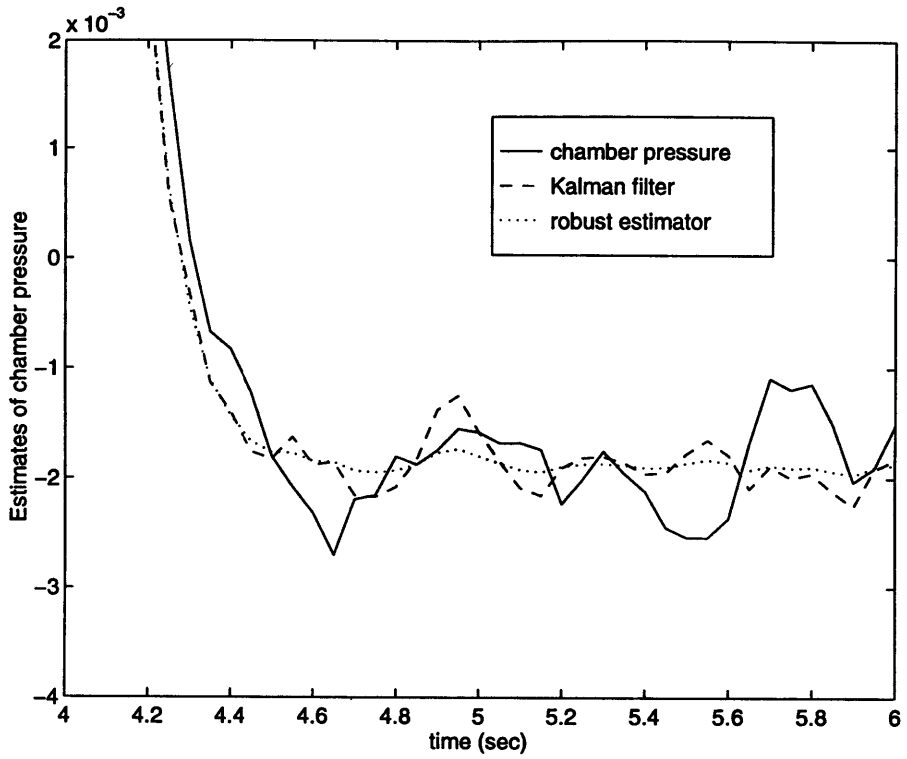

Figure 5.3: Estimators with nominal sensor model

Again, as expected, the Kalman filter performs better when the plant and noise models are known completely. The robust estimator, though, remains extremely close in quality. In fact, one needs to reference Table 5.10 in order to see that the Kalman filter is better--Figure 5.3 shows no clear advantage for either estimator.

	State	Squared error
Kalman filter	Chamber Pressure	0.0483×10^{-3}
	Mixture Ratio	0.2034×10^{-3}
Minimax estimator	Chamber Pressure	0.0498×10^{-3}
	Mixture Ratio	0.2114×10^{-3}

Table 5.10: Squared estimation error for nominal sensor model

In the case of sensor model uncertainty, therefore, it is clearly better to design a robust H_∞ estimator. The costs associated with the robust estimator are minimal, especially when compared to the estimates that are several times better when a sensor degrades.

5.3.2 Uncertainty in plant model

The next challenge is to build an estimator that is not only robust to plant dynamic and process noise model uncertainty (Section 5.2.1), but also robust to uncertainty in the control input. Despite repeated attempts, parametric uncertainty was not entirely successful at meeting this goal. The output shown in Figure 5.4 is from a robust filter designed around the 110% power model as nominal with 250% uncertainty. The uncertainty in the control input matrix, ΔB, was also given an additional 1000%. The perturbed system in this case is the 90% power level model. As in Chapter 3, the Kalman filter never recognizes the perturbed system, and its estimates remain close to the nominal plant states. Unfortunately, even though the robust filter estimates are much better than the Kalman filter, it is unable to completely eliminate the bias. This bias becomes especially pronounced immediately following the system's response to the step functions in Figure 2.3.

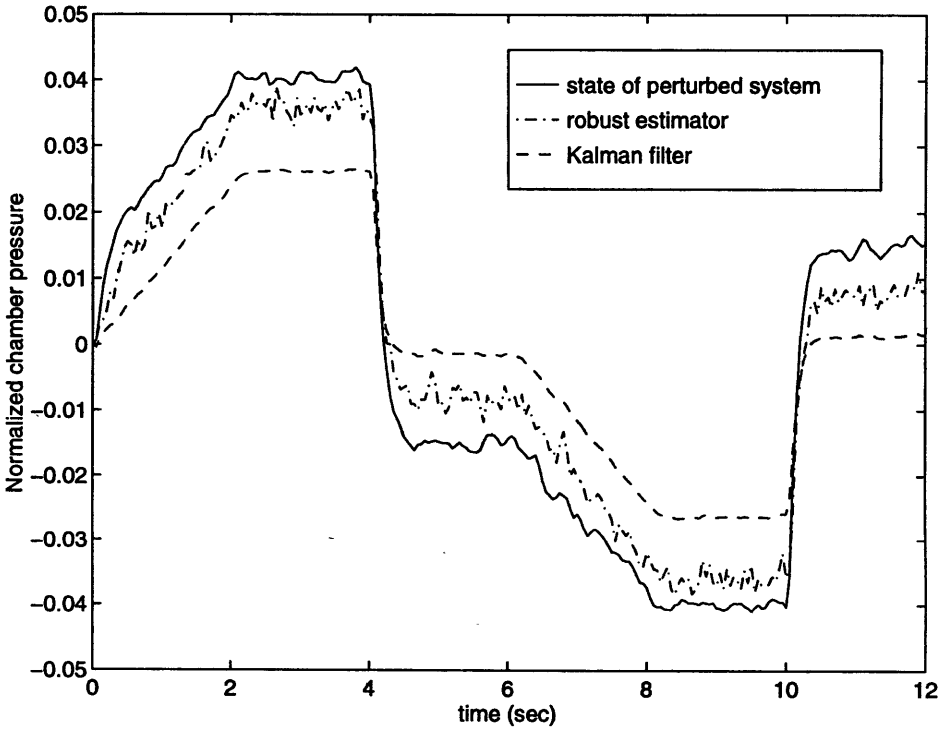

Figure 5.4: Robust estimator with parametric uncertainty

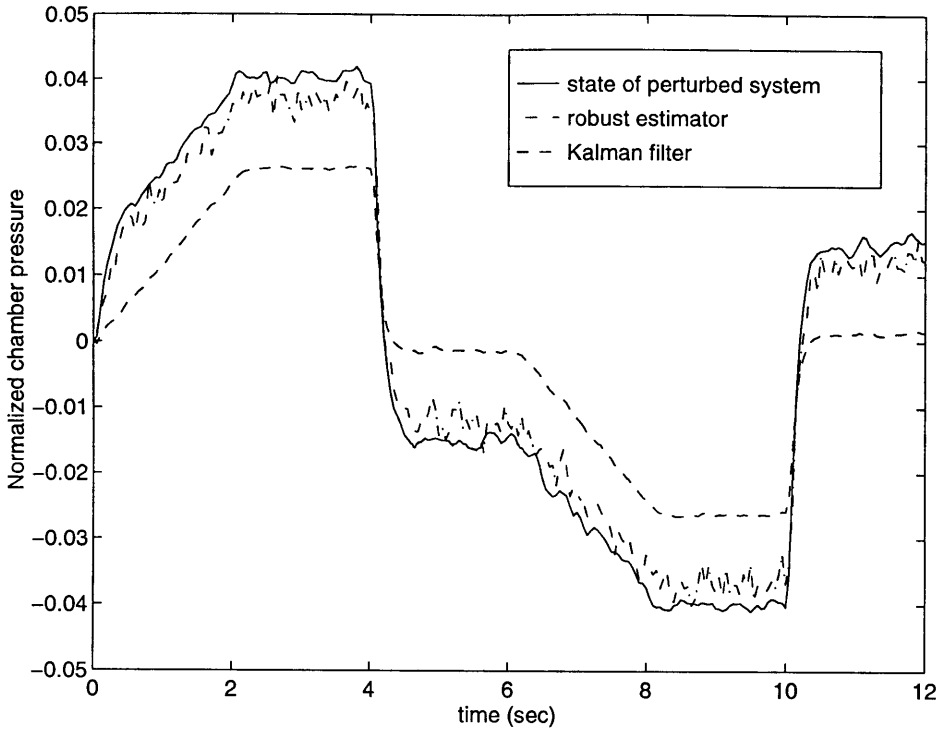

Figure 5.5: Robust estimator with modified plant and parametric uncertainty

In order to improve the performance of the robust estimator, several attempts were made at giving the plant various combinations of additive error and parametric uncertainty. The final method used to generate the data in Figure 5.5 is described in Appendix B. While the bias following the step functions has been reduced, it has not been entirely eliminated.

While the plots show the final results obtained with the different methods, they do not show the many previous attempts. As the understanding of the problem grew, and the subsequent designs of the estimators refined, the results continued to improve. Time did not allow for further development of a robust estimator that could fully eliminate the bias, although it is expected that with additional effort the desired results could be achieved.

5.4 Summary

With the robust H_∞ estimator, design engineers have several options based on their knowledge of the system. While the Kalman filter produces the same filter regardless of the uncertainty associated each system component, the robust filter allows for multiple designs that incorporate the level of uncertainty that the engineer feels is present. The first design presented in Section 5.2.1 allows for uncertainty in the plant model when there is no change in the control signal, but assumes that there is no uncertainty in the sensor model.

When the plant is nominal, the Kalman filter's performance is better, but at each of the other power levels the robust filter is better. Next, the sensor uncertainty was also included. Again, when the models are fully perturbed, the robust filter is superior. When the sensor is in fact nominal (instead of perturbed), though, the Kalman filter is slightly better. This allows the engineer to design a filter that allows for some sensor degradation to occur. Only sensor uncertainty was examined in the next case. The value of the robust filter is readily apparent, as the robust filter permits a sensor to degrade severely without much loss in the nominal case, while the Kalman filter is unable to process the noise measurements. Finally, the robust filter also is better equipped to deal with plant perturbations when the control signals have some variation. Although it's estimation error is large due to an unresolved bias, it still easily outperforms the Kalman filter.

6.0 Conclusion

This thesis was concerned with the application of recently developed robust estimation techniques to the Space Shuttle Main Engine. Unlike the fairly well-developed H_∞ control problem, robust estimation theory is relatively new. Promising results in this thesis indicate that it merits continued investigation. The results from the H_∞ estimators developed in this thesis demonstrate that these estimators, when compared with Kalman filtering techniques, gain robustness to both plant and noise model uncertainties without excessive sacrifices in performance when the models are known completely. Section 6.1 summarizes the results obtained in this thesis, and Section 6.2 provides some suggestions for additional research, both for this particular example and for robust estimation in general.

6.1 Summary of results

The thesis opens in Chapter 2 with a brief description of the SSME and how it is controlled through the use of various valves. From there, the linear models for five different power levels are given (as derived in [6]). Based on these linear models, noise models for both the process noise and the measurement noise were developed. Figures show the effect of these noises on the outputs of the system.

Chapter 3 introduced the estimation problem, and the enduring Kalman filter was derived as one solution. Using the models from the previous chapter, the Kalman filter demonstrated its optimality for estimating systems when it has complete knowledge of the plant and the noise. The Kalman filter's limitations were also shown for systems including some perturbation. In response, the H_∞ solution to the estimation problem is introduced.

Chapter 4 develops the H_∞ minimax filter, the estimator that is designed to minimize the estimation error when given the worst-case noise. When given white noise, the Kalman filter's performance is superior to the minimax filter (as expected). However, when a suitable approximation for the worst-case noise is applied to the system the minimax performs significantly better than the Kalman filter. Even when given a simple sinusoidal input, the minimax's error is less than the Kalman filter's (which is designed for white noise input).

The minimax derivation from Chapter 4 is expanded in Chapter 5 to include plant model uncertainty as well as noise model uncertainty. Several different robust estimators

were designed in this chapter based on different perturbations and uncertainty models. The first involved perturbations in just the dynamics and process noise inputs, and the second also included sensor model perturbations. There was no control input for these examples. Robust estimators were found for each of these examples that handled the perturbations better in each case than the Kalman filter designed for the nominal system. Next, an H_∞ estimator that was robust to sensor model uncertainty alone was designed. This estimator was applied both to the case where control input varies and the case where the control input was constant. This result allows for significant degradation of the sensors to occur without sacrificing the quality of the estimates. Finally, a filter that was robust to plant and process noise model uncertainties, including control inputs, was attempted. While the H_∞ estimator was more robust than the Kalman filter, it, too, was insufficient for the perturbed cases.

6.2 Suggestions for further work

There are many available avenues for further work. The first is readily apparent following the last section--to obtain an estimator that is sufficiently robust to all power level perturbations (including plant, control, and noise models). The next arises from expanded work by the authors of [6], who developed system models for 25 different linear points, as well as a method to link each of them together to cover all of the models over time [8]. Designing estimators that are robust to this linked model would be an achievement. Also, the description of the noise as purely white in nature for this application is a simplification. The noise models could be developed to more accurately reflect non-white noise dynamics that are present.

Other avenues for future work center around the estimation theory in this thesis. There are immediate advantages to making this system time-varying, especially in the case where the estimator needs to be robust to responses in the plant caused by changes in the control. By letting the estimator gain be time-varying, the initial covariance following a change in the input function can be made artificially high, thereby creating an abnormally high gain. This gain would make the system rely more on the measurements, and as the system rapidly changes the increased weighting of the measurements would cause the estimator to react much more quickly to abrupt changes. Hopefully this modification would also effectively fight the bias that remains in Section 5.3.2. As the system response settled, the covariance would continue to decrease, in turn decreasing the gain and the reliance on the measurements with their inherent noise.

Another suggestion is only an implementation issue. Should a robust filter be developed that is robust to all possible perturbations, two filters can be combined. After the system has reached plateau (where the control is not changing), the estimator from Section 5.2.2 can be used. When the control changes the second estimator would be used (such as the one explained in the previous paragraph). This would allow for better estimates by cutting down on the chatter at the plateaus while still responding well to changes in the control input.

A final suggestion involves using the robust estimator in a controller. Except for the linear quadratic regulator problem (LQR), which has full state feedback, current controllers solve two Riccati equations--one for the estimator gain, and one for the control gain. This includes not only the linear quadratic gaussian (LQG) controller (a Kalman filter providing the estimates for an LQR), but also the current H_∞ controllers. It is unknown what the effect of adding a *third* Riccati equation would do to the control. (There would be three Riccati equations by using a two-equation robust estimator followed by a third to generate the control gain.)

Appendix A Parametric Uncertainty

The following is a method from [1] showing how to incorporate parametric perturbations into the system matrices.

Given a state-space plant with parameter errors

$$\begin{aligned} x_{k+1} &= \left(A + \sum_{j=1}^{l} \Delta A_j \delta_j \right) x_k + \left(B + \sum_{j=1}^{l} \Delta B_j \delta_j \right) r_k \\ y_k &= \left(C + \sum_{j=1}^{l} \Delta C_j \delta_j \right) x_k + \left(D + \sum_{j=1}^{l} \Delta D_j \delta_j \right) r_k \end{aligned} \tag{A-1}$$

where each δ_j represents a parameter error that is normalized as

$$-1 < \delta_j < 1 \qquad \forall j = 1,\ldots,l \tag{A-2}$$

and

$$r_k = \begin{bmatrix} u_k \\ d_k \end{bmatrix} \tag{A-3}$$

The matrices associated with the uncertainty parameter can be collected into a single matrix

$$N_j = \begin{bmatrix} \Delta A_j & \Delta B_j \\ \Delta C_j & \Delta D_j \end{bmatrix} \in \Re^{(n_x+n_y)\times(n_x+n_r)} \tag{A-4}$$

where n_x, n_y, and n_r are the dimensions of the vectors x_k, y_k, and r_k, respectively. Generally, this matrix will not be of full rank since one parameter rarely affects all of the states and outputs. Hence, N_j can be decomposed using singular value decomposition (SVD) techniques into

$$N_j = \begin{bmatrix} Q_j \\ R_j \end{bmatrix} \begin{bmatrix} S_j & L_j \end{bmatrix} \tag{A-5}$$

where $Q_j \in \Re^{n_x \times n_j}$, $R_j \in \Re^{n_y \times n_j}$, $S_j \in \Re^{n_j \times n_x}$, $L_j \in \Re^{n_j \times n_r}$, and n_j is the rank of the matrix N_j. Combining Eqs. (A-5) and (A-1), the state-space model can now be written as

$$
\begin{aligned}
x_{k+1} &= \left[A + \sum_{j=1}^{l} Q_j \delta_j I_{n_j} S_j \right] x_k + \left[B + \sum_{j=1}^{l} Q_j \delta_j I_{n_j} L_j \right] r_k \\
&= Ax_k + \begin{bmatrix} Q_1 & \dots & Q_l \end{bmatrix} \begin{bmatrix} \eta_1 \\ \vdots \\ \eta_l \end{bmatrix} + Br_k \\
&= Ax_k + Q\eta_k + Br_k
\end{aligned} \tag{A-6}
$$

$$
\begin{aligned}
y_k &= \left[C + \sum_{j=1}^{l} R_j \delta_j I_{n_j} S_j \right] x_k + \left[D + \sum_{j=1}^{l} R_j \delta_j I_{n_j} L_j \right] r_k \\
&= Cx_k + \begin{bmatrix} R_1 & \dots & R_l \end{bmatrix} \begin{bmatrix} \eta_1 \\ \vdots \\ \eta_l \end{bmatrix} + Dr_k \\
&= Cx_k + R\eta_k + Dr_k
\end{aligned} \tag{A-7}
$$

$$
\begin{aligned}
\varepsilon_k &= \begin{bmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_l \end{bmatrix} \\
&= \begin{bmatrix} S_1 \\ \vdots \\ S_l \end{bmatrix} x_k + 0\eta_k + \begin{bmatrix} L_1 \\ \vdots \\ L_l \end{bmatrix} r_k \\
&= Sx_k + Lr_k
\end{aligned} \tag{A-8}
$$

and the relationship between η_k and ε_k is of the form

$$
\begin{aligned}
\eta_k &= \begin{bmatrix} \eta_1 \\ \vdots \\ \eta_l \end{bmatrix} \\
&= \begin{bmatrix} \delta_1 & & \\ & \ddots & \\ & & \delta_l \end{bmatrix} \begin{bmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_l \end{bmatrix} \\
&= \Delta\varepsilon_k
\end{aligned} \tag{A-9}
$$

Expanding Eq. (3.1-2) to include η_k and ε_k, the new open-loop transfer function for the plant becomes

$$\left[\begin{array}{c} x_{k+1} \\ \hline \varepsilon_k \\ e_k \\ y_k \end{array}\right] = \left[\begin{array}{c|cc} A & B' & 0 \\ \hline S & T & 0 \\ M & 0 & -M \\ C & D' & 0 \end{array}\right] \left[\begin{array}{c} x_k \\ \hline \eta_k \\ r_k \\ \hat{x}_k \end{array}\right] \tag{A-10}$$

where

$$\begin{aligned} B' &= [Q \quad B] \\ D' &= [R \quad D] \\ T &= [0 \quad L] \end{aligned} \tag{A-11}$$

This now fits the form of Figure 5.1.

Appendix B Plant Modification

This appendix describes how the results in Section 5.3.2 are obtained. Instead of the output, y, only depending on the plant, $\boldsymbol{P}$, an additional error term is included. This error term is created by the transfer function, $\boldsymbol{W}$, shown in Figure B.1.

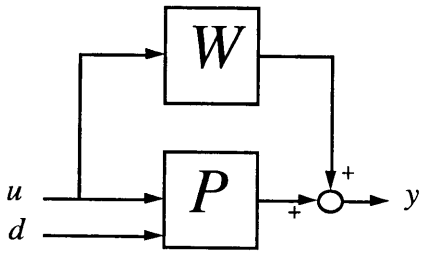

Figure B.1: Block diagram of additive error

Since there are two inputs and two outputs, the form for $\boldsymbol{W}$ is

$$\boldsymbol{W} = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} \tag{B-1}$$

In this case, $\boldsymbol{W}$ represents the difference between the the perturbed system and the nominal system open-loop transfer functions. When the nominal system is the 110% power level model, and the perturbed system is the 90% power level model, the elements of $\boldsymbol{W}$ are

$$\begin{aligned} W_{11} &= T^{90\%}_{P_c\beta_{OPOV}} - T^{110\%}_{P_c\beta_{OPOV}} \\ W_{12} &= T^{90\%}_{P_c\beta_{FPOV}} - T^{110\%}_{P_c\beta_{FPOV}} \\ W_{21} &= T^{90\%}_{MR\beta_{OPOV}} - T^{110\%}_{MR\beta_{OPOV}} \\ W_{22} &= T^{90\%}_{MR\beta_{FPOV}} - T^{110\%}_{MR\beta_{FPOV}} \end{aligned} \tag{B-2}$$

where, for example, $T^{90\%}_{P_c\beta_{OPOV}}$ is the open-loop transfer function from the rotary motion of the oxygen pre-oxidizer valve to chamber pressure for the 90% power level model. Since some of these transfer functions are of very high order, reasonable approximations of these transfer functions are matched to the actual values, and these approximations are used as the W_{ij}'s. In order to prevent a bias from this additional error term in the nominal case, the output from $\boldsymbol{W}$ is reduced to only 25% of its full value. This creates a new plant for the filter to estimate (shown in Figure B.2).

This new plant is then given some parametric uncertainty according to the methodology described in Appendix A (i.e., the estimator is designed for the plant shown in Figure B.2). The uncertainty associated with $\boldsymbol{P}$ is 50%, while that for $\boldsymbol{W}$ is 1000%. In addition to the overall uncertainty percentage, extra uncertainty is given to the input matrices by multiplying ΔB_p by 15, ΔG_p by 5, ΔB_w by 150, and ΔG_w by 30.

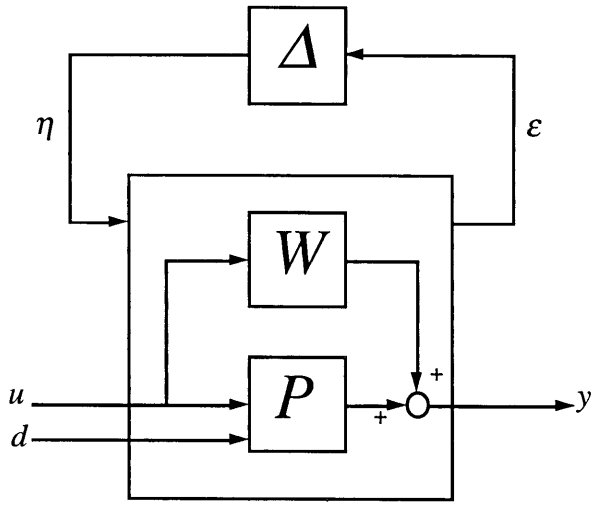

Figure B.2: Block diagram of additive error with parametric uncertainty

Bibliography

[1] Appleby, B., *Robust Estimator Design Using the H_∞ Norm and μ Synthesis*, Ph.D. Thesis, Department of Aeronautics and Astronautics, MIT, 1990.

[2] Brown, R. and Hwang, P., *Introduction to Random Signals and Applied Kalman Filtering*, 2nd ed., Wiley, New York, 1992.

[3] Desoer, C., and Vidyasagar, M., *Feedback Systems: Input-Output Properties*, Academic Press, New York, 1975.

[4] Deyst, J., "A Derivation of the Optimum Continuous Linear Estimator for Systems with Correlated Measurement Noise," *AIAA Journal*, Vol. 7, No. 11, pp. 2116-2119, November 1969.

[5] Doyle, J., Glover, K., Khargonekar, P., and Francis, B., "State-Space Solutions to the H_2 and the H_∞ Control Problems," *IEEE Transactions on Automatic Control*, Vol. 34, No. 8, pp. 831-847, August 1987.

[6] Duyar, A., Eldem, V., Merrill, W. C. and Guo, T., "State Space Representation of the Open-Loop Dynamics of the Space Shuttle Main Engine," *Transactions of the ASME Journal of Dynamic Systems, Measurement, and Control*, Vol. 113, pp. 684-690, December 1991.

[7] Duyar, A., Guo, T., and Merrill, W. C., "Identification of Space Shuttle Main Engine Dynamics," *IEEE Control Systems Magazine* , Vol. 10, No. 4, pp. 59-65, June 1990.

[8] Duyar, A., Eldem, V., Merrill, W. C. and Guo, T., "A Simplified Dynamic Model of the Space Shuttle Main Engine," *Transactions of the ASME Journal of Dynamic Systems, Measurement, and Control*, Vol. 116, pp. 815-819, December 1994.

[9] Duyar, A. Personal interview, April 1996.

[10] Fraser, D., and Potter, J., "The Optimum Linear Smoother as a Combination of Two Optimum Linear Filters" *IEEE Transactions on Automatic Control*, Vol. 7, No. 8, pp. 387-390, August 1969.

[11] Gelb, A., ed., *Applied Optimal Estimation*, The MIT Press, Cambridge, Mass., 1974.

[12] Glover, K., "All Optimal Hankel-Norm Approximations of Linear Multivariable Systems and their L_∞-Error Bounds," *IEEE Transactions on Automatic Control*, Vol. 21, pp. 319-338, 1976.

[13] Jenkins, M., *Main Propulsion Training Manual*, NASA Missions Operations Directorate Report MPS 2102, February 1995.

[14] Kalman, R. E., "A New Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME Journal of Basic Engineering*, Vol. 82D, pp. 34-45, March 1960.

[15] Mangoubi, R., *Robust Estimation and Failure Detection for Linear Systems*, Sc.D. Thesis, Department of Aeronautics and Astronautics, MIT, 1995.

[16] Merrill, W. C. Personal interview, April 1996.

[17] Potter, J., and Fraser, D., "A Formula for Updating the Determinant of the Covariance Matrix," *AIAA Journal*, Vol. 5, No. 7, pp. 1352-1354, July 1967.

[18] Rockwell International Corporation, "SSME Engine Redline and Control Overview," Report BD 88-62, 1988.

[19] Rockwell International Corporation, "Engine Balance and Dynamic Model," Report FSCM No. 02602, Spec. No. RL00001, 1981.

[20] Rosello, A., *A Vehicle Health Monitoring System for the Space Shuttle Reaction Control System During Reentry*, S.M. Thesis, Department of Aeronautics and Astronautics, MIT, 1995.

[21] Sorenson, H. W., *Kalman Filtering: Theory and Applications*, IEEE Press, 1985.

[22] Wiener, N., *Extrapolation, Interpolation, and Smoothing of Stationary Time Series, with Engineering Applications*, Technology Press and Wiley, 1949.