Title
Fault Detection And Identification With Application To Advanced Vehicle Control Systems: Final Report

Authors
Douglas, R. K.
Speyer, J. L.
Mingori, D. L.
et al.

Publication Date
1996

CALIFORNIA PATH PROGRAM
INSTITUTE OF TRANSPORTATION STUDIES
UNIVERSITY OF CALIFORNIA, BERKELEY

Fault Detection and Identification with Application to Advanced Vehicle Control Systems: Final Report

Randal K. Douglas, Jason L. Speyer, D. Lewis Mingori, Robert H. Chen, Durga P. Malladi, Walter H. Chung

California PATH Research Report
UCB-ITS-PRR-96-25

This work was performed as part of the California PATH Program of the University of California, in cooperation with the State of California Business, Transportation, and Housing Agency, Department of Transportation; and the United States Department of Transportation, Federal Highway Administration.

September 1996

ISSN 1055-1425

Fault Detection and Identification
With Application to
Advanced Vehicle Control Systems

Award No. 65H998, M.O.U. 126

Randal K. Douglas, Durga P. Malladi, Robert H. Chen, Walter H. Chung,
Thanh M. Nguyen, Jason L. Speyer and D. Lewis Mingori

Mechanical and Aerospace Engineering Department
University of California, Los Angeles
Los Angeles, California 90095

Fault Detection and Identification With Application to Advanced Vehicle Control Systems

Award No. 65H998, M.O.U. 126

Randal K. Douglas, Durga P. Malladi, Robert H. Chen,
Walter H. Chung, Thanh M. Nguyen, Jason L. Speyer and
D. Lewis Mingori

Mechanical and Aerospace Engineering Department
University of California, Los Angeles
Los Angeles, California 90095

June 20, 1996

Fault Detection and Identi¯cation With Application to Advanced Vehicle Control Systems

Award No. 65H998, M.O.U. 126

Randal K. Douglas, Durga P. Malladi, Robert H. Chen, Walter H. Chung,
Thanh M. Nguyen, Jason L. Speyer and D. Lewis Mingori

Mechanical and Aerospace Engineering Department
University of California, Los Angeles
Los Angeles, California 90095

June 20, 1996

Abstract

A preliminary design of a health monitoring system for automated vehicles is developed and tests in a high-¯delity nonlinear simulation are very encouraging. A new detailed nonlinear vehicle simulation which extends the current simulation is documented and will be used as a future testbed for evaluating the performance of the health monitoring system. A health monitoring system has been constructed for the lateral and longitudinal modes that monitors twelve sensors and three actuators. The approach is to fuse data from dissimilar instruments using modeled dynamic relationships and fault detection and identi¯cation ¯lters. The ¯lters are constructed so that the residual process has directional characteristics associated with the presence of a fault, that is, static patterns. Sensor noise, process disturbances, system parameter variations, unmodeled dynamics and nonlinearities can distort these static patterns. Two candidate residual processing schemes are developed and tested. A Bayesian neural network is trained to announce a fault and the probability of fault occurrence by recognizing fault patterns embedded in the residual. A new multiple hypothesis Shiryayev probability ratio test is also developed. Finally, development of a

time-varying fault detection filter, applicable to maneuvering vehicles with time-varying dynamics, is described.

Keywords. Automated Highway Systems, Automatic Vehicle Monitoring, Fault Detection and Fault Tolerant Control, Neural Networks, Reliability, Sensors, Vehicle Monitoring.

Fault Detection and Identification With Application to Advanced Vehicle Control Systems

Award No. 65H998, M.O.U. 126

Executive Summary

A preliminary design of a health monitoring system for automated vehicles is developed and tests in a high-fidelity nonlinear simulation are very encouraging. A new detailed nonlinear vehicle simulation which extends the current simulation is documented and will be used as a future testbed for evaluating the performance of the health monitoring system. A health monitoring system has been constructed for the lateral and longitudinal modes that monitors twelve sensors and three actuators. The approach is to fuse data from dissimilar instruments using modeled dynamic relationships and fault detection and identification filters. The filters are constructed so that the residual process has directional characteristics associated with the presence of a fault, that is, static patterns. Sensor noise, process disturbances, system parameter variations, unmodeled dynamics and nonlinearities can distort these static patterns. Two candidate residual processing schemes are developed and tested. A Bayesian neural network is trained to announce a fault and the probability of fault occurrence by recognizing fault patterns embedded in the residual. A new multiple hypothesis Shiryayev probability ratio test is also developed. Finally, development of a time-varying fault detection filter, applicable to maneuvering vehicles with time-varying dynamics, is described.

Contents

List of Figures

List of Tables

Chapter 1
Introduction

A proposed transportation system with vehicles traveling at high speed, in close formation and under automatic control demands a high degree of system reliability. This requires a health monitoring and maintenance system capable of detecting a fault as it occurs, identifying the faulty component and determining a course of action that restores safe operation of the system. This report is concerned with vehicle fault detection and identification and describes a vehicle health monitoring system approach based on analytic redundancy.

Analytic redundancy methods for fault detection and isolation use a modeled dynamic relationship between system inputs and measured system outputs to form a residual process. Nominally, the residual process is nonzero only when a fault has occurred and is zero at other times. For an observable system, this simple definition is met by the innovations process of any stable linear observer. A detection filter is a linear observer with the gain constructed so that when a fault occurs, the residual responds in a known and fixed direction. Thus, when a nonzero residual is detected, a fault can be announced and identified.

A complication arises when there are many possible faults because a fault detection ¯lter can only be designed to detect a limited number of faults. This is related to the order of the vehicle dynamics. When more faults need to be identi¯ed, several fault detection ¯lters have to be used with each ¯lter designed to detect and identify some but not all possible faults. The vehicle fault detection system described in this report has four fault detection ¯lters. This raises two di±cult design issues. First, some and probably all faults will not be included in the design of one or more fault detection ¯lters. When such a fault occurs, the residual of all ¯lters will respond, even the residuals of the ¯lters that do not have the fault included in their design. If a fault is not included in a fault detection ¯lter design, the directional characteristics of the residual will be unde¯ned and the fault cannot be properly identi¯ed. The challenge is to build a mechanism that recognizes when a fault detection ¯lter is responding to a fault for which it has not been designed and then to exclude the residual of all such ¯lters from the fault identi¯cation process. If it can be assumed that only one fault occurs at a time, then the residual processor can exclude the residual of any fault detection ¯lters that point to two or more faults.

A second design issue is how the faults should be grouped and identi¯cation delegated among the fault detection ¯lters. Several approaches are taken in the design described in this report. In one, the functional form of a given sensor is restricted. In particular, it is assumed that the sensor fault is a bias of unknown magnitude. The assumption allows this sensor and a certain actuator to be isolated by a single fault detection ¯lter. This point is signi¯cant because the conventional approach to fault detection ¯lter design would not allow a single ¯lter to isolate these two faults and would require this task to be passed on to the residual processing module.

A second fault grouping design consideration is a newly apparent tradeo® between ¯lter parameter robustness, as determined by the eigenvector conditioning, and fault input observability. Using an eigenstructure assignment algorithm, a design objective is to place well-conditioned eigenvectors. However, it has been found recently that for some fault groups, a fault might have only one highly observable direction. This means that while a fault might be large and dynamically active, the residual is small most of the time. The residual would always be large if all associated eigenvectors were placed close to being collinear with the most observable direction. Hence a tradeo® exists. An objective in assigning faults to fault groups is to minimize the impact of this tradeo®.

A third fault grouping design consideration is discussed in (Douglas et al. 1995). In a fault detection system that consists of a bank of fault detection ¯lters and a residual processor such as a neural network, fault isolation is done through the combined e®ort of both system elements. The fault detection ¯lter is a carefully tuned device that uses known dynamic relationships to isolate a fault. The neural network residual processor combines the residuals from several ¯lters and resolves any ambiguity. It is suggested that identifying a fault among a group of dynamically similar faults requires the precision of and is best delegated to the fault detection ¯lters. Furthermore, it is suggested that the reliability of the neural network training would be improved if the fault groups associated with each of the fault detection ¯lters are dynamically dissimilar.

In applications it is unrealistic to expect that a residual process would be nonzero only when a fault has occurred. Sensor noise, process disturbances, system parameter variations, unmodeled dynamics and nonlinearities all contribute to the magnitude of a residual. There are many methods to reduce the impact of these e®ects on the residual but none reduce their e®ect to zero. This means that some threshold detection mechanism must be built.

A simple threshold detection mechanism announces a fault when the size of a residual exceeds some prescribed value. This prescribed value could be determined from empirical studies which balance a rate of false alarm against a rate of miss alarm. A more complicated residual processor might take into account the thresholds of all other residuals as well. Reasoning that if the probability of simultaneous failures is very small, no fault is announced when more than one residual exceeds a threshold. It is more likely that the nonzero residuals are caused by noise or nonlinearities or some cause other than multiple faults.

Two residual processing systems are described in this report. In the ¯rst, a Bayesian neural network considers the residuals from all fault detection ¯lters as constituting a pattern, a pattern which contains information about the presence or absence of a fault. Hence, residual processing is treated as a pattern recognition problem.

The objective of a neural network as a feature classi¯er is to associate a given feature vector with a pattern class taken from a set of pattern classes de¯ned apriori. In an application to residual processing, the feature vector is a fault detection ¯lter residual and the pattern classes are a partitioning of the residual space into fault directions which include the null fault. A Bayesian neural network also provides probabilities of feature classi¯cation conditioned on an observation history. A stochastic training algorithm enhances robustness by treating training sets as as sample sets providing information about the entire population.

A second approach to residual processing described in this report is a modi¯ed Shiryayev sequential probability ratio test extended to include multiple hypotheses. The algorithm, which is derived as a dynamic programming problem, detects and isolates the occurrence of a failure in a conditionally independent measurement sequence in minimum time. The test has been further extended to the detection and identi¯cation of changes with unknown parameters.

This report is organized as follows. Section 2 describes the car models used for fault detection ¯lter design and evaluation. A nonlinear model is derived directly from one provided by the Berkeley PATH research team (Peng 1992). Low-dimensional linear models that include coupled longitudinal and lateral vehicle dynamics are used for fault detection ¯lter design. The high ¯delity nonlinear model is used for evaluation and to obtain the linear models used for design. Section 3 describes the faults to be identi¯ed by the fault detection system. Section 4 describes the design of the fault detection ¯lters. This includes how the faults are grouped for each fault detection ¯lter design and how the fault detection ¯lter eigenstructure placement is done. Section 5 presents an evaluation of the performance of the fault detection ¯lters in a nonlinear simulation.

Sections 6 and 7 describe two candidate fault detection ¯lter residual processing systems. In Section 6 a Bayesian neural network is developed and in Section 7 a multiple hypothesis Shiryayev sequential probability ratio test is described. Both are used to process residuals from all fault detection ¯lters to detect and identify which if any fault has occurred.

In section 8 a six degree of freedom nonlinear vehicle model is developed independently of the model used for the Berkeley simulation of Section 2. This work is done to provide a model that better accommodates a nonplanar, rough road surface, one where the road gradient is di®erent for all four wheels. The model will be used to evaluate the robustness of the health monitoring system to road excitation. This e®ort is a continuation of the work reported in (Douglas et al. 1995).

In section 9 describes a new, disturbance attenuation approach to fault detection ¯lter design. Here, a di®erential game is de¯ned where one player is the state estimate and the adversaries are all the exogenous signals except for the fault to be detected. By treating faults as disturbances to be attenuated, the usual invariant subspace structure associated with fault detection ¯lters is not present except in the limit. By treating model uncertainty as another element in the di®erential game, sensitivity to parameter variations can be reduced.

Section 9 also introduces the notion of a fault detection ¯lter for time-varying systems. This is especially important in applications where a vehicle follows a maneuver such as a merge or a split. While ¯rst considered in the game theoretic ¯lter derivation, it is expected that the Beard-Jones fault detection ¯lter de¯nition will be extended to time-varying systems in the same way.

Appendix A provides a theoretical review of the Beard-Jones detection ¯lter problem. This appendix also includes some early work in extending the Beard-Jones fault detection ¯lter de¯nition to time-varying systems.

Appendix B provides a review of a fault detection ¯lter left eigenvector assignment design algorithm (Douglas and Speyer 1996). The algorithm gives the user eigenvector conditioning information and provides a direct method for achieving maximally achievable eigenvector conditioning. This algorithm is used for the designs in this report.

Appendix C describes a stabilizing fault detection ¯lter gain that bounds the $\mathcal{H}_\infty$ norm of the transfer matrix from system disturbances and sensor noise to the residual. For multi-dimensional faults, a residual direction is identified that enhances the fault signal to noise ratio while maintaining the $\mathcal{H}_\infty$ norm bound.

Chapter 2

Vehicle Model and Simulation Development

In this section, vehicle models are developed for the design and evaluation of fault detection filters. The starting point is a model obtained from the Berkeley PATH research team and derived in (Peng 1992). A version of this model coded in C also is available from Berkeley.

Two variations of the Berkeley model are considered in this section. First, modifications are made to allow for variations in road slope and road noise. The slope is restricted to a constant because of assumptions made in the original derivation of the equations of motion. After modifications, the nonlinear model has 32 states, 3 control inputs and 3 noise inputs. Second, reduced-order linearized models used for detection filter design are developed for a vehicle in a constant radius turn. Linearized models developed for a vehicle operating with zero steering angle are described in (Douglas et al. 1995).

An independent derivation of a six degree of freedom nonlinear vehicle model is also developed to be sure that we understand all the assumptions, definitions and issues which underlie the Berkeley model. This model allows for arbitrary variations in road slope and road noise. Since this model represents a signi¯cant e®ort that was not completed in time to be used in the fault detection ¯lter development, it is described later in Section 8.

2.1 Modi¯cation of Berkeley's Model

Primary sources of vehicle dynamic disturbances are road roughness and variations in the road slope. First, allowing for a road roughness disturbance requires a modi¯cation to the suspension system of the Berkeley nonlinear model. A simple tire model is introduced so that high bandwidth road noise generates physically realistic suspension damping forces. Next a road noise model is described. Finally, the nonlinear model is modi¯ed to allow for nonzero road slope. It is important to note that because of assumptions made in the original derivation of the equations of motion, the slope is still a constant although now not necessarily zero.

2.1.1 Suspension System

In the Berkeley nonlinear model, the suspension system is modeled as a spring and damper and the tire is sti®. The sti® tire causes road displacements to pass directly to the suspension system resulting in unreasonably large damping forces. Modeling the tire as a mass and linear spring allows the tire to act as a low pass ¯lter with respect to road displacements and eliminates the unrealistic suspension damping forces. Since the mass of the tire is very small relative to the car, the tire model is simpli¯ed to a linear spring as shown in Figure 2.1. The vehicle equations are modi¯ed by adding four states, the suspension force of each wheel, which are derived as follows. The suspension force F_s acting on each wheel is given by

$$F_s = -C_1(x_2 - x_1 - x_{30})[1 + C_2(x_2 - x_1 - x_{30})^4] - D_1(\dot{x}_2 - \dot{x}_1) + mg \tag{2.1}$$

where x_{30} is the length of the suspension system when a nominal load mg is applied. The force F_t transmitted to the suspension by the tire spring is given by

$$F_t = -K_t(x_1 - r - x_{10}) \tag{2.2}$$

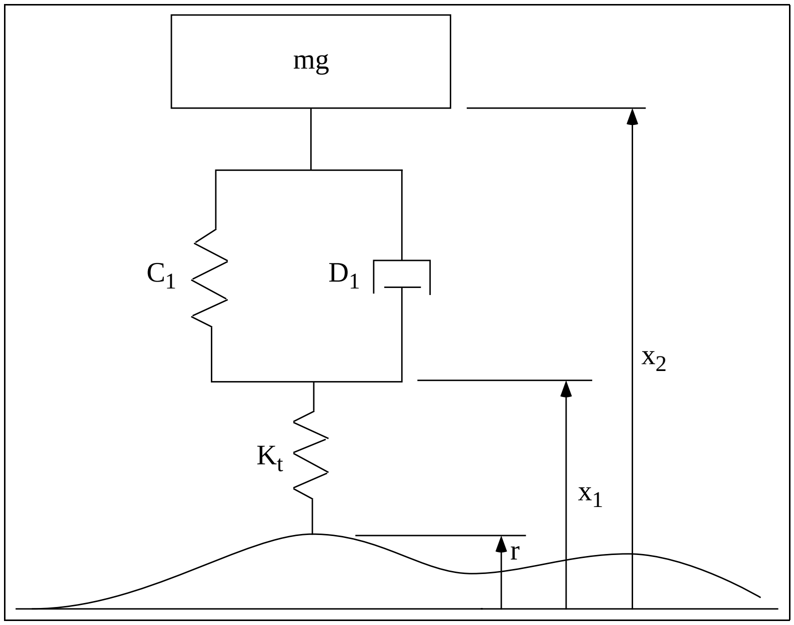

Figure 2.1: Simpli¯ed suspension and tire model.

where K_t is the tire spring sti®ness and x_{10} is the nominal tire radius. Since the tire is massless, the tire spring force is equal to the suspension force.

$$F_t = F_s \tag{2.3}$$

The tire spring force F_t is eliminated by rearranging (2.2) to get

$$x_1 = r + x_{10} - \frac{F_t}{K_t} \tag{2.4a}$$

$$\dot{x}_1 = \dot{r} - \frac{\dot{F}_t}{K_t} \tag{2.4b}$$

and then combining (2.1), (2.3) and (2.4) as

$$\dot{F} = \frac{K_t}{D_1}\{-F + mg - C_1(x_2 - r - x_{10} - x_{30} + \frac{F}{K_t})[1 + C_2(x_2 - r - x_{10} - x_{30} + \frac{F}{K_t})^4] - D_1(\dot{x}_2 - \dot{r})\}$$

where $F \equiv F_s$. Adding a suspension force state for each wheel to the nonlinear equations of motion brings the number of states to thirty.

2.1.2 Road Roughness

A road roughness model is derived from Robson (Robson 1980). Empirical data shows that the road displacement r can be modeled as a random process with power spectral density given by

$$P(\lambda) = \begin{cases} \frac{R_c}{\lambda_0^{2.5}}, & \lambda \le \lambda_0 \\ \frac{R_c}{\lambda^{2.5}}, & \lambda \ge \lambda_0 \end{cases} \tag{2.5}$$

where, for a typical freeway, $\lambda_0 = 0.01$cycles/m and $R_c = 10^{-7}\text{m}^2$. R_c depends on the road roughness.

The model (2.5) is di±cult to apply because $P(\lambda)$ is not a rational function of λ. An obvious simpli¯cation is to take the exponent of the denominator to be 2 and write the power spectral density as

$$P(\lambda) = \frac{R_c}{\lambda^2 + {\lambda_0}^2} \tag{2.6}$$

Using (2.6), random road displacements r are given by a ¯rst-order di®erential equation

$$\dot{r} = -\omega_0 r + \sqrt{2\pi V R_c}\,\omega$$

where $\omega_0 = 2\pi V \lambda_0$, V is the nominal velocity of the car and ω is white noise with power spectral density one.

For simplicity, road roughness for the right and left tires is the same. Also, road roughness for the front tires is applied to the rear tires but after a time delay. The delay is given by the speed of the car and the distance between the front and rear tires. Thus, only two road roughness noise states are added to the vehicle nonlinear equations of motion. This brings the number of states to thirty two.

2.1.3 Slope

Because of assumptions made in the original derivation of the vehicle equations of motion, allowing for non-zero road slope or superelevation can only be done by rotating the gravity vector. As shown in Figure 2.2, the forces acting on a car where the road slope is γ degrees are equivalent to the forces acting on a car where the road has zero slope and the gravity vector has been rotated γ degrees. Of course, this is only true if the road slope is constant.

Because of assumptions made in the original derivation of the vehicle equations of motion, allowing for time varying road slope or superelevation requires rederivation of the vehicle equations of motion. The equations of motion are derived from ¯rst principles in Section 8.

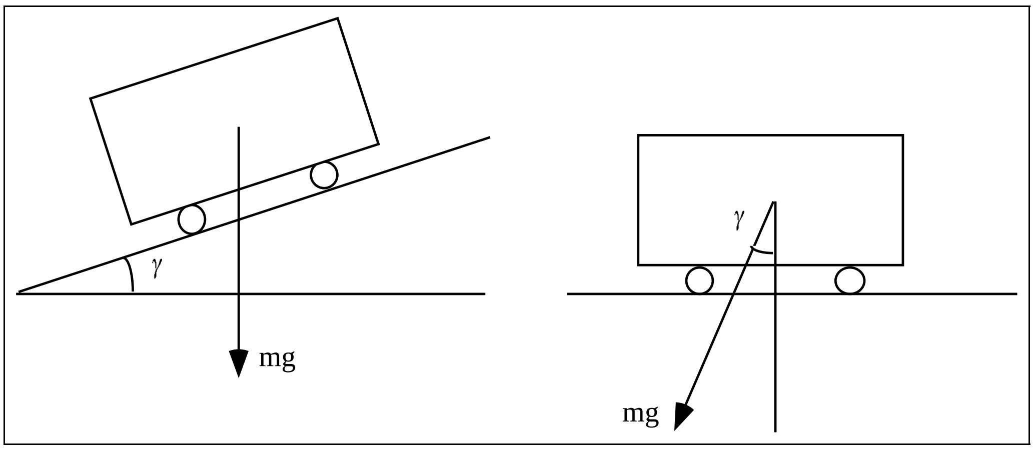

Figure 2.2: Constant non-zero road slope is simulated by rotating the gravity vector.

2.1.4 Evaluation

In this section, the modi¯ed tire, suspension system and rough road models are evaluated using the Berkeley nonlinear vehicle simulation. The car is put in a constant radius turn with a steering angle of 0.005 deg. and constant speed of $24.87\frac{\text{m}}{\text{sec}}$ which is about 56mph. In the next section, this nominal operating point is used to derive a linear model for fault detection ¯lter design. Figures 2.3, 2.4 and 2.5 illustrate some of the more relevant vehicle states and outputs. All variables but one appear to take on reasonable and expected values.

An explanation for the large longitudinal acceleration values, which are between -0.05g and 0.05g as shown in Figure 2.4, is that the tire and suspension system is modeled as rigid along the longitudinal direction. This rigid connection allows variations in the tractive force due to the rough road to directly a®ect the longitudinal acceleration of the car. Since the road noise model is only used for fault detection system robustness testing, large variations in the longitudinal acceleration only imply a more conservative testing environment.

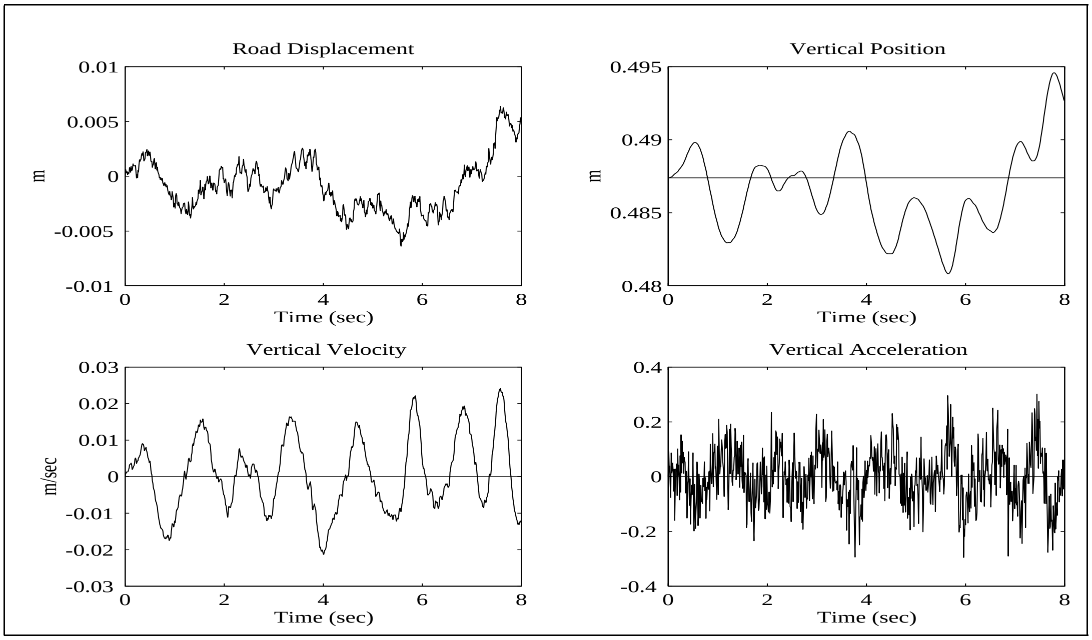

Figure 2.3: Rough road simulation.

2.2 Linear Model

In this section, a linearized model for a car making a constant radius turn is developed using the modi¯ed Berkeley nonlinear model. Linearized models developed for a vehicle operating with zero steering angle are described in (Douglas et al. 1995). Linearized models are found numerically rather than analytically. An analytical approach taking partial derivatives is impractical because the nonlinear model is too complicated. The procedure is as follows.

First, a computer run is made in which the car makes a turn at a constant speed of $24.87\frac{\mathrm{m}}{\mathrm{sec}} \simeq$ 56mph to obtain steady state values for each state. The tire steering angle is 0.005 rad which produces about 0.1g lateral acceleration and a 638.73 meter radius turn. The nonlinear model is then linearized about this nominal operating point using the central di®erence method.

The nonlinear model has the form:

$$\underset{\sim}{x} = f(x, u) \tag{2.7a}$$

$$y = Cx + D\underset{\sim}{x} \tag{2.7b}$$

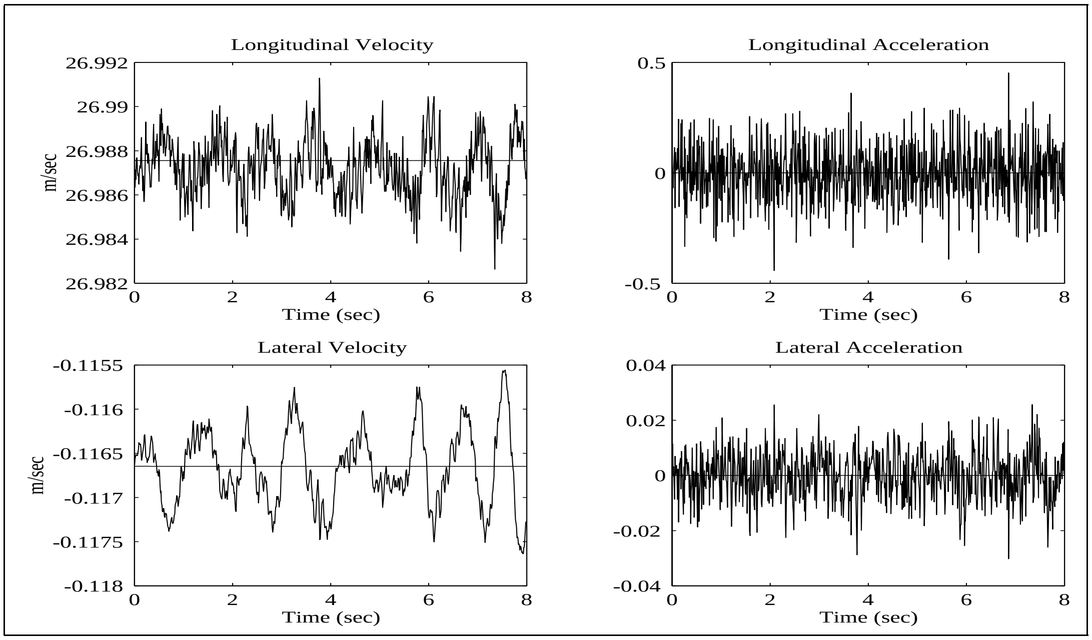

Figure 2.4: Rough road simulation.

Suppose the nominal operating point is $(x_0,\ u_0)$ where $f(x_0, u_0) = 0$. Take perturbations δx, δu about the nominal point, that is, let

$$\begin{aligned} x &= x_0 + \delta x \\ u &= u_0 + \delta u \end{aligned}$$

Also approximate $\frac{\partial f}{\partial x}$ and $\frac{\partial f}{\partial u}$ as

$$\begin{aligned} \frac{\partial f}{\partial x} &\approx \frac{\Delta f}{\Delta x} = \left.\frac{f(x+\delta x, u) - f(x-\delta x, u)}{2\delta x}\right|_{x=x_0, u=u_0} \\ \frac{\partial f}{\partial u} &\approx \frac{\Delta f}{\Delta u} = \left.\frac{f(x, u+\delta u) - f(x, u-\delta u)}{2\delta u}\right|_{x=x_0, u=u_0} \end{aligned}$$

Equation (2.7a) may now be approximated as

$$\dot{x}_0 + \delta\dot{x} = f(x_0, u_0) + \left.\frac{\partial f}{\partial x}\right|_{x=x_0, u=u_0} \delta x + \left.\frac{\partial f}{\partial u}\right|_{x=x_0, u=u_0} \delta u + \cdots$$

Truncating the higher order terms and using the approximations given above for the partial derivatives, produces the following linear equations in the perturbed state δx, input δu and

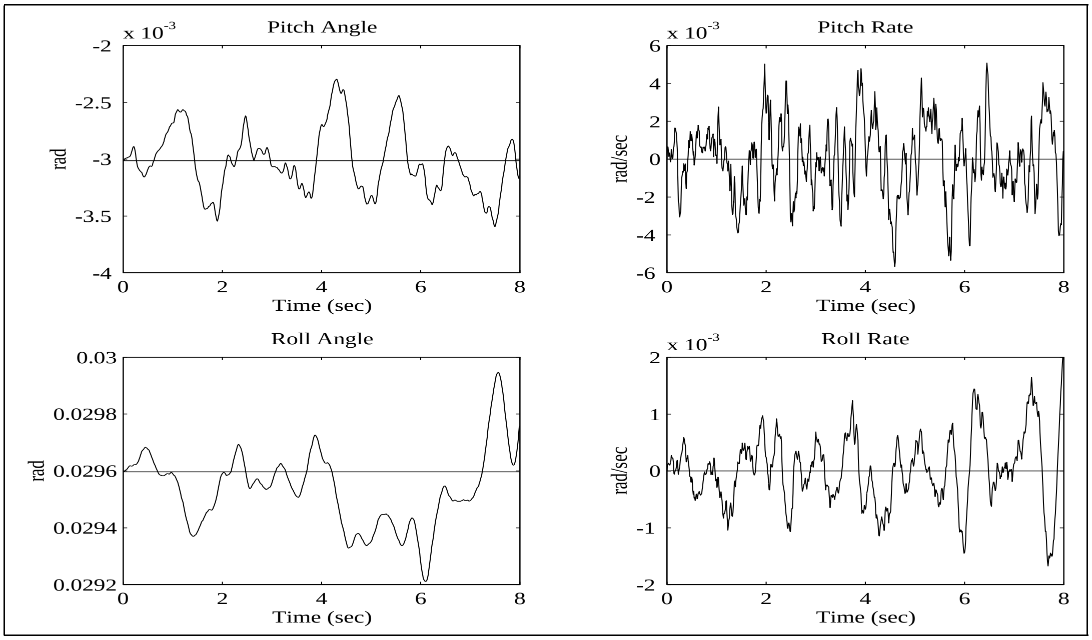

Figure 2.5: Rough road simulation.

output $\mathbf{y}$

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u} \tag{2.8a}$$

$$\begin{aligned} \mathbf{y} &= C\mathbf{x} + D\dot{\mathbf{x}} \\ &= (C + DA)\mathbf{x} + DB\mathbf{u} \end{aligned} \tag{2.8b}$$

where

$$\begin{aligned}
\mathbf{x} &= [m_a \quad w_e \quad v_x \quad x \quad v_y \quad y \quad v_z \quad z \quad \phi \quad \dot{\phi} \quad \theta \quad \dot{\theta} \quad \epsilon \quad \dot{\epsilon} \quad w_{fl} \quad w_{fr} \quad w_{rl} \\
&\quad\; w_{rr} \quad X \quad Y \quad yr \quad \dot{yr} \quad \epsilon_{des} \quad \alpha \quad \tau_b \quad \beta \quad F_{fl} \quad F_{fr} \quad F_{rl} \quad F_{rr} \quad r_r \quad r_f]^T \\
\mathbf{y} &= [m_a \quad w_e \quad \dot{v}_x \quad \dot{v}_y \quad \dot{v}_z \quad \dot{\phi} \quad \dot{\theta} \quad \dot{\epsilon} \quad w_{fl} \quad w_{fr} \quad w_{rl} \quad w_{rr}]^T \\
\mathbf{u} &= [\alpha_c \quad \tau_{bc} \quad \beta_c \quad r_{rc} \quad r_{fc} \quad \gamma]^T \\
A &= \left[\frac{\partial f}{\partial x}\right]\Bigg|_{x=x_0, u=u_0} \\
B &= \left[\frac{\partial f}{\partial u}\right]\Bigg|_{x=x_0, u=u_0}
\end{aligned}$$

and where A is a 32×32 real matrix, B is a 32×6 real matrix and DB is a zero matrix.

Symbols in $\tilde{x}$, $\tilde{y}$ and $\tilde{u}$ are defined in the list of symbols.

Several sizes of perturbations must be taken to find one that gives the best approximation of the partial derivatives. If the perturbation is too small, there is a truncation error in computing the difference $f(x+\tilde{x},u)-f(x-\tilde{x},u)$. If the perturbation is too large, a roundoff error occurs in computing $f(x+\tilde{x},u)$ and $f(x-\tilde{x},u)$; also nonlinearities become important. According to our experience, $\frac{\tilde{x}}{x}$ and $\frac{\tilde{u}}{u} \approx 10^{-4}$ is a good rule for selecting the size of the perturbation when using the central differences method.

The resulting linear model is tested in a simulation to see how well it describes the nonlinear model by comparing the states of the linear and nonlinear models when various control inputs are applied. Over the speed range of $23\frac{\mathrm{m}}{\mathrm{sec}}$ to $27\frac{\mathrm{m}}{\mathrm{sec}}$, errors in the states are less than 10% except for yaw rate where the error is less than 15%.

The linear model generated as described above was intended for use in designing the fault detection filters. Since the model dimension is large with 32 states, before the model is used for design, it is simplified to the extent possible without significant loss of accuracy. The model simplification is accomplished in three steps, the first two of which result in no loss of accuracy.

First, since the present fault detection filter designs do not explicitly include either road roughness or road slope, these three noise inputs and two associated states are truncated from the model (2.8). The model (2.8) becomes

$$
\begin{aligned}
\dot{\tilde{x}}_r &= A_r\tilde{x}_r + B_r\tilde{u}_r \\
\tilde{y} &= C_r\tilde{x}_r
\end{aligned}
$$

where

$$
\begin{aligned}
\tilde{x}_r &= [\,m_a \quad w_e \quad v_x \quad x \quad v_y \quad y \quad v_z \quad z \quad \phi \quad \dot{\phi} \quad \theta \quad \dot{\theta} \quad \epsilon \quad \dot{\epsilon} \quad w_{fl} \quad w_{fr} \quad w_{rl} \\
&\qquad w_{rr} \quad X \quad Y \quad yr \quad \dot{yr} \quad \epsilon_{des} \quad \alpha \quad \tau_b \quad \beta \quad F_{fl} \quad F_{fr} \quad F_{rl} \quad F_{rr}\,]^T \\
\tilde{u}_r &= [\,\alpha_c \quad \tau_{bc} \quad \beta_c\,]^T
\end{aligned}
$$

Next, by inspection of the equations, it is possible to rearrange the sequence of states such that the linearized equations assume the following partitioned form:

$$
\begin{aligned}
\dot{x}_r &= \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} A_1 & 0 \\ A_{21} & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u_r \\
y &= \begin{bmatrix} C_1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}
\end{aligned}
$$

where

$$
\begin{aligned}
x_1 &= [\, m_a \quad w_e \quad v_x \quad v_y \quad v_z \quad z \quad \phi \quad \dot{\phi} \quad \theta \quad \dot{\theta} \quad \dot{\epsilon} \quad w_{fl} \quad w_{fr} \quad w_{rl} \quad w_{rr} \\
&\qquad \alpha \quad \tau_b \quad \beta \quad F_{fl} \quad F_{fr} \quad F_{rl} \quad F_{rr} \,]^T \\
x_2 &= [\, x \quad y \quad \epsilon \quad X \quad Y \quad yr \quad \dot{yr} \quad \epsilon_{des} \,]^T
\end{aligned}
$$

In this form, both x_1 and y are independent of x_2. Thus x_2 can be deleted from the model without a®ecting the transfer function from u to y. Based on this observation, x_2 is removed from the model, which then becomes

$$
\begin{aligned}
\dot{x}_1 &= A_1 x_1 + B_1 u_r \\
y &= C x_1
\end{aligned}
$$

where A_1 is an 22×22 matrix, B_1 is an 22×3 matrix and C_1 is a 12×22 matrix.

As shown in (Douglas et al. 1995), when the nominal operating point associated with the linearized system is one where the car is not making a turn, the longitudinal and lateral dynamics decouple exactly. However, the case considered here has a nonzero nominal steering angle so the longitudinal and lateral dynamics do not decouple. All 22 states are included in the linear model order reduction process explained in the next section.

2.3 Reduced-Order Model

Previous manipulation involved no approximation. For further model simpli¯cation, some approximation must occur. After the linear model is derived, the ¯rst thing one should do is check the eigenvalues. Then, two approaches are presented to get reduced-order models. The ¯rst approach is to set the derivatives of certain fast states to zero. Using this philosophy, states with large negative eigenvalues can be dropped. However, a correction should be made using the deleted states to remove the steady state error. Consider a linear system modeled as:

$$
\begin{aligned}
\dot{x} &= Ax + Bu \\
y &= Cx + Du
\end{aligned}
$$

Suppose this model is written in a partitioned form

$$
\begin{aligned}
\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u \\
y &= \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + Du
\end{aligned}
$$

where x_2 contains the *fast states*. Set the derivative of x_2 to zero and solve the resulting equations for x_2 as a function of x_1 and u. This leads to

$$
x_2 = -A_{22}^{-1} A_{21} x_1 - A_{22}^{-1} B_2 u
$$

Substitute this result into the expressions for $\dot{x}_1$ and y to obtain the reduced order model:

$$
\begin{aligned}
\dot{x}_1 &= \left[A_{11} - A_{12} A_{22}^{-1} A_{21}\right] x_1 + \left[B_1 - A_{12} A_{22}^{-1} B_2\right] u \\
y &= \left[C_1 - C_2 A_{22}^{-1} A_{21}\right] x_1 + \left[D - C_2 A_{22}^{-1} B_2\right] u
\end{aligned}
$$

this model preserves the static input-output relationships.

A second approach is to use balanced realization. Balancing refers to an algorithm which finds a realization that has equal and diagonal controllability and observability grammians. The diagonal of the joint grammian can be used to reduce the order of the model. Since the diagonal elements of the grammian, the Hankel singular values $g(i)$, reflect the combined controllability and observability of each state, it is reasonable to remove those states from the model for which $g(i)$ is small. Elimination of these states retains the most important input-output characteristics of the original system. After balanced realization has been done, a truncation is used to obtain a reduced-order model. For example, if the full-order model is

$$
\begin{aligned}
\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u \\
y &= \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + Du
\end{aligned}
$$

then, the reduced-order model is

$$\dot{x}_1 = A_{11}x_1 + B_1 u$$

This is the approach originally proposed by Moore (Moore 1981). Using this approach it is possible to calculate a bound on the error introduced by deleting states.

At the end of the previous section, Section 2.2, a linear model is developed. Its eigenvalues are -227.45, -193.79, -159.51, $-132.21 \pm 2.62i$, $-135.35 \pm 1.85i$, -138.68, $-26.16 \pm 4.47i$, $-1.99 \pm 6.63i$, $-3.10 \pm 6.07i$, $-1.31 \pm 5.60i$, -0.046, $-7.09 \pm 2.48i$, -90.91, -1.25 and -80. Observe that ten of these eigenvalues are significantly larger than the rest. Two of the *fast* eigenvalues, -90.91 and -80, happen to be associated with the actuator dynamics. These modes should be retained if the linear model is to be used to design fault detection filters for actuator faults. From this we conclude that at least eight state variables can be dropped.

In method one, by looking at the eigenvectors corresponding to the large eigenvalues, the eight fast mode states are the four wheel speeds w_{fl}, w_{fr}, w_{rl}, w_{rr} and the four suspension forces F_{fl}, F_{fr}, F_{rl}, F_{rr}. Truncating these eight states produces a fourteenth-order model. In method two, the eight states with the smallest Hankel singular values are dropped. These methods combine the states in such a way that they lose their physical significance, so explicit identification of the deleted states is not possible. Table 2.1 summarizes a comparison of the two methods for model reduction.

The eigenvalues given in Table 2.1 show that the first method produces better results because the eigenvalues are closer to the full-order model eigenvalues. The second method truncates some *slow* states which results in a large change in the eigenvalues.

	Eigenvalues				
Method 1	-26.46 ± 4.32i	-7.53 ± 2.94i	-0.049	-2.07 ± 6.54i	
	-1.44 ± 5.50i	-2.82 ± 5.60i	-90.91	-80	-1.25
Method 2	-108.98 ± 37.67i	-0.046	-18.33 ± 13.19i	7.10 ± 2.49i	
	-79.27	-3.07 ± 6.10i	-2.39 ± 6.30i	-1.31 ± 5.62i	

Table 2.1: Eigenvalues for the vehicle dynamics using two model reduction methods.

Another test, based on frequency response, can also be performed to see which method is best. Singular values of the multivariable input to output frequency response are plotted from frequencies of 10^{-1} to $10^{2}\frac{\text{rad}}{\text{sec}}$. The reason for choosing this frequency range is that it roughly corresponds to that of the control inputs to a car. As shown in Figure 2.6, the error of largest singular value of the reduced-order model derived from method two is slightly better than the error of method one. However, the errors of the other two singular values of method two are much worse than the errors of method one.

An interpretation of this result is that the ¯rst order reduction method tends to preserve model ¯delity with respect to each input while the second method tends to preserve model ¯delity for only the most important input and output pair. For the purpose of fault detection ¯lter design, the ¯rst method is more appropriate because fault detection ¯lters are built for each control input.

A fourteen-state model is obtained using the ¯rst model order reduction method and is used subsequently to design fault detection ¯lters for actuators faults. The system matrices A, B, C and D are given in Appendix D. The measured outputs are

y_{m_a}	Engine manifold air mass (kg).
y_{ω_e}	Engine speed ($\frac{\text{rad}}{\text{sec}}$).
$y_{\ddot{x}}$	longitudinal acceleration ($\frac{\text{m}}{\text{sec}^2}$).
$y_{\ddot{y}}$	lateral acceleration ($\frac{\text{m}}{\text{sec}^2}$).
$y_{\ddot{z}}$	heave acceleration ($\frac{\text{m}}{\text{sec}^2}$).
$y_{\dot{\phi}}$	roll rate ($\frac{\text{rad}}{\text{sec}}$).

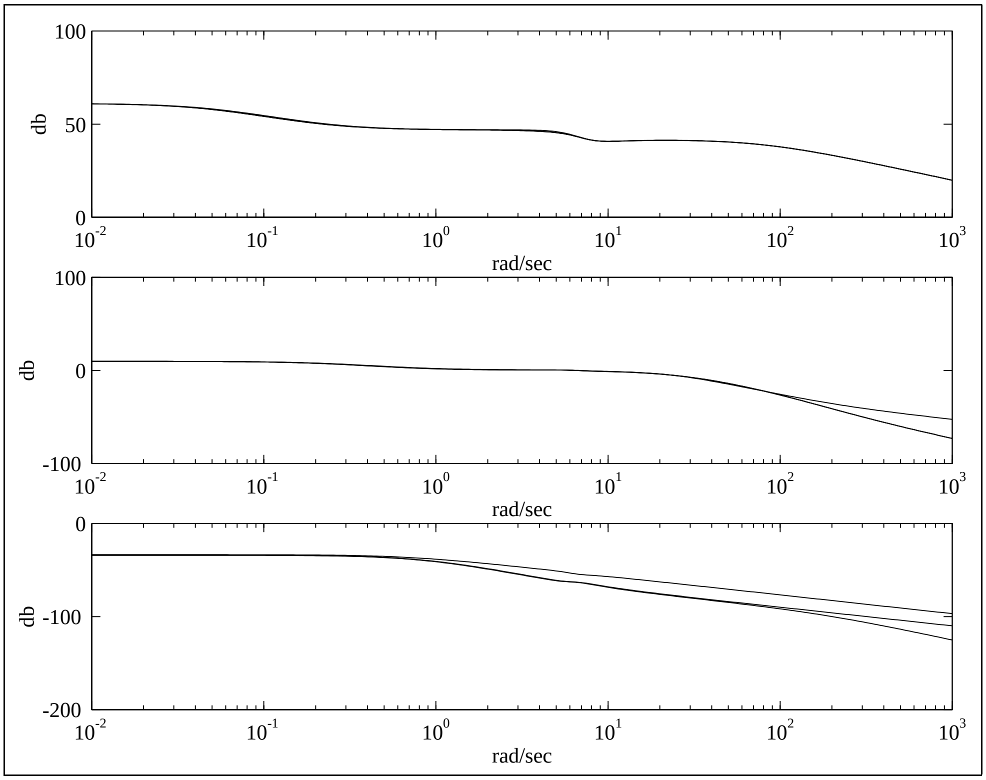

Figure 2.6: Singular value frequency response of full-order and fourteen state reduced-order models.

$y_{\dot{\theta}}$	pitch rate ($\frac{\text{rad}}{\text{sec}}$).
$y_{\dot{\epsilon}}$	yaw rate ($\frac{\text{rad}}{\text{sec}}$).
$y_{\omega_{fl}}$	front left wheel speed ($\frac{\text{rad}}{\text{sec}}$).
$y_{\omega_{fr}}$	front right wheel speed ($\frac{\text{rad}}{\text{sec}}$).
$y_{\omega_{rl}}$	rear left wheel speed ($\frac{\text{rad}}{\text{sec}}$).
$y_{\omega_{rr}}$	rear right wheel speed ($\frac{\text{rad}}{\text{sec}}$).

and the control inputs are

α	Throttle angle (deg).
τ_b	Brake torque (Nm).

β Steering angle (rad).

While a fourteen-state linear model is used for the design of actuator fault detection ¯lters, a twelveth-order model is used to design sensor fault detection ¯lters. Recall that two fast modes retained in the fourteen-state reduced-order model are associated with the actuator dynamics. The eigenvalue -90.91 is associated with the throttle actuator and -80 with the steering actuator. For the design of sensor fault detection ¯lters, these modes may also be deleted. Note that since the actuator dynamics are in series with the other dynamics, the reduced-order eigenvalues do not change. Singular values of the multivariable input to output frequency response are illustrated in Figure 2.7. The model reduction error is seen to be very slightly worse than for the fourteenth-order model. The system matrices

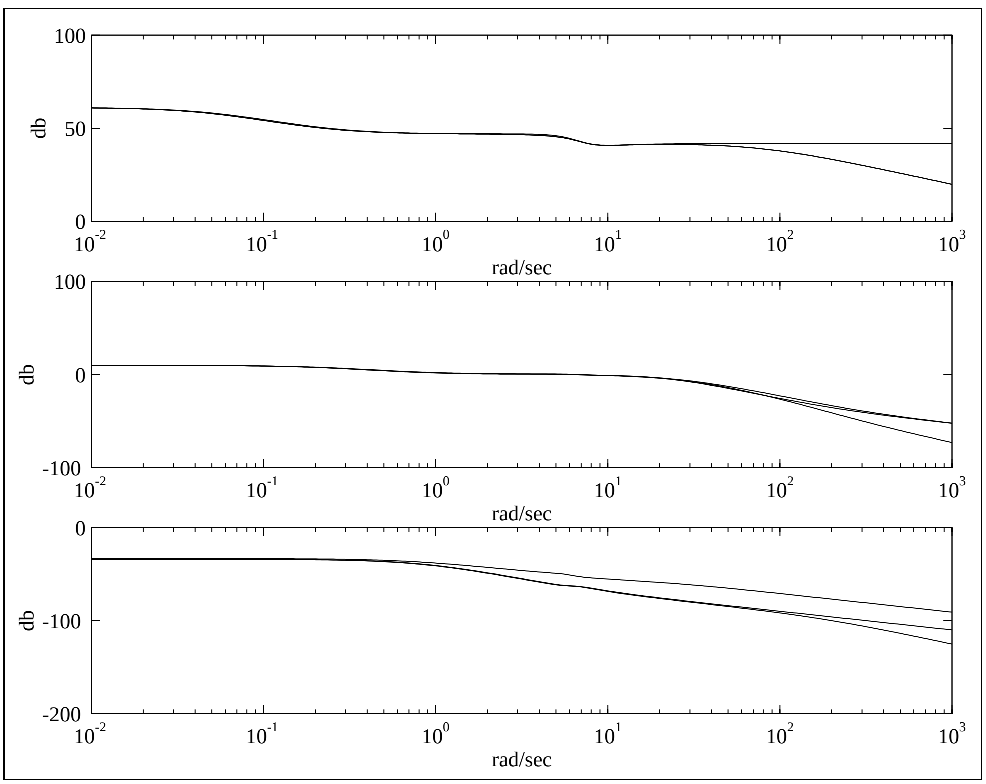

Figure 2.7: Singular value frequency response of full-order and twelve state reduced-order models.

A, B, C and D for the twelve-state reduced-order model are given in Appendix D.

CHAPTER 3

Fault Selection

ANALYTIC REDUNDANCY is an approach to health monitoring that compares dissimilar instruments using a detailed model of the system dynamics. Therefore, to detect a fault in a given sensor, there must be a dynamic relationship between the sensor and other sensors or actuators. That is, the information provided by a monitored sensor must, in some form, also be provided by other sensors. Analytic redundancy also can be used to e®ectively monitor the health of system actuators and even the dynamic behavior of the system itself. But, as with sensors, if some part of the vehicle is to be monitored for proper operation, then that part has to produce some observable dynamic e®ect.

In automated vehicles, these requirements preclude monitoring nonredundant sensors such as obstacle detection or lane position sensors. The information provided by a radar or infrared sensor designed to detect objects in the vehicle's path has no dynamic correlation with other sensors on the vehicle. A sensor that detects the vehicle's position in a lane is the only sensor that can provide this information. Actuators that do no observable action are also difficult to monitor. For example, the health of a power window actuator is easily monitored by the driver. But, unless specialized sensors are installed, no other part of the car is affected by the operation of this actuator and there is no analytic redundancy.

Before describing how faults are modeled, it is necessary to describe how a fault detection filter works. Most of the details are left to Appendix A. For a thorough background, several references are available, a few of which are (Douglas 1993), (White and Speyer 1987) and (Massoumnia 1986). Consider a linear time-invariant system with q failure modes and no disturbances or sensor noise

$$\dot{x} = Ax + Bu + \sum_{i=1}^{q} F_i m_i \tag{3.1a}$$

$$y = Cx + Du \tag{3.1b}$$

The system variables x, u, y and the m_i belong to real vector spaces and the system maps A, B, C, D and the F_i are of compatible dimensions. Assume that the input u and the output y both are known. The F_i are the failure signatures. They are known and fixed and model the directional characteristics of the faults. The m_i are the failure modes and model the unknown time-varying amplitude of faults. The m_i do not have to be scalar values.

A fault detection filter is a linear observer that, like any other linear observer, forms a residual process sensitive to unknown inputs. Consider a full-order observer with dynamics and residual

$$\dot{\hat{x}} = (A + LC)\hat{x} + Bu - Ly \tag{3.2a}$$

$$r = C\hat{x} + Du - y \tag{3.2b}$$

Form the state estimation error $e = \hat{x} - x$ and the dynamics and residual are

$$\begin{aligned} \dot{e} &= (A + LC)e - \sum_{i=1}^{q} F_i m_i \\ r &= Ce \end{aligned}$$

In steady-state, the residual is driven by the faults when they are present. If the system is (C, A) observable, and the observer dynamics are stable, then in steady-state and in the absence of disturbances and modeling errors, the residual r is nonzero only if a fault has occurred, that is, if some m_i is nonzero. Furthermore, when a fault does occur, the residual is nonzero except in certain theoretically relevant but physically unrealistic situations. This means that any stable observer can detect the presence of a fault. Simply monitor the residual and when it is nonzero a fault has occurred.

In addition to detecting a fault, a fault detection filter provides information to determine which fault has occurred. An observer such as (3.2) becomes a fault detection filter when the observer gain L is chosen so that the residual has certain directional properties that immediately identify the fault. The gain is chosen to partition the residual space where each partition is uniquely associated with one of the design fault directions F_i. A fault is identified by projecting the residual onto each of the residual subspaces and then determining which projections are nonzero.

Before the fault detection filter design (3.2) can begin, a system model with faults has to be found with the form (3.1). Twelve sensors and three actuators are associated with the linearized vehicle dynamics described in Section 2.3. The sensors measure the engine manifold airflow and engine speed, the vehicle forward, lateral and heave accelerations, the roll, pitch and yaw rate and the angular speed of each of the four wheels. The actuators control the engine throttle, the brake torque and the steering angle.

3.1 Sensor Fault Models

Sensor faults can be modeled as an additive term in the measurement equation

$$y = Cx + E_i\mu_i \tag{3.3}$$

where E_i is a column vector of zeros except for a one in the i^{th} position and where μ_i is an arbitrary time-varying real scalar. Since, for fault detection filter design, faults are expressed as additive terms to the system dynamics, a way must be found to convert the E_i sensor fault form of (3.3) to an equivalent F_i form as in (3.1). Let F_i satisfy

$$CF_i = E_i$$

and de¯ne a state estimation error e as

$$e \;=\; x - \hat{x} + F_i\mu_i$$

Using (3.2), the error dynamics are

$$\dot{e} = (A + LC)e + F_i\dot{\mu}_i - AF_i\mu_i \tag{3.4}$$

and a sensor fault E_i in (3.3) is equivalent to a two-dimensional fault F_i

$$\dot{x} = Ax + Bu + F_i m_i \qquad \text{with } F_i = \left[F_i^1, F_i^2\right]$$

where the directions F_i^1 and F_i^2 are given by

$$E_i \;=\; CF_i^1 \tag{3.5a}$$

$$F_i^2 \;=\; AF_i^1 \tag{3.5b}$$

An interpretation of the e®ect of a sensor fault on observer error dynamics follows from (3.4) where F_i^1 is the sensor fault rate $\dot{\mu}_i$ direction and F_i^2 is the sensor fault magnitude μ_i direction. This interpretation suggests a possible simpli¯cation when information about the spectral content of the sensor fault is available. If it is known that a sensor fault has persistent and signi¯cant high frequency components, such as in the case of a noisy sensor, the fault direction could be approximated by the F_i^1 direction alone. Or, if it is known that a sensor fault has only low frequency components, such as in the case of a bias, the fault direction could be approximated by the F_i^2 direction alone. For example, if a sensor were to develop a bias, a transient would be likely to appear in all fault directions but, in steady-state, only the residual associated with the faulty sensor should be nonzero.

Using the linearized dynamics of Section 2.3, an engine manifold air°ow measurement is given by the ¯rst element of the system output (d.1). Therefore, any fault in the engine manifold air°ow sensor can be modeled as an additive term in the measurement equation as in (3.3)

$$y = Cx + E_{y_{ma}}\mu_{y_{ma}}$$

where

$$E_{y_{m_a}} = \begin{bmatrix} 1, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0 \end{bmatrix}^T$$

and where $\mu_{y_{m_a}}$ is an arbitrary time-varying real scalar. An equivalent two-dimensional fault $F_{y_{m_a}}$ found by solving (3.5) is

$$F_{y_{m_a}} = \begin{bmatrix} 1, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0 \\ -22.42, & 306.69, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0 \end{bmatrix}^T$$

Other vehicle sensor fault directions are found in the same way.

3.2 Actuator Fault Models

A linear model partitioned to isolate ¯rst-order actuator dynamics can be expressed as

$$\begin{bmatrix} \dot{x} \\ \dot{x}_a \end{bmatrix} = \begin{bmatrix} A & B \\ 0 & -\omega \end{bmatrix} \begin{bmatrix} x \\ x_a \end{bmatrix} + \begin{bmatrix} 0 \\ \omega \end{bmatrix} u$$

where x_a is a vector of actuator states. A fault in a control input is modeled as an additive term in the system dynamics. In the case of a fault appearing at the input of an actuator, that is the actuator command, the fault has the same direction as the associated column of the $[0,\ \omega]^T$ matrix. A fault appearing at the output of an actuator, the actuator position, has the same direction as the associated column of the $[B^T,\ 0]^T$ matrix. In the Berkeley nonlinear vehicle model, the actuator dynamics are relatively fast and, in an approximation made here, are removed from the system model. Thus, the control inputs are applied directly to the system through a column of the B matrix.

The engine throttle control is the ¯rst element of the system input so the direction of an engine throttle control fault is the ¯rst column of the B matrix from (d.1)

$$F_{u_\alpha} = \begin{bmatrix} 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 90.91, & 0, & 0 \end{bmatrix}^T \tag{3.6}$$

Fault directions for the brake torque and steering angle are developed in the same way and are given by

$$\begin{aligned} F_{u_{\tau_b}} &= \begin{bmatrix} 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 1.25, & 0 \end{bmatrix}^T \\ F_{u_\beta} &= \begin{bmatrix} 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 80.0 \end{bmatrix}^T \end{aligned}$$

Chapter 4

Fault Detection Filter Design

The fault detection filter design process consists of two steps. First, determine how many fault detection ¯lters are needed and, if more than one, which ¯lters will detect and identify which faults. In a detection ¯lter, the state estimation error in response to a fault in the direction F_i remains in a state subspace $\mathcal{T}_i^*$, an unobservability subspace or detection space. See Appendix A for details. The ability to identify a fault, to distinguish one fault from another, requires for an observable system that the detection spaces be independent. Therefore, the number of faults that can be detected and identi¯ed by a fault detection ¯lter is limited by the size of the state space and the sizes of the detection spaces associated with each of the faults. If the problem considered has more faults than can be accommodated by one fault detection ¯lter, then a bank of ¯lters will have to be constructed. The health monitoring system described in this section for a vehicle in a steady-state constant radius turn, considers ¯fteen system faults: twelve sensor faults and three actuator faults. Since the linearized vehicle models have either fourteen or twelve states, clearly more than one fault detection filter is needed. As with the longitudinal mode system of (Douglas et al. 1995), a bank of four fault detection filters is built.

The second design step is to design the fault detection filters using eigenstructure assignment while making sure that the eigenvectors are not ill-conditioned. The essential feature of a fault detection filter is the detection space structure embedded in the filter dynamics. An eigenvector assignment design algorithm explicitly places eigenvectors to span these subspaces. An eigenvector assignment design algorithm also has to balance the objective of having well-conditioned eigenvectors for robustness against the objective of each fault being highly input observable for fault detection performance. System disturbances, sensor noise and system parameter variations are not considered in the fault detection filter designs described in this report. Note that they are considered in performance evaluation. For such a benign environment, the filter designs are based on spectral considerations only; there is little else that can be used to distinguish a good design from a bad design.

4.1 Fault Detection Filter Configuration

To determine how many and which faults may be included in a fault detection filter design, the detection spaces for each of the faults, also called unobservability subspaces, are formed. A detection space for a fault F_i is denoted by $\mathcal{T}_i^*$. First, the dimensions of the detection spaces are needed. Since the detection spaces are independent subspaces, the sum of their dimensions for any given fault detection filter cannot exceed the dimension of the state-space. Second, the detection spaces for any given fault detection filter are usually output separable and mutually detectable. These concepts are described in detail in Appendix A but briefly, output separability means that the output subspaces $C\mathcal{T}_i^*$ are independent. Mutual detectability means that the sum of the detection spaces $\sum \mathcal{T}_i^*$ is an unobservability subspace. This condition ensures that the spectrum of the detection filter can be assigned arbitrarily.

In practice it is just as easy to find a basis for the detection space as it is to find only the dimension. The method used here is suggested for numerical stability in (Wonham 1985) and is described in Appendix A. Brie°y, for a fault F_i, the approach is to ¯nd the minimal (C, A)-invariant subspace $\mathcal{W}_i^*$ that contains F_i and then to ¯nd the invariant zero directions of the triple (C, A, F_i), if any. With the invariant zero directions are denoted by $\mathcal{V}_i$, the minimal unobservability subspace $\mathcal{T}_i^*$ is given by

$$\mathcal{T}_i^* = \mathcal{W}_i^* + \mathcal{V}_i$$

The linear model of Section 2.3 has either fourteen or twelve states, twelve sensors and three controls. As explained in Section 3, each sensor and each actuator is to be monitored for a fault. It turns out that for all twelve sensor faults and for the steering actuator fault described in Section 3, the detection spaces are given by the fault directions themselves, that is,

$$\mathcal{T}_i^* = \operatorname{Im} F_i$$

For the throttle actuator fault, $CF_{u_\alpha} = 0$, so the detection space for this fault is

$$\mathcal{T}_{u_\alpha}^* = \operatorname{Im}\left[F_{u_\alpha}, AF_{u_\alpha}\right]$$

For the brake actuator fault, $CF_{u_{\tau_b}} \neq 0$ in the reduced-order model used for ¯lter design. However, $CF_{u_{\tau_b}} = 0$ in the full-order model so $F_{u_{\tau_b}}$ is considered to be a very weakly observable direction. The detection space for brake actuator fault is taken to be second-order as for the throttle fault

$$\mathcal{T}_{u_{\tau_b}}^* = \operatorname{Im}\left[F_{u_{\tau_b}}, AF_{u_{\tau_b}}\right]$$

Before designing any fault detection ¯lters, it is useful to determine which faults are output separable. A detection ¯lter designed with faults that are not output separable will generate co-linear residuals and the faults cannot be isolated. Such faults are also considered detection equivalent (Beard 1971). Output separability of two faults F_i and F_j is determined by checking for column independence of realizations for $C\mathcal{T}_i$ and $C\mathcal{T}_j$. Performing this check reveals that the throttle actuator and air mass sensor faults are not output separable because

$$CT^*_{u_\alpha} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad CT^*_{y_{m_a}} = \begin{bmatrix} 1 & 0 \\ 0 & 0.9968 \\ 0 & 0.0597 \\ 0 & -0.0001 \\ 0 & -0.0016 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0.0404 \\ 0 & 0.0340 \end{bmatrix}$$

Since $CF_{y_{m_a}} = CAF_{u_\alpha}$, the throttle actuator and air mass sensor faults would not normally be part of a single fault detection ¯lter design. However, it is possible to include both in one ¯lter if the sensor fault is approximated as a one-dimensional fault. As explained in Section 3.1, the direction of the sensor fault magnitude is $AF_{y_{m_a}}$ while the direction of the fault rate is $F_{y_{m_a}}$. The throttle actuator and air mass sensor faults become output separable if only the sensor fault magnitude direction is used. This design decision could allow a noisy but zero mean sensor fault to remain undetected. However, a throttle actuator fault could never stimulate the air mass sensor fault residual. Also, since the throttle fault detection space is spanned by F_{u_α} and AF_{u_α}, an air mass sensor fault rate will stimulate the throttle fault residual. Finally, as long as an air mass sensor fault spectral components are low frequency, the two faults should be detectable and isolated.

Another consideration in grouping the faults among the fault detection ¯lters is to group faults which are robust to system nonlinearities. Note that an actuator fault changes the vehicle operating point possibly introducing nonlinear e®ects into all measurements. The nonlinear e®ect is small if the residual response is small compared to that for some nominal fault. Also, sensor faults that are open-loop are easily isolated since they do not stimulate any dynamics. One approach to fault grouping is to always group actuator and sensor faults with di®erent fault detection ¯lters.

Finally, usually an attempt is made to group as many faults as possible in each ¯lter.

When full-order ¯lters are used, this approach minimizes the number of ¯lters needed. When reduced-order ¯lters are used, this approach minimizes the order of each complementary space and, therefore, the order of each reduced-order ¯lter. Note that each fault included in a fault detection ¯lter design imposes more constraints on the ¯lter eigenvectors. Sometimes, the objective of obtaining well-conditioned ¯lter eigenvectors imposes a tradeo® between robustness and the reduced-order ¯lter size.

Given the above considerations, fault detection ¯lters are designed for the following groups of faults:

Fault detection ¯lter 1.

$F_{y_{\omega_e}}$: Engine speed sensor.

$F_{y_{\ddot{y}}}$: Lateral acceleration sensor.

$F_{y_{\ddot{z}}}$: Vertical acceleration sensor.

$F_{y_{\dot{\theta}}}$: Pitch rate sensor.

Fault detection ¯lter 2.

$F_{y_{\ddot{x}}}$: Longitudinal acceleration sensor.

$F_{y_{\dot{\phi}}}$: Roll rate sensor.

$F_{y_{\dot{\epsilon}}}$: Yaw rate sensor.

$F_{y_{\omega_e}}$: Engine speed sensor.

Fault detection ¯lter 3.

$F_{y_{\omega_{fl}}}$: Front left wheel speed sensor.

$F_{y_{\omega_{fr}}}$: Front right wheel speed sensor.

$F_{y_{\omega_{rl}}}$: Rear left wheel speed sensor.

$F_{y_{\omega_{rr}}}$: Rear right wheel speed sensor.

Fault detection filter 4.

$$\begin{aligned} F_{u_\alpha} &: \text{ Throttle angle actuator.} \\ F_{u_{\tau_b}} &: \text{ Brake torque actuator.} \\ F_{u_\beta} &: \text{ Steering angle actuator.} \\ F_{y_{m_a}} &: \text{ Manifold air mass sensor.} \end{aligned}$$

Showing that the fault sets are mutually detectable involves calculating invariant zeros of each triple $(C, A, F_1), \ldots, (C, A, F_q)$ and then showing that these are the same invariant zeros as of the triple $(C, A, [F_1, \ldots, F_q])$. For example, for the first fault detection filter, define the sets of invariant zeros

$$\begin{aligned} \Omega_{y_{\omega_e}} &= \Omega(C, A, F_{y_{\omega_e}}) \\ \Omega_{y_{\ddot{y}}} &= \Omega(C, A, F_{y_{\ddot{y}}}) \\ \Omega_{y_{\ddot{z}}} &= \Omega(C, A, F_{y_{\ddot{z}}}) \\ \Omega_{y_{\dot{\theta}}} &= \Omega(C, A, F_{y_{\dot{\theta}}}) \\ \Omega_{y} &= \Omega(C, A, [F_{y_{\omega_e}}, F_{y_{\ddot{y}}}, F_{y_{\ddot{z}}}, F_{y_{\dot{\theta}}}]) \end{aligned}$$

where $\Omega(C, A, F_i)$ means the set of invariant zeros of the triple (C, A, F_i). The first fault detection filter is mutually detectable because

$$\Omega_y = \Omega_{y_{\omega_e}} + \Omega_{y_{\ddot{y}}} + \Omega_{y_{\ddot{z}}} + \Omega_{y_{\dot{\theta}}}$$

4.2 Eigenstructure Placement

The fault detection filters are found using a left eigenvector assignment algorithm described in Appendix B. Since the calculations are somewhat long and they are the same for each detection filter, the calculation details are given for only the actuator fault detection filter and one of the sensor fault detection filters. Algorithm B.1 is applied to the design of fault detection filters for the third fault group, which has the four wheel speed sensors, and the fourth fault group, which has the throttle actuator, the brake actuator, the steering actuator and the manifold air mass sensor.

4.2.1 Sensor Fault Design

This section presents the details of a fault detection ¯lter design for fault group three, the four wheel speed sensors. The twelve state reduced-order linear model derived in Section 2.3 is used. The ¯rst step is to ¯nd the dimension of each detection space. This was discussed in Section 4.1 where it was shown that the detection spaces are given by the fault directions themselves, that is, $\mathcal{T}_i^* = \operatorname{Im} F_i$. The fault directions assigned to the third fault detection ¯lter are all sensor faults and all have dimension two

$$\begin{aligned} \nu_{y_{w_{fl}}} &= \dim \mathcal{T}^*_{y_{w_{fl}}} = 2 \\ \nu_{y_{w_{fr}}} &= \dim \mathcal{T}^*_{y_{w_{fr}}} = 2 \\ \nu_{y_{w_{rl}}} &= \dim \mathcal{T}^*_{y_{w_{rl}}} = 2 \\ \nu_{y_{w_{rr}}} &= \dim \mathcal{T}^*_{y_{w_{rr}}} = 2 \end{aligned}$$

The dimension of the fault detection ¯lter complementary space $\mathcal{T}_0$ is also needed. The complementary space is any subspace independent of the detection spaces that completes the state-space. Thus, for the ¯rst fault detection ¯lter

$$\mathcal{X} = \mathcal{T}^*_{y_{w_{fl}}} \oplus \mathcal{T}^*_{y_{w_{fr}}} \oplus \mathcal{T}^*_{y_{w_{rl}}} \oplus \mathcal{T}^*_{y_{w_{rr}}} \oplus \mathcal{T}_0$$

and the dimension of $\mathcal{T}_0$ is four

$$\begin{aligned} \nu_0 &= n - \nu_{y_{w_{fl}}} - \nu_{y_{w_{fr}}} - \nu_{y_{w_{rl}}} - \nu_{y_{w_{rr}}} \\ &= 12 - 2 - 2 - 2 - 2 \\ &= 4 \end{aligned}$$

Next de¯ne the complementary fault sets. There are four faults $F_{y_{w_{fl}}}$, $F_{y_{w_{fr}}}$, $F_{y_{w_{rl}}}$ and $F_{y_{w_{rr}}}$ so there are ¯ve complementary fault sets which are:

$$\hat{F}_{y_{w_{fl}}} = \left[F_{y_{w_{fr}}}, F_{y_{w_{rl}}}, F_{y_{w_{rr}}} \right] \tag{4.1a}$$

$$\hat{F}_{y_{w_{fr}}} = \left[F_{y_{w_{fl}}}, F_{y_{w_{rl}}}, F_{y_{w_{rr}}} \right] \tag{4.1b}$$

$$\hat{F}_{y_{w_{rl}}} = \left[F_{y_{w_{fl}}}, F_{y_{w_{fr}}}, F_{y_{w_{rr}}} \right] \tag{4.1c}$$

$$\hat{F}_{y_{w_{rr}}} = \left[F_{y_{w_{fl}}}, F_{y_{w_{fr}}}, F_{y_{w_{rl}}}\right] \tag{4.1d}$$

$$\hat{F}_0 = \left[F_{y_{w_{fl}}}, F_{y_{w_{fr}}}, F_{y_{w_{rl}}}, F_{y_{w_{rr}}}\right] \tag{4.1e}$$

Now choose the fault detection filter closed-loop eigenvalues. Since the system model includes no sensor noise, no disturbances and no parameter variations, there is little basis for preferring one set of detection filter closed-loop eigenvalues over another. The poles are chosen here to give a reasonable response time but are not unrealistically fast. The assigned eigenvalues are

$$\begin{aligned}
\Lambda_{y_{w_{fl}}} &= \{-3, -10\} \\
\Lambda_{y_{w_{fr}}} &= \{-4, -9\} \\
\Lambda_{y_{w_{rl}}} &= \{-5, -8\} \\
\Lambda_{y_{w_{rr}}} &= \{-6, -7\} \\
\Lambda_0 &= \{-11, -12, -13, -14\}
\end{aligned}$$

The next step is to find the closed-loop fault detection filter left eigenvectors. For each eigenvalue $\lambda_{i_j} \in \Lambda_i$, the left eigenvectors v_{i_j} generally are not unique and must be chosen from a subspace as $v_{i_j} \in V_{i_j}$ where V_{i_j} and another space W_{i_j} are found by solving

$$\begin{bmatrix} A^T - \lambda_{i_j} I & C^T \\ \hat{F}_i^T & 0 \end{bmatrix} \begin{bmatrix} V_{i_j} \\ W_{i_j} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{4.2}$$

There are twelve V_{i_j} associated with twelve eigenvalues. Only two V_{i_j}, the two associated with the front left wheel speed sensor fault, are shown here because this intermediate result is easily reproduced. They are shown in Appendix E. As explained in Appendix B and (Douglas and Speyer 1995b), to help desensitize the fault detection filter to parameter variations, the left eigenvectors are chosen from $v_{i_j} \in V_{i_j}$ as the set with the greatest degree of linear independence. The degree of linear independence is indicated by the smallest singular value of the matrix formed by the left eigenvectors. Upper bounds on the singular values of the left eigenvectors are given by the singular values of

$$V = [V_{0_1}, V_{0_2}, V_{0_3}, V_{0_4}, V_{y_{w_{fl1}}}, V_{y_{w_{fl2}}}, V_{y_{w_{fr1}}}, V_{y_{w_{fr2}}}, V_{y_{w_{rl1}}}, V_{y_{w_{rl2}}}, V_{y_{w_{rr1}}}, V_{y_{w_{rr2}}}]$$

These singular values are

$$\sigma(V) = \{3.4641,\ 3.4641,\ 3.4641,\ 3.4641,\ 2.5763,\ 2.0626,\ 1.9404,\ 1.1563,\ 0.0627,\ 0.0431,\ 0.0099,\ 0.0014\} \tag{4.3}$$

If the left eigenvector singular value upper bounds were small, then all possible combinations of detection filter left eigenvectors would be ill-conditioned and the filter eigenstructure would be sensitive to small parameter variations. Since (4.3) indicates that the upper bounds are not small, continue by looking for a set of fault detection filter left eigenvectors that are reasonably well-conditioned. For this case, one possible set of left eigenvectors from the set V nearly meets the upper bound, is well-conditioned and is given in Appendix E. The singular values of this set of detection filter left eigenvectors are

$$\sigma(\hat{V}) = \{1.82,\ 1.46,\ 1.37,\ 1.00,\ 1.00,\ 1.00, 1.00,\ 0.818,\ 0.0443,\ 0.0305,\ 0.0070,\ 0.0010\}$$

Since the difference between the largest and the smallest singular values is only three orders of magnitude, the detection filter gain will be reasonably small and the filter eigenstructure should not be sensitive to small parameter variations.

The fault detection filter gain L is found by solving

$$\hat{V}^T L = \hat{W}^T \tag{4.4}$$

where $\hat{V}$ is the matrix of left eigenvectors as found above, and $\hat{W}$ is a matrix of vectors w_{i_j} which satisfy (b.10)

$$\begin{bmatrix} A^T - \lambda_{i_j} I & C^T \\ \hat{F}_i^T & 0 \end{bmatrix} \begin{bmatrix} v_{i_j} \\ w_{i_j} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

If the left eigenvector v_{i_j} is a linear combination of the columns of V_{i_j}, w_{i_j} is the same linear combination of the columns of W_{i_j} where V_{i_j} and W_{i_j} are from (4.2). The $\hat{W}$ matrix is given in Appendix E. The detection filter gain is found from (4.4) and is also given in Appendix E.

To complete the detection filter design, output projection matrices $\hat{H}_{y_{w_{fl}}}$, $\hat{H}_{y_{w_{fr}}}$, $\hat{H}_{y_{w_{rl}}}$ and $\hat{H}_{y_{w_{rr}}}$ are needed to project the residual along the respective output subspaces $C\hat{T}^*_{y_{w_{fl}}}$, $C\hat{T}^*_{y_{w_{fr}}}$, $C\hat{T}^*_{y_{w_{rl}}}$ and $C\hat{T}^*_{y_{w_{rr}}}$. What this means is that, for example, $\hat{T}^*_{y_{w_{fl}}}$ becomes the unobservable subspace of the pair $(\hat{H}_{y_{w_{fl}}} C, A + LC)$. Remember that by the definition of the complementary faults (4.1), faults $F_{y_{w_{fr}}}$, $F_{y_{w_{rl}}}$ and $F_{y_{w_{rr}}}$ lie in $\hat{\mathcal{T}}^*_{y_{w_{fl}}}$ and fault $F_{y_{w_{fl}}}$ does not. The effect is that the projected residual is driven by fault $F_{y_{w_{fl}}}$ and only fault $F_{y_{w_{fl}}}$ as shown in Figure 4.3.

A projection $\hat{H}_i$ is computed by first finding a basis for the range space of $C\hat{T}^*_i$ where again, $\hat{T}^*_i$ is any basis for the detection space $\hat{\mathcal{T}}^*_i$. This is done by finding the left singular vectors of $C\hat{T}^*_i$. Denote this basis for now as h_i. Then $\hat{H}_i$ is given by

$$\hat{H}_i = I - h_i h_i^T$$

An output projection for the front left wheel speed sensor is given in (e.2) of Appendix E.

In summary, a fault detection filter for the system with sensor faults $E_{y_{w_{fl}}}$, $E_{y_{w_{fr}}}$, $E_{y_{w_{rl}}}$ and $E_{y_{w_{rr}}}$ as in (3.3)

$$\begin{aligned}
\dot{x} &= Ax + Bu \\
y &= Cx + Du + E_{y_{w_{fl}}}\mu_{y_{w_{fl}}} + E_{y_{w_{fr}}}\mu_{y_{w_{fr}}} + E_{y_{w_{rl}}}\mu_{y_{w_{rl}}} + E_{y_{w_{rr}}}\mu_{y_{w_{rr}}}
\end{aligned}$$

is equivalent to a fault detection filter for the system with faults $F_{y_{w_{fl}}}$, $F_{y_{w_{fr}}}$, $F_{y_{w_{rl}}}$ and $F_{y_{w_{rr}}}$ as in (3.5)

$$\begin{aligned}
\dot{x} &= Ax + Bu + F_{y_{y_{fl}}} m_{y_{w_{fl}}} + F_{y_{w_{fr}}} m_{y_{w_{fr}}} + F_{y_{w_{rl}}} m_{y_{w_{rl}}} + F_{y_{w_{rr}}} m_{y_{w_{rr}}} \\
y &= Cx + Du
\end{aligned}$$

and has the form

$$\begin{aligned}
\dot{\hat{x}} &= (A + LC)\hat{x} + (B + LD)u - Ly \\
z_{y_{w_{fl}}} &= \hat{H}_{y_{w_{fl}}}(C\hat{x} + Du - y) \\
z_{y_{w_{fr}}} &= \hat{H}_{y_{w_{fr}}}(C\hat{x} + Du - y) \\
z_{y_{w_{rl}}} &= \hat{H}_{y_{w_{rl}}}(C\hat{x} + Du - y) \\
z_{y_{w_{rr}}} &= \hat{H}_{y_{w_{rr}}}(C\hat{x} + Du - y)
\end{aligned}$$

with L and the $\hat{H}_{y_{w_{fl}}}$, $\hat{H}_{y_{w_{fr}}}$, $\hat{H}_{y_{w_{rl}}}$ and $\hat{H}_{y_{w_{rr}}}$ given by (e.1) and (e.2). Calculations for the detection filters for the other two sensor fault groups 1 and 2 are carried out in the same way and are not shown here.

Figures 4.1, 4.2 and 4.3 show the singular value frequency responses of fault detection filters for fault groups one, two and three, the sensor fault groups. The frequency responses are from all faults for which the filter has been designed to each of the filter residuals. The singular values show that each residual only responds to the fault it was designed to detect when no noise or parametric uncertainties are present.

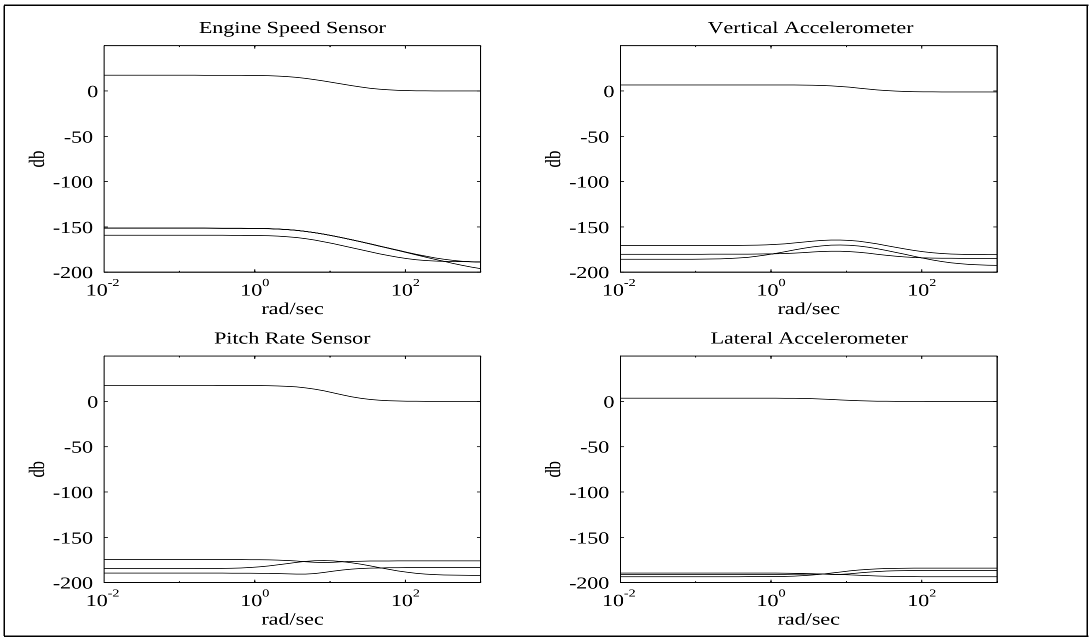

Figure 4.1: Singular value frequency response from all faults to residuals of fault detection filter one.

4.2.2 Actuator Fault Design

This section presents the details of a fault detection filter design for fault group four. The fault directions assigned to fault group four are the throttle actuator, the brake actuator, the steering actuator and the manifold air mass sensor faults. The fourteen state reduced-order linear model derived in Section 2.3 is used.

The design procedure is similar to the previous section but does have a twist. As discussed in Section 4.1, a reduced-order air mass sensor fault is used to achieve output separability with the throttle actuator fault. The dimension of each detection space was

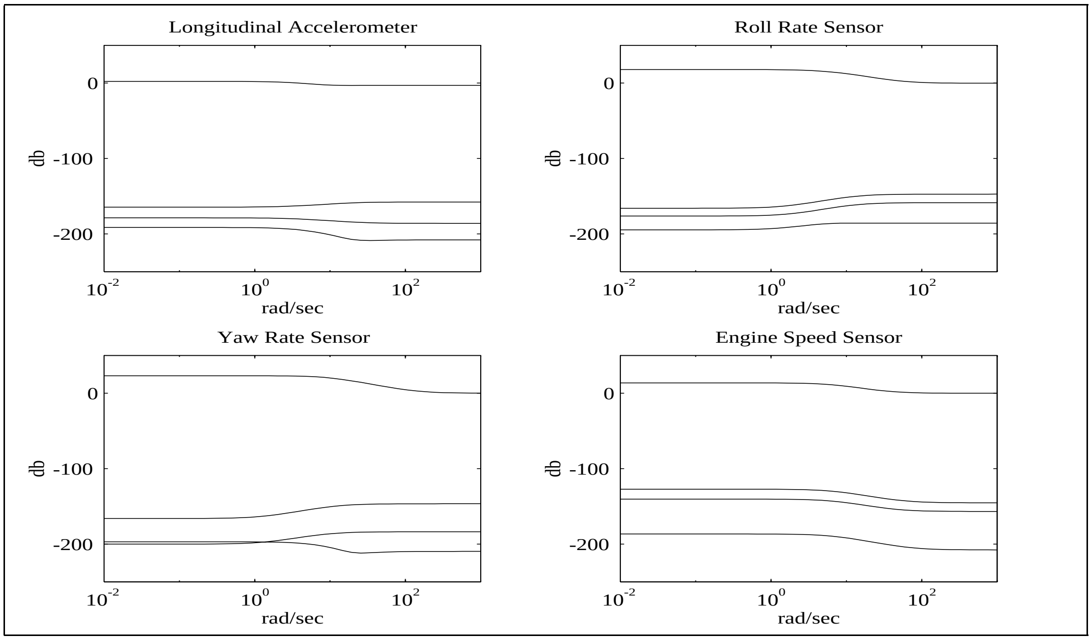

Figure 4.2: Singular value frequency response from all faults to residuals of fault detection ¯lter two.

found in Section 4.1 as

$$\begin{aligned}
\nu_{u_\alpha} &= \dim \mathcal{T}^*_{u_\alpha} = 2 \\
\nu_{u_{\tau_b}} &= \dim \mathcal{T}^*_{u_{\tau_b}} = 2 \\
\nu_{u_{beta}} &= \dim \mathcal{T}^*_{u_\beta} = 1 \\
\nu_{y_{m_a}} &= \dim \mathcal{T}^*_{y_{m_a}} = 1
\end{aligned}$$

and the dimension of the fault detection ¯lter complementary space $\mathcal{T}_0$ where

$$\mathcal{X} = \mathcal{T}^*_{u_\alpha} \oplus \mathcal{T}^*_{u_{\tau_b}} \oplus \mathcal{T}^*_{u_\beta} \oplus \mathcal{T}^*_{y_{m_a}} \oplus \mathcal{T}_0$$

is eight

$$\begin{aligned}
\nu_0 &= n - \nu_{u_\alpha} - \nu_{u_{\tau_b}} - \nu_{u_\beta} - \nu_{y_{m_a}} \\
&= 14 - 2 - 2 - 1 - 1 \\
&= 8
\end{aligned}$$

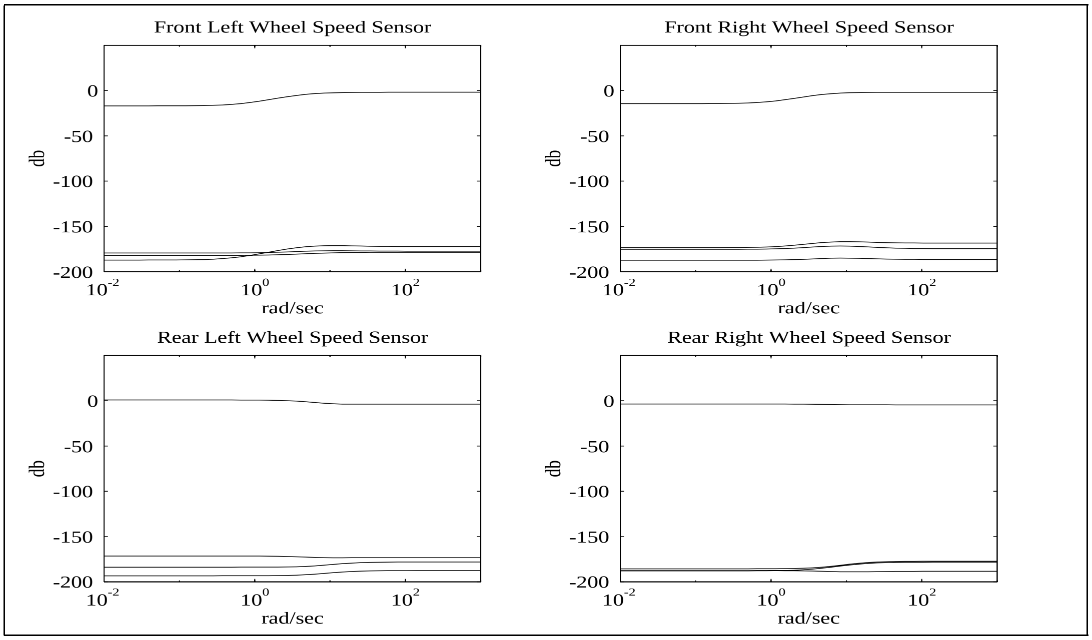

Figure 4.3: Singular value frequency response from all faults to residuals of fault detection filter three.

Next define the complementary faults sets. There are four faults F_{u_α}, $F_{u_{\tau_b}}$, F_{u_β} and $F_{y_{m_a}}$ so there are five complementary fault sets which are:

$$\hat{F}_{u_\alpha} = \left[F_{u_{\tau_b}}, F_{u_\beta}, F_{y_{m_a}}\right] \tag{4.5a}$$

$$\hat{F}_{u_{\tau_b}} = \left[F_{u_\alpha}, F_{u_\beta}, F_{y_{m_a}}\right] \tag{4.5b}$$

$$\hat{F}_{u_\beta} = \left[F_{u_\alpha}, F_{u_{\tau_b}}, F_{y_{m_a}}\right] \tag{4.5c}$$

$$\hat{F}_{y_{m_a}} = \left[F_{u_\alpha}, F_{u_{\tau_b}}, F_{u_\beta}\right] \tag{4.5d}$$

$$\hat{F}_0 = \left[F_{u_\alpha}, F_{u_{\tau_b}}, F_{u_\beta}, F_{y_{m_a}}\right] \tag{4.5e}$$

Now choose the fault detection filter closed-loop eigenvalues.

$$\Lambda_{u_\alpha} = \{-4, -9\}$$

$$\Lambda_{u_{\tau_b}} = \{-5, -8\}$$

$$\Lambda_{u_\beta} = \{-6\}$$

$$\Lambda_{y_{m_a}} = \{-7\}$$

$$\Lambda_0 = \{-10, -11, -12, -13, -14, -15, -16, -17\}$$

The next step is to find the closed-loop fault detection filter left eigenvectors. As in Section 4.2.1, the left eigenvectors v_{i_j} for each eigenvalue $\lambda_{i_j} \in \Lambda_i$ generally are not unique and must be chosen from a subspace as $v_{i_j} \in V_{i_j}$ where V_{i_j} is found by solving

$$\begin{bmatrix} A^T - \lambda_{i_j} I & C^T \\ \hat{F}_i^T & 0 \end{bmatrix} \begin{bmatrix} V_{i_j} \\ W_{i_j} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{4.6}$$

There are fourteen V_{i_j} associated with fourteen eigenvalues. Upper bounds on the singular values of the left eigenvectors are given by the singular values of

$$V = [V_{u_{\tau_{b1}}}, V_{u_{\tau_{b2}}}, V_{0_1}, V_{0_2}, V_{0_3}, V_{0_4}, V_{0_5}, V_{0_6}, V_{0_7}, V_{0_8}, V_{u_{\alpha_1}}, V_{u_{\alpha_2}}, V_{u_\beta}, V_{y_{m_a}}] \tag{4.7}$$

These singular values are

$$\begin{aligned} \sigma(V) = \{&3.74,\ 3.74,\ 3.74,\ 3.74,\ 3.74,\ 3.74,\ 3.71, \\ &2.19,\ 1.65,\ 0.734,\ 0.466,\ 0.0918,\ 0.0272,\ 0.0005\} \end{aligned} \tag{4.8}$$

Since (4.8) indicates that the upper bounds are not small, continue by looking for a set of fault detection filter left eigenvectors that are reasonably well-conditioned. One possible choice is, given in Appendix E, has the following singular values

$$\begin{aligned} \sigma(\bar{V}) = \{&1.73,\ 1.47,\ 1.39,\ 1.34,\ 1.02,\ 1.00,\ 1.00, \\ &1.00,\ 0.955,\ 0.350,\ 0.117,\ 0.0073,\ 0.0026,\ 0.0005\} \end{aligned}$$

Since these singular values are quite close to their respective upper bounds, the detection filter gain should not be large and the filter eigenstructure should not be sensitive to small parameter variations. As in Section 4.2.1, the fault detection filter gain L is found by solving

$$\bar{V}^T L = \bar{W}^T \tag{4.9}$$

where the columns of $\bar{V}$ and $\bar{W}$ are found from (4.6). Both $\bar{W}$ and L are given in Appendix E. Output projection matrices $\hat{H}_{u_\alpha}$, $\hat{H}_{u_{\tau_b}}$, $\hat{H}_{u_\beta}$ and $\hat{H}_{y_{m_a}}$ are needed to complete the fault detection ¯lter design These are found in the same way as for the sensor fault example of Section 4.2.1 and are given in Appendix E.

A note should be made regarding the throttle actuator fault residual. By the de¯nition of the complementary faults (4.5), $F_{u_{\tau_b}}$, F_{u_β} and $F_{y_{m_a}}$ lie in $\hat{T}^*_{u_\alpha}$ while F_{u_α} does not. The e®ect is that the projected residual is not driven by fault $F_{u_{\tau_b}}$, F_{u_β} or $F_{y_{m_a}}$. Now recall that $F_{y_{m_a}}$ is a reduced-order approximation for $E_{y_{m_a}}$ so the throttle actuator residual is not only driven by F_{u_α}, but also the part of $E_{y_{m_a}}$ not modeled by $F_{y_{m_a}}$. As shown in Figure 4.4, the throttle actuator residual can only isolate faults well at low frequency while other residuals isolate all faults.

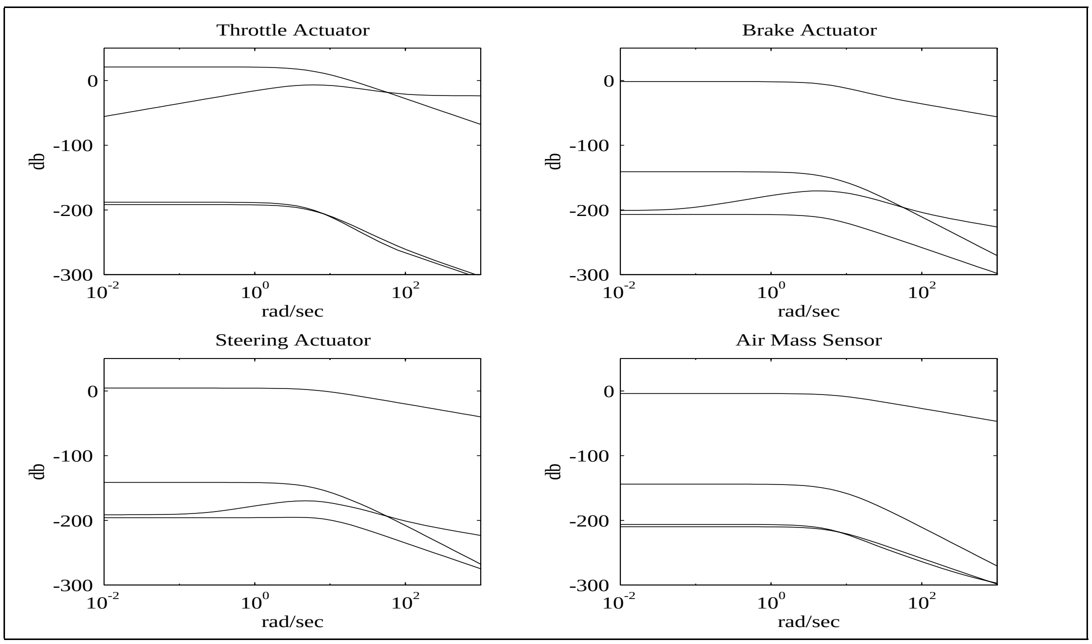

Figure 4.4: Singular value frequency response from all faults to residuals of fault detection ¯lter four.

Chapter 5

Fault Detection Filter Evaluation

Fault detection filter performance is evaluated using the nonlinear simulation discussed in Section 2.1. The fault detection ¯lters designed in Sections 4.2.1 and 4.2.2 are tested on smooth and rough roads. Performance is evaluated with respect to robustness to model nonlinearities and road noise. The performance of a longitudinal mode fault detection ¯lter described in (Douglas et al. 1995) is also evaluated.

5.1 Fault Detection Filter Evaluation On A Curved Road

Fault detection ¯lter performance is evaluated using the nonlinear vehicle simulation of Section 2.1. Sensor fault detection performance is evaluated by introducing a sensor bias into the data provided by the nonlinear simulation. In the most benign test, the nonlinear vehicle simulation is run in a steady state turn with $24.87\frac{\mathrm{m}}{\mathrm{sec}}$ forward speed while a bias is added to one of the sensor outputs. The turn is achieved using a 0.005 rad. steering angle. In this test, the operating point is the same as that used to derive the linearized dynamics for the fault detection ¯lter design. Furthermore, the vehicle dynamics are not stimulated resulting in data that is essentially linear. Thus, the fault detection ¯lter is operating in a nominal environment and the test does not provide much useful information. The results of these tests are not shown here.

In a more useful test, the ¯lters operate at an o®-nominal condition, that is, the vehicle operates in a steady state condition but not the same one used to generate the linearized dynamics. These tests are discussed in Section 5.1.1. Dynamic disturbances are introduced by simulating a rough road surface as in Section 2.1.2. Fault detection ¯lter testing in the presence of dynamic disturbances is discussed in Section 5.1.2.

5.1.1 Evaluation On Smooth Road

In this section, the fault detection ¯lters of Section 4.2 are tested at an o®-nominal operating point, that is, the vehicle operates in a steady state condition but not the same one used to generate the linearized dynamics. This is achieved by increasing the throttle two degrees from the nominal value causing the steady state vehicle speed to be about two meters per second faster than the nominal. The road is °at and smooth so only vehicle nonlinearities corrupt the ¯lter residuals. If the vehicle dynamics were linear, the increased throttle setting would have only a transient e®ect, if any, on the linear fault detection ¯lter state estimates. The state estimate errors and the ¯lter residuals would asymptotically go to zero. Since the vehicle dynamics are not linear and the vehicle operating condition is not the same as it would be if the dynamics were linear, the ¯lter state estimates and the residuals are not zero.

Since most residuals are not zero, as is to be expected, the natural question to ask is what magnitude residual should be considered small. The answer lies in comparing the size of a nonzero residual due to non-linearities and the size of a nonzero residual due to a fault. A residual scaling factor is chosen such that when a fault is introduced into the *linearized* dynamics the magnitude of the corresponding reduced-order fault detection ¯lter residual is one. Since all residuals generated by the o®-nominal operating condition have magnitude less than 0.25, they should not be easily mistaken for residuals generated by a fault.

Of course, the size of the residual is proportional to the size of the fault. The size of the fault used for ¯nding the residual scaling factors is determined as follows. For most sensors, the size of the fault is given by the di®erence in magnitude between the sensor output at the nominal and o®-nominal steady state operating conditions. For some sensors, such as the accelerometers and the angular rate sensors, the output is zero in any steady state condition and another method has to be used. For the longitudinal accelerometer, the size of the fault is given as the largest transient value of the sensor output while a two-degree step throttle command takes the vehicle from the nominal to o®-nominal condition. For the lateral and vertical accelerometers, even the transient is small during an acceleration maneuver. Thus the same nominal fault value used for longitudinal acceleration fault is also used for the lateral and heave accelerometers. The pitch, roll and yaw rate sensors are treated the same way as the lateral and heave accelerometers. The value $0.02\frac{\mathrm{rad}}{\mathrm{sec}}$ is chosen as a value for vehicle rotation rates reasonably encountered during normal vehicle operation.

Figure 5.1 shows the magnitudes of the residuals for the four fault detection ¯lters derived from the ¯rst fault design group: the engine speed sensor, lateral and vertical accelerometers and pitch rate sensor. A sensor bias fault is added after two seconds when ¯lter initialization errors have died out. Only one sensor fault is added at a time; simultaneous faults are not allowed. It is important to note that when any of the sensor faults from the ¯rst fault design group occur, the residuals associated with a fault detection ¯lter designed for other faults have no meaning. This is why only four residuals are shown in each plot of Figures 5.1, 5.2, 5.3, 5.4 and 5.5 while sixteen residuals are generated by the entire fault detection system. Distinguishing a meaningful residual from a non-meaningful residual is left to the residual processing system described in sections 6 and 7. The residual associated with the fault quickly approaches one and other residuals *in the fault group* remain una®ected.

Figures 5.2 and 5.3 show the residuals for the four fault detection ¯lters derived from the second and third sensor fault design groups. Residual scaling factors are chosen in the same way as for the ¯rst fault design group. The fault detection ¯lter performance indicated by

Figures 5.2 and 5.3 is the same as that indicated by Figure 5.1.

The performance of the filter for the fourth fault group which includes actuator faults is shown in Figure 5.4. A throttle fault is simulated by sending a two-degree step throttle command to the nonlinear simulation but not to the fault detection filter. Even though a throttle fault stimulates the vehicle nonlinear dynamics and the residual associated with other faults, Figure 5.4 shows that both positive and negative throttle faults are clearly identifiable from other faults. A brake fault is simulated by applying a brake torque just large enough to slow the vehicle from $25\frac{\mathrm{m}}{\mathrm{sec}}$ to $21\frac{\mathrm{m}}{\mathrm{sec}}$. This changes the vehicle steady state operating point by the same amount as a minus four degree throttle fault. Figure 5.4 shows that the brake fault is clearly identified. A steering fault is simulated by a 0.001 rad. steering angle bias. Recall that the nominal turn is achieved with a 0.005 rad. steering angle. Figure 5.4 shows that the steering fault is clearly identified.

An interesting observation of the throttle actuator residual behavior follows from the discussion of Section 4.1 and is illustrated in Figure 5.5. Since one direction of the throttle actuator fault corresponds to the air mass sensor fault rate, a bias fault in the air mass sensor causes a response in the throttle actuator residual. Since the throttle actuator residual only responds to air mass sensor fault rate, the residual response is transient and dies out quickly. There should be no problem distinguishing throttle actuator and air mass sensor faults as long as the air mass sensor fault only has low frequency components.

5.1.2 Evaluation On Rough Road

Tests performed on the fault detection filters in this section closely follow those of the last section except that the road is no longer smooth. The same types and sizes of faults are used here as in Section 5.1.1

It has already been demonstrated that when no road noise is present, filter residuals not associated with a given sensor fault do not respond when that fault occurs. Therefore, only residuals associated with a fault are shown in the plots. For comparison, the residuals for the no fault case are also given.

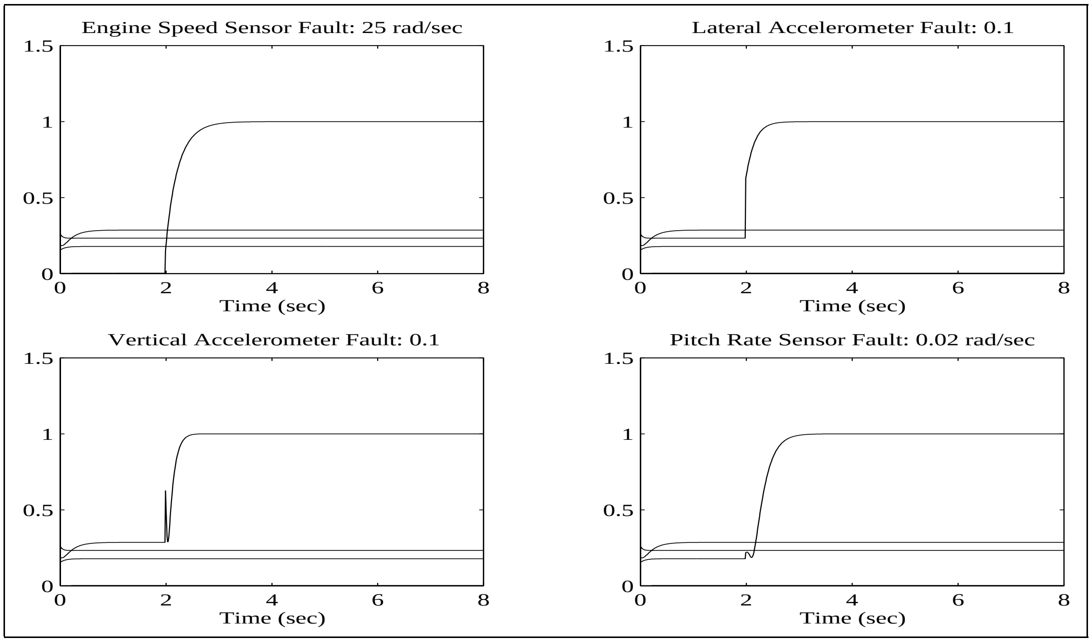

Figure 5.1: Residuals for fault detection ¯lter one.

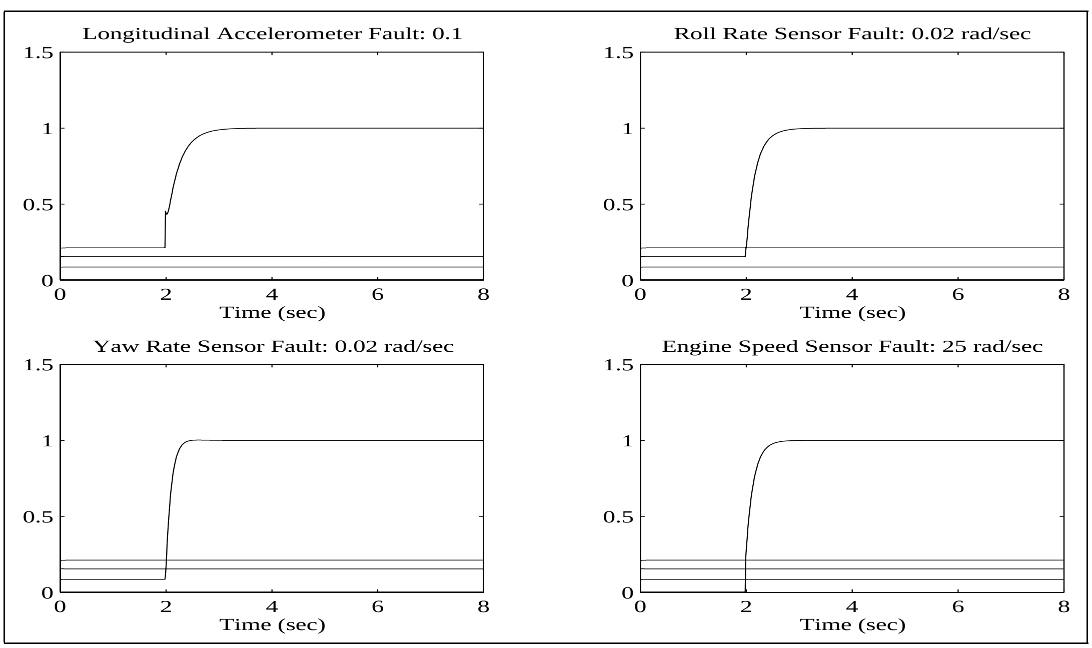

Figure 5.2: Residuals for fault detection ¯lter two.

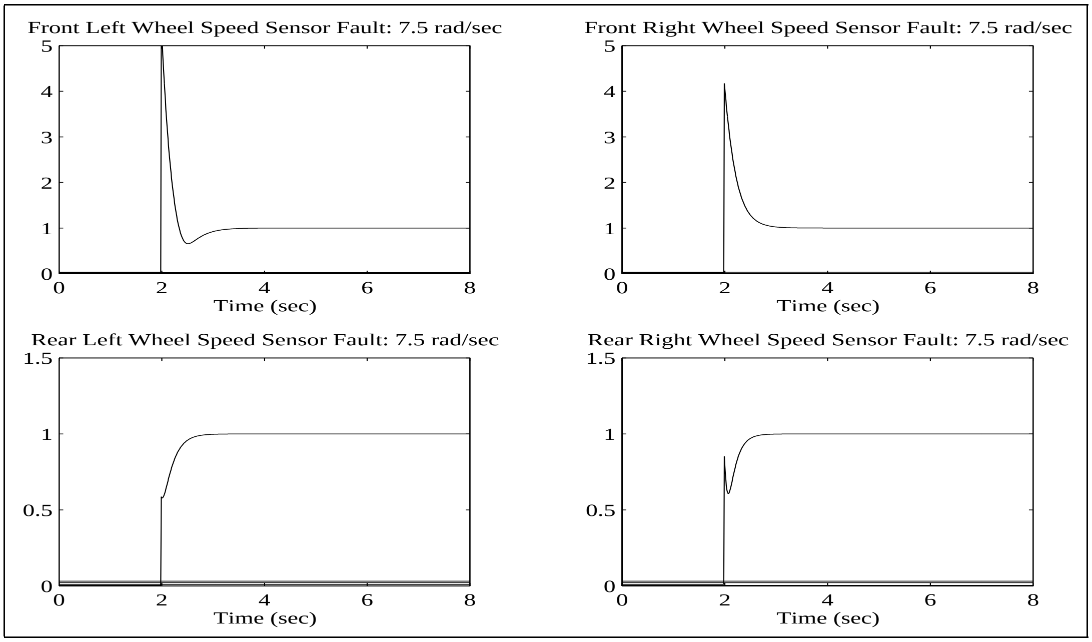

Figure 5.3: Residuals for fault detection ¯lter three.

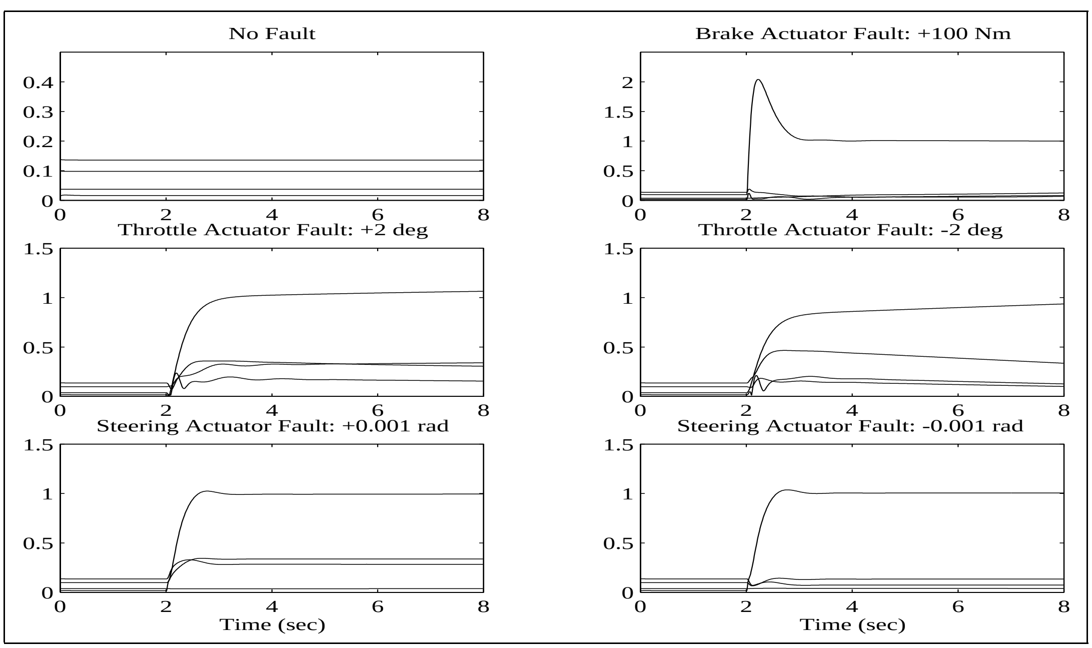

Figure 5.4: Residuals for fault detection ¯lter four.

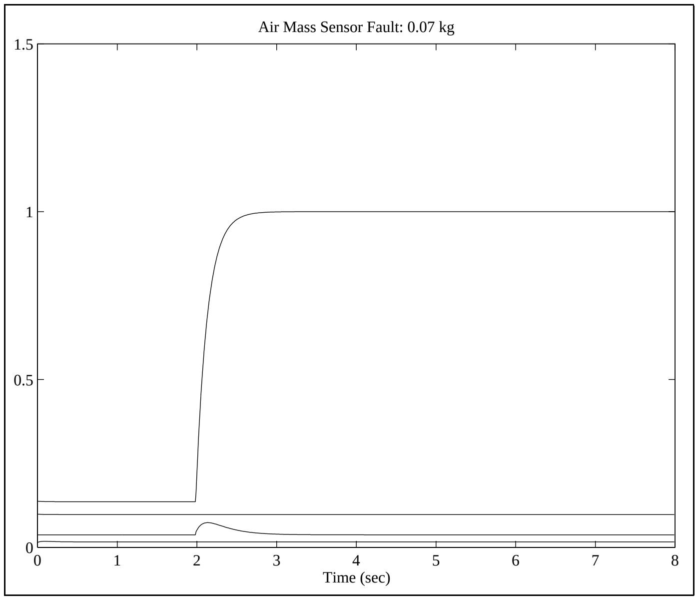

Figure 5.5: Residuals for fault detection ¯lter four.

Figures 5.6 and 5.7 show the residuals for the four fault detection ¯lters derived from the ¯rst fault group. Figure 5.6 illustrates a visually obvious contrast between cases where no fault occurs and where a step fault does occur in the engine speed sensor and lateral accelerometer residuals. In Figure 5.7, bias faults in either the pitch rate sensor or the vertical accelerometer are only barely visually detectable. The reason is the the nominal bias fault size is dominated by the noise produced by the rough road model. In the case of the vertical accelerometer, the noise standard deviation is about $0.3\frac{\mathrm{m}}{\mathrm{sec}^2}$ while the nominal bias fault size is $0.1\frac{\mathrm{m}}{\mathrm{sec}^2}$. While the fault may not be visually detectable, both residual processing systems, the Bayesian neural network of Section 6 and the Shiryayev sequential probability ratio test of Section 7, quickly and unambiguously detect the fault.

Figures 5.8 and 5.9 show the the residuals for the four fault detection ¯lters derived from the second sensor fault group. Figures 5.10 and 5.11 show the the residuals for the four reduced-order fault detection ¯lters derived from the third sensor fault group.

Analysis is more di±cult for the residuals produced by the fault group four detection ¯lter. The actuator faults in this group stimulate the nonlinear vehicle dynamics, alter the operating point and cause all residuals to respond, not just the residual associated with given fault. Thus all residuals are examined as an actuator fault occurs. Figures 5.12 through 5.18 show that all faults are clearly identi¯able and distinguishable from one another.

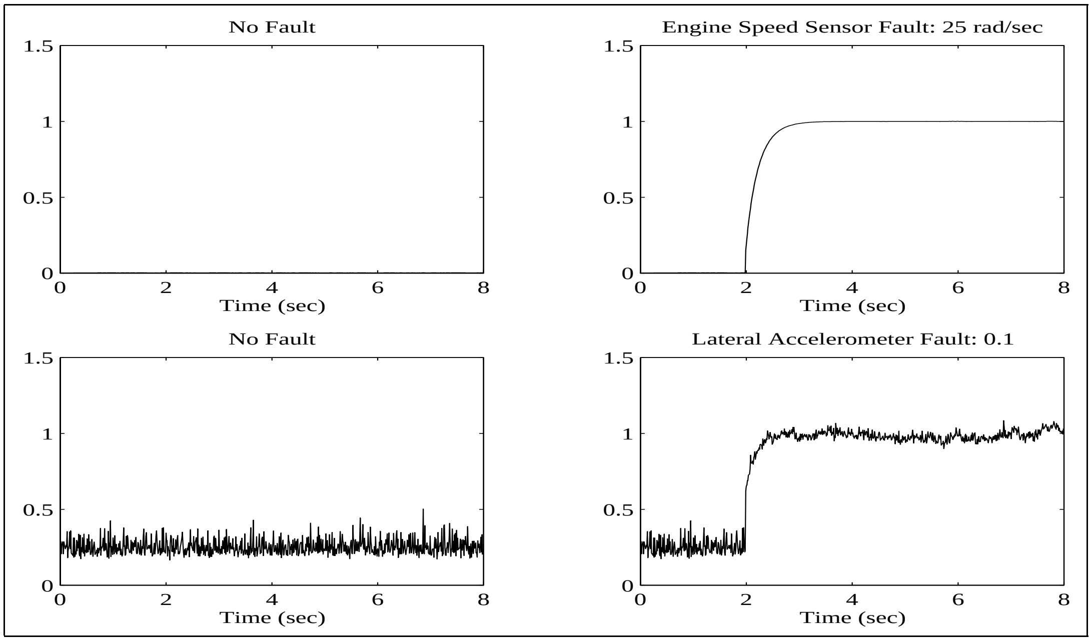

Figure 5.6: Residuals for fault detection ¯lter one.

5.2 Fault Detection Filter Evaluation On A Straight Rough Road

In this section, the performance of a longitudinal mode fault detection ¯lter described in (Douglas et al. 1995) is evaluated for robustness to noise caused by rough roads. The same types and sizes of faults are used here as in (Douglas et al. 1995). Figures 5.19, 5.20 and 5.21 illustrate detection ¯lter performance for the ¯rst, second and third fault groups Because the rough road noise dominates the nominal vertical accelerometer bias fault, this fault is hard to detect by inspection of the residual. However, both residual processing systems, the Bayesian neural network of Section 6 and the Shiryayev sequential probability ratio test of Section 7, quickly and unambiguously detect the fault.

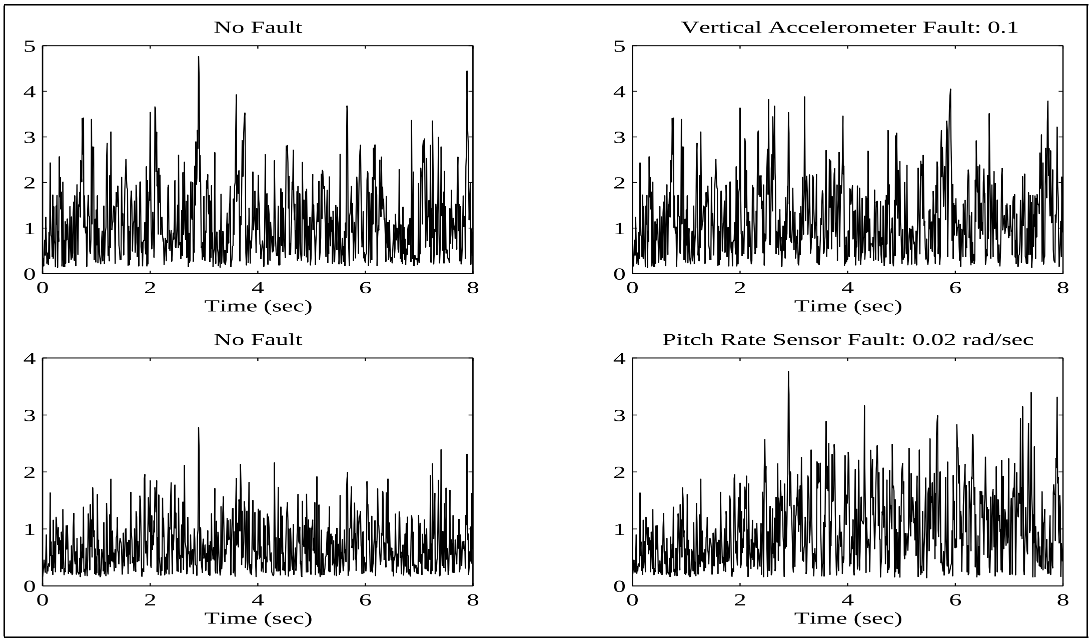

Figure 5.7: Residuals for fault detection filter one.

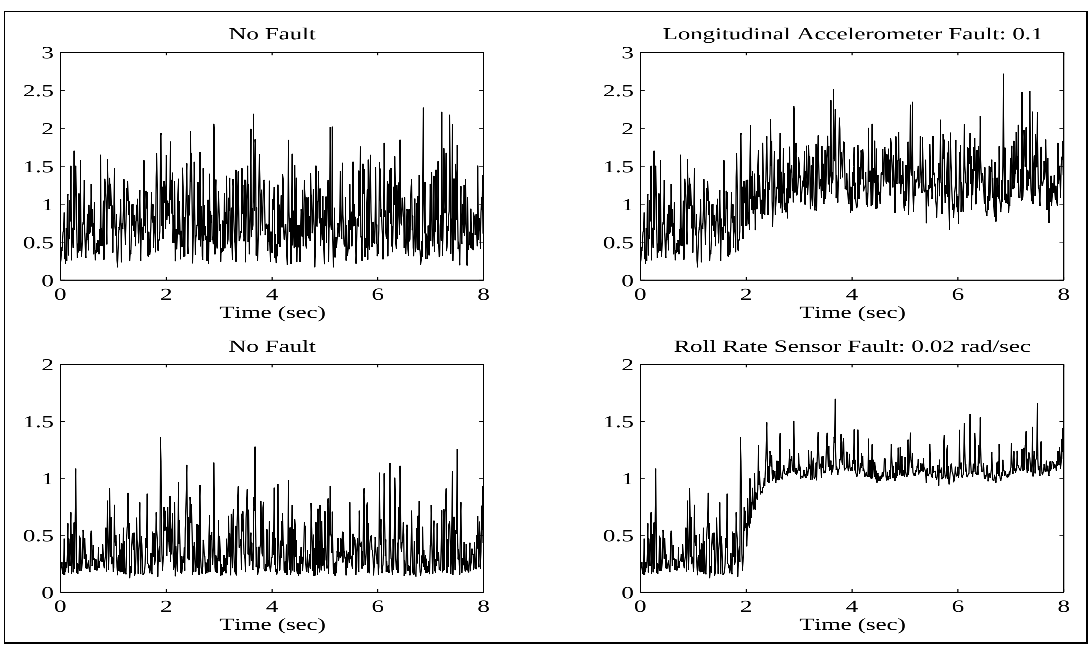

Figure 5.8: Residuals for fault detection filter two.

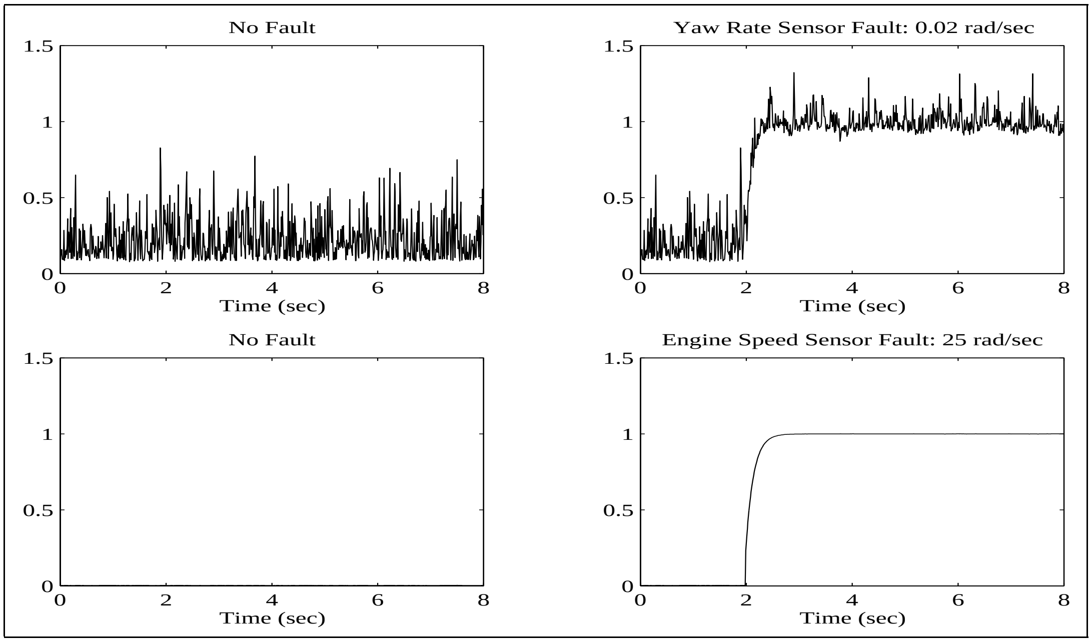

Figure 5.9: Residuals for fault detection ¯lter two.

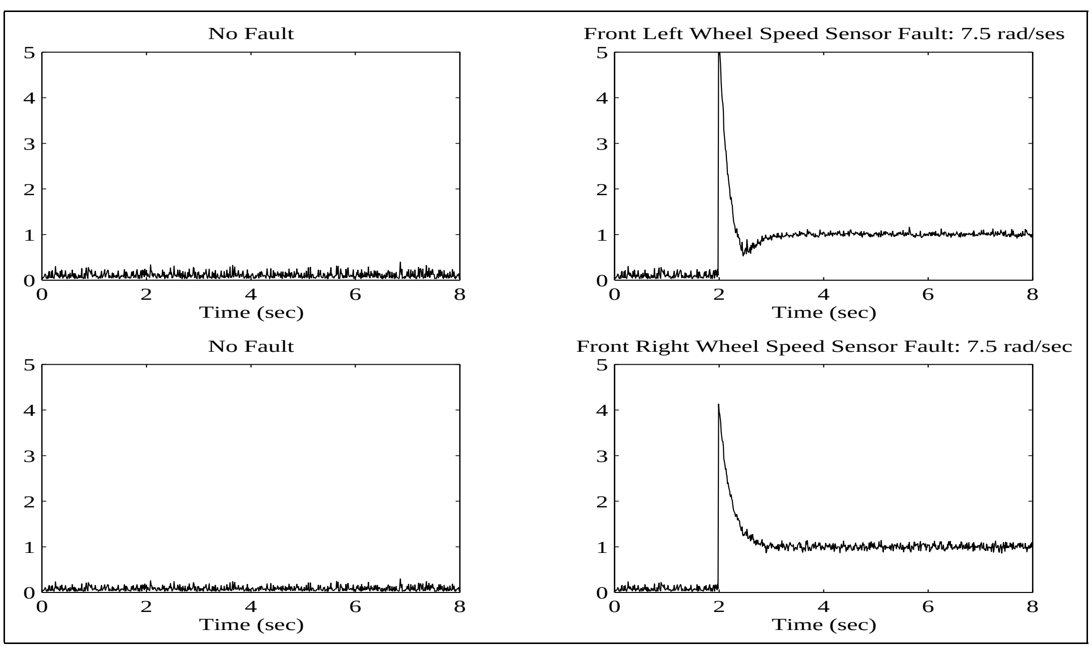

Figure 5.10: Residuals for fault detection ¯lter three.

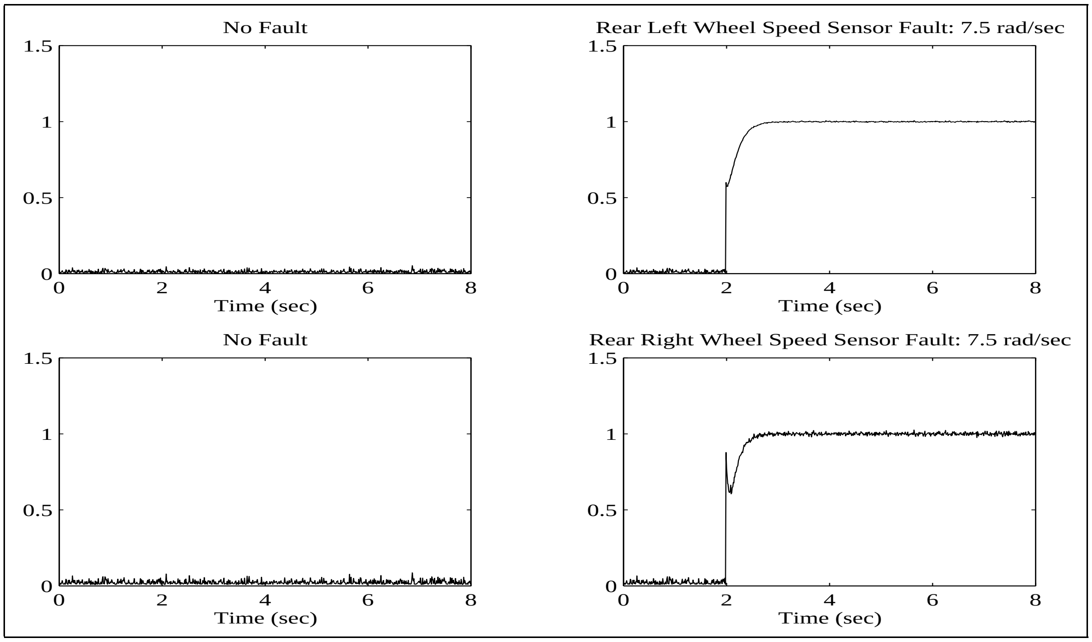

Figure 5.11: Residuals for fault detection ¯lter three.

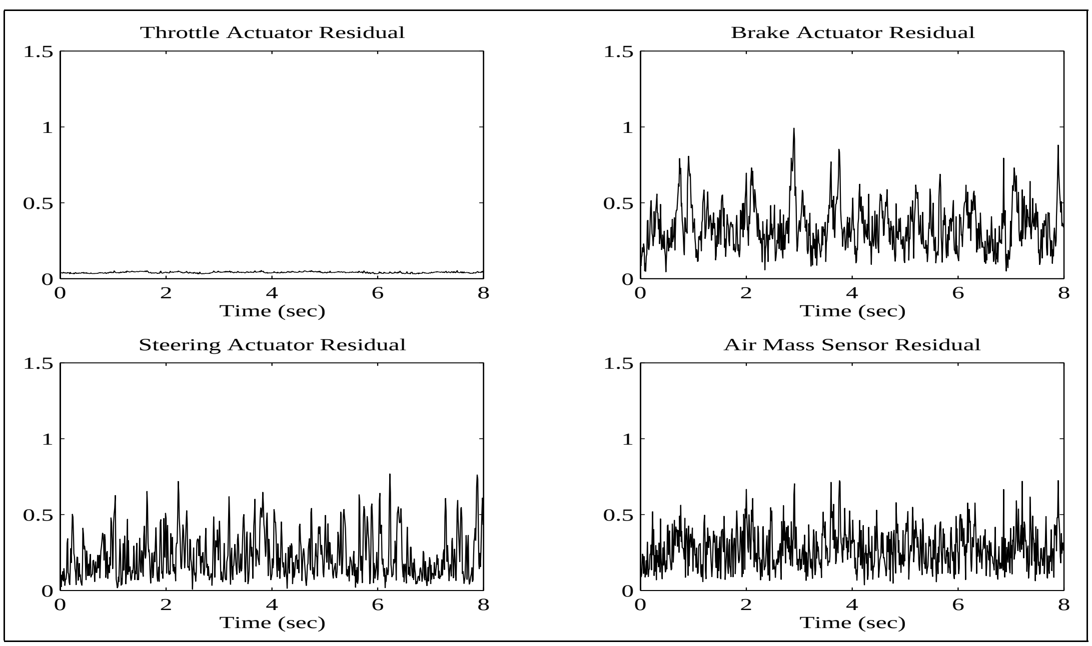

Figure 5.12: Residuals for fault detection ¯lter four, no fault.

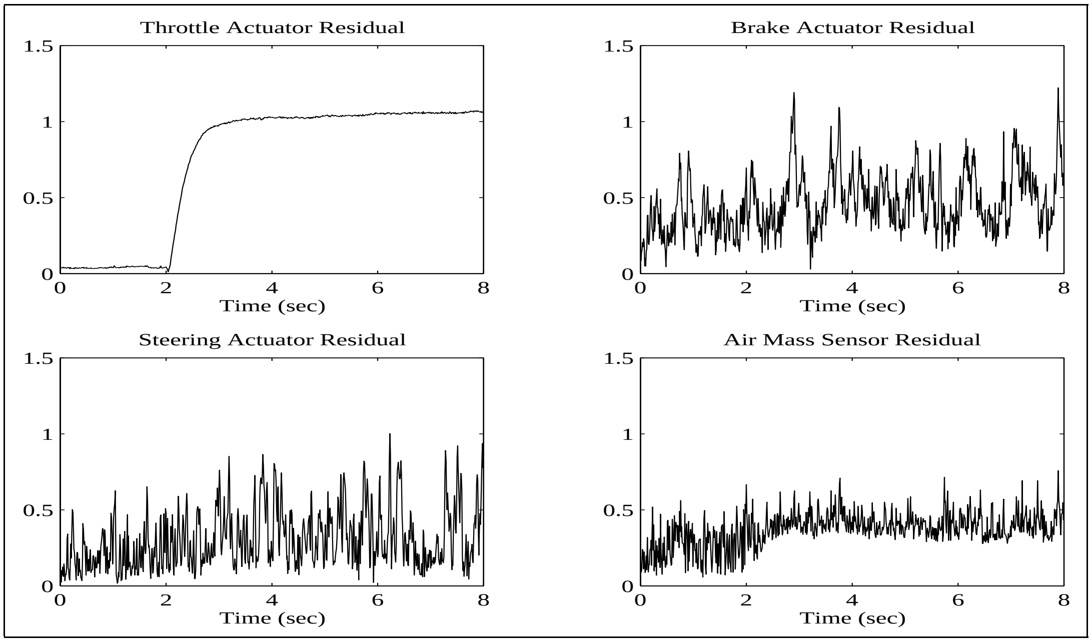

Figure 5.13: Residuals for fault detection ¯lter four, throttle actuator fault +2 deg.

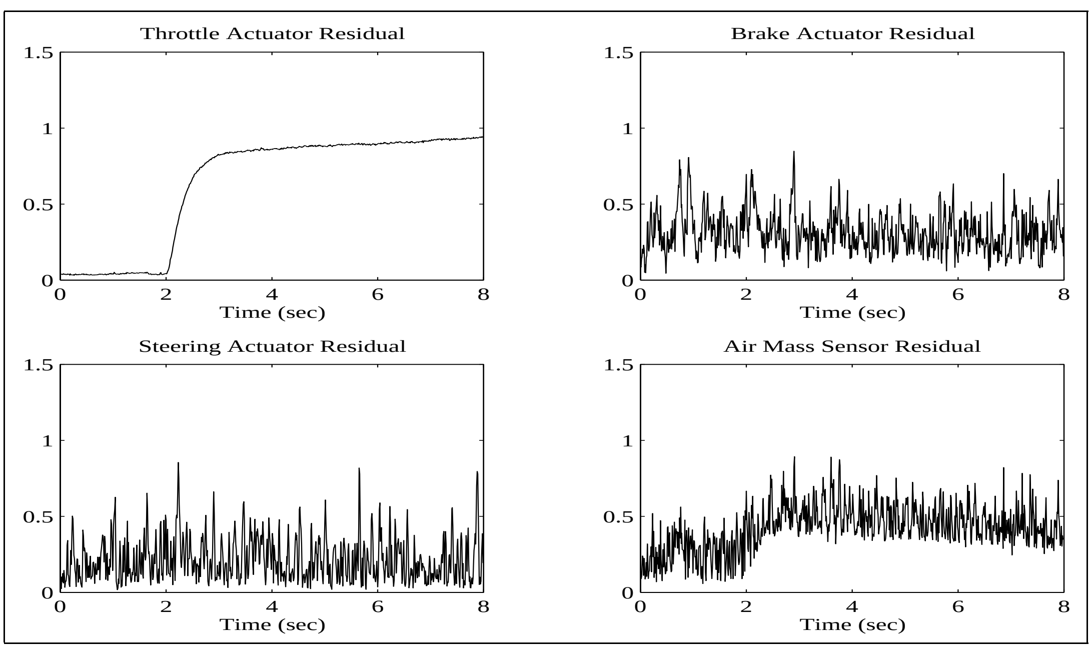

Figure 5.14: Residuals for fault detection ¯lter four, throttle actuator fault -2 deg.

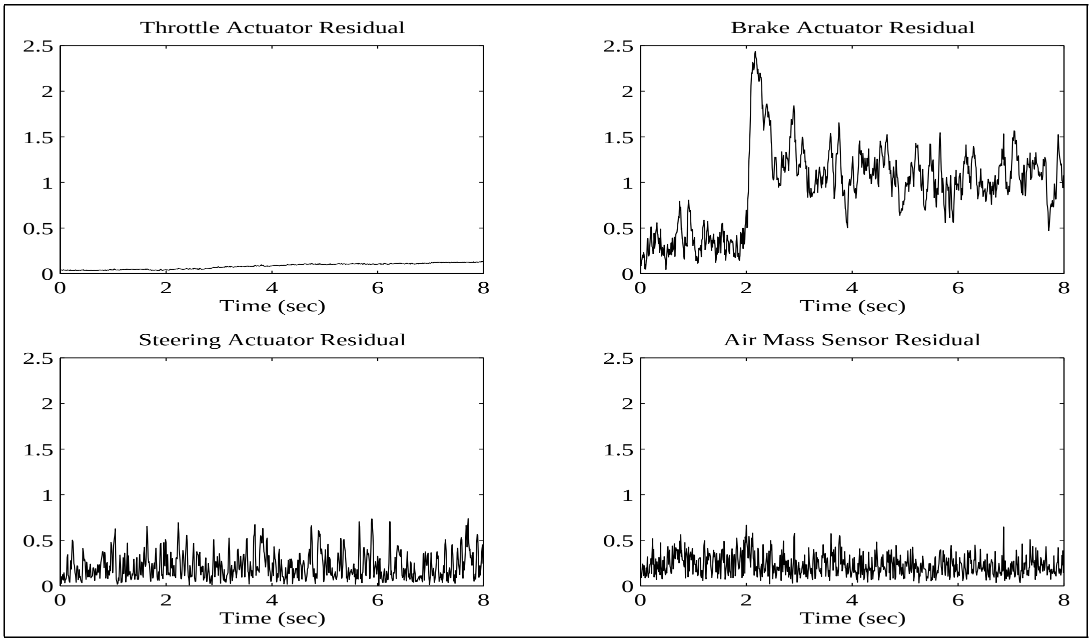

Figure 5.15: Residuals for fault detection filter four, brake actuator fault +100 Nm.

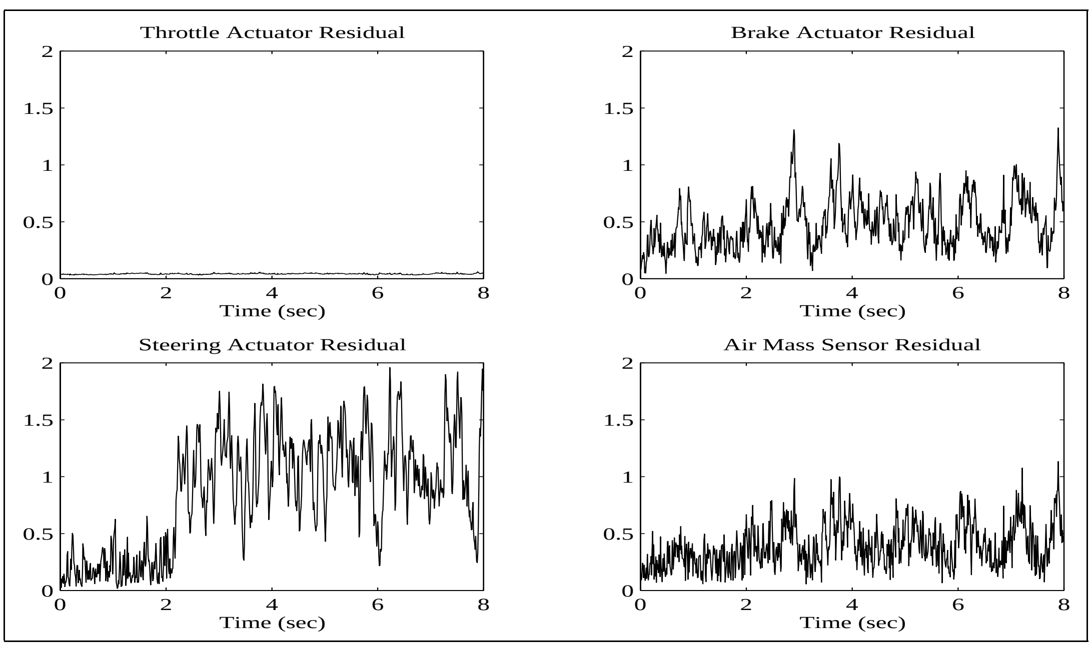

Figure 5.16: Residuals for fault detection filter four, steering actuator fault +0.001 rad.

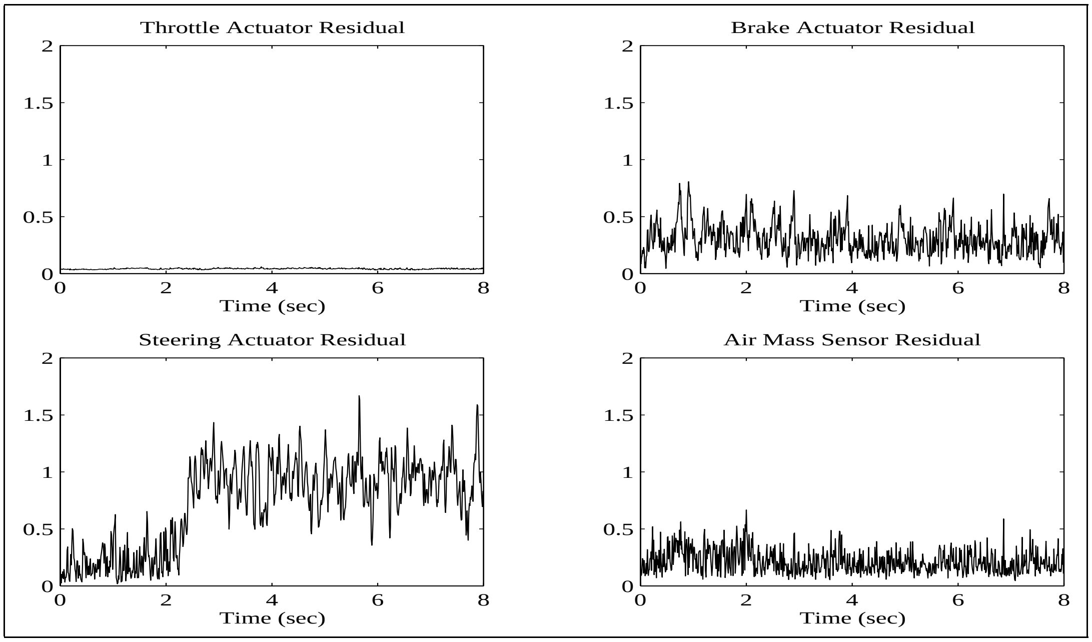

Figure 5.17: Residuals for fault detection ¯lter four, steering actuator fault -0.001 rad.

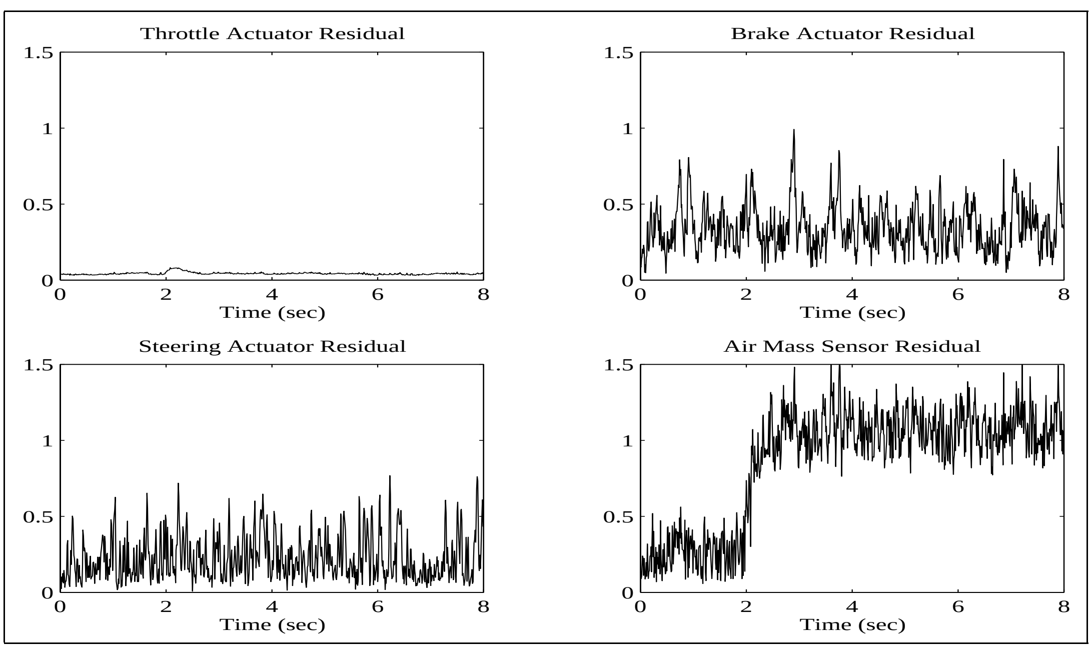

Figure 5.18: Residuals for fault detection ¯lter four, air mass sensor fault 0.07 kg.

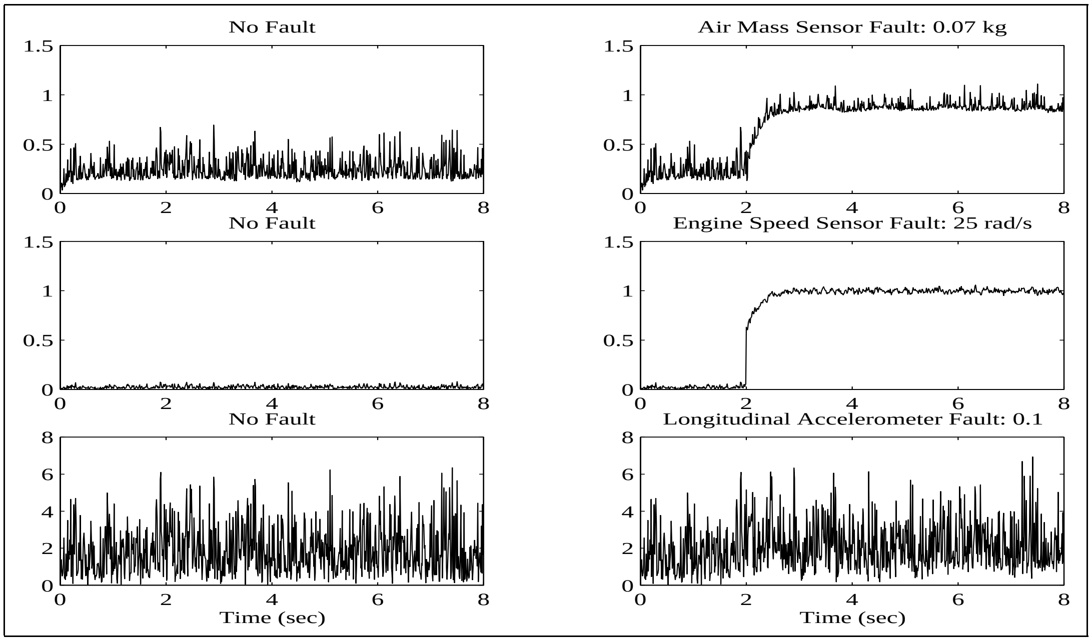

Figure 5.19: Residuals for fault detection ¯lter one: air mass sensor, engine speed sensor and forward accelerometer.

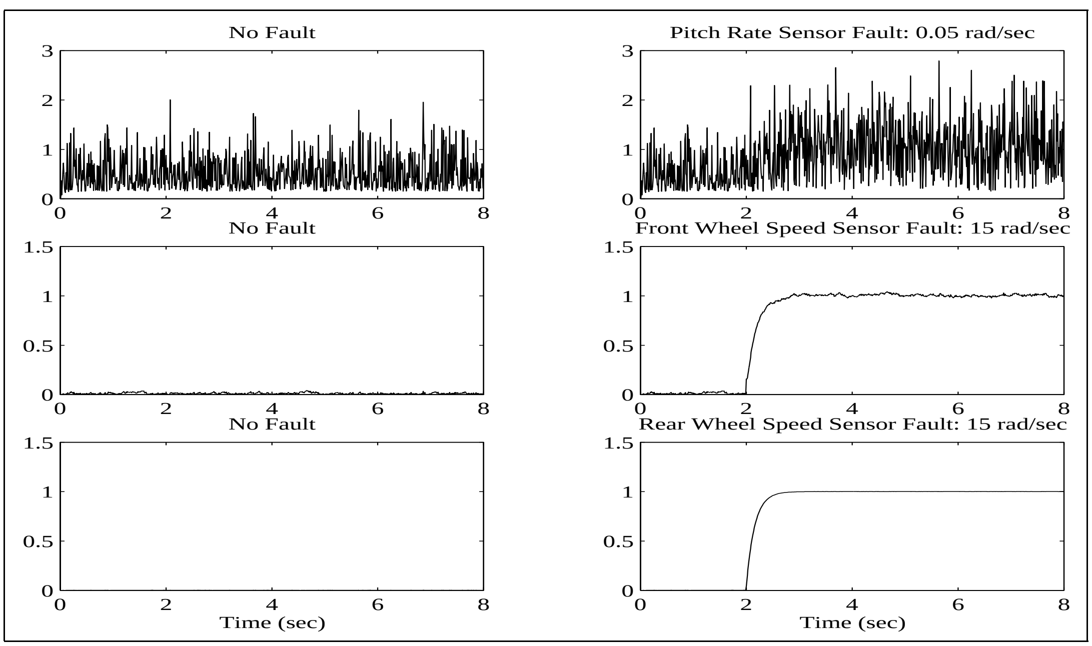

Figure 5.20: Residuals for fault detection ¯lter two: pitch rate sensor, forward wheel speed sensor and rear wheel speed sensor.

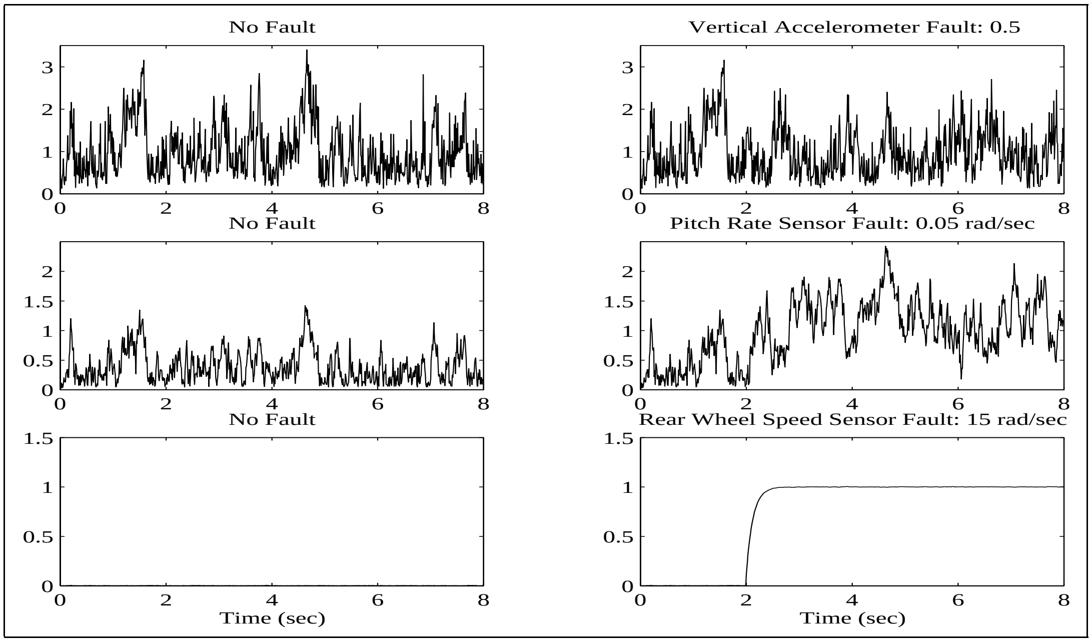

Figure 5.21: Residuals for fault detection ¯lter three: vertical accelerometer, pitch rate sensor and rear wheel speed sensor.

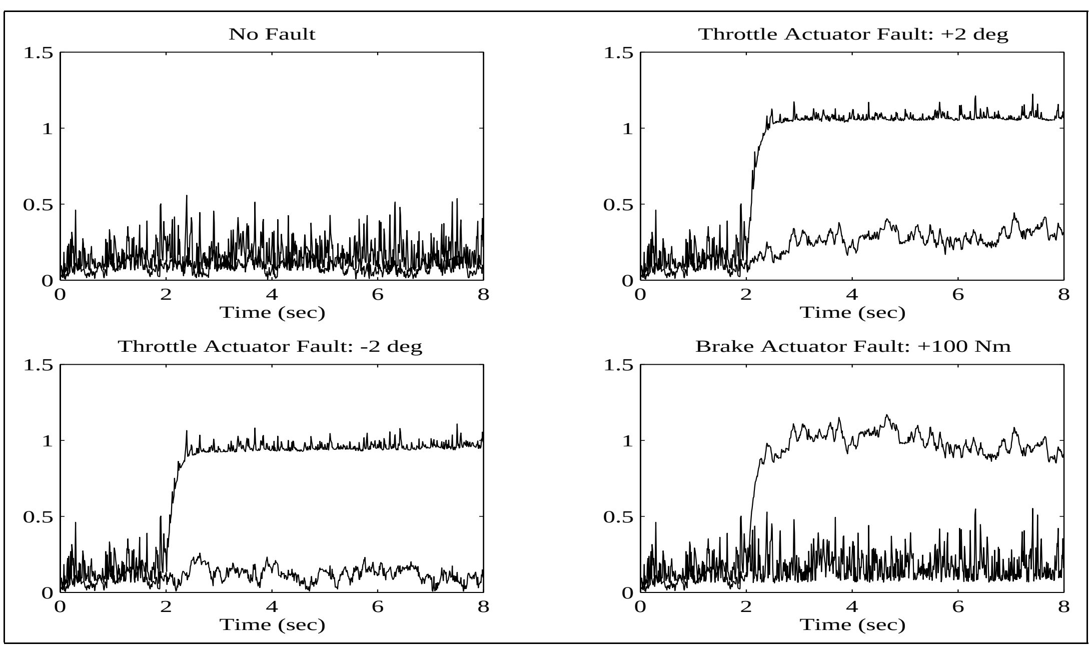

Figure 5.22: Residuals for fault detection ¯lter four: throttle actuator and brake actuator.

Chapter 6

Bayesian Neural Networks

The essential feature of a residual processor is to analyze the residual process generated by all fault detection ¯lters and announce whether or not a fault has occurred and with what probability. This requires higher level decision making and creation of rejection thresholds. Nominally, the residual process is zero in the absence of a fault and non-zero otherwise. However, when driven by sensor noise, dynamic disturbances and nonlinearities, the residual process fails to go to zero even in the absence of faults. This is noted in the simulation studies of the detection ¯lters. Furthermore, the residual process may be nonzero when a fault occurs for which the detection ¯lter is not designed. In this case, the detection ¯lter detects but cannot isolate the fault because the residual directional properties are not de¯ned.

The approach taken in this section is to consider that the residuals from all fault detection ¯lters constitute a pattern, a pattern which contains information about the presence or absence of a fault. Hence, residual processing is treated as a pattern recognition problem. This class of problems is ideally suited for application to a neural network.

The objective of a neural network as a feature classi¯er is to associate a given feature vector with a pattern class taken from a set of pattern classes de¯ned apriori. In an application to residual processing, the feature vector is a fault detection ¯lter residual and the pattern classes are a partitioning of the residual space into fault directions which include the null fault.

Three types of neural network classi¯ers are considered for the pattern recognition problem: a single layer perceptron, a multilayer perceptron and a Bayesian neural network. The single layer perceptron is the simplest continuous input neural network classi¯er and has the ability to recognize only simple patterns. It decides whether an input belongs to one of the classes by forming decision regions separated by hyperplanes. It is shown later that the decision regions formed by the single layer perceptron are similar to those formed by a maximum likelihood gaussian classi¯er if the inputs are gaussian, uncorrelated and the distributions for di®erent classes di®er only in the mean values. Note that the perceptron training procedure may lead to oscillating decision boundaries if the underlying distributions of the input intersect, that is, if the classes are not mutually exclusive.

The multilayer perceptron is a feedforward network with input, output and, possibly, hidden layers. Unlike the single layer perceptron, which partitions the decision space with hyperplanes, the multilayer perceptron forms arbitrarily complex convex decision regions. Furthermore, since no assumptions are required about the shapes of the underlying input probability distributions, the multilayer perceptron is a robust classi¯er that may be used to classify strongly non-gaussian inputs driven by nonlinear processes.

The Bayesian neural network is a multilayer perceptron with output feedback and is modi¯ed to include a sigmoidal activation function at each ouput node. The output activation functions take values between zero and one. It is shown later, in Section 6.2.2, that the output activation functions of a Bayesian neural network provide posterior probabilities of classi¯cation conditioned on the applied input history. A stochastic training algorithm further enhances robustness in that training sets are considered as sample sets providing information about the entire population. This is explained in Section 6.3.

6.1 Notation

Notation for a q-layer multilayer perceptron is as follows.

n_i	number of nodes in layer i.
$u_k \in \mathrm{R}^{n_1}$	network input at time k.
$x_k^i \in \mathrm{R}^{n_i}$	input to layer i at time k where $i \in \{2, \ldots, q\}$.
$y_k^i \in \mathrm{R}^{n_i}$	output of layer i at time k where $i \in \{1, 2, \ldots, q\}$.
$S(x)$	activation function.
$©^i \in \mathrm{R}^{n_i}$	bias vector of layer i where $i \in \{2, \ldots, q-1\}$.
$W^i \in \mathrm{R}^{n_i \times n_{i-1}}$	weighting matrix of layer i where $i \in \{2, \ldots, q\}$.

Connections for a q-layer multilayer perceptron with one step delayed output feedback are de¯ned in (6.1). The connections are illustrated in Figure 6.1 for a ¯ve layer network.

$$x_k^1 = u_k + y_{k-1} \tag{6.1a}$$

$$x_k^i = W^i y_k^{i-1} + ©^i, \qquad \text{where } i \in \{2, \ldots, q\} \tag{6.1b}$$

$$y_k^i = S(x_k^i), \qquad \text{where } i \in \{1, \ldots, q\} \tag{6.1c}$$

$$S(x) = \frac{e^x}{e^x + 1} \tag{6.1d}$$

6.2 Bayesian Feature Classi¯cation and Neural Networks

A Bayesian feature classi¯er is optimal in the sense that it assigns a feature to the pattern class with the highest posterior probability, that is, a feature vector x is associated with a pattern class $\mathcal{A}_i$ if

$$P(\mathcal{A}_i/x) > P(\mathcal{A}_j/x) \qquad \forall j \neq i$$

Most classi¯ers use probabilities conditioned on the class $P(x/\mathcal{A}_i)$ and use Bayes' rule to generate posterior probabilities, that is,

$$P(\mathcal{A}_i/x) = \frac{p(x/\mathcal{A}_i)p(\mathcal{A}_i)}{p(x)}$$

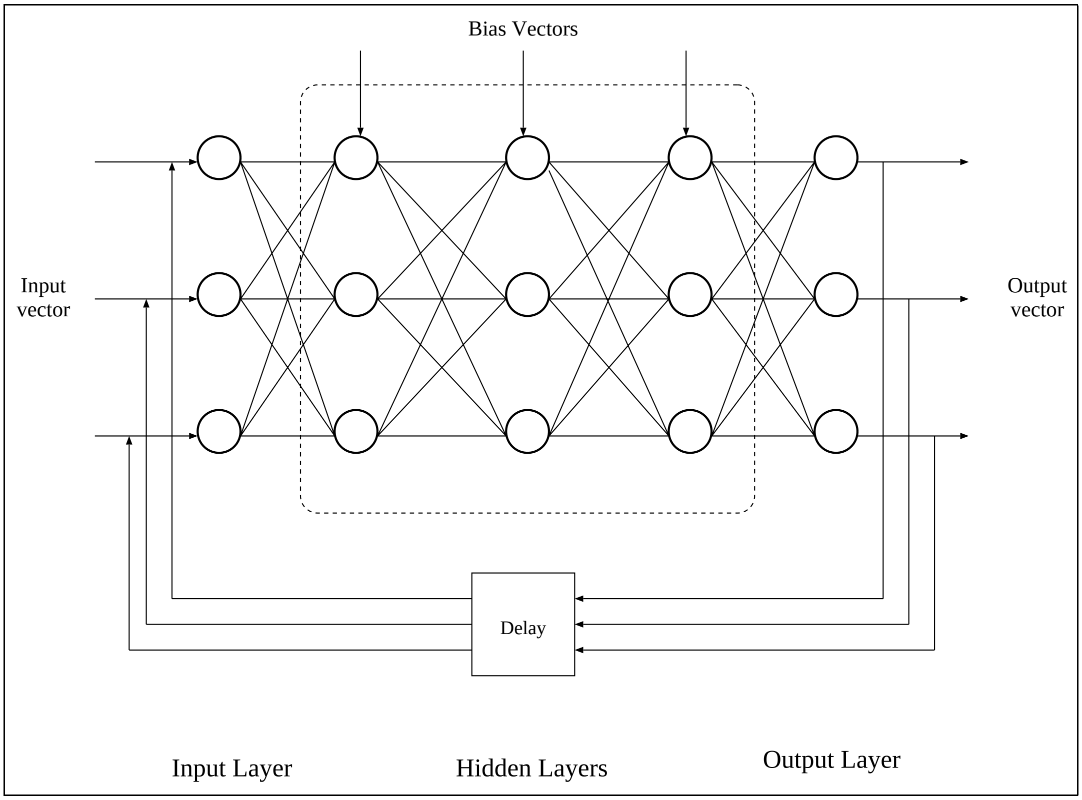

Figure 6.1: Bayesian neural network with feedback.

$$P(x) = \sum_{j=1}^{m} p(x/\mathcal{A}_j)p(\mathcal{A}_j)$$

This indirect way of calculating posterior probabilities makes assumptions about the form of the parametric models $P(x/\mathcal{A}_i)$ and the apriori probabilities (Morgan and Bourlard 1995).

Multilayer perceptrons do not require any assumptions about the pattern distributions and can form complex decision surfaces. Several authors (Richard and Lippmann 1991, Bourlard and Wellekens 1994) show that the outputs of multilayer perceptron classi¯ers can be interpreted as estimates of posterior probabilities of output classi¯cation conditioned on the input. Blaydon (Blaydon 1967) proved the same for a two-class linear classi¯er.

The following subsections provide two results that establish the utility of multilayer perceptrons as Bayesian feature classi¯ers. Section 6.2.1 shows that the decision regions created by a maximum likelihood gaussian classi¯cation algorithm can be generated using a multilayer perceptron with sigmoidal node functions. Section 6.2.2 shows that the output of a Bayesian neural network can be interpreted as an estimate, conditioned on the input history, of the posterior probabilities of feature classi¯cation.

6.2.1 A Maximum Likelihood Gaussian Classi¯er as a Multilayer Perceptron

In this section, it is shown that the decision regions created by a maximum likelihood gaussian classi¯cation algorithm can be implemented using a multilayer perceptron with sigmoidal node functions. First, consider a binary hypothesis case, that is, one where the input is assumed to be associated with one of two classes.

Let the input of a maximum likelihood gaussian classi¯cation algorithm be $x \in \mathrm{R}^n$, the output $y \in \mathrm{R}^2$ and the two classes $\mathcal{H}_i$ and $\mathcal{H}_j$. For simplicity, assume that the underlying conditional probability density functions of x have identical covariances but di®erent means as in

$$
\begin{aligned}
\mathcal{H}_i : \qquad & x \sim \mathcal{N}(m_i, ¤) \\
\mathcal{H}_j : \qquad & x \sim \mathcal{N}(m_j, ¤)
\end{aligned}
$$

where

$$
f(x/\mathcal{H}_i) \quad = \quad \frac{1}{(2\pi)^{n/2}|\ ¤|^{1/2}} \exp\left\{-\frac{1}{2}\|x - m_i\|^2_{\Lambda^{-1}}\right\}
$$

Now de¯ne a log likelihood function L_i as:

$$
\begin{aligned}
L_i \quad & \triangleq \quad 2\ln\left[(2\pi)^{n/2}|¤|^{1/2} f(x/\mathcal{H}_i)\right] \\
& = \quad -(x - m_i)^T ¤^{-1}(x - m_i)
\end{aligned}
$$

Then, the di®erence between the two likelihood functions $L_i - L_j$ has the form

$$
\begin{aligned}
L_{ij} \quad & \triangleq \quad L_i - L_j \\
& = \quad 2(m_i - m_j)^T ¤^{-1} x + (m_j^T ¤^{-1} m_j - m_i^T ¤^{-1} m_i) \\
& = \quad Wx + ©
\end{aligned}
$$

An input classi¯cation decision function follows as

$$
\begin{aligned}
L_{ij} &> 0 \qquad \Rightarrow \text{Declare } \mathcal{H}_i \\
L_{ij} &< 0 \qquad \Rightarrow \text{Declare } \mathcal{H}_j
\end{aligned}
$$

The same decision region could be obtained using a single layer perceptron in which the weighting matrix is W, the bias vector is © and the output is a sigmoidal function of the form $S(L_{ij})$. An easy extension to the multiple hypothesis case follows from a decision function based on $L_i - \max_{j \neq i} L_j$.

The similarity in form between a maximum likelihood gaussian classi¯er, as above, and a perceptron is obvious. However, note that traditional statistical classi¯ers require prior knowledge of the stochastic properties of the inputs. This is not so for perceptrons. Furthermore, it can be shown that multilayer perceptrons with hidden layers and sigmoidal nodal functions behave as universal approximators, that is, they have the capability of approximating any function to any degree of accuracy given a su±cient number hidden nodes. Refer to (Funahashi 1989, Hornik et al. 1989) for details.

6.2.2 A Bayesian Neural Network Provides Feature Classi¯cation Probabilities

This section shows that the output of a Bayesian neural network can be interpreted as an estimate of the posterior probabilities of feature classi¯cation conditioned on the input history. Let $x_k \in \mathrm{R}^n$ be a feature vector and $\mathcal{X}_k = \{x_1, \ldots, x_k\}$ be a history of feature vectors. Let

$$
\mathcal{A} = \{\mathcal{A}_1, \ldots, \mathcal{A}_q\}
$$

be a set of q pattern classes $\mathcal{A}_i$ into which a feature vector may be classi¯ed and let $y(w, x) \in \mathrm{R}^m$ be the output of a multilayer perceptron. The parameter w is a vector containing the perceptron weights and bias vectors. Let $z_k \in \mathrm{R}^m$ be a vector de¯ned as:

$$
z_k^T \triangleq \begin{cases} [0, \ldots, 1, \ldots, 0]\,, & \mathcal{X}_k \in \mathcal{A}_i \\ [0, \ldots, 0, \ldots, 0]\,, & \mathcal{X}_k \notin \mathcal{A}_i, \qquad i \in \{1, \ldots, m\} \end{cases}
$$

From the de¯nition of the Bayesian neural network connections, (6.1), the conditional expectation of z_k is

$$E[z_k/x_k] = P(\mathcal{A}/\mathcal{X}_k)$$

where $E[\cdot]$ is the expectation operator and where $P(\mathcal{A})$ is a vector of probabilities

$$P(\mathcal{A}/\mathcal{X}_k) = \begin{bmatrix} P(\mathcal{A}_1/\mathcal{X}_k) \\ \vdots \\ P(\mathcal{A}_q/\mathcal{X}_k) \end{bmatrix}$$

Note that if the $\mathcal{A}_i$ are mutually exclusive and exhaustive events, then $\|P(\mathcal{A}/\mathcal{X}_k)\|_1 = 1$.

Consider the regression function

$$J(w) = E_{x,z}\left[\|z - y(w,x)\|^2\right] \tag{6.2}$$

An expansion of the norm and the expectation operator lead to

$$\begin{aligned} J(w) &= E_x\left[E_z\left[\|z - y(w,x)\|^2/x\right]\right] \\ &= E_x\left[E_z\left[\|z\|^2 - 2y^T(w,x)z + \|y(w,x)\|^2/x\right]\right] \\ &= E_x\left[\sum P(\mathcal{A}_i/\mathcal{X}) - 2y^T(w,x)P(\mathcal{A}/\mathcal{X}) + \|y(w,x)\|^2\right] \\ &= E_x\left[\sum P(\mathcal{A}_i/\mathcal{X}) - \|P(\mathcal{A}/\mathcal{X})\|^2\right] + E_x\left[\|P(\mathcal{A}/\mathcal{X}) - y(w,x)\|^2\right] \end{aligned}$$

Since the ¯rst expectation term is independent of the multilayer perceptron parameters, minimization of J is the same as minimization of F where

$$J(w) = E_x\left[\sum P(\mathcal{A}_i/\mathcal{X}) - \|P(\mathcal{A}/\mathcal{X})\|^2\right] + F(w)$$

and

$$F(w) \triangleq E_x\left[\|P(\mathcal{A}/\mathcal{X}) - y(w,x)\|^2\right]$$

Thus, when the network parameters are chosen to minimize a mean-squared error cost function, the outputs are estimates of the Bayesian posterior probabilities.

6.3 Learning Algorithms for Neural Networks

The learning phase of a neural network involves the determination of the synaptic weights and bias vectors of the network. The backpropagation algorithm, the most widely used learning algorithm in neural network applications, consists of two passes through the layers of the network: a forward pass and a backward pass. In the forward pass, an input is applied to the input layer and allowed to propagate through the network to produce an output. During this pass, the synaptic weights and bias vectors are held fixed. In the backward pass, the network output is compared to a desired output and an error vector is formed. As the error vector propagates backward through the network, the synaptic weights and bias vectors are adjusted with an error correction rule to minimize the error. Together, the applied input and desired output form a neural network *training set*.

The learning phase may be viewed as a nonlinear unconstrained parameter optimization problem. Depending upon the nature of the input, two types of algorithms are considered: deterministic and stochastic learning algorithms. With deterministic algorithms, the cost function is specific to the given training set. Networks trained this way tend to produce unexpected results when inputs are given that were not part of the training set. With stochastic algorithms, the cost function is the expected error for a given training set. Networks trained this way tend to be more robust to unknown inputs.

6.3.1 Deterministic Learning Algorithms

Define a learning cost function J as the mean squared error between the actual and desired output

$$J(w) = \sum_{k=1}^{N} \frac{e_k}{N}$$

where

$$e_k = (z_k - y_k)^T (z_k - y_k)$$

and where y_k is the network output for training set k, z_k is the desired output from training set k and N is the number of training sets. Recall that J depends on the network weight and bias vectors.

A Davidon-Fletcher-Powell algorithm may be used to solve the unconstrained parameter optimization problem. For a quadratic cost with n parameters, the Davidon-Fletcher-Powell algorithm converges in n iterations. A rank two update for the Hessian matrix will ensure that the Hession is positive de¯nite at the end of each iteration. A suitable test for convergence is to check whether the change in the Hessian matrix is small.

6.3.2 Stochastic Learning Algorithms

From Section 6.2.2, the problem of training a Bayesian neural network may be viewed as a nonlinear regression function minimization. Consider the cost $J(w)$, a function of the network weights and bias vectors, given by (6.2)

$$J(w) = E_{x,z}\left[\|z - y(w,x)\|^2\right]$$

Let

$$\begin{aligned} \phi(w) &\triangleq \|z - y(w,x)\|^2 \\ g(w) &\triangleq -2[z - y(w,x)]^T \frac{\partial y(w,x)}{\partial w} \end{aligned}$$

For the minimization problem $\min_w J(w)$, a necessary condition for a parameter vector w to be minimizing is that

$$\nabla J(w) = E_{x,z}[g(w)] = 0$$

Since both z and x are random variables, $g(w)$ is a noisy gradient of the cost. Samples of $\phi(w)$ and $g(w)$ are available for the minimization process.

The stochastic minimization $\min_w J(w)$, may be implemented with a Robbins-Munro algorithm. The algorithm is a variation of the steepest descent algorithm

$$w_{k+1} = w_k - \rho_k g(w_k)$$

where $\rho_k > 0$, $\sum \rho_k^2 < \infty$ and $\sum \rho_k = \infty$.

It can be shown that under the following three assumptions, the algorithm converges in the mean square sense, that is,

$$E[\|w_k - w_0\|^2] = 0 \qquad k \to \infty$$

1. $\phi(w)$ has a unique zero w_0, which is bounded.

2. $g(w)$ is linear near w_0.

3. The variance of $g(w)$ is bounded above by a quadratic function of w as in

$$E[\|g(w)\|^2] \leq h[1 + \|w - w_0\|^2] \qquad h > 0$$

Of course, from Chebyshev's inequality, the algorithm also converges with probability one.

An initial solution to the problem is found by removing the expectation operator and using the Davidon-Fletcher-Powell deterministic algorithm. This validates the ¯rst two assumptions for the multilayer perceptrons. By taking partial derivatives and exploiting the fact that multilayer perceptrons have sigmoidal activation functions, it is seen that the variance of $g(w)$ is always bounded. Thus all three assumptions hold for multilayer perceptrons and the stochastic training algorithm converges with probability one.

6.4 Bayesian Neural Networks as Residual Processors

The objective of a residual processor is not just to announce a fault but to provide an associated probability of false alarm. While, multilayer perceptrons have proved to be very successful in static pattern recognition problems (Haykin 1994), a recurrent Bayesian neural network can be shown to approximate the posterior probability of feature classi¯cation conditioned on an input history.

The residual processor designs described in this section are applied to the fault detection ¯lters of (Douglas et al. 1995). These ¯lters are used when the vehicle is operating at a nominal $27\frac{\mathrm{m}}{\mathrm{sec}}$ on a straight road so vehicle lateral dynamics are not considered. Four Bayesian neural network residual processors are designed, one for each fault detection ¯lter.

A schematic of one network is provided in Figure 6.1. Each network has the following properties.

- Each has ¯ve layers: one input layer, three hidden layers and one output layer.
- A feedback loop is included where a one step delayed output is summed with the current input at the input layer.
- The activation function $S(\cdot)$ is a sigmoidal function. This function has a smooth nonlinearity which is useful for gradient calculations.
- Network connections are de¯ned in (6.1) and are illustrated in Figure 6.1. All vectors are in R^3 except for the network associated with fault group four where the vectors are in R^2.

Figure 6.2 shows the residual processing scheme using Bayesian neural networks and the fault detection ¯lters for the longitudinal simulation.

6.5 Simulation Results

Each Bayesian neural network is trained to announce the probability of a fault in a particular sensor conditioned on the residual process. The training data for each network is obtained by simulating bias faults of some nominal size in the vehicle nonlinear simulation.

Two types of faults are considered for residual processor testing: step faults and ramp faults. Step faults are an abrupt change from a no fault situation to a nominally sized fault in a particular sensor. Step faults are considered in the pitch rate and air mass sensors. Since the Bayesian neural networks are tested on the training set, no e®ort to generalize responses to unknown faults is made here.

Ramp faults correspond to a gradual, linear change from a no fault situation to a fault in a particular sensor. In contrast with the step faults, ramp faults necessarily represent fault sizes that have not been encountered by the Bayesian neural networks in their respective training sets. These kinds of faults illustrate the generalization capability of the Bayesian neural networks.

6.5.1 Step Faults

Figures 6.3, 6.4 and 6.5 each show one of the outputs of the Bayesian neural network for fault group three. This network analyzes the residuals from the fault detection ¯lter which considers sensor faults for the pitch rate, forward symmetric wheel speed and the rear symmetric wheel speed sensors.

In each ¯gure, no fault occurs from $t = 0$ to $t = 4$ sec. From $t = 4$ sec. onwards, step faults in di®erent sensors and actuators are applied one at a time and in the following order:

- pitch rate sensor (T)
- front wheel speed sensor (FS)
- rear wheel speed sensor (RS)
- air mass sensor (M)
- engine speed sensor (W)
- longitudinal accelerometer (X)
- throttle actuator (A)
- brake torque actuator (Tb)

Note in the ¯gures that there are two cases for the throttle fault. Figure 6.3 shows the posterior probability of a pitch rate sensor fault conditioned on the residual process. Figure 6.4 shows the posterior probability of a front wheel speed fault conditioned on the residual process. Figure 6.5 shows the posterior probability of a rear wheel speed sensor fault conditioned on the residual process.

Each ¯gure shows that the Bayesian neural network gives a high probability of a fault when a fault occurs in the corresponding sensor or actuator and a low probability of a fault otherwise. Note that the residual process is nonzero when a fault occurs in any sensor apart from the sensors for which the ¯lter is designed. Even though the residual is nonzero, the network correctly does not announce a fault.

6.5.2 Ramp Faults

In this section, ramp faults are considered in the pitch rate sensor, vertical accelerometer, longitudinal accelerometer and the air mass sensor.

Figures 6.6 and 6.7 show fault detection ¯lter residuals and outputs of a Bayesian neural network for fault group three. In these ¯gures, Z, T and RS denote the magnitudes of the vertical accelerometer, pitch rate and real wheel speed residuals and P(Z), P(T) and P(RS) denote the posterior probability of the corresponding fault conditioned on the residual process. Figures 6.8 and 6.9 show the same results but for fault group one. In these ¯gures, M, W and X denote the magnitudes of the air mass sensor, engine speed sensor and longitudinal accelerometer residuals and P(M), P(W) and P(X) denote the posterior probability of the corresponding fault conditioned on the residual process. In each ¯gure, no fault occurs from $t = 0$ to $t = 1$ sec. and from $t = 1$ sec. onwards, a ramp fault occurs.

- Figure 6.6 shows results when a ramp fault of size 0 to $0.5\frac{\mathrm{rad}}{\mathrm{sec}}$ occurs in the pitch rate sensor. Note that the Bayesian neural network has been trained with a nominal pitch rate sensor step fault of $0.05\frac{\mathrm{rad}}{\mathrm{sec}}$.

- Figure 6.7 shows results when a ramp fault of size 0 to $5\frac{\mathrm{m}}{\mathrm{sec}^2}$ occurs in the vertical accelerometer. The network has been trained with a nominal vertical accelerometer step fault of $0.5\frac{\mathrm{m}}{\mathrm{sec}^2}$.

- Figure 6.8 shows results when a ramp fault of size 0 to $1\frac{\mathrm{m}}{\mathrm{sec}^2}$ occurs in the longitudinal accelerometer. Training has been done with a nominal longitudinal accelerometer step fault of $0.1\frac{\mathrm{m}}{\mathrm{sec}^2}$.

- Figure 6.9 shows results when a ramp fault of size 0 to 0.14 kg. occurs in the air mass sensor. The network has been trained with a nominal air mass sensor step fault of 0.07 kg.

6.6 Discussion

At this stage, an interesting comparison may be made of the stochastic and deterministic training approaches. In the stochastic approach, the training sets are considered as sample sets which provide information about the entire population. In the deterministic approach, the training sets are the entire population hence no effort is made to generalize. The classes may intersect in the pattern space for the stochastic problem, while the deterministic approach theoretically considers mutually exclusive classes only.

From a theoretical perspective, when Bayesian neural networks are trained for pattern classification using the mean square criterion, their outputs are estimates of classification probabilities conditioned on the input. This conclusion is valid for *any* approach based on the minimization of the mean-squared error criterion. However, in theory, multilayer perceptrons can approximate any non-linear mapping (Lippmann 1987), hence, they are more likely to fit the posterior probabilities. The simulation studies conducted demonstrate the above assertion.

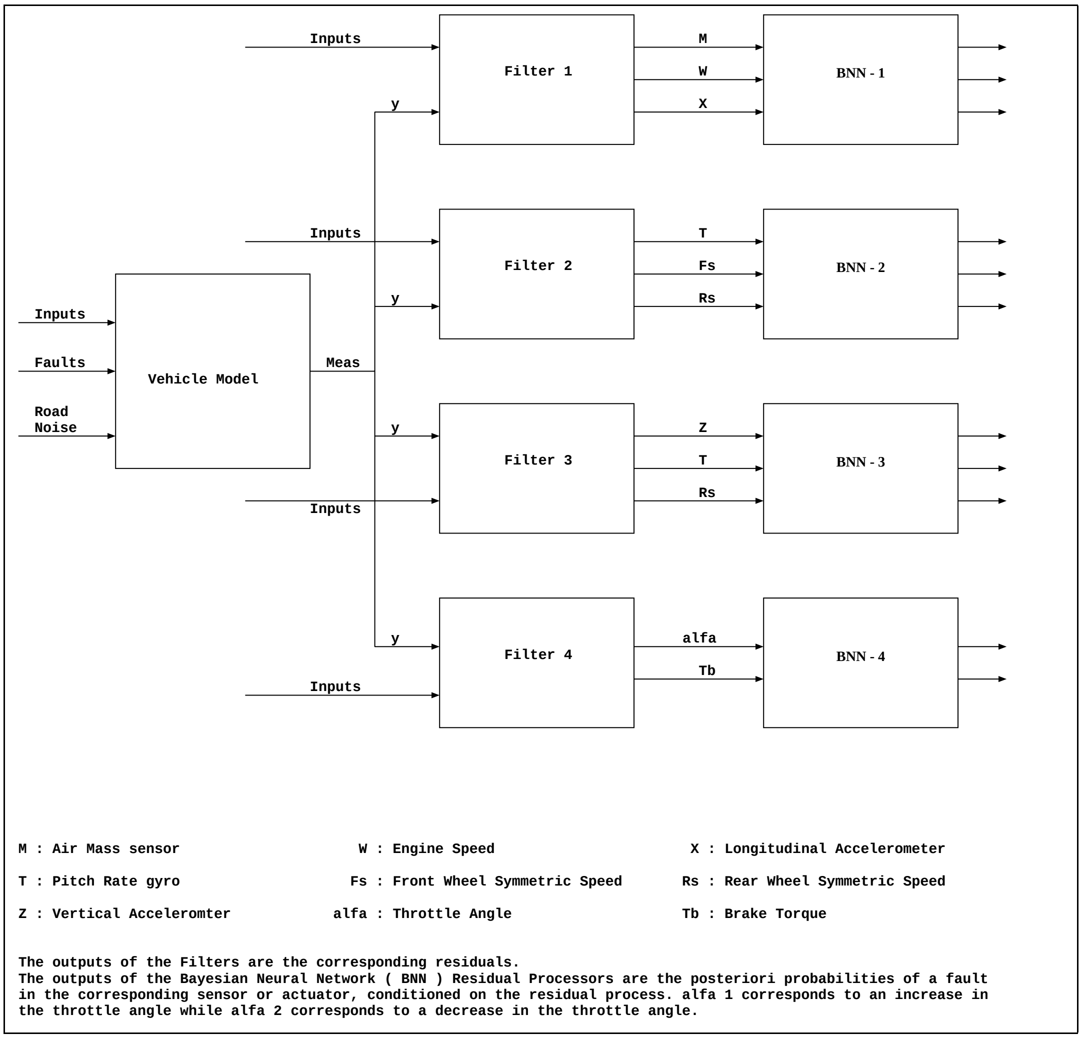

Figure 6.2: Residual processing scheme for the longitudinal simulation.

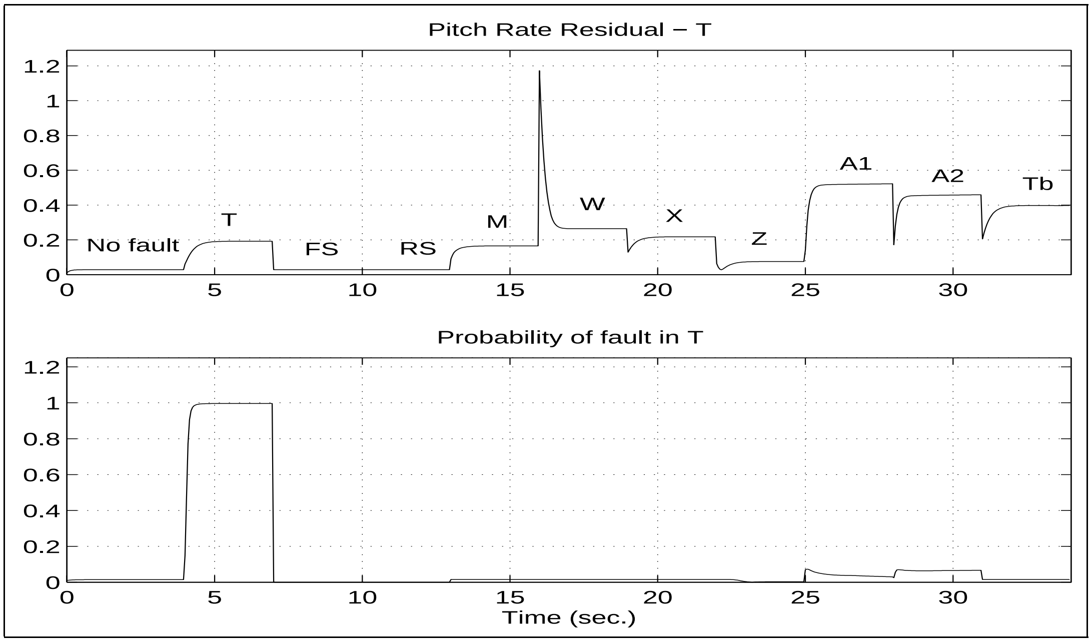

Figure 6.3: Posterior probability of a fault in the pitch rate sensor.

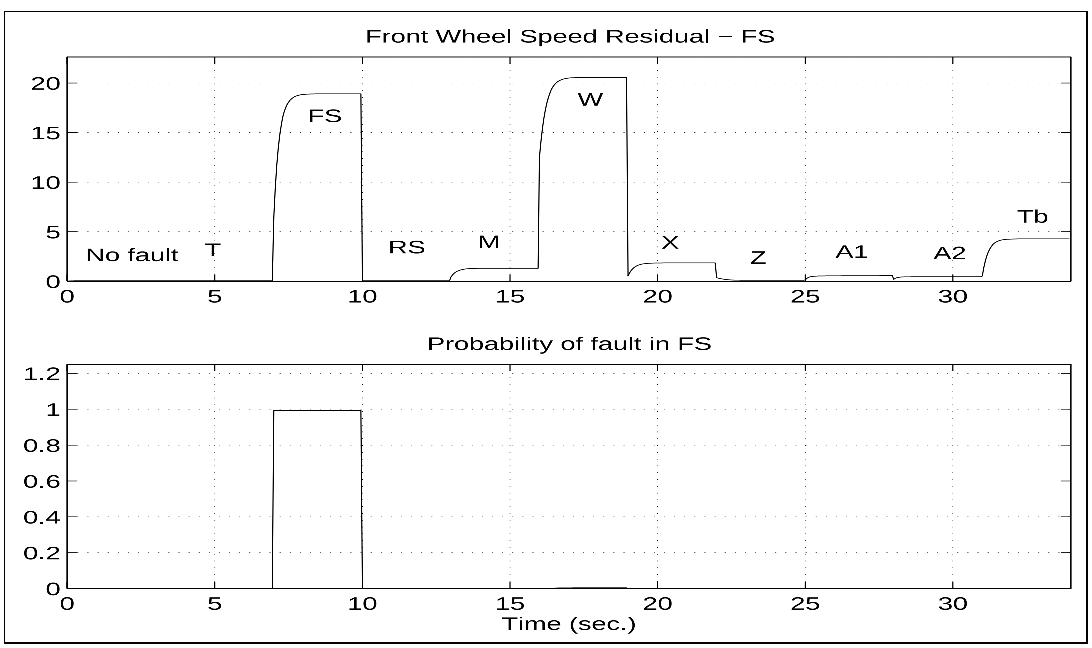

Figure 6.4: Posterior probability of a fault in the front wheel speed sensor.

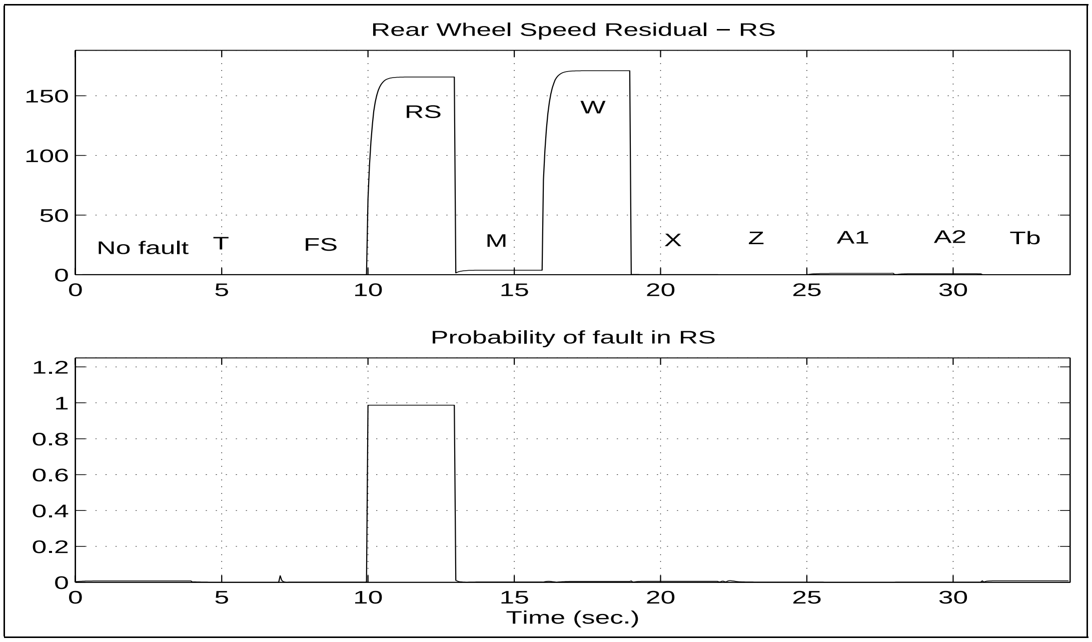

Figure 6.5: Posterior probability of a fault in the rear wheel speed sensor.

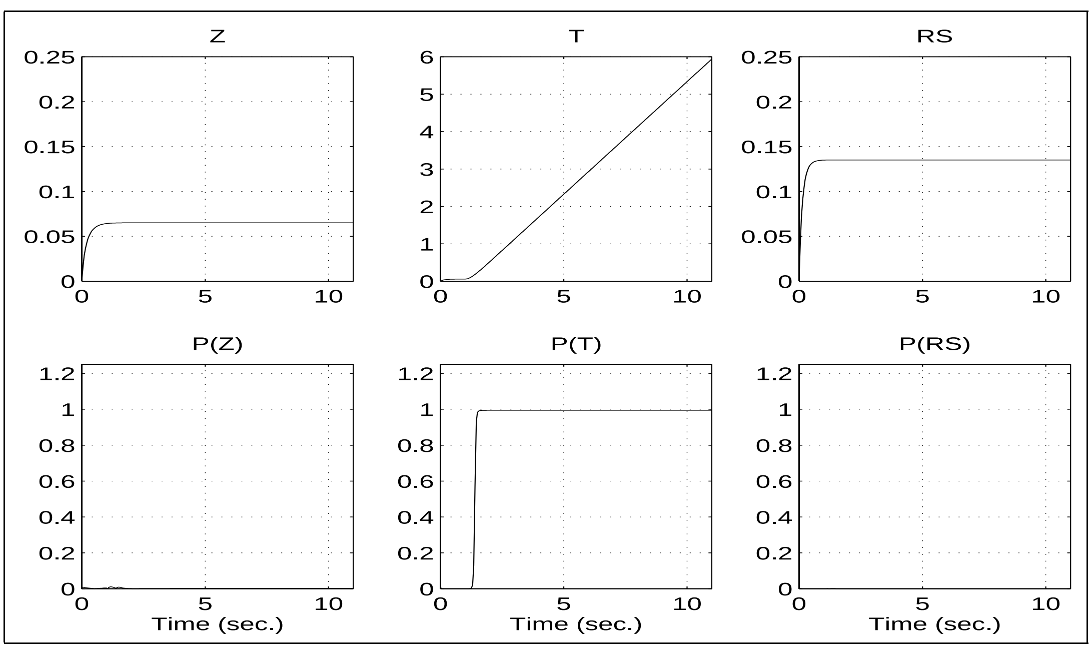

Figure 6.6: Ramp fault in pitch rate sensor.

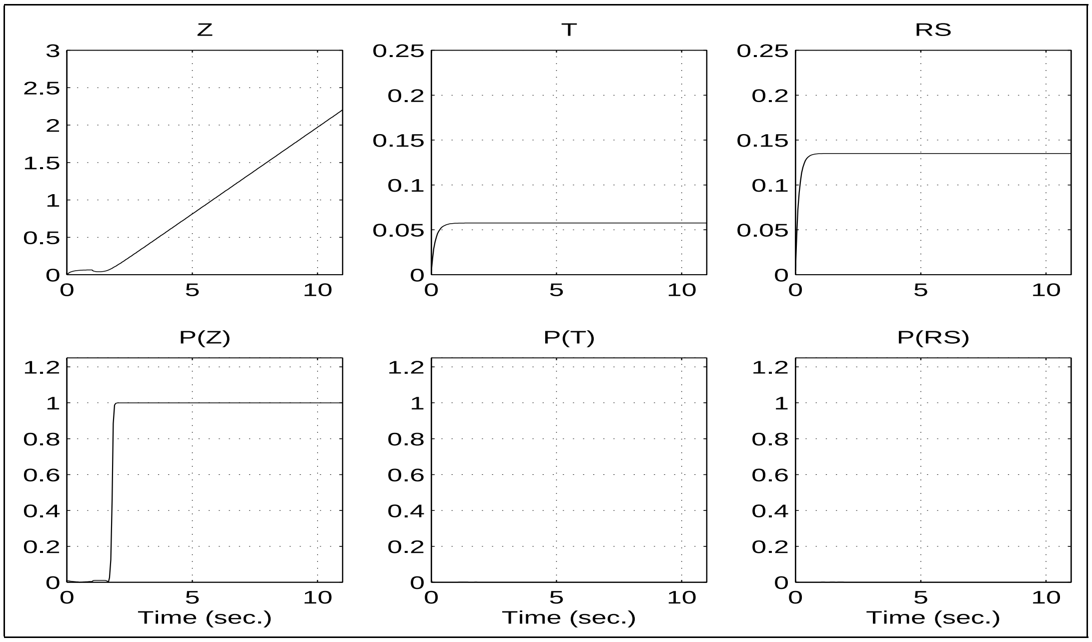

Figure 6.7: Ramp fault in vertical accelerometer.

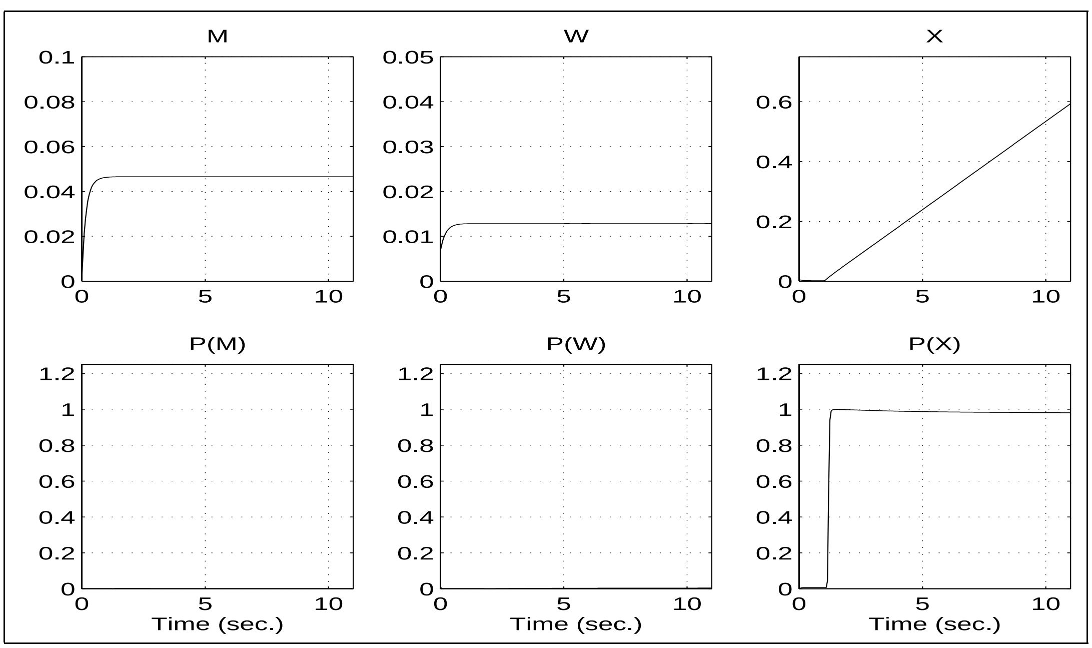

Figure 6.8: Ramp fault in longitudinal accelerometer.

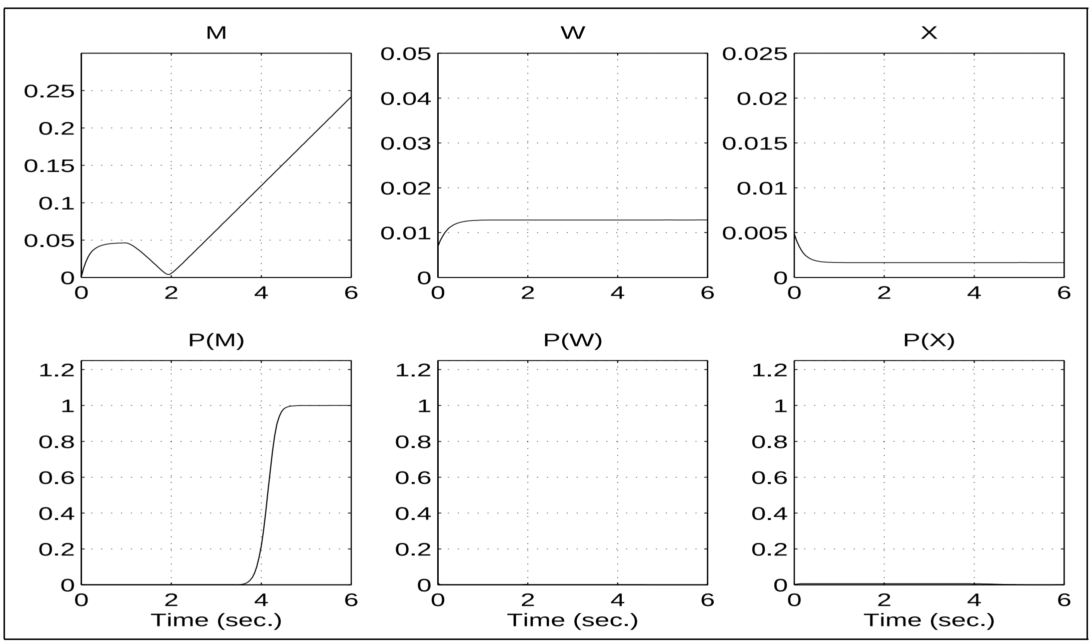

Figure 6.9: Ramp fault in air mass sensor.

Chapter 7

Sequential Probability Ratio Tests

THE RESIDUAL PROCESSING PROBLEM is considered in this section as a hypothesis detection and identification problem. Both Bayesian (Shiryayev 1977) and non-Bayesian approaches (Nikiforov 1995, Basseville and Nikiforov 1995) to the classical change detection problem have been developed. A binary hypothesis Shiryayev test, which is a Bayesian approach, is formulated by Speyer and White (Speyer and White 1984) as a dynamic programming problem. A similar approach, one also using a dynamic programming formulation, is taken here to derive an online multiple hypothesis Shiryayev Sequential Probability Ratio Test (SPRT).

It is shown that for a certain criterion of optimality, this extended Shiryayev SPRT detects and isolates the occurrence of a failure in a conditionally independent measurement sequence in minimum time. The algorithm is shown to be optimal even in the asymptotic sense and the theoretical results have been extended to the detection and identification of changes with unknown parameters. The dynamic programming analysis includes the measurement cost, the cost of a false alarm and the cost of a miss-alarm.

Note that with the Shiryayev SPRT, a change in the residual hypothesis is detected in minimum time. In contrast, the Wald SPRT detects the presence or absence of a failure in the entire measurement sequence. Here, the residual hypothesis is unknown but is assumed to be constant through the measurement sequence.

A non-Bayesian approach to the classical change detection problem is the Generalized CUMulative SUM (CUSUM) algorithm (Nikiforov 1995, Basseville and Nikiforov 1995). It has been shown that there exists a lower bound for the worst mean detection delay and that the CUSUM algorithm reaches this lower bound. This establishes the algorithms worst mean detection time minimax optimality.

Recently, the algorithm has been extended to solve the change detection and isolation problem (Nikiforov 1995). This extension is based on the log likelihood ratio between two hypotheses $\mathcal{H}_i$ and $\mathcal{H}_j$. When the di®erence between the log likelihood ratio and its current minimum value for a given hypotheses $\mathcal{H}_i$ and other hypotheses exceeds a chosen threshold, hypothesis $\mathcal{H}_i$ is announced. This implies that a hypothesis announcement requires that the recent measurements be signi¯cant enough to support the announcement.

Several important observations are made regarding the extended CUSUM algorithm.

- The algorithm is computationally intensive and is not recursive. If the number of hypotheses is m, the number of computations is of the order of m^2. This problem can be avoided by modifying the algorithm to compare all the hypotheses to the null hypothesis $\mathcal{H}_0$ while doing the computations. This modi¯cation would reduce the number of computations from the order of m^2 to m.

- No assumption is made about the apriori probability of change from hypothesis $\mathcal{H}_0$ to $\mathcal{H}_i$ from one measurement to the next. This probability is embedded explicitly in the Shiryayev SPRT.

- Unlike the Shiryayev SPRT, the posterior probability of a hypothesis change is not calculated in the CUSUM algorithm.

- Thresholds for hypothesis change announcements must be made apriori whereas in the Shiryayev SPRT, a methodology for interpreting the choice of the threshold is explicit.

- The CUSUM algorithm is similar to the Wald SPRT in that a ¯nite size, sliding data window allows for changes in hypothesis to be detected but that the hypothesis essentially is assumed to be constant throughout the window.

This chapter is organized as follows. Notation is de¯ned in Section 7.1. Section 7.2 has the main development of a multiple hypothesis Shiryayev sequential probability ratio test. First, a conditional probability propagation equation is developed. Next, a dynamic programming problem is de¯ned and some of the asymptotic properties of the cost function are demonstrated. Next, a decision rule is de¯ned by building thresholds. Finally, the test is generalized to the detection and isolation of changes with unknown parameters. In Section 7.3 a few illustrative examples are given and in Section 7.4, the algorithm is applied to a health monitoring system for automated vehicles using a high-¯delity nonlinear simulation. The performance of the algorithm is evaluated by implementing it in a fault detection and identi¯cation scheme in the longitudinal nonlinear vehicle simulation. Finally, in Section 7.5, a few comments are made about assumptions underlying the MHSSPRT.

7.1 Preliminaries and Notation

Let x_k be a measurement vector at time t_k and $\mathrm{X}_k \triangleq \{x_k\}$ be a conditionally independent measurement sequence. A fault is said to occur when there exists a discrete jump in the probabilistic description of X_k. The probabilistic description of X_k is assumed to be known both before and after a fault occurs. The fault hypotheses are enumerated as faults of type i with the total number of faults $m + 1$ being ¯xed. The fault type 0 is also called the no-fault or null-fault hypothesis.

The probability density function of x_k in the no-fault or type i fault state is denoted $f_0(\cdot)$ or $f_i(\cdot)$. These probability density functions are constant so no subscript k is indicated.

However, note that in the following development, the density functions are not required to be constant.

During any time interval $t_k < t \leq t_{k+1}$, the probability that the measurement sequence X_k will switch from a no-fault state to a type i fault state is known apriori and is denoted p_i. The time that the measurement sequence switches from a no-fault state to a type i fault state is not known and is denoted θ_i.

The probability that a type i fault has occurred before time t_0 is $\pi_i \triangleq P(\theta_i \leq t_0)$. The probability, conditioned on the measurement sequence X_k, that a type i fault has occurred before time t_k is $F_{k,i} = P(\theta_i \leq t_k/\mathrm{X}_k)$. The above notation and definitions are summarized as follows.

$x_k \triangleq$ Measurement vector at time t_k.

$\mathrm{X}_k \triangleq$ Measurement history through t_k.

$m \triangleq$ Number of fault types.

$f_0(\cdot) \triangleq$ Probability density function of x_k under no-fault hypothesis.

$f_i(\cdot) \triangleq$ Probability density function of x_k under type i fault hypothesis.

$p_i \triangleq$ Apriori probability of change from no-fault to type i fault for $t_k < t \leq t_{k+1}$.

$\theta_i \triangleq$ Time of type i fault.

$\pi_i \triangleq P(\theta_i \leq t_0)$.

$F_{k,i} \triangleq P(\theta_i \leq t_k/\mathrm{X}_k)$

7.2 Development of a Multiple Hypothesis Shiryayev SPRT

An extension of the Shiryayev sequential probability ratio test to allow multiple hypotheses is as follows. First, a conditional probability propagation equation is developed. Next, a dynamic programming problem is defined and some of the asymptotic properties of the cost function are demonstrated. Next, a decision rule is defined by building thresholds. Finally, the test is generalized to the detection and isolation of changes with unknown parameters.

7.2.1 Recursive Relation for the Posteriori Probability

The results of this section are encapsulated in two propositions. The ¯rst proposition provides a recursive update for $F_{k,i}$, the conditional probability that a type i fault has occurred. The second proposition shows that $F_{k,i}$, as given by the recursion, is consistent with the de¯nition of a probability.

Proposition 7.1. A recursive update formula for $F_{k,i}$ is

$$F_{0,i} = \pi_i \tag{7.1a}$$

$$F_{k+1,i} = \frac{M_{k,i} f_i(x_{k+1})}{\left(\sum_{i=1}^{m} M_{k,i}\right) f_i(x_{k+1}) + \left(1 - \sum_{i=1}^{m} M_{k,i}\right) f_0(x_{k+1})} \tag{7.1b}$$

where

$$M_{k,i} = F_{k,i} + p_i(1 - F_{k,i}) \tag{7.1c}$$

Proof. The proof is done by induction. The probability $F_{1,i}$ that a type i fault has occurred before t_1 given a measurement x_1 is given by Bayes' rule as

$$P(\theta_i \leq t_1/x_1) = \frac{P(x_1/\theta_i \leq t_1)P(\theta_i \leq t_1)}{P(x_1)} \tag{7.2}$$

where

$$P(x_1) = \sum_{i=1}^{m} [P(x_1/\theta_i \leq t_1)P(\theta_i \leq t_1) + P(x_1/\theta_i > t_1)P(\theta_i > t_1)] \tag{7.3a}$$

$$\begin{aligned} P(\theta_i \leq t_1) &= P(\theta_i \leq t_0) + P(t_0 < \theta_i \leq t_1) \\ &= \pi_i + p_i(1 - \pi_i) \end{aligned} \tag{7.3b}$$

$$P(x_1/\theta_i > t_1) = f_0(x_1)dx_1 \tag{7.3c}$$

$$P(x_1/\theta_i \leq t_1) = f_i(x_1)dx_1 \tag{7.3d}$$

$$\sum_{i=1}^{m} P(\theta_i > t_1) = 1 - \sum_{i=1}^{m} P(\theta_i \leq t_1) \tag{7.3e}$$

Strictly, (7.3d) denotes the probability that the measurement lies between x_1 and $x_1 + dx_1$ given the occurrence of a type i fault at or before t_1. Expanding (7.2) with the identities of (7.3) produces the fault probability $F_{1,i}$.

$$F_{1,i} = \frac{[\pi_i + p_i(1-\pi_i)]f_i(x_1)}{\sum_{i=1}^{m}[\pi_i + p_i(1-\pi_i)]f_i(x_1) + (1-\sum_{i=1}^{m}[\pi_i + p_i(1-\pi_i)])f_0(x_1)} \tag{7.4}$$

The probability $F_{k,i}$, conditioned on a measurement sequence $mathrmX_k$, that a type i fault has occurred before t_k is given by Bayes' rule as

$$P(\theta_i \le t_{k+1}/\mathrm{X}_{k+1}) = \frac{P(\mathrm{X}_{k+1}/\theta_i \le t_{k+1})P(\theta_i \le t_{k+1})}{\mathrm{X}_{k+1}}$$

Since the measurement sequence is conditionally independent, this expands to

$$P(\theta_i \le t_{k+1}/\mathrm{X}_{k+1}) = \frac{P(x_{k+1}/\theta_i \le t_{k+1})P(\mathrm{X}_k/\theta_i \le t_{k+1})P(\theta_i \le t_{k+1})}{P(\mathrm{X}_{k+1})}$$

and ¯nally to

$$P(\theta_i \le t_{k+1}/\mathrm{X}_{k+1}) = \frac{P(x_{k+1}/\theta_i \le t_{k+1})P(\theta_i \le t_{k+1}/\mathrm{X}_k)}{P(x_{k+1}/\mathrm{X}_k)} \tag{7.5}$$

which follows from the identity

$$P(\mathrm{X}_{k+1}) = P(x_{k+1}/\mathrm{X}_k)P(\mathrm{X}_k)$$

Now, consider the following identities

$$\begin{aligned}
P(x_{k+1}/\mathrm{X}_k) &= \sum_{i=1}^{m}[P(x_{k+1}/\theta_i \le t_{k+1})P(\theta_i \le t_{k+1}/\mathrm{X}_k) + \\
&\quad P(x_{k+1}/\theta_i > t_{k+1})P(\theta_i > t_{k+1}/\mathrm{X}_k)] && \text{(7.6a)} \\
P(\theta_i \le t_{k+1}/\mathrm{X}_k) &= P(\theta_i \le t_k/\mathrm{X}_k) + P(t_k < \theta_i <= t_{k+1}/\mathrm{X}_k) && \text{(7.6b)} \\
&= F_{k,i} + p_i(1-F_{k,i}) && \text{(7.6c)} \\
P(x_{k+1}/\theta_i > t_{k+1}) &= f_0(x_{k+1})dx_{k+1} && \text{(7.6d)} \\
P(x_{k+1}/\theta_i \le t_{k+1}) &= f_i(x_{k+1})dx_{k+1} && \text{(7.6e)} \\
\sum_{i=1}^{m} P(\theta_i > t_{k+1}/\mathrm{X}_k) &= 1 - \sum_{i=1}^{m} P(\theta_i \le t_{k+1}/\mathrm{X}_k) && \text{(7.6f)}
\end{aligned}$$

Expanding (7.5) with the identities of (7.6) produces the fault probability $F_{k+1,i}$.

$$F_{k+1,i} = \frac{M_{k,i}f_i(x_{k+1})}{\left(\sum_{i=1}^{m} M_{k,i}\right)f_i(x_{k+1}) + \left(1-\sum_{i=1}^{m} M_{k,i}\right)f_0(x_{k+1})} \tag{7.7}$$

where $M_{k,i}$ is de¯ned in (7.1c) Relations (7.4) and (7.7) together prove the induction.

The following proposition states that a simple requirement on the initial conditions ensures that the $F_{k,i}$ are consistent with the de¯nition of a probability

Proposition 7.2. The condition $\sum_{i=1}^{m} \pi_i \leq 1$ implies that

$$0 \leq F_{k,i} \leq 1 \qquad \forall\, k$$

and

$$\sum_{i=1}^{m} F_{k,i} \leq 1 \qquad \forall\, k$$

Proof. The proof follows as a direct application of the recursion (7.1).

Note that $F_{k,0} = 1 - \sum_{i=1}^{m} F_{k,i}$. Finally, note that (7.1) reduces to a multiple hypothesis Wald SPRT if $p_i = 0 \;\; \forall i$.

7.2.2 Dynamic Programming Formulation

At each time t_k one of two actions are possible:

1. Terminate the measurement sequence and announce a fault of type i. The cost of making a correct announcement is zero while the cost of a false alarm of type i is Q_i.
2. Take another measurement. The cost of the measurement is C and the cost of a miss-alarm of type i is S_i.

An optimal decision algorithm is derived by minimizing the expected cost at a time t_N. Suppose N measurements are taken and that at time t_N, a type i fault is announced. Assuming further that only one fault may have occurred, the cost is

$$J_{N,i} = (1 - F_{N,i})Q_i$$

so the optimal cost at t_N is

$$J_N^* = \min_i (1 - F_{N,i})Q_i \tag{7.8}$$

The expected cost at time t_{N-1} is

$$J_{N-1,i} \quad = \quad \min\left[(1 - F_{N-1,i})Q_i,\ C + S_i F_{N-1,i} + E_{x_N}[J_N^*/\mathrm{X}_{N-1}]\right]$$

and the optimal cost at t_{N-1} is

$$J_{N-1}^* \quad = \quad \min_i \min\left[(1 - F_{N-1,i})Q_i,\ C + S_i F_{N-1,i} + E_{x_N}[J_N^*/\mathrm{X}_{N-1}]\right]$$

In general, the optimal expected cost at time t_k is

$$J_k^* \quad = \quad \min_i \min\left[(1 - F_{k,i})Q_i,\ C + S_i F_{k,i} + A_k(F_k)\right]$$

where

$$A_k(F_k) \quad \triangleq \quad E_{x_{k+1}}\left[J_{k+1}^*/\mathrm{X}_k\right] \tag{7.9a}$$

$$F_k \quad \triangleq \quad [F_{k,1}, F_{k,2}, \ldots, F_{k,m}]^T \tag{7.9b}$$

The expectation is taken with respect to the conditional probability density functions $f_i(x_{k+1}/\mathrm{X}_k)$.

The optimal policy, one that minimizes the expected cost at each time t_k, is stated with respect to a threshold probability $F_{T_k,i}$:

- If $F_{k,i} \geq F_{T_k,i}$, announce a type i fault.
- If $F_{k,i} < F_{T_k,i}$ for each $i \in \{1, \ldots, m\}$, take another measurement.

The threshold probability $F_{T_k,i}$ is determined at each time t_k as the value at which the expected cost of terminating the test by announcing a fault, and possibly a false alarm, is the same as the expected cost of continuing the test by taking another measurement.

$$(1 - F_{T_k,i})Q_i \quad = \quad C + A_k(F_{T_k}) + S_i F_{T_k,i} \tag{7.10a}$$

with

$$Q_i \quad > \quad C + A_k(0) \tag{7.10b}$$

Unfortunately, determining the threshold probabilities $F_{T_k,i}$ is a numerically intractable problem, even in the scalar case where $m = 1$. This is because the $A_k(F_{T_k})$ expectations are evaluated with respect to the conditional probability density functions $f_i(x_{k+1}/\mathrm{X}_k)$ or (7.6a) in the proof of Proposition 7.1,

$$P(x_{k+1}/\mathrm{X}_k) = \sum_{i=1}^{m} [P(x_{k+1}/\theta_i \leq t_{k+1})P(\theta_i \leq t_{k+1}/\mathrm{X}_k) + P(x_{k+1}/\theta_i > t_{k+1})P(\theta_i > t_{k+1}/\mathrm{X}_k)]$$

The following lemma establishes properties of $A_k(F_{T_k})$ which allow for a tractable policy, one which is optimal in the limit as $(N - k) \to \infty$.

Lemma 7.3. The functions $A_k(F_k)$ satisfy the following properties $\forall k \in \{1, \ldots, m\}$

1. If $\pi_i = 1$ for any $1 \leq i \leq m$, then $A_k(F_k) = 0$
2. $A_k(F_k) \leq A_{k-1}(F_{k-1})$
3. $A_k(F_k)$ is concave

Proof.

Property 1: Note that by the recursion relation (7.1) of Proposition 7.1, $\pi_i = 1 \quad\Rightarrow\quad F_{1,i} = 1$ and $F_{k,i} = 1 \quad\Rightarrow\quad F_{k+1,i} = 1$. By induction, $\pi_i = 1 \quad\Rightarrow\quad F_{k,i} = 1$, $\forall k \in \{1, \ldots, N\}$. Also, note that by Proposition 7.2, $\pi_i = 1 \quad\Rightarrow\quad F_{k,j} = 0$ for $j \neq i$ and $\forall k \in \{1, \ldots, N\}$.

Suppose $\pi_i = 1$ for some i. By the de¯nition of $A_k(F_k)$

$$\begin{aligned} A_{N-1}(F_{N-1}) &= E_{x_N}[J_N^*/\mathrm{X}_{N-1}] && (7.11a) \\ &= E_{x_N}[\min_i(1 - F_{N,i})Q_i/\mathrm{X}_{N-1}] && (7.11b) \\ &= 0 && (7.11c) \end{aligned}$$

since $\pi_i = 1 \quad\Rightarrow\quad F_{N-1,i} = 1$. Now, suppose $A_k(F_k) = 0$ where again $\pi_i = 1$. Then

$$\begin{aligned} A_{k-1}(F_{k-1}) &= E_{x_k}[J_k^*/\mathrm{X}_{k-1}] && (7.12a) \\ &= E_{x_k}\left[\min_i \min\left[(1 - F_{k,i})Q_i,\ C + S_iF_{k,i} + A_k(F_k)\right]/\mathrm{X}_{k-1}\right] && (7.12b) \\ &= 0 && (7.12c) \end{aligned}$$

Relations (7.11) and (7.12) prove property 1 by induction.

Property 2: By the de¯nition of $A_{k-1}(F_{k-1})$,

$$\begin{aligned} A_{k-1}(F_{k-1}) &= E_{x_k}[J_k^*/\mathrm{X}_{k-1}] \\ &= E_{x_k}\left[\min_i \min\left[(1-F_{k,i})Q_i,\ C + S_iF_{k,i} + A_k(F_k)\right]/\mathrm{X}_{k-1}\right] \end{aligned}$$

Since the test terminates at time t_N, it must happen that for the minimizing i,

$$\min\left[(1-F_{k,i})Q_i,\ C + S_iF_{k,i} + A_k(F_k)\right] = C + S_iF_{k,i} + A_k(F_k)$$

So,

$$\begin{aligned} A_{k-1}(F_{k-1}) &= E_{x_k}\left[\min_i \left(C + S_iF_{k,i} + A_k(F_k)\right)/\mathrm{X}_{k-1}\right] \\ &= C + \min_i S_iF_{k,i} + E_{x_k}\left[A_k(F_k)/\mathrm{X}_{k-1}\right] \end{aligned}$$

Therefore,

$$A_{k-1}(F_{k-1}) \geq A_k(F_k)$$

Property 3: Now show that $A_k(\cdot)$ is concave. By inspection of (7.8) and (7.9), J_N^* is concave. Since the test ends at t_N:

$$J_{N-1}^* = C + A_{N-1} + \min_i S_iF_{N-1,i} \tag{7.13}$$

Clearly, J_{N-1}^* is concave if A_{N-1} is concave. Let the elements of the countably in¯nite measurement space be denoted by x_k^j where $j = 1, 2, \ldots, \infty$ and $k = 1, 2, \ldots, N$. From (7.9) and (7.6a) :

$$\begin{aligned} A_k &= \sum_{j=1}^{\infty}\left[\left(\sum_{i=1}^{m} M_{k,i}\right) f_i(x_{k+1}^j) + \left(1 - \sum_{i=1}^{m} M_{k,i}\right) f_0(x_{k+1}^j)\right] J_{k+1}^*(F_{k+1}) \\ &= \sum_{j=1}^{\infty}\sum_{i=1}^{m} h_i^j(F_k) \\ &= \sum_{j=1}^{\infty} h^j(F_k) \end{aligned} \tag{7.14}$$

where

$$h_i^j = \left(M_{k,i}\left[f_i(x_{k+1}^j) - f_0(x_{k+1}^j)\right] + \frac{1}{m} f_0(x_{k+1}^j)\right) J_{k+1}^*(F_{k+1})$$

and where

$$M_{k,i} = F_{k,i} + p_i(1 - F_{k,i})$$

If each of the h_i^j is concave, the summation is concave. Therefore, it remains to show that

$$h^j\left(\lambda F_k^1 + (1-\lambda)F_k^2\right) \geq \lambda h^j(F_k^1) + (1-\lambda)h^j(F_k^2) \tag{7.15}$$

where $\lambda,\ F_k^1,\ F_k^2\ \in [0,1]$. De¯ne

$$\xi_{k,i}^r \triangleq \left[F_{k,i}^r + p_i(1 - F_{k,i}^r)\right]\left[f_i(x_{k+1}^j) - f_0(x_{k+1}^j)\right] + \frac{1}{m} f_0(x_{k+1}^j) \qquad \text{for } r = 1, 2.$$

so that the convexity inequality (7.15) becomes

$$\left[\lambda\xi_{k,i}^1 + (1-\lambda)\xi_{k,i}^2\right] J_{k+1}^*(\widehat{F}_{k+1}) \geq \lambda\xi_{k,i}^1 J_{k+1}^*(F_{k+1}^1) + (1-\lambda)\xi_{k,i}^2 J_{k+1}^*(F_{k+1}^2) \tag{7.16}$$

where $\widehat{F}_{k+1} = \widehat{F}_{k+1}(\lambda, F_k^1, F_k^2)$. Now,

$$F_{k+1,i}^{1,2} = \frac{M_{k,i}^{1,2} f_i(x_{k+1})}{\sum_{s=1}^m \xi_{k,s}^{1,2}} \tag{7.17}$$

$$\widehat{F}_{k+1,i} = \frac{\left[\lambda M_{k,i}^1 + (1-\lambda)M_{k,i}^2\right] f_i(x_{k+1})}{\sum_{s=1}^m \lambda\xi_{k,s}^1 + (1-\lambda)\xi_{k,s}^2} \tag{7.18}$$

Let $\xi_k^{1,2} = \sum_{s=1}^m \xi_{k,s}^{1,2}$. Then, from (7.17) and (7.18)

$$\widehat{F}_{k+1,i} = \frac{\lambda F_{k+1,i}^1 \xi_k^1 + (1-\lambda)F_{k+1,i}^2 \xi_k^2}{\lambda\xi_k^1 + (1-\lambda)\xi_k^2}$$

Take a summation from $i = 1, \ldots, m$ in (7.16) to get

$$J_{k+1}^*\left[\frac{\lambda\xi_k^1 F_{k+1}^1 + (1-\lambda)\xi_k^2 F_{k+1}^2}{\lambda\xi_k^1 + (1-\lambda)\xi_k^2}\right] \geq$$
$$\frac{\lambda\xi_k^1}{\lambda\xi_k^1 + (1-\lambda)\xi_k^2} J_{k+1}^*(F_{k+1}^1) + \frac{(1-\lambda)\xi_k^2}{\lambda\xi_k^1 + (1-\lambda)\xi_k^2} J_{k+1}^*(F_{k+1}^2) \tag{7.19}$$

From (7.8) and (7.9), J_N^* is concave and hence satis¯es (7.19). This implies that A_{N-1} is concave. But from (7.13), J_{N-1}^* is concave if A_{N-1} is concave. Hence, by induction, $A_k(\cdot)$ are concave $\forall k$.

7.2.3 Thresholds for the Optimal policy

Lemma 7.3 showed that the $A_k(F_k)$ are monotonically decreasing in k and bounded because of the concavity property in F. This implies that for an infinite number of stages, that is, $(N-k) \to \infty$, each threshold probability $F_{T_k,i}$ also approaches a limit. To see this, rearrange (7.10) as

$$F_{T_k,i} = \frac{Q_i - C - A_k(F_{T_k})}{Q_i + S_i}$$

Then,

$$F_{T_k,i} \leq F_{T_{k-1},i} \leq \ldots \leq F_{T,i} \leq \frac{Q_i - C}{Q_i + S_i}$$

The dynamic programming algorithm for infinite time reduces to

$$J^*(F) = \min_i \min[(1 - F_{T,i})Q_i,\ C + S_i F_{T,i} + A(F_T)] \tag{7.20}$$

where

$$A(F) =$$

and the threshold probabilities $F_{T,i}$ are determined by

$$(1 - F_{T,i})Q_i = C + S_i F_{T,i} + A(F_T) \tag{7.21}$$

Since $A(F_T)$ is still hard to evaluate, a workaround is proposed. The idea is to choose the $F_{T,i}$ where

$$\alpha_i \triangleq 1 - F_{T,i}$$

are interpreted as false alarm rates and imply unknown Q_i, S_i and C through (7.21). In the context of (Q_i, S_i, C), the Shiryayev SPRT, extended here to multiple hypotheses, gives the minimum stopping time out of the set of stopping times $\{\tau_i\}$. This comes from an interpretation of the dynamic programming algorithm for infinite time (7.20) as a Bayes' risk minimizing cost (Shiryayev 1977)

$$J^*(F) = \min_i \inf_{\tau_i} E\left[(1 - F_{\tau_i,i})Q_i + F_{\tau_i,i}S_i + C \max\{\tau_i - \theta_i, 0\}\right]$$

Here, $E[1-F_{\tau_i,i}]$ is the expected type i false alarm probability and $E[\max\{\tau_i-\theta_i,0\}]$ is the expected delay of detecting a type i fault correctly.

The optimization problem is to minimize the mean time of delay in announcing a type i fault subject to the constraint that the probability of false alarm $\alpha_i = 1 - F_{\tau,i}$ is ¯xed at $\alpha_i = 1 - F_{T,i}$. The quantity $(\frac{S_i - Q_i}{C})$ becomes the Lagrange multiplier.

In the binary hypothesis case, $m = 1$, the Shiryayev SPRT policy is often expressed in a likelihood ratio form (Speyer and White 1984). This allows for an easy comparison with the Generalized Likelihood Ratio Test of (Nikiforov 1995),(Basseville and Nikiforov 1995). De¯ne the likelihood ratio L_k, where the i subscript is dropped because there are only two hypotheses,

$$L_k = \frac{F_k}{1-F_k}$$

and use (7.1) of Proposition 7.1 to develop a recursion relation

$$\begin{aligned} L_0 &= \frac{\pi}{1-\pi} \\ L_{k+1} &= \left[\frac{f_1(x_{k+1})}{f_0(x_{k+1})}\right]\left(\frac{L_k + p}{1-p}\right) \end{aligned}$$

Given a threshold likelihood ratio,

$$L_T = \frac{F_T}{1-F_T}$$

the fault announcement policy becomes

- If $L_k \geq L_T$, announce that a fault has occurred.
- If $L_k < L_T$, take another measurement.

Likelihood ratios can be de¯ned in the obvious way for the multiple hypotheses case $m > 1$

$$\begin{aligned} L_{0,i} &= \frac{\pi_i}{1-\pi_i} \\ L_{k,i} &= \frac{F_{k,i}}{1-F_{k,i}} \end{aligned}$$

but no simple recursion relation can be developed from (7.1) to propagate the likelihood ratios.

7.2.4 Detection of Unknown Changes

The Shiryayev SPRT and the multiple hypotheses generalization, as described above, are developed for measurement sequences with known probability density functions, known both before and after a fault. It is an easy extension to allow the density functions to depend on a scalar unknown parameter α. Assume that the unknown parameter is also a random variable de¯ned over a set - and has probability density function ψ_α(-). Then, the conditional density function of the measurement sequence becomes

$$\begin{aligned} \tilde{f}_i(x) &\triangleq f(x/\mathcal{H}_i) \\ &= \int_\Omega f(x/\mathcal{H}_i, \eta)\; \psi_\alpha(\eta) d\eta \end{aligned} \tag{7.22}$$

Now, replace $f_i(\cdot)$ with the new density function $\tilde{f}_i(\cdot)$ in the recursive relation (7.1). The rest of the analysis remains the same.

7.3 Examples

Before considering the development of a residual processing module for Advanced Vehicle Control Systems, two examples are considered to illustrate the application of a multiple hypothesis Shiryayev SPRT. In the ¯rst example, the measurement sequence is taken as a white noise sequence with one of ¯ve possible means. In the second example, the measurement sequence is modeled as a scalar white noise sequence with unit power spectral density however, the mean is unknown.

7.3.1 Example 1

Here, the measurement sequence is modeled as a scalar white noise sequence with unit power spectral density and one of ¯ve possible means. The ¯ve hypotheses including the null hypothesis are summarized as

$$\mathcal{H}_i : \quad x \sim \mathcal{N}(0.5i, 1) \qquad \text{where } i \in \{0, 1, 2, 3, 4\}$$

For example, introduction of a bias with unit magnitude means the measurement sequence switches from the state $\mathcal{H}_0$ to $\mathcal{H}_2$. Extension to the case of vector valued measurements is trivial.

A simulated white noise measurement sequence is illustrated in Figure 7.1. Each measurement has a Gaussian distribution with unit variance and is uncorrelated with other measurements. During the interval $0 \leq t < 1$ the measurements have zero mean. This is hypothesis $\mathcal{H}_0$. During the interval $1 \leq t \leq 5$ a unit bias is introduced so at $t = 1$, the measurement sequence switches from $\mathcal{H}_0$ to $\mathcal{H}_2$. The posteriori probabilities found from the recursion relation (7.1) and illustrated in Figure 7.1 very clearly show the measurement hypothesis switch. The apriori probabilities π_i are taken as 0.001 for $i \in \{1, 2, 3, 4\}$.

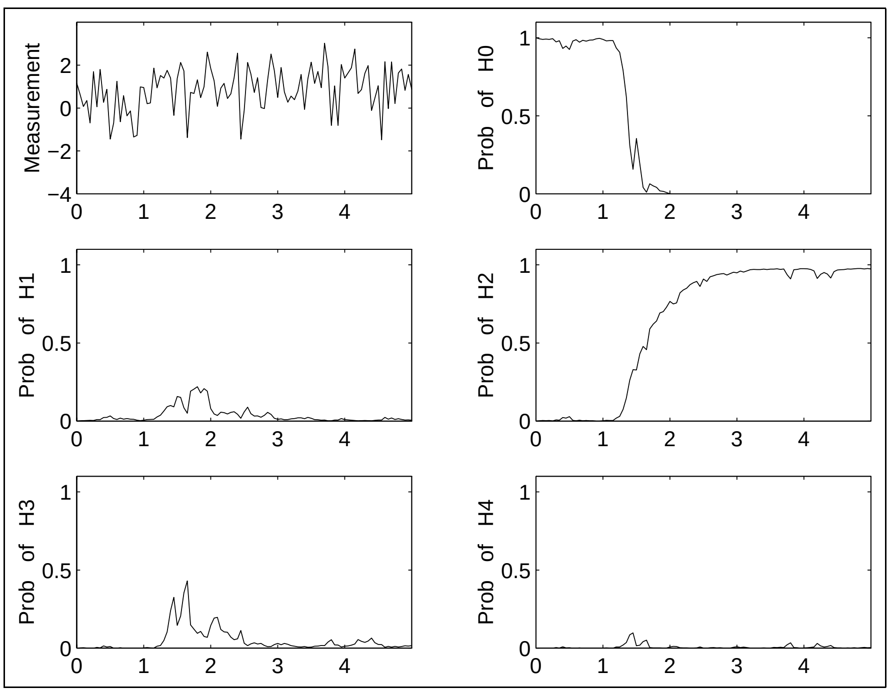

Figure 7.1: Change from $\mathcal{H}_0$to $\mathcal{H}_2$at time t = 1 sec.

7.3.2 Example 2

Again, the measurement sequence is modeled as a scalar white noise sequence with unit power spectral density. However, here the mean is also taken as random variable with one of ¯ve possible uniform distributions. The ¯ve hypotheses including the null hypothesis are summarized as

$$\mathcal{H}_i : x \sim \mathcal{N}(m_i, 1)$$

where

$$\begin{aligned} m_0 &= 0 \\ m_1 &\sim \text{Unif}[0,\ 1] \\ m_2 &\sim \text{Unif}[0.5,\ 1.5] \\ m_3 &\sim \text{Unif}[1,\ 2] \\ m_4 &\sim \text{Unif}[1.5,\ 2.5] \end{aligned}$$

Following are two propositions that provide relations for normally distributed random variables with unknown means. The ¯rst proposition shows that if the measurement means have a Gaussian distribution, the problem reduces to one in which the measurement means are known and the covariances take on a larger value.

Proposition 7.4. Consider a vector valued random variable $x \in \mathrm{R}^n$ where both the mean and the distribution about the mean are Gaussian

$$\begin{array}{lll} x \sim \mathcal{N}(m, ¤_x) & mathrmwherem \in \mathrm{R}^n, & ¤_x \in \mathrm{R}^{n \times n} \\ m \sim \mathcal{N}(m^*, ¤_m) & \text{where} m^* \in \mathrm{R}^n, & ¤_m \in \mathrm{R}^{n \times n} \end{array}$$

Then

$$x \sim \mathcal{N}(m^*, ¤_x + ¤_m)$$

Proof. A proof is provided at the end of this section.

The second proposition provides a probability density function for a Gaussian random variable where the mean has a uniform distribution.

Proposition 7.5. Consider a vector valued random variable $x \in \mathrm{R}^n$ where the mean has a uniform distribution and where the distribution about the mean is Gaussian

$$\begin{aligned} x &\sim \mathcal{N}(m, \Sigma_x) & \text{where} m \in \mathrm{R}^n, \quad & \Sigma_x \in \mathrm{R}^{n \times n} \\ m &\sim \mathrm{Unif}[b, b + 2m^*] & \text{where} b,\ m^* \in \mathrm{R}^n & \end{aligned}$$

Then the probability density function $f(x)$ is

$$f(x) = \frac{1}{4^n \prod_j m_j^*} \left[\mathrm{erf}\{\frac{1}{\sqrt{2}} \Sigma_x^{-0.5}(x - b)\} - \mathrm{erf}\{\frac{1}{\sqrt{2}} \Sigma_x^{-0.5}(x - b - 2m^*)\} \right]$$

where

$$m_j^* = [m_1^*, \ldots, m_n^*]^T$$

Note that a property of the error function $\mathrm{erf}(x)$ is that for $x \in \mathrm{R}^n$

$$\mathrm{erf}(x) = \mathrm{erf}(x_1)\,\mathrm{erf}(x_2) \cdots \mathrm{erf}(x_n)$$

Proof. A proof is provided at the end of this section.

A simulated measurement sequence is illustrated in Figure 7.2. Each measurement has a Gaussian distribution with unit variance and is uncorrelated with other measurements. During the interval $0 \le t < 1$ the measurements have zero mean. This is hypothesis $\mathcal{H}_0$. During the interval $1 \le t \le 5$ a constant but unknown bias with uniform distribution Unif[0.5, 1.5] is introduced. Thus, at $t = 1$, the measurement sequence switches from $\mathcal{H}_0$ to $\mathcal{H}_2$. The measurements were generated as

$$x = n + s$$

where

$$\begin{aligned} n &\sim \mathcal{N}(0,\ 1) \\ s &= \begin{cases} 0 & 0 \le t < 1 \\ \mathrm{Unif}[0.5,\ 1.5] & 1 \le t \le 5 \end{cases} \end{aligned}$$

As in the previous example, the posteriori probabilities found from the recursion relation (7.1) and illustrated in Figure 7.2 very clearly show the measurement hypothesis switch. Again, the apriori probabilities π_i are taken as 0.001 for $i \in \{1, 2, 3, 4\}$.

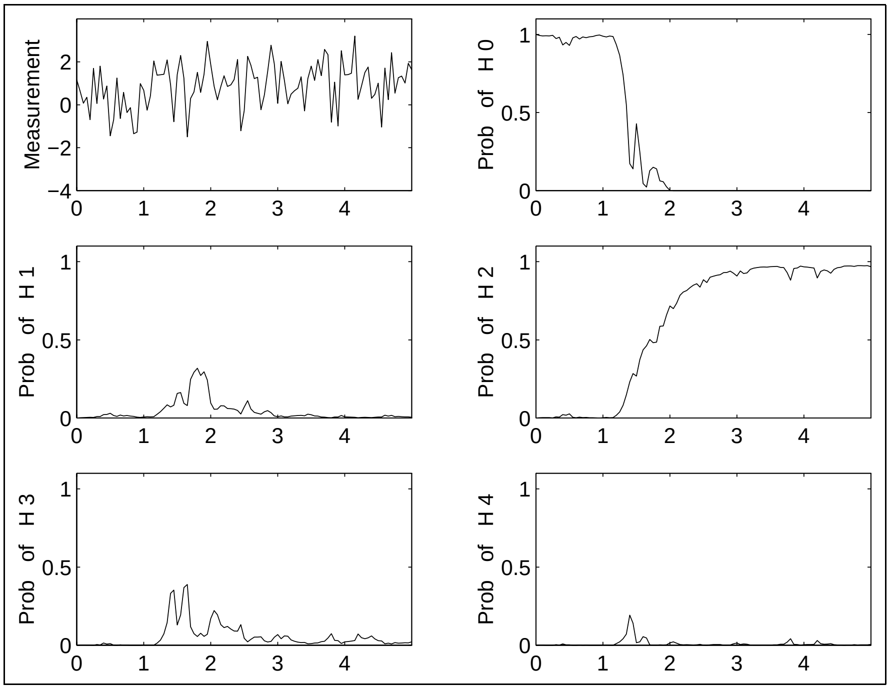

Figure 7.2: Change from $\mathcal{H}_0$ to $\mathcal{H}_2$ at time t = 1 sec.

Proof. (Of Proposition 7.4) From (7.22)

$$\begin{aligned} f(x/\mathcal{H}_i) &= \int_{\mathrm{R}^n} f(x/\mathcal{H}_i, m_i)\psi(m_i)dm_i \\ &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \frac{1}{2\pi^n |\Lambda_{x_i}\Lambda_{m_i}|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}\|x - m_i\|^2_{\Lambda^{-1}_{x_i}} + \frac{1}{2}\|m_i - m_i^*\|^2_{\Lambda^{-1}_{m_i}}\right\} |dm_i| \\ &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \frac{1}{2\pi^n |\Lambda_{x_i}\Lambda_{m_i}|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}D\right\} |dm_i| \qquad (7.23) \end{aligned}$$

Define

$$C_1 = \Lambda^{-1}_{x_i} + \Lambda^{-1}_{m_i} \qquad (7.24a)$$

$$C_2 = \Lambda^{-1}_{x_i} x + \Lambda^{-1}_{m_i} m_i^* \qquad (7.24b)$$

$$C_3 = x^T \Lambda^{-1}_{x_i} x + m_i \Lambda^{-1}_{m_i} m_i \qquad (7.24c)$$

$$D = m_i^T C_1 m_i - 2m_i^T C_2 + C_3 \qquad (7.24d)$$

Then

$$D = \|m_i - C_1^{-1}C_2\|_{C_1}^2 + \left[C_3 - \|C_2\|_{C_1^{-1}}^2\right]$$

We note that since $¤_{x_i}$ and $¤_{m_i}$ are covariance matrices, they are invertible and so is C_1. Now, from (7.23)

$$f(x/\mathcal{H}_i) = \frac{\exp\left\{-\frac{1}{2}(C_3 - \|C_2\|_{C_1^{-1}}^2)\right\}}{2\pi^n|¤_{x_i}¤_{m_i}|^{\frac{1}{2}}} \int\limits_{\mathrm{R}^n} \exp\left\{-\frac{1}{2}\|m_i - C_1^{-1}C_2\|_{C_1}^2\right\}|dm_i|$$

Now change the variable m_i. Let

$$\boldsymbol{m}_i = \frac{1}{\sqrt{2}}C_1^{0.5}[m_i - C_1^{-1}C_2]$$

so that

$$f(x/\mathcal{H}_i) = \frac{1}{2\pi^{\frac{n}{2}}|C_1¤_{x_i}¤_{m_i}|}^{\frac{1}{2}} \exp\left\{-\frac{1}{2}(C_3 - \|C_2\|_{C_1^{-1}}^2)\right\}$$

Now from (7.24) it follows that

$$\begin{aligned} C_1¤_{x_i}¤_{m_i} &= ¤_{x_i} + ¤_{m_i} \\ C_3 - \|C_2\|_{C_1^{-1}}^2 &= x^TAx + m_i^{*T}Bm_i^* - 2x^TC \cdot m_i^* \end{aligned}$$

where

$$\begin{aligned} C &= ¤_{x_i}^{-1}(¤_{x_i}^{-1} + ¤_{m_i}^{-1})^{-1}¤_{m_i}^{-1} \\ &= \left[¤_{m_i}(¤_{m_i}^{-1} + ¤_{x_i}^{-1})¤_{x_i}\right]^{-1} \\ &= (¤_{x_i} + ¤_{m_i})^{-1} \\ A &= ¤_{x_i}^{-1} - ¤_{x_i}^{-1}(¤_{x_i}^{-1} + ¤_{m_i}^{-1})^{-1}¤_{x_i}^{-1} \\ &= ¤_{x_i}^{-1} - ¤_{x_i}^{-1}¤_{m_i}¤_{m_i}^{-1}(¤_{x_i}^{-1} + ¤_{m_i}^{-1})^{-1}¤_{x_i}^{-1} \end{aligned}$$

Therefore,

$$\begin{aligned} ¤_{x_i}A &= I - ¤_{m_i}(¤_{x_i} + ¤_{m_i})^{-1} \\ &= [(¤_{x_i} + ¤_{m_i}) - ¤_{m_i}](¤_{x_i} + ¤_{m_i})^{-1} \end{aligned}$$

so that

$$\begin{aligned} A &= (\Lambda_{x_i} + \Lambda_{m_i})^{-1} \\ B &= \Lambda_{m_i}^{-1} - \Lambda_{m_i}^{-1}(\Lambda_{x_i}^{-1} + \Lambda_{m_i}^{-1})^{-1}\Lambda_{m_i}^{-1} \\ &= (\Lambda_{x_i} + \Lambda_{m_i})^{-1} \end{aligned}$$

This implies

$$f(x/\mathcal{H}_i) = \frac{1}{2\pi^{\frac{n}{2}}|\Lambda_{x_i} + \Lambda_{m_i}|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}\|x - m_i^*\|^2_{(\Lambda_{x_i}+\Lambda_{m_i})^{-1}}\right\}$$

and finally that

$$x \sim \mathcal{N}(m_i^*, \Lambda_{x_i} + \Lambda_{m_i})$$

Proof. (Of Proposition 7.5) From (7.22)

$$f(x/\mathcal{H}_i) = \int_{\mathrm{R}^n} f(x/\mathcal{H}_i, m_i)\psi(m_i)dm_i$$

If the mean m_i has a uniform distribution, then

$$\psi(m_i) = \frac{1}{2^n \prod_{j=1}^n m_{ij}^*} \qquad b_i \le m_i \le b_i + 2m_i^* \qquad \text{where } m_i^* = [m_{i1}^*, \ldots, m_{in}^*]^T \tag{7.25}$$

From (7.22) and (7.25)

$$f(x/\mathcal{H}_i) = \frac{1}{2\pi^n|\Lambda_{x_i}|^{\frac{1}{2}}} \int_{b_i}^{b_i+2m_i^*} \exp\left\{-\frac{1}{2}\|x_i - m_i\|^2_{\Lambda_{x_i}^{-1}}\right\}|dm_i|$$

Now change the variable m_i

$$\begin{aligned} \mathfrak{m}_i &= \frac{1}{\sqrt{2}}\Lambda_{x_i}^{-0.5}(m_i - x_i) \\ f(x/\mathcal{H}_i) &= \frac{1}{4^n \prod_j m_{ij}^*}\frac{2^n}{\pi^{\frac{n}{2}}} \int_{\frac{1}{\sqrt{2}}\Lambda_{x_i}^{-0.5}(b_i - x_i)}^{\frac{1}{\sqrt{2}}\Lambda_{x_i}^{-0.5}(b_i+2m_i^*-x_i)} \exp\left\{-\|\mathfrak{m}_i\|^2\right\}|dm_i| \end{aligned}$$

The desired result follows as

$$f(x/\mathcal{H}_i) = \frac{1}{4^n \prod_j m_{ij}^*}\left[\mathrm{erf}\left\{\frac{1}{\sqrt{2}}\Lambda_{x_i}^{-0.5}(x - b_i)\right\} - \mathrm{erf}\left\{\frac{1}{\sqrt{2}}\Lambda_{x_i}^{-0.5}(x - b_i - 2m_i^*)\right\}\right]$$

7.4 Application to Advanced Vehicle Control Systems

In this section, a multiple hypothesis Shiryayev SPRT residual processor is applied to the same fault detection ¯lters as the Bayesian neural networks of Section 6.4. These fault detection ¯lters are designed with the Berkeley nonlinear vehicle simulation operating at $27\frac{\mathrm{m}}{\mathrm{sec}}$ on a straight road. Vehicle lateral dynamics are not considered. A complete description of the fault detection ¯lter design is in (Douglas et al. 1995). Figure 7.3 shows the residual processing scheme using the multiple hypothesis Shiryayev SPRT and the fault detection ¯lters for the longitudinal simulation.

A detailed description of the modeled sensor and actuator faults can be found in (Douglas et al. 1995). Recall that the vehicle longitudinal model has seven two-dimensional sensor faults and two three-dimensional actuator faults. These are combined in output separable and mutually detectable groups with seven or fewer directions. The following list shows the fault groups with fault notation as indicated in Figure 7.3.

Fault detection ¯lter 1.

(M) : Manifold air mass sensor.

(W) : Engine speed sensor.

(X) : Forward acceleration sensor.

Fault detection ¯lter 2.

(T) : Heave acceleration sensor.

(Fs) : Rear symmetric wheel speed sensor.

(Rs) : Forward symmetric wheel speed sensor.

Fault detection ¯lter 3.

(T) : Pitch rate sensor.

(Z) : Heave acceleration sensor.

(Rs) : Rear symmetric wheel speed sensor.

Fault detection filter 4.

$(alfa)$: Throttle angle actuator.

(Tb) : Brake torque actuator.

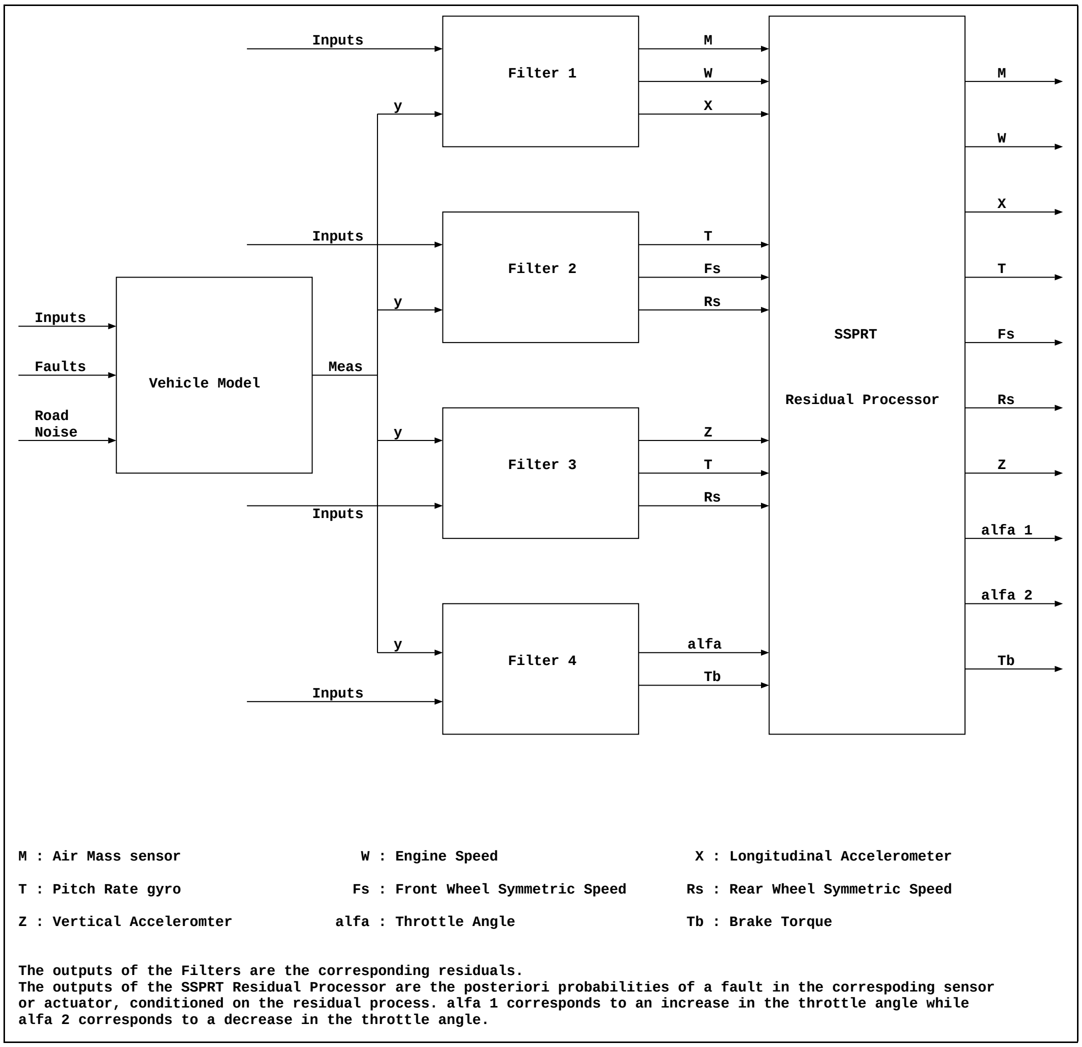

Figure 7.3: Fault detection scheme for AVCS.

A residual processor design should focus on resolving two issues. First, when residuals are driven by model uncertainties, nonlinearities, sensor noise and dynamic disturbances such as road noise, a nonzero residual need not indicate that a fault has occurred. The residual processor should distinguish between a nonzero residual driven by a fault and a nonzero residual driven by something else.

Second, when a fault occurs and the fault is not one included in the fault detection ¯lter design, the directional properties of the residual are unde¯ned. The residual processor should recognize the pattern of a design fault and ignore all other patterns.

Both issues are addressed by a multiple hypothesis SSPRT residual processor. Consider each fault direction as corresponding to a particular hypothesis. Thus, in the present application, there are ten hypotheses $\{\mathcal{H}_0, \ldots, \mathcal{H}_9\}$. Now consider the fault detection ¯lter residual sequence as the measurement sequence for the SPRT. In the present application, the measurement sequence $\{x_k \in \mathrm{R}^{11}\}$ is assumed to be conditionally independent and gaussian. The density functions for all hypotheses are constructed by computing the sample means and covariance matrices. Finally consider that a step fault models a sudden increase in the mean of the residual process while a ramp fault models a gradual increase in the mean. For the detection and identi¯cation of an unknown fault size, the mean of the residual process was assumed to be uniformly distributed.

As an example, step faults are considered in the pitch rate gyro, vertical accelerometer and longitudinal accelerometer. For simplicity, only the residuals corresponding to the particular fault direction are shown in the ¯gures. Figure 7.4 shows a step fault of size $0.05\frac{\mathrm{rad}}{\mathrm{sec}}$ in the pitch rate sensor occurring at 8 seconds. Note that the posteriori probability of a fault in the pitch rate sensor jumps to one almost immediately after the fault occurs. The posteriori probabilities of faults in other sensors and actuators are zero and are not shown.

Figure 7.5 shows a step fault of size $0.5\frac{\mathrm{ft}}{\mathrm{sec}^2}$ in the vertical accelerometer occurring at 8 seconds. Again, the posteriori probability of a fault in the vertical accelerometer jumps to one almost immediately after the fault occurs.

Figure 7.6 shows a step fault of size $0.1\frac{\mathrm{ft}}{\mathrm{sec}^2}$ in the longitudinal accelerometer occurring at 8 seconds. Once again, the posteriori probability of a fault in the longitudinal accelerometer jumps to one almost immediately after the fault occurs.

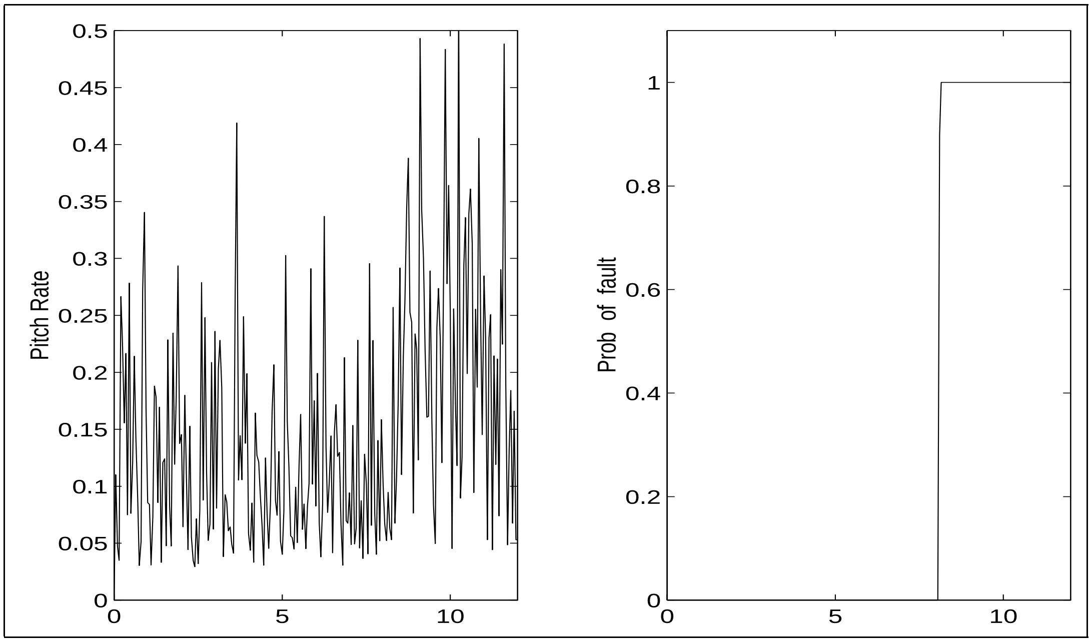

Figure 7.4: Pitch rate sensor fault occurs at 8 sec.

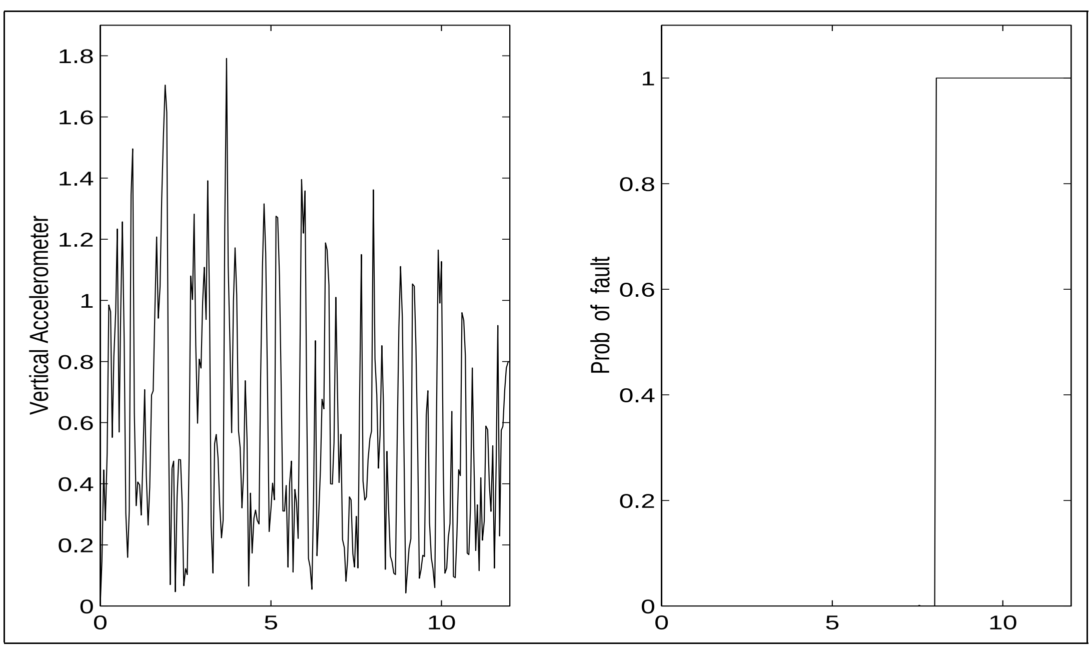

Figure 7.5: Vertical accelerometer fault occurs at 8 sec.

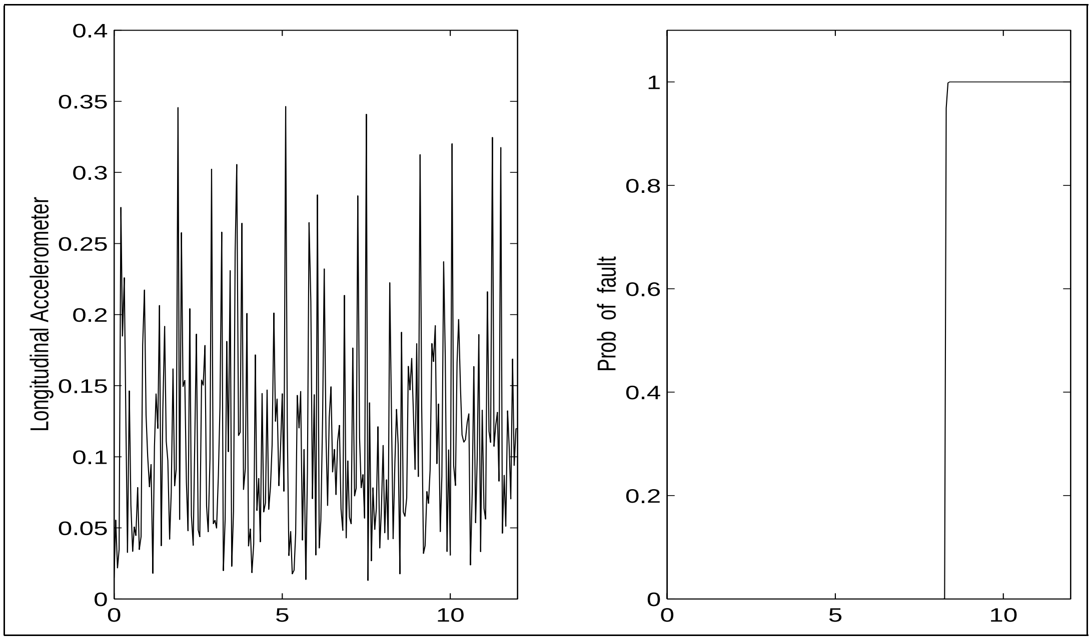

Figure 7.6: Longitudinal accelerometer fault occurs at 8 sec.

7.5 Summary of SPRT Development and Application

A multiple hypothesis SSPRT is derived for the detection and isolation of changes in a conditionally independent measurement sequence. The recursive relation which propagates the posteriori probabilities of all hypotheses requires an approximate knowledge of their apriori probabilities π_i and the probability of change of state p_i from $\mathcal{H}_0$ to $\mathcal{H}_i$. This is not considered as an impediment as the test is found to be insensitive to both parameters as long as they assume reasonable values. The derivation makes no assumption about the structure of the density functions corresponding to all hypotheses and hence, the measurement sequence can be quite general. The generalized Shiryayev SPRT is found to be extremely sensitive to changes even when the underlying density functions for the hypotheses overlap to a large extent. This enhances applicability to practical situations where the fault sizes are typically unknown.

Chapter 8

Vehicle Nonlinear Equations of Motion

A six degree of freedom nonlinear vehicle model is developed independently of the model used for the Berkeley simulation of Section 2 and described in (Peng 1992). This e®ort is a continuation of the work reported in (Douglas et al. 1995). The original motivation for an independent derivation was to be sure that all assumptions, de¯nitions and issues which underlie the Berkeley simulation model were well understood. This exercise proved worthwhile in that some di®erences between the model described here and the Berkeley model were uncovered. The most notable di®erence relates to assumptions made in the Berkeley model that make it di±cult to modify to allow for changes in road slope and superelevation. These assumptions include small angle approximations, a planar road surface and that the road gradient is the same for all four wheels. These modi¯cations are needed, for example, in the design and robustness evaluation of the health monitoring system described in Sections 3 through 7. Various other vehicle models are available, for example, in (Hedrick et al. 1993, Lukowski et al. 1990, Lukowski and Medeksza 1992, Peng 1992, Smith and Starkey 1992, Willumeit et al. 1992). But in each, some feature is missing that is important to health monitoring applications.

A common and economical approach to vehicle dynamics model development is to make simplifying assumptions and to neglect various features of the vehicle system when the loss in fidelity does not significantly affect the application of the model. For example, vehicle models developed by Smith *et al.* (Smith and Starkey 1992) use the load transfer method to model the suspension characteristics. The load transfer method models a load redistribution at the four suspension supports when the vehicle accelerates or corners. When the vehicle accelerates, the load shifts between the front and the rear suspension supports. When the vehicle corners, there is a lateral acceleration and the load shifts between the left and right suspension supports. With the load transfer approach, development of the governing equations is simplified because the suspension characteristics are not modeled directly. Model fidelity is adequate when the road is smooth and flat and when a model of the vertical motion is not important.

In the following model development, the approach is to derive the full equations of motion while making as few approximations as possible. Simplifications as allowed by specific applications are introduced later. Two features included here that are not part of the Berkeley model are a steering system and a road noise model.

This section is organized as follows. Section 8.1 contains a derivation of the vehicle longitudinal dynamics and the various subcomponents of the vehicle. In the longitudinal model, motion is restricted to longitudinal and vertical translation and pitch rotation. The applied forces and moments include those of the suspension model, the aerodynamics model, the tire traction model, the brake model, and the engine model.

Section 8.2 deals with the derivation of the full six degree of freedom vehicle model. All vehicle dynamics modes are included: longitudinal, lateral and vertical translations and roll, pitch and yaw rotations. Including kinematic relations, the system of equations is 12^{th} order. In addition, subcomponents from the longitudinal model are generalized to the full nonlinear model and a steering system and road noise model are added.

Section 8.3 presents the simulation results of the longitudinal model and the full model. In one simulation study, a comparison is made between the responses of the full nonlinear model and nonlinear model modi¯ed with small angle approximations. The study shows that small angle approximations do not contribute signi¯cant errors and are a reasonable model simpli¯cation. In another simulation study, linearized models from various operating points are obtained. Their responses are compared to those of the nonlinear model to ¯nd the size of an acceptable linear operating region. The MatLab^{TM} computer simulation codes used in Section 8.3 are available in (Nguyen 1996).

8.1 NonLinear Longitudinal Vehicle Model

In order to gain a better understanding of vehicle dynamics and to have a simple model for simulation, a longitudinal vehicle dynamics model is developed ¯rst. In the longitudinal model, motion is restricted to longitudinal and vertical translation and pitch rotation. These dynamics couple with the engine, brake, suspension, and wheel rotational dynamics .

8.1.1 Reference Frames

Figure 8.1 shows the de¯nition of coordinates and variables of the longitudinal model. First an Earth-¯xed frame E with origin $\mathcal{O}$ is de¯ned with unit vectors $\left(\underline{e}_x, \underline{e}_y, \underline{e}_z\right)$, where $\underline{e}_y$ points into the page. Next de¯ne the vehicle-¯xed frame, having the origin C at the vehicle center of mass, with unit vectors $\left(\underline{c}_x, \underline{c}_y, \underline{c}_z\right)$ along the vehicle's principal axes. This vehicle-¯xed frame is obtained by rotating the Earth-¯xed frame around its axis by an angular displacement θ, the pitch angle. Finally two sets of road axes are used to describe the road surface at the front and the rear wheels. These axes are described by the unit vectors $\left(\underline{r}_{x_i}, \underline{r}_{y_i}, \underline{r}_{z_i}\right)$ with i = 1 and 2 referring to front and rear wheels, respectively. These road-¯xed frames with unit vectors $\left(\underline{r}_{x_i}, \underline{r}_{y_i}, \underline{r}_{z_i}\right)$ are obtained by rotating the Earth-¯xed frame by an amount ¢$_i\underline{e}_y$. Hence the coordinate transformation matrices are

$$\begin{bmatrix} \underline{c}_x \\ \underline{c}_y \\ \underline{c}_z \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} \underline{e}_x \\ \underline{e}_y \\ \underline{e}_z \end{bmatrix} \tag{8.1}$$

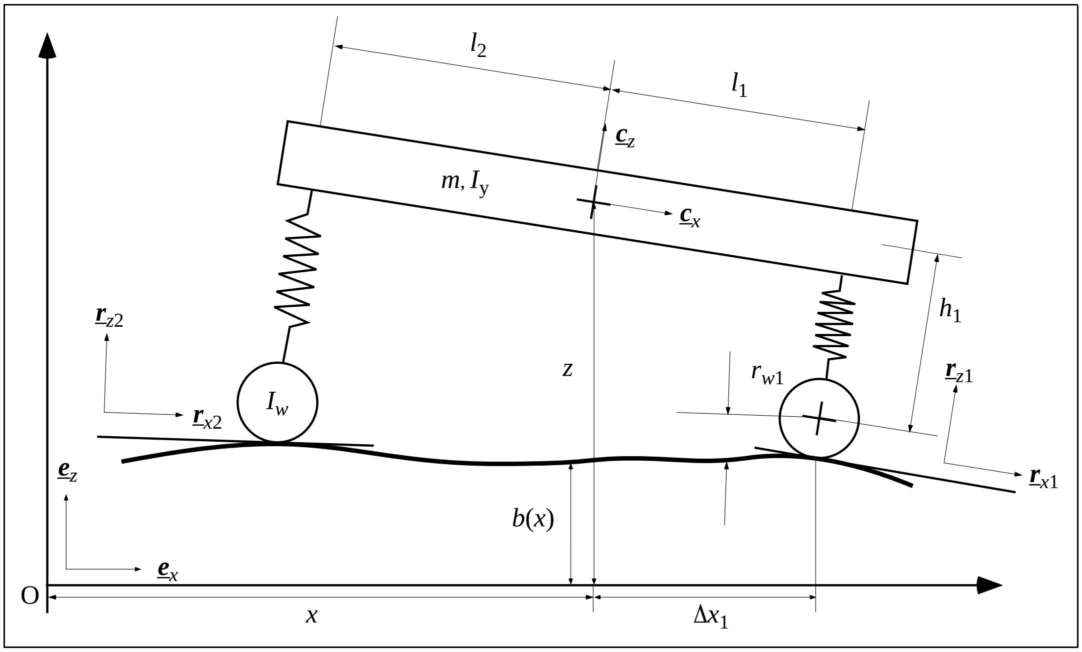

Figure 8.1: Vehicle con¯guration for the nonlinear longitudinal model.

$$\begin{bmatrix} \underline{r}_{x_i} \\ \underline{r}_{y_i} \\ \underline{r}_{z_i} \end{bmatrix} = \begin{bmatrix} \cos ¢_i & 0 & -\sin ¢_i \\ 0 & 1 & 0 \\ \sin ¢_i & 0 & \cos ¢_i \end{bmatrix} \begin{bmatrix} \underline{e}_x \\ \underline{e}_y \\ \underline{e}_z \end{bmatrix} \qquad i = 1, 2 \tag{8.2}$$

Note that the subscript i will be used from now on to refer to quantities that have front and rear components.

8.1.2 Vehicle Dynamics

The dynamic equations of motion are derived from Newton's law applied in an inertial reference frame. The pitch dynamics are derived ¯rst. The longitudinal and vertical translation dynamics follow.

Rotational Equations of Motion

The angular velocity of the vehicle relative to the Earth-¯xed frame is given by

$$\underline{\omega} = \dot{\theta}\underline{e}_y \tag{8.3}$$

$$= \omega_y \underline{e}_y \tag{8.4}$$

The rotational kinematic equation becomes

$$\dot{\theta} = \omega_y \tag{8.5}$$

The angular acceleration follows by taking the time derivative of Equation (8.3),

$$\dot{\underline{\omega}} = \dot{\omega}_y \underline{e}_y \tag{8.6}$$

Hence the rotational dynamic equation of motion is obtained from Euler's equation,

$$\dot{\omega}_y = \frac{M_y}{I_y} \tag{8.7}$$

where M_y, which will be derived later in Section 8.1.5, is the y-axis component of the total moment applied about the vehicle center of mass by suspension and aerodynamic forces and I_y is the moment of inertia of the sprung mass around the same y-axis. The sprung mass is the portion of the vehicle that is supported by the suspension system. The remaining portion which includes the drivetrain and the wheel assemblies is known as the unsprung mass.

Translational Equations of Motion

Let $\underline{P}_{\mathrm{CM}} = x\underline{e}_x + z\underline{e}_z$ be the position vector from the Earth-¯xed origin $\mathcal{O}$ to the vehicle center of mass as seen in Figure 8.1, then the velocity of the mass center can be expressed either in Earth-¯xed or vehicle-¯xed coordinates as

$$\underline{v}_{\mathrm{CM}} = \dot{x}\underline{e}_x + \dot{z}\underline{e}_z \tag{8.8}$$

$$= v_x\underline{c}_x + v_z\underline{c}_z \tag{8.9}$$

Applying coordinate transformation Equation (8.1) to Equation (8.9), we obtain

$$\underline{v}_{\mathrm{CM}} = (v_x \cos\theta + v_z \sin\theta)\,\underline{e}_x + (-v_x \sin\theta + v_z \cos\theta)\,\underline{e}_z \tag{8.10}$$

The translational kinematic equations then follow immediately from (8.8) and (8.10)

$$\dot{x} = v_x \cos\theta + v_z \sin\theta \tag{8.11}$$

$$\dot{z} = -v_x \sin\theta + v_z \cos\theta \tag{8.12}$$

The acceleration of the vehicle center of mass can be found by differentiating (8.9).

$$\begin{aligned} \underline{a} &= \dot{v}_x \underline{c}_x + \dot{v}_z \underline{c}_z + \omega_y \underline{c}_y \times (v_x \underline{c}_x + v_z \underline{c}_z) \\ &= (\dot{v}_x + \omega_y v_z)\, \underline{c}_x + (\dot{v}_z - \omega_y v_x)\, \underline{c}_z \end{aligned} \tag{8.13}$$

If the total external force, $\underline{F} = F_x \underline{c}_x + F_z \underline{c}_z$, applied to the vehicle is known, the translational dynamic equations are obtained from Newton's second law,

$$\dot{v}_x = -\omega_y v_z + \frac{F_x}{m} \tag{8.14}$$

$$\dot{v}_z = \omega_y v_x + \frac{F_z}{m} \tag{8.15}$$

where m is the sprung mass of the vehicle. The vehicle unsprung mass is neglected throughout this work. If it were not, the mass term in Equation (8.15) would need to be modified to account for the vehicle unsprung mass. The forces F_x and F_z will be derived later in Section 8.1.4.

8.1.3 Suspension Model

The suspension and tire assembly is modeled as shown in Figure 8.2. The spring and dashpot in the upper portion represent the suspension, while the spring in the lower portion models the tire stiffness. At any instant, the orientation of the tire spring K_w is assumed to be normal to the road surface. The tire damping behavior and its mass are neglected. The exclusion of the tire mass and its damping characteristic will allow a higher portion of high-frequency noise to pass from the road to the sprung mass. Note that the suspension height, h_i, is defined as the distance along the vehicle axis $\underline{c}_z$ measured from the tire center to the vehicle center of mass and not as the length of the spring.

In simulations where the road surface is a straight line, as seen on the left half of Figure 8.3, the relationship between the tire radius r_{w_i} and the suspension height h_i can be easily found using a geometric approach by summing all the vectors in a loop. The loop starts from the vehicle center of mass, goes to the tip of the suspension, down to the road, follows back along the road surface, and returns to the vehicle center of mass. Following

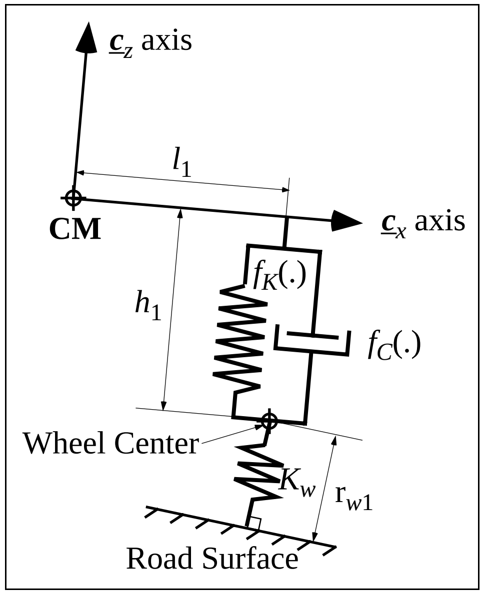

Figure 8.2: Schematic view of suspension and tire models showing the front half of the vehicle.

this path leads to:

$$[l_i\underline{c}_x - h_i\underline{c}_z - (\xi_i + r_{w_i})\,\underline{r}_z - y_i\underline{r}_x + (z - b(x))\,\underline{e}_z] \cdot \underline{r}_z = 0 \qquad i = \{1, 2\} \tag{8.16}$$

where l_i is the half wheelbase from the center of mass to the i^{th} wheel, ξ_i represents road variations which can be used to model bumps, potholes, road noise and any other road irregularities, and $b(x)$ is a function describing the road height at any location x. Furthermore l_i is positive whereas l_2 is negative since l_2 points in the negative $\underline{c}_x$ direction. The reason for naming the wheelbase in a vector format is that the simulation code can be written more compactly.

Using equations (8.1) and (8.2) to transform Equation (8.16), a relationship between tire radius and the suspension height is found

$$r_{w_i} + h_i \cos(\theta - \text{¢}) = (z - b(x)) \cos \text{¢} - l_i \sin(\theta - \text{¢}) - \xi_i \qquad i = \{1, 2\} \tag{8.17}$$

The relationship between the tire radius and the suspension height in situations involving varying road surface can be found by going around a similar loop as seen on the right half of

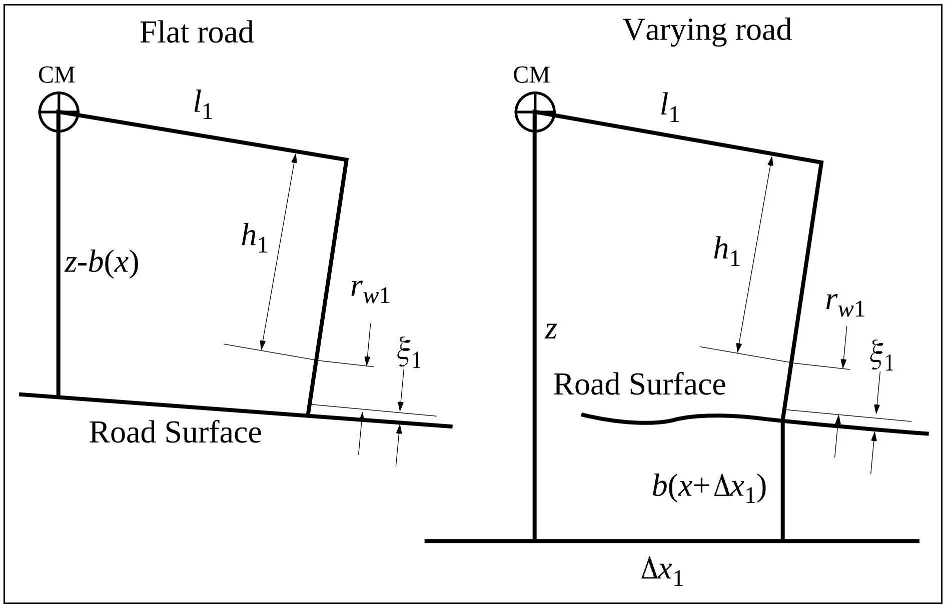

Figure 8.3: Geometric constraints involving the suspension height showing the front half of the vehicle for planar and arbitrary road surfaces.

Figure 8.3. However, solving for the suspension height requires solving a nonlinear equation,

$$l_i \underline{c}_x - h_i \underline{c}_z - (\xi_i + r_{w_i})\, \underline{r}_{z_i} - b(x + ¢\, x_i)\underline{e}_z - ¢\, x_i \underline{e}_x + z\underline{e}_z = 0 \qquad i = \{1, 2\} \tag{8.18}$$

in which relative tire position $¢\, x_i$, the suspension height, and the wheel radius are not independent. An additional equation is required to provide a relationship between the tire radius and the suspension height in order to yield a unique solution in the equation above. This additional equation comes from a single state equation using a force balance and the assumption of a massless wheel. If the wheel is assumed to be massless, the total force applied at the center of the wheel must vanish an any direction. Consider the all the forces in the $\underline{c}_x$ direction. The tire force in the $\underline{c}_x$ direction must balance the suspension force which is generated by the suspension spring and damper. This leads to the following state equation involving the suspension height.

$$-K_w\, (r_{w_i} - r_{w_0}) \cos(\theta - ¢_i) = f_K(h_i) + f_C(\dot{h}_i) \qquad i = \{1, 2\} \tag{8.19}$$

where $f_K(\cdot)$ and $f_C(\cdot)$ are functions describing the force response of suspension spring and damper, respectively. These functions will be specified in the next section. Depending on the damping function, we can solve for h_i in closed-form if the function $f_C(\cdot)$ is invertible; otherwise we will have to approximate.

With the addition of Equation (8.19), Equation (8.18) now contains two unknown but dependent variables, which are the relative tire position and the wheel radius. There is no closed-formed solution to Equation (8.18) if the road surface is arbitrary.

Two methods of solving this nonlinear equation have been examined. The first approach uses a nonlinear equation solver routine to approximate the solution. The generality and flexibility of the routine supplied with MatLab^{TM} causes this application to require a prohibitively long computation time. The second approach is to exploit some of the special properties inherent in the system to make some approximations so that the relative tire position and the wheel radius can be determined. Consider the most general situation where the vehicle is traveling on an arbitrary road surface. By taking the dot product of Equation (8.18) with unit vector $\underline{e}_x$, the relative tire position can be expressed as

$$\Delta x_i = l_i \cos\theta - h_i \sin\theta - (r_{w_i} + \xi_i) \sin \Delta_i \qquad i = \{1, 2\} \tag{8.20}$$

where r_{w_i} and Δ_i are functions of x_i. It is not possible to solve this equation analytically. However, by examining the last term closely, one can make some reasonable assumptions which permit an approximate solution. First the road variation is assumed to be zero. Since the wheel stiffness constant is very high, it is reasonable to assume that the wheel radius is equal to the nominal wheel radius at equilibrium. Furthermore to eliminate the dependency of the road angle on the relative tire location, we will assume that the road angle Δ_i is approximately the same as at the position where the center of the wheel projects down to the road surface. In the worst case scenario where the road elevation is taken to be 15%, the deviation between the real location and the assumed location where the road elevation is used is at most 5 cm. This is a very small distance for the road elevation to vary significantly. Hence the solution for the relative tire position can be approximated as

$$\Delta x_i = l_i \cos\theta - h_i \sin\theta - r_{w_0} \sin \Delta_i \qquad i = \{1, 2\} \tag{8.21}$$

where the road angle ¢$_i$ is evaluated at the projection of the wheel center down to the road surface. This point can be expressed as $x + l_i \cos\theta - h_i \sin\theta$.

Once the relative tire position is known, the approximate wheel radius can be obtained from Equation (8.18) by taking the dot product with unit vector $\underline{r}_{z_i}$ at the point of contact between the tire and the road surface. This leads to:

$$r_{w_i} = -l_i \sin(\theta - ¢_i) - h_i \cos(\theta - ¢_i) - \xi_i + (z - b(x + ¢x_i)) \cos ¢_i - ¢x_i \sin ¢_i, \quad i = \{1, 2\}$$

where the road angle ¢$_i$ is evaluated at the approximated tire position.

8.1.4 Forces

The forces developed in this section include the gravitational force, aerodynamic forces, and suspension forces. The gravitational force on the vehicle is expressed as

$$\begin{aligned} \underline{F}_g &= -mg\underline{e}_z \\ &= -F_g\underline{e}_z \end{aligned} \tag{8.22}$$

When the vehicle longitudinal speed is large or high wind speed is present, air drag plays a signi¯cant role. The longitudinal drag is proportional to the square of the relative wind speed, $v_{wr} = v_w - v_x$, that is, the di®erence between the wind speed and vehicle speed, and has the same direction as the relative wind speed,

$$\begin{aligned} \underline{D} &= \frac{1}{2} C_D A_f \rho_a v_{wr}^2 \underline{c}_x \\ &= D\underline{c}_x \end{aligned} \tag{8.23}$$

where C_D is the drag coe±cient, A_f is the vehicle e®ective frontal area, and ρ_a is the air density. The sign of the coe±cient determines the direction of the drag force based on the direction of the relative wind speed. In addition, there is also a lift component due to the asymmetric shape of the top and bottom of the vehicle. The lift force can be described by the following equation,

$$\begin{aligned} \underline{L} &= \frac{1}{2} C_L A_f \rho_a v_{wr}^2 \underline{c}_z \\ &= L\underline{c}_z \end{aligned} \tag{8.24}$$

where C_L is the lift coefficient. These drag and lift coefficients are specific to each vehicle. However one can generalize to a class of vehicles, such as sedans, sport cars and vans. Data for these coefficients obtained by Yip *et al.* (Yip et al. 1992) for typical sedans is used in the simulation. Both the drag and lift forces are assumed to act at the vehicle center of mass.

Here the relative velocity is assumed to be negative. If it were not, equations (8.23) and (8.24) would have to be modified to account for situations where v_{wr} is positive. Furthermore, there is also a vertical wind speed component along the vehicle vertical direction, but it is ignored since the relative wind speed in this direction is small resulting in a negligible force as compared to the suspension forces.

Given the suspension height, a nonlinear function is used to model the response of the suspension spring which is governed by the following equation,

$$F_{si} = -K_{si}(h_i - h_{0i}) - \tilde{K}_{si}(h_i - h_{0i})^5, \qquad i = \{1, 2\} \tag{8.25}$$

where h_{0i} is the uncompressed suspension height, which can be found once the vehicle height at equilibrium is known.

The tire elastic characteristic is modeled as a linear spring having a stiffness constant K_w,

$$F_{w_i} = -K_w(r_{w_i} - r_{w0}), \qquad i = \{1, 2\} \tag{8.26}$$

where r_{w_0} is the uncompressed tire radius, assuming each tire has the same properties.

The suspension damper is modeled as piecewise linear damper having discontinuous slope at $\pm\dot{w}$ as seen in Figure 8.4,

$$F_{di} = \begin{cases} C_{di}\dot{h}_i & |\dot{h}_i| < \dot{w} \\ C_{di}\dot{w} + \tilde{C}_{di}(\dot{h}_i - \dot{w}) & \dot{h}_i \geq \dot{w} \\ -C_{di}\dot{w} + \tilde{C}_{di}(\dot{h}_i + \dot{w}) & \dot{h}_i \leq -\dot{w} \end{cases} \qquad i = \{1, 2\} \tag{8.27}$$

where C_{di} and $\tilde{C}_{di}$ specify the slope in the first and second regions, respectively.

Let the force applied at the ground by the tire at the contact point between the road surface and the tire be

$$\underline{F}_{w_i} = F_{wf_i}\underline{r}_x + N_i\underline{r}_z, \qquad i = \{1, 2\} \tag{8.28}$$

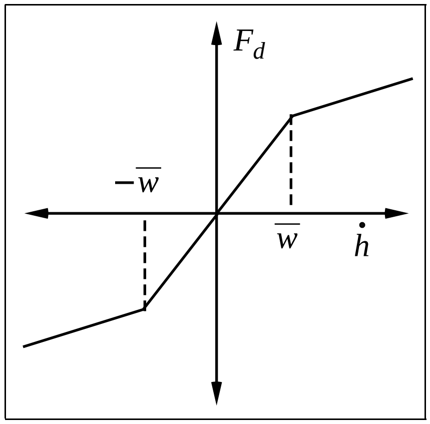

Figure 8.4: Damper characteristic.

then the road normal force, N_i, is simply the force exerted on the road by the tire.

$$N_i = -K_w(r_{w_i} - r_{w_0}), \qquad i = \{1, 2\} \tag{8.29}$$

Furthermore the tire tractive force, F_{wf_i}, is a function of the normal force and the tire slip ratio. Various tire models have been formulated. The longitudinal tire model by Bakker *et al.* (Bakker et al. 1987, Bakker and Pacejka 1989) is used in the simulation discussed in Section 8.3. The tire model is described in detail in Section 8.1.8.

With the external force known, the total force acting on the vehicle is obtained by combining equations (8.22), (8.23), (8.24), (8.28) and (8.29). This leads to:

$$F_x = \sum_{i=1}^{2} [F_{wf_i} \cos(\theta - \text{¢}_i) + K_w(r_{w_i} - r_{w_0}) \sin(\theta - \text{¢}_i)] + F_g \sin\theta + D \tag{8.30a}$$

$$F_z = \sum_{i=1}^{2} [F_{wf_i} \sin(\theta - \text{¢}_i) - K_w(r_{w_i} - r_{w_0}) \cos(\theta - \text{¢}_i)] - F_g \cos\theta + L \tag{8.30b}$$

8.1.5 Moments About the Vehicle Center of Mass

The moment about the car center of mass is generated from two sources. The ¯rst source is from the suspension force and the second is from the aerodynamic e®ect due to the asymmetric shape of the vehicle. Since this section concerns pitch rotation only, only the moment about the y-axis is needed. Knowing the forces at the suspension supports and the corresponding moment arms, the moment term generated by the suspension forces is given as

$$
\begin{aligned}
\underline{M}_{\text{sus}} &= (l_i \underline{c}_x - h_i \underline{c}_z) \times (F_{wf_i} \underline{r}_x + N_i \underline{r}_z) \\
&= \sum_{i=1}^{2} M_{\text{sus}_i} \underline{c}_y
\end{aligned}
\tag{8.31}
$$

where

$$
M_{\text{sus}_i} = -h_i \left[F_{wf_i} \cos(\theta - \phi_i) - N_i \sin(\theta - \phi_i) \right] + l_i \left[F_{wf_i} \sin(\theta - \phi_i) + N_i \cos(\theta - \phi_i) \right]
$$

The aerodynamic contribution to the moment about the car center of mass has been investigated by Yip *et al.* (Yip et al. 1992) and is given below

$$
\begin{aligned}
\underline{M}_w &= \frac{1}{2} C_{wy} A_f \rho_a L v_{wr}^2 \underline{c}_y \\
&= M_{wy} \underline{c}_y
\end{aligned}
\tag{8.32}
$$

where L is the wheelbase length and the y-axis moment coefficient, C_{wy} is determined experimentally for each vehicle.

Hence the total moment applied about the car center of mass is the sum of the two moment components given above in (8.31) and (8.32).

$$
\underline{M}_y = (M_{\text{sus}_1} + M_{\text{sus}_2} + M_{wy}) \, \underline{c}_y \tag{8.33}
$$

8.1.6 Brake Dynamics

The total brake torque, T_{ba}, applied to the wheels and the commanded brake torque, T_{bc}, are presumed to be related by the following first order lag equation,

$$
\dot{T}_{ba} = \frac{T_{bc} - T_{ba}}{\tau_b} \tag{8.34}
$$

where τ_b is the time delay constant which models, to the first order, the dynamics of the brake actuators and hydraulics. The total brake torque is then distributed between the front and the rear tire according to a brake biasing constant, k_b.

$$T_{b1} = k_b T_{ba} \quad (8.35a)$$

$$T_{b2} = (1 - k_b T_{ba} \quad (8.35b)$$

Each torque T_{bi} is positive and is limited to a maximum value where wheel lockup occurs. When the wheel angular velocity reaches zero, the brake torque is changed appropriately to prevent the wheel from rotating backwards.

8.1.7 Wheel Dynamics

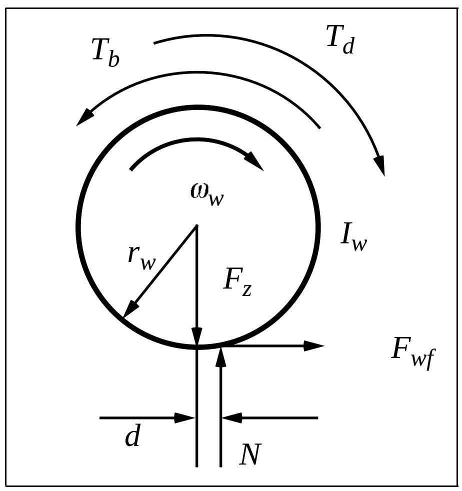

Figure 8.5: Wheel rotation.

In this model the wheels are assumed to be massless, but they are allowed to have nonzero moment of inertia I_w. Figure 8.5 shows the details of the wheel model which are used to obtain the front and rear wheel rotational dynamic equations.

$$\underset{\sim}{\omega}_{w_i} = \frac{(T_{di} - r_{w_i} F_{wf_i} - d_i N_i - T_{bi})}{I_w} \qquad i = \{1, 2\} \quad (8.36)$$

The applied torques are the engine torque, T_d, and the brake torque, T_b. The road normal force, N_i, is o®set to the front of the wheel by a distance d. Furthermore, the engine torque applied to each wheel is a function of the total engine output torque, T_e, which will be described in Section 8.1.9, and is distributed between the front and the rear wheels according to a drive biasing constant, k_d.

$$T_{d1} \quad = \quad k_d T_e \tag{8.37a}$$

$$T_{d2} \quad = \quad (1 - k_d T_e \tag{8.37b}$$

For example, set $k_d = 1$ for front-wheel drive vehicles.

8.1.8 Tire Traction Model

The longitudinal tire tractive force, F_{wf_i}, is correlated with the tire normal force, $N_i = -K_w(r_{w_i} - r_{w0})$, and its slip ratio, λ_i, through the *Magic Formula* which was developed by Bakker and Pacejka (Bakker et al. 1987, Bakker and Pacejka 1989). This model can accurately ¯t experimental tire data through the use of twelve coe±cients and will be described shortly.

Finding the tire slip ratio requires knowing the wheel forward velocity parallel the road surface. Let $\underline{P}_{w_i}$ be the position vector locating the wheel center,

$$\underline{P}_{w_i} = \underline{P}_{\mathrm{CM}} + l_i \underline{c}_x - h_i \underline{c}_z, \qquad i = \{1, 2\} \tag{8.38}$$

hence the wheel velocity follows by taking the inertial time derivative of the position vector $\underline{P}_{w_i}$.

$$\dot{\underline{P}}_{w_i} = (v_x - h_i \omega_y)\underline{c}_x + (v_z - l_i \omega_y - \dot{h}_i)\underline{c}_z, \qquad i = \{1, 2\} \tag{8.39}$$

The wheel forward velocity can now be found by taking the dot product with the road unit vector $\underline{r}_{xi}$.

$$\begin{aligned} v_{wf_i} &= \dot{\underline{P}}_{w_i} \cdot \underline{r}_{xi} \\ &= (v_x - h_i \omega_y) \cos(\theta - ¢_i) + (v_z - l_i \omega_y - \dot{h}_i) \sin(\theta - ¢_i), \qquad i = \{1, 2\} \end{aligned} \tag{8.40}$$

The slip ratio is de¯ned as

$$\lambda_i = 1 - \frac{v_{wf_i}}{r_w \omega_{w_i}}, \qquad i = \{1, 2\} \tag{8.41}$$

Finally the tire tractive force can be expressed as a nonlinear function of the normal force and slip ratio.

$$F_{wf_i} = f(N_i, \lambda_i), \qquad i = \{1, 2\} \tag{8.42}$$

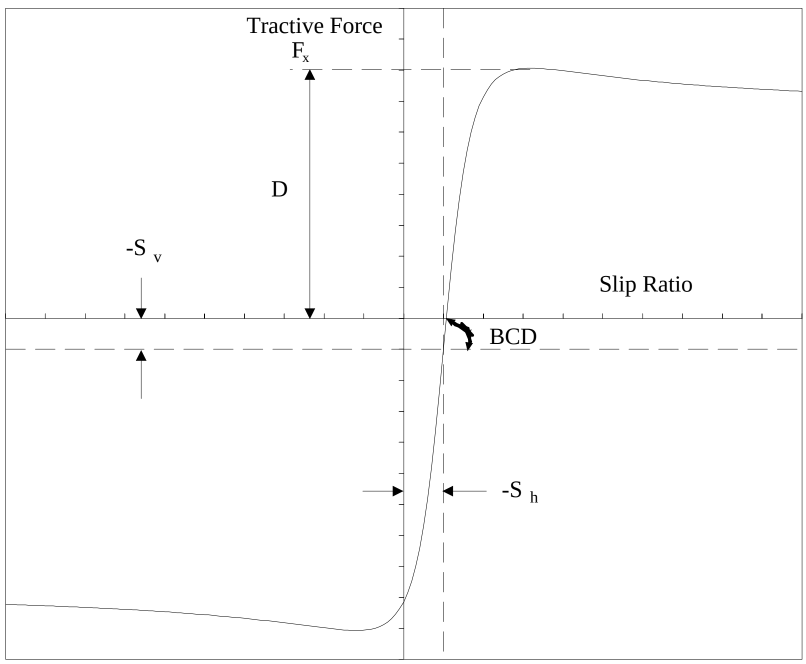

Figure 8.6: Exaggerated plot of the *Magic Formula*, showing the in°uence of the coe±cients.

As mentioned above, Bakker (Bakker et al. 1987, Bakker and Pacejka 1989) proposes the following *Magic Formula* to ¯t the tire tractive force numerically. This formula has been shown to accurately ¯t experimental tire data and has the form

$$y(x) = D \sin\left(C \tan^{-1}\left(Bx - E\left[Bx - \tan^{-1}(Bx)\right]\right)\right) \tag{8.43}$$

with

$$
\begin{aligned}
x &= \lambda + S_h && (8.44a)\\
f(N,\lambda) &= y(x) + S_v && (8.44b)
\end{aligned}
$$

Figure 8.6, a plot of the tractive force versus the slip ratio, shows the physical meaning of the coe±cients in Equations (8.43) and (8.44). Since the tractive force is also a function of the normal force, these coe±cients may be related to the normal force with following quantities.

$$
\begin{aligned}
D &= a_1 N^2 + a_2 N && (8.45a)\\
BCD &= (a_3 N^2 + a_4 N)\exp^{-a_5 N} && (8.45b)\\
C &= a_0 && (8.45c)\\
E &= a_6 N^2 + a_7 N + a_8 && (8.45d)\\
B &= BCD/CD && (8.45e)\\
S_h &= a_9 N + a_{10} && (8.45f)\\
S_v &= a_{11} && (8.45g)
\end{aligned}
$$

Once the experimental data for tire tractive force of a speci¯c tire is collected, the quantities a_0 to a_{11} can be obtained using various curve-¯tting techniques.

8.1.9 Engine Model

A simple engine model taken from Smith and Starkey (Smith and Starkey 1992) is used here. The output torque T_e, is a function of the engine speed ω_e, gear ratio ζ, drive train e±ciency η, and throttle position TP. Thus,

$$T_e = \mathrm{TP}\zeta\eta\left[c_1\left(\frac{\omega_e}{100}\right)^2 + c_2\left(\frac{\omega_e}{100}\right) + c_3\right] \tag{8.46}$$

By choosing the coe±cients c_1, c_2, and c_3, engine torque curves can be closely approximated. For a manual transmission, the engine speed is given by

$$
\begin{aligned}
\omega_e &= \zeta\omega_{w1} && \text{front-wheel drive} && (8.47)\\
\omega_e &= \zeta\omega_{w2} && \text{rear-wheel drive} && (8.48)
\end{aligned}
$$

The range of TP is between zero, for no output torque, and one, for maximum torque output at a certain engine speed. In addition, the actual throttle position response to the commanded throttle position is modeled as a ¯rst order lag,

$$\dot{\mathrm{TP}} = \frac{(\mathrm{TP}_c - \mathrm{TP})}{\tau_t} \tag{8.49}$$

where τ_t is the throttle delay time constant.

8.2 Nonlinear Lateral and Longitudinal Model

The full six degree of freedom model includes longitudinal, lateral and vertical translations and roll, pitch and yaw rotations. Including kinematic relations, the system of equations is 12^{th} order. Development of the six degree of freedom model closely follows the derivation where motion is restricted to the vertical plane. Subcomponents from the longitudinal model are generalized to the full nonlinear model and a steering system and road noise model are added.

8.2.1 Reference Frames

Using the longitudinal model as the stepping stone, we now can proceed to explore the complex behavior of the vehicle's lateral and longitudinal dynamics. As seen before, the ¯rst step is to de¯ne all the reference frames, which consist of the Earth-¯xed frame, the vehicle-¯xed frame, and the four road frames associated with the four tires.

First the Earth-¯xed reference frame E with origin $\mathcal{O}$ as seen in Figure 8.7 is de¯ned with unit vectors $(\underline{e}_x, \underline{e}_y, \underline{e}_z)$. A second frame C ¯xed in the vehicle with origin at the vehicle center of mass is de¯ned with unit vectors $(\underline{c}_x, \underline{c}_y, \underline{c}_z)$. As seen in Figure 8.8 this frame C may be described by three successive rotations from frame E. First rotate the Earth-¯xed frame about $\underline{e}_z$ axis by an amount ε, which is known as yaw angle. This leads to frame A with unit vectors $(\underline{a}_x, \underline{a}_y, \underline{a}_z)$. Next rotate frame A about $\underline{a}_x$ by an amount ϕ to obtain intermediate frame B with unit vectors $(\underline{b}_x, \underline{b}_y, \underline{b}_z)$. This angular rotation is called the roll angle. Finally rotate frame B about $\underline{b}_y$ by an angular displacement θ, which is the pitch

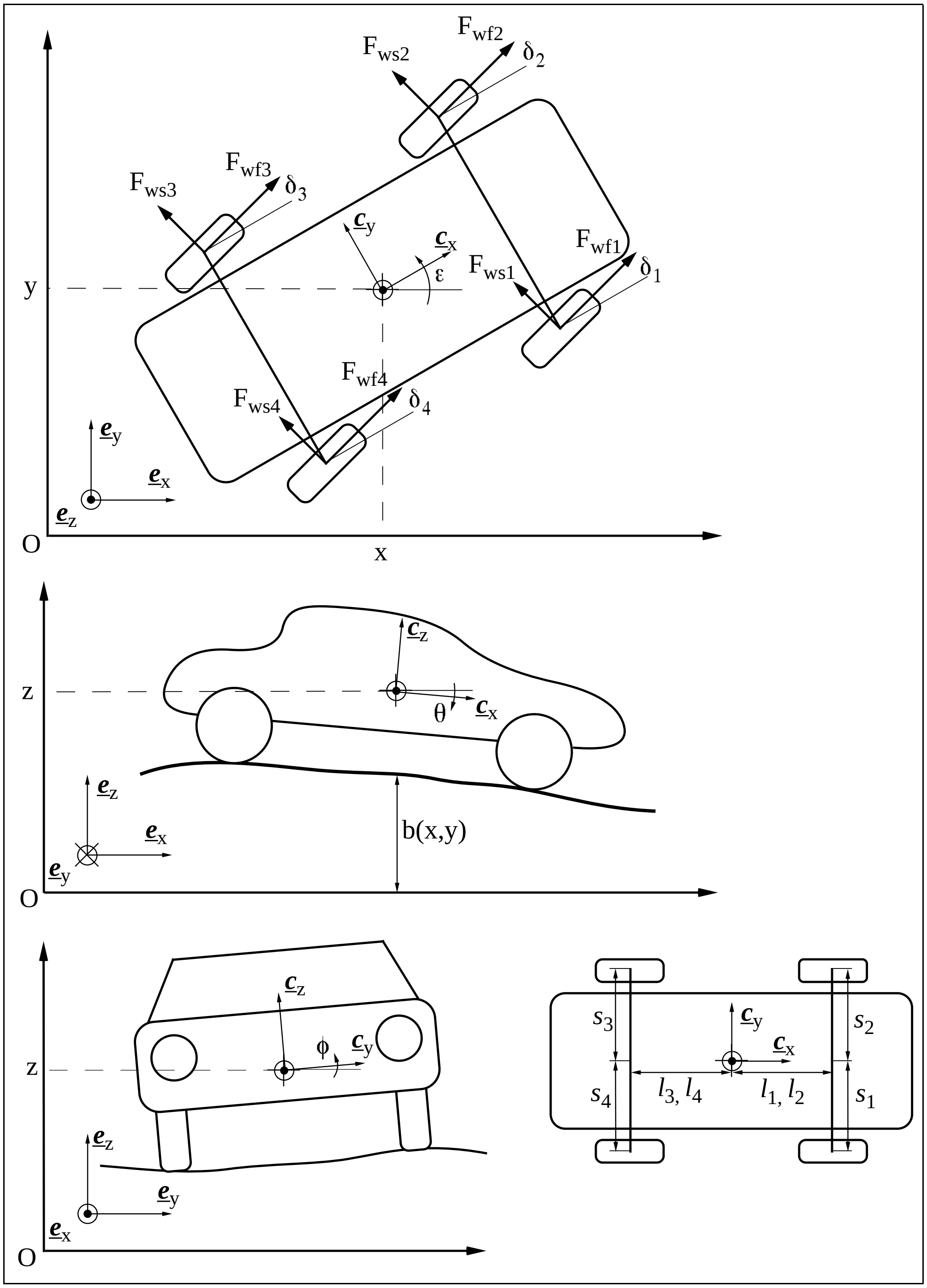

Figure 8.7: Representation of nonlinear vehicle model.

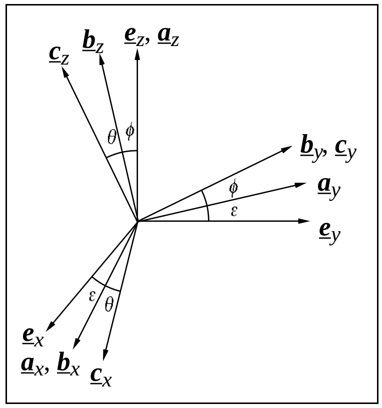

Figure 8.8: Relationship between reference frames.

angle, to obtain the vehicle-¯xed frame C. The corresponding coordinate transformation matrices are given below:

$$\begin{bmatrix} \underline{a}_x \\ \underline{a}_y \\ \underline{a}_z \end{bmatrix} = \begin{bmatrix} \cos\varepsilon & \sin\varepsilon & 0 \\ -\sin\varepsilon & \cos\varepsilon & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \underline{e}_x \\ \underline{e}_y \\ \underline{e}_z \end{bmatrix} \tag{8.50}$$

$$\begin{bmatrix} \underline{b}_x \\ \underline{b}_y \\ \underline{b}_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} \underline{a}_x \\ \underline{a}_y \\ \underline{a}_z \end{bmatrix} \tag{8.51}$$

$$\begin{bmatrix} \underline{c}_x \\ \underline{c}_y \\ \underline{c}_z \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} \underline{b}_x \\ \underline{b}_y \\ \underline{b}_z \end{bmatrix} \tag{8.52}$$

Now the transformation matrix from unit vectors in E to unit vectors in C reference frame can be readily determined as:

$$\begin{bmatrix} \underline{c}_x \\ \underline{c}_y \\ \underline{c}_z \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} \cos\varepsilon & \sin\varepsilon & 0 \\ -\sin\varepsilon & \cos\varepsilon & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \underline{e}_x \\ \underline{e}_y \\ \underline{e}_z \end{bmatrix} \tag{8.53}$$

In addition, the inverse of the above transformation matrix is its transpose.

The road reference frame R with unit vectors $(\underline{r}_x, \underline{r}_y, \underline{r}_z)$ for each tire is de¯ned with the origin located at the point of contact between the tire and the road surface. As shown in Figure 8.9, the orientation of this frame R is such that the $\underline{r}_z$ component coincides with the road normal vector, which is speci¯ed at each tire location (x, y) and is given as

$$\underline{n} = n_x \underline{e}_x + n_y \underline{e}_y + n_z \underline{e}_z$$

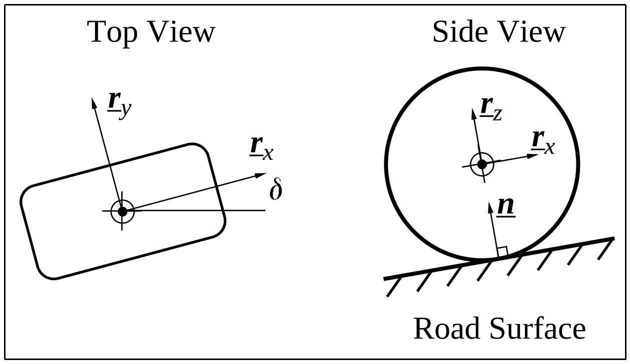

Figure 8.9: De¯nition of road frame.

Using the transpose of the transformation matrix of Equation (8.53), the $\underline{r}_z$ component can be expressed in the vehicle-¯xed reference frame as:

$$\underline{r}_z = r_{zx}\underline{c}_x + r_{zy}\underline{c}_y + r_{zz}\underline{c}_z \tag{8.54}$$

where

$$\begin{aligned} r_{xx} &= n_x(\cos\varepsilon\cos\theta - \sin\varepsilon\sin\phi\sin\theta) + n_y(\sin\varepsilon\cos\theta + \cos\varepsilon\sin\phi\sin\theta) - \\ &\quad n_z\cos\phi\sin\theta && (8.55) \\ r_{xy} &= -n_x\sin\varepsilon\cos\phi + n_y\cos\varepsilon\cos\phi + n_z\sin\phi && (8.56) \\ r_{xz} &= n_x(\cos\varepsilon\sin\theta + \sin\varepsilon\sin\phi\cos\theta) + n_y(\sin\varepsilon\sin\theta - \cos\varepsilon\sin\phi\cos\theta) + \\ &\quad n_z\cos\phi\cos\theta && (8.57) \end{aligned}$$

A second unit vector $\underline{r}_x$ of frame R is chosen such that it is normal to the tire axis of rotation and points in the direction of the tire heading.

Let $\underline{r}_x$ be expressed as

$$\underline{r}_x = r_{xx}\underline{c}_x + r_{xy}\underline{c}_y + r_{xz}\underline{c}_z$$

the components of $\underline{r}_x$ can be found by noting that

$$\begin{aligned} \underline{r}_x \cdot (-\sin\delta\underline{c}_x + \cos\delta\underline{c}_y) &= 0 && (8.58a) \\ \underline{r}_z \cdot \underline{r}_x &= 0 && (8.58b) \\ \|\underline{r}_x\| &= 1 && (8.58c) \end{aligned}$$

We can use the ¯rst property in Equation (8.58) to solve for r_{xy} in terms of r_{xx} and the tire steering angle.

$$r_{xy} = r_{xx} \tan\delta \tag{8.59}$$

Invoking the second property in Equation (8.58) and Equation (8.59) to solve for r_{xz} in terms of r_{xx} and the known components of $\underline{r}_z$, leads to the following equation:

$$r_{xz} = -\frac{r_{zx} + r_{zy}\tan\delta}{r_{zz}} r_{xx} \tag{8.60}$$

Note that r_{zz} can never be zero because it would mean that the road surface is vertical with respect to the vehicle body. Finally we can use the third property in Equation (8.58), that is, $r_{xx}^2 + r_{xy}^2 + r_{xz}^2 = 1$ and Equations (8.59) and (8.60) to solve for r_{xx} as:

$$r_{xx} = \frac{1}{\sqrt{1 + \tan^2\delta + \left(\frac{r_{zx}+r_{zy}\tan\delta}{r_{zz}}\right)^2}} \tag{8.61}$$

Hence the solutions for r_{xy} and r_{xz} follow directly from equations (8.59), (8.60) and (8.61).

$$r_{xy} = \frac{\tan\delta}{\sqrt{1 + \tan^2\delta + \left(\frac{r_{zx}+r_{zy}\tan\delta}{r_{zz}}\right)^2}} \tag{8.62}$$

$$r_{xz} = \frac{r_{zx} + r_{zy}\tan\delta}{r_{zz}\sqrt{1 + \tan^2\delta + \left(\frac{r_{zx}+r_{zy}\tan\delta}{r_{zz}}\right)^2}} \tag{8.63}$$

Then reference frame R is completely speci¯ed based on the right-handed orthogonal axis system and the third unit vector is given by $\underline{r}_y = \underline{r}_z \times \underline{r}_x$. Hence the unit vectors of the road frame can be expressed compactly in terms of the vehicle-¯xed unit vectors as:

$$\begin{bmatrix} \underline{r}_x \\ \underline{r}_y \\ \underline{r}_z \end{bmatrix} = \begin{bmatrix} r_{xx} & r_{xy} & r_{xz} \\ r_{yx} & r_{yy} & r_{yz} \\ r_{zx} & r_{zy} & r_{zz} \end{bmatrix} \begin{bmatrix} \underline{c}_x \\ \underline{c}_y \\ \underline{c}_z \end{bmatrix} \tag{8.64}$$

Furthermore if it may be assumed that each tire lies on an independent road surface, then a subscript i is added. Subscripts $i = \{1, 2, 3, 4\}$ refer to front right, front left, rear left, and rear right tires respectively.

8.2.2 Vehicle Dynamics

The dynamic equations of motion are derived from Newton's law applied in an inertial reference frame. The rotational dynamics are derived first. The translational dynamics follow.

Rotational Equations of Motion

With the angular rotations defined above, the angular velocity of the vehicle is given by:

$$\underline{\omega} = \dot{\varepsilon}\underline{e}_z + \dot{\phi}\underline{a}_x + \dot{\theta}\underline{b}_y \tag{8.65}$$

Use the coordinate transformation matrices in (8.50) through (8.52) to obtain the vehicle angular velocity in the vehicle-fixed coordinate frame as

$$\begin{aligned}\underline{\omega} &= (\dot{\phi}\cos\theta - \dot{\varepsilon}\cos\phi\sin\theta)\underline{c}_x + (\dot{\theta} + \dot{\varepsilon}\sin\phi)\underline{c}_y + (\dot{\phi}\sin\theta + \dot{\varepsilon}\cos\phi\cos\theta)\underline{c}_z \\ &= \omega_x\underline{c}_x + \omega_y\underline{c}_y + \omega_z\underline{c}_z \end{aligned} \tag{8.66}$$

Solving for $\dot{\varepsilon}$, $\dot{\phi}$ and $\dot{\theta}$, the rotational kinematic equations of motion are:

$$\dot{\varepsilon} = \frac{1}{\cos\phi}(-\sin\theta\omega_x + \cos\theta\omega_z) \tag{8.67a}$$

$$\dot{\phi} = \cos\theta\omega_x + \sin\theta\omega_z \tag{8.67b}$$

$$\dot{\theta} = \tan\phi(\sin\theta\omega_x - \cos\theta\omega_z) + \omega_y \tag{8.67c}$$

Furthermore the rotational dynamic equations are obtained from Euler's equations.

$$\dot{\omega}_x = \frac{M_x}{I_x} + \omega_y\omega_z\frac{I_y - I_z}{I_x} \tag{8.68a}$$

$$\dot{\omega}_y = \frac{M_y}{I_y} + \omega_z\omega_x\frac{I_z - I_x}{I_y} \tag{8.68b}$$

$$\dot{\omega}_z = \frac{M_z}{I_z} + \omega_x\omega_y\frac{I_x - I_y}{I_z} \tag{8.68c}$$

where M_x, M_y and M_z, which will be derived later in Section 8.2.5, are the total moment applied about the ($\underline{c}_x$, $\underline{c}_y$ and $\underline{c}_z$) axes resulting from the suspension and aerodynamic interactions, and I_x, I_y and I_z are the moments of inertia of the sprung mass about the $(\underline{c}_x, \underline{c}_y, \underline{c}_z)$ axes, respectively. The unsprung mass is neglected in this work.

Translational Equations of Motion

Let $\underline{P}_{\mathrm{CM}} = x\underline{e}_x + y\underline{e}_y + z\underline{e}_z$ be the position vector from the Earth-¯xed origin $\mathcal{O}$ to the vehicle center of mass as seen in Figure 8.7. Then the velocity of the mass center can be expressed either in Earth-¯xed or vehicle-¯xed coordinates as

$$\begin{aligned} \underline{v}_{\mathrm{CM}} &= \dot{x}\underline{e}_x + \dot{y}\underline{e}_y + \dot{z}\underline{e}_z && (8.69) \\ &= v_x\underline{c}_x + v_y\underline{c}_y + v_z\underline{c}_z && (8.70) \end{aligned}$$

Applying Equation (8.53) to transform Equation (8.70) into an Earth-¯xed frame leads to

$$\begin{aligned} &\underline{v}_{\mathrm{CM}} = \\ &\quad [v_x(\cos\varepsilon\cos\theta - \sin\varepsilon\sin\phi\sin\theta) - v_y\sin\varepsilon\cos\phi + v_z(\cos\varepsilon\sin\theta + \sin\varepsilon\sin\phi\cos\theta)]\,\underline{e}_x + \\ &\quad [v_x(\sin\varepsilon\cos\theta + \cos\varepsilon\sin\phi\sin\theta) + v_y\cos\varepsilon\cos\phi + v_z(\sin\varepsilon\sin\theta - \cos\varepsilon\sin\phi\cos\theta)]\,\underline{e}_y + \\ &\quad [-v_x\cos\phi\sin\theta + v_y\sin\phi + v_z\cos\phi\cos\theta]\,\underline{e}_z && (8.71) \end{aligned}$$

Hence the translational kinematic equations follow immediately from (8.69) and (8.71).

$$\begin{aligned} \dot{x} &= v_x(\cos\varepsilon\cos\theta - \sin\varepsilon\sin\phi\sin\theta) - v_y\sin\varepsilon\cos\phi + \\ &\quad v_z(\cos\varepsilon\sin\theta + \sin\varepsilon\sin\phi\cos\theta) && (8.72a) \\ \dot{y} &= v_x(\sin\varepsilon\cos\theta + \cos\varepsilon\sin\phi\sin\theta) + v_y\cos\varepsilon\cos\phi + \\ &\quad v_z(\sin\varepsilon\sin\theta - \cos\varepsilon\sin\phi\cos\theta) && (8.72b) \\ \dot{z} &= -v_x\cos\phi\sin\theta + v_y\sin\phi + v_z\cos\phi\cos\theta && (8.72c) \end{aligned}$$

The acceleration of the vehicle center of mass can be found by di®erentiating (8.70).

$$\begin{aligned} \underline{a} &= \dot{v}_x\underline{c}_x + \dot{v}_y\underline{c}_y + \dot{v}_z\underline{c}_z + (\omega_x\underline{c}_x + \omega_x\underline{c}_x + \omega_x\underline{c}_x) \times (\omega_x\underline{c}_x + \omega_x\underline{c}_x + \omega_x\underline{c}_x) \\ &= (\dot{v}_x + \omega_y v_z - \omega_z v_y)\underline{c}_x + (\dot{v}_y + \omega_z v_x - \omega_x v_z)\underline{c}_y + (\dot{v}_z + \omega_x v_y - \omega_y v_x)\underline{c}_z && (8.73) \end{aligned}$$

If the total external force applied to the vehicle is known,

$$\underline{F} = F_x\underline{c}_x + F_y\underline{c}_y + F_z\underline{c}_z$$

the translational dynamic equations are obtained from Newton's second law,

$$\dot{v}_x = \omega_z v_y - \omega_y v_z + \frac{F_x}{m} \tag{8.74a}$$

$$\dot{v}_y = \omega_x v_z - \omega_z v_x + \frac{F_y}{m} \tag{8.74b}$$

$$\dot{v}_z = \omega_y v_x - \omega_x v_y + \frac{F_z}{m} \tag{8.74c}$$

where m is the sprung mass of the vehicle. The forces are derived in Section 8.2.4.

8.2.3 Suspension Model

The suspension model for lateral and longitudinal vehicle motion is similar in every aspect to the longitudinal model. The extension to the three dimensional model slightly changes the geometric constraint equation corresponding to Equation (8.18) and is given below for the most general case,

$$0 = l_i \underline{e}_x - s_i \underline{e}_y - h_i \underline{e}_z - b(x + \Delta x_i, y + \Delta y_i)\underline{e}_z - (r_{w_i} + \xi_i)\underline{r}_{z_i} - \Delta x_i \underline{e}_x - \Delta y_i \underline{e}_y + z\underline{e}_z, \quad i = \{1,2,3,4\} \tag{8.75}$$

where l_i is the half wheelbase from the vehicle center of mass to the i^{th} wheel, s_i is the half track width from the vehicle center of mass to the i^{th} wheel, Δx_i and Δy_i are the relative tire distances from the center of mass to the i^{th} wheel, and the function $b(x, y)$ describes the road surface at location (x, y).

Solving for the relationship between r_{w_i} and h_i requires solving a nonlinear equation. In the special case where the road surface is planar, it is possible to solve for the relationship between the tire radius and the suspension height analytically as in the longitudinal model.

$$r_{w_i} + h_i r_{zz} = [z - b(x, y)]\, n_z + l_i r_{zx} - s_i r_{zy} - \xi_i, \qquad i = \{1,2,3,4\} \tag{8.76}$$

In addition four state equations governing the suspension height at four wheels are needed:

$$-K_w(r_{w_i} - r_{w0})\cos(\theta - \Delta_i) = f_K(f_i + f_C(\dot{h}_i), \qquad i = \{1,2,3,4\} \tag{8.77}$$

As stated in Section 8.1.3, solving for $\dot{h}_i$ depends on the damping function $f_C(\cdot)$.

To solve for the wheel radius and the relative tire position for an arbitrary road surface, requires making some approximations. Using the same concept as in Section 8.1.3, ¯rst approximate the relative tire position which is denoted by $¢x_i$ and $¢y_i$. These two relative tire position locators can be found by taking the dot product of Equation (8.75) with unit vectors $\underline{e}_x$ and $\underline{e}_x$ respectively. This leads to:

$$\begin{aligned} ¢x_i &= l_i(\cos\varepsilon\cos\theta - \sin\varepsilon\sin\phi\sin\theta) + s_i\sin\varepsilon\cos\phi - \\ &\quad h_i(\cos\varepsilon\sin\theta + \sin\varepsilon\sin\phi\cos\theta) - (r_{w_i} + \xi_i)n_{x_i}, \qquad i = \{1,2,3,4\} \\ ¢y_i &= l_i(\sin\varepsilon\cos\theta + \cos\varepsilon\sin\phi\sin\theta) + s_i\cos\varepsilon\cos\phi - \\ &\quad h_i(\sin\varepsilon\sin\theta + \cos\varepsilon\sin\phi\cos\theta) - (r_{w_i} + \xi_i)n_{y_i}, \qquad i = \{1,2,3,4\} \end{aligned} \tag{8.78}$$

Following the same approach in Section 8.1.3, assume that the road variation is zero, the wheel radius is constant, and the road normal vector is evaluated at the point where wheel center projects down to the road surface. This leads to the following equations where the relative tire position locators can be approximated as:

$$\begin{aligned} ¢x_i &= l_i(\cos\varepsilon\cos\theta - \sin\varepsilon\sin\phi\sin\theta) + s_i\sin\varepsilon\cos\phi - \\ &\quad h_i(\cos\varepsilon\sin\theta + \sin\varepsilon\sin\phi\cos\theta) - r_{w_0}n_{x_i}, \qquad i = \{1,2,3,4\} \end{aligned} \tag{8.79}$$

$$\begin{aligned} ¢y_i &= l_i(\sin\varepsilon\cos\theta + \cos\varepsilon\sin\phi\sin\theta) + s_i\cos\varepsilon\cos\phi - \\ &\quad h_i(\sin\varepsilon\sin\theta + \cos\varepsilon\sin\phi\cos\theta) - r_{w_0}n_{y_i}, \qquad i = \{1,2,3,4\} \end{aligned} \tag{8.80}$$

Once the tire location is approximated, the wheel radius can be found by taking the dot product of Equation (8.75) with unit vector $\underline{r}_{z_i}$, leading to:

$$\begin{aligned} r_{w_i} &= l_i z_{zx_i} - s_i r_{zy_i} - h_i r_{zz_i} - \xi_i - ¢x_i n_{x_i} - ¢y_i n_{y_i} + [z - b(x + ¢x_i, y + ¢y_i)]\, n_{z_i}, \\ &\quad i = \{1,2,3,4\} \end{aligned} \tag{8.81}$$

where the quantities n_{x_i}, n_{y_i} and n_{z_i} are evaluated at the approximated tire location $(x + ¢x_i, y + ¢y_i)$.

8.2.4 Forces

The gravitational force on the vehicle is $\underline{F}_g = -mg\underline{e}_z$. In addition to longitudinal wind lift and drag forces,

$$\begin{aligned} \underline{L} &= \frac{1}{2} C_L A_f \rho_a v_{wr}^2 \underline{c}_z \\ \underline{D} &= \frac{1}{2} C_D A_f \rho_a v_{wr}^2 \underline{c}_x \end{aligned}$$

there is now a lateral wind component which comes from crosswinds, large passing vehicles or fast lateral maneuvers. Moreover, these wind forces may have a considerable e®ect on lateral vehicle dynamics. This side force is modeled here as:

$$\underline{F}_s = \frac{1}{2} C_S A_f \rho_a v_{wr}^2 \underline{c}_y \tag{8.82a}$$

$$= F_s \underline{c}_y \tag{8.82b}$$

Work by Yip *et al.* (Yip et al. 1992) has correlated the force coe±cients C_L, C_D and C_Y to the relative wind speed and its angle relative to the vehicle longitudinal axis. These two variables are shown in Figure 8.10, and the analytical expressions for β and v_{wr} are given as:

$$v_{wr} = \sqrt{(v_{wx} - v_x)^2 + (v_{wy} - v_y)^2} \tag{8.83}$$

$$\beta = \tan^{-1}\left(\frac{v_{wy} - v_y}{v_{wx} - v_x}\right) \tag{8.84}$$

Let the force applied to each tire by the road be expressed as

$$\underline{F}_{w_i} = F_{wf_i}\underline{r}_x + F_{ws_i}\underline{r}_y + N_i\underline{r}_z$$

where the tire tractive and side force are obtained from the tire model in Section 8.2.7 and the tire normal force is simply

$$N_i = -K_w(r_{w_i} - r_{w_0})$$

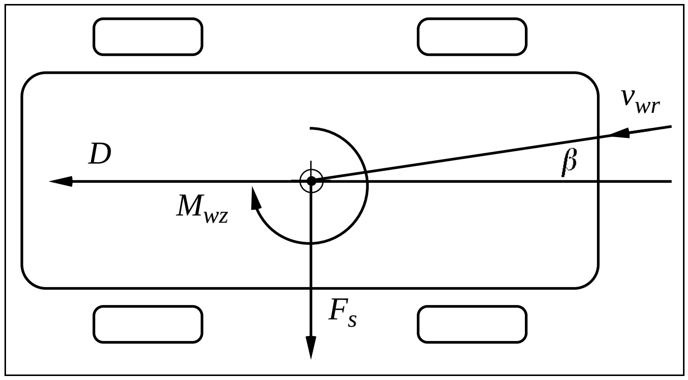

Figure 8.10: Aerodynamic forces acting on the vehicle have three components.

Then the force components applied to the vehicle along its three principal axes $(\underline{c}_x, \underline{c}_y, \underline{c}_z)$ can be expressed as:

$$F_x = \sum_{i=1}^{4} [F_{wf_i} r_{xx} + F_{ws_i} r_{yx} - K_w(r_{w_i} - r_{w0}) r_{zx}] + F_g \cos\phi \sin\theta + D \tag{8.85}$$

$$F_y = \sum_{i=1}^{4} [F_{wf_i} r_{xy} + F_{ws_i} r_{yy} - K_w(r_{w_i} - r_{w0}) r_{zy}] - F_g \sin\phi + F_S \tag{8.86}$$

$$F_z = \sum_{i=1}^{4} [F_{wf_i} r_{xz} + F_{ws_i} r_{yz} - K_w(r_{w_i} - r_{w0}) r_{zz}] - F_g \cos\phi \cos\theta + L \tag{8.87}$$

8.2.5 Moments About the Vehicle Center of Mass

Aerodynamics also contributes to the moment about the vehicle center of mass. Work by Yip *et al.* (Yip et al. 1992) has correlated the aerodynamic moment to the relative wind speed. The moment equation has a form similar to the aerodynamic force equation and is given below in vector form,

$$\underline{M}_w = \frac{1}{2}\rho_a v_{wr}^2 A_f L(C_{wx}\underline{c}_x + C_{wy}\underline{c}_y + C_{wz}\underline{c}_z) \tag{8.88a}$$

$$= M_{wx}\underline{c}_x + M_{wy}\underline{c}_y + M_{wz}\underline{c}_z \tag{8.88b}$$

where L is the wheel base length, and the moment coe±cients C_{wx}, C_{wy} and C_{wz} can be correlated to the relative wind speed and its angle in equations (8.83) and (8.84).

The total moment about the center of mass, which is contributed by the suspension forces and the aerodynamic forces, is obtained below:

$$\underline{M} = \sum_{i=1}^{4}(l_i\underline{c}_x - s_i\underline{c}_y - h_i\underline{c}_z) \times (F_{wf_i}\underline{r}_x + F_{ws_i}\underline{r}_y + N_i\underline{r}_z) + \underline{M}_w \tag{8.89}$$

Decomposing the moment equation into the three components about the vehicle principal axes using Equation (8.53) leads to the following moment equations.

$$M_x = \sum_{i=1}^{4} M_{x_i} + M_{wx} \tag{8.90a}$$

$$M_y = \sum_{i=1}^{4} M_{y_i} + M_{wy} \tag{8.90b}$$

$$M_z = \sum_{i=1}^{4} M_{z_i} + M_{wz} \tag{8.90c}$$

where

$$M_{x_i} = (F_{wf_i}(-s_i r_{xz_i} + h_i r_{xy_i}) + F_{ws_i}(-s_i r_{yz_i} + h_i r_{yy_i}) - K_w(r_{w_i} - r_{w0})(-s_i r_{zz_i} + h_i r_{zy_i}))$$

$$M_{y_i} = (F_{wf_i}(-l_i r_{xz_i} + h_i r_{xx_i}) + F_{ws_i}(l_i r_{yz_i} + h_i r_{yx_i}) - K_w(r_{w_i} - r_{w0})(l_i r_{zz_i} + h_i r_{zx_i}))$$

$$M_{z_i} = (F_{wf_i}(l_i r_{xy_i} + s_i r_{xx_i}) + F_{ws_i}(l_i r_{yy_i} + s_i r_{yx_i}) - K_w(r_{w_i} - r_{w0})(l_i r_{zy_i} + s_i r_{zx_i}))$$

8.2.6 Brake Dynamics

The brake dynamics are modeled as a ¯rst order lag similar to that used in the longitudinal model. The total brake torque T_{ba} is distributed between the front and the rear wheels according to a brake biasing constant k_b and is evenly divided between the left and the right wheels.

$$\begin{array}{lllll} T_{b1} & = & T_{b2} & = & \frac{k_b}{2}T_{ba} \quad \text{front wheels} \\ T_{b3} & = & T_{b4} & = & \frac{(1-k_b)}{2}T_{ba} \quad \text{rear wheels} \end{array} \tag{8.91}$$

Again T_{bi} is positive and is limited to a maximum value which is where wheel lockup occurs.

8.2.7 Wheel Dynamics and Tire Traction Model

The wheel dynamics are the same as that of the longitudinal model, however the tire traction model requires an additional variable since a lateral force and self-aligning moment are present. This additional variable is known as the lateral slip angle α and is de¯ned below. In Bakker's nonlinear tire model (Bakker et al. 1987, Bakker and Pacejka 1989, Pacejka and Bakker 1991), the tire tractive force, side force and self-aligning moment are functions of the normal force, the tire longitudinal slip ratio, and lateral slip angle. In order to ¯nd the tire tractive, side force and self-aligning moment, de¯ne the longitudinal slip and the slip angle. The longitudinal slip is de¯ned in the same way as in the longitudinal model, that is,

$$\lambda_i = 1 - \frac{v_{wf_i}}{r_{w_i}\omega_{w_i}}, \qquad i = \{1,2,3,4\} \tag{8.92}$$

The wheel forward velocity v_{wf_i} can be found by ¯rst ¯nding the velocity at the center of the tire.

$$\begin{aligned} \underline{P}_{w_i} &= (v_x - h_i\omega_y + s_i\omega_z)\underline{c}_x + (v_y + h_i\omega_x + l_i\omega_z)\underline{c}_y + (v_z - l_i\omega_y - s_i\omega_x - \dot{h}_i)\underline{c}_z, \\ & i = \{1,2,3,4\} \end{aligned} \tag{8.93}$$

Using Equation (8.64) we can transform Equation (8.93) to the road reference frame and the wheel forward velocity follows directly.

$$\begin{aligned} v_{w_i} &= (v_x - h_i\omega_y + s_i\omega_z)r_{xx_i} + (v_y + h_i\omega_x + l_i\omega_z)r_{xy_i} + (v_z - l_i\omega_y - s_i\omega_x - \dot{h}_i)r_{xz_i}, \\ & i = \{1,2,3,4\} \end{aligned} \tag{8.94}$$

The tire slip angle as seen in Figure 8.11 is de¯ned as the angle between the wheel velocity vector and the wheel heading vector. Thus,

$$\begin{aligned} \alpha_i &= \delta_i - \tan\left(\frac{v_{wy}}{v_{wx}}\right) \\ &= \delta_i - \tan\left(\frac{v_y + h_i\omega_x + l_i\omega_z}{v_x - h_i\omega_y + s_i\omega_z}\right), \qquad i = \{1,2,3,4\} \end{aligned} \tag{8.95}$$

where δ_i is the steering angle of each wheel.

The tractive force, side force and self-aligning moment can now be expressed as nonlinear functions of the tire normal force, slip ratio, slip angle, and other variables such as road surface conditions, and camber angle. The camber angle is de¯ned as the inclination of

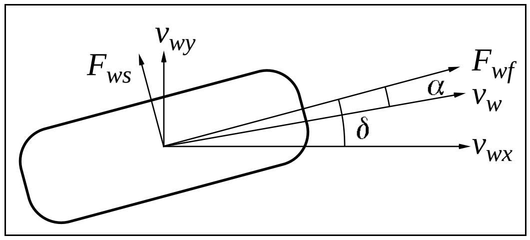

Figure 8.11: Top view of a tire under steering maneuver.

	Brake Force	Side Force	Self-aligning Moment
D	$a_1N^2 + a_2N$	$b_1N^2 + b_2$	$c_1N^2 + c_2N$
BCD	$(a_3N^2 + a_4N)\exp^{-a_5N}$	$\left[b_3\sin(b_4\tan^{-1}(b_5N))\right]\cdot$ $(1 - b_{12}\gamma)$	$(c_3N^2 + c_4N)\exp^{-c_5N}\cdot$ $(1 - c_{12}\lvert\gamma\rvert)$
C	a_0	b_0	a_0
E	$a_6N^2 + a_7N + a_8$	$b_6N^2 + b_7N + b_8$	$4\frac{a_6N^2+a_7N+a_8}{1-c_{13}\lvert\gamma\rvert}$
B	BCD/CD	BCD/CD	BCD/CD
S_h	$a_9N + a_{10}$	$b_9\gamma$	$a_9\gamma$
S_v	a_{11}	$(b_{10}N^2 + b_{11}N)\gamma$	$(a_{10}N^2 + a_{11}N)\gamma$

Table 8.1: Tire model coe±cients.

the wheel plane from a plane perpendicular to the road surface and parallel to the vehicle longitudinal axis.

The general formulation of the tire model developed by Bakker *et al.* has the form:

$$y(x) = D\sin\left(C\tan^{-1}\left(Bx - E\left[Bx - \tan^{-1}(Bx)\right]\right)\right) \tag{8.96}$$

with

$$x = X + S_h \tag{8.97a}$$

$$Y(X) = y(x) + S_v \tag{8.97b}$$

where the variable $Y(X)$ represents either the tire tractive force, side force or self-aligning moment, and the variable X represents the corresponding slip ratio or slip angle. The coe±cients above may be related to the tire normal force and camber angle γ as in Table 8.1. The above formulations are developed in cases of pure traction or pure cornering maneuvers. When the vehicle experiences a combination of cornering and braking, equations relating the tractive force, side force and self-aligning moment to the slip quantities require modi¯cation. Bakker (Bakker et al. 1987, Bakker and Pacejka 1989, Pacejka and Bakker 1991) provides the following method. First, de¯ne normalized slip quantities as follows:

$$\lambda^* = \frac{\lambda}{\lambda_{\max}} \tag{8.98}$$

$$\alpha^* = \frac{\alpha}{\alpha_{\max}} \tag{8.99}$$

where $\lambda_{\max}$ and $\alpha_{\max}$ are values where the tractive and side forces, respectively, reach a maximum. Next de¯ne the correction factor σ^* as:

$$\sigma^* = \sqrt{(\lambda^*)^2 + (\alpha^*)^2} \tag{8.100}$$

The modi¯ed equations for the tractive force, side force and self-aligning moment can be expressed as:

$$F_x = \frac{\lambda}{\sigma^*} F_{x_0}(\sigma^*, N) \tag{8.101}$$

$$F_y = \frac{\alpha^*}{\sigma^*} F_{y_0}(\sigma^*, N) \tag{8.102}$$

$$M_z = \frac{\alpha^*}{\sigma^*} M_{z_0}(\sigma^*, N) \tag{8.103}$$

where F_{x_0}, F_{y_0} and M_{z_0} are functions that provide the tractive force, side force and self-aligning moment as obtained from pure traction or pure cornering.

8.2.8 Engine Model

The same engine model described in Section 8.1.9 is used to develop the full six degree of freedom vehicle model. Since this model consists of four tires instead of two, the front and rear torque is divided evenly between the left and the right tires, resulting in the following equations:

$$\begin{array}{lllll} T_{d1} & = T_{d2} & = \frac{k_d}{2} T_e & & \text{front wheels} \\ T_{d3} & = T_{d4} & = \frac{(1-k_d)}{2} T_e & & \text{rear wheels} \end{array} \tag{8.104}$$

8.2.9 Steering Model

The type of steering model implemented in this work is a ¯xed-control steering model. With this model, the angular displacement of the steering wheel is speci¯ed. The other type of steering model is the free-control steering system in which the torque applied to the steering wheel is speci¯ed. This type of steering model is more complex since the steering angular displacement must be solved as a function of the resultant moments and the current angular displacement of the steering wheel. As shown in Figure 8.12, the steering system is modeled as a lumped mass system described in Lukowski *et al.* (Lukowski et al. 1990). The governing equation for the front-wheel steering system is given below,

$$\ddot{\delta} = -\frac{C_{ws}}{2I_{ws}}\dot{\delta} + \frac{K_{ws}}{2I_{ws}}(\delta_c - \delta) + \frac{K_{wp}(F_{wf1} - F_{wf2}) + M_{sa}}{2I_{ws}} \tag{8.105}$$

where δ_c is the commanded angular displacement of the steering wheel, I_{ws} is the moment of inertia of front wheels about their steering axis, K_{ws} and C_{ws} are the steering rotational sti®ness and damping constants, M_{sa} is the total self-aligning moment of the front wheels, and K_{wp} is the steering axis o®set.

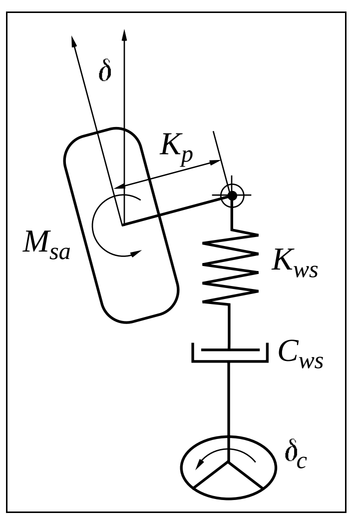

Figure 8.12: Lumped-mass representation of the steering system.

8.2.10 Random Road Excitation Model

One method of introducing random road excitation to the vehicle simulation is to generate a road noise profile at every point prior to the simulation. Such a method is developed by Cebon *et al.* (Cebon and Newland 1983) using Fourier transform methods to generate a two dimensional random road surface. However this approach is impractical, since it requires storage of enormous amounts of data. A more efficient and elegant method is to generate random road excitation on-line. With this scheme, the need to store all the road noise data is eliminated except for a small segment used to correlate the noise input between the front and the rear wheels. The method used here uses a first-order shaping filter approach and is developed by Gill (Gill 1983).

The idea behind this approach is to shape the spectral density of first order processes driven by stationary Gaussian white noise to closely approximate the measured road spectral density. Another important road characteristic besides the spectral density of the tracks, is the correlation between the left and right tracks. In order to achieve the above properties, the road noise at the left and the right wheels can be expressed as functions of two uncorrelated random processes ξ_M and θ_M.

$$\begin{bmatrix} \eta_1 \\ \eta_2 \end{bmatrix} = \begin{bmatrix} 1 & s_1 \\ 1 & s_2 \end{bmatrix} \begin{bmatrix} \xi_M \\ \theta_M \end{bmatrix} \tag{8.106}$$

The variable ξ_M describes the random road excitation at the point coinciding with the center of mass between the left and right tracks. The variable θ_M describes the noise difference between the left and the right tracks. The constants s_1 and s_2 are the half track widths from the car center to the left and right wheels respectively. Note that the constant s_1 is negative since it points in the negative $\underline{c}_y$ direction.

The random processes ξ_M and θ_M are first order processes driven by white noise.

$$\begin{bmatrix} \dot{\xi}_M \\ \dot{\theta}_M \end{bmatrix} = v_x \begin{bmatrix} \gamma_1 & 0 \\ 0 & \gamma_2 \end{bmatrix} \begin{bmatrix} \xi_M \\ \theta_M \end{bmatrix} + v_x \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \tag{8.107}$$

By specifying the constants γ_1, γ_2, σ_1 and σ_2, random road excitation may be generated with spectral density and correlation functions closely matching the experimentally measured data. Furthermore, the the constants σ_1 and σ_2 may be rede¯ned as functions of more physically meaningful constants, for example,

$$\sigma_1 = \sqrt{S_0 2\pi(1+\alpha)} \tag{8.108a}$$

$$\sigma_2 = \sigma_1 s_2 \sqrt{\alpha} \tag{8.108b}$$

where S_0 is the spectral intensity constant and α is the coherence constant. The values of the coherence constant range from zero to one, where a value of zero indicates that there is no correlation between the left and the right tracks and a value of one indicates that the two tracks are completely correlated. For vehicles traveling straight ahead at a constant speed v_x, the random road noise at the rear wheels is that of the front wheels delayed by a time interval $t_d = \frac{l}{v_x}$. The road noise at the rear wheels can be expressed as functions of the front wheels as follows:

$$\begin{bmatrix} \eta_3(t) \\ \eta_4(t) \end{bmatrix} = \begin{bmatrix} \eta_1(t-t_d) \\ \eta_2(t-t_d) \end{bmatrix} \tag{8.109}$$

8.3 Simulation Results

8.3.1 Longitudinal Model

Response of Vehicle to Various Inputs

In this section, the longitudinal model is subjected to various inputs and its responses are examined. Figure 8.13 shows the vehicle speed and pitch angle in response to a step throttle input when the vehicle is initially traveling at $10\frac{\mathrm{m}}{\mathrm{sec}}$. As expected, the vehicle should pitch upward, translating to a negative pitch angle in the simulation, when the vehicle is accelerating. As time passes, the vehicle pitches downward slowly as the vehicle acceleration decreases and speed increases. The reason for this behavior is that the moment caused by the wind about the y-axis dominates at high speed and low acceleration. This moment tends to pitch the car downward as a consequence of the asymmetric design of the vehicle top and bottom. The three jumps apparent in the plot of the pitch angle, occur when the lower gear switches to higher gear. This creates a discontinuity in engine output torque, which causes the vehicle to jerk.

After holding half-throttle for 60 seconds, the throttle is released and a step brake input is applied for the next 15 seconds. Figure 8.14 shows the plots of the vehicle speed and pitch angle as a total of 1000 N of brake force is applied to the wheels. The applied torque is about 10% of the maximum torque required to lock up the wheels, assuming a skidding coefficient of friction of 0.7. As expected, the vehicle pitches down as it decelerates, corresponding to a positive pitch angle. Again the small jumps in the pitch angle plot indicate the discontinuity of engine output torque due to the gear changes before the throttle position reaches zero.

The vehicle is then simulated while traveling on an inclined road surface. There is no throttle or brake input to the vehicle. Figure 8.15 shows the plots of the vehicle pitch angle and speed when coasting down a 5% grade road. The vehicle speeds up as a result of the gravitational force. The oscillations in the pitch angle plot reflect the fact that the vehicle is not initially at equilibrium. The pitch angle plotted is referenced to the Earth-fixed horizontal axis. The difference between the pitch angle and the angle of the road is known as the relative pitch angle, a measurement of the vehicle pitch relative to the road surface. As mentioned previously, this relative pitch angle does not vanish at steady state since there is a wind generated moment about the vehicle center of mass when the vehicle is traveling at high speed.

Next, a road disturbance is modeled. The vehicle is driven over a sharp sinusoidal bump 0.01 meters high and 0.3 meters wide while traveling at $27\frac{\mathrm{m}}{\mathrm{sec}}$. The responses of the vehicle height and pitch angle are plotted in Figure 8.16. The first sharp *corner* in the pitch angle plot indicates the point where the front wheel reaches the bump and the second sharp *corner* follows when the rear wheel passes over the bump. Looking at the vehicle height, one can conclude that this is a reasonable response of the vehicle since a well maintained vehicle with good shocks should not oscillate more than once or twice when it is disturbed from equilibrium.

Finally, random road excitation is added to the front and the rear wheels. Since the vehicle is traveling along a straight path, the road noise at the rear wheel is that of the front wheel delayed by the time interval required for the rear wheel to reach to the former location of the front wheel. If the vehicle is traveling at a constant speed v_x, the delay time can be expressed as $t_d = \frac{l}{v_x}$, where l is the distance between the front and the rear wheels. The vehicle height, pitch angle, and random road input at the front wheels are plotted in Figure 8.17 while the vehicle is traveling at $27\frac{\text{m}}{\text{sec}}$. As seen in the plot of the vehicle height and the noise amplitude, the suspension system ¯lters out the high frequency noise but passes through the low frequency components of the noise. From the plot of the pitch angle, one can also conclude that the pitch angle is more susceptible than the vehicle height to high frequency noise, even thought it also does some ¯ltering out of the high frequency components. In addition, the simulated spectral density of the random noise process obtained by averaging 100 realizations is plotted with the theoretical spectral density in Figure 8.18. This random road excitation is typical of rough highway roads.

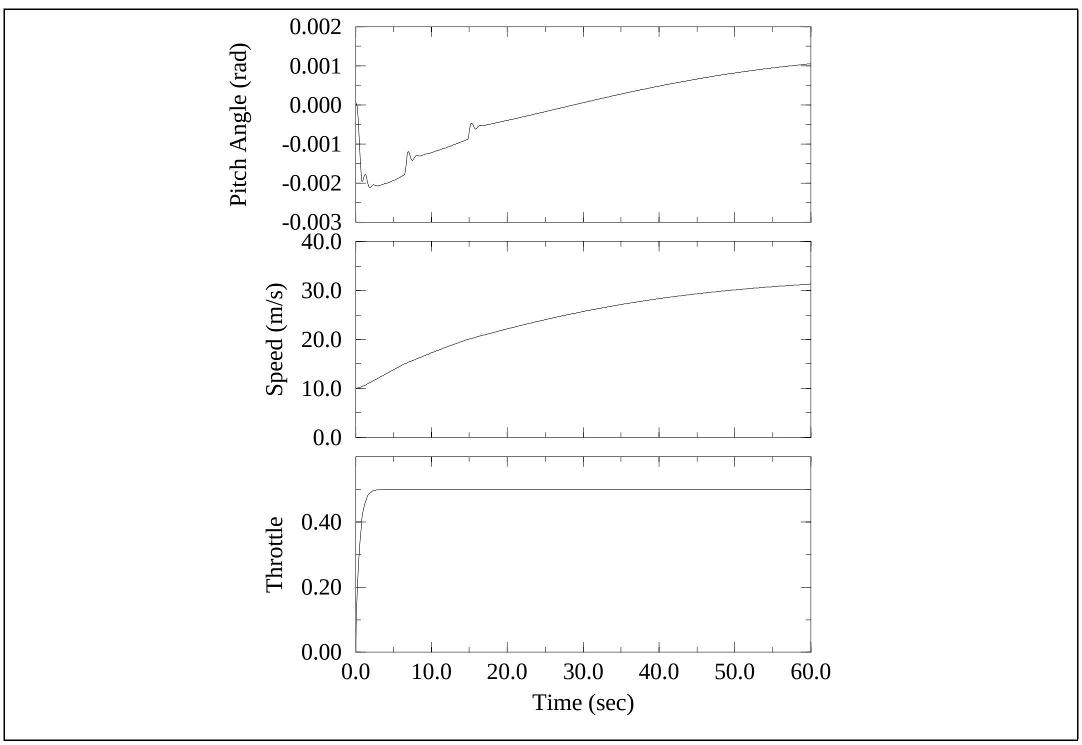

Figure 8.13: Vehicle response due to a step throttle input.

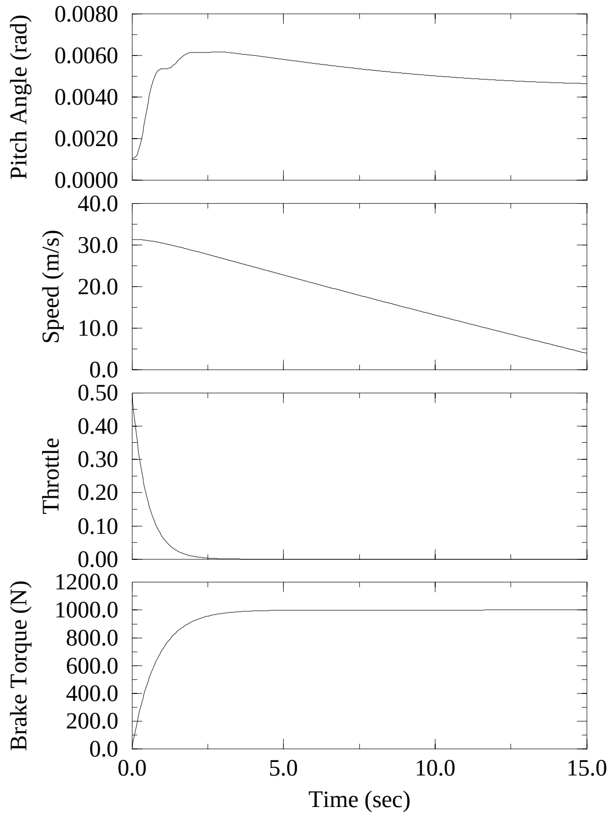

Figure 8.14: Vehicle response due to a step brake input subsequent to a step throttle input.

Small Angle Approximation

In steady-state, the magnitude of the pitch angle relative to the road surface, that is, $\theta -$ ¢, is at most on the order of 10^{-3} radian. The reason that the pitch angle does not vanish is because there is a moment about the vehicle center of mass caused by the wind at high speed. Furthermore, the maximum pitch angle relative to the road surface during a transient response of the vehicle is on the order of 10^{-2} radian. Since the relative pitch angle is small, we can make a ¯rst order approximation of the trigonometric functions without degrading the model accuracy. For any angle x, the small angle approximation of $\cos(x)$ is taken as one and that of $\sin(x)$ is taken as x. In the operating range of the pitch angle whose magnitude is less than 10^{-2} radian, the maximum error resulted from

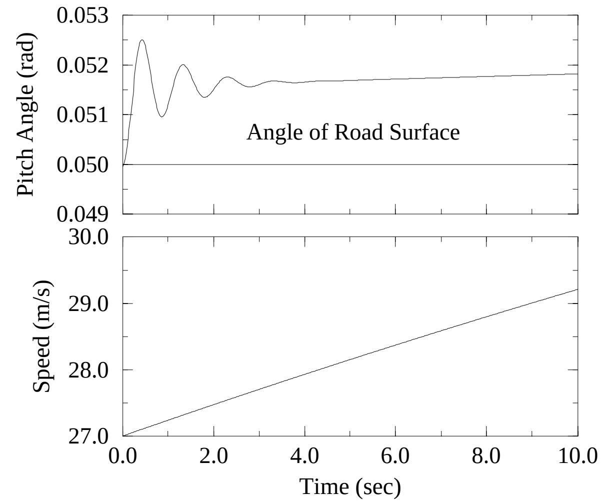

Figure 8.15: Vehicle response when decending down a 5% grade road.

making small angle approximations is less than 0.1 percent. This is too small an error to have any signi¯cant e®ect on the simulation accuracy. To verify this, the approximated and non-approximated systems are simulated by initially setting the relative pitch angle to a maximum value, which is taken to be 0.05 radian. The responses of the states of the approximated and non-approximated systems are compared for any signi¯cant deviations. As seen in Figure 8.19, there is no notable di®erence between the original model and the one using small angle approximations.

Knowing that making a small angle approximation on the relative pitch angle does not reduce the simulation accuracy, we would also like to investigate the consequences of making an approximation on the absolute pitch angle, which is referenced from the Earth-¯xed horizontal axis. This might reduce the simulation accuracy if the elevation of the road is large, since the absolute pitch angle is the sum of the road angle and the vehicle pitch angle relative to the road. According to transportation literature, a typical road grade limit for highways is around 10 to 15 percent. To take a worst case scenario, we will use a maximum road grade of 15% and a maximum relative pitch angle of 0.05 radian as used previously. This will constrain the maximum limit of the absolute pitch angle to about 0.2

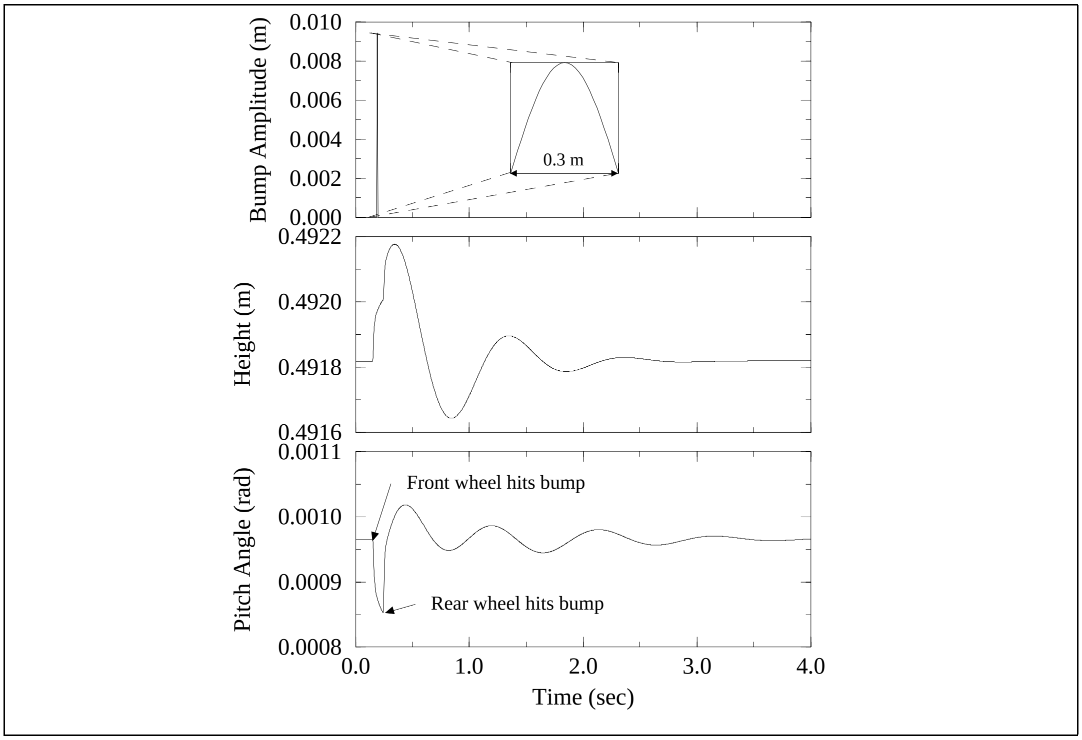

Figure 8.16: Vehicle response when passing over a sinusoidal bump.

radian. Setting the road elevation to the maximum allowable limit of 15% and the absolute pitch angle to 0.2 radians, the vehicle is simulated as it is initially traveling at $27\frac{\text{m}}{\text{sec}}$ with the nominal throttle position of 22.555% of the maximun throttle position. Comparing the response of the approximated system to the non-approximated system, we found that there are no signi¯cant deviations between the two models. The deviation in all states is below two orders of magnitude. Figure 8.20 shows the vehicle pitch angle and velocity as well as the longitudinal velocity. There are no visible di®erences between the approximated and non-approximated systems.

In conclusion, it is permissible to use a small angle approximation on the pitch angle. By making a small angle approximation, we can save about 5 percent in computational time. The reason that the computational gain is not signi¯cant is because we only save one multiplication operation for each cosine term. For each sine term, we still have to use one multiplication operation regardless of whether we make a small angle approximation or not.

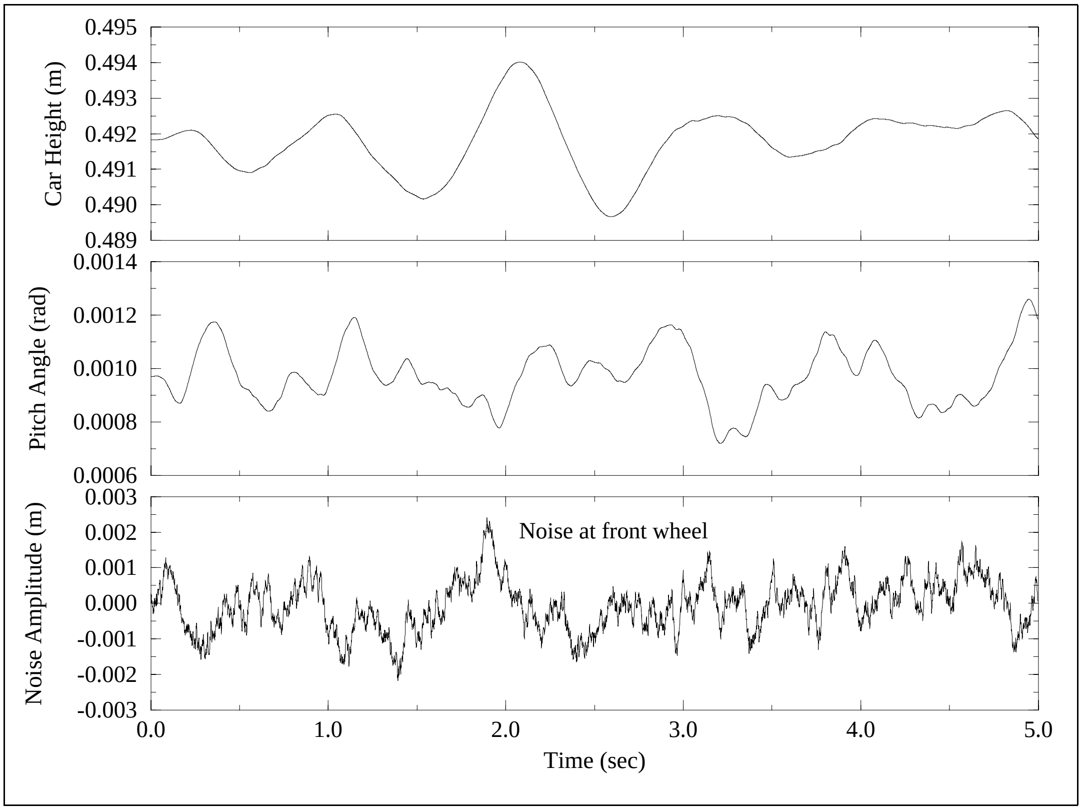

Figure 8.17: Vehicle response due to random road excitation.

Linearization at a Nominal Operating Point

A linear model of the vehicle operating at some nominal point (x_0, u_0), where $\underline{f}(x_0, u_0) = \underline{0}$, is needed to implement the fault detection and identi¯cation ¯lter. Due to the complexity of the nonlinear model, it is impractical to linearize the system analytically. Therefore the linearized system is obtained numerically. The process to linearize the system numerically is described below.

First, a nominal operating point needs to be speci¯ed where the linearized model is obtained. This nominal point can be found by specifying the inputs and simulating the system to reach steady state. It takes about 300 seconds for the system to reach steady state. After obtaining the nominal operating point, a numerical linearization process can be implemented to obtain the linearized model.

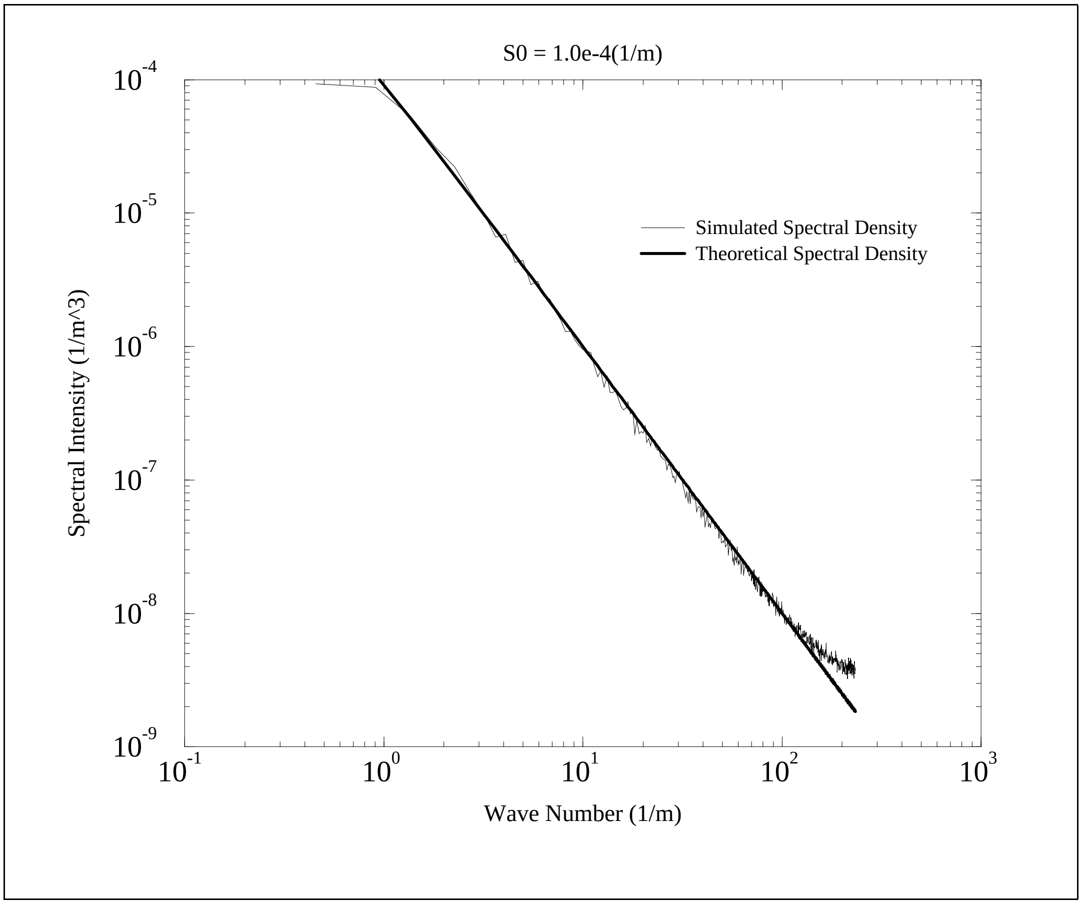

Figure 8.18: Power spectral densities of simulated and theoretical random noise processes.

Starting with the nonlinear system $\underline{x} = \underline{f}(\underline{x}, \underline{u})$, one would like to linearize this system at some nominal point (x_0, u_0). Using Taylor's expansion, one can expand the nonlinear system around with $\underline{x} = \underline{x}_0 + \underline{\mathfrak{x}}$ and $\underline{u} = \underline{u}_0 + \underline{\mathfrak{u}}$ as

$$\underline{\mathfrak{x}} = \underline{f}(\underline{x}_0, \underline{u}_0) + \nabla_{\underline{x}}\underline{f}(\underline{x}, \underline{u})\,\Big|_{\substack{\underline{x} = \underline{x}_0 \\ \underline{u} = \underline{u}_0}} \underline{\mathfrak{x}} + \nabla_{\underline{u}}\underline{f}(\underline{x}, \underline{u})\,\Big|_{\substack{\underline{x} = \underline{x}_0 \\ \underline{u} = \underline{u}_0}} \underline{\mathfrak{u}} + \text{higher order terms}$$

By neglecting the higher order terms and noting that $\underline{f}(x_0, u_0)$ vanishes, the linearized system becomes

$$\underline{\mathfrak{x}} \quad = \quad A\underline{\mathfrak{x}} + B\underline{\mathfrak{u}}$$

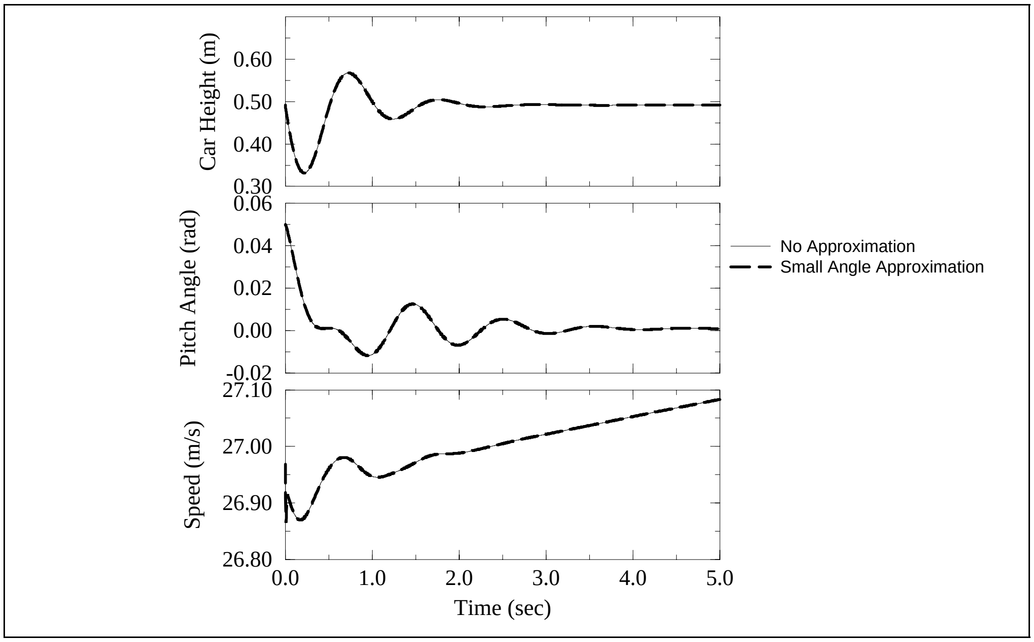

Figure 8.19: E®ect of making a small angle approximation of the relative pitch angle.

$$A = \nabla_{\underline{x}} f(\underline{x}, \underline{u}) \Big|_{\substack{\underline{x} = \underline{x}_0 \\ \underline{u} = \underline{u}_0}}$$

$$B = \nabla_{\underline{u}} f(\underline{x}, \underline{u}) \Big|_{\substack{\underline{x} = \underline{x}_0 \\ \underline{u} = \underline{u}_0}}$$

As mentioned previously, analytically calculating the gradient of the nonlinear system is impractical. Therefore an approximation scheme will be used.

Using the central di®erence method, the A and B matrix coe±cients are approximated as

$$a_{ij} = \frac{\partial f_i}{\partial x_j} \Big|_{\substack{\underline{x} = \underline{x}_0 \\ \underline{u} = \underline{u}_0}} \simeq \frac{f_i(\underline{x}_0 + [\delta x]_j, \underline{u}_0) - f_i(\underline{x}_0 - [\delta x]_j, \underline{u}_0)}{2\delta x}$$

$$b_{ij} = \frac{\partial f_i}{\partial u_j} \Big|_{\substack{\underline{x} = \underline{x}_0 \\ \underline{u} = \underline{u}_0}} \simeq \frac{f_i(\underline{x}_0, \underline{u}_0 + [\delta u]_j) - f_i(\underline{x}_0, \underline{u}_0 - [\delta u]_j)}{2\delta x}$$

where the notation $[\delta x]_j$ denotes a vector with zero elements everywhere except for the j^{th} element which has the value δx.

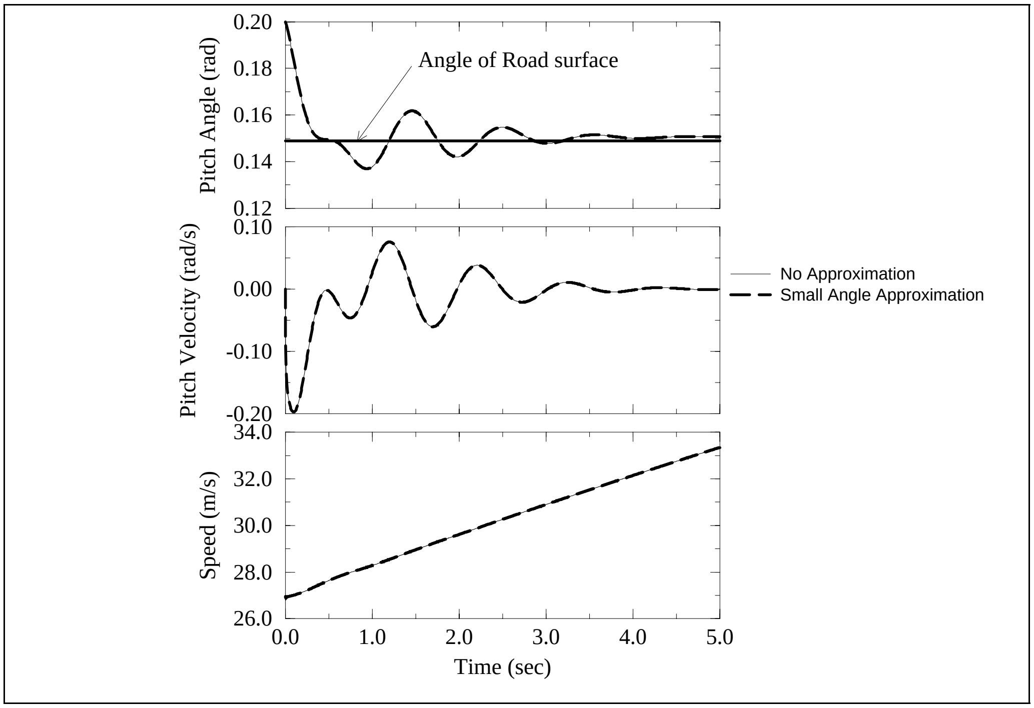

Figure 8.20: E®ect of making a small angle approximation of the absolute pitch angle.

Care must be taken in choosing the perturbation values δx and δu. Truncation errors due to ¯nite signi¯cant digits in digital computers will result if perturbation size is too small; whereas error produced by nonlinearities will result if the perturbation size is too large. Each coe±cient should be plotted versus the perturbation size and each coe±cient should be chosen individually within the region where the curve remains °at. Figure 8.21 shows a typical plot of one coe±cient versus perturbation size in which the curve can be characterized by three regions. In region I, errors are induced by ¯nite computer word length and indicate that the perturbation size is too small. In region III, errors are induced by model nonlinearities and indicate that the perturbation size is too large. The most accurate representation of each coe±cient lies in region II where the error curve is °at. In our experience, typical values for the normalized perturbation sizes of $\frac{\delta x}{x_0}$ and $\frac{\delta u}{u_0}$ range from 10^{-6} to 10^{-3} for the central di®erences method.

The system is linearized at a highway speed of $27\frac{\mathrm{m}}{\mathrm{sec}}$ or 65mph. To maintain at this

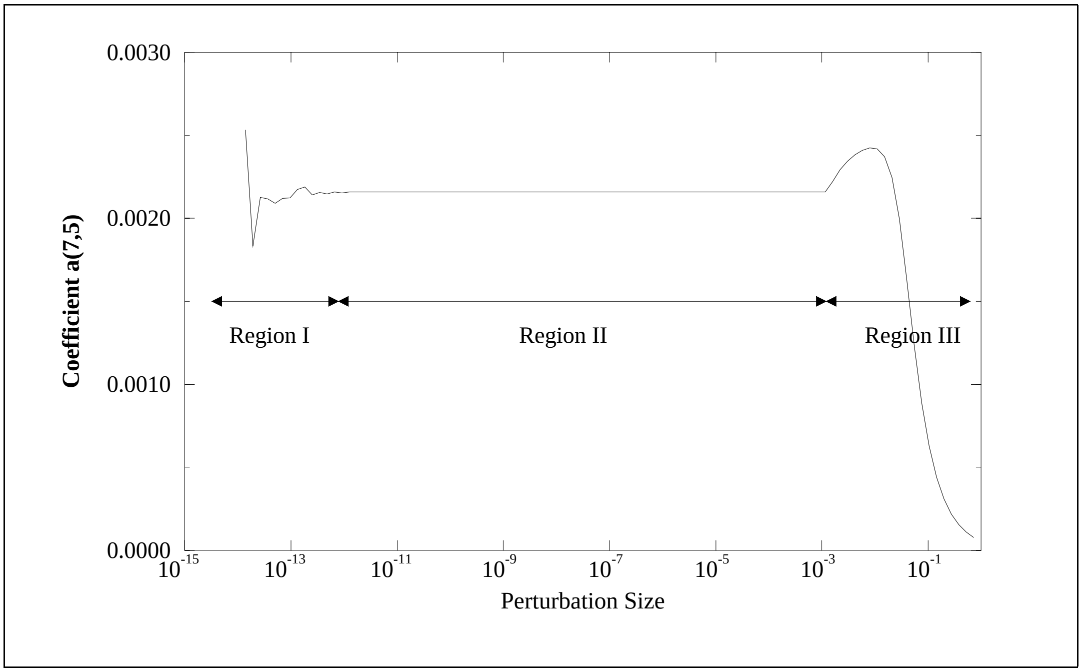

Figure 8.21: E®ect of perturbation size on numerical derivative computation.

speed, the throttle position is set at 22.555% of the maximum throttle position. Figure 8.22 shows the transient responses of the vehicle when the throttle input is perturbed upward by 15 percent. The responses of the linearized system match very well to those of the nonlinear system. In addition, the vehicle steady-state responses are plotted in Figure 8.23. However, the steady state responses of the linearized system deviate from the nonlinear model considerably for large perturbations. By comparing all of the states of the linearized and nonlinear systems, we found that deviation errors between the linearized and nonlinear systems at steady state are below 10 percent for a 15 percent increase or 15 percent decrease in throttle position input. This corresponds to a range of speed from $25.5\frac{\mathrm{m}}{\mathrm{sec}}$ to $28.5\frac{\mathrm{m}}{\mathrm{sec}}$. Furthermore the brake input is also perturbed to compare the accuracy of the linearized model to that of the nonlinear model. Figure 8.24 shows that the maximum perturbation size for the brake input is 34 N such that the deviation errors of the states between the two models are less than 10 percent. As evident in the plots, the responses of the system to a brake perturbation are much more linear than those due to a throttle perturbation.

This is not surprising since the brake torque is related to the brake input through a linear ¯rst order dynamics; whereas the engine torque is not only controlled by throttle position but is also a nonlinear function of the wheel speed. If we eliminate the engine model and specify the engine torque directly, the deviation errors between the two models are less than 3 percent for the same range of speed.

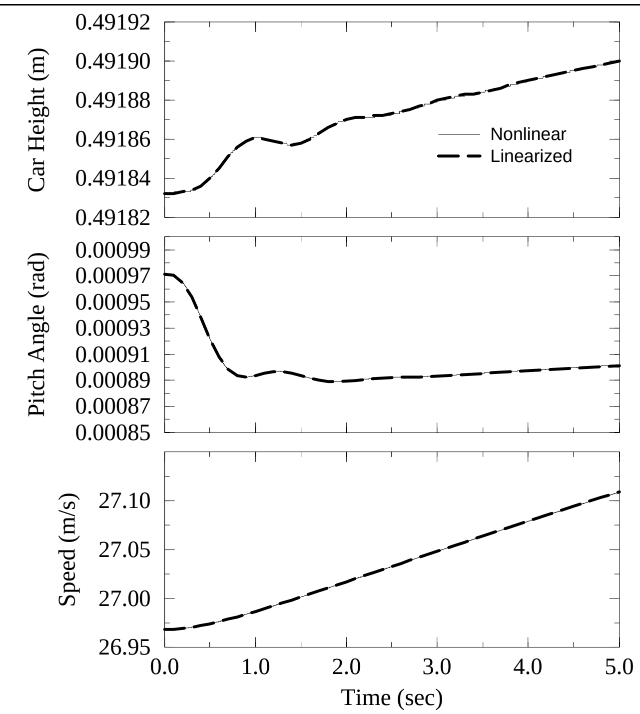

Figure 8.22: Transient response of the linearized and nonlinear systems with a perturbed throttle input (+15%).

8.3.2 Lateral and Longitudinal Model

Response of Vehicle to Various Inputs

The longitudinal response of the vehicle was analyzed in Section 8.3.1, therefore it is only necessary to investigate the vehicle lateral modes at this point. First the vehicle is stimulated with a step steering input of 0.01 radian while the vehicle is initially traveling at $27\frac{\mathrm{m}}{\mathrm{sec}}$ at its corresponding nominal throttle position of 22.555% of maximum throttle position.

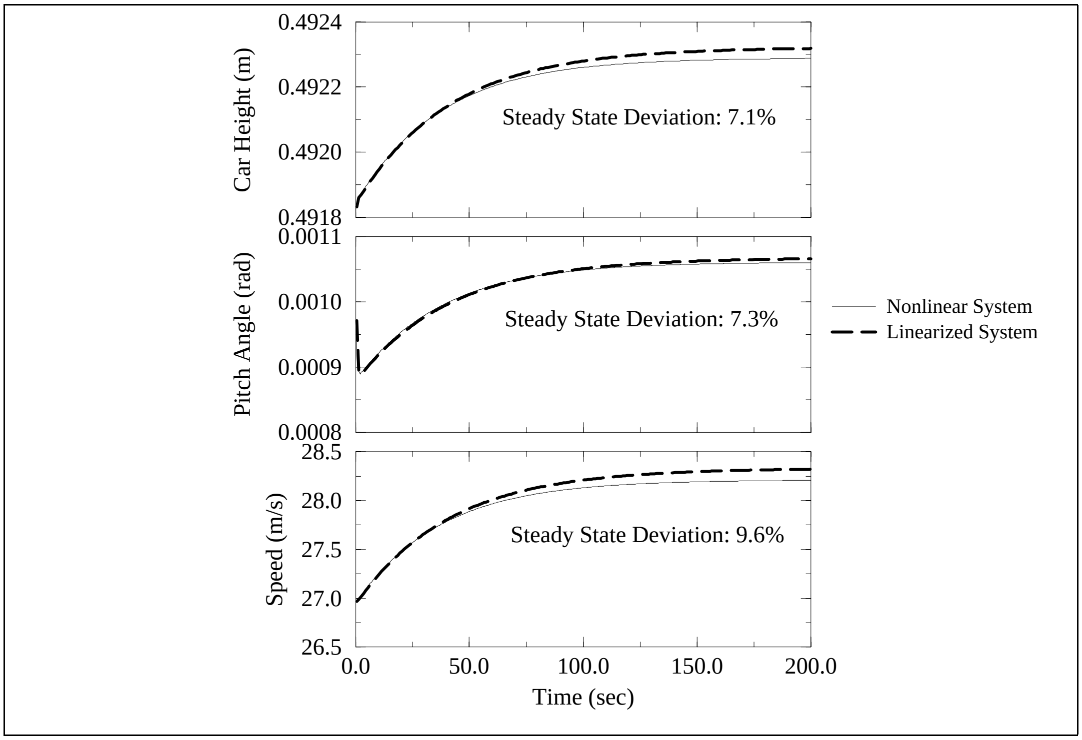

Figure 8.23: E®ect of perturbation size on numerical derivative computation.

Figure 8.25 shows the vehicle roll angle, yaw velocity and path. As the vehicle turns left, the vehicle should roll to the right, for a positive roll angle, and the yaw velocity should increase to reach a constant in steady state as seen in Figure 8.25. At this speed, a turn of 0.01 radian is considered to be a medium cornering maneuver which generates a lateral acceleration of about 0.2g. If the vehicle is allowed to reach steady state, a constant steering angle of 0.01 radian will steer the vehicle around a constant radius of 310 meters.

Next, lateral response is examined as a pulse of crosswind is applied to the vehicle while the vehicle is traveling straight ahead at $27\frac{\mathrm{m}}{\mathrm{sec}}$. The applied wind velocity is $15\frac{\mathrm{m}}{\mathrm{sec}}$ with 10 seconds duration. The lateral response of the vehicle is plotted in Figure 8.26, showing the vehicle path without any steering correction is made. Plots of the crosswind pro¯le and yaw velocity are also shown. The decrease in the yaw velocity re°ects that the magnitude of the side wind applied to the vehicle is decreasing since the vehicle is gradually turning away from the crosswind disturbance.

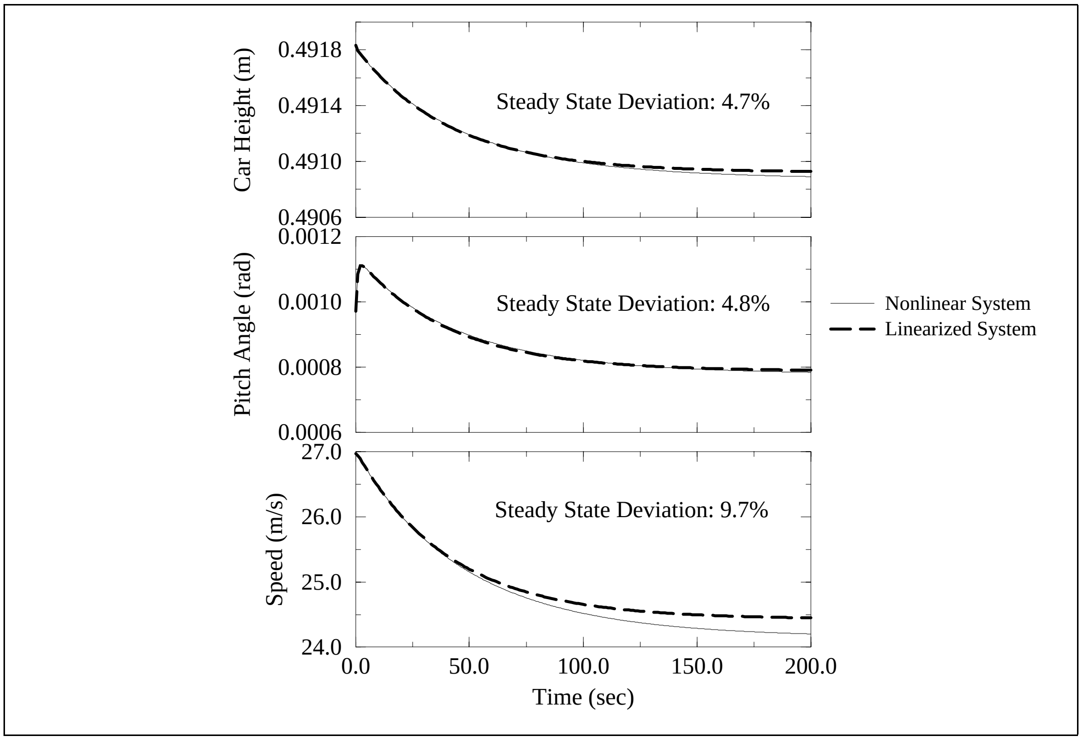

Figure 8.24: Steady-state response of the linearized and nonlinear systems with a perturbed brake input (+34 N).

Finally, random road excitation is introduced to the vehicle model, simulating the road condition of typical highways. The vehicle roll and pitch angle as well as its height are shown in Figure 8.27 together with the random road excitation of the right and left tracks. The left and right tracks are taken to have the same spectral density function and are also correlated, with the correlation coe±cient having a value of 0.75. Averaging from 100 realizations, the simulated spectral density of the random processes are plotted along with the theoretical density in Figure 8.28. Similarly, the coherency functions which characterize the dependency between the left and right tracks are also shown on lower half of Figure 8.28. Looking at the road noise of the left and right tracks, one can see that they are highly correlated at low frequencies. On the other hand, high frequency components of the noise do not seem to be correlated between left and right tracks. An alternative way to look at this is by the means of the coherency function as seen in Figure 8.28. At low wave number or spatial frequency, the left and right tracks are strongly correlated and the coherency function rapidly decreases as the wave number increases.

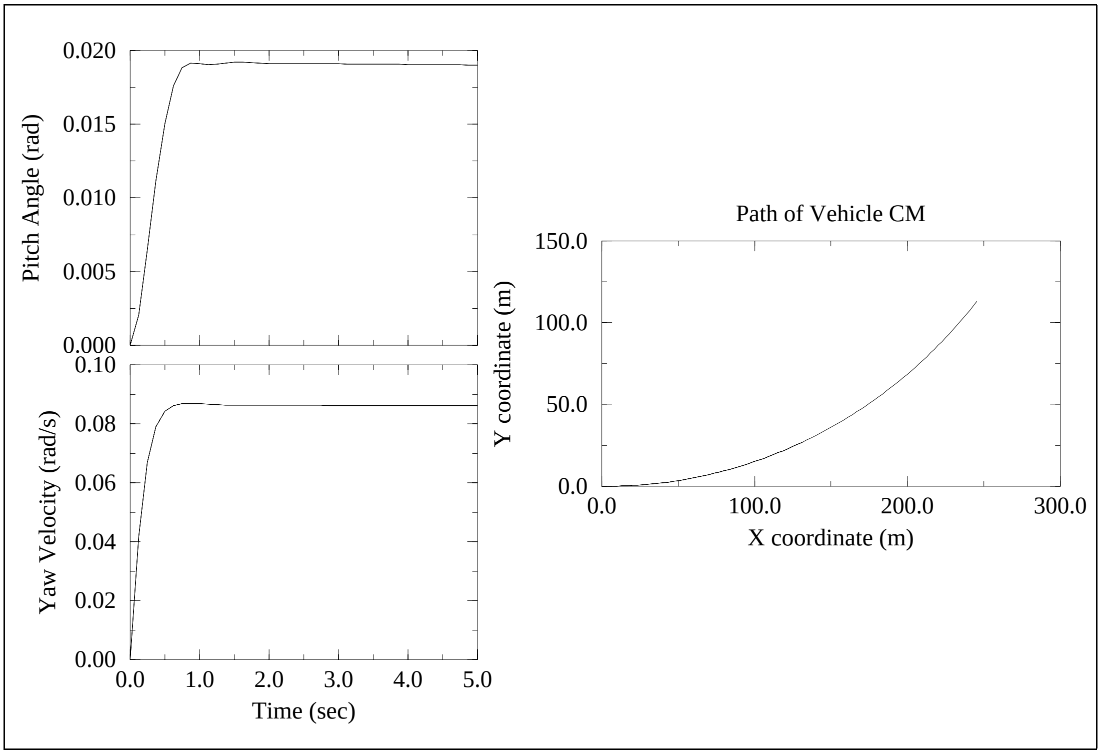

Figure 8.25: Vehicle response due to step steering input of 0.01 radian.

Small Angle Approximation

We would like to investigate the e®ects, if any, of a small angle approximation of the pitch and roll angles, on the accuracy of the full model simulation. We have already established that the operating range of the pitch angle is small enough that a small pitch angle approximation does not have a signi¯cant e®ect on the simulation accuracy of the vehicle model. The operating range of the roll angle is similar to that of the pitch angle. Therefore we should also expect that making a small angle approximation to the roll angle does not signi¯cantly reduce the model accuracy. Again we would like to ¯nd out under what situations the vehicle might experience a large roll angle. During high lateral acceleration, the maximum limit of the roll angle relative the ground surface can be at most around

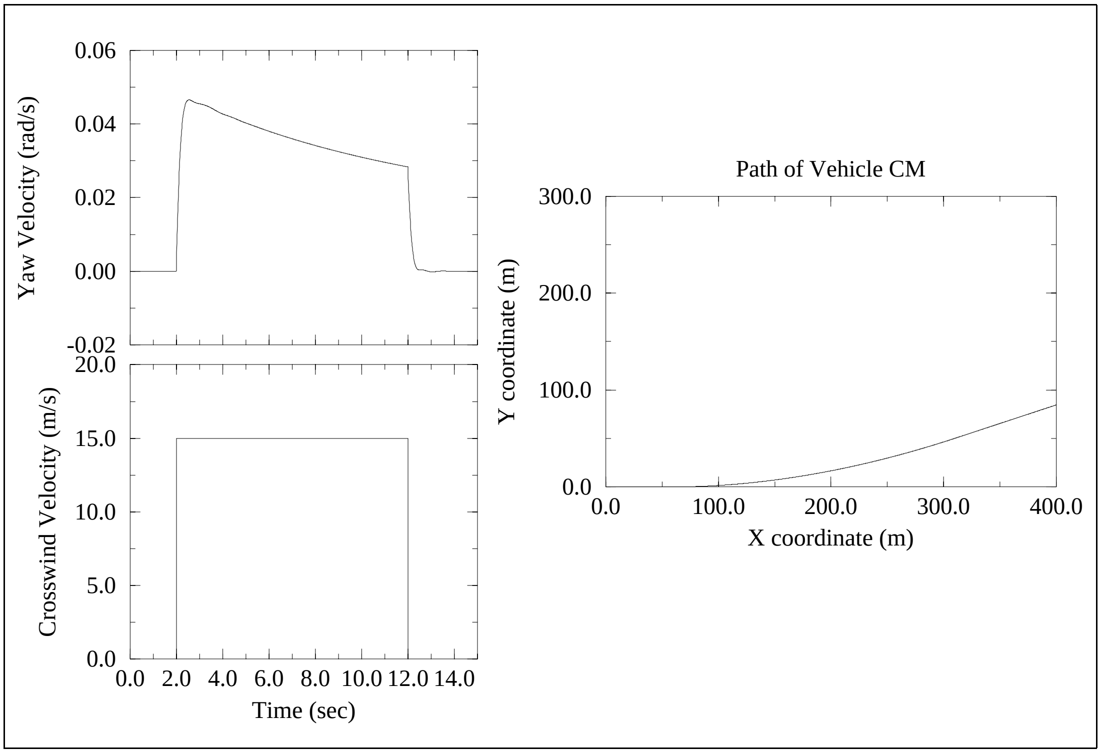

Figure 8.26: Vehicle response due to a crosswind pulse of $15\frac{\text{m}}{\text{sec}}$.

0.05 radians. Since roll angle in the model is the sum of the relative roll and the road superelevation, it is also necessary to obtain the maximum limit of the road superelevation. Usually on regular highways, road superelevations are quite small, typically under 1%. The only sections of the highway system where the road superelevation may be large are the ramps connecting one highway to another. Nevertheless the superelevation of these ramps are not large either. They are at most on the order of a few percent. To be on the conservative end, we will use a road superelevation of 10% in our simulation to test the e®ect of making a small angle approximation of the roll angle.

The roll angle of the vehicle is plotted in Figure 8.29 as the vehicle is traveling on a planar road with a superelevation of 10% and the vehicle is initially rolled to the right by 0.05 radians relative to the road surface. This sets the initial condition of the roll angle to approximately 0.15 radians. As shown in Figure 8.29, there is no noticeable deviation of the response between the approximated and the non-approximated system. The maximum di®erence of the pitch angle between the two models is below two orders of magnitude. In addition, the maximum error during transient response for any states is 2%, and during steady state is much lower. Therefore, we conclude that it is reasonable to make a small angle approximation of the roll angle.

Since the yaw angle can have any value, it is incorrect to use a small angle approximation of the yaw angle. By making small angle approximation to the pitch and the roll angle, we can achieve a 1% reduction in computation time. The reason that this improvement is less than that in the longitudinal model is because the sub-components are more complicated and there are more of them. In summary, it is reasonable to use a small angle approximation of the pitch and roll angles. While the savings in computational time is minimal it is welcome.

Linearization Around a Constant Steering Angle

At some point during a trip, the vehicle will have to travel along a curve, which can be a curvy stretch of freeway or a transition ramp from one freeway to another. Therefore it is necessary to have a linearized model for fault detection and identi¯cation system to process as the vehicle is traveling through a curved path. Each path can be considered as a constant radius curve, hence we can linearize our model around a constant steering angle.

The linearization process is identical to that of the longitudinal case except that one must be more careful in choosing the perturbation size for each coe±cient. The acceptable range for perturbation size now becomes smaller and is di®erent for each coe±cient. As shown previously, it is best to plot each coe±cient versus the perturbation size and pick the coe±cient at the appropriate region.

Once the linearized model is obtained at some nominal operating point, we can proceed to measure the e®ective range of the linearized model which can reproduce the response of the nonlinear model within a 10% error in all of the states. First, a linearized model is obtained from the nonlinear model when the vehicle is traveling straight ahead. No further investigation of the longitudinal response is required since it was already done in

Section 8.3.1. Figure 8.30 shows the longitudinal speed, lateral speed and yaw angle of the vehicle when the steering angle is perturbed by 0.01 radians. Even for a relatively large perturbation of the steering angle, the yaw rate of the linearized model matches very well that of the nonlinear model.

On the other hand, the steering input has no e®ect on the longitudinal velocity in the linearized model. The reason is that a linear system is incapable of modeling even symmetric responses of a nonlinear system. An even symmetric response is characterized by an output that is a®ected only by the magnitude and not by the direction of the input. Therefore all the modes that exhibit even symmetric behavior around zero steering angle input will be not be captured by the linearized model. The longitudinal velocity is such a mode, hence it is una®ected by any amount of perturbation applied to the steering angle. The modes that are not even symmetric are the lateral and yaw velocities. Thus a perturbation in the steering angle will directly perturb these modes as shown in Figure 8.30. Also apparent in the plot is that the yaw velocity response is much more linear than the lateral velocity in response to a steering input. With this in mind, a system linearized around a zero steering angle must be used with caution in situations where a perturbation in steering angle might be present.

Next, the system is linearized around a constant steering angle of 0.005 radians which will steer the vehicle around a constant radius curve of 620 meters at steady state. This results in a gentle lateral acceleration of about 0.1g while the vehicle is traveling at a constant speed around $26.5\frac{\mathrm{m}}{\mathrm{sec}}$. To achieve a maximum limit of 10% error between the nonlinear and the linearized system in all the signi¯cant states at steady state, the range of the perturbation size for each input variable is found and tabulated in Table 8.2. In addition, some responses of the perturbed system between the linearized and the nonlinear systems are compared. Figure 8.31 shows the responses of the longitudinal, lateral, and yaw velocities as the throttle position is increased by 15 percent. The most nonlinear state is the yaw velocity, since it is not directly a®ected by the throttle position but, rather, indirectly coupled with other states which can be directly or indirectly driven by the throttle position. In addition, a perturbation in the brake should also produce similar results as seen in Figure 8.32. As discussed in Section 8.3.1, the responses of the nonlinear system due to brake input are more linear than those due to throttle input. Hence, one should expect that the response of the linearized model due to a brake perturbation covers a wider range such that the steady state errors between the linearized and nonlinear system can be at most 10% when a step input is applied. Lastly, the vehicle longitudinal, lateral, and yaw velocities are plotted in Figure 8.33. Now the yaw velocity is directly coupled with the steering angle. Therefore one can expect that the response of the yaw angle is quite linear with respect to a perturbation of the steering angle. This can be clearly observed in Figure 8.33.

Steering Angle	Throttle Range	Steering Range	Braking Range
0 rad.	-15% to +15%	N/A	0 to 34N
0.005 rad.	-14% to +14%	-15% to +25%	0 to 27N

Table 8.2: Effective range of the linearized system.

Unlike the system linearized around a zero steering angle, this system is able to capture part of the coupled dynamics between the longitudinal and lateral motion. The reason is that the even symmetric modes around a zero steering angle are not symmetric around 0.005 steering angle. Therefore the linearized system can model the nonlinear system more accurately when the odd symmetric modes are dominant.

8.4 Summary of Model Development and Suggestions for Future Work

Two vehicle dynamics models have been developed using analytical mechanics. One is a simplified longitudinal model and the other is a full lateral and longitudinal model. The vehicle models include all major components including the suspension, tire traction, engine, brake and steering models. In addition, the model allows for arbitrary road gradient variations. Random road excitation is introduced using a first-order shaping filter approach.

In looking for ways to reduce computational complexity, a simulation study showed that small angle approximations do not significantly affect the accuracy of the simulation.

However, the same simulation study indicated no substantial reduction in computational time is realized by this approximation. Lastly, linearized vehicle dynamic models at various operating points, including straight and curved paths, are derived numerically.

The following suggestions are recommended for future work in order to refine and incorporate more features in the vehicle model. First, only a theoretical model is developed here and unfortunately vehicle parameters from different sources are used. Hence it is important to experimentally obtain all the vehicle parameters from a single test vehicle and then validate the theoretical model using experimental data. In any development process it is impossible to include all of the vehicle features at once. The list below covers the important items which have been omitted and therefore require further investigation.

- Change in steering angle due to suspension geometry.
- Linear stabilizers.
- Modeling of the unsprung mass.
- Modeling of the wheel mass.
- Static camber.
- Dynamic camber induced by suspension movement.
- Static toe-in.
- Dynamic toe-in induced by suspension movement.

Since this vehicle model will be used in fault detection filter design and evaluation, it is important to be able to model malfunctions or total failures in critical vehicle components. Modeled failures might include, for example, a flat tire, brake failure, engine malfunction and out-of-alignment steering. This development is especially important to health monitoring system evaluation applications.

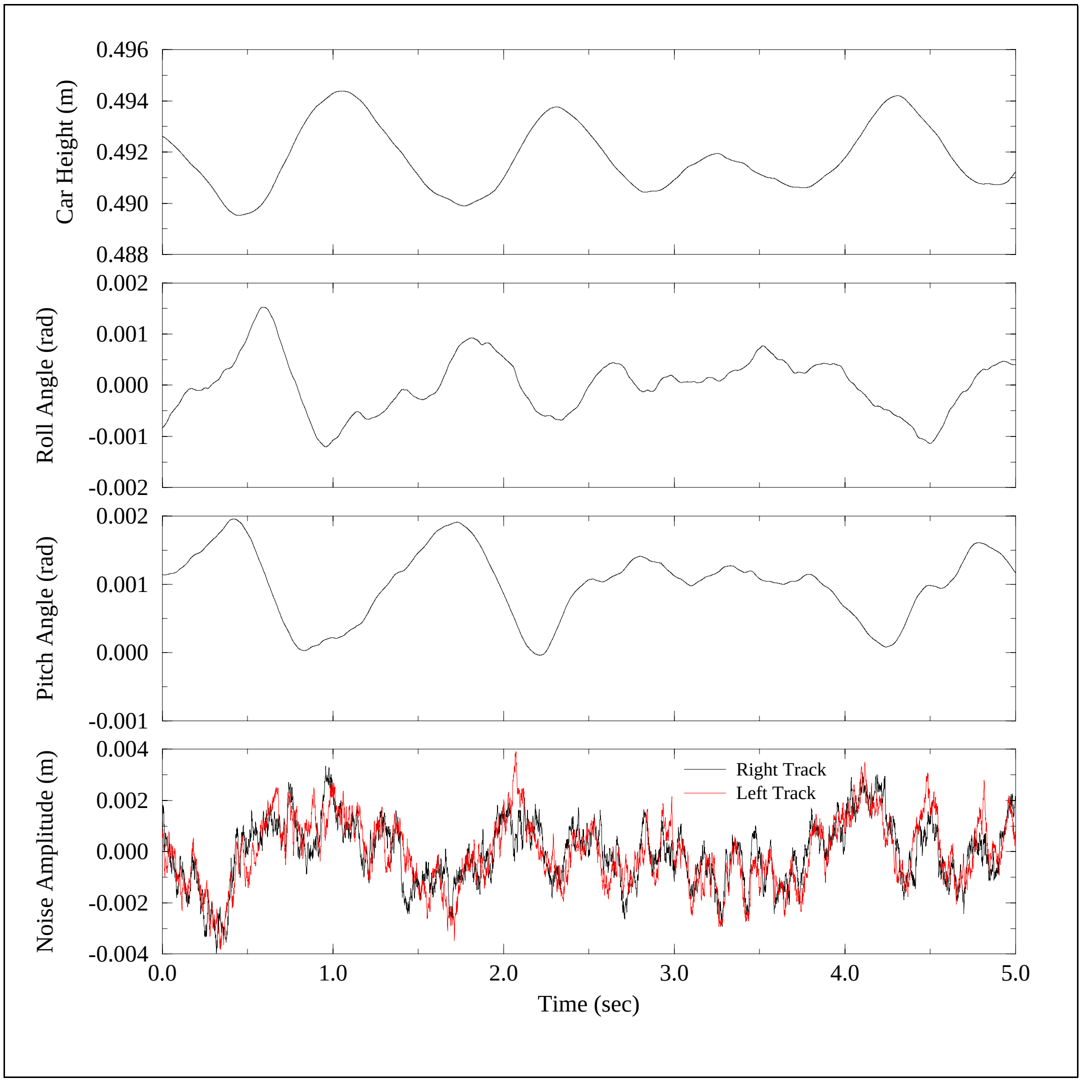

Figure 8.27: Vehicle response due to road noise.

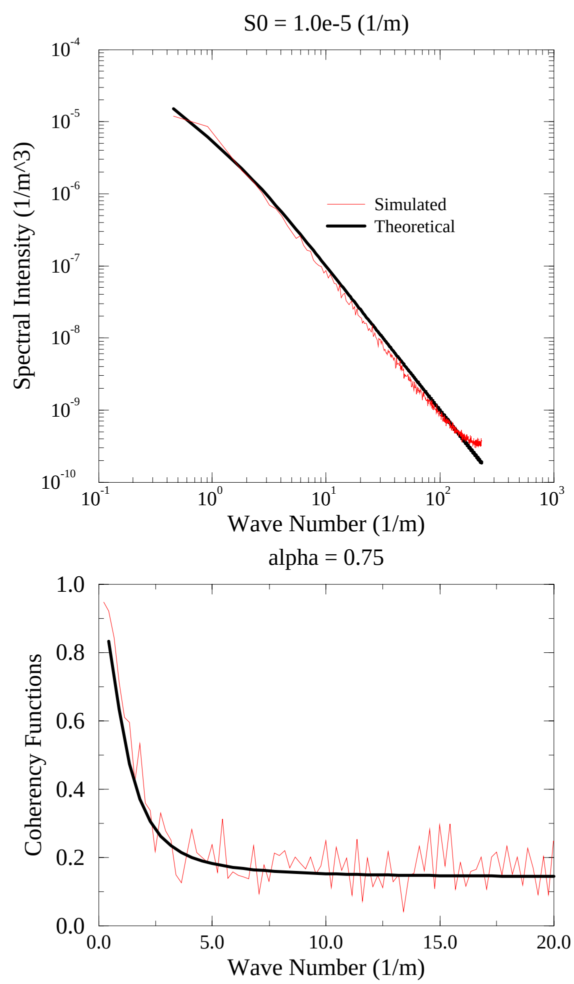

Figure 8.28: Spectral densities and coherency functions of left and right tracks.

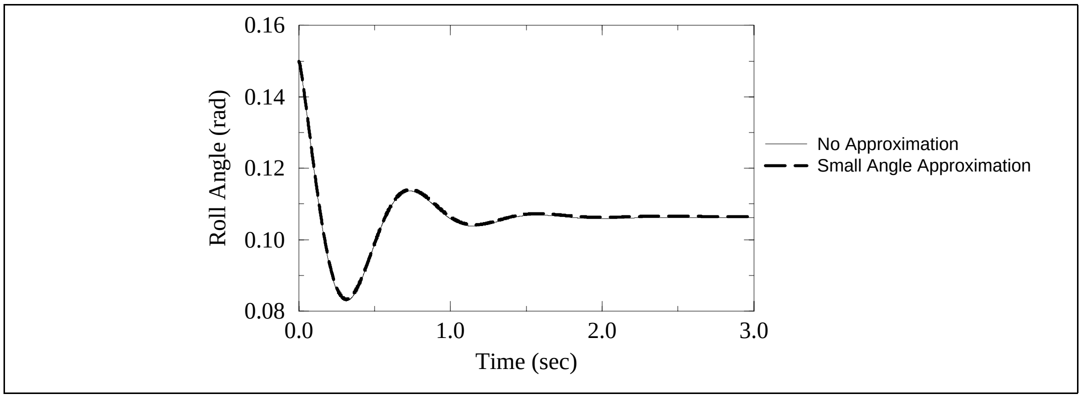

Figure 8.29: E®ect of making a small angle approximation of the roll angle.

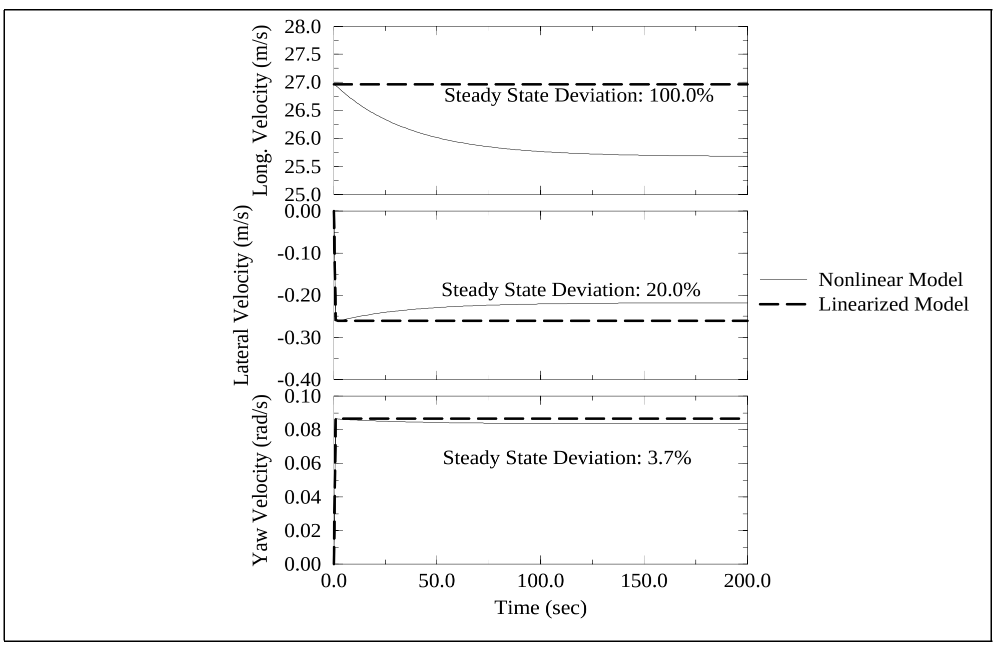

Figure 8.30: Comparison of vehicle linearized and nonlinear system responses where steering angle is perturbed by 0.01 radian.

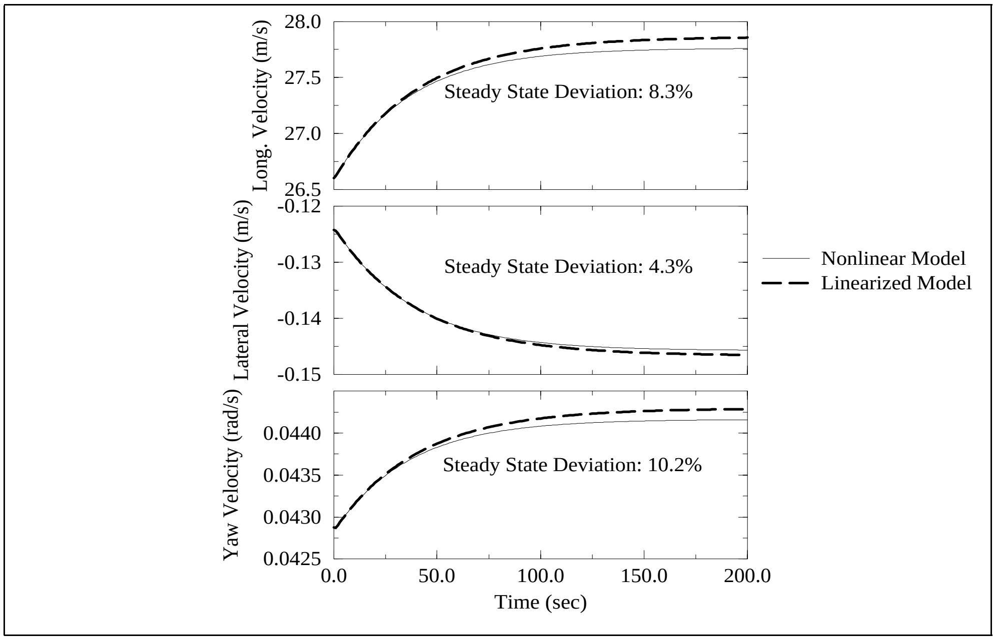

Figure 8.31: Comparison of vehicle linearized and nonlinear system responses where throttle position is perturbed by 15%.

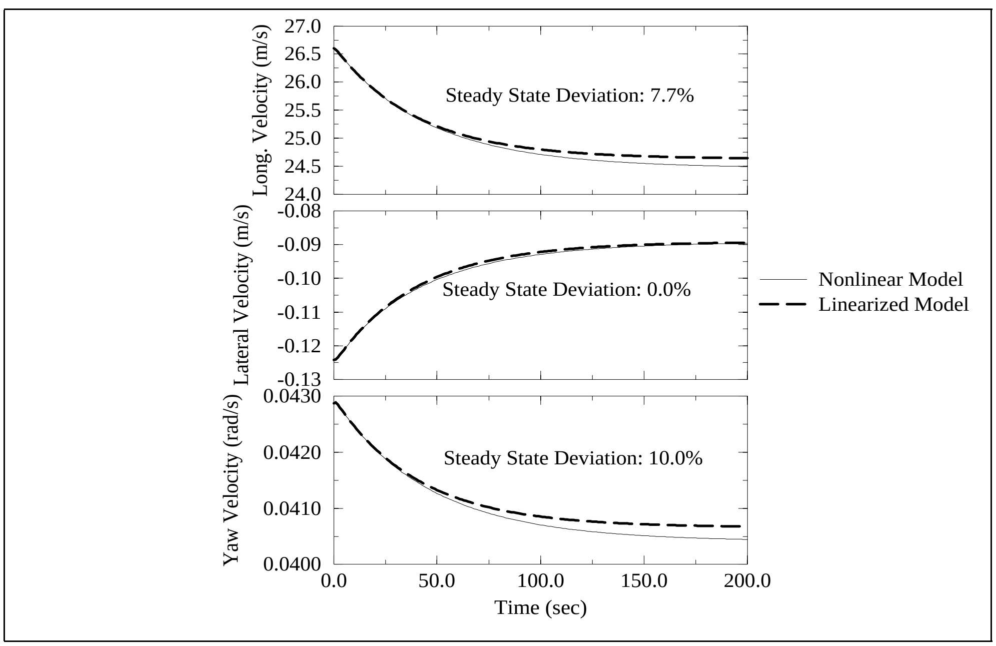

Figure 8.32: Comparison of vehicle linearized and nonlinear system responses where brake torque is perturbed by 27 N.

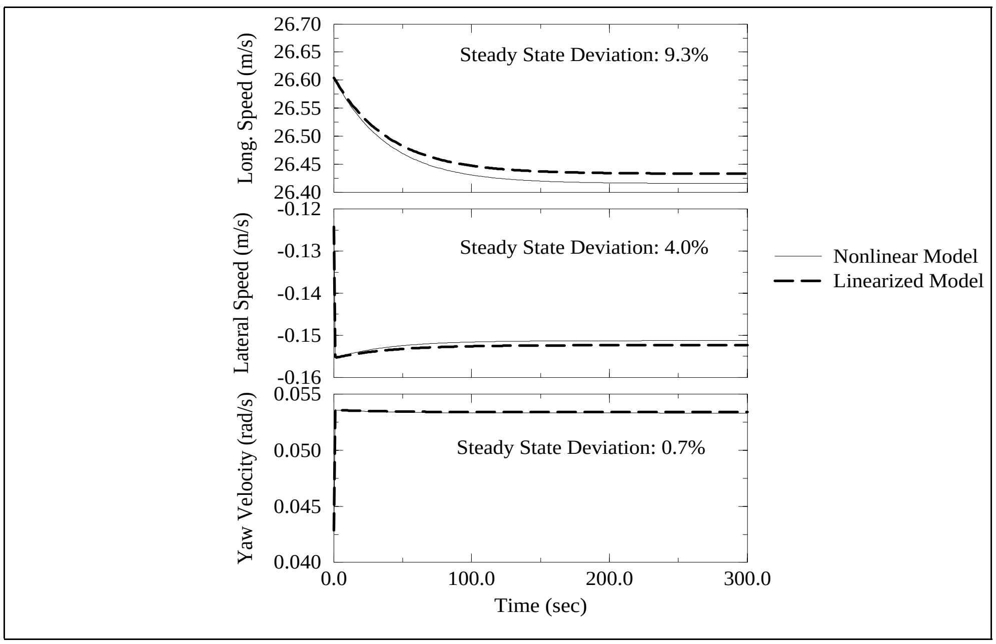

Figure 8.33: Comparison of vehicle linearized and nonlinear system responses where steering angle is perturbed by 25%.

Chapter 9

A Game Theoretic Fault Detection Filter

The fault detection filter was introduced by Beard (Beard 1971) in his doctoral thesis and later re¯ned by Jones (Jones 1973) who gave it a geometric interpretation. Since then, the fault detection ¯lter has undergone many re¯nements. White (White and Speyer 1987) derived an eigenstructure assignment design algorithm. Massoumnia (Massoumnia 1986) used advances in geometric theory to derive a complete and elegant geometric version of a fault detection ¯lter and derived a reduced-order fault detector (Massoumnia et al. 1989). Most recently, Douglas robusti¯ed the ¯lter to parameter variations (Douglas 1993) and (Douglas and Speyer 1996) and also derived a version of the ¯lter which bounds disturbance transmission (Douglas and Speyer 1995a). The background of Appendix A, design methods of Appendices B and C and the application to vehicle fault detection of Sections 2 through 5 all follow from these sources.

Common to all of these sources is an underlying structure of independent, invariant subspaces. Most design algorithms, an exception being (Douglas and Speyer 1995a), rely on spectral methods, that is, specifying eigenvalues and eigenvectors, since these methods lead directly to the needed filter structure. Spectral methods, however, also limit the applicability of fault detection filters to linear, time-invariant systems and filters designed by these methods can have poor robustness to parameter variations (Lee 1994).

For these reasons, we take a different approach to detection filter design. We look at the fault detection process as a disturbance attenuation problem and convert the process into a differential game which leads to the final design. The game is one in which the player is a state estimate and the adversaries are all of the exogenous signals, save the fault to be detected. The player attempts to exclude the adversaries from a specified portion of the state-space much in the same way that the invariant subspace structure of the fault detection filter restricts state trajectories when driven by faults. The end result is an $\mathcal{H}_\infty$-type filter which bounds disturbance transmission.

Since fault detection filters block transmission, it would seem reasonable to expect that in the limiting case when the $\mathcal{H}_\infty$ transmission bound is brought to zero, the game filter no longer approximates, but actually becomes a fault detection filter. We will prove that this is indeed the case. For linear time-invariant (LTI) systems, we will show, in fact, that the game filter becomes a Beard-Jones fault detector in the sense of (Douglas 1993): faults other than the one to be detected are restricted to a subspace which is invariant and unobservable.

The method developed here has wider applicability than current techniques since time-invariance is never assumed in the game solution. Thus, for a class of time-varying systems, results analogous to the LTI case exist in the limit as disturbance bounds are taken to zero. It is also possible with this method to deal with model uncertainty by treating it as another element in the differential game (Chichka and Speyer 1995, Mangoubi et al. 1994). In this manner, sensitivity to parameter variations can be reduced. Finally, by using a game theoretic approach, the designer has the freedom to choose the extent to which the game filter behaves as an $\mathcal{H}_\infty$ filter and the extent to which it behaves like a detection filter. This flexibility is unique to this method of fault detection filter design.

The development of game theoretic estimation closely followed the development of game theoretic control theory. The most notable and the most cited (and most unreadable) work in the latter was the paper by Doyle *et al.* (Doyle et al. 1989). The ascendant of the work presented here is the paper by Rhee and Speyer (Rhee and Speyer 1991) which derived the two Riccati solution of (Doyle et al. 1989) via the calculus of variations. It is hard to credit the first derivation of the game theoretic estimator, though (Banavar and Speyer 1991) or (Yaesh and Shaked 1993) are probable candidates.

In Sections 9.1 and 9.2, we pose a disturbance attenuation problem which models the fault detection process for a large class of systems which includes some time-varying systems. The solution to this problem leads to the game theoretic fault detection filter. In Section 9.3, we analyze sufficient conditions for our game cost to be non-positive. This will enable us to show the existence of the filter in the limit and analyze its structure. In Section 9.4, we return to the LTI case and prove that the limiting detection filter is equivalent to the Beard-Jones fault detection filter. In Section 9.5, we use the limiting form of the game theoretic filter to derive a reduced-order estimator for fault detection. Finally, in Section 9.6 we go through an example which shows that the filter is an effective fault detector for finite values of the disturbance attenuation bound and in the limit.

9.1 A Disturbance Attenuation Approach to Fault Detection

Consider a linear system in which q possible faults have been modeled:

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) + F_1(t)\mu_1(t) + \sum_{i=2}^{q} F_i(t)\mu_i(t) \tag{9.1}$$

$$y(t) = C(t)x(t) + v(t). \tag{9.2}$$

It is desired to detect the appearance of μ_1, the *target fault*, in the presence of sensor noise, v, and the possible presence of other faults $\mu_i, i \neq 1$, the *nuisance faults*. Following the standard assumptions of Appendix A, we will assume that each of the F_i's are monic and that (C,A) is an observable pair. Also, since u is a known function of $t \in [t_0, t_1]$, we will drop the Bu term for convenience. We will also neglect to explicitly show the possible time dependence of the system matrices, though the reader should keep this possibility in mind. For convenience, we define:

$$\hat{\mu}_2 = \begin{Bmatrix} \mu_2 \\ \vdots \\ \mu_q \end{Bmatrix},$$

and use the definition of $\hat{F}_i$ (a.10) so that the state equation becomes:

$$\dot{x} = Ax + F_1\mu_1 + \hat{F}_1\hat{\mu}_2$$

The definition that we propose is based upon disturbance attenuation. We use (a.11) and define the corresponding residual signal z_1 associated with μ_1 as the output signal. A disturbance attenuation problem would be to limit the transmission of the nuisance faults and the sensor noise to this output. For a fault detection filter problem we want to block this transmission entirely.

Definition 9.1 (Fault Detection Filter Problem). Find an estimator such that:

$$\frac{\|z_1\|^2}{\|\hat{\mu}_2\|^2} = 0 \quad \text{and} \quad \frac{\|z_1\|^2}{\|\mu_1\|^2} \neq 0.$$

Clearly, in the time-invariant case, the solution to the fault detection filter problem as defined by Definition A.1 solves the general fault detection filter problem that we have defined above. Later on, we will show that these definitions are equivalent in the time-invariant case by showing that the solution to Definition 9.1 solves the problem defined by Definition A.1. We need this alternative definition to account for time-varying systems. In such cases, we cannot talk about invariant subspaces and also observability becomes a trickier concept. Thus instead of defining the filter structure, we must content ourselves with merely describing its action.

9.2 A Game Theoretic Filter for Fault Detection in a General Class of Systems

We arrive at a solution to the fault detection filter problem as defined by Definition 9.1 by first solving the disturbance attenuation problem. The solution to the fault detection filter problem then comes when we take the limit of the disturbance attenuation solution. The results that we find here, however, are valuable in their own right. As we will see, the game filter that we get from the disturbance attenuation problem is itself a useful filter for fault detection.

We begin by quantifying the problem objective with a disturbance attenuation function, the ratio of the norm of the output to the norms of the inputs. For this problem, the function is:

$$D_{af} = \frac{\int_{t_0}^{t_1} \|\hat{H}_1 C(x-\hat{x})\|_{Q_1}^2 dt}{\int_{t_1}^{t_2}[\|\hat{\mu}_2\|_{M_2{}^{-1}}^2 + \|v\|_{V^{-1}}^2 + \|N_1 C(x-\hat{x})\|_{R_1}^2]dt + \|x(t_0)-\hat{x}_0\|_{P_0^{-1}}^2}$$

where $N_1 \triangleq I - \hat{H}_1$ and M_2, V, R_1, P_0 are weighting matrices. The disturbance attenuation problem is to find an estimator so that for all adversaries $\hat{\mu}_2$, $v \in L_2[t_1, t_2]$, $x(0) \in \mathcal{R}^n$:

$$D_{af} \leq \gamma. \tag{9.3}$$

We will refer to γ as the disturbance attenuation bound. Once again, the assumptions that we will make are: 1) (C, A) is a an observable pair 2) F_i, $i = 1 \dots q$ is monic 3) i, the number of iterations of (a.9) needed to make CB_i full rank is constant over the whole time interval.

To solve (9.3), convert it into a differential game with cost function:

$$\begin{aligned} J \quad = \quad & -\|x(t_0)-\hat{x}_0\|_{\Pi_0}^2 \\ & + \int_{t_0}^{t_1} \left[\|\hat{H}_1 C(x-\hat{x})\|_{Q_1}^2 - \gamma\left(\|\hat{\mu}_2\|_{M_2^{-1}}^2 + \|v\|_{V^{-1}}^2 + \|N_1 C(x-\hat{x})\|_{R_1}^2\right)\right] dt \end{aligned} \tag{9.4}$$

Note that $\Pi_0 \triangleq \gamma P_0^{-1}$. We want to find:

$$\min_{\hat{x}} \max_{v} \max_{\hat{\mu}_2} \max_{x(t_0)} J \leq 0 \tag{9.5}$$

subject to:

$$\dot{x} = Ax + \hat{F}_1 \hat{\mu}_2. \tag{9.6}$$

In anticipation of the steps which will be required for the game solution, we will rewrite the sensor noise term $\|v\|^2_{V^{-1}}$ to the equivalent $\|y - Cx\|^2_{V^{-1}}$:

$$\begin{aligned} J &= -\|x(t_0) - \hat{x}_0\|^2_{\Pi_0} \\ &\quad + \int_{t_0}^{t_1} \left[\|\hat{H}_1 C(x - \hat{x})\|^2_{Q_1} - \gamma \left(\|\hat{\mu}_2\|^2_{M_2^{-1}} + \|y - Cx\|^2_{V^{-1}} + \|N_1 C(x - \hat{x})\|^2_{R_1} \right) \right] dt \end{aligned}$$

This is a common step in the solution of quadratic minimization problems. The game problem then becomes:

$$\min_{\hat{x}} \max_{y} \max_{\hat{\mu}_2} \max_{x(t_0)} J \leq 0.$$

An interpretation of the maximization of the cost with respect to y is elusive given the measurement equation (9.2), the presence of v in (9.2), and the interplay of the different players in determining the state, x. Our view is taken from (Banavar and Speyer 1991) which looks at this extremization as incorporating a "worst-case measurement" into the game. There are other interpretations (see for instance (Yaesh and Shaked 1993)), but ultimately the question of proper interpretation becomes an exercise in tail-chasing since the mechanics of the solution remains the same as does the solution itself.

An element that is missing in our problem statement (9.4), (9.5), (9.6) is the target fault, μ_1. This is not an oversight. It would seem logical to include enhancing the transmission of μ_1 as part of the game, but there is no obvious way to include such an objective in the game cost. Moreover, extremizing the cost with respect to μ_1 leads to assumptions upon the temporal behavior of the target fault. This can be quite detrimental to filter performance if these assumptions are wrong (which is why fault detection filters are designed without any such assumptions). Thus, since μ_1 is not part of the differential game, we set it to zero for convenience when we work through the solution. This places the burden on the designer to make sure the set of faults that he chooses for the filter design leads to a well-posed problem. Well-posedness is discussed in Section 9.1 and for LTI systems is easily checked by Equation a.7.

9.2.1 Maximization with Respect to $x(t_0)$ and $\hat{\mu}_2$

We will solve our problem in two steps beginning with the subproblem:

$$\max_{\hat{\mu}_2} \max_{x(t_0)} J \leq 0.$$

The reasoning for this order of the extremizations is given in (Banavar and Speyer 1991).

We begin by appending the dynamics of the system to the cost with a Lagrange multiplier, λ^T:

$$\begin{aligned} J \;=\; & -\|x(t_0) - \hat{x}_0\|^2_{\Pi_0} + \int_{t_0}^{t_1} \Big[\|\hat{H}_1 C(x - \hat{x})\|^2_{Q_1} \\ & -\gamma \left(\|\hat{\mu}_2\|^2_{M_2^{-1}} + \|y - Cx\|^2_{V^{-1}} + \|N_1 C(x - \hat{x})\|^2_{R_1} \right) + \lambda^T (Ax + \hat{F}_1 \hat{\mu}_2 - \dot{x}) \Big] \, dt \end{aligned} \tag{9.7}$$

Integrate $\lambda \dot{x}$ by parts:

$$\begin{aligned} J = & -\|x(t_0) - \hat{x}_0\|^2_{\Pi_0} + \lambda(t_0)^T x(t_0) - \lambda(t_1)^T x(t_1) + \int_{t_0}^{t_1} \Big[\|\hat{H}_1 C(x - \hat{x})\|^2_{Q_1} \\ & -\gamma \left(\|\hat{\mu}_2\|^2_{M_2^{-1}} + \|y - Cx\|^2_{V^{-1}} + \|N_1 C(x - \hat{x})\|^2_{R_1} \right) + \lambda^T (Ax + \hat{F}_1 \hat{\mu}_2) + \dot{\lambda}^T x \Big] dt \end{aligned} \tag{9.8}$$

and then take the variation of (9.8) with respect to $\hat{\mu}_2$ and $x(t_0)$:

$$\begin{aligned} \delta J \;=\; & -\left[(x(t_0) - \hat{x}_0)^T \Pi_0 + \lambda(t_0)^T \right] \delta x(t_0) - \lambda(t_1)^T \delta x(t_1) \\ & + \int_{t_0}^{t_1} \Big\{ \Big[(x - \hat{x})^T C^T \hat{H}_1^T Q_1 \hat{H}_1 C + \gamma (y - Cx)^T V^{-1} C - \gamma (x - \hat{x})^T C^T N_1^T R_1 N_1 C \\ & \qquad + \dot{\lambda}^T + \lambda^T A \Big] \delta x + \left[-\gamma \hat{\mu}_2^T M_2^{-1} + \lambda^T \hat{F}_1 \right] \delta \hat{\mu}_2 \Big\} \, dt \end{aligned} \tag{9.9}$$

The above implies that first-order necessary conditions for J to be maximized are:

$$\hat{\mu}_2 = \frac{1}{\gamma} M_2 \hat{F}_1^T \lambda \tag{9.10a}$$

$$-\dot{\lambda} = A^T \lambda + C^T (\hat{H}_1^T Q_1 \hat{H}_1^T - \gamma N_1^T R_1 N_1) C (x - \hat{x}) + \gamma C^T V^{-1} (y - Cx) \tag{9.10b}$$

$$\lambda(t_1) = 0 \tag{9.10c}$$

$$\lambda(t_0) = \Pi_0 [x(t_0) - \hat{x}_0] \tag{9.10d}$$

Substituting (9.10a) into our dynamics (9.6) and using (9.10b), we obtain a two point boundary value problem:

$$\left\{ \begin{array}{c} \dot{x} \\ \dot{\lambda} \end{array} \right\} = \left[\begin{array}{cc} A & \frac{1}{\gamma}\hat{F}_1 M_2 \hat{F}_1^T \\ -C^T(\hat{H}_1^T Q_1 \hat{H}_1^T - \gamma N_1^T R_1 N_1 - \gamma V^{-1})C & -A^T \end{array} \right] \left\{ \begin{array}{c} x \\ \lambda \end{array} \right\} + \left\{ \begin{array}{c} 0 \\ C^T(\hat{H}_1^T Q_1 \hat{H}_1^T - \gamma N_1^T R_1 N_1)C\hat{x} - \gamma C^T V^{-1} y \end{array} \right\} \tag{9.11}$$

If we assume solutions x^* and λ^* to (9.11) and a quadratic form of the optimal return function, then:

$$\lambda^* = \Pi (x^* - x_p) \tag{9.12}$$

where x_p is a measurement dependent variable which will be shown to reduce to the estimate of the optimal state. Using (9.12) and the first equation of (9.11), the second equation of (9.11) becomes:

$$\begin{aligned} 0 \;=\; & \left[\dot{\Pi} + A^T \Pi + \Pi A + \frac{1}{\gamma} \Pi \hat{F}_1 M_2 \hat{F}_1^T \Pi + C^T \left(\hat{H}_1^T Q_1 \hat{H}_1^T - \gamma N_1^T R_1 N_1 - \gamma V^{-1} \right) C \right] x^* \\ & - \dot{\Pi} x_p - \Pi \dot{x}_p - A^T \Pi x_p - C^T(\hat{H}_1^T Q_1 \hat{H}_1^T - \gamma N_1^T R_1 N_1)C\hat{x} + \gamma C^T V^{-1} y \end{aligned} \tag{9.13}$$

Now, add and subtract

$$\gamma C^T V^{-1} C \hat{x}$$

and

$$\left[\Pi A + C^T \left(\hat{H}_1^T Q_1 \hat{H}_1^T - \gamma N_1^T R_1 N_1 - \gamma V^{-1} \right) C \right] x_p$$

to (9.13) to get:

$$\begin{aligned} 0 \;=\; & \left[\dot{\Pi} + A^T \Pi + \Pi A + \frac{1}{\gamma} \Pi \hat{F}_1 M_2 \hat{F}_1^T \Pi + C^T \left(\hat{H}_1^T Q_1 \hat{H}_1^T - \gamma N_1^T R_1 N_1 - \gamma V^{-1} \right) C \right] (x^* - x_p) \\ & - \Pi \dot{x}_p + \Pi A x_p - \left[C^T \left(\hat{H}_1^T Q_1 \hat{H}_1^T - \gamma N_1^T R_1 N_1 - \gamma C^T V^{-1} \right) C \right] (\hat{x} - x_p) \\ & + \gamma C^T V^{-1} (y - C\hat{x}) \end{aligned} \tag{9.14}$$

Thus, if we set:

$$-\dot{\Pi} = A^T \Pi + \Pi A + \frac{1}{\gamma} \Pi \hat{F}_1 M_2 \hat{F}_1^T \Pi + C^T \left(\hat{H}_1^T Q_1 \hat{H}_1^T - \gamma N_1^T R_1 N_1 - \gamma V^{-1} \right) C \tag{9.15}$$

$$\Pi \dot{x}_p = \Pi A x_p - C^T \left(\hat{H}_1^T Q_1 \hat{H}_1^T - \gamma N_1^T R_1 N_1 - \gamma V^{-1} \right) C(\hat{x} - x_p) + \gamma C^T V^{-1} (y - C\hat{x}) \tag{9.16}$$

(9.14) is satis¯ed identically. (9.15) is an estimator Riccati equation. If we set:

$$\Pi = \gamma P^{-1},$$

we can convert (9.15) into a Riccati equation:

$$\dot{P} = PA^T + PA - PC^T(V^{-1} + N_1^T R_1 N_1 - \frac{1}{\gamma}\hat{H}_1^T Q_1 \hat{H}_1^T)CP + \hat{F}_1 M_2 \hat{F}_1^T \tag{9.17}$$

as seen in (Banavar and Speyer 1991), (Rhee and Speyer 1991) and (Doyle et al. 1989). (9.16) looks like an estimator, but its ¯nal form will not become apparent until we solve the second half of the game problem.

9.2.2 Minimization with Respect to $\hat{x}$ and Maximization with Respect to y

The ¯rst part of our game solution led to optimal values for μ and $x(t_0)$:

$$\begin{aligned} \mu^* &= \frac{1}{\gamma}\hat{F}_1 M_2 \hat{F}_1^T \lambda \\ x(t_0)^* &= \Pi_0^{-1}\lambda(t_0) + \hat{x}_0 \end{aligned}$$

If we substitute these optimal values into the cost function (9.4) we obtain a new cost, $\hat{J}$, which is written as:

$$\begin{aligned} \hat{J} &= -\|\lambda(t_0)\|^2_{\Pi_0^{-1}} + \\ &\int_{t_0}^{t_1}\left[\|x - \hat{x}\|^2_{C^T(\hat{H}_1^T Q_1 \hat{H}_1 - \gamma N_1^T R_1 N_1)C} - \|\lambda\|^2_{\frac{1}{\gamma}\hat{F}_1 M_2 \hat{F}_1^T} - \gamma\|y - Cx\|^2_{V^{-1}}\right]dt \end{aligned} \tag{9.18}$$

The game is then:

$$\min_{\hat{x}} \max_{y} \hat{J} \le 0$$

subject to the dynamic equation (9.16). We begin towards the solution to this game by adding the identically zero term:

$$\|\lambda(t_0)\|^2_{\Pi(t_0)^{-1}} - \|\lambda(t_1)\|^2_{\Pi(t_1)^{-1}} + \int_{t_0}^{t_1}\frac{d}{dt}\|\lambda(t)\|^2_{\Pi^{-1}}dt = 0$$

to (9.18). After applying the boundary condition for λ at t_1 (9.10c) and carrying out the di®erentiation of the $\|\lambda\|^2_{\Pi^{-1}}$ term, we get:

$$\begin{aligned} \hat{J} = \int_{t_0}^{t_1} &\left[\|(x - \hat{x})\|^2_{C^T(\hat{H}_1^T Q_1 \hat{H}_1 - \gamma N_1^T R_1 N_1)C} - \|\lambda\|^2_{\frac{1}{\gamma}\hat{F}_1 M_2 \hat{F}_1^T} - \gamma\|y - Cx\|^2_{V^{-1}}\right. \\ &\left. + \dot{\lambda}^T \Pi^{-1}\lambda^T + \lambda^T \dot{\Pi}^{-1}\lambda + \lambda^T \Pi^{-1}\dot{\lambda}\right] dt + \|\lambda(t_0)\|^2_{\Pi^{-1}(t_0) - \Pi_0^{-1}} \end{aligned} \tag{9.19}$$

Note that (9.19) provides a boundary condition for (9.15):

$$\Pi(t_0) = \Pi_0$$

Applying this boundary condition and substituting the differential equation for λ, (9.10b), into (9.19) leads to:

$$\begin{aligned}\bar{J} = \int_{t_0}^{t_1} \Big[& \lambda^T \left(-A\Pi^{-1} - \Pi^{-1}A^T - \hat{F}_1 M_2 \hat{F}_1^T + \dot{\Pi}^{-1}\right)\lambda \\ & + (x-\hat{x})^T C^T \left(\hat{H}_1^T Q_1 \hat{H}_1 - \gamma N_1^T R_1 N_1\right) C(x-\hat{x}) \\ & - (x-\hat{x})^T C^T \left(\hat{H}_1^T Q_1 \hat{H}_1 - \gamma N_1^T R_1 N_1\right) C\Pi^{-1}\lambda \\ & - \lambda^T \Pi^{-1} C^T \left(\hat{H}_1^T Q_1 \hat{H}_1 - \gamma N_1^T R_1 N_1\right) C(x-\hat{x}) \\ & - \gamma(y-Cx)^T V^{-1}(y-Cx) \\ & + (y-Cx)^T V^{-1} C\Pi^{-1}\lambda + \lambda^T \Pi^{-1} C^T V^{-1}(y-Cx)\Big] dt \end{aligned} \tag{9.20}$$

From (9.15) the differential equation for Π^{-1} is:

$$\dot{\Pi}^{-1} = -\Pi^{-1}\dot{\Pi}\Pi^{-1} \tag{9.21a}$$

$$= \Pi^{-1}A^T + A\Pi^{-1} + \frac{1}{\gamma}\hat{F}_1 M_2 \hat{F}_1^T + \Pi^{-1}C^T(\hat{H}_1^T Q_1 \hat{H}_1 - \gamma N_1^T R_1 N_1 - \gamma V^{-1})C\Pi^{-1} \tag{9.21b}$$

After we insert (9.21) into (9.20) and cancel terms, we are left with what turns out to be a pair of quadratic terms:

$$\begin{aligned}\bar{J} = \int_{t_0}^{t_1} \Big\{ & \left[\Pi^{-1}\lambda - (x-\hat{x})\right]^T C^T \left(\hat{H}_1^T Q_1 \hat{H}_1 - \gamma N_1^T R_1 N_1\right) C \left[\Pi^{-1}\lambda - (x-\hat{x})\right] \\ & - \gamma\left[C\Pi^{-1}\lambda + (y-Cx)\right]^T V^{-1}\left[C\Pi^{-1}\lambda + (y-Cx)\right]\Big\} dt \end{aligned} \tag{9.22}$$

Now, use the solution for the optimal value of λ (9.12) and substitute into (9.22) to get:

$$\begin{aligned}\bar{J} = \int_{t_0}^{t_1} \Big[& (\hat{x}-x_p)^T C^T \left(\hat{H}_1^T Q_1 \hat{H}_1 - \gamma N_1^T R_1 N_1\right) C(\hat{x}-x_p) \\ & - \gamma(y-Cx_p)^T V^{-1}(y-Cx_p)\Big] dt \end{aligned} \tag{9.23}$$

Given the cost (9.23) and the dynamics (9.16), the solutions to this game are:

$$\hat{x}^* = x_p \tag{9.24a}$$

$$y^* = Cx_p \tag{9.24b}$$

From (9.24) we can rewrite (9.16) as:

$$\Sigma \dot{\hat{x}}^* = \Sigma A\hat{x}^* + \gamma C^T V^{-1}(y - C\hat{x}^*) \tag{9.25}$$

Since Σ is positive-definite for $\gamma > 0$, we can rewrite (9.25):

$$\dot{\hat{x}}^* = A\hat{x}^* + \gamma \Sigma^{-1} C^T V^{-1}(y - C\hat{x}^*) \tag{9.26}$$

Alternatively, the analyst could use (9.17) and:

$$\dot{\hat{x}}^* = A\hat{x}^* + PC^T V^{-1}(y - C\hat{x}^*)$$

This form of the filter is equivalent to (9.26); however, experience has shown that numerical problems are more likely to be seen when trying to find a solution to (9.17) than (9.15) when γ is brought to extremely small values. For convenience, we will write $\hat{x}$ instead of $\hat{x}^*$ when referring to the optimal state estimate with the understanding that it is the estimate that comes from the game solution which is being used.

9.2.3 Steady-State Results

In many cases, it is desired to extend the finite-time solutions of game theoretic problems to the steady-state (or infinite horizon) condition. For linear-quadratic problems, the detectability and stabilizablity of (C, A, B) ensures the existence of a unique, positive semi-definite, stabilizing solution of the Riccati equation in steady-state. Unfortunately, no such conditions exist for game-theoretic problems, except in special case where the A matrix is asymptotically stable (Green and Limebeer 1995, Lemma 3.7.3).

On the other hand, when it has possible to find a steady-state solution to the disturbance attenuation problem, it has been shown (Green and Limebeer 1995) that this solution will be in the form of the estimator given by (9.26) with Σ found via the solution of the algebraic Riccati equation:

$$0 = A^T \Sigma + \Sigma A + \frac{1}{\gamma}\Sigma \hat{F}_1 M_2 \hat{F}_1^T \Sigma + C^T(\hat{H}_1^T Q_1 \hat{H}_1^T - \gamma N_1^T R_1 N_1 - \gamma V^{-1})C$$

9.2.4 Finding the Limiting Solution

The solution of the fault detection ¯lter problem exists at the limit of the game solution when γ is taken to zero. Finding the solution or even showing that it exists in the limit, however, is not a straightforward matter. In both versions of the game Riccati equation, (9.15) and (9.17), there are terms which go to in¯nity as γ goes to zeroA similar limit has been studied in the linear quadratic regulator problem (Kwakernaak and Sivan 1972) where the cost function is always non-negative. These results are not directly applicable here since the game cost can be either positive or negative. Furthermore, it is well known (Doyle et al. 1989) that for game Riccati equations, γ has a greatest upper bound $gam\hat{m}a$ at or below which the equation has no positive-de¯nite solution. When $\gamma \leq gam\hat{m}a$ any number of di®erent phenomena can occur, for example, eigenvalues on the imaginary axis, which make positive-de¯nite solutions impossible.

By decreasing the noise weighting V to zero along with γ, that is, $V \to 0$ as $\gamma \to 0$, we can ¯nd solutions to (9.15) and (9.17) for smaller and smaller γ. While solutions are obtainable for a range of $\gamma \in (0, \infty]$ where $\gamma = \infty$ corresponds to the Kalman ¯lter, what is needed is a solution for when $\gamma = 0$. The solution follows from a pair of techniques from singular optimal control theory which are discussed in the next section.

9.3 The Limiting Case Solution via Singular Optimal Control Techniques

9.3.1 Conditions for Game Cost Non-Positivity: A Game LMI

In this section, we will ¯nd su±cient conditions for the non-positivity of the game cost. These conditions fall out after we manipulate the cost function and then set $\hat{\mu}$ to its optimal strategy found in Section 9.1. The game cost then becomes a single quadratic form:

$$J(\hat{\mu}, x(t_0), \hat{\mu}_2, v) = \int_{t_0}^{t_1} \xi^T \overline{W} \xi dt$$

where ξ is some linear vector combination of the game players. The non-negativity of the cost hinges on the sign de¯niteness of $\overline{W}$, giving rise to a linear matrix inequality. This technique was first seen in the singular optimal control theory (Bell and Jacobsen 1973) and (Clements and Anderson 1978) and the derivation seen here follows in that vein.

We begin with the cost function as given by (9.7). Note that the $(x - \hat{x})$ terms have been combined:

$$\begin{aligned} J &= -\|x(t_0) - \hat{x}_0\|^2_{\Pi_0} + \\ &\int_{t_0}^{t_1} \left[\|(x - \hat{x})\|^2_{C^T(\hat{H}_1^T Q_1 \hat{H}_1 - \gamma N_1^T R_1 N_1)C} - \gamma \|\hat{w}_2\|^2_{M_2^{-1}} - \gamma \|y - Cx\|^2_{V^{-1}} \right] dt \end{aligned} \tag{9.27}$$

We now append the dynamics of the system to (9.27) through the Lagrange Multiplier $(x - \hat{x})^T \Pi$:

$$\begin{aligned} J &= -\|x(t_0) - \hat{x}_0\|^2_{\Pi_0} + \int_{t_0}^{t_1} \Big[\|(x - \hat{x})\|^2_{C^T(\hat{H}_1^T Q_1 \hat{H}_1 - \gamma N_1^T R_1 N_1)C} \\ &\quad - \gamma \|\hat{w}_2\|^2_{M_2^{-1}} - \gamma \|y - Cx\|^2_{V^{-1}} + (x - \hat{x})^T \Pi (Ax + \hat{F}_1 \hat{w}_2 - \dot{x}) \Big] dt \end{aligned}$$

Add and subtract to (9.8) the terms $(x - \hat{x})^T \Pi A \hat{x}$ and $(x - \hat{x})^T \Pi \dot{\hat{x}}$. Collect terms to get:

$$\begin{aligned} J &= -\|x(t_0) - \hat{x}_0\|^2_{\Pi_0} + \int_{t_0}^{t_1} \Big\{ \|(x - \hat{x})\|^2_{\Pi A + C^T(\hat{H}_1^T Q_1 \hat{H}_1 - \gamma N_1^T R_1 N_1)C} \\ &\quad -\gamma \|\hat{w}_2\|^2_{M_2^{-1}} - \gamma \|y - Cx\|^2_{V^{-1}} \\ &\quad + (x - \hat{x})^T \Pi \hat{F}_1 \hat{w}_2 - (x - \hat{x})^T \Pi (\dot{x} - \dot{\hat{x}}) + (x - \hat{x})^T \left[\Pi A \hat{x} - \Pi \dot{\hat{x}} \right] \Big\} dt \end{aligned} \tag{9.28}$$

Note, we have moved ΠA into the weighting of $\|(x - \hat{x})\|^2$. More terms will appear in the weighting of $\|(x - \hat{x})\|^2$ as we manipulate the cost function. Now, integrate $(x - \hat{x})^T \Pi (\dot{x} - \dot{\hat{x}})$ by parts:

$$\begin{aligned} J &= -\|x(t_0) - \hat{x}_0\|^2_{\Pi_0 - \Pi(t_0)} - \|x(t_1) - \hat{x}(t_1)\|^2_{\Pi(t_1)} + \\ &\quad + \int_{t_0}^{t_1} \Big\{ \|(x - \hat{x})\|^2_{\dot{\Pi} + \Pi A + C^T(\hat{H}_1^T Q_1 \hat{H}_1 - \gamma N_1^T R_1 N_1)C} \\ &\qquad - \gamma \|\hat{w}_2\|^2_{M_2^{-1}} - \gamma \|y - Cx\|^2_{V^{-1}} + (x - \hat{x})^T \Pi \hat{F}_1 \hat{w}_2 \\ &\qquad + (x - \hat{x})^T \left[\Pi A \hat{x} - \Pi \dot{\hat{x}} \right] + (\dot{x} - \dot{\hat{x}})^T \Pi (x - \hat{x}) \Big\} dt \end{aligned} \tag{9.29}$$

Substitute the state equation for $\dot{x}$ (9.6) and add and subtract $\hat{x}^T A^T \Pi (x - \hat{x})$:

$$
\begin{aligned}
J \quad = \quad & -\|x(t_0) - \hat{x}_0\|^2_{\Pi_0 - \Pi(t_0)} - \|x(t_1) - x(\hat{t}_1)\|^2_{\Pi(t_1)} \\
& + \int_{t_0}^{t_1} \Big\{ \|(x - \hat{x})\|^2_{\Pi + \Pi A + A^T\Pi + C^T(\hat{H}_1^T Q_1 \hat{H}_1 - \gamma N_1^T R_1 N_1)C} \\
& \qquad - \gamma\|\hat{\mu}_2\|^2_{M_2^{-1}} - \gamma\|y - Cx\|^2_{V^{-1}} \\
& \qquad + (x - \hat{x})^T \Pi \hat{F}_1 \hat{\mu}_2 + \hat{\mu}_2^T \hat{F}_1^T \Pi (x - \hat{x})^T \\
& \qquad + (x - \hat{x})^T \left[-\Pi \dot{\hat{x}} + \Pi A \hat{x} \right] + \left[-\Pi \dot{\hat{x}} + \Pi A \hat{x} \right]^T (x - \hat{x}) \Big\} \, dt \qquad (9.30)
\end{aligned}
$$

We are now going to rewrite the $\|y - Cx\|^2_{V^{-1}}$ term by adding and subtracting $C\hat{x}$ inside of the term so that it reads $\|(y - C\hat{x}) - C(x - \hat{x})\|^2_{V^{-1}}$. Expand this quadratic term out and collect terms so that we end up with:

$$
\begin{aligned}
J \quad = \quad & -\|x(t_0) - \hat{x}_0\|^2_{\Pi_0 - \Pi(t_0)} - \|x(t_1) - \hat{x}(t_1)\|^2_{\Pi(t_1)} \\
& + \int_{t_0}^{t_1} \Big\{ \|(x - \hat{x})\|^2_{\Pi + \Pi A + A^T\Pi + C^T(\hat{H}_1^T Q_1 \hat{H}_1 - \gamma N_1^T R_1 N_1 - \gamma V^{-1})C} \\
& \quad - \gamma\|\hat{\mu}_2\|^2_{M_2^{-1}} - \gamma\|y - C\hat{x}\|^2_{V^{-1}} + (x - \hat{x})^T \Pi \hat{F}_1 \hat{\mu}_2 + \hat{\mu}_2^T \hat{F}_1^T \Pi (x - \hat{x})^T \\
& \quad + (x - \hat{x})^T \left[-\Pi \dot{\hat{x}} + \Pi A \hat{x} + \gamma C^T V^{-1}(y - C\hat{x}) \right] \\
& \quad - \left[\Pi \dot{\hat{x}} + \Pi A \hat{x} + \gamma C^T V^{-1}(y - C\hat{x}) \right]^T (x - \hat{x}) \Big\} \, dt \qquad (9.31)
\end{aligned}
$$

Using (9.25) we can eliminate a pair of terms in (9.31). We are then left with a quadratic in the form:

$$
J = \int_{t_0}^{t_1} \xi^T \overline{W} \xi dt - \|x(t_0) - \hat{x}_0\|^2_{\Pi_0 - \Pi(t_0)} - \|x(t_1) - x(\hat{t}_1)\|^2_{\Pi(t_1)},
$$

where

$$
\xi = \left\{ \begin{array}{c} (x - \hat{x}) \\ \hat{\mu}_2 \\ (y - C\hat{x}) \end{array} \right\}
$$

and

$$
\overline{W} \triangleq \begin{bmatrix} W(\Pi) & 0 \\ 0 & -\gamma V^{-1} \end{bmatrix} \qquad (9.32)
$$

and where $W(\Pi)$ is given by

$$
W(\Pi) \triangleq \begin{bmatrix} C^T\left(\hat{H}_1^T Q_1 \hat{H}_1 - \gamma V^{-1} - \gamma N_1^T R_1 N_1\right)C + A^T\Pi + \Pi A + \dot{\Pi} & \Pi \hat{F}_1 \\ \hat{F}_1^T \Pi & -\gamma M_2^{-1} \end{bmatrix} \qquad (9.33)
$$

Clearly $\overline{W}$ is negative semi-de¯nite for $¦ \geq 0$ such that:

$$W(¦) \leq 0 \tag{9.34a}$$

$$¦_0 - ¦(t_0) \geq 0 \tag{9.34b}$$

$$¦(t_1) \geq 0 \tag{9.34c}$$

Hence, we need only pay attention to the smaller LMI, $W(¦)$.

For $\gamma > 0$, it is easy to see that the Riccati equation (9.15) of the previous section is embedded in (9.33). In fact, the solution of (9.15) is the solution of $W(¦)$ which minimizes its rank (Schumacher 1983). Thus with (9.33) and (9.25), we retain the results of the previous section, but in a form which can be easily analyzed in the limit $\gamma \to 0$. If we de¯ne $\overline{V} = \lim_{\gamma \to 0} \gamma V$, su±cient conditions for $J \leq 0$ in the limit as $\gamma \to 0$ are:

$$0 = ¦\hat{F}_1 \tag{9.35a}$$

$$0 \geq \dot{¦} + A^T ¦ + ¦A + C^T \left(\hat{H}_1^T Q_1 \hat{H}_1 - \overline{V}^{-1} \right) C \tag{9.35b}$$

along with the boundary conditions (9.34b) and (9.34c).

Condition (9.35a) shows that in the limit, the Riccati matrix $¦$ has a non-trivial null space which contains the image of the nuisance failure map, $\hat{F}_1$. Moreover, those familiar with singular optimal control theory will recognize (9.35) as conditions seen previously for the singular LQ regulator. See, for example, (Bell and Jacobsen 1973)). This tells us, ¯rst of all, that the limiting form of this game ¯lter is a singular ¯lter. It is likely that similar results hold for all game theoretic ($\mathcal{H}_\infty$) ¯lters or controllers. Secondly, singular optimal control provides a wealth of results and insights which we can apply to the analysis of this ¯lter. This is, in fact, what we will do next.

9.3.2 A Riccati Equation for the Limiting Form of the Game Theoretic Filter

In Appendix A many components for the general fault detection ¯ltering problem are derived using the Goh transformation. In this section, we will again use the Goh transformation on the nuisance fault input space to obtain a Riccati equation for the limiting case game filter. The existence of the solution to this equation gives the condition for the existence of the game solution in the limit. Because this Riccati Matrix must also have a non-trivial null space, we will not be able to use the solution to this Riccati equation directly in a game filter, but this matrix will prove to be useful when we look at reduced-order detection filters.

We start with the game cost for the limiting case:

$$J^* = \lim_{\gamma \to 0} J = \int_{t_0}^{t_1} \left(\|x - \hat{x}\|^2_{C^T \hat{H}_1^T Q_1 \hat{H}_1 C} - \|y - Cx\|^2_{\overline{V}^{-1}} \right) dt$$

where $\overline{V}^{-1} \triangleq \lim_{\gamma \to 0} (\gamma V)^{-1}$. Now, define a new nuisance fault vector, ρ_1 and a new state vector, α_1:

$$\rho_1 \triangleq \int_{t_0}^{t} \hat{\mu}_2 \, dt \tag{9.36}$$

$$\alpha_1 \triangleq x - \hat{F}_1 \rho_1 \equiv x - B_1 \rho_1 \tag{9.37}$$

Note that we have defined a matrix $B_1 \triangleq \hat{F}_1$. The reason for the numbered subscripts will become apparent later. Differentiating (9.37) produces a new state equation

$$\dot{\alpha}_1 = A\alpha_1 + (AB_1 - \dot{B}_1)\rho_1 \tag{9.38}$$

and a new game cost

$$\begin{aligned} J^* = \int_{t_0}^{t_1} \Big[& \|\alpha_1 - \hat{x}\|^2_{C^T \hat{H}_1^T Q_1 H_1 C} + (\alpha_1 - \hat{x})^T C^T \hat{H}_1^T Q_1 \hat{H}_1 C B_1 \rho_1 \\ & + \rho_1^T B_1^T C^T \hat{H}_1^T Q_1 \hat{H}_1 C B_1 \rho_1 - \|y - C\alpha_1\|^2_{\overline{V}^{-1}} - (y - C\alpha_1)^T \overline{V}^{-1} C B_1 \rho_1 \\ & - \rho_1^T B_1^T C^T \overline{V}^{-1} (y - C\alpha_1) - \|\rho_1\|^2_{B_1^T C^T \overline{V}^{-1} C B_1} \Big] \, dt \end{aligned} \tag{9.39}$$

Because $\hat{H}_1$ is a projector constructed so that $\hat{H}_1 C \hat{F}_1 = 0$, the cost (9.39) is simplified as:

$$\begin{aligned} J^* = \int_{t_0}^{t_1} \Big[& \|\alpha_1 - \hat{x}\|^2_{C^T \hat{H}_1^T Q_1 \hat{H}_1 C} - \|y - C\alpha_1\|^2_{\overline{V}^{-1}} - (y - C\alpha_1)^T \overline{V}^{-1} C B_1 \rho_1 \\ & - \rho_1^T B_1^T C^T \overline{V}^{-1} (y - C\alpha_1) - \|\rho_1\|^2_{B_1^T C^T \overline{V}^{-1} C B_1} \Big] \, dt. \end{aligned}$$

Now, if $B_1^T C^T \overline{V}^{-1} C B_1 > 0$, we can solve the following differential game:

$$\min_{\hat{x}} \max_{\rho_1} J^* \leq 0$$

subject to (9.38). Because of its similarity to the derivation given in Section 9.2, we do not provide the solution here. A starting point is to convert $y - C\alpha$ into $(y - C\hat{\alpha}) + C(\alpha - \hat{\alpha})$. The solution leads to the Riccati equation:

$$
\begin{aligned}
-\dot{S} &= SA + A^T S + C^T \left(\hat{H}_1^T Q_1 \hat{H}_1 - \overline{V}^{-1}\right) C \\
&\quad + \Big[S(AB_1 - \dot{B}_1) - C^T \overline{V}^{-1} CB_1\Big](B_1^T C^T \overline{V}^{-1} CB_1)^{-1}\Big[(AB_1 - \dot{B}_1)^T S - B_1^T C^T \overline{V}^{-1} C\Big]
\end{aligned} \tag{9.40}
$$

with the boundary condition:

$$
S(t_0) = 0. \tag{9.41}
$$

It may happen, however, that $CB_1 = 0$, which would make $B_1^T C^T \overline{V}^{-1} CB_1 = 0$ and which would invalidate our Riccati equation (9.40). The remedy to this situation is to perform the same transformation as before but on the ρ_1 input space via the recursion equations:

$$
\begin{aligned}
\rho_i &= \int_{t_0}^{t} \rho_{i-1} dt \\
B_i &= AB_{i-1} - \dot{B}_{i-1} \\
\alpha_i &= x - B_i \rho_i.
\end{aligned}
$$

The process stops once a B_i is found such that $CB_i \neq 0$. The game is then:

$$
\begin{aligned}
\min_{\hat{x}} \max_{\rho_i} J^* = \int_{t_0}^{t_1} \Big[& \|\alpha_i - \hat{\alpha}\|^2_{C^T \hat{H}_1^T Q_1 H_1 C} - \|y - C\alpha_i\|^2_{\overline{V}^{-1}} - (y - C\alpha_i)^T \overline{V}^{-1} CB_i \rho_i \\
& - \rho_i^T B_i^T C^T \overline{V}^{-1} (y - C\alpha_i) - \|\rho_i\|^2_{B_i^T C^T \overline{V}^{-1} CB_i} \Big] dt
\end{aligned} \tag{9.42}
$$

subject to:

$$
\dot{\alpha}_i = A\alpha_i + (AB_i - \dot{B}_i)\rho_i. \tag{9.43}
$$

The general form of the Goh Riccati equation is then:

$$
\begin{aligned}
-\dot{S} &= SA + A^T S + C^T (\hat{H}_1^T Q_1 \hat{H}_1 - \overline{V}^{-1}) C \\
&\quad + \Big[S(AB_i - \dot{B}_i) - C^T \overline{V}^{-1} CB_i\Big](B_i^T C^T \overline{V}^{-1} CB_i)^{-1} \Big[(AB_i - \dot{B}_i)^T S - B_i^T C^T \overline{V}^{-1} C\Big]
\end{aligned} \tag{9.44}
$$

The following theorem shows that (9.44) is a Riccati equation for the limiting form of the game theoretic filter.

Theorem 9.1. The solution S to (9.44) satisfies the sufficient conditions for non-positivity of the game cost, that is, (9.35a) and (9.35b).

Proof. (The proof follows Bell and Jacobson (Bell and Jacobsen 1973, pg. 121). Due to its importance, we list it here.) Clearly, (9.44) implies that:

$$\dot{S} + SA + A^T S + C^T(\hat{H}_1^T Q_1 \hat{H}_1 - \overline{V}^{-1})C \leq 0, \qquad \forall t \in [t_0, t_1]. \tag{9.45}$$

which is (9.35a). Now, pre-multiply (9.44) by B_i^T and add $-\dot{B}_i^T S$ to both sides of the resulting equation to get:

$$\begin{aligned} -B_i^T \dot{S} - \dot{B}_i^T S &= B_i^T SA - \dot{B}_i^T S + B_i^T A^T S - B_i^T C^T \overline{V}^{-1} C \\ &+ B_i^T \Big[S(AB_i - \dot{B}_i) - C^T \overline{V}^{-1} CB_i\Big](B_i^T C^T \overline{V}^{-1} CB_i)^{-1} \Big[(AB_i - \dot{B}_i)^T S - B_i^T C^T \overline{V}^{-1} C\Big] \end{aligned} \tag{9.46}$$

Rearranging terms leads to a differential equation in $B_i^T S$ with (9.41) as the boundary condition:

$$\begin{aligned} -\frac{d}{dt}[B_i^T S] &= B_i^T SA \\ &+ B_i^T S(AB_i - \dot{B}_i)(B_i^T C^T \overline{V}^{-1} CB_i)^{-1} \Big[(AB_i - \dot{B}_i)^T S - C^T \overline{V}^{-1} CB_i\Big]. \end{aligned} \tag{9.47}$$

The solution to (9.47) given (9.41) is:

$$B_i^T(t) S(t) = 0, \qquad \forall t \in [t_0, t_1] \tag{9.48}$$

The necessary condition (9.35a) actually requires that $\hat{F}_1^T S(t) = 0$. However, $B_1 = \hat{F}_1$ and the following proposition tell us that (9.48) implies (9.35a).

Proposition 9.2. Let $i \in \mathcal{N}$ be the smallest number such that $CB_i \neq 0$. Then, the solution, S, to (9.81) is such that

$$SB_j = 0, \qquad \forall j \leq i, \ \forall t \in [t_0, t_1]$$

Proof. See (Moylan and Moore 1971). The proof given there is identical to the one just used to show that $SB_i = 0$. Induction is then used to show that $SB_j = 0$ is also true for all $j < i$.

9.4 An Unobservability Subspace Structure in the Limit

In this section, we return to time-invariant case and show that for these systems the solution to the fault detection filter problem as stated in Definition 9.1 also solves the problem as stated by Definition A.1. Thus, we can conclude that the limiting form of the game theoretic filter is a Beard-Jones fault detection filter.

Beard-Jones filters are constructed from invariant subspaces and so we will need to find an invariant subspace that is constructed by the game filter in order to prove our claim. This will require that we not only restrict ourselves to the time-invariant case, but also that we restrict our attention to the infinite-horizon problem. Hence, $\dot{¦} = 0$ and (9.35b) becomes:

$$A^T ¦ + ¦ A + C^T(\hat{H}_1^T Q_1 \hat{H}_1 - \overline{V}^{-1})C \leq 0 \tag{9.49}$$

When we specialize our analysis in this manner, we find that the required invariant subspace is the kernal of $¦$.

Theorem 9.3. Ker $¦$ is a subspace which solves the fault detection filter problem

Proof. The three conditions listed by Definition A.1 are subspace inclusion, output separability and (C, A)-invariance. Condition (9.35a) clearly implies subspace inclusion. Since we are trying to detect only one fault, output separability is satisfied trivially. Thus, all that remains is to show (C, A)-invariance.

From Wonham (Wonham 1985), a necessary and sufficient condition for Ker $¦$ to be (C, A)-invariant is that:

$$A(\mathrm{Ker}\, ¦ \cap \mathrm{Ker}\, C) \subset \mathrm{Ker}\, ¦$$

Therefore, let $x \in A(\mathrm{Ker}\, ¦ \cap \mathrm{Ker}\, C)$. That is, there exists a vector $ς$ such that:

$$x = Aς \quad \text{and} \quad ¦ ς = Cς = 0.$$

Now consider (9.35b). If we post-multiply (9.35b) by $ς$ we get:

$$¦ Aς \leq 0 \quad \Rightarrow \quad ς^T A^T ¦ Aς \leq 0.$$

Since ¦ ≥ 0, this means that:

$$\varsigma^T A^T ¦ A\varsigma = 0.$$

which implies that:

$$¦ A\varsigma = ¦ x = 0 \Rightarrow x \in \text{Ker } ¦$$

Therefore, $A(\text{Ker } ¦ \cap \text{Ker } C) \subset \text{Ker } ¦$ and so Ker ¦ is (C, A)-invariant.

Remark 1. In practice, it is not necessary to use the limiting form of the ¯lter. In many H_∞ designs, γ is not taken to its smallest possible value, but left at one which results in an acceptable compromise between all of the (usually competing) design objectives. The virtue of a game theoretic approach to fault detection ¯lter design is that it provides a *knob* with which to make the ¯lter more like a Beard-Jones ¯lter (small γ and small V) or more like a sensor noise attenuating $\mathcal{H}_\infty$ ¯lter (large γ and V).

Remark 2. It should be noted that a Beard-Jones fault detection ¯lter can detect all of the μ_j's. The ¯lter that we propose here can detect only one fault.

Remark 3. Lee and Gibson derive a ¯lter for fault detection via a minimax solution in (Lee 1994). Their results are similar to ours except that they do not investigate the relationship between their ¯lter and fault detection ¯lters and they do not look at limiting solutions.

In Section 9.1 we noted that unobservability subspaces are used in current fault detection ¯lter design methods because they allow the designer to specify (within complex conjugate symmetry) all of the eigenvalues of the ¯lter. Such design freedom exists with these subspaces because they include any invariant zero directions which arise out of the triple $(C, A, \hat{F}_1)$. It remains to be seen where the game theoretic ¯lter places invariant zeros. If all of the zeros are placed in Ker ¦, then Ker ¦ would be a detection space since it would be a (C, A)-invariant subspace containing the invariant zeros. It turns out, however, that only the right-half plane and purely imaginary zeros are contained in Ker ¦.

Theorem 9.4. Let $\overline{\mathcal{V}}^{+}$ be the subspace spanned by the invariant zero directions that correspond to the invariant zeros lying in the right-half plane. Let $\overline{\mathcal{V}}^{0}$ be the corresponding subspace for purely imaginary zeros. The (C, A)-invariant subspace, Ker ¦, created by the game-theoretic fault detection ¯lter is such that

$$\overline{\mathcal{V}}^{+} \subset \text{Ker } ¦ .$$

If $(A, \hat{F}_1)$ is stabilizable, then

$$\overline{\mathcal{V}}^{+} + \overline{\mathcal{V}}^{0} \subset \text{Ker } ¦$$

Proof. Our proof is essentially the same as the one given in (Francis 1979), though modi¯ed to ¯t the particulars of our problem. The arguments that we present here rely on geometric control theory, which means that we will have to spend a fair amount of time de¯ning subspaces and mappings between these subspaces. Once this is done, however, the actual proof comes together quickly.

We begin by de¯ning a new subspace, $\mathcal{V}^*$, the maximal $(A, \hat{F}_1)$-invariant subspace contained in $\operatorname{Ker} C$. $\mathcal{V}^*$ is the dual of the minimal (C, A)-invariant subspace $\mathcal{W}_*$ de¯ned by Theorem A.1 and in a similar manner it can be found as the limit of an iteration (Wonham 1985):

$$\begin{aligned} \mathcal{V}_0 &= \operatorname{Ker} C \\ \mathcal{V}_{i+1} &= \operatorname{Ker} C \cap A^{-1}(\operatorname{Im} \hat{F}_1 + \mathcal{V}_i) \end{aligned}$$

The notation A^{-1} should be understood as an inverse mapping and not an inverse of the matrix A. That is:

$$A^{-1}(\operatorname{Im} \hat{F}_1 + \mathcal{V}_i) \triangleq \left\{ x \in \mathcal{X} : Ax \in \operatorname{Im} \hat{F}_1 + \mathcal{V}_i \right\}$$

To be $(A, \hat{F}_1)$-invariant means that if μ were a control input then for any $x(t_0) \in \mathcal{V}^*$ there exists a matrix K such that $\mu = Kx$ and:

$$x(t) = e^{A + \hat{F}_1 K} x(t_0) \;\in\; \mathcal{V}^* \qquad \forall t \in [t_0, t_1].$$

This is not to say that we are specifying the time history of $\mu(t)$ to be a linear feedback of the states. It is just a way of illustrating the meaning of $\mathcal{V}^*$. In fact we do not need all of the space $\mathcal{V}^*$, but a portion of it. This portion, it turns out, corresponds to the invariant zeros. We define the following factor spaces:

$$\begin{aligned}\overline{\mathcal{X}} &= \mathcal{X}/(\mathcal{V}^* \cap \mathcal{W}_*)\\ \overline{\mathcal{V}} &= \mathcal{V}^*/(\mathcal{V}^* \cap \mathcal{W}_*)\end{aligned}$$

The significance of these factor spaces is through the relationship between $\overline{\mathcal{V}}$ and the $(C, A, \hat{F}_1)$ invariant zeros. If $\mathcal{M}$ is the failure input space and $K : \mathcal{M} \to \mathcal{X}$ is a feedback matrix which makes $\mathcal{V}^*$ an $(A, \hat{F}_1)$-invariant subspace, the spectrum of $A + \hat{F}_1 K$ induced on $\overline{\mathcal{V}}$ is precisely the set of invariant zeros of the triple $(C, A, \hat{F}_1)$. The invariant zero directions span $\overline{\mathcal{V}}$. Given that we are trying to prove a result about the invariant zeros, the space $\overline{\mathcal{V}}$ will clearly play a key role in our proof.

The equivalence of $\overline{\mathcal{V}}$ and the space spanned by the invariant zero directions follows from a pair of results from geometric control theory. The first, which can be found in (Morse 1973), is that the space $\mathcal{V}^* \cap \mathcal{W}_*$ is equal to the maximal controllability subspace, which we will label $\mathcal{R}^*$. $\mathcal{R}^*$ is the largest $(A, \hat{F}_1)$-invariant subspace on which the spectrum of $A + \hat{F}_1 K$ can be arbitrarily specified, hence $\mathcal{R}^* \subseteq \mathcal{V}^*$. Moreover, $\mathcal{R}^*$ is the dual to the unobservability spaces, or detection spaces, which we described earlier. The second result is that the factor space $\mathcal{V}^*/\mathcal{R}^*$, which is our space $\overline{\mathcal{V}}$, is the space spanned by the invariant zero directions. This result can be found in many places, in particular (Wonham 1985).

Define $\mathcal{V}^+$ to be the subspace of $\mathcal{V}$ on which the restriction of $A + \hat{F}_1 K$ yields eigenvalues with positive real parts. $\mathcal{V}^0$ is the corresponding space for purely imaginary eigenvalues and $\mathcal{V}^-$ the space for eigenvalues with negative real parts. Let $M : \mathcal{X} \to \overline{\mathcal{X}}$ be the canonical projection. Therefore:

$$\overline{\mathcal{V}}^+ = M\mathcal{V}^+, \qquad \overline{\mathcal{V}}^0 = M\mathcal{V}^0, \qquad \overline{\mathcal{V}}^- = M\mathcal{V}^-$$

and

$$\overline{\mathcal{V}} = \overline{\mathcal{V}}^+ + \overline{\mathcal{V}}^0 + \overline{\mathcal{V}}^-.$$

Finally, let $L : \mathcal{V} \to \mathcal{X}$ and $\overline{L} : \overline{\mathcal{V}} \to \overline{\mathcal{X}}$ be natural insertions.

To aid our understanding, we will make use of a commutative diagram. Commutative diagrams are a common tool in abstract algebra and show, pictorily, the relationships between the di®erent subspaces and the maps which take vectors from one space to another. For this proof the corresponding commutative diagram is given by Figure 9.1.

$$\begin{tikzcd}
& \mathcal{X} \arrow[r, "A+\hat{F}_1K"] \arrow[d, "M"'] & \mathcal{X} \arrow[d, "M"'] \arrow[dr, "C"] & \\
\mathcal{V} \arrow[ur] \arrow[dr] & \overline{\mathcal{X}} \arrow[r, "\overline{A}_K"] & \overline{\mathcal{X}} \arrow[r, "\overline{C}"] & \mathcal{Y} \\
& \overline{\mathcal{V}} \arrow[u, "\overline{L}"] \arrow[r, "\overline{\overline{A}}_K"] & \overline{\mathcal{V}} \arrow[u, "\overline{L}"] &
\end{tikzcd}$$

Figure 9.1: Commutative diagram for fault detection ¯lter structure.

Through the actions of M and $\overline{L}$ on the invariant subspaces $\overline{\mathcal{X}}$ and $\overline{\mathcal{V}}$ we can infer the existence of a number of induced mappings. $\overline{A}_K : \overline{X} \to \overline{X}$ is the map induced by $A + \hat{F}_1 K$ on $\overline{\mathcal{X}}$. From Figure 9.1, $\overline{A}_K$ is related to $A + \hat{F}_1 K$ via:

$$(A + \hat{F}_1 K)M = M\overline{A}_K \tag{9.50}$$

$\overline{\overline{A}}_K$ is the restriction of $\overline{A}_K$ to $\overline{\mathcal{V}}$. Its existence is guaranteed by the $\overline{A}_K$-invariance of $\overline{V}$ and it is related to $\overline{A}_K$ by:

$$\overline{L}\,\overline{A}_K = \overline{\overline{A}}_K\overline{L} \tag{9.51}$$

Finally, the map $\overline{C}$ is the unique solution to:

$$\overline{C}M = C \tag{9.52}$$

Its existence and uniqueness is guaranteed by the fact that $(\mathcal{V}^* \cap \mathcal{W}_*) \subset \operatorname{Ker} C$.

We can now begin with the actual proof. We begin by asserting that:

$$(\mathcal{V}^* \cap \mathcal{W}_*) \subset \operatorname{Ker} \text{¦}. \tag{9.53}$$

We know that this is true because Ker ¦ is a (C, A)-invariant subspace containing the range of $\hat{F}_1$ and $\mathcal{W}_*$ is the smallest of all such subspaces. Hence, $\mathcal{W}_* \subset \operatorname{Ker} \text{¦}$, which implies (9.53). From (9.53) we can assert that there exists a unique symmetric matrix $\tilde{\text{¦}}$ such that:

$$\text{¦} = M^T \tilde{\text{¦}} M. \tag{9.54}$$

Using (9.54),(9.50), and (9.52), we can rewrite (9.49) as:

$$M^T \left[\overline{A}_K^T \tilde{\text{¦}} + \tilde{\text{¦}} \overline{A}_K + \overline{C}^T (\hat{H}_1^T Q_1 \hat{H}_1 - \overline{V}^{-1}) \overline{C} \right] M \leq 0$$

Because M is a canonical projector, it has a right inverse which means that we can rework the above inequality into:

$$\overline{A}_K^T \tilde{\text{¦}} + \tilde{\text{¦}} \overline{A}_K + \overline{C}^T (\hat{H}_1^T Q_1 \hat{H}_1 - \overline{V}^{-1}) \overline{C} \leq 0 \tag{9.55}$$

We need now need to go one step further and consider the system restricted to the subspace $\overline{\mathcal{V}}$. Pre-multiply (9.55) by $\overline{L}^T$ and post-multiply by $\overline{L}$. Since $\overline{L}$ is insertion map of a space which lies in $\operatorname{Ker} C$, it follows that $\overline{C}\ \overline{L} = 0$. Thus, from (9.51) we can rewrite (9.55) as:

$$\overline{\overline{A}}_K^T \overline{L}^T \tilde{\text{¦}} \overline{L} + \overline{L}^T \tilde{\text{¦}} \overline{L}\ \overline{\overline{A}}_K \leq 0 \tag{9.56}$$

Now, let λ_j be the jth eigenvalue of $\overline{\overline{A}}_K$ such that $\operatorname{Re} \lambda > 0$ and let $z_{j_{i_0}}$, $j_i = 1 \ldots j_{i_0} \ldots \alpha_j$ be one of the corresponding generalized eigenvectors. Here α_j is the algebraic multiplicity of λ_j. Pre-multiply (9.56) by $z_{j_{i_0}}^*$, the conjugate transpose of $z_{j_{i_0}}$, and post-multiply by $z_{j_{i_0}}$ to get:

$$z_{j_{i_0}}^* \left(\overline{\overline{A}}_K^T \overline{L}^T \tilde{\text{¦}} \overline{L} + \overline{L}^T \tilde{\text{¦}} \overline{L}\ \overline{\overline{A}}_K \right) z_{j_{i_0}} = (2 \operatorname{Re} \lambda)\, z_{j_{i_0}}^* \overline{L}^T \tilde{\text{¦}} \overline{L} z_{j_{i_0}} \leq 0.$$

The latter inequality implies that $\hat{\Pi}\overline{L}z_{j_{i_0}} = 0$ since $\hat{\Pi}$ is positive semi-definite. As stated earlier, the eigenvalues of $\overline{\overline{A}}_K$ are the invariant zeros of the triple $(C, A, \hat{F}_1)$, meaning that $\overline{L}z_{j_{i_0}}$ is the invariant zero direction. We have just shown that this direction lies in the kernal of $\hat{\Pi}$ which is sufficient to claim that it lies in the kernal of Π itself. Since $z_{j_{i_0}}$ was chosen arbitrarily out of the set of generalized eigenvalues, this holds for all z_{j_i} in the set. Since λ_j was chosen arbitrarily out of the set of unstable eigenvalues of $\overline{\overline{A}}_K$, this holds for all such eigenvalues. This proves the first half of our theorem.

To prove the second half of our theorem we need to make the additional assumption that $(A, \hat{F}_1)$ is stabilizable. This new assumption is fairly benign and was also made by (Banavar and Speyer 1991). $(A, \hat{F}_1)$ stabilizable implies that $(A + \hat{F}_1 K, \hat{F}_1)$ and $(\overline{A}_K, M\hat{F}_1)$ are stabilizable. The latter is proven in (Wonham 1985)). Now let $\lambda_k = j\omega$ be an eigenvalue of $\overline{\overline{A}}_K$ and let $z_{j_{k_0}}$, $j_k = 1 \ldots j_{k_0} \ldots \alpha_k$ be one of the corresponding generalized eigenvectors. Pre-multiplying (9.55) by $z^*_{j_{k_0}}\overline{L}^T$ and post-multiplying by $\overline{L}z_{j_{k_0}}$ leads to:

$$z^*_{j_{k_0}}\overline{L}^T\left(\overline{A}_K\hat{\Pi} + \hat{\Pi}\overline{A}_K\right)\overline{L}z_{j_{k_0}} = (2\,\mathrm{Re}\,\lambda)\, z^*_{j_{k_0}}\overline{L}^T\hat{\Pi}\overline{L}z_{j_{k_0}} = 0$$

which implies

$$z^*_{j_{k_0}}\overline{L}^T\left(\overline{A}_K\hat{\Pi} + \hat{\Pi}\overline{A}_K\right) = z^*_{j_{k_0}}\overline{L}^T\hat{\Pi}(-\lambda I + \overline{A}_K) = 0$$

We also know that $\hat{\Pi}\hat{F}_1 M = 0$ since $\Pi\hat{F}_1 = 0$. Hence we can augment the above equation to read:

$$z^*_{j_{k_0}}\overline{L}^T\hat{\Pi}\left[\overline{A}_K - \lambda I,\ \hat{F}_1 M\right] = 0$$

This implies that $z^*_{j_{k_0}}\overline{L}^T\hat{\Pi} = 0$, since the stabilizability assumption implies $[\overline{A}_K - \lambda I,\ \hat{F}_1 M]$ is full rank. From this we can conclude that $\overline{L}z_{j_{k_0}} \in \mathrm{Ker}\,\hat{\Pi}$ which, by using the same arguments as before, leads to the conclusion that the invariant zero directions corresponding to the purely imaginary zeros lie in the kernal of Π.

Even though invariant zeros will not destabilize the game-theoretic filter as was just shown, it is still possible that a left-half plane zero could be in a location which is undesirable. This potential shortcoming is mitigated somewhat by the fact that zeros are rare for non-square systems.

9.5 Fault Detection with the Limiting Form of the Game Theoretic Filter

In this section, we will show that a reduced-order fault detector can be derived from the limiting form of the game theoretic filter. The results from this section are more easily applied to time-invariant systems, but we will give an overview of how to apply these results to time-varying systems.

The reduced-order filter falls out from the fact that positive semi-definite, symmetric matrices such as Π always have non-singular, transformations - say Γ - that are orthonormal ($\Gamma^T \Gamma = I$) and that convert the matrix into the form:

$$\Gamma \Pi \Gamma^T = \begin{bmatrix} \bar{\Pi} & 0 \\ 0 & 0 \end{bmatrix}, \tag{9.57}$$

where $\bar{\Pi}$ is positive definite. From (9.57), we can derive transformations on system matrices which will allow us to factor out the portion of the state-space which corresponds to Ker Π. First define:

$$C \Gamma^T = \begin{bmatrix} C_1 & C_2 \end{bmatrix}, \qquad \Gamma A \Gamma^T = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \qquad \Gamma \hat{F}_1 = \begin{bmatrix} F_{11} \\ F_{12} \end{bmatrix}$$

Because $\Pi \hat{F}_1 = 0$ implies $\Gamma \Pi \hat{F}_1 = 0$, we can immediately conclude that:

$$\Gamma \Pi \Gamma^T \Gamma \hat{F}_1 = \begin{bmatrix} \bar{\Pi} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} F_{11} \\ F_{22} \end{bmatrix} = \bar{\Pi} F_{11} = 0.$$

Which, since $\bar{\Pi}$ is positive-definite, implies:

$$F_{11} = 0.$$

Now, using Γ we can partition the state-space as:

$$\hat{\eta} = \begin{Bmatrix} \hat{\eta}_1 \\ \hat{\eta}_2 \end{Bmatrix} = \Gamma \hat{x}.$$

Pre-multiply (9.25) by Γ and make use of the identity $\Gamma^T \Gamma = I$ to get:

$$(\Gamma \Pi \Gamma^T) \dot{\hat{\eta}} = (\Gamma \Pi \Gamma^T)(\Gamma A \Gamma^T) \hat{\eta} + \Gamma C^T \overline{V}^{-1} (y - C \Gamma^T \hat{\eta}) \tag{9.58}$$

The transformed filter equation (9.58) is seen to be:

$$\begin{bmatrix} \bar{\Pi} & 0 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} \dot{\hat{\eta}}_1 \\ \dot{\hat{\eta}}_2 \end{Bmatrix} = \begin{bmatrix} \bar{\Pi} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{Bmatrix} \hat{\eta}_1 \\ \hat{\eta}_2 \end{Bmatrix} + \begin{Bmatrix} C_1^T \\ C_2^T \end{Bmatrix} \overline{V}^{-1} \left(y - \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{Bmatrix} \hat{\eta}_1 \\ \hat{\eta}_2 \end{Bmatrix} \right) \tag{9.59}$$

From (9.59) we get a dynamic equation for $\hat{\eta}_1$:

$$\bar{\Pi}\dot{\hat{\eta}}_1 = \bar{\Pi}A_{11}\hat{\eta}_1 + \bar{\Pi}A_{12}\hat{\eta}_2 + C_1^T\overline{V}^{-1}(y - C_1\hat{\eta}_1 - C_2\hat{\eta}_2) \tag{9.60}$$

and a static equation for $\hat{\eta}_2$:

$$\hat{\eta}_2 = (C_2^T\overline{V}^{-1}C_2)^{-1}C_2^T\overline{V}^{-1}(y - C_1\hat{\eta}_1). \tag{9.61}$$

Define

$$K \triangleq (C_2^T\overline{V}^{-1}C_2)^{-1}C_2^T\overline{V}^{-1} \tag{9.62}$$

so that the substitution of (9.62) and (9.61) into (9.60) gives us an estimator for $\hat{\eta}_1$:

$$\dot{\hat{\eta}}_1 = A_{11}\hat{\eta}_1 + \left[\bar{\Pi}^{-1}C_1^T\overline{V}^{-1}(I - C_2K) + A_{12}K\right](y - C_1\hat{\eta}_1). \tag{9.63}$$

To see that the reduced-order estimator (9.63) is unaffected by the nuisance fault $\hat{\mu}_2$, we will derive the error equation for the reduced-order filter. Define:

$$\eta = \begin{Bmatrix} \eta_1 \\ \eta_2 \end{Bmatrix} \triangleq \Gamma x, \qquad e_1 \triangleq \hat{\eta}_1 - \eta_1, \qquad e_2 \triangleq \hat{\eta}_2 - \eta_2$$

We begin by premultiplying the dynamic equation (9.6) by the Riccati matrix Π. Since $\Pi\hat{F}_1 = 0$, we get:

$$\Pi\dot{x} = \Pi A x.$$

This can be pre-multiplied by Γ and manipulated into:

$$\begin{bmatrix} \bar{\Pi} & 0 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} \dot{\eta}_1 \\ \dot{\eta}_2 \end{Bmatrix} = \begin{bmatrix} \bar{\Pi} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{Bmatrix} \eta_1 \\ \eta_2 \end{Bmatrix}. \tag{9.64}$$

As with the estimator equation, (9.64) shows that only a portion of the state-space possesses dynamics:

$$\bar{\Pi}\dot{\eta}_1 = \bar{\Pi}A_{11}\eta_1 + \bar{\Pi}A_{12}\eta_2 \tag{9.65}$$

Using (9.65) to get an error equation would leave terms in η_2 or e_2. In anticipation of this, we transform the measurement equation:

$$y = Cx + v = C\Gamma^T\Gamma x + v = C_1\eta_1 + C_2\eta_2 + v \tag{9.66}$$

and use (9.61) to solve for e_2:

$$e_2 = (C_2^T\overline{V}^{-1}C_2)^{-1}(C_2^T\overline{V}^{-1}C_1e_1 + C_2^T\overline{V}^{-1}v) = K(C_1e_1 - v) \tag{9.67}$$

Subtract (9.65) from (9.60) and substitute (9.66) for y:

$$\overline{\Gamma}\dot{e}_1 = \overline{\Gamma}A_{11}e_1 + \overline{\Gamma}A_{12}e_2 + C_1^T\overline{V}^{-1}C_1e_1 + C_1^T\overline{V}^{-1}C_2e_2 + C^T\overline{V}^{-1}v$$

Using (9.67) and collecting terms, we can turn the previous equation into:

$$\begin{aligned} \dot{e}_1 &= \left[A_{11} - \overline{\Gamma}^{-1}C_1^T\overline{V}^{-1}(I - C_2K)C_1 - A_{12}KC_1\right]e_1 \\ &\quad + \left[\overline{\Gamma}^{-1}C_1^T\overline{V}^{-1}(I - C_2K) + A_{12}K\right]v. \end{aligned} \tag{9.68}$$

Note that the nuisance fault, μ_2, appears nowhere in the estimator (9.63) nor in the error equation (9.68). Thus, in the limit, we get a reduced-order estimator completely uninfluenced by the nuisance faults. The term $(C_2^T\overline{V}^{-1}C_2)^{-1}$ appears in various places in the reduced-order estimator. This inverse will always exist since $\overline{V}$ is positive definite and since the assumption of (C, A) observability guarantees that C_2 will have full column rank.

Remark 4. The reduced-order filter derived here is similar to the residual generator derived by Massoumnia, *et al.* in (Massoumnia et al. 1989). An important difference, however, is that Massoumnia begins his design process by factoring out the reachable space of the nuisance faults. As a result, he has the freedom to use any kind of filter design technique for the lower dimensional state-space. The trade-off, however, is that the system reduction in Massoumnia's filter is sensitive to the inexactness of the plant model. Variations in the plant will change the reachable subspace and may, as a result, degrade the performance of the reduced-order detector. In the game filter, the order reduction comes at the end of the design process. Thus, there is no design freedom left to tune the reduced-order ¯lter, but the game formulation used to obtain the ¯lter makes it possible to account for model uncertainties.

The Goh transformation and corresponding Riccati equation greatly extend our ability to analyze the reduced-order estimator. In fact with the Goh Riccati equation we can show that there always exists a stabilizing solution for the reduced order estimator. Applying the transformation ¡ to (9.44), we get:

$$\begin{aligned} -\text{¡}\dot{S}\text{¡}^T &= \text{¡}S\text{¡}^T\text{¡}A\text{¡}^T + \text{¡}A^T\text{¡}^T\text{¡}S\text{¡}^T \\ &\quad + \text{¡}C^T(\hat{H}_1^T Q_1 \hat{H}_1 - \overline{V}^{-1})C\text{¡}^T + \overset{1}{\text{¡}}\left(B_i^T C^T \overline{V}^{-1} C B_i\right)^{-1}\overset{1}{\text{¡}}^T \end{aligned}$$

where, for notational convenience, $\overset{1}{\text{¡}}$ is de¯ned as

$$\overset{1}{\text{¡}} = \left[\text{¡}S\text{¡}^T\left(\text{¡}A\text{¡}^T\text{¡}B_i - \text{¡}\dot{B}_i\right) - \text{¡}C^T\overline{V}^{-1}C\text{¡}^T\text{¡}B_i\right]$$

De¯ne:

$$\text{¡}B_i = \begin{bmatrix} B_{11} \\ B_{12} \end{bmatrix}.$$

As in section 9.4, the necessary condition $SB_i = 0$ will lead to $B_{11} = 0$ since $\text{¡}S\text{¡}^T\text{¡}B_i = 0 \quad \Rightarrow \quad \overline{S}B_{11} = 0$ and $\overline{S}$ is positive-de¯nite. Also, if we carry the transformation through, a number of terms fall out because the projector $\hat{H}_1$ has been constructed so that:

$$\begin{aligned} \hat{H}_1 C B_i = 0 \quad &\Rightarrow \quad \hat{H}_1 C\text{¡}^T\text{¡}B_i = 0 \\ &\Rightarrow \quad \begin{bmatrix} \hat{H}_1 C_1 & \hat{H}_1 C_2 \end{bmatrix}\begin{bmatrix} 0 \\ B_{12} \end{bmatrix} = 0 \\ &\Rightarrow \quad \hat{H}_1 C_2 B_{12} = 0 \end{aligned} \tag{9.69}$$

We we show later that B_i can always be augmented so that B_{12} is an invertible square matrix. Hence (9.69) implies:

$$\hat{H}_1 C_2 = 0. \tag{9.70}$$

Using (9.70) and working through all of the transformations leads to:

$$\begin{bmatrix} -\dot{\overline{S}} & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \overline{S}A_{11} & \overline{S}A_{12} \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} A_{11}^T\overline{S} & 0 \\ A_{12}^T\overline{S} & 0 \end{bmatrix}$$

$$+\left(\begin{bmatrix}\overline{S}A_{12}B_{12}-\overline{S}B_{11}\\ 0\end{bmatrix}-\begin{bmatrix}C_1^T\overline{V}^{-1}C_2B_{12}\\ C_2^T\overline{V}^{-1}C_2B_{12}\end{bmatrix}\right)(B_{12}^TC_2^T\overline{V}^{-1}C_2B_{12})^{-1}$$
$$\times\left(\begin{bmatrix}B_{12}^TA_{12}\overline{S}-B_{11}\overline{S} & 0\end{bmatrix}-\begin{bmatrix}B_{12}^TC_2^T\overline{V}^{-1}C_1^T & B_{12}^TC_2^T\overline{V}^{-1}C_2^T\end{bmatrix}\right)$$
$$+\begin{bmatrix}C_1^T(\hat{H}_1^TQ_1\hat{H}_1-\overline{V}^{-1})C_1 & -C_1^T\overline{V}^{-1}C_2\\ -C_2^T\overline{V}^{-1}C_1 & -C_2^T\overline{V}^{-1}C_2\end{bmatrix} \tag{9.71}$$

From (9.71) we get three equations:

$$\begin{aligned}-\dot{\overline{S}} &= C_1^T(\hat{H}_1^TQ_1\hat{H}_1-\overline{V}^{-1})C_1+\overline{S}A_{11}+A_{11}^T\overline{S}\\ &\quad+\left(\overline{S}A_{12}B_{12}-\overline{S}B_{11}-C_1^T\overline{V}^{-1}C_2B_{12}\right)\left(B_{12}^TC_2^T\overline{V}^{-1}C_2B_{12}\right)^{-1}\\ &\quad\times\left(\overline{S}A_{12}B_{12}-\overline{S}B_{11}-C_1^T\overline{V}^{-1}C_2B_{12}\right)^T && (9.72)\\ 0 &= -C_1^T\overline{V}^{-1}C_2+\overline{S}A_{12}-\left(\overline{S}A_{12}B_{12}-\overline{S}B_{11}-C_1^T\overline{V}^{-1}C_2^TB_{12}\right)\\ &\quad\times\left(B_{12}^TC_2^T\overline{V}^{-1}C_2B_{12}\right)^{-1}B_{12}^TC_2^T\overline{V}^{-1}C_2 && (9.73)\\ 0 &= -C_2^T\overline{V}^{-1}C_2+C_2^T\overline{V}^{-1}C_2B_{12}\left(B_{12}^TC_2^T\overline{V}^{-1}C_2B_{12}\right)^{-1}B_{12}^TC_2^T\overline{V}^{-1}C_2. && (9.74)\end{aligned}$$

However, if we post-multiply (9.74) by B_{12} and cancel terms we obtain the identity $0 = 0$. If we post-multiply (9.73) by B_{12} we obtain:

$$0=\overline{S}B_{11} \quad\Rightarrow\quad B_{11}=0. \tag{9.75}$$

Thus, we need only (9.72), which thanks to (9.75) can be simplified to:

$$\begin{aligned}-\dot{\overline{S}} &= C_1^T(\hat{H}_1^TQ_1\hat{H}_1-\overline{V}^{-1})C_1+\overline{S}A_{11}+A_{11}^T\overline{S}+\left(\overline{S}A_{12}B_{12}-C_1^T\overline{V}^{-1}C_2B_{12}\right)\\ &\quad\times\left(B_{12}^TC_2^T\overline{V}^{-1}C_2B_{12}\right)^{-1}\left(\overline{S}A_{12}B_{12}-C_1^T\overline{V}^{-1}C_2B_{12}\right)^T. && (9.76)\end{aligned}$$

Now if $i=1$, then $B_i = \hat{F}_1$ and the rank of $\hat{F}_1$ equals the dimension of the kernal of S. $B_{12} = F_{12}$ will then be square and, moreover, it will be invertible since $\hat{F}_1$ was assumed monic. Given this, we can simplify (9.76) to:

$$\begin{aligned}-\dot{\overline{S}} &= C_1^T\left(\hat{H}_1^TQ_1\hat{H}_1-\overline{V}^{-1}\right)C_1+\overline{S}A_{11}+A_{11}^T\overline{S}\\ &\quad+\left(\overline{S}A_{12}-C_1^T\overline{V}^{-1}C_2\right)\left(C_2^T\overline{V}^{-1}C_2\right)^{-1}\left(\overline{S}A_{12}-C_1^T\overline{V}^{-1}C_2\right)^T && (9.77)\\ \overline{S}(t_0) &= 0 && (9.78)\end{aligned}$$

where the boundary condition comes from (9.41). This leads us to the key result of this section.

Theorem 9.5. The solution $\overline{S}$ to (9.77) gives a stabilizing solution for the reduced-order estimator (9.63).

Proof. Using the same transformation to derive both (9.77) and (9.63) will ensure that $\overline{S}$ is of proper dimension for (9.63). Substitute $\overline{S}$ into (9.63) directly for Π. The resulting estimator is:

$$\dot{\hat{\eta}}_1 = \left(A_{11} - \left[\overline{S}^{-1} C_1^T \overline{V}^{-1}(I - C_2 K) + A_{12} K\right] C_1\right) \hat{\eta}_1 + \left[\overline{S}^{-1} C_1^T \overline{V}^{-1}(I - C_2 K) + A_{12} K\right] y.$$

where $K \triangleq (C_2^T \overline{V}^{-1} C_2)^{-1} C_2^T \overline{V}^{-1}$. Clearly, the stability of the estimator depends upon the closed-loop state matrix, $(A_{11} - [\overline{S}^{-1} C_1^T \overline{V}^{-1}(I - C_2 K) + A_{12} K] C_1)$. Now, if we go back to (9.77), multiply out the quadratic, and use the de¯nition for K, we get:

$$\begin{aligned} -\overline{S} \quad &= \quad \overline{S}(A_{11} - A_{12} K C_1) + (A_{11} - A_{12} K C_1)^T \overline{S} \\ &\quad + C_1^T \left[\hat{H}_1^T Q_1 \hat{H}_1 - \overline{V}^{-1}(I - C_2 K)\right] C_1 + \overline{S} A_{12} (C_2^T \overline{V}^{-1} C_2)^{-1} A_{12}^T \overline{S}. \end{aligned} \tag{9.79}$$

If we add and subtract $C_1^T \overline{V}^{-1}(I - C_2 K) C_1$ to (9.79) and rearrange terms we get:

$$\begin{aligned} -\overline{S} \quad &= \quad \overline{S}\left[A_{11} - A_{12} K C_1 - \overline{S}^{-1} C_1^T \overline{V}^{-1}(I - C_2 K) C_1\right] \\ &\quad + \left[A_{11} - A_{12} K C_1 - \overline{S}^{-1} C_1^T \overline{V}^{-1}(I - C_2 K) C_1\right]^T \overline{S} \\ &\quad + C_1^T \left[\hat{H}_1^T Q_1 \hat{H}_1 + \overline{V}^{-1}(I - C_2 K)\right] C_1 + \overline{S} A_{12} (C_2^T \overline{V}^{-1} C_2)^{-1} A_{12}^T \overline{S}. \end{aligned} \tag{9.80}$$

Note that $C_1^T \overline{V}^{-1}(I - C_2 K) C_1$ is symmetric. (9.80) implies:

$$\begin{aligned} &\overline{S} + \overline{S}\left[A_{11} - A_{12} K C_1 - \overline{S}^{-1} C_1^T \overline{V}^{-1}(I - C_2 K) C_1\right] \\ &\quad + \left[A_{11} - A_{12} K C_1 - \overline{S}^{-1} C_1^T \overline{V}^{-1}(I - C_2 K) C_1\right)^T \overline{S} \leq 0, \end{aligned}$$

which by Lyapunov's direct method (Brogan 1991) implies that

$$A_{11} - A_{12} K C_1 - \overline{S}^{-1} C_1^T \overline{V}^{-1}(I - C_2 K) C_1$$

is stable. For time-invariant systems, this implies that the closed-loop eigenvalues lie in the open left-half plane.

What happens, however, when $i > 1$ and dim(Ker S) $>$ Rank B_i? The matrix B_{12} will no longer be square and the reduced-order Riccati equation will be stuck in the form of (9.76) which is not the same as what is needed in the proof for stability (9.77). It would seem that we cannot guarantee stability in the general case.

It turns out, however, that by augmenting the failure map in the original problem statement, we can always convert the reduced-order Riccati equation into the desired form (9.77). The necessary augmentation turns out to be:

$$\overline{F}_1 = \left[\begin{array}{cccc} B_i & B_{i-1} & \dots & B_1 \end{array} \right]$$

The new game problem for the limiting case is:

$$\begin{aligned}\min_{\hat{x}} \max_{\overline{\mu}_2} J^* &= \int_{t_0}^{t_1} \Big[\|x - \hat{x}\|^2_{C^T \hat{H}_1^T Q_1 \hat{H}_1 C} + (x - \hat{x})^T C^T \hat{H}_1^T Q_1 \hat{H}_1 C \overline{F}_1 \overline{\mu}_2 \\ &\quad + \|\overline{\mu}_2\|^2_{\overline{F}_1^T C^T \hat{H}_1^T Q_1 \hat{H}_1 C \overline{F}_1} - \|y - Cx\|^2_{\overline{V}^{-1}} - (y - Cx)^T \overline{V}^{-1} C \overline{F}_1 \overline{\mu}_2 \\ &\quad - \overline{\mu}_2^T \overline{F}_1^T C^T \overline{V}^{-1} (y - Cx) - \|\overline{\mu}_2\|^2_{\overline{F}_1^T C^T \overline{V}^{-1} C \overline{F}_1} \Big] \, dt\end{aligned}$$

subject to:

$$\dot{x} = Ax + \overline{F}_1 \overline{\mu}_2$$

where $\overline{\mu}_2$ is the augmented failure signal which has as many inputs as there are columns in $\overline{F}_1$. Note, that here we have gone back to the pre-transformed problem where the state is x, not α_i. We will show that this new problem leads to a Riccati equation which is equivalent to (9.44). In this equation, however, the reduced-order version is easily seen to reduce to the desired form (9.77). The equivalence of the two equations then implies that the same reduced form holds for both.

The augmented failure map, $\overline{F}_1$ is such that $C\overline{F}_1 \neq 0$, so the transformation process converges after one iteration. The solution to this game leads to a Goh Riccati equation:

$$\begin{aligned}-\dot{S} &= SA + A^T S + C^T (\hat{H}_1^T Q_1 \hat{H}_1 - \overline{V}^{-1}) C \\ &\quad + \left[S(A\overline{F}_1 - \dot{\overline{F}}_1) - C^T \overline{V}^{-1} C \overline{F}_1 \right] (\overline{F}_1^T C^T \overline{V}^{-1} C \overline{F}_1)^{-1} \\ &\quad \times \left[(A\overline{F}_1 - \dot{\overline{F}}_1)^T S - \overline{F}_1^T C^T \overline{V}^{-1} C \right] \qquad (9.81)\end{aligned}$$

with a boundary condition given by (9.41). The solution, S, to (9.81) is such that

$$\dim(\operatorname{Ker}\overline{S}) = \operatorname{Rank}\overline{F}_1.$$

Hence, after the transformation and de¯ning:

$$\begin{bmatrix} \overline{F}_{11} \\ \overline{F}_{12} \end{bmatrix} = \text{¡}\,\overline{F}_1,$$

the reduced-order Riccati equation:

$$\begin{aligned} -\overline{S} \;=\;& C_1^T\left(\hat{H}_1^T Q_1 \hat{H}_1 - \overline{V}^{-1}\right) C_1 + \overline{S}A_{11} + A_{11}^T\overline{S} \\ &+ \left(\overline{S}A_{12}\overline{F}_{12} - C_1^T\overline{V}^{-1}C_2\overline{F}_{12}\right)\left(\overline{F}_{12}^T C_2^T\overline{V}^{-1}C_2\overline{F}_{12}\right)^{-1}\left(\overline{S}A_{12}\overline{F}_{12} - C_1^T\overline{V}^{-1}C_2\overline{F}_{12}\right)^T. \end{aligned}$$

can be simpli¯ed to (9.77) because $\overline{F}_{12}$ is square and invertible. We know that $\overline{F}_{12}$ is square and invertible because the construction of $\overline{F}_1$ ensures that $\overline{F}_1$ has full column rank and that the size of Ker S, which determines the order reduction, is equal to this column rank.

Proposition 9.6. The Goh Riccati equation of the augmented system (9.81) is equivalent to the Goh Riccati equation of the original system (9.44).

Proof. It is immediate that

$$C\overline{F}_1 = C\begin{bmatrix} B_i & B_{i-1} & \dots & B_1 \end{bmatrix} = CB_i \tag{9.82}$$

If we examine the term $SA\overline{F}_i - \overline{F}_1$ in (9.81):

$$\begin{aligned} S(A\overline{F}_1 - \overline{F}_1) &= SA[B_i,\ B_{i-1},\ \dots,\ B_1] + S\left[B_i,\ B_{i-1},\ \dots,\ B_1\right] \\ &= \left[SAB_i - SB_i,\ SAB_{i-1} - SB_i,\ \dots,\ SAB_1 - SB_1\right] \\ &= \left[SAB_i - SB_i,\ SB_i,\ SB_{i-1},\ \dots,\ SB_2\right]. \end{aligned}$$

Because of Proposition 9.2, this simpli¯es to

$$S(A\overline{F}_1 - \overline{F}_1) = S(AB_i - B_i). \tag{9.83}$$

Given, (9.82) and (9.83), the Goh Riccati equation for the augmented system (9.81) reduces to (9.44). ∎

Remark 5. The proposed "augmentation" is simply a restatement of the problem.

Reduced-order filters for the time-varying case are much harder to come by since the transformation matrix, Γ, will now be a function of time. In this case, the only likely option left to the analyst is to use the results of (Oshman and Bar-Itzhack 1985) which give differential equations for the eigenvectors and eigenvalues of the solution to a time-varying Riccati equation. From here the reduced-order Riccati matrix, the transformed system equation and finally the reduced-order filter can be formed through a transformation matrix based upon the eigenvectors. Needless to say, the computation required here will be quite intensive. The state and measurement matrices will also have to be transformed at each time step and only then can the filter be formed and propagated. The point here is that it is possible to find a reduced filter for the time-varying case, though the effort may outweigh the benefits. Since the full-order filter is always available, this is not a serious problem.

The analyst has many options when designing a game theoretic filter. In the case of the full-order filter he has the freedom to choose the different weighting matrices and γ. For reduced-order filters, he can use either the solution to the Goh Riccati equation (9.44) or the solution of linear matrix inequality (9.33) with $\gamma = 0$ to find the needed transformation matrix and reduced-order filter gain. He also has the reduced-order Riccati equation (9.77). Moreover, he can mix the two approaches, for example, by using the LMI to find the transformation matrix and using the reduced-order Goh Riccati equation to find the gain. This flexibility is important, because the solution to the Goh equations may be ill-conditioned when several iterations of the Goh transformation are needed to generate the Riccati equation. The appearance of powers of A in the resulting equation may cause problems with the numerical solution.

9.6 Application to AVCS: An Engine Air Mass Sensor Fault Detection Filter

To demonstrate the effectiveness of the game theoretic filter, we will apply our results to an example derived from (Douglas et al. 1995). In that report, a fault detection and identification system consisting of a bank of Beard-Jones fault detection filters was designed a for a single automobile using the methodology of (Douglas and Speyer 1996). Since we are only trying to provide a design example, we will not attempt to repeat the entire FDI system construction of (Douglas et al. 1995), but will merely design a game theoretic filter for one of the subproblems given in (Douglas et al. 1995): the monitoring of the engine air mass sensor.

In (Douglas et al. 1995), the nonlinear dynamics of a single vehicle was linearized about at straight line path at the constant speed of $25\frac{\mathrm{m}}{\mathrm{sec}}$. The resulting linear dynamics were then further reduced via spectral separation and balanced realizations until a 2-input, 7-output, 7^{th}-order state-space model representing the longitudinal dynamics was found:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du + v. \end{aligned}$$

The measurements are:

$$y = \left\{ \begin{array}{c} y_m \\ y_\omega \\ y_{\ddot{x}} \\ y_{\ddot{z}} \\ y_q \\ y_{y_{fs}} \\ y_{y_{rs}} \end{array} \right\} \quad \begin{array}{l} \text{Engine Manifold Air Mass (kg)} \\ \text{Engine Speed } (\frac{\mathrm{rad}}{\mathrm{sec}}) \\ \text{longitudinal acceleration } (\frac{\mathrm{m}}{\mathrm{sec}^2}) \\ \text{heave acceleration } (\frac{\mathrm{m}}{\mathrm{sec}^2}). \\ \text{Pitch Rate } (\frac{\mathrm{rad}}{\mathrm{sec}}). \\ \text{Forward Symmetric Wheel Speed } (\frac{\mathrm{rad}}{\mathrm{sec}}). \\ \text{Rear Symmetric Wheel Speed } (\frac{\mathrm{rad}}{\mathrm{sec}}). \end{array} \tag{9.84}$$

The inputs are:

$$u = \left\{ \begin{array}{c} \alpha \\ \beta \end{array} \right\} \quad \begin{array}{l} \text{Throttle Angle (deg)} \\ \text{Brake Torque (N-m)} \end{array} \tag{9.85}$$

Because of the balanced realization, the states have no physical meaning.

In all, there are 9 possible actuator/sensor faults. As we discussed earlier, the sensor faults will require detection spaces which are at least 2^{nd}-order. Actuator faults typically need no more than a 1^{st}-order detection space, but because of the direct feedthrough matrix D, the actuator faults in this example will require 3^{rd}-order detection spaces. See (Douglas et al. 1995) for details. Given that we have only 7 states, we will not be able to monitor all of the sensor and actuator faults with a single filter. In (Douglas et al. 1995), the 9 failures were divied up among 4 fault detection ¯lters with some of the failures included in more than one ¯lter for dynamical reasons. To keep our example simple, we will apply the game theoretic ¯lter to only one of the failure sets, which is designated \Filter 1" in (Douglas et al. 1995). In that ¯lter, the following three failures were grouped together:

F_{y_m} : Air Mass Sensor Failure

F_{y_ω} : Engine Speed Sensor Failure

$F_{y_{\ddot{x}}}$: Forward Acceleration Sensor Failure

In this example we will attempt to detect the air mass sensor failure, μ_{y_m}, given the possible presence of an engine speed sensor failure, μ_{y_ω}, and forward acceleration sensor failure, $\mu_{y_{\ddot{x}}}$. For comparison, the ¯lter designed in (Douglas et al. 1995) was able to detect and identify each of the three faults. As we noted before, a limitation of the game theoretic ¯lter is that, in its present form, it can only look for one fault per ¯lter and in this example we see this limitation brought to the forefront. Finally, we should also note that the ¯lter we design here will detect μ_{y_m} in the presence of any other failure that enters the system in the same way as μ_{y_ω} and $\mu_{y_{\ddot{x}}}$ or in the presence of any failure whose reachable subspace lies in the sum of the reachable subspaces of F_{y_ω} and $F_{y_{\ddot{x}}}$.

The failure model for this example is:

$$\dot{x} = Ax + F_{y_\omega}\mu_{y_\omega} + F_{y_{\ddot{x}}}\mu_{y_{\ddot{x}}} = Ax + \hat{F}_{y_m}\hat{\mu}_{y_m} \tag{9.86}$$

$$y = Cx + v, \tag{9.87}$$

where the system matrices are:

$$A = \begin{bmatrix} -0.0521 & -0.2213 & 0.2681 & -0.0121 & 0.0136 & 0.0084 & -0.0078 \\ -0.3007 & -8.0277 & -19.0734 & -1.1013 & 0.0795 & 0.2471 & 0.0378 \\ -0.3263 & -19.7571 & -51.0638 & -3.2675 & -4.8766 & -2.4258 & 0.0040 \\ 0.0454 & 2.4036 & 15.7922 & -2.1857 & 6.4655 & -0.2062 & 0.0495 \\ 0.0219 & 1.1136 & 8.6428 & -7.1817 & -0.6526 & -0.2171 & 0.9316 \\ 0.0116 & 0.5928 & 3.8335 & -1.0926 & -0.6513 & -0.9851 & 5.9628 \\ 0.0154 & 0.7868 & 4.8494 & -1.4900 & -1.0329 & -6.5688 & -2.5996 \end{bmatrix}$$

$$C = \begin{bmatrix} 0.0075 & 0.4605 & 0.3710 & 0.1023 & 0.0513 & 0.0340 & -0.0137 \\ 0.7318 & 2.7938 & -2.8640 & 0.1680 & -0.0415 & -0.0491 & -0.0029 \\ 0.0028 & 0.1711 & -0.2654 & 0.0765 & -0.0161 & 0.0093 & -0.0008 \\ 0.0000 & -0.0007 & -0.0005 & -0.0216 & -0.0496 & -0.0438 & 0.0697 \\ -0.0000 & -0.0024 & 0.0050 & 0.0111 & 0.0205 & -0.0027 & 0.0009 \\ 0.4214 & -0.1440 & 0.0371 & 0.2203 & -0.1764 & -0.0129 & 0.1051 \\ 0.4211 & 0.1318 & -0.4410 & -0.2741 & -0.0304 & -0.0734 & 0.0585 \end{bmatrix}$$

For simplicity, the inputs u will be disregarded.

What remains is to calculate F_{y_ω} and $F_{y_{\ddot{x}}}$. Following the the modeling techniques described in Section 9.1, we begin by augmenting the measurement equation to re°ect the presence of the engine speed and accelerometer sensor failures:

$$\underset{\sim}{x} = Ax \tag{9.88}$$

$$y = Cx + E_{y_\omega} m_{y_\omega} + E_{y_{\ddot{x}}} m_{y_{\ddot{x}}} + v. \tag{9.89}$$

where

$$E_{y_\omega} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T$$
$$E_{y_{\ddot{x}}} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}^T$$

We then calculate f_{y_ω} as the solution of $E_{y_\omega} = Cf_{y_\omega}$ and $f_{y_{\ddot{x}}}$ as the solution to $E_{y_{\ddot{x}}} = Cf_{y_{\ddot{x}}}$. The second column of the failure map is then obtained by multiplying $f_{y_{\ddot{x}}}$ and $f_{y_{\ddot{x}}}$ by the state matrix A. We then have the following failure maps:

$$F_{y_\omega} = \begin{bmatrix} f_{y_\omega} & Af_{y_\omega} \end{bmatrix} = \begin{bmatrix} 0.2107 & -0.0681 \\ 0.2986 & -1.1171 \\ 0.3791 & 14.0532 \\ 1.7301 & -9.9008 \\ -2.3516 & -13.4314 \\ -13.8538 & -43.7274 \\ -9.8358 & 118.5002 \end{bmatrix}$$

and

$$F_{y_{\ddot{x}}} = \begin{bmatrix} f_{y_{\ddot{x}}} & Af_{y_{\ddot{x}}} \end{bmatrix} = \begin{bmatrix} 0.0873 & 0.0209 \\ 0.9262 & 7.7252 \\ 0.2544 & -99.5538 \\ -3.0910 & 35.2772 \\ 4.0831 & 33.4690 \\ 24.1122 & 80.5043 \\ 17.1083 & -200.5111 \end{bmatrix}$$

For the purposes of the filter design we combine the two failure maps into a single complementary failure map:

$$\hat{F}_{y_m} = \begin{bmatrix} F_{y_\omega} & F_{y_{\ddot{x}}} \end{bmatrix}$$

Since $C\hat{F}_{y_m}$ is full rank we do not need to go into a Goh iteration sequence to form the projector $\hat{H}_1$. Thus, this projector is simply:

$$\begin{aligned} \hat{H}_1 &= I - (C\hat{F}_{y_m})[(C\hat{F}_{y_m})^T(C\hat{F}_{y_m})]^{-1}(C\hat{F}_{y_m})^T \\ &= \begin{bmatrix} 0.9986 & -0.0000 & 0.0000 & 0.0098 & -0.0008 & 0.0165 & -0.0317 \\ -0.0000 & 0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 \\ 0.0000 & 0.0000 & 0.0000 & -0.0000 & -0.0000 & 0.0000 & -0.0000 \\ 0.0098 & -0.0000 & -0.0000 & 0.6340 & 0.0062 & -0.4785 & -0.0540 \\ -0.0008 & 0.0000 & -0.0000 & 0.0062 & 0.9995 & 0.0102 & -0.0179 \\ 0.0165 & -0.0000 & -0.0000 & -0.4785 & 0.0102 & 0.3620 & 0.0397 \\ -0.0317 & -0.0000 & -0.0000 & -0.0540 & -0.0179 & 0.0397 & 0.0058 \end{bmatrix} \end{aligned} \tag{9.90}$$

9.6.1 Full-Order Filter Design

Equation 9.15, the Riccati equation in terms of ¦, was used for this example. To bring sensor noise weighting, V $(= \nu I)$, to zero with the disturbance bound, it is assumed that ν is some multiple of γ. By trial and error, it was found that:

$$\nu = 1 \times 10^{-8}, \quad \frac{\nu}{\gamma} = 0.8, \quad Q_1 = R_1 = M_2 = I$$

gave the results seen in Figure 9.2. For the parameters above, the solution of (9.15) is:

$$\Pi = \begin{bmatrix} 0.0108 & -0.0001 & 0.0009 & 0.0043 & -0.0035 & 0.0011 & 0.0003 \\ -0.0001 & 0.0044 & -0.0003 & -0.0033 & -0.0034 & 0.0005 & -0.0004 \\ 0.0009 & -0.0003 & 0.0014 & 0.0020 & 0.0011 & 0.0000 & 0.0001 \\ 0.0043 & -0.0033 & 0.0020 & 0.0059 & 0.0025 & 0.0000 & 0.0005 \\ -0.0035 & -0.0034 & 0.0011 & 0.0025 & 0.0051 & -0.0009 & 0.0003 \\ 0.0011 & 0.0005 & 0.0000 & 0.0000 & -0.0009 & 0.0002 & 0.0000 \\ 0.0003 & -0.0004 & 0.0001 & 0.0005 & 0.0003 & 0.0000 & 0.0000 \end{bmatrix} \tag{9.91}$$

resulting in a gain:

$$L = 10^6 \times \begin{bmatrix} -0.0000 & 0.0037 & -0.0344 & -0.0002 & 0.0000 & -0.0003 & 0.0007 \\ 0.0003 & 0.0218 & -0.0470 & 0.2636 & -0.0004 & 0.3208 & 0.2517 \\ -0.0003 & -0.1411 & -0.1145 & 0.3172 & -0.0007 & 0.3878 & 0.2879 \\ -0.0006 & 0.1147 & -0.1078 & -0.3230 & 0.0006 & -0.3935 & -0.3032 \\ 0.0015 & 0.1110 & 0.3183 & -0.5540 & 0.0012 & -0.6768 & -0.5083 \\ 0.0066 & 0.2818 & 1.7919 & -2.4235 & 0.0050 & -2.9591 & -2.2383 \\ 0.0035 & -1.2066 & 0.1269 & 6.9371 & -0.0120 & 8.4546 & 6.5149 \end{bmatrix} \tag{9.92}$$

When applied to the 7^{th}-order car model, the result is a stable filter with closed-loop poles at: -2, 128, 332.1, -458867.7, -11, 157.0, -856.2, -259.7, -9.1 and -0.31. As Figure 9.2 shows, the filter achieves roughly 80 db. of separation in transmission between the target fault (an engine air mass sensor failure) and the larger of the two nuisance faults. As a comparison, Figure 9.3 plots the results of the Beard-Jones filter design from (Douglas et al. 1995) for the same set of faults. The closed-loop poles for this filter were selected to be: -3, -4, -5, -6, -7, -8 and -9.

A comparison of the two filters shows that they both do an adequate job of separating the target fault and the nuisance faults. The Beard-Jones filter has less separation, but it also amplifies the target fault signal. For the residual processing stage of fault detection and identification, this might prove to be useful side effect. Moreover, the game theoretic filter achieves its impressive transmission separation at the cost of extremely high gains. This is due the aggressively low value of γ chosen for this design example. Higher values of γ can be chosen which achieve less separation but also result in smaller gains. We will also show, in the next section, how to design a reduced-order filter which achieves our fault detection goals and which also possesses very reasonable gains.

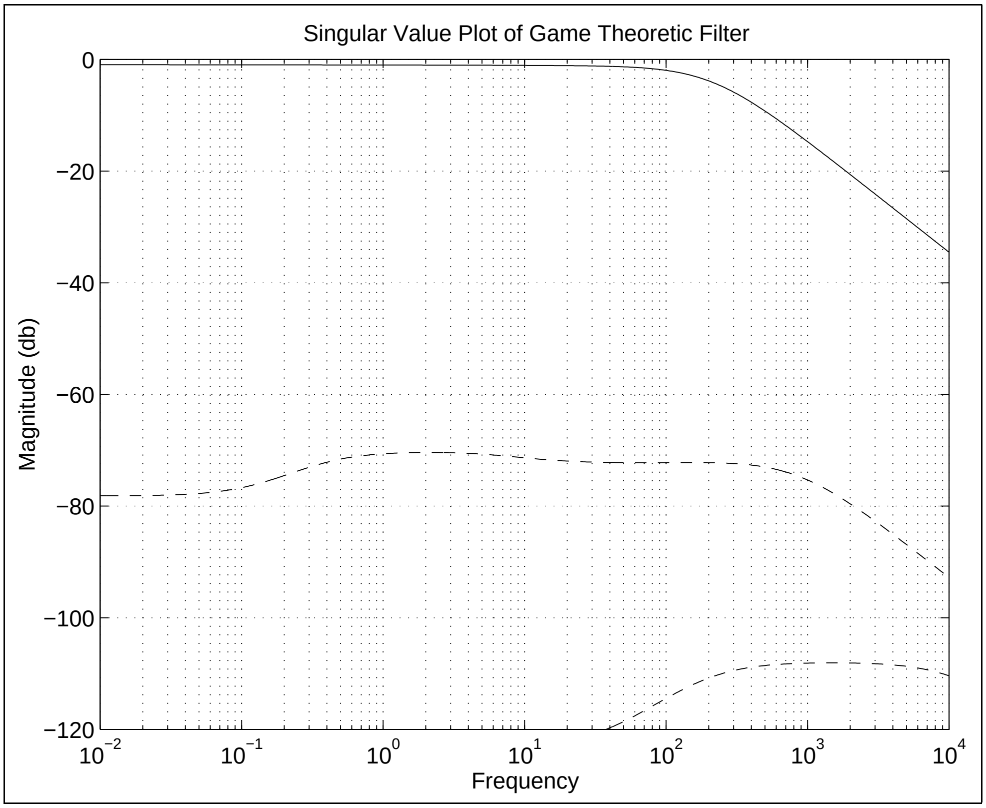

Figure 9.2: Game Theoretic Filter Singular Value Plot of Air Mass Fault Signal versus Singular Values of Engine Speed and Accelerometer Faults (solid line - output due to μ_{y_m}; dashed lines - outputs due to μ_{y_ω} and $\mu_{y_{\ddot{x}}}$).

Another factor to consider is the issue of sensor noise transmission. As (Lee 1994) points out, Beard-Jones ¯lters can have fairly poor noise properties. This is demonstrated by Figure 9.4 which shows that the largest singular value for noise transmission is consistently larger than the singular value for the target fault transmission. On the other hand, Figure 9.5 shows that the game theoretic ¯lter achieves separation between sensor noise and target fault transmission at frequencies above $10\frac{\text{rad}}{\text{sec}}$ for all of the the noise channels except for the one which comes into the ¯lter dynamics in the same way as the target fault itself. This noise signal is indistinguishable from the target fault and its singular value plot is identical to the target faults over all frequencies. Separating the fault signal from measurement noise will then have to come in the residual evaluation stage. Typically, this

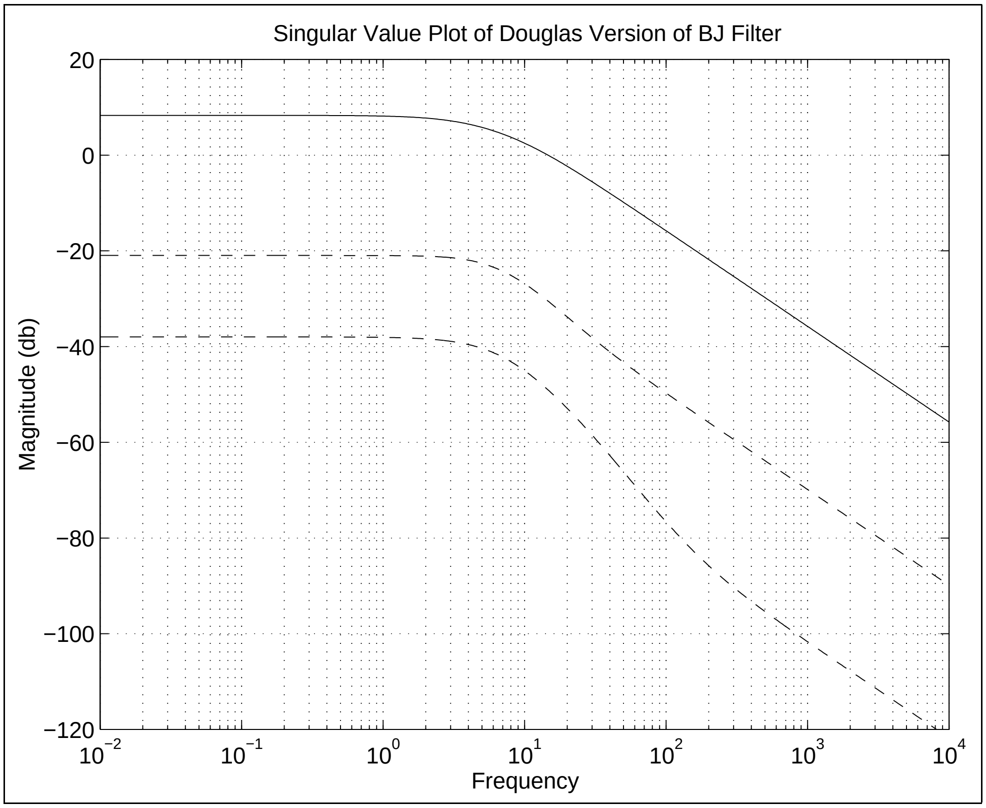

Figure 9.3: Beard-Jones Filter Singular Value Plot of Air Mass Fault Signal versus Singular Values of Engine Speed and Accelerometer Faults (solid line - output due to μ_{y_m}; dashed lines - outputs due to μ_{y_ω} and $\mu_{y_{\ddot{x}}}$).

involves making assumptions about the failure signal and about the statistics of the sensor noise. See for example (Douglas et al. 1995) and (Emami-Naeini et al. 1988).

9.6.2 Reduced-Order Filter Design via the Goh Riccati Equations

We now repeat the example, but now we will design a lower-order ¯lter using the Goh Riccati equations. The ¯rst step is to derive the transformation matrix, ¡. Since the transformation is determined via the null space of the full-order Riccati matrix, the design process begins by ¯nding the solution to the full-order Goh Riccati equation (9.44). Because $C\hat{F}_1$ is full-rank, we are spared the step of going through a Goh iteration to set up the correct Goh Riccati equation.

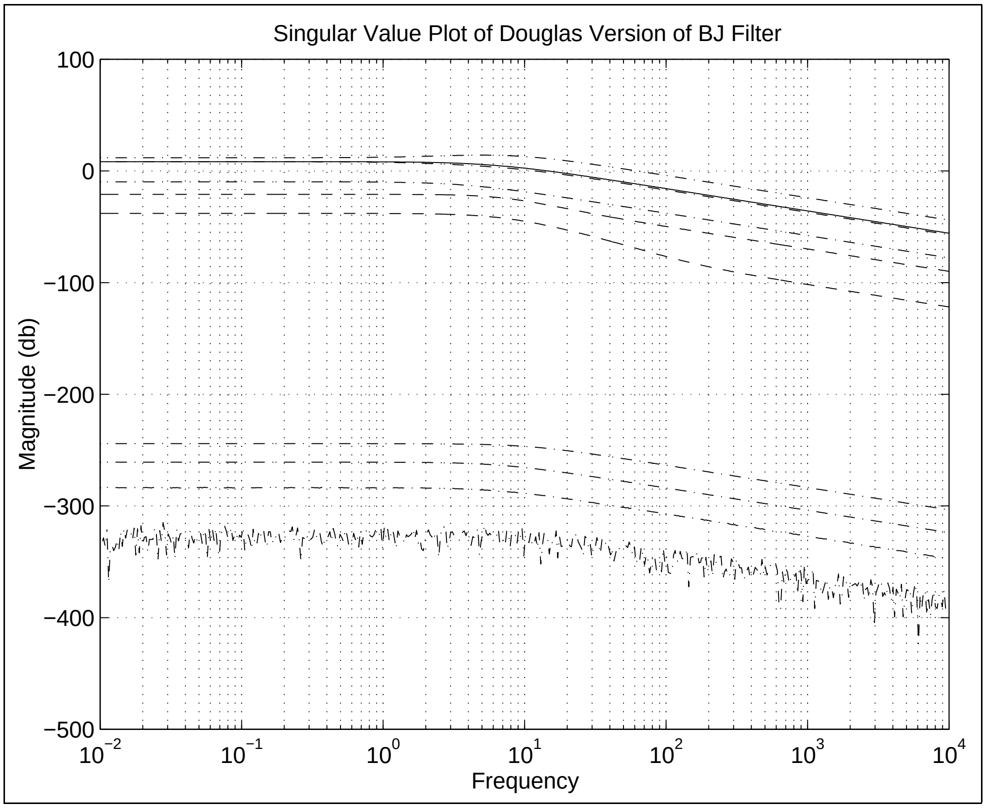

Figure 9.4: Beard-Jones Filter Singular Value Plot of Air Mass Fault Signal versus Singular Values of Engine Speed and Accelerometer Faults (solid line - output due to μ_{y_m}; dashed lines - nuisance faults, dot-dashed lines - noise).

Using the same weightings as in the full-order design, we ¯nd that the solution to the Goh Riccati equation (9.44) is:

$$S = \begin{bmatrix} 21.8547 & -0.2217 & -0.0277 & -0.0358 & 0.0271 & 0.0141 & 0.0114 \\ 63.2776 & -0.8201 & -0.0969 & -0.0807 & 0.1369 & 0.0093 & 0.0352 \\ -21.9515 & 0.2891 & 0.0331 & 0.0270 & -0.0496 & -0.0023 & -0.0122 \\ -61.5141 & 0.8211 & 0.0953 & 0.0749 & -0.1416 & -0.0047 & -0.0345 \\ -76.2310 & 0.9668 & 0.1138 & 0.0996 & -0.1586 & -0.0148 & -0.0421 \\ 10.9799 & -0.1357 & -0.0162 & -0.0148 & 0.0216 & 0.0028 & 0.0060 \\ -6.5160 & 0.0860 & 0.0101 & 0.0081 & -0.0146 & -0.0007 & -0.0036 \end{bmatrix}$$

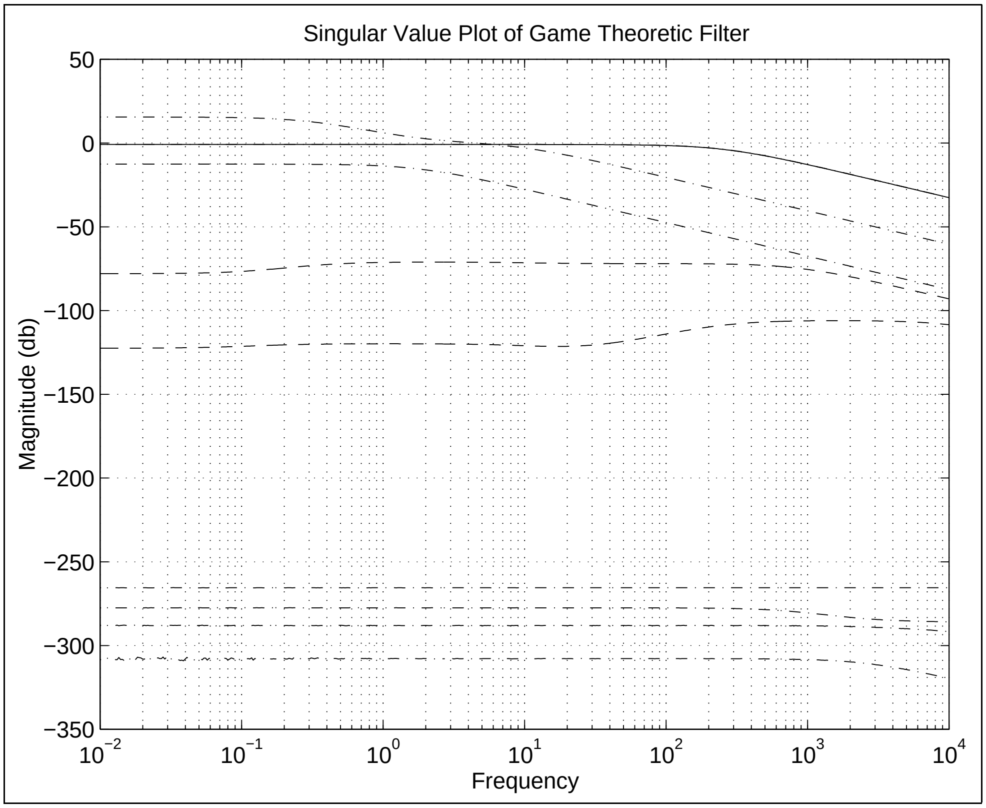

Figure 9.5: Game Theoretic Filter Singular Value Plot of Air Mass Fault Signal versus Nuisance Faults and Noise (solid line - output due to μ_{y_m}; dashed lines - nuisance faults, dot-dashed lines - noise).

Using the QR decomposition we ¯nd obtain a transformation matrix:

$$¡^T = \begin{bmatrix} -0.1801 & 0.8639 & 0.0800 & -0.0329 & -0.3166 & 0.3369 & -0.0035 \\ -0.5215 & -0.0913 & -0.6879 & -0.4917 & -0.0018 & 0.0687 & -0.0056 \\ 0.1809 & 0.0982 & -0.6580 & 0.7204 & -0.0020 & 0.0693 & -0.0312 \\ 0.5070 & 0.4348 & -0.2304 & -0.3580 & 0.4084 & -0.4414 & -0.1051 \\ 0.6283 & -0.1984 & -0.1801 & -0.3258 & -0.5190 & 0.3693 & 0.1467 \\ -0.0905 & 0.0797 & -0.0416 & 0.0575 & -0.3232 & -0.5328 & 0.7695 \\ 0.0537 & 0.0312 & 0.0166 & -0.0237 & 0.5992 & 0.5118 & 0.6118 \end{bmatrix} \quad (9.93)$$

Using this transformation, we reduce our state-space to a third-order system, that is, we ¯nd the matrices A_{11}, C_1 etc. From here we employ the reduced-order system matrices in the reduced order Goh Riccati equation, (9.81). The solution to (9.81) using (9.93) is:

$$\overline{S} = \begin{bmatrix} -0.0417 & 0.0216 & -0.3085 \\ 0.0216 & -0.0073 & 0.1923 \\ -0.3085 & 0.1923 & -2.1336 \end{bmatrix} \tag{9.94}$$

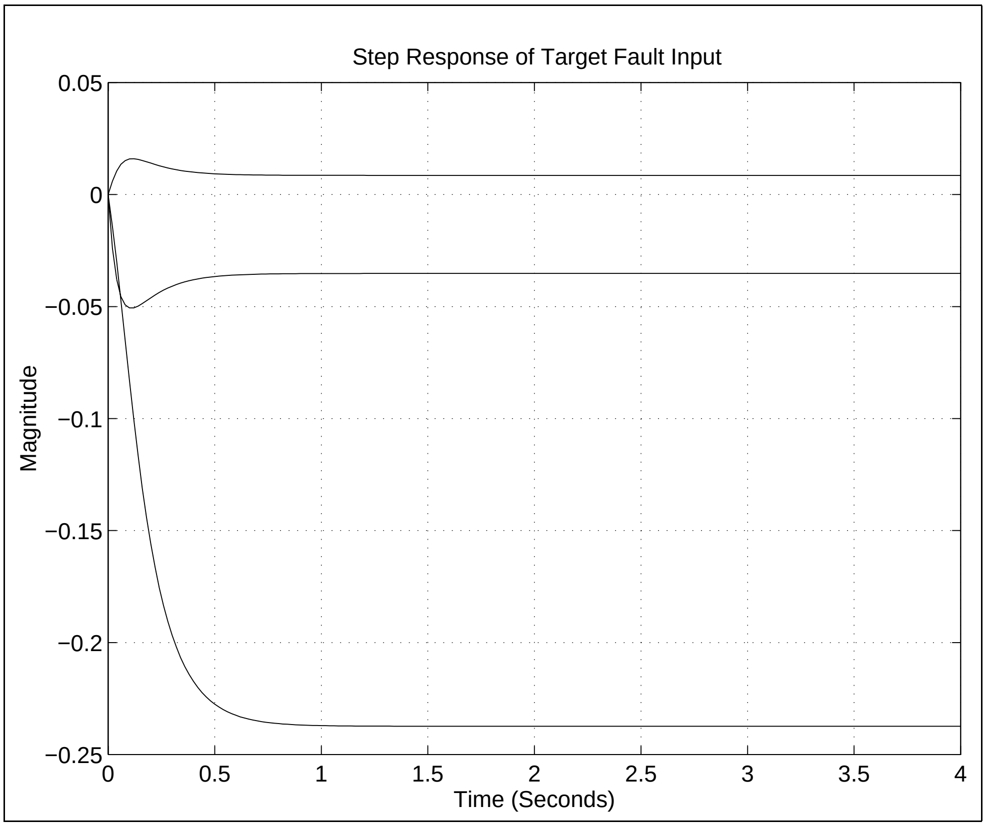

Figure 9.6: Reduced-Order Goh Filter Residual due to step in μ_{A_z} (fault to be detected).

with a corresponding gain:

$$L = \begin{bmatrix} -4.8971 & 0.0001 & 0.0000 & -213.6617 & 39.9079 & 154.1616 & 30.1926 \\ -1.4088 & 0.0000 & 0.0001 & -98.3020 & 24.3178 & 73.0444 & 11.9234 \\ 0.2842 & -0.0001 & 0.0002 & 21.6742 & -3.5535 & -16.2795 & -1.9279 \end{bmatrix} \tag{9.95}$$

The closed-loop eigenvalues are: -7.0976, -23.3114 and -35.2309. To demonstrate the e®ectiveness of the reduced-order ¯lter a linear simulation of the system was run for two cases: one with a engine air mass sensor fault input (modeled as a step) the other with a engine speed sensor fault input (also a step). Figures 9.7 and 9.6 show that the reduced-order

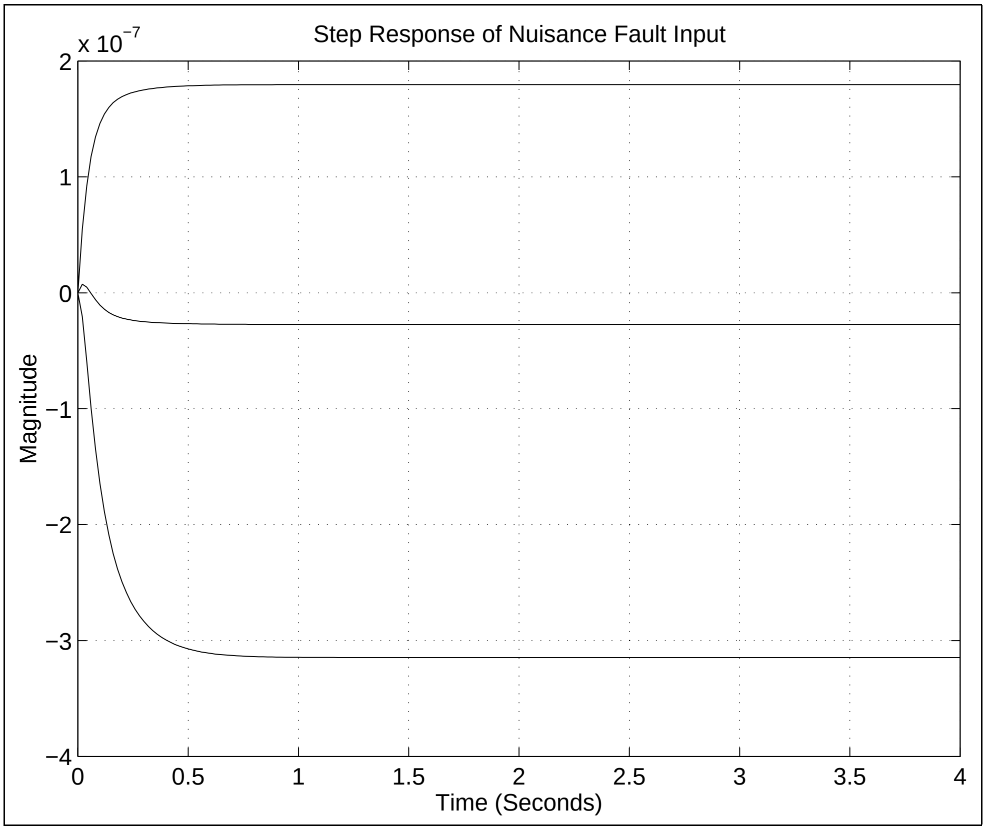

Figure 9.7: Reduced-Order Goh Filter Residual due to step in μ_{wg} (nuisance fault).

¯lter responds to the air mass sensor fault input and is relatively insensitive to the engine speed sensor fault.

9.7 Discussion

By solving the fault detection problem via disturbance attenuation, we obtain a game theoretic ¯lter that bounds the transmission of disturbances and nuisance faults. By going to the limit of this solution, we get a fault detection ¯lter which in the time-invariant case is equivalent to the Beard-Jones fault detection ¯lter. That is, the presence of the nuisance faults is restricted to an invariant subspace that can be made unobservable through a projection. This unobservable subspace can be factored out of total space to get a lower-order system which is unin°uenced by the nuisance faults. The same factoring process can then be applied to the game filter to get a reduced-order fault detector for the newly reduced state-space. Extensions of this latter result exist for the time-varying case, though the computation involved may be intensive.

The game theoretic approach to fault detection filter design is more flexible than current design methods. The designer can choose the degree to which the game filter possesses the structure of the Beard-Jones filter. This allows him to make tradeoffs between nuisance fault blocking and sensor noise rejection. The linear quadratic game used to solve the disturbance attenuation problem admits time-varying systems and can be used to incorporate parameter uncertainty into the filter design. Recent extensions of robust control such as designs which constrain pole-placement and designs with multiple objectives, for example, the so-called mixed $\mathcal{H}_2/\mathcal{H}_\infty$ problems, suggest that the same can be done here. The latter is of particular interest since it appears to be a logical way to detect and identify multiple faults with a single game theoretic filter.

Finally, we have shown that the limiting form of the game filter is a singular filter. Since any disturbance attenuation problem can be solved in the same manner as this one, it is likely that this result applies to all such problems. That is, the limiting form of a disturbance attenuation problem is a singular optimization problem. This makes applicable a wealth of results from singular control and it provides a new way to understand $\mathcal{H}_\infty$ problems by looking at them as "almost" singular optimal control problems.

Chapter 10
Conclusions

Analytic redundancy is a viable approach to vehicle health monitoring. The fault detection filters developed here perform well in a high-fidelity nonlinear simulation. The filter residuals quickly and clearly respond to the introduction of faults even in the presence of significant vehicle nonlinearities from both longitudinal and lateral modes. Two candidate residual processing systems both effectively automate fault announcement. A Bayesian neural network examines the fault detection filter residual for activity characteristic of a static pattern associated with a fault. A fault and an associated probability of occurrence are announced by the neural network soon after the fault is introduced in the vehicle nonlinear simulation. A modified Shiryayev sequential probability ratio test extended to include multiple hypotheses examines the filter residuals and tests for a fault hypothesis change. Both systems respond well to hard and soft failures in the presence of sensor noise, dynamic disturbances and vehicle nonlinearities.

By directing development of the project components in parallel and seeing significant progress in all areas, we are able to identify several important areas for future work: model re¯nement, robust fault detection ¯lter design, time-varying fault detection ¯lter design, system integration and platoon health monitoring.

Model Re¯nement: This year, a re¯ned nonlinear vehicle model and simulation was completed. This model allows for arbitrarily changing road gradients for each of the four wheels. Work will now continue by developing uncertainty models associated with process disturbances such as rough and hilly roads, winds, system parameter uncertainty and unmodeled dynamics. Through a good working relation with the Berkeley PATH researchers, model ¯delity will be improved further using empirically derived data. Fidelity of the modeled nonlinearities and uncertainties is very important for a realistic assessment of any health monitoring system performance.

Robust Fault Detection Filter Design: Development of robust fault detection ¯lters will continue with two directions of investigation. First, the system will be examined for the possibility of treating nonlinearities and disturbances as pseudo-fault directions. This approach e®ectively decouples the nonlinearity or disturbance from fault identifying residuals. Second, parameter uncertainty in the linearized vehicle dynamics is modeled as an input-output decomposition. This allows model uncertainty to be treated as a disturbance.

Time-Varying Detection Filter Design: Automated vehicles engaged in merge and split maneuvers may follow a trajectory that induces time-varying vehicle dynamics. The notion of a fault detection ¯lter for time-varying systems was introduced in the game theoretic fault detection ¯lter development described in this report. It is expected that these notions will be extended to invariant subspace ¯lter structures.

System Integration: Having developed preliminary fault detection and isolation system designs for one longitudinal and one lateral mode, work will proceed by considering several other design points and then combining all the designs into one integrated package.

Platoon Health Monitoring: Work will begin towards extending the health monitoring system for one vehicle to include the presence of multiple vehicles in a controlled platoon con¯guration. Sensors required for control such as distance measurements will be included in the fault set. Transmission of vehicle sensor outputs will be transmitted to all vehicles. Feasibility and performance of an expanded health monitoring system will be evaluated in an extended nonlinear simulation.

Appendix A
Fault Detection Filter Background

A LINEAR TIME-INVARIANT SYSTEM with q failure modes and no disturbances or sensor noise can be modeled (Beard 1971), (White and Speyer 1987), (Massoumnia 1986) by

$$\dot{x} = Ax + Bu + \sum_{i=1}^{q} F_i m_i \tag{a.1a}$$

$$y = Cx. \tag{a.1b}$$

All system variables belong to real vector spaces $x \in \mathcal{X}$, $u \in \mathcal{U}$, $y \in \mathcal{Y}$ and $m_i \in \mathcal{M}_i$ with $n = \dim \mathcal{X}$, $p = \dim \mathcal{U}$, $m = \dim \mathcal{Y}$ and $q_i = \dim \mathcal{M}_i$. The input $u \in \mathcal{U}$ is known as is the output $y \in \mathcal{Y}$. The failure modes $m_i \in \mathcal{M}_i$ are vectors that are unknown and arbitrary functions of time and are zero when there is no failure. The failure signatures $F_i : \mathcal{M}_i \mapsto \mathcal{F}_i \subseteq \mathcal{X}$ are maps that are known, ¯xed and unique. A failure mode m_i models the time-varying amplitude of a failure while a failure signature F_i models the directional characteristics of a failure. Assume the F_i are monic so that $m_i \neq 0$ implies $F_i m_i \neq 0$. Actuator and plant faults are modeled with F_i as the appropriate direction from A or B.

For example, a stuck actuator is modeled with F_i as the column of A associated with the actuator dynamics and with $m_i(t) = -u_i(t) + u_{ic}$ where u_{ic} is some constant.

Sensor faults are most naturally modeled as an additive term in the measurement equation as follows where E_i is a column vector of zeros except for a one in the i^{th} position and where μ_i is an arbitrary time-varying real scalar.

$$y = Cx + E_i\mu_i \tag{a.2}$$

It can be shown that the E_i sensor fault form of (a.2) may be converted to an equivalent F_i form (a.1) with no need for appended dynamics (Beard 1971), (White and Speyer 1987), (Douglas 1993). This is demonstrated shortly.

A.1 The Detection Filter Problem

Consider a full-order observer of the form

$$\dot{\hat{x}} = (A + LC)\hat{x} + Bu - Ly \tag{a.3a}$$

$$z = C\hat{x} - y. \tag{a.3b}$$

The state estimation error $e = \hat{x} - x$ dynamics are

$$\dot{e} = (A + LC)e - \sum_{i=1}^{q} F_i m_i \tag{a.4}$$

If (C, A) is observable and L is chosen so that $A + LC$ is stable, then in steady-state and in the absence of disturbances and modeling errors, the residual r is nonzero only if a failure mode m_i is nonzero and is almost always nonzero whenever m_i is nonzero. It follows that any stable observer can detect the occurrence of a fault. Simply monitor the residual z and when it is nonzero a fault has occurred. A more difficult task is to determine which fault has occurred and that is what a fault detection filter is designed to do.

A fault detection filter is an observer with the property that when an unknown input or fault is nonzero, $m_i(t) \neq 0$, the error $e(t)$ remains in a (C, A)-invariant subspace $\mathcal{W}_i$ which contains the reachable subspace of $(A + LC, F_i)$. Thus, the residual remains in the output subspace $C\mathcal{W}_i$. Furthermore, the output subspaces $C\mathcal{W}_1, \ldots, C\mathcal{W}_q$ are independent so that $z \in \sum_{i=1}^{q} C\mathcal{W}_i$ has a unique representation $z = z_1 + \cdots + z_q$ with $z_i \in C\mathcal{W}_i$. The fault is identified by projecting z onto each of the output subspaces $C\mathcal{W}_i$. The following statement of the detection filter problem, sometimes called the Beard-Jones detection filter problem, is essentially the same as that found in (Beard 1971) and (White and Speyer 1987) but is stated in the geometric language of (Massoumnia 1986).

Definition A.1 (Detection Filter Problem). Given the system (a.1), with state-space $\mathcal{X}$ and measurement-space $\mathcal{Y}$, the detection filter problem is to find a set of subspaces $\mathcal{W}_i \subseteq \mathcal{X}$, $i = 1, \ldots, q$ such that for some map $L : \mathcal{Y} \mapsto \mathcal{X}$ the following conditions are met:

$$(A + LC)\mathcal{W}_i \subseteq \mathcal{W}_i \qquad \text{Subspace invariance.}$$

$$\mathcal{F}_i \subseteq \mathcal{W}_i \qquad \text{Failure inclusion.}$$

$$C\mathcal{W}_i \cap (\sum_{j \neq i} C\mathcal{W}_j) = 0 \qquad \text{Output separability.}$$

It can be shown (Massoumnia 1986), (White and Speyer 1987) that the last condition, output separability, implies that the subspaces $\mathcal{W}_1, \ldots, \mathcal{W}_q$ are independent when (C, A) is observable

A.2 Sensor Fault Models

It is now shown how the E_i sensor fault form of (a.2) is converted to an equivalent F_i form with no need for appended dynamics. While this is also shown in (Beard 1971), (White and Speyer 1987) and (Douglas 1993), the following original demonstration is more easily extended to time varying systems. Let F_i be any map that satisfies

$$CF_i = E_i$$

and define a new state estimation error $\tilde{e}$ as

$$\tilde{e} = e - F_i \mu_i$$

This is a Goh transformation on the error space (Jacobson 1971). The residual is then.

$$r = C\bar{e}$$

Using (a.4), the dynamics of $\bar{e}$ are

$$\dot{\bar{e}} = (A + LC)\bar{e} + AF_i\mu_i - F_i\dot{\mu}_i \tag{a.5}$$

and a sensor fault E_i in (a.2) is equivalent to a two-dimensional fault F_i

$$\dot{x} = Ax + Bu + F_i m_i \qquad \text{with } F_i = \left[F_i^1, F_i^2\right]$$

where the directions F_i^1 and F_i^2 are given by

$$E_i = CF_i^1 \tag{a.6a}$$

$$F_i^2 = AF_i^1 \tag{a.6b}$$

An interpretation of the effect of a sensor fault on observer error dynamics follows from (a.5) where F_i^1 is the sensor fault rate $\dot{\mu}_i$ direction and F_i^2 is the sensor fault magnitude μ_i direction. This interpretation suggests a possible simplification when information about the spectral content of the sensor fault is available. If it is known that a sensor fault has persistent and significant high frequency components, such as in the case of a noisy sensor, the fault direction could be approximated by the F_i^1 direction alone. Or, if it is known that a sensor fault has only low frequency components, such as in the case of a bias, the fault direction could be approximated by the F_i^2 direction alone. For example, if a sensor were to develop a bias, a transient would be likely to appear in all fault directions but, in steady-state, only the residual associated with the faulty sensor should be nonzero.

In the case where the dynamics (a.1) are time varying, the error dynamics (a.5) become

$$\dot{\bar{e}} = (A + LC)\bar{e} + (AF_i - \dot{F}_i)\mu_i - F_i\dot{\mu}_i$$

so that once again, a sensor fault E_i in (a.2) is equivalent to a two-dimensional fault F_i

$$\dot{x} = Ax + Bu + F_i m_i \qquad \text{with } F_i = \left[F_i^1, F_i^2\right]$$

but where the directions F_i^1 and F_i^2 are given by

$$
\begin{aligned}
E_i &= CF_i^1 \\
F_i^2 &= AF_i^1 - E_i^1
\end{aligned}
$$

A.3 Solving The Detection Filter Problem

It should be pointed out that for any subspace $\mathcal{F}_i \subseteq \mathcal{X}$ there is a minimal (C, A)-invariant subspace $\mathcal{F}_i \subseteq \mathcal{W}_i^* \subseteq \mathcal{X}$. A recursive algorithm, the (C, A)-invariant subspace algorithm, for computing a minimal invariant subspace is suggested by (Wonham 1985) and restated in the following theorem.

Theorem A.1 (CAISA). Let $\mathcal{W}(\mathcal{F})$ be a family of (C, A)-invariant subspaces where $\mathcal{F} \subseteq \mathcal{W} \in \mathcal{W}(\mathcal{F})$. Then, there exists a minimal (C, A)-invariant subspace $\mathcal{W}^* \in \mathcal{W}(\mathcal{F})$ where for any $\mathcal{W} \in \mathcal{W}(\mathcal{F})$, $\mathcal{W}^* \subseteq \mathcal{W}$. Furthermore, $\mathcal{W}^* = \lim \mathcal{W}^k$ where $\mathcal{W}^k$ is given by the recursive algorithm

$$
\begin{aligned}
\mathcal{W}^0 &= \emptyset \\
\mathcal{W}^{k+1} &= \mathcal{F} + A\left(\mathcal{W}^k \cap \operatorname{Ker} C\right)
\end{aligned}
$$

Proof. The proof given in (Wonham 1985) follows from the result of (Willems 1982) that the set $\mathcal{W}(\mathcal{F})$ is closed under subspace intersection.

Note that the algorithm given in Theorem A.1 implies that for $\dim \mathcal{F}_i = 1$, the minimal (C, A)-invariant subspace $\mathcal{W}_i^*$ is spanned by $\{F_i, AF_i, \ldots, A^{\mu_i} F_i\}$ where μ_i is the smallest integer such that $CA^{\mu_i} F_i \neq 0$. For one-dimensional faults, the algorithm of Theorem A.1 is a very simple way to ¯nd $\mathcal{W}_i^*$.

Theorem A.1 also suggests a check for output separability. Let $\{f_{i_1}, \ldots, f_{i_{qi}}\}$ be any set of basis vectors for $\mathcal{F}_i$. An output separability check is that

$$
\operatorname{rank}\left[CA^{\beta_{1_1}} f_{1_1}, \ldots, CA^{\beta_{i_j}} f_{i_j}, \ldots, CA^{\beta_{q_{qq}}} f_{q_{qq}}\right] = p \tag{a.7}
$$

where $p = \sum q_i$ is the total number of basis vectors for the q failure spaces $\mathcal{F}_i$ and β_{i_j} is the smallest integer such that $CA^{\beta_{1_1}} f_{1_1} \neq 0$. Note that if (a.7) is not satis¯ed, then usually, the designer needs to discard some failures from the design set.

In the case where the dynamics (a.1) are time varying, an output separability check is that

$$\operatorname{rank}\left[Cb_{1_1}^{\beta_{1_1}}(t), \ldots, Cb_{i_j}^{\beta_{i_j}}(t), \ldots, Cb_{q_{q_q}}^{\beta_{q_{q_q}}}(t)\right] = p, \qquad \forall t \in [t_0, t_1] \tag{a.8}$$

where β_i is the smallest integer such that the following iteration:

$$b_{i_j}^1(t) = f_{i_j}(t) \tag{a.9a}$$

$$b_{i_j}^{k+1}(t) = Ab_{i_j}^k(t) + \dot{b}_{i_j}^k(t) \tag{a.9b}$$

results in a vector $b_{i_j}^k(t)$ such that $Cb_{i_j}^k(t) \neq 0$ for all $t \in [t_0, t_1]$. Note that (a.9) are the product of a Goh transformation on the output error space.

It is assumed that the system matrices $A(t)$, $C(t)$ and $F_i(t)$ are such that the number of iterations of (a.9) needed for the full rank condition is constant over the entire interval $[t_0, t_1]$, that is, the time variations of the system do not change the dimensionality of the detection problem. This restricts the applicability of this analysis to a subclass of time varying systems, but it avoids pathological cases. Assumptions such as this seem to be unavoidable when dealing with time varying systems. See, for example, (Clements and Anderson 1978).

When $Cf_{i_j} = 0$, both output separability tests fail immediately. However, this is not indicative of whether or not the system is output separable. As we will see in the next section, $Cf_{i_j} = 0$ is a sign that a f_{i_j} possess a higher-order detection space, meaning that it takes more than one vector to span this space. From Theorem A.1, one of these must lie outside the kernal of C and is, thus, the vector which must be used in the output separability test.

To ensure stability, the invariant subspaces $\mathcal{W}_i$ are usually chosen as a set of mutually detectable, minimal unobservability subspaces or detection spaces (Beard 1971) as they are also called in the context of fault detection. An unobservability subspace $\mathcal{T} \subseteq \mathcal{X}$ or UOS is a subspace with the property that $\mathcal{T}$ is the unobservable subspace of the pair $(HC, A + LC)$ for some L and H. This means not only that $\mathcal{T}$ is (C, A)-invariant but also that the spectrum of $(A + LC)$ induced on the factor space $\mathcal{X}/\mathcal{T}$ may be placed arbitrarily within a conjugate symmetry constraint and with respect to L such that $(A + LC)\mathcal{T} \subseteq \mathcal{T}$. Furthermore, when (C, A) is observable, the entire spectrum of $(A + LC)$ is arbitrary. If $\mathcal{T}(\mathcal{F})$ is the set of (C, A)-unobservability subspaces that contain $\mathcal{F}$, then it can be shown that $\mathcal{T}(\mathcal{F})$ has a smallest element denoted $\mathcal{T}^*$ (Willems 1982). The detection space is usually found as a minimal UOS, $\mathcal{T}^*$, because there is no known parameterization of all UOS and algorithms exist to compute the minimal UOS (White and Speyer 1987), (Massoumnia 1986).

One method for computing $\mathcal{T}^*$ is suggested by (Wonham 1985) as a numerically stable method for ¯nding supremal controllability subspaces. These are the dual of minimal unobservability subspaces or detection spaces. There are two steps. First, for a fault F_i, ¯nd the minimal (C, A)-invariant subspace $\mathcal{W}_i^*$ using the recursive (C, A)-invariant subspace algorithm as explained above. Next, calculate the invariant zero directions of the triple (C, A, F_i), if any. Denote the invariant zero directions as $\mathcal{V}_i$. Then

$$\mathcal{T}_i^* = \mathcal{W}_i^* \oplus \mathcal{V}_i$$

Detection space calculations are described in detail in (Wonham 1985) with ampli¯cation and examples given in (Douglas 1993).

Finally, a mutually detectable set of unobservability subspaces $\{\mathcal{T}_1^*, \dots, \mathcal{T}_q^*\}$ is one which satis¯es De¯nition A.1 such that the sum $\sum_{i=1}^q \mathcal{T}_i^*$ is also an UOS. While for any one UOS $\mathcal{T}_i$, the spectrum of $(A + LC)$ induced on $\mathcal{X}/\mathcal{T}_i$ may be placed arbitrarily with respect to L, it is not necessarily true that the factor space spectrum is arbitrary when several UOS are considered simultaneously. When a set of UOS $\mathcal{T}_1^*, \dots, \mathcal{T}_q^*$ is mutually detectable, the spectrum of $(A + LC)$ induced on $\mathcal{X}/\sum_{i=1}^q \mathcal{T}_i^*$ is arbitrary and, when (C, A) is observable, the entire spectrum of $(A + LC)$ is arbitrary.

A.4 The Restricted Diagonal Detection Filter Problem

In (Massoumnia 1986), the Beard-Jones detection ¯lter problem is shown to be a special case of the *restricted diagonal detection ¯lter problem* (RDDFP). First, de¯ne the complementary failure map $\hat{F}_i$ as

$$\hat{F}_i = [F_1, \ldots, F_{i-1}, F_{i+1}, \ldots, F_q] \tag{a.10}$$

The RDDFP, which is the dual of the *restricted decoupling problem* (Wonham 1985), is to find a set of q unobservability subspaces $\hat{\mathcal{T}}_1, \ldots, \hat{\mathcal{T}}_q$ such that

$$\mathcal{F}_i \cap \hat{\mathcal{T}}_i = 0$$
$$\hat{\mathcal{F}}_i \subseteq \hat{\mathcal{T}}_i$$

In the Beard-Jones detection filter, the idea is to confine each fault to an invariant subspace and then monitor that subspace through the residual for fault activity. In the RDDFP, the idea is to confine all the faults but one to an unobservable subspace, then monitor the observable factor space for activity caused by the remaining fault. By the definition of an unobservability subspace, there exists a projector H_i and a gain L such that $\hat{\mathcal{T}}_i$ is the unobservable subspace of the pair $(H_i C, A + LC)$. The signal

$$z_i = H_i(y - C\hat{x}) \tag{a.11}$$

is decoupled from all faults except F_i. Furthermore, $\mathcal{F}_i \cap \hat{\mathcal{T}}_i = 0$ implies that F_i is input observable so that $F_i m_i \neq 0$ implies that $z_i \neq 0$. Also, by construction, $\hat{H}_i$ satisfies

$$\operatorname{Ker} \hat{H}_i C = \hat{\mathcal{T}}_i + \operatorname{Ker} C$$

An explicit construction of $\hat{H}_i$ is to form CM_i as in (a.7)

$$CM_i = \left[CA^{\beta_{1_1}} f_{1_1}, \ldots, CA^{\beta_{i_j}} f_{i_j}, \ldots, CA^{\beta_{q_{qq}}} f_{q_{qq}} \right]$$

Then

$$\hat{H}_i = I - (CM_i)[(CM_i)^T (CM_i)]^{-1} (CM_i)^T$$

In the case where the dynamics (a.1) are time varying, $\hat{H}_i(t)$ may be constructed by forming $CM_i(t)$ as in (a.8)

$$CM_i = \left[Cb_{1_1}^{\beta_{1_1}}(t), \ldots, Cb_{i_j}^{\beta_{i_j}}(t), \ldots, Cb_{q_{qq}}^{\beta_{q_{qq}}}(t) \right]$$

where β_i is the smallest integer such that the following iteration:

$$
\begin{aligned}
b^1_{i_j}(t) &= \hat{f}_{i_j}(t) \\
b^{k+1}_{i_j}(t) &= Ab^k_{i_j}(t) + \dot{b}^k_{i_j}(t)
\end{aligned}
$$

results in a vector $b^k_{i_j}(t)$ such that $Cb^k_{i_j}(t) \neq 0$ for all $t \in [t_0, t_1]$. This time, $\hat{f}_{i_j}$ is taken to be vector from a basis for $\hat{\mathcal{F}}_i$.

It is easy to show that a Beard-Jones detection filter is always a restricted diagonal detection filter. For example, suppose a Beard-Jones detection filter is formed as a set of mutually detectable unobservability subspaces $\mathcal{T}^*_1, \ldots, \mathcal{T}^*_q$. Let

$$
\hat{\mathcal{T}}^*_i = \sum_{j \neq i} \mathcal{T}^*_j \qquad \text{(a.12)}
$$

Then, by the definition of mutual detectability, $\hat{\mathcal{T}}^*_i$ is itself a minimal unobservability subspace for the fault group $\hat{F}_i$.

APPENDIX B

Parameter Robustness By Left Eigenvector Assignment

ONCE the detection spaces are found, the next step is to find a fault detection filter gain. The gain is not unique and several methods exist for finding one. Eigenstructure assignment algorithms, which are the most accessible, are described in (Douglas and Speyer 1995b) and (White and Speyer 1987). An $\mathcal{H}_\infty$ disturbance bounded fault detection filter described in (Douglas and Speyer 1995a) is reviewed in Appendix C. The procedure applied in this report is a left eigenvector assignment algorithm introduced in (Douglas and Speyer 1996) and (Douglas 1993). This procedure is used because it extends directly to one that hedges against sensitivity to parameter uncertainty. Noise robustness algorithms such as the $\mathcal{H}_\infty$-bounded fault detection filter of (Douglas and Speyer 1995a) and Appendix C are not used here because disturbances and sensor noise are not yet included in the vehicle model. Furthermore, later, when they are included, the reduced-order fault detection filters provide a natural way to accommodate noise without the need for redesigning the filter.

The left eigenvector assignment algorithm works by assigning an eigenstructure in the dual space to a set of intersecting detection space annihilators. This means that left eigenvectors, which annihilate the detection spaces, are placed instead of right eigenvectors, which span the detection spaces, as is done in (White and Speyer 1987). Since the detection space annihilators intersect, care must be taken to ensure that the assigned eigenvectors are consistent.

Before proceeding, it is necessary to establish a dual relation between unobservability and controllability subspaces. First, introduce the following notation. $\mathcal{X}'$ denotes the dual space of $\mathcal{X}$ and if $C : \mathcal{X} \mapsto \mathcal{Y}$, then C' denotes the dual map $C'\mathcal{Y}' \mapsto \mathcal{X}'$. Writing C^T, the transpose of matrix C, for the dual map C' implies that bases have been chosen for $\mathcal{X}$ and $\mathcal{Y}$. Now, in (Wonham 1985) it is shown that if $\mathcal{T} \subseteq \mathcal{X}$ is a (C, A)-unobservability subspace then the annihilator of $\mathcal{T}$ denoted here by $\mathcal{T}^\perp \subseteq \mathcal{X}'$ is an (A', C')-controllability subspace in the dual system. Second, if $\mathcal{T}$ is a (C, A)-unobservability subspace, the observable part of the system is characterized by the factor space $\mathcal{X}/\mathcal{T}$ and the induced system maps. Furthermore, for any subspace $\mathcal{T} \subseteq \mathcal{X}$, the annihilator of $\mathcal{T}$ and the factor space $\mathcal{X}/\mathcal{T}$ are isomorphic, $\mathcal{T}^\perp \simeq (\mathcal{X}/\mathcal{T})'$.

The dual relation between unobservability and controllability subspaces is useful because any result found for controllability subspaces can be applied easily to the unobservability subspaces of a detection ¯lter. Consider the results of (Moore and Laub 1978) which are paraphrased as follows. The ¯rst statement describes a set of vectors in the kernal of C that can be assigned as closed-loop eigenvectors.

Theorem B.1. Let $A : \mathcal{X} \mapsto \mathcal{X}$, $B : \mathcal{U} \mapsto \mathcal{X}$ and $C : \mathcal{X} \mapsto \mathcal{Y}$. Then a set of linearly independent vectors $\{v_1, \ldots, v_k \mid v_i \in \operatorname{Ker} C \subseteq \mathcal{X}\}$ satis¯es $(A + BK)v_i = \lambda_i v_i$ for some $K : \mathcal{X} \mapsto \mathcal{U}$ and distinct self-conjugate complex numbers $\lambda_1, \ldots, \lambda_k$ if and only if v_i and v_j are conjugate pairs when λ_i and λ_j are and there exists a set of vectors $\{w_1, \ldots, w_k | w_i \in \mathcal{U}\}$ such that

$$\begin{bmatrix} A - \lambda_i I & B \\ C & 0 \end{bmatrix} \begin{bmatrix} v_i \\ w_i \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

It follows immediately that for a monic B, a set of vectors $\{v_1, \ldots, v_k\}$ satis¯es theorem B.1 if and only if $Kv_i = w_i$.

The second result also from (Moore and Laub 1978) characterizes the set of eigenvectors that span a supremal (A, B)-controllability subspace $\mathcal{R}^*$.

Theorem B.2. Let $\lambda_1, \ldots, \lambda_k$ be a set of distinct, self-conjugate complex numbers that satisfy

1) $k \geq \dim(\mathcal{R}^*)$ where $\mathcal{R}^*$ is the supremal (A, B)-controllability subspace in $\operatorname{Ker} C$

2) at least one λ_i is real

3) no λ_i or $\mathrm{Re}(\lambda_i)$ is a transmission zero of (C, A, B)

Let V_i and W_i solve

$$\begin{bmatrix} A - \lambda_i I & B \\ C & 0 \end{bmatrix} \begin{bmatrix} V_i \\ W_i \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Then $\mathcal{R}^* = \operatorname{Im} V_1 + \cdots + \operatorname{Im} V_k$.

Given the dual relationship between controllability and unobservability subspaces, the application of Theorems B.1 and B.2 to detection ¯lter design is immediate. First, consider just one detection space $\mathcal{T}_i^*$. Characterize the left eigenvectors that annihilate $\mathcal{T}_i^*$ and ¯nd a detection ¯lter gain L_i that produces $\mathcal{T}_i^*$. Next establish a consistency requirement on a detection ¯lter gain L that is to produce q detection spaces $\mathcal{T}_1^*, \ldots, \mathcal{T}_q^*$.

If $\mathcal{T}_i^* \subseteq \mathcal{X}$ with dimension ν_i is a detection space for fault F_i, the annihilator $(\mathcal{T}_i^*)^\perp$ is the supremal controllability subspace of the dual system with $(\mathcal{T}_i^*)^\perp \subseteq \operatorname{Ker} F_i'$ and has dimension $n - \nu_i$. Let $\Lambda_i = \{\lambda_{i_1}, \ldots, \lambda_{i_{n-\nu_i}}\}$ be a set of distinct self-conjugate complex numbers that does not include any of the invariant zeros of the triple (F_i', A', C'). By Theorem B.2 the annihilator of $\mathcal{T}_i^*$ satis¯es

$$(\mathcal{T}_i^*)^\perp = \operatorname{Im} V_{i_1} + \cdots + \operatorname{Im} V_{i_{n-\nu_i}}$$

where the V_{i_j} are found, along with W_{i_j}, by solving

$$\begin{bmatrix} A^T - \lambda_{i_j} I & C^T \\ F_i^T & 0 \end{bmatrix} \begin{bmatrix} V_{i_j} \\ W_{i_j} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{b.1}$$

where $j = 1, \ldots, n - \nu_i$ and where $\lambda_{i_j} \in \Lambda_i$. A set of linearly independent closed-loop left eigenvectors $v_{i_1}, \ldots, v_{i_{n-\nu_i}}$ that spans $(\mathcal{T}_i^*)^\perp$ satisfies Theorem B.1 and is found by solving

$$\begin{bmatrix} A^T - \lambda_{i_j} I & C^T \\ F_i^T & 0 \end{bmatrix} \begin{bmatrix} v_{i_j} \\ w_{i_j} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{b.2}$$

Since $v_{i_j} \in \operatorname{Im} V_{i_j}$ (b.1), the left eigenvectors may not be unique but they are constrained to be arranged in conjugate pairs when the given closed-loop eigenvalues λ_{i_j} are in conjugate pairs.

Now find a detection filter gain L_i. By the remark following Theorem B.1, L_i^T satisfies

$$L_i^T v_{i_j} = w_{i_j} \tag{b.3}$$

and $(A^T + C^T L_i^T) v_{i_j} = \lambda_{i_j} v_{i_j}$ for each $j = 1, \ldots, n - \nu_i$. Form two matrices $\hat{V}_i$ and $\hat{W}_i$

$$\hat{V}_i = \left[v_{i_1}, \ldots, v_{i_{n-\nu_i}} \right] \tag{b.4a}$$

$$\hat{W}_i = \left[w_{i_1}, \ldots, w_{i_{n-\nu_i}} \right] \tag{b.4b}$$

and solve $L_i^T \hat{V}_i = \hat{W}_i$. A real solution for L_i^T always exists because the v_{i_j} are linearly independent and the assigned closed-loop poles λ_{i_j} and eigenvectors v_{i_j} when complex are arranged in conjugate pairs. Finally, L_i, the detection filter gain found as the transpose

$$\hat{V}_i^T L_i = \hat{W}_i^T \tag{b.5}$$

satisfies $(A + L_i C)\mathcal{T}_i^* \subseteq \mathcal{T}_i^*$ and places the spectrum of $(A + L_i C)$ induced on $\mathcal{X}/\mathcal{T}_i^*$ as $\sigma(A + L_i C | \mathcal{X}/\mathcal{T}_i^*) = \Lambda_i$.

Because the detection filter has q detection spaces $\mathcal{T}_1^*, \ldots, \mathcal{T}_q^* \subseteq \mathcal{X}$, the detection filter gain L has to satisfy (b.5) for $i = 1, \ldots, q$ or

$$L^T \left[\hat{V}_1, \ldots, \hat{V}_q \right] = \left[\hat{W}_1, \ldots, \hat{W}_q \right] \tag{b.6}$$

Since the $\hat{V}_i$ and $\hat{W}_i$ represent $\sum_{i=1}^q (n - \nu_i)$ pairs of vectors (v_{i_j}, w_{i_j}), care must be taken to construct the $\hat{V}_i$ and $\hat{W}_i$ conformably. If (b.6) is to have a solution for L, there can be no more than n distinct pairs (v_{i_j}, w_{i_j}) and of these, the v_{i_j} must be linearly independent and arranged in conjugate pairs if a solution is to be unique and real.

Finding a set of left eigenvectors consistent with (b.6) is not di±cult but requires careful bookkeeping. Since $(\mathcal{T}_i^*)^\perp$ and $(\mathcal{X}/\mathcal{T}_i^*)'$ are isomorphic, the closed-loop spectrum induced on the factor space $\mathcal{X}/\mathcal{T}_i^*$ is

$$\sigma(A + L_iC|\mathcal{X}/\mathcal{T}_i^*) = \sigma(A' + C'L_i'|(\mathcal{T}_i^*)^\perp) = \hat{¤}_i$$

If $¤_i$ is the spectrum of $(A + L_iC)$ restricted to the invariant subspace $\mathcal{T}_i^*$

$$¤_i = \sigma(A + LC|\mathcal{T}_i^*)$$

then the spectrum of $(A + L_iC)$ is just

$$¤ = \sigma(A + L_iC) = ¤_i \cup \hat{¤}_i \tag{b.7}$$

Now, the subspaces $\mathcal{T}_1^*, \ldots, \mathcal{T}_q^*$ are independent when the faults are output separable and (C, A) is observable (Massoumnia 1986), (White and Speyer 1987), so

$$¤ = ¤_1 \cup \cdots \cup ¤_q \cup ¤_0$$

where $¤_0$ is a set of $\nu_0 = n - \nu_1 - \cdots - \nu_q$ eigenvalues associated with the complementary space $\hat{\mathcal{X}}_0 = \mathcal{X}/\sum_{i=1}^q \mathcal{T}_i^*$, $\nu_0 = \dim(\hat{\mathcal{X}}_0)$,

$$¤_0 = \sigma(A + LC|\mathcal{X}/\sum_{i=1}^q \mathcal{T}_i^*)$$

It follows from (b.7) that

$$\hat{¤}_i = \bigcup_{\substack{k=0 \\ k\neq i}}^q ¤_k \tag{b.8}$$

Since the sets of assigned closed-loop poles $\hat{¤}_i$ intersect, the sets of vectors v_{i_j} and w_{i_j} that solve (b.2) should also form intersecting sets compliant with (b.8). By (b.8), if $\lambda_{i_j} \in ¤_i$ for $i \neq 0$, then $\lambda_{i_j} \in \hat{¤}_{k\neq i}$ and the v_{i_j} and w_{i_j} that satisfy (b.2) now must satisfy

$$\begin{aligned} 0 &= (A^T - \lambda_{i_j}I)v_{i_j} + C^T w_{i_j} \\ 0 &= F_1^T v_{i_j} \end{aligned}$$

$$
\begin{aligned}
&\vdots \\
0 &= F_{i-1}^T v_{i_j} \\
0 &= F_{i+1}^T v_{i_j} \\
&\vdots \\
0 &= F_q^T v_{i_j}
\end{aligned}
$$

For $i = 0$ and $\lambda_{i_j} \in \Lambda_0$, then $\lambda_{i_j} \in \hat{\Lambda}_k$ for $k = 1, \ldots, q$ and the v_{i_j} and w_{i_j} that satisfy (b.2) now must satisfy

$$
\begin{aligned}
0 &= (A^T - \lambda_{i_j} I) v_{i_j} + C^T w_{i_j} \\
0 &= F_1^T v_{i_j} \\
&\vdots \\
0 &= F_q^T v_{i_j}
\end{aligned}
$$

The fault detection filter gain computation algorithm suggested by (b.2)-(b.6) and modified to force consistency among eigenvectors which span the intersecting detection space annihilators, is as follows.

Algorithm B.1.

1) Find the dimensions of the detection spaces $\nu_i = \dim \mathcal{T}_i^*$ for $i = 1, \ldots, q$ and the dimension of the complementary space $\nu_0 = n - \sum_{i=1}^{q} \nu_i$.

2) Define the complementary fault sets

$$
\hat{F}_i = \begin{cases} [F_1, \ldots, F_q] & \text{for } i = 0 \\ [F_1, \ldots, F_{i-1}, F_{i+1}, \ldots, F_q] & \text{for } 1 \leq i \leq q \end{cases} \tag{b.9}
$$

Define $(q + 1)$ sets of distinct self-conjugate complex numbers $\Lambda_0, \Lambda_1, \ldots, \Lambda_q$ where $\dim \Lambda_i = \nu_i$ and where no elements of Λ_i are zeros of the triple $(C, A, \hat{F}_i)$. By the remarks at the end of Appendix A, each of these sets may be specified arbitrarily except for conjugate symmetry when (C, A) is observable and when the detection spaces $\mathcal{T}_i^*$ are mutually detectable.

3) For $i = 0, \ldots, q$ and $j = 1, \ldots, \nu_i$ and for $\lambda_{i_j} \in$ ¤$_i$ solve

$$\begin{bmatrix} A^T - \lambda_{i_j} I & C^T \\ \hat{F}_i^T & 0 \end{bmatrix} \begin{bmatrix} v_{i_j} \\ w_{i_j} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{b.10}$$

for pairs (v_{i_j}, w_{i_j}) where the v_{i_j} are linearly independent for all i, j. Let

$$\mathcal{V}_i = \left[v_{i_1}, \ldots, v_{i_{\nu_i}} \right] \tag{b.11a}$$

$$\mathcal{W}_i = \left[w_{i_1}, \ldots, w_{i_{\nu_i}} \right] \tag{b.11b}$$

4) Solve for the detection ¯lter gain L as

$$\left[\mathcal{V}_0, \mathcal{V}_1, \ldots, \mathcal{V}_q \right]^T L = \left[\mathcal{W}_0, \mathcal{W}_1, \ldots, \mathcal{W}_q \right]^T \tag{b.12}$$

Appendix C

An $\mathcal{H}_\infty$ Bounded Fault Detection Filter

Analytical redundancy methods for fault detection and identi¯cation use a modeled dynamic relationship between system inputs and measured system outputs to form a residual process. Nominally, faults are detected as the residual process is nonzero only when a fault has occurred and is zero at other times. An example of a residual process for an observable system when no disturbances or sensor noise are present is the innovations process of any stable linear observer. A detection ¯lter is a linear observer with the gain constructed so that when a fault occurs, the residual responds in a known and ¯xed direction. Thus, when a nonzero residual is detected, a fault can be announced and identi¯ed at the same time. Since process disturbances and sensor noise also produce a nonzero residual, the ambiguity must be resolved with an appropriate threshold.

An objective of a detection ¯lter design in the presence of disturbances is to reduce the component of the residual due to the disturbance without at the same time degrading the component of the residual due to the fault. This suggests as a cost function, a ratio of transfer matrix norms (Frank and Wünnenberg 1989), (Lee 1994). In the numerator is the transfer matrix from the disturbance to the detection filter residual and in the denominator is the transfer matrix from the fault to the detection filter residual. This formulation works well when only one fault is to be detected. Generalized eigenvector solutions are found using a parity equation approach in (Frank and Wünnenberg 1989) and an optimization approach in (Lee 1994). Unfortunately, for the detection filter structure where several faults are isolated simultaneously, no similar problem formulation is available.

The approach taken here follows two steps. First, bound the $\mathcal{H}_\infty$ norm of the transfer matrix from the disturbance to the detection filter fault isolation residuals. Next, for each multi-dimensional fault isolation residual and working within the noise bound constraint, enhance the component due to the fault signal to be isolated. This is done by maximizing the ratio of the residual component due to a fault to the residual component due to the noise.

In the case of one-dimensional faults, the primary effect of the first step is to bound noise transmission through the complementary space, the state subspace independent of all detection spaces. The second step is not usually needed. This is because, generically, a fault detection space is given by the fault direction itself, which means the detection space is spanned by a single fixed eigenvector. The associated eigenvalue is the only degree of freedom left so there is no way to increase the residual component due to a fault without at the same time increasing the residual component due to the noise. In practical applications, plant and actuator failures usually are modeled as one-dimensional faults. Sensor faults generically require a two dimensional detection space so a design freedom exists where a residual component due to a fault could be enhanced.

This Appendix is organized as follows. Section C.1 shows that the detection filter gain is not unique and, given a set of invariant subspaces that solve the detection filter problem, parameterizes the set of detection filter gains. Section C.2 defines a disturbance robust detection filter problem and Section C.3 provides a stabilizing and $\mathcal{H}_\infty$ bounding detection filter gain by solving a modified algebraic Riccati equation. Section C.4 enhances the residual component due to the associated isolated fault signal by solving a generalized eigenvalue problem. Section C.5 provides an application to a simplified aircraft elevon and accelerometer fault detection filter where wind and sensor noise is present. The example illustrates how a numerical integration approach can be applied to solve the modified Riccati equation. Section C.6 contains a few concluding remarks.

C.1 Detection Filter Gain Parameterization

Given a set of subspaces $\mathcal{W}_1, \ldots, \mathcal{W}_q$ that solve the detection filter problem, the next problem is to characterize the set of maps $L : \mathcal{Y} \mapsto \mathcal{X}$ such that $L \in \cap_{i=1}^{q} \underline{L}(\mathcal{W}_i)$ where

$$\underline{L}(\mathcal{W}_i) \triangleq \{L \mid (A + LC)\mathcal{W}_i \subseteq \mathcal{W}_i\}$$

A first step is to find a set $\underline{L}(\mathcal{W})$ for any one (C, A)-invariant subspace $\mathcal{W}$. Proposition C.3 parameterizes $L \in \underline{L}(\mathcal{W})$ in two parameters $\alpha : C\mathcal{W} \mapsto \mathcal{W}$ and $\beta : \mathcal{Y} \mapsto \mathcal{X}$. Then, given a set of (C, A)-invariant subspaces $\mathcal{W}_1, \ldots, \mathcal{W}_q$ that solve the detection filter problem, Proposition C.4 parameterizes $L \in \cap_{i=1}^{q} \underline{L}(\mathcal{W}_i)$ in $q + 1$ parameters $\alpha_1, \ldots, \alpha_q$ and β. First, a Lemma from (White and Speyer 1987, Lemma 1), except for the geometric language, is restated to provide a solution to a generalized inverse problem. Lemma C.2 provides a few well-known properties of projections.

Lemma C.1. Let $B : \mathcal{U} \mapsto \mathcal{X}$, $C : \mathcal{X} \mapsto \mathcal{Y}$ and $D : \mathcal{U} \mapsto \mathcal{Y}$ where B is monic. Then a general solution of $CB = D$ for C is given by

$$C = D\bar{P}_B + K(I - P_B) \tag{c.1}$$

where $P_B : \mathcal{X} \mapsto \mathcal{X}$ is any projection such that $\Im P_B = \Im B$, $\bar{P}_B : \mathcal{X} \mapsto \mathcal{U}$ is the natural projection where $B\bar{P}_B = P_B$ and $K : \mathcal{X} \mapsto \mathcal{Y}$ is arbitrary.

Lemma C.2. Let $C : \mathcal{X} \mapsto \mathcal{Y}$ and let $P : \mathcal{X} \mapsto \mathcal{X}$ be any projection. Then $\operatorname{Ker} P \subseteq \operatorname{Ker} C$ if and only if $C = CP$. Now let $\operatorname{Ker} P = \operatorname{Ker} C$ and let V decompose P as $VV^T = P$ and $V^T V = I$. Then CV is monic with $\Im CV = \Im C$.

An easy way to find a projector P that satisfies Lemma C.2 is to find the singular value decomposition of C. For $C = U\S V^T$ where $\S$ is a diagonal matrix of nonzero singular values, the V of the lemma are the right singular vectors of C. Thus $P = VV^T$ and $CV = U\S V^T V = U\S$ is monic with $\Im C = \Im U\S$.

Proposition C.3. Let $\mathcal{W} \subset \mathcal{X}$ be a (C, A)-invariant subspace with insertion map $W : \mathcal{W} \mapsto \mathcal{X}$. Let $P : \mathcal{W} \mapsto \mathcal{W}$ be any projection where $\operatorname{Ker} P = \operatorname{Ker} CW$ and let $\hat{F}$ decompose P as $\hat{F}\hat{F}^T = P$ and $\hat{F}^T\hat{F} = I$. Let $H : \mathcal{Y} \mapsto \mathcal{Y}$ be another projection where $\Im H = C\mathcal{W}$ and let $\bar{H}$ be the associated natural projection that satisfies $CW\hat{F}\bar{H} = H$ and $\bar{H}CW\hat{F} = I$. Then $L : \mathcal{Y} \mapsto \mathcal{X}$ satisfies $(A + LC)W = WA_W$ for some $A_W : \mathcal{W} \mapsto \mathcal{W}$ if and only if

$$L = (-AW\hat{F} + W\alpha)\bar{H} + \beta(I - H) \tag{c.2}$$

for some $\alpha : C\mathcal{W} \mapsto \mathcal{W}$ and $\beta : \mathcal{Y} \mapsto \mathcal{X}$.

Proof. ($\Rightarrow$) Assume L satisfies $(A + LC)W = WA_W$ for some map A_W. Then

$$LCW = -AW + WA_W$$

and

$$LCW\hat{F} = -AW\hat{F} + WA_W\hat{F} \tag{c.3}$$

Now $\hat{F}$ is defined so that $\hat{F}\hat{F}^T$ is a projection with $\operatorname{Ker} CW = \operatorname{Ker} \hat{F}\hat{F}^T$ and $\hat{F}^T\hat{F} = I$. Therefore, by Lemma C.2, $CW\hat{F}$ is monic and by (c.3) and Lemma C.1

$$L = (-AW\hat{F} + WA_W\hat{F})\bar{H} + \beta(I - H)$$

So $(A + LC)W = WA_W$

$$\Rightarrow \quad L = (-AW\hat{F} + W\alpha)\bar{H} + \beta(I - H)$$

where $\alpha = A_W\hat{F}$ and β is anything.

($\Leftarrow$) Suppose $L = (-AW\hat{F} + W\alpha)\bar{H} + \beta(I - H)$. Now $HCW\hat{F} = CW\hat{F}$ and $\bar{H}CW\hat{F} = I$ so $LCW\hat{F} = (-AW\hat{F} + W\alpha)$ and

$$(A + LC)W\hat{F} = W\alpha \tag{c.4}$$

$\hat{F}$ is de¯ned so that $\hat{F}\hat{F}^T$ is a projector with $\operatorname{Ker} CW = \operatorname{Ker} \hat{F}\hat{F}^T$ and $\hat{F}^T\hat{F} = I$. Therefore, by Lemma C.2, $CW = CW\hat{F}\hat{F}^T$ and it follows that $CW(I - \hat{F}\hat{F}^T) = 0$ and

$$\Im\left[W(I - \hat{F}\hat{F}^T)\right] \subseteq \mathcal{W} \cap \operatorname{Ker} C \tag{c.5}$$

Since for any (C, A)-invariant subspace $\mathcal{W}$ it is true that $A(\mathcal{W} \cap \operatorname{Ker} C) \subseteq \mathcal{W}$, it follows from (c.5) that for some $\bar{A}_W$

$$AW(I - \hat{F}\hat{F}^T) = W\bar{A}_W \tag{c.6}$$

and

$$(A + LC)W(I - \hat{F}\hat{F}^T) = W\bar{A}_W$$

By (c.4), $(A + LC)W\hat{F}\hat{F}^T = W\alpha\hat{F}^T$. So

$$(A + LC)W = W\left(\alpha\hat{F}^T + \bar{A}_W\right)$$

and $L = (-AW\hat{F} + W\alpha)\bar{H} + \beta(I - H)$

$$\Rightarrow \quad (A + LC)W = WA_W$$

where $A_W = \alpha\hat{F}^T + \bar{A}_W$ and where $\bar{A}_W$ satis¯es (c.6). Note that $\bar{A}_W = \bar{A}_W(I - \hat{F}\hat{F}^T)$ so

$$A_W = \alpha\hat{F}^T + \bar{A}_W(I - \hat{F}\hat{F}^T)$$

By Lemma C.1 A_W is a particular solution to $\alpha = A_W\hat{F}$.

The remark following Lemma C.2 shows that $\hat{F}$ is the set of right singular vectors of CW.

Proposition C.4. Let $\mathcal{W}_1, \ldots, \mathcal{W}_q \subset \mathcal{X}$ be a set of (C, A)-invariant subspaces that solve the detection ¯lter problem and let the $W_i : \mathcal{W}_i \mapsto \mathcal{X}$ be the insertion maps. Let P_i, $\hat{F}_i$, H_i and $\bar{H}_i$ associated with $\mathcal{W}_i$ be as in Proposition C.3 but partially specify the kernal of H_i and $\bar{H}_i$ as $\sum_{j \neq i} C\mathcal{W}_j \subseteq \operatorname{Ker} H_i = \operatorname{Ker} \bar{H}_i$. Also, de¯ne the projection $H_0 = (I - \sum_{i=1}^{q} H_i)$ and the associated natural projection $\bar{H}_0$. Finally, de¯ne a set of maps

$$\underline{L}(\mathcal{W}_i) = \{L : \mathcal{Y} \mapsto \mathcal{X} \mid (A + LC)\mathcal{W}_i \subseteq \mathcal{W}_i\}$$

Then $L \in \cap_{i=1}^{q} \underline{L}(\mathcal{W}_i)$ if and only if

$$L = \sum_{i=1}^{q}(-AW_i\hat{F}_i + W_i\alpha_i)\bar{H}_i + \beta\bar{H}_0 \tag{c.7}$$

for some $\alpha_0 : \Im H_0 \mapsto \mathcal{X}$ and $\alpha_i : C\mathcal{W}_i \mapsto \mathcal{W}_i$ where $i = 1, \ldots, q$.

Proof. ($\Rightarrow$) Assume $L \in \underline{L}(\mathcal{W}_i)$. Then L satisfies $(A + LC)W_i = W_iA_{W_i}$ for some $A_{W_i} : \mathcal{W} \mapsto \mathcal{W}$ for $i = 1, \ldots, q$. So

$$LCW_i = -AW_i + W_iA_{W_i}$$

and

$$LCW_i\hat{F}_i = -AW_i\hat{F}_i + W_iA_{W_i}\hat{F}_i$$

and

$$\begin{aligned} &L\left[CW_1\hat{F}_1, \ldots, CW_q\hat{F}_q\right] = \\ &\quad \left[\left(-AW_1\hat{F}_1 + W_1A_{W_1}\hat{F}_1\right), \ldots, \left(-AW_q\hat{F}_q + W_qA_{W_q}\hat{F}_q\right)\right] \end{aligned} \tag{c.8}$$

The $\hat{F}_i$ are defined so that $\hat{F}_i\hat{F}_i^T$ is a projector with $\operatorname{Ker} CW_i = \operatorname{Ker} \hat{F}_i\hat{F}_i^T$ and $\hat{F}_i^T\hat{F}_i = I$. Therefore, Lemma C.2 shows that $\Im CW_i = \Im CW_i\hat{F}_i$ and $CW_i\hat{F}_i$ is monic. Since the $\mathcal{W}_1, \ldots, \mathcal{W}_q$ solve the detection filter problem, they are output separable, which means the output subspaces $C\mathcal{W}_1, \ldots, C\mathcal{W}_q$ are independent. Therefore, $[CW_1\hat{F}_1, \ldots, CW_q\hat{F}_q]$ is monic.

In Proposition C.3 $\operatorname{Ker} H$ is not specified and is not important. Here however, $H_iCW_j = 0$ so if H is the projection $H = \sum_{i=1}^{q} H_i$ then

$$H\left[CW_1\hat{F}_1, \ldots, CW_q\hat{F}_q\right] = \left[CW_1\hat{F}_1, \ldots, CW_q\hat{F}_q\right]$$

A natural projection $\bar{H}$ associated with H is

$$\bar{H} = \begin{bmatrix} \bar{H}_1 \\ \vdots \\ \bar{H}_q \end{bmatrix}$$

because

$$\begin{aligned}
\left[CW_1\hat{F}_1, \ldots, CW_q\hat{F}_q\right]\bar{H} &= \sum_{i=1}^{q} CW_i\hat{F}_i\bar{H}_i \\
&= \sum_{i=1}^{q} H_i \\
&= H
\end{aligned}$$

and

$$\bar{H}\left[CW_1\hat{F}_1, \ldots, CW_q\hat{F}_q\right] = \text{diag}\left(\bar{H}_iCW_i\hat{F}_i\right) = I$$

Since, $[CW_1\hat{F}_1, \ldots, CW_q\hat{F}_q]$ is monic and H and $\bar{H}$ meet the requirements of Lemma C.1, the general solution of (c.8) for L is

$$\begin{aligned}
L &= \left[\left(-AW_1\hat{F}_1 + W_1A_{W_1}\hat{F}_1\right), \ldots, \left(-AW_q\hat{F}_q + W_qA_{W_q}\hat{F}_q\right)\right]\bar{H} + \hat{\beta}(I - H) \\
&= \sum_{i=1}^{q}(-AW_i\hat{F}_i + W_iA_{W_i}\hat{F}_i)\bar{H}_i + \hat{\beta}(I - H) \\
&= \sum_{i=1}^{q}(-AW_i\hat{F}_i + W_i\alpha_i)\bar{H}_i + \hat{\beta}(I - H)
\end{aligned}$$

where $\alpha_i = A_{W_i}\hat{F}_i$ and $\hat{\beta}$ is anything. Finally, it follows directly from the definitions of H and $\bar{H}_0$ that for any $\hat{\beta}$, there exists β such that $\hat{\beta}(I - H) = \beta\bar{H}_0$. So,

$$L = \sum_{i=1}^{q}(-AW_i\hat{F}_i + W_i\alpha_i)\bar{H}_i + \beta\bar{H}_0$$

($\Leftarrow$) Assume

$$\begin{aligned}
L &= \sum_{i=1}^{q}(-AW_i\hat{F}_i + W_i\alpha_i)\bar{H}_i + \beta\bar{H}_0 \\
&= \sum_{i=1}^{q}(-AW_i\hat{F}_i + W_i\alpha_i)\bar{H}_i + \hat{\beta}(I - H)
\end{aligned}$$

where the equality follows from the definitions of H and $\bar{H}_0$. Since $H_iH_j = 0$,

$$(I - H) = (I - \sum_{i=1}^{q} H_i) = (I - \sum_{j\neq i} H_j)(I - H_i)$$

Then

$$\begin{aligned}
L &= \sum_{i=1}^{q}(-AW_i\hat{F}_i + W_i\alpha_i)\hat{H}_i + \beta(I - \sum_{i=1}^{q} H_i) \\
&= \sum_{i=1}^{q}(-AW_i\hat{F}_i + W_i\alpha_i)\hat{H}_i + \beta(I - \sum_{j\neq i} H_j)(I - H_i) \\
&= (-AW_i\hat{F}_i + W_i\alpha_i)\hat{H}_i + \left[\sum_{j\neq i}(-AW_j\hat{F}_j + W_j\alpha_j)\hat{H}_j + \beta(I - \sum_{j\neq i} H_j)\right](I - H_i)
\end{aligned}$$

Therefore, L has the form

$$L = (-AW_i\hat{F}_i + W_i\alpha_i)\hat{H}_i + \beta_i(I - H_i)$$

where

$$\beta_i = \sum_{j\neq i}(-AW_j\hat{F}_j + W_j\alpha_j)\hat{H}_j + \beta(I - \sum_{j\neq i} H_j)$$

By Proposition C.3, $L \in \underline{L}(\mathcal{W}_i)$ for each $\mathcal{W}_i$ which means $L \in \cap_{i=1}^{q}\underline{L}(\mathcal{W}_i)$.

C.2 A Disturbance Robust Detection Filter Problem

Section C.1 showed that a detection filter gain associated with a set of detection filter solution spaces $\mathcal{W}_1, \ldots, \mathcal{W}_q$ is easy to find, but generally is not unique. In this section, the $\mathcal{W}_1, \ldots, \mathcal{W}_q$ are found as for the deterministic case, but the nonuniqueness of the detection filter gain is treated as a degree-of-freedom in the detection filter design. This leads to the definition of a noise robust detection filter problem where the objective is to find a detection filter gain that minimizes or bounds a norm of the transfer matrix from the disturbance to the residual.

The linear time-invariant system of (a.1) with q failure modes is extended to include disturbances as

$$\dot{x} = Ax + B\omega + B_u u + \sum_{i=1}^{q} F_i m_i \qquad \text{(c.9a)}$$

$$y = Cx + D\omega. \qquad \text{(c.9b)}$$

The input ω includes dynamic disturbances and sensor noise and is square integrable over $[0, \infty)$.

The error dynamics and residual of a full-order filter have the same form as the observer (a.3, a.4)

$$\dot{e} = (A + LC)e - (B + LD)\omega - \sum_{i=1}^{q} F_i m_i \tag{c.10a}$$

$$r = C\hat{x} - y = Ce - D\omega. \tag{c.10b}$$

Since only forcing terms differentiate the residual process of the observer (a.3, a.4) from (c.10), the detection filter structure does not change with the introduction of disturbances and sensor noise. However, with the residual driven by an unknown signal, a nonzero residual does not necessarily mean a fault has occurred.

An objective of a detection filter design in the presence of disturbances is to reduce the component of the residual due to the disturbance without at the same time degrading the component of the residual due to the fault. This suggests as a cost function, a ratio of transfer matrix norms (Frank and Wünnenberg 1989). The transfer matrix from the disturbance to the residual is in the numerator and the transfer matrix from the fault to the residual is in the denominator. Unfortunately, this formulation requires some assumption about the functional form of the fault because a transfer matrix norm does not convey much information about the size of a transfer matrix output when nothing can be said about the input. Since it is a standard and reasonable assumption that process and sensor noise is white or nearly so, only the transfer matrix from the disturbance to the detection filter residual is retained in the definition of a noise robust detection filter problem.

Before continuing, it is necessary to carefully define what is meant by the component of the residual due to the fault. Define z_i as a projection of the observer residual (c.10) onto the output subspace $C\mathcal{W}_i$. Let $H_i : \mathcal{Y} \mapsto \mathcal{Y}$ be any projection onto $C\mathcal{W}_i$ and along the $C\mathcal{W}_{j \neq i}$ so that $C\mathcal{W}_i = \Im H_i$ and $\sum_{j \neq i} C\mathcal{W}_j \subseteq \operatorname{Ker} H_i$. Let $\bar{H}_i$ be the associated natural projection and define $\bar{z}_i$, a fault residual, as

$$\bar{z}_i = \bar{H}_i r \tag{c.11}$$

Using $\tilde{H}_i$ rather than H_i in (c.11) doesn't change any information given by the fault residual but is convenient later when certain matrix inverses are needed.

Now consider that for a system with q faults as in (c.9), there are q transfer matrices from the system disturbance to each of the fault residuals $\tilde{z}_i$ (c.11). There are several ways to proceed. One approach is to define a multi-objective problem where a detection filter gain L is found that in some way simultaneously bounds or makes small all the transfer matrix norms $\|T_{\tilde{z}_i\omega}\|$, for example, a Pareto optimal solution. Another is to abandon the structure of the full-order detection filter for a system of q *residual generators* (Massoumnia et al. 1989). The q reduced-order filter gains are found independently of one another with the penalty that the order of the combined system usually is somewhat larger than the full-order detection filter. The approach taken here is to combine the fault residuals into a single detection filter output as follows.

Define a combined fault residual $z \in (C\mathcal{W}_1 \times \cdots \times C\mathcal{W}_q)$ by forming a map H from the $\tilde{H}_i$ in the expected way:

$$z = Hr, \qquad H^T = \left[\tilde{H}_1^T, \ldots, \tilde{H}_q^T\right] \tag{c.12}$$

The combined fault residual z provides the same information as the fault residuals, but it combines the $\tilde{z}_1, \ldots, \tilde{z}_q$ so that a single cost function can be defined for the detection filter. A noise robust detection filter problem is to find a set of subspaces $\mathcal{W}_i$ that solve the detection filter problem of Definition A.1. Then, given the $\mathcal{W}_i$ and the associated filter gain sets

$$\underline{L}(\mathcal{W}_i) = \{L_i \mid (A + L_i C)\mathcal{W}_i \subseteq \mathcal{W}_i\}$$

find a filter gain $L \in \cap \underline{L}(\mathcal{W}_i)$ that bounds or minimizes some norm $\|T_{z\omega}\|$ where $T_{z\omega}$ is the transfer matrix from the disturbance ω to the combined fault residual z of (c.12).

Note that L is found in a two-step process. First, a set of subspaces $\mathcal{W}_i$ is found that satisfies Definition A.1. Then a map L is found from the set $\cap \underline{L}(\mathcal{W}_i)$. The alternative is to find L from the union of sets $\cap \underline{L}(\mathcal{W}_i)$, where the union is taken over all sets of subspaces $\mathcal{W}_i$ that satisfy Definition A.1. While the latter statement certainly is more general, it is impractical because there is no known parameterization of all (C, A)-invariant subspaces $\mathcal{W}_i$.

C.3 An $\mathcal{H}_\infty$ Bounded Detection Filter

The main result of this section is a proposition that provides an $\mathcal{H}_\infty$ norm bounding detection filter gain. Before this result is stated, a more general $\mathcal{H}_\infty$ norm bounding theorem is needed. Consider an observer with error dynamics and output

$$\dot{e} = (A + LC)e + (B + LD)\omega \tag{c.13a}$$

$$z = C_z e + D_z \omega \tag{c.13b}$$

The following theorem and corollary provide a filter gain L that stabilizes the filter and bounds the $\mathcal{H}_\infty$ norm of the transfer matrix from ω to z. This standard result is mainly from Lemma 1 of (Willems 1971) so no proof is provided here.

Theorem C.5. Consider a system G with the form (c.13), where $(A - BD^T(DD^T)^{-1}C)$ has no purely imaginary eigenvalues and where $(DD^T)^{-1}$ exists. Suppose there exists a scalar real constant $\gamma > 0$ and a symmetric positive definite real matrix $Y > 0$ that satisfies the following algebraic Riccati equation

$$\begin{aligned} 0 \;=\; & (A + LC)Y + Y(A + LC)^T + (B + LD)(B + LD)^T \\ & + \gamma^{-2}(YC_z^T + BD_z^T)(YC_z^T + BD_z^T)^T \end{aligned} \tag{c.14}$$

Then $(A + LC)$ is stable and $\|G\|_\infty \leq [\gamma^2 + \sigma^2_{\max}(D_z)]^{1/2}$ where $\sigma_{\max}(D_z)$ is the largest singular value of D_z.

When the terms of (c.14) are manipulated to isolate L, a corollary which provides an L that stabilizes G and bounds $\|G\|_\infty$ follows immediately.

Corollary C.6. Suppose a symmetric positive definite real matrix $Y > 0$ satisfies the following algebraic Riccati equation

$$\begin{aligned} 0 \;=\;& \left[A - BD^T(DD^T)^{-1}C + \gamma^{-2}BD_z^TC_z\right]Y \\ &+ Y\left[A - BD^T(DD^T)^{-1}C + \gamma^{-2}BD_z^TC_z\right]^T \\ &+ B\left[I - D^T(DD^T)^{-1}D + \gamma^{-2}D_z^TD_z\right]B^T \\ &- Y\left[C^T(DD^T)^{-1}C - \gamma^{-2}C_z^TC_z\right]Y \end{aligned} \tag{c.15}$$

Then for

$$L = -(YC^T + BD^T)(DD^T)^{-1} \tag{c.16}$$

$(A + LC)$ is stable and $\|G\|_\infty \le [\gamma^2 + \sigma_{\max}^2(D_z)]^{1/2}$ where $\sigma_{\max}(D_z)$ is the largest singular value of D_z.

Standard results strengthen Corollary C.6 by replacing (c.15) with conditions on an associated Hamiltonian matrix and adding a system detectability requirement (Kučera 1972, Doyle 1984). That is not done here because in the next proposition, the Riccati equation (c.15) is modified to provide a detection filter gain and has no associated Hamiltonian matrix.

In the detection filter problem, L is constrained to generate a set of q invariant subspaces $\mathcal{W}_1, \ldots, \mathcal{W}_q$. There is no reason to expect that L, at the same time, should satisfy (c.16). In the next proposition, (c.16) is modified so that L satisfies both constraints. When the modified relation is substituted for L in (c.14) and L is eliminated, the result is an algebraic Riccati equation with an extra term. The modified Riccati equation has no associated Hamiltonian and conditions for the uniqueness or even the existence of a solution are unknown. However, (Veillette et al. 1992) reports success in finding iterative numerical solutions to a similar relation arising from a decentralized control problem. An example in the next section illustrates the application a numerical integration approach.

Before stating the main proposition, it is convenient to rearrange the detection filter error dynamics by combining the error dynamics (c.10) with the detection filter gain (c.7).

Then the problem of choosing the parameters α_0 and $\alpha_1, \ldots, \alpha_q$ has the same form as the problem of choosing a set of $q + 1$ constant feedback gains for the system

$$\underline{e} = \hat{A}e - \hat{B}\omega - \sum_{i=1}^{q} F_i m_i + W_1 u_1 + \cdots + W_q u_q + u_0 \tag{c.17a}$$

$$y_1 = \bar{H}_1 Ce - \bar{H}_1 D\omega, \qquad u_1 = \alpha_1 y_1 \tag{c.17b}$$

$$\vdots \tag{c.17c}$$

$$y_q = \bar{H}_q Ce - \bar{H}_q D\omega, \qquad u_q = \alpha_q y_q \tag{c.17d}$$

$$y_0 = \bar{H}_0 Ce - \bar{H}_0 D\omega, \qquad u_0 = \alpha_0 y_0 \tag{c.17e}$$

where

$$\hat{A} = A + \hat{L}C \tag{c.17f}$$

$$\hat{B} = B + \hat{L}D \tag{c.17g}$$

$$\hat{L} = -\sum_{i=1}^{q} AW_i \hat{F}_i \bar{H}_i \tag{c.17h}$$

Proposition C.7. Consider the system G with output given by (c.12)

$$G = \left[\begin{array}{c|ccccc} \hat{A} & -\hat{B} & W_1 & \cdots & W_q & I \\ \hline HC & -HD & 0 & \cdots & 0 & 0 \\ \bar{H}_1 C & -\bar{H}_1 D & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \bar{H}_q C & -\bar{H}_q D & 0 & \cdots & 0 & 0 \\ \bar{H}_0 C & -\bar{H}_0 D & 0 & \cdots & 0 & 0 \end{array}\right]$$

De¯ne

$$C_2 = \begin{bmatrix} \bar{H}_1 C \\ \vdots \\ \bar{H}_q C \\ \bar{H}_0 C \end{bmatrix} \quad D_{21} = \begin{bmatrix} \bar{H}_1 D \\ \vdots \\ \bar{H}_q D \\ \bar{H}_0 D \end{bmatrix} \quad V = D_{21} D_{21}^T$$

and the partitioning matrices ¦$_1, \ldots,$ ¦$_q$ and ¦$_0$

$$¦_1 = \begin{bmatrix} I \\ \vdots \\ 0 \\ 0 \end{bmatrix} \quad \cdots \quad ¦_q = \begin{bmatrix} 0 \\ \vdots \\ I \\ 0 \end{bmatrix} \quad ¦_0 = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ I \end{bmatrix}$$

such that

$$
\begin{aligned}
\text{¦}_i^T[\ C_2, \quad D_{21}\ ] &= [\ \hat{H}_i C, \quad \hat{H}_i D\ ], \\
\text{¦}_0^T[\ C_2, \quad D_{21}\ ] &= [\ \hat{H}_0 C, \quad \hat{H}_0 D\ ]
\end{aligned}
$$

Now define a set of projections $P_{W_1}, \ldots, P_{W_q}$ where $\Im P_{W_i} = \Im W_i$ and define a set of associated natural projections $\hat{P}_{W_i}$, which satisfy $W_i \hat{P}_{W_i} = P_{W_i}$. Assume $(\hat{A} - \hat{B} D_{21}^T V^{-1} C_2)$ has no eigenvalues on the imaginary axis. Let $\gamma > 0$ be a constant real scalar and suppose there exists $Y > 0$ such that

$$
\begin{aligned}
0 \quad = \quad & \left[\hat{A} - \hat{B} D_{21}^T V^{-1} C_2 + \gamma^{-2} \hat{B} D^T H^T C_2\right] Y + Y \left[\hat{A} - \hat{B} D_{21}^T V^{-1} C_2 + \gamma^{-2} \hat{B} D^T H^T C_2\right]^T \\
& + \hat{B} \left[I - D_{21}^T V^{-1} D_{21} + \gamma^{-2} D^T H^T H D\right] \hat{B}^T \\
& - Y \left[C_2^T V^{-1} C_2 - \gamma^{-2} C^T H^T H C\right] Y \\
& + \left(\sum_{i=1}^{q} (I - P_{W_i})(Y C_2^T + \hat{B} D_{21}^T) V^{-1} \text{¦}_i \hat{H}_i D\right) \\
& \qquad \times \left(\sum_{i=1}^{q} (I - P_{W_i})(Y C_2^T + \hat{B} D_{21}^T) V^{-1} \text{¦}_i \hat{H}_i D\right)^T
\end{aligned}
\tag{c.18}
$$

Then

$$
\begin{aligned}
\alpha_1 &= -\hat{P}_{W_1}(Y C_2^T + \hat{B} D_{21}^T) V^{-1} \text{¦}_1 \\
&\vdots \\
\alpha_q &= -\hat{P}_{W_q}(Y C_2^T + \hat{B} D_{21}^T) V^{-1} \text{¦}_q \\
\alpha_0 &= -(Y C_2^T + \hat{B} D_{21}^T) V^{-1} \text{¦}_0
\end{aligned}
$$

stabilizes G and bounds the transfer matrix $T_{z\omega}$ as $\|T_{z\omega}\|_\infty \le [\gamma^2 + \sigma_{\max}^2(HD)]^{1/2}$ where $\sigma_{\max}(HD)$ is the largest singular value of HD.

Proof. The transfer matrix $T_{z\omega}$ is

$$
T_{z\omega} = \left[\begin{array}{c|c} A_T & -B_T \\ \hline HC & -HD \end{array}\right]
$$

where

$$
\begin{aligned}
A_T &= \hat{A} + \sum_{i=1}^{q} W_i \alpha_i \hat{H}_i C + \alpha_0 \hat{H}_0 C \\
B_T &= \hat{B} + \sum_{i=1}^{q} W_i \alpha_i \hat{H}_i D + \alpha_0 \hat{H}_0 D
\end{aligned}
$$

By Theorem C.5 and since $(\hat{A} - \hat{B}D_{21}^T V^{-1} C_2)$ has no eigenvalues on the imaginary axis, it is sufficient to show that $S = 0$ for some $Y > 0$ where

$$
\begin{aligned}
S &= A_T Y + Y A_T^T + B_T B_T^T + \\
&\quad \gamma^{-2}(YC^T + \hat{B}D^T)H^T H (YC^T + \hat{B}D^T)^T
\end{aligned}
$$

The rest of the proof involves algebraic manipulations that put S in the form of the modified algebraic Riccati equation (c.18).

C.4 Fault Enhancement

As discussed in the introduction, it is not enough to bound the residual component due to the process disturbances and sensor noise since this might, at the same time, make the fault residual component small. The approach taken here is to enhance each fault residual component while maintaining the disturbance and sensor noise bound.

Consider a cost function given as the fault signal to noise ratio

$$
J_i = \frac{\|T_{z_i m_i}\|_\infty}{\|T_{z_i \omega}\|_\infty} \tag{c.19}
$$

This is the same cost function as given in (Frank and Wünnenberg 1989) for a set of parity equations. Combining the filter gain of Proposition C.7 with results from (Doyle et al. 1989) provide a Youla parameterization of stable and $\mathcal{H}_\infty$ norm bounded transfer matrices. This could be applied to the fault detection filter by restricting the Youla parameter to those which maintain the invariant subspace structure. Maximizing (c.19) with respect to a restricted set of Youla parameters is a very difficult problem. A more tractable problem may be defined as follows.

First, consider the fault detection filter transfer matrix for the fault isolation residual z_i. By the filter unobservability subspace structure, only the fault m_i influences residual z_i, so a reduced-order realization is written. The subscript i is dropped for notational convenience.

$$
\begin{aligned}
\dot{\hat{e}}(t) &= \hat{A}\hat{e}(t) + \hat{F}m(t) + \hat{B}\omega(t) \\
z(t) &= \hat{C}\hat{e}(t) + D_m m(t) + D_\omega \omega(t)
\end{aligned}
$$

The error $\hat{e}$ lies in the factor space $\hat{e} \in \mathcal{X} / \sum_{j \neq i} \mathcal{T}_j$, the observable factor space with respect to z. All maps are taken as induced on this factor space. Now consider signals $m(t)$ and $\omega(t)$ as elements of $\mathcal{L}_2(-\infty, 0]$ spaces of appropriate dimensions and define the controllability operators

$$
\begin{aligned}
\psi_m : \mathcal{L}_2(-\infty, 0] \mapsto \mathcal{R}^n &\triangleq \int_{-\infty}^{0} e^{-\bar{A}\tau} \hat{F} m(\tau) d\tau \\
\psi_\omega : \mathcal{L}_2(-\infty, 0] \mapsto \mathcal{R}^n &\triangleq \int_{-\infty}^{0} e^{-\bar{A}\tau} \hat{B} \omega(\tau) d\tau
\end{aligned}
$$

Then $z = z_m + z_\omega$ where z_m and z_ω are residual components due to $m(t)$ and $\omega(t)$ given by

$$
\begin{aligned}
z_m(t) &= \hat{C} e^{\bar{A}t} \hat{e}_{0_m} = \hat{C} e^{\bar{A}t} \psi_m m \\
z_\omega(t) &= \hat{C} e^{\bar{A}t} \hat{e}_{0_\omega} = \hat{C} e^{\bar{A}t} \psi_\omega \omega
\end{aligned}
$$

A detection filter fault enhancement problem may be stated as follows. Consider the residual components z_m and z_ω as elements of $\mathcal{L}_2[0, T)$ spaces where T is an observation *window*. Find a constant mapping q^T that maximizes the cost

$$
J = \left[\max_\omega \left(\frac{\|q^T z_m\|^2_{\mathcal{L}_2[0,T)}}{\|\omega\|^2_{\mathcal{L}_2(-\infty,0]}} \right) \right]^{-1} \left[\max_m \left(\frac{\|m\|^2_{\mathcal{L}_2(-\infty,0]}}{\|q^T z_m\|^2_{\mathcal{L}_2[0,T)}} \right) \right] \tag{c.20}
$$

Note that $q^T z_m$ and $q^T z_\omega$ are scalars. When maximized with respect to q^T, this cost penalizes large residual components due to a disturbance ω and small residual components due to a fault m.

The choice of the observation window T and the fault detection threshold is a design decision based on the functional form of the expected faults and disturbances. A detailed discussion is found in (Emami-Naeini et al. 1988). However, it is worthwhile to point out that a window of zero length, $T = 0$, is not practical. First, since faults and disturbances enter the residual directly through D_m and D_ω, it is not possible to distinguish a fault from a disturbance at any one point in time. Second, the operators that map signals $m(t)$ and $\omega(t) \in \mathcal{L}_2(-\infty, 0]$ to the respective residual components at time $t = 0$ are given by

$$\begin{aligned}\bar{\psi}_m : \mathcal{L}_2(-\infty, 0] \mapsto \mathcal{R}^m \quad &\triangleq \quad \bar{C}\psi_m m(t) + D_m m(0) \\ \bar{\psi}_\omega : \mathcal{L}_2(-\infty, 0] \mapsto \mathcal{R}^m \quad &\triangleq \quad \bar{C}\psi_\omega \omega(t) + D_\omega \omega(0)\end{aligned}$$

These operators are not bounded. For example, let

$$m_h(t) = \begin{cases} 1/\sqrt{h} & -h \leq t \leq 0 \\ 0 & t < -h \end{cases} \tag{c.21}$$

Then $m_h(t) \in \mathcal{L}_2(-\infty, 0]$ and $\|m_h\| = 1$ for all h but $\bar{\psi}_m m_h \to \infty$ as $h \to 0$. Hence, further restrictions on m and ω need to be made before a cost function such as the following could be used.

$$\frac{\|\bar{C}\psi_m m + D_m m(0)\|_{\mathcal{R}^m}}{\|\bar{C}\psi_\omega \omega + D_\omega \omega(0)\|_{\mathcal{R}^m}}$$

A well-known result (Doyle et al. 1989) is that for a given initial state $\bar{e}_{0_\omega}$, the smallest signal $\omega \in \mathcal{L}_2(-\infty, 0]$ that produces $\bar{e}_{0_\omega}$ has a norm given by

$$\inf_{\omega \in \mathcal{L}_2(-\infty, 0]} \{\|\omega\|^2 | \bar{e}(0) = \bar{e}_{0_\omega}\} = \bar{e}_{0_\omega}^T X_\omega^{-1} \bar{e}_{0_\omega} \tag{c.22}$$

where X_ω is the controllability grammian given as the solution to the steady-state Lyapunov equation

$$0 = \bar{A} X_\omega + X_\omega \bar{A}^T + \bar{B}\bar{B}^T$$

If q were known, an initial state $\bar{e}_{0_\omega}$ could be found by maximizing the ratio

$$\begin{aligned}J_\omega \quad &= \quad \sup_{\omega \in \mathcal{L}_2(-\infty, 0]} \frac{\|q^T z_\omega\|^2_{\mathcal{L}_2[0,T)}}{\|\omega\|^2_{\mathcal{L}_2(-\infty, 0]}} \\ &= \quad \max_{\bar{e}_{0_\omega} \neq 0} \frac{\bar{e}_{0_\omega}^T \left[\int_0^T e^{\bar{A}^T \tau} \bar{C}^T q q^T \bar{C} e^{\bar{A}\tau} d\tau\right] \bar{e}_{0_\omega}}{\bar{e}_{0_\omega}^T X_\omega^{-1} \bar{e}_{0_\omega}}\end{aligned}$$

This is solved as an eigenvalue problem

$$J_\omega = \lambda_{\max} \left[\int_0^T e^{\bar{A}^T \tau} \bar{C}^T q q^T \bar{C} e^{\bar{A}\tau} d\tau X_\omega\right]$$

where $\bar{e}_{0_\omega}$ is the eigenvector associated with the largest eigenvalue $\lambda_{\max}$. Note that in the case where $T = \infty$, J' is the Hankel norm of the transfer matrix.

Since q is not known, consider a worst case $\bar{e}_{0_\omega}$ as the eigenvector associated with

$$\left[\int_0^T e^{\bar{A}^T\tau}\bar{C}^T\bar{C}e^{\bar{A}\tau}d\tau X_\omega\right]\bar{e}_{0_\omega} = \lambda_{\omega_{\max}}\bar{e}_{0_\omega} \tag{c.23}$$

Similarly, a worst-case fault maximizes the ratio

$$\begin{aligned} J_m &= \sup_{m\in\mathcal{L}_2(-\infty,0]} \frac{\|m\|^2_{\mathcal{L}_2(-\infty,0]}}{\|q^T z_m\|^2_{\mathcal{L}_2[0,T)}} \\ &= \max_{\bar{e}_{0_m}\neq 0} \frac{\bar{e}_{0_m}^T X_m \bar{e}_{0_m}}{\bar{e}_{0_m}^T\left[\int_0^T e^{\bar{A}^T\tau}\bar{C}^T q q^T \bar{C} e^{\bar{A}\tau}d\tau\right]\bar{e}_{0_m}} \end{aligned}$$

where $\bar{e}_{0_m}$ is the eigenvector associated with

$$\left[\int_0^T e^{\bar{A}^T\tau}\bar{C}^T\bar{C}e^{\bar{A}\tau}d\tau X_m\right]\bar{e}_{0_m} = \lambda_{m_{\max}}\bar{e}_{0_m} \tag{c.24}$$

Now maximize (c.20) with respect to q using $\bar{e}_{0_\omega}$ and $\bar{e}_{0_m}$ from (c.23) and (c.24). This is solved as another eigenvalue problem.

$$J = \max_{q\neq 0}\frac{\|q^T z_m\|^2_{\mathcal{L}_2[0,T)}}{\|q^T z_\omega\|^2_{\mathcal{L}_2[0,T)}} = \lambda_{\max} \tag{c.25}$$

where

$$\begin{aligned} &\left(\bar{C}\int_0^T e^{\bar{A}\tau}\bar{e}_{0_m}\bar{e}_{0_m}^T e^{\bar{A}^T\tau}d\tau\bar{C}^T\right)^T q = \\ &\lambda_{\max}\left(\bar{C}\int_0^T e^{\bar{A}\tau}\bar{e}_{0_\omega}\bar{e}_{0_\omega}^T e^{\bar{A}^T\tau}d\tau\bar{C}^T\right)^T q \end{aligned} \tag{c.26}$$

Finally, the controllability gramians in (c.26) for the case $T = \infty$ may be found as solutions to a pair of steady-state Lyapunov equations. Let

$$\begin{aligned} X_{0_m} &= \int_0^T e^{\bar{A}\tau}\bar{e}_{0_m}\bar{e}_{0_m}^T e^{\bar{A}^T\tau}d\tau \\ X_{0_\omega} &= \int_0^T \bar{e}^{\bar{A}\tau}e_{0_\omega}\bar{e}_{0_\omega}^T e^{\bar{A}^T\tau}d\tau \end{aligned}$$

Then

$$\begin{aligned} 0 &= \bar{A}X_{0_m} + X_{0_m}\bar{A}^T + \bar{e}_{0_m}\bar{e}_{0_m}^T \\ 0 &= \bar{A}X_{0_\omega} + X_{0_\omega}\bar{A}^T + \bar{e}_{0_\omega}\bar{e}_{0_\omega}^T \end{aligned}$$

C.5 Application to an Aircraft Fault Detection System

This example considers a simpli¯ed aircraft fault detection ¯lter. The dynamics of an F16XL are linearized about a trimmed level °ight condition at 10,000 feet altitude and Mach 0.9. The ¯ve-state model includes longitudinal dynamics only, no lateral dynamics and no actuator dynamics. A ¯rst-order Dryden wind gust model is included.

$$
\begin{aligned}
\dot{x} &= Ax + B_\omega \omega + B_\delta \delta \\
y &= Cx + D\nu
\end{aligned}
$$

The states are

u	longitudinal body axis velocity (ft/sec)
w	normal body axis velocity (ft/sec)
q	pitch rate (deg/sec)
θ	pitch angle (deg)
w_g	wind gust (ft/sec)

the measurements are

q	pitch rate (deg/sec)
α	angle of attack (deg)
A_z	normal acceleration (ft/sec^2)
A_x	longitudinal acceleration (ft/sec^2)

the disturbances are

ω	wind gust (ft/sec)
ν_q	pitch rate sensor noise
ν_α	angle of attack sensor noise
$\nu_A z$	normal accelerometer sensor noise

$\nu_A x$ longitudinal accelerometer sensor noise

and the input is

δ elevon de°ection angle (deg)

All disturbances are zero-mean uncorrelated white noise processes with unit spectral density. The port and starboard elevon is modeled as a slaved system because only longitudinal dynamics are considered for this simple example. The elevon actuator dynamics are not included. The system matrices are

$$A = \begin{bmatrix} -0.0674 & 0.0430 & -0.8886 & -0.5587 & 0.0430 \\ 0.0205 & -1.4666 & 16.5800 & -0.0299 & -1.4666 \\ 0.1377 & -1.6788 & -0.6819 & 0 & -1.6788 \\ 0 & 0 & 1.0000 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1.1948 \end{bmatrix}$$

$$B_\omega = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1.57 \end{bmatrix}, \qquad B_\delta = \begin{bmatrix} -0.1672 \\ -1.5179 \\ -9.7842 \\ 0 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 1.0000 & 0 & 0 \\ 0 & 0.0591 & 0 & 1.0000 & 0 \\ 0.0139 & 1.0517 & 0.1485 & -0.0299 & 0 \\ -0.0677 & 0.0431 & 0.0171 & 0 & 0 \end{bmatrix}$$

$$D = \begin{bmatrix} 0.01 & 0 & 0 & 0 \\ 0 & 0.143 & 0 & 0 \\ 0 & 0 & 0.245 & 0 \\ 0 & 0 & 0 & 0.245 \end{bmatrix}$$

Now consider a fault detection system with two faults: a normal accelerometer sensor fault and an elevon fault. The normal accelerometer fault can be modeled as an additive term in the measurement equation

$$y = Cx + E_{Az}\mu_{Az} \qquad \text{where } E_{Az} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \tag{c.27}$$

and where μ_{Az} is an arbitrary time-varying real scalar. For the purpose of determining an associated detection space, the fault E_{Az} in (c.27) is equivalent to a two-dimensional fault F_{Az} (Douglas 1993)

$$\underset{\sim}{x} = Ax + F_{Az}m_{Az} \qquad \text{with } F_{Az} = [F^1_{Az}, F^2_{Az}]$$

where the directions F^1_{Az} and F^2_{Az} are given by

$$\begin{aligned} E_{Az} &= CF^1_{Az} \\ F^2_{Az} &= AF^1_{Az} \end{aligned}$$

so that

$$F_{Az} = \begin{bmatrix} 0 & 0.9986 \\ 0 & 0.0534 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \end{bmatrix}$$

The elevon fault is given simply as $F_\delta = B_\delta$. Since CF^1_{Az}, CF^2_{Az} and CF_δ are all nonzero and since none of the triples (C, A, F^1_{Az}), (C, A, F^2_{Az}), (C, A, F_δ) have invariant zeros, the minimal unobservability subspaces for the faults are given by the fault directions themselves, that is, $\mathcal{T}^{1*}_{Az} = \text{Span}\, F^1_{Az}$, $\mathcal{T}^{2*}_{Az} = \text{Span}\, F^2_{Az}$ and $\mathcal{T}^*_\delta = \text{Span}\, F_\delta$. The faults are mutually detectable so there are no constraints on the spectrum of the detection ¯lter.

The ¯rst step toward ¯nding a fault detection ¯lter gain is to ¯nd $\hat{L}$ as in (c.17h). This gain forms an observer with the correct detection space structure but without regard to stability or any performance considerations.

$$\hat{L} = -\sum_{i=1}^{q} AW_i\hat{F}_i\hat{H}_i$$

Considering the two-dimensional normal accelerometer sensor fault as a pair of output separable faults, the $\hat{F}_i$ are identity matrices and the W_i are just the fault directions themselves. To ¯nd the $\hat{H}_i$, let $F = [F_{Az}, F_\delta]$ and form the left inverse of CF as $(CF)^{-\ell} = (F^TC^TCF)^{-1}F^TC^T$. Now take $\hat{H}_{Az}$ as the ¯rst two rows of $(CF)^{-\ell}$ and $\hat{H}_\delta$ as the third row. Finally $\hat{L} = -AF_{Az}\hat{H}_{Az} - AF_\delta\hat{H}_\delta$ and all components needed to apply Proposition C.7 are now given.

Application of Proposition C.7 involves solving a modi¯ed algebraic Riccati equation. One approach which has achieved practical success is to form a modi¯ed *di®erential* Riccati equation and to numerically integrate until a steady state is reached. An initial condition for the integration is chosen by solving the algebraic Riccati equation found by truncating the modifying quadratic term. Choosing an $\mathcal{H}_\infty$ bounding parameter γ = 1.2 results in a ¯lter with eigenvalues $-29.4629, -1.6062, -0.4351, -0.0032$ and -1.1013.

Figure c.1 shows the maximum singular values in decibels of two fault detection ¯lter transfer matrices. One is from the wind disturbance and sensor noise to the residual which isolates a normal accelerometer fault. The other is from the normal accelerometer sensor to the same residual. A third transfer matrix, one from the elevon de°ection is zero, as it should be, and is not shown. Figure c.2 shows the maximum singular values of transfer matrices to the elevon fault residual. Here the transfer matrix from the normal accelerometer sensor is zero and is not shown. In both ¯gures, the residual is scaled so that the DC gain of the disturbance component is 0 db. Both faults have been scaled by two to emphasize that fault detection in the presence of disturbances resolves to a threshold selection problem.

Note that in the case of the elevon fault, both the residual and the detection space are one-dimensional so the associated ¯lter eigenvector is ¯xed. There is no way to increase the residual component due to the fault without at the same time increasing the component due to the noise.

This is not the case for the normal accelerometer residual since it is two dimensional. A fault enhancing residual direction is found from (c.26) as $q_i^T = [-0.126, -0.992]$. The singular value frequency responses for the improved residual are also shown in Figure c.1. Disturbance reduction is seen mainly at frequencies above 1 rad/sec. A modest increase in the fault signal is seen at all frequencies.

Figures c.3 and c.4 show residual histories where white noise is applied to the wind gust model and the sensors. Figure c.3 shows the normal accelerometer residual history when a 2 ft/sec^2 bias is added to the accelerometer signal after one second. Figure c.4 shows the elevon residual history when a 2 degree bias is added to the elevon de°ection after one second. Clearly, in both cases, a hard fault is detectable with an appropriate threshold (Emami-Naeini et al. 1988).

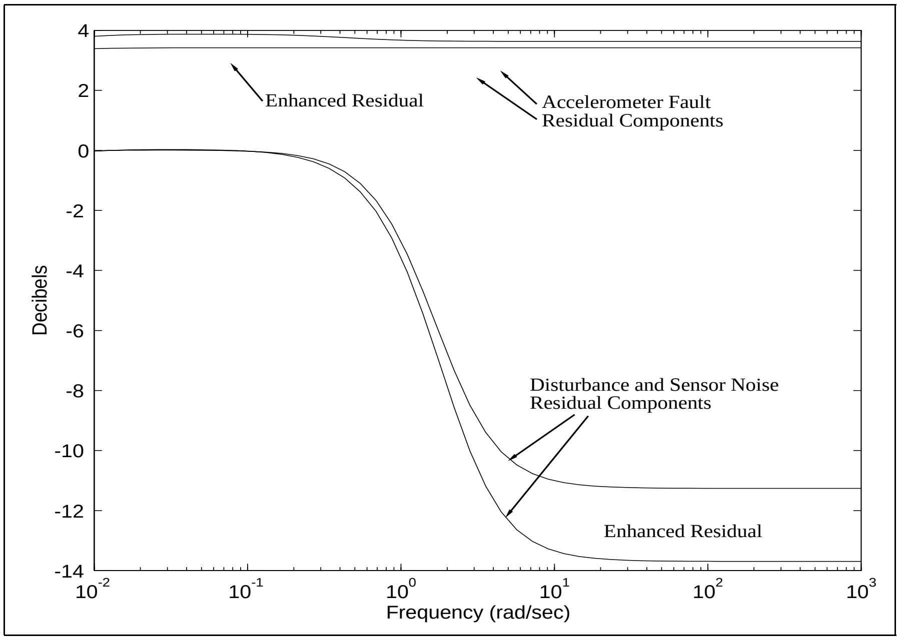

Figure c.1: Magnitude of transfer functions to the normal accelerometer fault isolation residual.

C.6 Conclusions

A stable and $\mathcal{H}_\infty$ bounded detection ¯lter is found by solving a modi¯ed algebraic Riccati equation (c.18). This equation does not have an associated Hamiltonian and its properties are not well known; however, in (Veillette et al. 1992), a similar equation appears in the context of decentralized system control and there it is reported that a solution when it exists can usually be found by iterative, numerical means. Future work will focus on ¯nding necessary and su±cient conditions for (c.18) to have a solution.

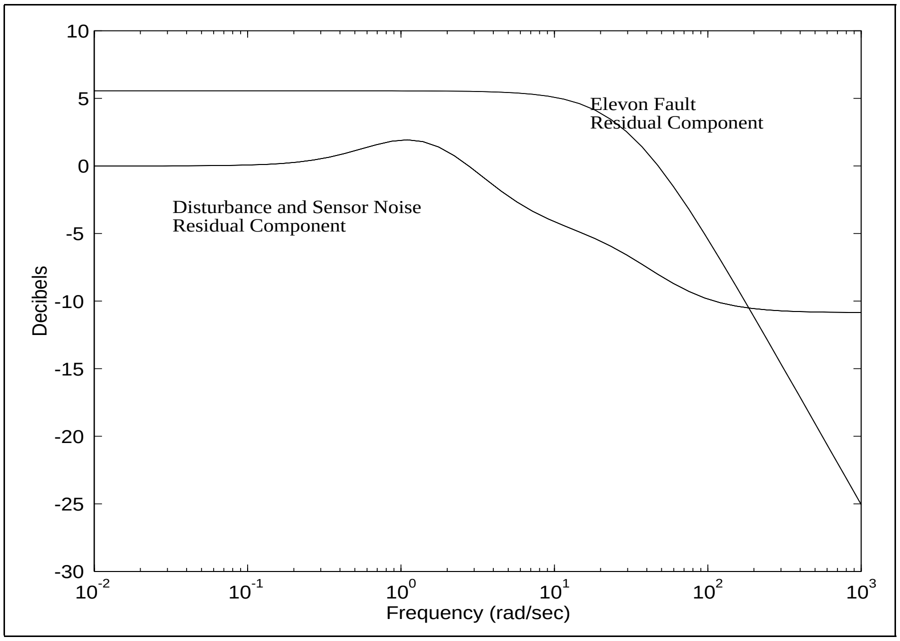

Figure c.2: Magnitude of transfer functions to the elevon fault isolation residual.

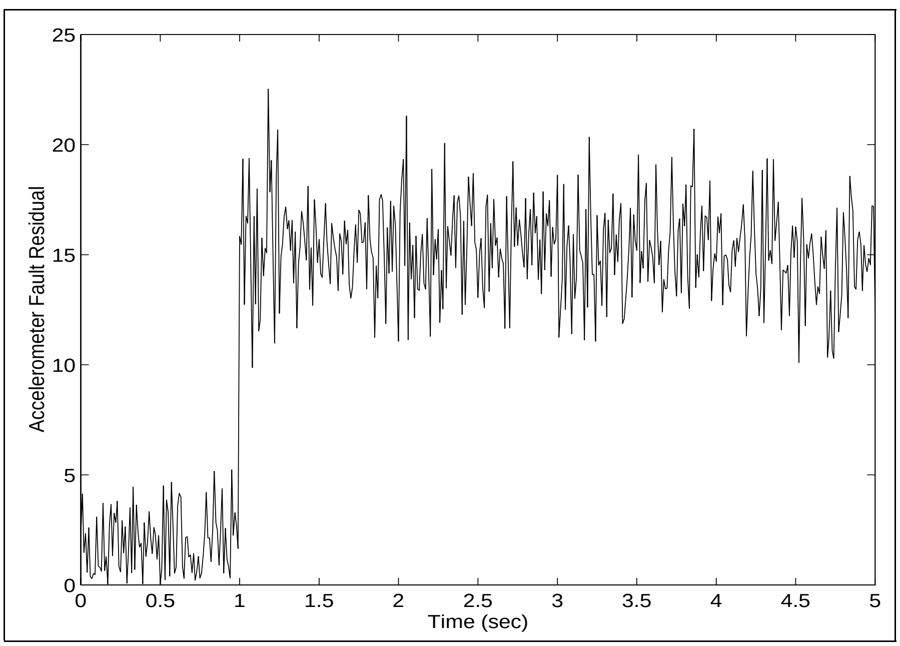

Figure c.3: Normal accelerometer fault isolation residual. $2\frac{\text{ft}}{\text{sec}^2}$ accelerometer fault occurs at $t = 1$ sec.

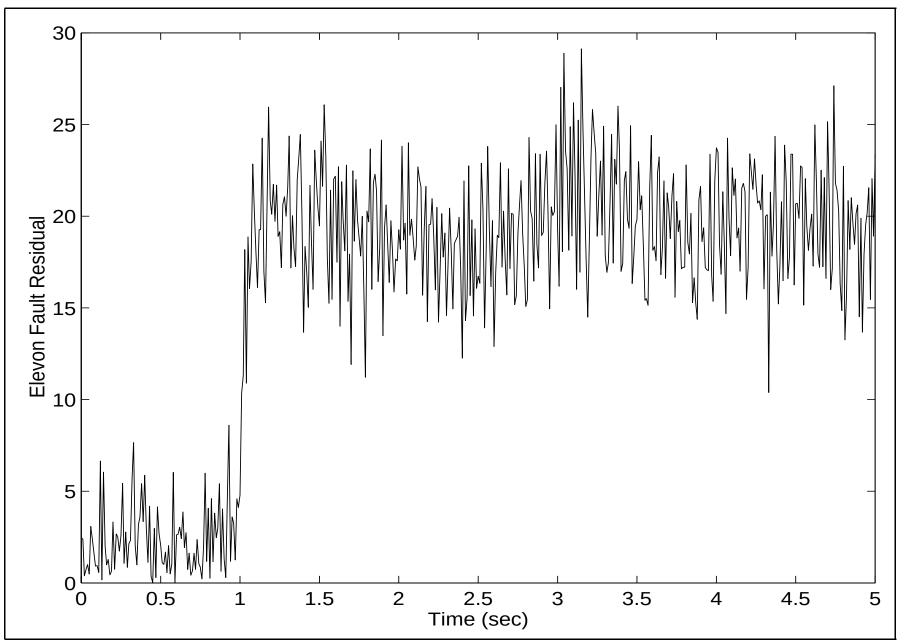

Figure c.4: Elevon fault isolation residual. 2 degree elevon fault occurs at t=1 sec.

Appendix D

Vehicle Linear Model Data

The fourteen-order linear system matrices used for actuator fault detection ¯lter design are:

$$A = \begin{bmatrix} -22.42 & -0.12 & 0 & 0 & 0 & 0 & 0 & \dots \\ 306.69 & -29.75 & 331.11 & -1.17 & -196.62 & -2278.82 & 56.77 & \dots \\ 0 & 0.06 & -0.68 & 0.06 & 0.51 & 5.95 & -0.18 & \dots \\ 0 & -0.00 & -0.07 & -9.03 & -0.33 & -2.84 & -18.33 & \dots \\ 0 & -0.00 & 0.02 & -0.00 & -3.55 & -41.21 & 0.03 & \dots \\ 0 & 0 & 0 & 0 & 1.00 & 0 & 0 & \dots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ 0 & -0.00 & -0.03 & -9.75 & -0.41 & -1.62 & -60.48 & \dots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ 0 & -0.02 & 0.18 & -0.01 & -0.59 & -7.05 & 0.32 & \dots \\ 0 & -0.00 & 0.01 & 0.33 & 0.05 & 0.53 & -0.17 & \dots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \end{bmatrix}$$

$$\begin{matrix} \dots & 0 & 0 & 0 & 0 & 2.35 & 0 & 0 \\ \dots & 2.92 & -3169.84 & -273.50 & 2.99 & 0 & -0.07 & 0 \\ \dots & -0.00 & 15.84 & 1.40 & -0.34 & 0 & -0.00 & -1.09 \\ \dots & -1.83 & -0.79 & -0.10 & -23.98 & 0 & -0.00 & 123.72 \\ \dots & -0.00 & -9.21 & -0.81 & 0.01 & 0 & 0.00 & 0.04 \\ \dots & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \dots & 1.00 & 0 & 0 & 0 & 0 & 0 & 0 \\ \dots & -6.05 & -0.09 & -0.10 & 0.91 & 0 & -0.00 & 134.16 \\ \dots & 0 & 0 & 1.00 & 0 & 0 & 0 & 0 \\ \dots & -0.01 & -40.09 & -3.59 & 0.06 & 0 & 0.00 & 0.39 \\ \dots & -0.01 & 2.03 & 0.19 & -5.62 & 0 & 0.00 & 49.17 \\ \dots & 0 & 0 & 0 & 0 & -90.91 & 0 & 0 \\ \dots & 0 & 0 & 0 & 0 & 0 & -1.25 & 0 \\ \dots & 0 & 0 & 0 & 0 & 0 & 0 & -80.00 \end{matrix}\Bigg] \tag{d.1a}$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 90.91 & 0 & 0 \\ 0 & 1.25 & 0 \\ 0 & 0 & 80.00 \end{bmatrix} \tag{d.1b}$$

$$C = \Bigg[\begin{matrix} 1.00 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ 0 & 1.00 & 0 & 0 & 0 & 0 & 0 & \dots \\ 0 & 0.06 & -0.68 & 0.06 & 0.51 & 5.95 & -0.18 & \dots \\ 0 & -0.00 & -0.07 & -9.03 & -0.33 & -2.84 & -18.33 & \dots \\ 0 & -0.00 & 0.02 & -0.00 & -3.55 & -41.21 & 0.03 & \dots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ 0 & 0 & 3.54 & -0.01 & -2.24 & -26.36 & -19.11 & \dots \\ 0 & 0 & 3.58 & -0.01 & -2.30 & -26.33 & 19.09 & \dots \\ 0 & 0.04 & 3.02 & -0.01 & -1.76 & -20.06 & -16.88 & \dots \\ 0 & 0.03 & 3.12 & -0.01 & -1.89 & -22.20 & 17.93 & \dots \end{matrix}$$

$$\begin{matrix} \dots & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \dots & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \dots & -0.00 & 15.84 & 1.40 & -0.34 & 0 & -0.00 & -1.09 \\ \dots & -1.83 & -0.79 & -0.10 & -23.98 & 0 & -0.00 & 123.72 \\ \dots & -0.00 & -9.21 & -0.81 & 0.01 & 0 & 0.00 & 0.04 \\ \dots & 1.00 & 0 & 0 & 0 & 0 & 0 & 0 \\ \dots & 0 & 0 & 1.00 & 0 & 0 & 0 & 0 \\ \dots & 0 & 0 & 0 & 1.00 & 0 & 0 & 0 \\ \dots & -1.63 & 29.89 & 2.54 & -2.56 & 0 & -0.00 & -0.00 \\ \dots & 1.67 & 29.86 & 2.61 & 2.24 & 0 & -0.00 & -0.00 \\ \dots & -1.47 & -27.90 & -2.45 & -2.50 & 0 & -0.00 & 0 \\ \dots & 1.53 & -30.88 & -2.62 & 2.56 & 0 & -0.00 & 0 \end{matrix} \Bigg] \quad \text{(d.1c)}$$

$$D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \text{(d.1d)}$$

The twelveth-order linear system matrices used for sensor fault detection ¯lter design are:

$$A = \Bigg[\begin{matrix} -22.42 & -0.12 & 0 & 0 & 0 & 0 & \dots \\ 306.69 & -29.75 & 331.11 & -1.17 & -196.62 & -2278.82 & \dots \\ 0 & 0.06 & -0.68 & 0.06 & 0.51 & 5.95 & \dots \\ 0 & -0.00 & -0.07 & -9.03 & -0.33 & -2.84 & \dots \\ 0 & -0.00 & 0.02 & -0.00 & -3.55 & -41.21 & \dots \\ 0 & 0 & 0 & 0 & 1.00 & 0 & \dots \\ 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ 0 & -0.00 & -0.03 & -9.75 & -0.41 & -1.62 & \dots \\ 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ 0 & -0.02 & 0.18 & -0.01 & -0.59 & -7.05 & \dots \\ 0 & -0.00 & 0.01 & 0.33 & 0.05 & 0.53 & \dots \\ 0 & 0 & 0 & 0 & 0 & 0 & \dots \end{matrix}$$

$$
\begin{matrix}
\cdots & 0 & 0 & 0 & 0 & 0 & 0 \\
\cdots & 56.77 & 2.92 & -3169.84 & -273.50 & 2.99 & -0.07 \\
\cdots & -0.18 & -0.00 & 15.84 & 1.40 & -0.34 & -0.00 \\
\cdots & -18.33 & -1.83 & -0.79 & -0.10 & -23.98 & -0.00 \\
\cdots & 0.03 & -0.00 & -9.21 & -0.81 & 0.01 & 0.00 \\
\cdots & 0 & 0 & 0 & 0 & 0 & 0 \\
\cdots & 0 & 1.00 & 0 & 0 & 0 & 0 \\
\cdots & -60.48 & -6.05 & -0.09 & -0.10 & 0.91 & -0.00 \\
\cdots & 0 & 0 & 0 & 1.00 & 0 & 0 \\
\cdots & 0.32 & -0.01 & -40.09 & -3.59 & 0.06 & 0.00 \\
\cdots & -0.17 & -0.01 & 2.03 & 0.19 & -5.62 & 0.00 \\
\cdots & 0 & 0 & 0 & 0 & 0 & -1.25
\end{matrix}\Bigg] \tag{d.2a}
$$

$$
B = \begin{bmatrix}
2.35 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & -1.09 \\
0 & 0 & 123.72 \\
0 & 0 & 0.04 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 134.16 \\
0 & 0 & 0 \\
0 & 0 & 0.39 \\
0 & 0 & 49.17 \\
0 & 1.25 & 0
\end{bmatrix} \tag{d.2b}
$$

$$
C = \Bigg[\begin{matrix}
1.00 & 0 & 0 & 0 & 0 & 0 & \cdots \\
0 & 1.00 & 0 & 0 & 0 & 0 & \cdots \\
0 & 0.06 & -0.68 & 0.06 & 0.51 & 5.95 & \cdots \\
0 & -0.00 & -0.07 & -9.03 & -0.33 & -2.84 & \cdots \\
0 & -0.00 & 0.02 & -0.00 & -3.55 & -41.21 & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
0 & 0 & 3.54 & -0.01 & -2.24 & -26.36 & \cdots \\
0 & 0 & 3.58 & -0.01 & -2.30 & -26.33 & \cdots \\
0 & 0.04 & 3.02 & -0.01 & -1.76 & -20.06 & \cdots \\
0 & 0.03 & 3.12 & -0.01 & -1.89 & -22.20 & \cdots
\end{matrix}
$$

$$\begin{matrix} \cdots & 0 & 0 & 0 & 0 & 0 & 0 \\ \cdots & 0 & 0 & 0 & 0 & 0 & 0 \\ \cdots & -0.18 & -0.00 & 15.84 & 1.40 & -0.34 & -0.00 \\ \cdots & -18.33 & -1.83 & -0.79 & -0.10 & -23.98 & -0.00 \\ \cdots & 0.03 & -0.00 & -9.21 & -0.81 & 0.01 & 0.00 \\ \cdots & 0 & 1.00 & 0 & 0 & 0 & 0 \\ \cdots & 0 & 0 & 0 & 1.00 & 0 & 0 \\ \cdots & 0 & 0 & 0 & 0 & 1.00 & 0 \\ \cdots & -19.11 & -1.63 & 29.89 & 2.54 & -2.56 & -0.00 \\ \cdots & 19.09 & 1.67 & 29.86 & 2.61 & 2.24 & -0.00 \\ \cdots & -16.88 & -1.47 & -27.90 & -2.45 & -2.50 & -0.00 \\ \cdots & 17.93 & 1.53 & -30.88 & -2.62 & 2.56 & -0.00 \end{matrix} \Bigg] \tag{d.2c}$$

$$D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1.09 \\ 0 & 0 & 123.72 \\ 0 & 0 & 0.04 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -0.00 \\ 0 & 0 & -0.00 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{d.2d}$$

Appendix E
Fault Detection Filter Design Data

This appendix collects data associated with fault detection ¯lter designs of Section 4. Section E.1 has the data for a fault detection ¯lter design for fault group three, the four wheel speed sensors. Section E.2 has the data for a fault detection ¯lter design for fault group four, the four wheel speed sensors.

E.1 Design Data for Fault Group Three

This appendix presents data associated with the fault detection ¯lter design for fault group three, the four wheel speed sensors. The design details and a discussion are in Section 4.2.1.

For each eigenvalue $\lambda_{i_j} \in$ ¤$_i$, the left eigenvectors v_{i_j} generally are not unique and must be chosen from a subspace as $v_{i_j} \in V_{i_j}$ where V_{i_j} is found by solving (4.2)

$$\begin{bmatrix} A^T - \lambda_{i_j} I & C^T \\ \hat{F}_i^T & 0 \end{bmatrix} \begin{bmatrix} V_{i_j} \\ W_{i_j} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

There are twelve V_{i_j} associated with twelve eigenvalues. Only two V_{i_j}, the two associated with the front left wheel speed sensor fault, are shown here.

$$V_{y_{w_{fl1}}} = \begin{bmatrix} 0.0147 & -0.0095 & 0.0089 & -0.0572 & 0.1287 & 0.9898 \\ 0.0009 & -0.0006 & 0.0005 & -0.0033 & 0.0041 & -0.0008 \\ -0.0680 & -0.1045 & -0.2833 & -0.3041 & -0.3300 & 0.0279 \\ -0.1436 & -0.2173 & -0.0126 & -0.5593 & -0.6139 & 0.0476 \\ -0.0010 & -0.0036 & -0.0021 & -0.0121 & -0.0036 & -0.0002 \\ 0.0100 & -0.0087 & 0.0032 & -0.0526 & 0.0543 & -0.0104 \\ -0.1343 & 0.0826 & 0.9460 & -0.1194 & -0.0980 & 0.0001 \\ 0.0176 & -0.0340 & 0.0090 & -0.1649 & 0.1749 & -0.0329 \\ 0.4216 & -0.8713 & 0.1488 & 0.1971 & -0.0085 & -0.0035 \\ -0.0751 & 0.1252 & 0.0007 & 0.6939 & -0.6443 & 0.1262 \\ 0.8790 & 0.3985 & 0.0489 & -0.1635 & -0.1980 & 0.0066 \\ -0.0001 & -0.0000 & -0.0003 & -0.0003 & -0.0003 & 0.0000 \end{bmatrix}$$

$$V_{y_{w_{fl2}}} = \begin{bmatrix} -0.0077 & 0.0039 & -0.0135 & 0.4552 & -0.8901 & 0.0166 \\ -0.0003 & 0.0001 & -0.0005 & 0.0042 & 0.0021 & -0.0027 \\ 0.0098 & 0.0673 & 0.2871 & 0.1475 & 0.0793 & 0.4276 \\ -0.1385 & 0.0536 & 0.0486 & 0.2694 & 0.1533 & 0.7997 \\ -0.0014 & 0.0012 & 0.0025 & 0.0091 & 0.0048 & 0.0079 \\ -0.0031 & 0.0013 & -0.0026 & 0.0627 & 0.0315 & -0.0326 \\ -0.3370 & -0.3002 & -0.8473 & 0.0387 & 0.0369 & 0.1386 \\ 0.0048 & 0.0010 & -0.0121 & 0.2002 & 0.1006 & -0.1002 \\ 0.0897 & 0.9304 & -0.3524 & -0.0069 & 0.0052 & 0.0030 \\ 0.0020 & -0.0034 & 0.0008 & -0.8059 & -0.4057 & 0.3461 \\ -0.9268 & 0.1918 & 0.2698 & -0.0581 & -0.0280 & -0.1649 \\ 0.0000 & -0.0000 & 0.0003 & 0.0001 & 0.0001 & 0.0004 \end{bmatrix}$$

The left eigenvectors are chosen from $v_{i_j} \in V_{i_j}$ as the set with the greatest degree of linear independence. The matrix of assigned left eigenvectors $\mathcal{V}$ is

$$\mathcal{V} = \left[v_{0_1}, v_{0_2}, v_{0_3}, v_{0_4}, v_{y_{w_{fl1}}}, v_{y_{w_{fl2}}}, v_{y_{w_{fr1}}}, v_{y_{w_{fr2}}}, v_{y_{w_{rl1}}}, v_{y_{w_{rl2}}}, v_{y_{w_{rr1}}}, v_{y_{w_{rr2}}}\right]$$

$$
= \begin{bmatrix}
0.2991 & 0.0000 & -0.0000 & -0.9542 & -0.0000 & 0.0000 & \dots \\
-0.0001 & -0.0002 & 0.0000 & -0.0000 & -0.0017 & 0.0052 & \dots \\
0.0119 & 0.0434 & -0.0131 & 0.0037 & -0.4802 & -0.2551 & \dots \\
0.1401 & -0.1312 & -0.3954 & 0.0439 & -0.6638 & -0.3524 & \dots \\
0.0013 & -0.0009 & -0.0037 & 0.0004 & -0.0126 & 0.0008 & \dots \\
-0.0012 & 0.0002 & -0.0005 & -0.0004 & -0.0316 & 0.0708 & \dots \\
0.2894 & -0.2470 & -0.8074 & 0.0907 & 0.3290 & 0.1744 & \dots \\
-0.0175 & 0.0031 & -0.0078 & -0.0055 & -0.1006 & 0.2234 & \dots \\
-0.2274 & -0.9478 & 0.2053 & -0.0713 & -0.0254 & 0.0095 & \dots \\
0.0446 & -0.0280 & 0.0265 & 0.0140 & 0.4558 & -0.8500 & \dots \\
0.8678 & -0.1439 & 0.3855 & 0.2720 & -0.0280 & 0.0530 & \dots \\
0.0000 & 0.0001 & -0.0000 & 0.0000 & -0.0005 & -0.0002 & \dots
\end{bmatrix.
$$

$$
\begin{bmatrix}
\dots & 0.0000 & -0.0000 & -0.0000 & -0.0000 & 0.0000 & -0.0000 \\
\dots & -0.0009 & -0.0054 & -0.0212 & -0.0007 & 0.0219 & 0.0008 \\
\dots & 0.5079 & -0.0023 & 0.0003 & -0.2479 & 0.0140 & -0.2304 \\
\dots & -0.7729 & 0.0055 & 0.0053 & -0.8662 & -0.0570 & 0.8683 \\
\dots & -0.0088 & 0.0039 & -0.0102 & -0.0033 & -0.0105 & 0.0048 \\
\dots & 0.0031 & 0.0444 & -0.1179 & -0.0009 & -0.1176 & -0.0080 \\
\dots & 0.3784 & -0.0025 & -0.0026 & 0.4275 & 0.0280 & -0.4265 \\
\dots & 0.0176 & 0.2375 & -0.9924 & -0.0068 & -0.9902 & -0.0660 \\
\dots & 0.0316 & -0.0205 & -0.0006 & -0.0013 & 0.0006 & -0.0180 \\
\dots & 0.0028 & 0.9688 & 0.0146 & -0.0738 & 0.0194 & -0.0789 \\
\dots & 0.0002 & -0.0503 & -0.0211 & 0.0041 & -0.0213 & 0.0031 \\
\dots & 0.0005 & 0.0000 & -0.0000 & 0.0005 & -0.0000 & 0.0006
\end{bmatrix}
$$

where

$$
\begin{array}{rclrcl}
v_{0_1} & \in & V_{0_1} & v_{0_2} & \in & V_{0_2} \\
v_{0_3} & \in & V_{0_3} & v_{0_4} & \in & V_{0_4} \\
v_{y_{w_{fl1}}} & \in & V_{y_{w_{fl1}}} & v_{y_{w_{fl2}}} & \in & V_{y_{w_{fl2}}} \\
v_{y_{w_{fr1}}} & \in & V_{y_{w_{fr1}}} & v_{y_{w_{fr2}}} & \in & V_{y_{w_{fr2}}} \\
v_{y_{w_{rl1}}} & \in & V_{y_{w_{rl1}}} & v_{y_{w_{rl2}}} & \in & V_{y_{w_{rl2}}} \\
v_{y_{w_{rr1}}} & \in & V_{y_{w_{rr1}}} & v_{y_{w_{rr2}}} & \in & V_{y_{w_{rr2}}}
\end{array}
$$

The matrix $\bar{W}$ associated with the left eigenvectors $\bar{V}$ is

$$
\bar{W} = \left[w_{0_1}, w_{0_2}, w_{0_3}, w_{0_4}, w_{y_{w_{fl1}}}, w_{y_{w_{fl2}}}, w_{y_{w_{fr1}}}, w_{y_{w_{fr2}}}, w_{y_{w_{rl1}}}, w_{y_{w_{rl2}}}, w_{y_{w_{rr1}}}, w_{y_{w_{rr2}}} \right]
$$

$$
= \left[\begin{array}{rrrrrrc}
3.1340 & 0.0475 & -0.0087 & -7.0773 & 0.5261 & -1.5937 & \dots \\
0.0240 & -0.0486 & 0.0116 & -0.1165 & -0.0064 & 0.1703 & \dots \\
0.1726 & 0.7200 & -0.1664 & 0.0690 & -0.0379 & -1.2071 & \dots \\
0.0977 & -0.0613 & -0.1967 & 0.0453 & 0.4735 & -0.3263 & \dots \\
0.0161 & 0.1358 & 0.0176 & 0.0061 & -0.4006 & -0.3341 & \dots \\
0.2604 & -0.1254 & -0.2117 & 0.1250 & 0.0587 & -2.4525 & \dots \\
-0.5476 & 0.2188 & -0.3407 & -0.2320 & -1.7301 & 9.2842 & \dots \\
0.2466 & -3.2948 & -17.4840 & -0.3811 & -3.2417 & -17.0804 & \dots \\
-0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.5766 & 0.0589 & \dots \\
-0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & 0.0000 & \dots \\
0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 & \dots \\
0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 & \dots
\end{array}\right.
$$

$$
\left.\begin{array}{crrrrrr}
\dots & 0.2793 & 1.6517 & 6.4910 & 0.2300 & -6.7146 & -0.2571 \\
\dots & -0.0784 & -0.1105 & -0.5915 & -0.1546 & 0.5887 & -0.1013 \\
\dots & 0.4392 & 0.3352 & -0.0791 & 1.7645 & -0.1720 & 1.7127 \\
\dots & 0.3324 & -0.2558 & 1.0693 & 0.0188 & 1.0810 & -0.0187 \\
\dots & 0.5166 & -0.1356 & -0.0141 & -0.2256 & 0.0014 & -0.3392 \\
\dots & -0.3991 & -2.2347 & 5.3986 & -0.2538 & 6.3527 & 0.8144 \\
\dots & -0.0502 & -9.5742 & -0.0536 & 0.6232 & -0.0978 & 0.5578 \\
\dots & -9.2347 & -7.0837 & 32.5183 & -16.9163 & 31.4442 & 18.7676 \\
\dots & -0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 & -0.0000 \\
\dots & -0.4555 & 0.5162 & 0.0000 & -0.0000 & -0.0000 & -0.0000 \\
\dots & 0.0000 & -0.0000 & 2.3178 & 1.1515 & -0.0000 & 0.0000 \\
\dots & 0.0000 & -0.0000 & 0.0000 & 0.0000 & -2.3751 & 0.8506
\end{array}\right]
$$

The fault detection filter gain L is found from (4.4), $\bar{V}^T L = \bar{W}^T$ and is

$$
L = \left[\begin{array}{rrrrrrc}
7.69 & 0.12 & -0.01 & -0.01 & -0.00 & -0.04 & \dots \\
-306.69 & 27.24 & 0.22 & 0.19 & -0.13 & 23.12 & \dots \\
0.01 & 0.08 & -1.43 & 0.09 & 1.04 & 0.42 & \dots \\
0.13 & 0.01 & -0.16 & 0.31 & -0.11 & 0.18 & \dots \\
0.00 & -0.51 & 17.06 & -44.36 & 4.24 & 46.22 & \dots \\
-0.00 & 0.58 & -13.13 & 0.06 & -4.21 & -28.33 & \dots \\
0.26 & -0.00 & 0.01 & 0.30 & -0.02 & 0.06 & \dots \\
-0.02 & -0.05 & 1.45 & -0.63 & 0.47 & -3.05 & \dots \\
-0.20 & 0.02 & -0.45 & -0.01 & -0.09 & 0.06 & \dots \\
0.04 & 0.03 & 0.51 & 0.07 & -0.08 & -0.31 & \dots \\
0.78 & -0.00 & -0.02 & 0.01 & 0.01 & 0.12 & \dots \\
0.00 & -186.37 & 2481.31 & 64.36 & -80.30 & 148.64 & \dots
\end{array}\right.
$$

$$
\left.\begin{matrix}
\dots & 0.06 & 0.44 & 0.00 & 0.00 & -0.00 & -0.00 \\
\dots & -0.26 & 4.45 & -0.00 & 0.00 & -53.97 & -53.73 \\
\dots & -1.74 & 1.70 & -0.36 & -0.34 & -0.71 & -0.59 \\
\dots & 1.26 & 28.02 & 0.26 & -0.22 & -1.00 & 0.85 \\
\dots & -216.65 & -867.22 & -43.03 & 39.17 & 80.36 & -68.11 \\
\dots & 11.29 & -1.91 & 0.32 & 4.51 & 1.99 & -2.63 \\
\dots & 0.41 & 8.69 & 0.07 & -0.07 & 0.13 & -0.11 \\
\dots & 0.80 & -23.48 & 0.40 & -0.93 & -2.25 & 2.15 \\
\dots & -0.17 & -0.56 & -0.04 & -0.04 & 0.08 & 0.07 \\
\dots & -9.75 & 1.55 & 0.06 & 0.39 & -0.16 & -0.43 \\
\dots & -0.14 & -6.85 & 0.00 & -0.03 & -0.02 & 0.06 \\
\dots & -580.26 & 2141.09 & -62.03 & -164.78 & 466.76 & 651.96
\end{matrix}\right] \qquad \text{(e.1)}
$$

The output projection matrices $\hat{H}_{y_{w_{fl}}}$, $\hat{H}_{y_{w_{fr}}}$, $\hat{H}_{y_{w_{rl}}}$ and $\hat{H}_{y_{w_{rr}}}$ are as follows.

$$
\hat{H}_{y_{w_{fl}}} = \left[\begin{matrix}
1.0000 & 0.0000 & -0.0000 & 0.0000 & 0.0000 & -0.0000 & \dots \\
0.0000 & 0.0043 & -0.0631 & -0.0027 & 0.0007 & -0.0021 & \dots \\
-0.0000 & -0.0631 & 0.9757 & 0.0674 & 0.0867 & -0.0313 & \dots \\
0.0000 & -0.0027 & 0.0674 & 0.3023 & 0.1353 & 0.3567 & \dots \\
0.0000 & 0.0007 & 0.0867 & 0.1353 & 0.2517 & -0.0936 & \dots \\
-0.0000 & -0.0021 & -0.0313 & 0.3567 & -0.0936 & 0.8168 & \dots \\
-0.0000 & -0.0061 & -0.0085 & 0.0901 & -0.0223 & -0.0463 & \dots \\
-0.0000 & 0.0003 & -0.0002 & 0.0021 & -0.0006 & -0.0011 & \dots \\
0.0000 & -0.0155 & 0.0815 & -0.2296 & -0.3915 & 0.1026 & \dots \\
-0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & \dots \\
-0.0000 & -0.0000 & 0.0000 & 0.0000 & 0.0000 & -0.0000 & \dots \\
0.0000 & -0.0000 & 0.0000 & 0.0000 & 0.0000 & -0.0000 & \dots
\end{matrix}\right.
$$

$$
\left.\begin{matrix}
\dots & -0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 & 0.0000 \\
\dots & -0.0061 & 0.0003 & -0.0155 & 0.0000 & -0.0000 & -0.0000 \\
\dots & -0.0085 & -0.0002 & 0.0815 & -0.0000 & 0.0000 & 0.0000 \\
\dots & 0.0901 & 0.0021 & -0.2296 & 0.0000 & 0.0000 & 0.0000 \\
\dots & -0.0223 & -0.0006 & -0.3915 & -0.0000 & 0.0000 & 0.0000 \\
\dots & -0.0463 & -0.0011 & 0.1026 & 0.0000 & -0.0000 & -0.0000 \\
\dots & 0.9883 & -0.0003 & 0.0267 & 0.0000 & -0.0000 & -0.0000 \\
\dots & -0.0003 & 1.0000 & 0.0006 & -0.0000 & 0.0000 & 0.0000 \\
\dots & 0.0267 & 0.0006 & 0.6610 & -0.0000 & 0.0000 & -0.0000 \\
\dots & 0.0000 & -0.0000 & -0.0000 & -0.0000 & 0.0000 & -0.0000 \\
\dots & -0.0000 & 0.0000 & 0.0000 & 0.0000 & -0.0000 & 0.0000 \\
\dots & -0.0000 & 0.0000 & -0.0000 & -0.0000 & 0.0000 & -0.0000
\end{matrix}\right] \qquad \text{(e.2a)}
$$

$$
\begin{aligned}
\hat{H}_{y_{w_{fr}}} &= \left[\begin{matrix}
1.0000 & 0.0000 & 0.0000 & 0.0000 & -0.0000 & -0.0000 & \dots \\
0.0000 & 0.0042 & -0.0623 & -0.0010 & 0.0023 & -0.0021 & \dots \\
0.0000 & -0.0623 & 0.9773 & -0.0532 & 0.0921 & 0.0225 & \dots \\
0.0000 & -0.0010 & -0.0532 & 0.2016 & -0.1514 & 0.2985 & \dots \\
-0.0000 & 0.0023 & 0.0921 & -0.1514 & 0.2915 & 0.0383 & \dots \\
-0.0000 & -0.0021 & 0.0225 & 0.2985 & 0.0383 & 0.8879 & \dots \\
0.0000 & -0.0050 & -0.0057 & -0.0729 & -0.0101 & 0.0273 & \dots \\
-0.0000 & 0.0002 & 0.0008 & 0.0102 & 0.0014 & -0.0038 & \dots \\
0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 & \dots \\
-0.0000 & -0.0174 & 0.0806 & 0.2017 & -0.4166 & -0.0877 & \dots \\
0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & 0.0000 & \dots \\
0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & 0.0000 & \dots
\end{matrix}\right. \\
&\qquad \left.\begin{matrix}
\dots & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & 0.0000 \\
\dots & -0.0050 & 0.0002 & 0.0000 & -0.0174 & -0.0000 & -0.0000 \\
\dots & -0.0057 & 0.0008 & -0.0000 & 0.0806 & 0.0000 & 0.0000 \\
\dots & -0.0729 & 0.0102 & 0.0000 & 0.2017 & -0.0000 & -0.0000 \\
\dots & -0.0101 & 0.0014 & -0.0000 & -0.4166 & 0.0000 & 0.0000 \\
\dots & 0.0273 & -0.0038 & -0.0000 & -0.0877 & 0.0000 & 0.0000 \\
\dots & 0.9933 & 0.0009 & 0.0000 & 0.0208 & -0.0000 & -0.0000 \\
\dots & 0.0009 & 0.9999 & 0.0000 & -0.0029 & 0.0000 & 0.0000 \\
\dots & 0.0000 & 0.0000 & -0.0000 & -0.0000 & -0.0000 & 0.0000 \\
\dots & 0.0208 & -0.0029 & -0.0000 & 0.6443 & -0.0000 & -0.0000 \\
\dots & -0.0000 & 0.0000 & -0.0000 & -0.0000 & -0.0000 & 0.0000 \\
\dots & -0.0000 & 0.0000 & 0.0000 & -0.0000 & 0.0000 & 0
\end{matrix}\right]
\end{aligned}
\tag{e.2b}
$$

$$
\hat{H}_{y_{w_{rl}}} = \left[\begin{matrix}
1.0000 & 0.0000 & -0.0000 & -0.0000 & -0.0000 & -0.0000 & \dots \\
0.0000 & 0.0106 & -0.0607 & -0.0477 & 0.0231 & 0.0212 & \dots \\
-0.0000 & -0.0607 & 0.8686 & -0.1648 & 0.0110 & -0.0009 & \dots \\
-0.0000 & -0.0477 & -0.1648 & 0.7369 & -0.2094 & -0.0029 & \dots \\
-0.0000 & 0.0231 & 0.0110 & -0.2094 & 0.1044 & -0.0051 & \dots \\
-0.0000 & 0.0212 & -0.0009 & -0.0029 & -0.0051 & 0.9995 & \dots \\
-0.0000 & -0.0006 & -0.1267 & -0.1929 & -0.1201 & -0.0028 & \dots \\
0.0000 & 0.0004 & 0.0072 & 0.0109 & 0.0068 & 0.0002 & \dots \\
0.0000 & 0.0000 & -0.0000 & 0.0000 & 0.0000 & 0.0000 & \dots \\
0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & \dots \\
0.0000 & -0.0598 & 0.2589 & 0.2885 & -0.1857 & 0.0048 & \dots \\
-0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 & -0.0000 & \dots
\end{matrix}\right.
$$

$$\left.\begin{matrix} \ldots & -0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & -0.0000 \\ \ldots & -0.0006 & 0.0004 & 0.0000 & 0.0000 & -0.0598 & -0.0000 \\ \ldots & -0.1267 & 0.0072 & -0.0000 & -0.0000 & 0.2589 & 0.0000 \\ \ldots & -0.1929 & 0.0109 & 0.0000 & 0.0000 & 0.2885 & -0.0000 \\ \ldots & -0.1201 & 0.0068 & 0.0000 & -0.0000 & -0.1857 & -0.0000 \\ \ldots & -0.0028 & 0.0002 & 0.0000 & 0.0000 & 0.0048 & -0.0000 \\ \ldots & 0.8556 & 0.0082 & 0.0000 & 0.0000 & 0.2362 & 0.0000 \\ \ldots & 0.0082 & 0.9995 & -0.0000 & -0.0000 & -0.0134 & -0.0000 \\ \ldots & 0.0000 & -0.0000 & 0 & -0.0000 & -0.0000 & 0.0000 \\ \ldots & 0.0000 & -0.0000 & -0.0000 & 0 & -0.0000 & 0.0000 \\ \ldots & 0.2362 & -0.0134 & -0.0000 & -0.0000 & 0.4247 & 0.0000 \\ \ldots & 0.0000 & -0.0000 & 0.0000 & 0.0000 & 0.0000 & -0.0000 \end{matrix}\right] \tag{e.2c}$$

$$\hat{H}_{y_{w_{rr}}} = \left[\begin{matrix} 1.0000 & -0.0000 & -0.0000 & -0.0000 & -0.0000 & -0.0000 & \ldots \\ -0.0000 & 0.0079 & -0.0750 & 0.0276 & 0.0007 & -0.0222 & \ldots \\ -0.0000 & -0.0750 & 0.8882 & -0.1346 & -0.0005 & -0.0035 & \ldots \\ -0.0000 & 0.0276 & -0.1346 & 0.7679 & -0.2226 & -0.0030 & \ldots \\ -0.0000 & 0.0007 & -0.0005 & -0.2226 & 0.1085 & -0.0048 & \ldots \\ -0.0000 & -0.0222 & -0.0035 & -0.0030 & -0.0048 & 0.9994 & \ldots \\ -0.0000 & -0.0184 & -0.1079 & -0.1688 & -0.1297 & -0.0030 & \ldots \\ 0.0000 & 0.0006 & 0.0061 & 0.0096 & 0.0073 & 0.0002 & \ldots \\ 0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & \ldots \\ 0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & \ldots \\ -0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & \ldots \\ 0.0000 & -0.0243 & 0.2527 & 0.2849 & -0.1740 & 0.0029 & \ldots \end{matrix}\right.$$

$$\left.\begin{matrix} \ldots & -0.0000 & 0.0000 & 0.0000 & 0.0000 & -0.0000 & 0.0000 \\ \ldots & -0.0184 & 0.0006 & 0.0000 & 0.0000 & -0.0000 & -0.0243 \\ \ldots & -0.1079 & 0.0061 & -0.0000 & -0.0000 & 0.0000 & 0.2527 \\ \ldots & -0.1688 & 0.0096 & 0.0000 & 0.0000 & -0.0000 & 0.2849 \\ \ldots & -0.1297 & 0.0073 & -0.0000 & -0.0000 & 0.0000 & -0.1740 \\ \ldots & -0.0030 & 0.0002 & 0.0000 & 0.0000 & -0.0000 & 0.0029 \\ \ldots & 0.8741 & 0.0071 & 0.0000 & 0.0000 & -0.0000 & 0.2296 \\ \ldots & 0.0071 & 0.9996 & -0.0000 & -0.0000 & 0.0000 & -0.0130 \\ \ldots & 0.0000 & -0.0000 & -0.0000 & 0.0000 & 0.0000 & 0.0000 \\ \ldots & 0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 & 0.0000 \\ \ldots & -0.0000 & 0.0000 & 0.0000 & -0.0000 & -0.0000 & 0.0000 \\ \ldots & 0.2296 & -0.0130 & 0.0000 & 0.0000 & 0.0000 & 0.3544 \end{matrix}\right] \tag{e.2d}$$

E.2 Design Data for Fault Group Four

This section presents data associated with the fault detection ¯lter design for fault group four, the throttle actuator, the brake actuator, the steering actuator and the manifold air mass sensor faults. The design details and a discussion are in Section 4.2.2 The matrix of assigned left eigenvectors $\mathcal{V}$ is

$$\mathcal{V} = \left[v_{u_{\tau_{b1}}}, v_{u_{\tau_{b2}}}, v_{0_1}, v_{0_2}, v_{0_3}, v_{0_4}, v_{0_5}, v_{0_6}, v_{0_7}, v_{0_8}, v_{u_{\alpha_1}}, v_{u_{\alpha_2}}, v_{u_\beta}, v_{y_{m_a}}\right]$$

$$= \begin{bmatrix} 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & \dots \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & \dots \\ -0.2987 & -0.0798 & -0.0347 & -0.2210 & -0.0072 & -0.1135 & -0.0660 & \dots \\ 0.2474 & -0.0862 & -0.0975 & 0.1370 & -0.0600 & -0.2142 & -0.0554 & \dots \\ 0.0741 & 0.2022 & -0.0180 & -0.0077 & 0.7732 & 0.0533 & -0.5932 & \dots \\ 0.0969 & 0.8501 & 0.4343 & -0.0493 & -0.0974 & -0.1769 & 0.1505 & \dots \\ 0.6112 & -0.1922 & 0.3729 & -0.2343 & -0.3427 & 0.2620 & -0.4229 & \dots \\ 0.5307 & 0.2180 & -0.6104 & -0.2680 & 0.1409 & 0.2070 & 0.3641 & \dots \\ -0.2724 & 0.3714 & -0.4539 & 0.1310 & -0.4681 & 0.3416 & -0.4727 & \dots \\ -0.1459 & 0.0026 & -0.1328 & -0.8021 & -0.1030 & -0.4189 & -0.1773 & \dots \\ -0.2900 & -0.0119 & 0.2551 & -0.3776 & 0.1484 & 0.7095 & 0.2277 & \dots \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & \dots \\ 0.0033 & -0.0108 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & \dots \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & \dots \end{bmatrix}$$

$$\begin{bmatrix} \dots & 0.0000 & 0.0000 & 0.0000 & -0.4154 & -0.4154 & 0.0000 & 0.0000 \\ \dots & 0.0000 & 0.0000 & 0.0000 & -0.0304 & -0.0304 & 0.0000 & -0.0334 \\ \dots & -0.0674 & -0.0227 & -0.0321 & 0.8761 & 0.8761 & 0.0005 & 0.9632 \\ \dots & 0.1102 & 0.1017 & 0.1833 & 0.0017 & 0.0004 & -0.5040 & 0.0009 \\ \dots & 0.0725 & -0.2177 & 0.0046 & -0.0241 & -0.0242 & -0.0075 & -0.0266 \\ \dots & 0.1496 & -0.8294 & 0.2028 & 0.0000 & 0.0000 & -0.0006 & 0.0000 \\ \dots & 0.5781 & 0.0477 & -0.3889 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ \dots & 0.5729 & -0.1711 & 0.0478 & 0.0028 & 0.0028 & -0.0026 & 0.0030 \\ \dots & -0.2232 & -0.2393 & 0.3537 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ \dots & -0.2431 & -0.0606 & -0.1128 & -0.2413 & -0.2413 & -0.0020 & -0.2653 \\ \dots & -0.4294 & -0.4017 & -0.7955 & -0.0014 & -0.0015 & -0.3289 & -0.0017 \\ \dots & 0.0000 & 0.0000 & 0.0000 & -0.0112 & -0.0119 & 0.0000 & 0.0000 \\ \dots & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ \dots & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & -0.7986 & 0.0000 \end{bmatrix}$$

where

$$\begin{array}{rclcrcl} v_{u_{\tau_{b1}}} & \in & V_{u_{\tau_{b1}}} & \quad & v_{u_{\tau_{b2}}} & \in & V_{u_{\tau_{b2}}} \\ v_{0_1} & \in & V_{0_1} & & v_{0_2} & \in & V_{0_2} \\ v_{0_3} & \in & V_{0_3} & & v_{0_4} & \in & V_{0_4} \\ v_{0_5} & \in & V_{0_5} & & v_{0_6} & \in & V_{0_6} \\ v_{0_7} & \in & V_{0_7} & & v_{0_8} & \in & V_{0_8} \\ v_{u_{\alpha_1}} & \in & V_{u_{\alpha_1}} & & v_{u_{\alpha_2}} & \in & V_{u_{\alpha_2}} \\ v_{u_\beta} & \in & V_{u_\beta} & & v_{y_{m_a}} & \in & V_{y_{m_a}} \end{array}$$

The matrix $\mathbb{W}$ associated with the left eigenvectors $\mathbb{V}$ is

$$\mathbb{W} = \left[w_{u_{\tau_{b1}}}, w_{u_{\tau_{b2}}}, w_{0_1}, w_{0_2}, w_{0_3}, w_{0_4}, w_{0_5}, w_{0_6}, w_{0_7}, w_{0_8}, w_{u_{\alpha_1}}, w_{u_{\alpha_2}}, w_{u_\beta}, w_{y_{m_a}}\right]$$

$$= \begin{bmatrix} 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & \dots \\ -0.3985 & 2.1071 & -0.0011 & -0.0153 & -0.5254 & -0.0527 & 0.4772 & \dots \\ 5.2129 & -30.3751 & -0.0633 & -0.2311 & -2.1086 & -0.3043 & 1.8758 & \dots \\ -0.6640 & -0.4129 & 0.6576 & 0.3023 & -0.1710 & -0.2946 & -0.4125 & \dots \\ -0.0144 & -0.5023 & 0.0763 & -0.3587 & 2.5332 & -0.0086 & -2.4937 & \dots \\ -2.6707 & -1.5851 & 5.6803 & 2.8445 & -1.6433 & -2.6140 & -5.0102 & \dots \\ 0.7404 & 1.1030 & 1.3954 & 8.7177 & 2.4947 & 5.5332 & 1.1599 & \dots \\ -10.2358 & -10.5460 & 16.9547 & 13.0929 & 46.8558 & -13.5944 & -65.6255 & \dots \\ -2.8892 & -14.0216 & 0.4363 & 0.5322 & -74.6867 & -5.6399 & 67.4728 & \dots \\ 2.2630 & 17.8320 & -0.3809 & -0.6196 & 72.6355 & 5.3593 & -65.4520 & \dots \\ 3.4007 & 12.0161 & 0.4531 & -0.0412 & 82.5552 & 6.4825 & -75.2208 & \dots \\ -1.0436 & -22.6153 & -0.3933 & 0.8835 & -78.7647 & -5.6159 & 71.8864 & \dots \end{bmatrix}$$

$$\begin{bmatrix} \dots & 0.0000 & 0.0000 & 0.0000 & 1.6614 & 3.7382 & 0.0000 & 10.2391 \\ \dots & -0.0551 & 0.1783 & -0.0013 & -0.9469 & -0.7825 & 0.0009 & -0.8789 \\ \dots & -0.2684 & 0.6723 & -0.0262 & -0.6606 & -0.2928 & -0.0017 & -0.4836 \\ \dots & -0.5631 & 0.2499 & 0.0809 & -0.0018 & 0.0026 & 0.1598 & 0.0012 \\ \dots & 0.1675 & -1.3179 & -0.0569 & 0.6414 & 1.3682 & 0.0278 & 1.1846 \\ \dots & -8.0615 & 2.6989 & -0.8829 & -0.0100 & -0.0232 & -0.0714 & -0.0197 \\ \dots & 3.6905 & 0.4336 & 2.0548 & 0.9218 & 2.1104 & 0.0171 & 1.7975 \\ \dots & -4.3303 & -5.1871 & 13.0855 & -0.6735 & -1.3325 & -7.3383 & -1.1740 \\ \dots & -7.6009 & 25.4596 & 0.2275 & 0.5095 & 1.1843 & 0.2158 & 1.0052 \\ \dots & 7.4717 & -24.7021 & -0.2517 & -1.0911 & -2.3345 & -0.2088 & -2.0197 \\ \dots & 8.0677 & -28.0256 & -0.5744 & 0.5309 & -0.8923 & -0.0534 & -0.3551 \\ \dots & -7.4781 & 26.7949 & 0.7636 & 2.3150 & 3.0237 & 0.0384 & 3.0126 \end{bmatrix}$$

The fault detection ¯lter gain L found from (4.9), $\bar{V}^T L = \bar{W}^T$ is

$$
L = \left[\begin{array}{rrrrrrc}
100.33 & 6.83 & 15.00 & 0.12 & 29.87 & -0.04 & \ldots \\
-306.69 & -15.59 & 405.89 & 3.54 & 80.18 & -442.98 & \ldots \\
-0.00 & -1.37 & 12.59 & 0.12 & 3.95 & -13.94 & \ldots \\
0.00 & -0.24 & -0.61 & -1.01 & 2.60 & 8.74 & \ldots \\
0.00 & -0.70 & -3.27 & -0.03 & 3.64 & 0.79 & \ldots \\
0.00 & -0.02 & -0.06 & -0.00 & 0.49 & -1.44 & \ldots \\
0.00 & -0.04 & -0.17 & -0.00 & 0.22 & -0.70 & \ldots \\
0.00 & 0.02 & 0.18 & -1.08 & -0.15 & -12.01 & \ldots \\
0.00 & -0.04 & -0.16 & -0.00 & 0.27 & 0.84 & \ldots \\
-0.00 & 0.39 & -3.21 & -0.03 & -0.58 & 4.99 & \ldots \\
0.00 & -0.06 & -0.17 & -0.40 & 0.74 & 2.61 & \ldots \\
-3027.92 & -239.30 & -534.91 & -4.36 & -1065.10 & 1.59 & \ldots \\
-0.00 & -196.99 & 2649.44 & 23.23 & 104.54 & -120.99 & \ldots \\
-0.00 & 0.18 & 0.50 & 0.60 & -2.01 & -6.49 & \ldots
\end{array}\right.
$$

$$
\left.\begin{array}{crrrrrr}
\ldots & 47.69 & -28.13 & 25.51 & -48.83 & -55.88 & 26.71 \\
\ldots & -49.87 & 861.41 & -1354.33 & 1270.58 & 1534.55 & -1405.89 \\
\ldots & -2.84 & 28.89 & -45.73 & 41.99 & 52.36 & -45.78 \\
\ldots & -15.97 & -20.21 & -35.32 & 34.12 & 39.29 & -38.53 \\
\ldots & 1.49 & 74.30 & -102.96 & 100.08 & 114.05 & -108.98 \\
\ldots & -0.03 & 1.81 & -2.70 & 2.65 & 2.96 & -2.87 \\
\ldots & 0.24 & 5.62 & -5.27 & 5.12 & 6.26 & -5.97 \\
\ldots & 0.80 & -34.87 & 3.14 & -3.00 & -4.59 & 4.42 \\
\ldots & 0.27 & 3.83 & -5.14 & 4.79 & 5.68 & -5.20 \\
\ldots & -10.98 & -6.83 & 10.91 & -9.84 & -13.07 & 10.24 \\
\ldots & -4.54 & -23.55 & -8.92 & 8.58 & 10.12 & -9.93 \\
\ldots & -1700.87 & 1002.82 & -910.40 & 1741.89 & 1993.37 & -953.17 \\
\ldots & 95.53 & 1803.00 & -228.45 & -149.32 & 531.03 & 501.29 \\
\ldots & 11.93 & 31.10 & 26.59 & -25.68 & -29.88 & 29.31
\end{array}\right] \qquad \text{(e.3)}
$$

The output projection matrices $\hat{H}_{u_\alpha}$, $\hat{H}_{u_{\tau_b}}$, $\hat{H}_{u_\beta}$ and $\hat{H}_{y_{m_a}}$ are as follows.

$$
\hat{H}_{u_\alpha} = \left[\begin{array}{rrrrrrl}
0.8886 & 0.0680 & 0.1377 & 0.0012 & -0.0183 & -0.0001 & \ldots \\
0.0680 & 0.0062 & 0.0109 & 0.0001 & -0.0009 & -0.0000 & \ldots \\
0.1377 & 0.0109 & 0.0263 & 0.0002 & 0.0457 & 0.0002 & \ldots \\
0.0012 & 0.0001 & 0.0002 & 0.0000 & 0.0001 & 0.0000 & \ldots \\
-0.0183 & -0.0009 & 0.0457 & 0.0001 & 0.9962 & -0.0000 & \ldots \\
-0.0001 & -0.0000 & 0.0002 & 0.0000 & -0.0000 & 1.0000 & \ldots \\
0.0129 & 0.0006 & -0.0168 & -0.0001 & 0.0022 & 0.0000 & \ldots \\
0.0000 & 0.0000 & -0.0000 & -0.0000 & 0.0000 & 0.0000 & \ldots \\
-0.1168 & 0.0098 & -0.0321 & -0.0003 & -0.0149 & -0.0002 & \ldots \\
-0.1209 & 0.0074 & -0.0001 & 0.0000 & -0.0164 & -0.0002 & \ldots \\
-0.1464 & -0.0265 & -0.0556 & -0.0005 & -0.0187 & -0.0002 & \ldots \\
-0.1587 & -0.0239 & -0.0008 & -0.0000 & -0.0218 & -0.0002 & \ldots
\end{array}\right.
$$

$$
\left.\begin{array}{lrrrrrr}
\ldots & 0.0129 & 0.0000 & -0.1168 & -0.1209 & -0.1464 & -0.1587 \\
\ldots & 0.0006 & 0.0000 & 0.0098 & 0.0074 & -0.0265 & -0.0239 \\
\ldots & -0.0168 & -0.0000 & -0.0321 & -0.0001 & -0.0556 & -0.0008 \\
\ldots & -0.0001 & -0.0000 & -0.0003 & 0.0000 & -0.0005 & -0.0000 \\
\ldots & 0.0022 & 0.0000 & -0.0149 & -0.0164 & -0.0187 & -0.0218 \\
\ldots & 0.0000 & 0.0000 & -0.0002 & -0.0002 & -0.0002 & -0.0002 \\
\ldots & 0.9984 & -0.0000 & 0.0141 & 0.0146 & 0.0180 & 0.0195 \\
\ldots & -0.0000 & 1.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\
\ldots & 0.0141 & 0.0000 & 0.8316 & -0.1670 & -0.2181 & -0.2223 \\
\ldots & 0.0146 & 0.0000 & -0.1670 & 0.8334 & -0.2159 & -0.2219 \\
\ldots & 0.0180 & 0.0000 & -0.2181 & -0.2159 & 0.7158 & -0.2887 \\
\ldots & 0.0195 & 0.0000 & -0.2223 & -0.2219 & -0.2887 & 0.7034
\end{array}\right] \quad \text{(e.4a)}
$$

$$
\hat{H}_{u_{\tau_b}} = \left[\begin{array}{rrrrrrl}
0 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & 0.0000 & \ldots \\
-0.0000 & 0.0063 & -0.0595 & -0.0005 & 0.0016 & 0.0000 & \ldots \\
0.0000 & -0.0595 & 0.9964 & 0.0087 & 0.0001 & 0.0000 & \ldots \\
-0.0000 & -0.0005 & 0.0087 & 0.0001 & -0.0003 & 0.0000 & \ldots \\
0.0000 & 0.0016 & 0.0001 & -0.0003 & 1.0000 & -0.0000 & \ldots \\
0.0000 & 0.0000 & 0.0000 & 0.0000 & -0.0000 & 1.0000 & \ldots \\
-0.0000 & -0.0000 & -0.0000 & -0.0000 & 0.0000 & 0.0000 & \ldots \\
-0.0000 & -0.0000 & -0.0000 & -0.0000 & 0.0000 & 0.0000 & \ldots \\
0.0000 & 0.0000 & -0.0000 & 0.0000 & 0.0000 & -0.0000 & \ldots \\
-0.0000 & 0.0000 & -0.0000 & 0.0000 & 0.0000 & -0.0000 & \ldots \\
0.0000 & -0.0402 & -0.0024 & -0.0000 & 0.0001 & 0.0000 & \ldots \\
-0.0000 & -0.0339 & -0.0020 & -0.0000 & 0.0001 & 0.0000 & \ldots
\end{array}\right.
$$

$$\begin{array}{l} \left.\begin{matrix} \dots & -0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 \\ \dots & -0.0000 & -0.0000 & 0.0000 & 0.0000 & -0.0402 & -0.0339 \\ \dots & -0.0000 & -0.0000 & -0.0000 & -0.0000 & -0.0024 & -0.0020 \\ \dots & -0.0000 & -0.0000 & 0.0000 & 0.0000 & -0.0000 & -0.0000 \\ \dots & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0001 & 0.0001 \\ \dots & 0.0000 & 0.0000 & -0.0000 & -0.0000 & 0.0000 & 0.0000 \\ \dots & 1.0000 & -0.0000 & 0.0000 & 0.0000 & -0.0000 & -0.0000 \\ \dots & -0.0000 & 1.0000 & 0.0000 & 0.0000 & -0.0000 & -0.0000 \\ \dots & 0.0000 & 0.0000 & 1.0000 & -0.0000 & 0.0000 & 0.0000 \\ \dots & 0.0000 & 0.0000 & -0.0000 & 1.0000 & 0.0000 & 0.0000 \\ \dots & -0.0000 & -0.0000 & 0.0000 & 0.0000 & 0.9984 & -0.0014 \\ \dots & -0.0000 & -0.0000 & 0.0000 & 0.0000 & -0.0014 & 0.9988 \end{matrix}\right] \end{array} \tag{e.4b}$$

$$\hat{H}_{u_\beta} = \left[\begin{matrix} 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & 0.0000 & \dots \\ -0.0000 & 0.0010 & 0.0003 & 0.0002 & 0.0005 & 0.0000 & \dots \\ 0.0000 & 0.0003 & 0.0050 & -0.0063 & 0.0485 & 0.0002 & \dots \\ -0.0000 & 0.0002 & -0.0063 & 0.9999 & 0.0006 & 0.0000 & \dots \\ 0.0000 & 0.0005 & 0.0485 & 0.0006 & 0.9959 & -0.0000 & \dots \\ 0.0000 & 0.0000 & 0.0002 & 0.0000 & -0.0000 & 1.0000 & \dots \\ 0.0000 & -0.0004 & -0.0188 & -0.0004 & 0.0025 & 0.0000 & \dots \\ 0.0000 & -0.0000 & -0.0000 & -0.0000 & 0.0000 & 0.0000 & \dots \\ -0.0000 & 0.0187 & -0.0140 & 0.0028 & -0.0173 & -0.0002 & \dots \\ 0.0000 & 0.0167 & 0.0186 & 0.0030 & -0.0189 & -0.0002 & \dots \\ -0.0000 & -0.0153 & -0.0330 & 0.0035 & -0.0217 & -0.0002 & \dots \\ 0.0000 & -0.0117 & 0.0238 & 0.0039 & -0.0251 & -0.0003 & \dots \end{matrix}\right.$$

$$\left.\begin{matrix} \dots & 0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 \\ \dots & -0.0004 & -0.0000 & 0.0187 & 0.0167 & -0.0153 & -0.0117 \\ \dots & -0.0188 & -0.0000 & -0.0140 & 0.0186 & -0.0330 & 0.0238 \\ \dots & -0.0004 & -0.0000 & 0.0028 & 0.0030 & 0.0035 & 0.0039 \\ \dots & 0.0025 & 0.0000 & -0.0173 & -0.0189 & -0.0217 & -0.0251 \\ \dots & 0.0000 & 0.0000 & -0.0002 & -0.0002 & -0.0002 & -0.0003 \\ \dots & 0.9982 & -0.0000 & 0.0158 & 0.0164 & 0.0201 & 0.0218 \\ \dots & -0.0000 & 1.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ \dots & 0.0158 & 0.0000 & 0.8163 & -0.1829 & -0.2373 & -0.2431 \\ \dots & 0.0164 & 0.0000 & -0.1829 & 0.8169 & -0.2358 & -0.2435 \\ \dots & 0.0201 & 0.0000 & -0.2373 & -0.2358 & 0.6917 & -0.3149 \\ \dots & 0.0218 & 0.0000 & -0.2431 & -0.2435 & -0.3149 & 0.6751 \end{matrix}\right] \tag{e.4c}$$

$$
\hat{H}_{y_{m_a}} = \begin{bmatrix}
0 & -0.0000 & -0.0000 & -0.0000 & 0.0000 & 0.0000 & \dots \\
-0.0000 & 0.0478 & 0.0950 & 0.0008 & -0.0121 & -0.0001 & \dots \\
-0.0000 & 0.0950 & 0.1966 & 0.0017 & 0.0231 & 0.0000 & \dots \\
-0.0000 & 0.0008 & 0.0017 & 0.0000 & -0.0001 & 0.0000 & \dots \\
0.0000 & -0.0121 & 0.0231 & -0.0001 & 0.9992 & -0.0000 & \dots \\
0.0000 & -0.0001 & 0.0000 & 0.0000 & -0.0000 & 1.0000 & \dots \\
0.0000 & 0.0084 & -0.0008 & -0.0000 & 0.0001 & 0.0000 & \dots \\
-0.0000 & 0.0000 & -0.0000 & -0.0000 & 0.0000 & 0.0000 & \dots \\
0.0000 & -0.0615 & -0.1765 & -0.0016 & 0.0042 & -0.0000 & \dots \\
0.0000 & -0.0664 & -0.1496 & -0.0013 & 0.0035 & -0.0000 & \dots \\
0.0000 & -0.1159 & -0.2367 & -0.0021 & 0.0054 & -0.0000 & \dots \\
0.0000 & -0.1208 & -0.1971 & -0.0017 & 0.0042 & -0.0000 & \dots
\end{bmatrix}
$$

$$
\begin{bmatrix}
\dots & 0.0000 & -0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\
\dots & 0.0084 & 0.0000 & -0.0615 & -0.0664 & -0.1159 & -0.1208 \\
\dots & -0.0008 & -0.0000 & -0.1765 & -0.1496 & -0.2367 & -0.1971 \\
\dots & -0.0000 & -0.0000 & -0.0016 & -0.0013 & -0.0021 & -0.0017 \\
\dots & 0.0001 & 0.0000 & 0.0042 & 0.0035 & 0.0054 & 0.0042 \\
\dots & 0.0000 & 0.0000 & -0.0000 & -0.0000 & -0.0000 & -0.0000 \\
\dots & 0.9999 & -0.0000 & 0.0005 & 0.0006 & 0.0010 & 0.0011 \\
\dots & -0.0000 & 1.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\
\dots & 0.0005 & 0.0000 & 0.9540 & -0.0402 & -0.0646 & -0.0559 \\
\dots & 0.0006 & 0.0000 & -0.0402 & 0.9646 & -0.0569 & -0.0496 \\
\dots & 0.0010 & 0.0000 & -0.0646 & -0.0569 & 0.9083 & -0.0801 \\
\dots & 0.0011 & 0.0000 & -0.0559 & -0.0496 & -0.0801 & 0.9296
\end{bmatrix} \quad \text{(e.4d)}
$$

References

Bakker, E., L. Nyborg, and H. Pacejka 1987. Tyre Modeling for use in Vehicle Dynamics Studies. pages 1{15. SAE. Paper 870421.

Bakker, E. and H. Pacejka 1989. A New Tyre Model with an Application in Vehicle Dynamics Studies. pages 83{93. SAE. Paper 890087.

Banavar, R. N. and J. L. Speyer 1991. An Linear-Quadratic Game Approach to Estimation and Smoothing. In *Proceedings of the American Control Conference*, pages 2818{2822.

Basseville, M. and I. Nikiforov 1995. *Detection of Abrupt Changes : Theory and Applications*. Prentice Hall. Edited by Thomas Kailath.

Beard, R. V. 1971. *Failure Accomodation in Linear Systems through Self-Reorganization*. PhD thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA.

Bell, D. J. and D. H. Jacobsen 1973. *Singular Optimal Control.* Academic Press.

Blaydon, C. C. 1967. Recursive Algorithms for Pattern Classi¯cation. Research note, O±ce of Naval Research.

Bourlard, H. and C. J. Wellekens 1994. *Connectionist Speech Recognition : A Hybrid Approach*. Kluwer Academic Press.

Brogan, W. L. 1991. *Modern Control Theory*. Prentice-Hall, 3rd edition.

Cebon, D. and D. E. Newland 1983. The Arti¯cial Generation of Road Surface Topography by the Inverse FFT Method. In *Proceedings of the 8th IAVSD Symposium on the Dynamics of Vehicles on Roads and Railway Tracks*, pages 29{42.

Chichka, D. F. and J. L. Speyer 1995. An Adaptive Controller Based on Disturbance Attenuation. *IEEE Transactions on Automatic Control*, Vol. AC-40, No. 7:1220{1233.

Clements, D. J. and B. D. O. Anderson 1978. *Singular Optimal Control: The Linear-Quadratic Problem*, volume 5 of *Lecture Notes in Control and Information Sciences*. Springer-Verlag.

Douglas, R. K. 1993. *Robust Fault Detection Filter Design*. PhD thesis, The University of Texas at Austin, Austin, TX.

Douglas, R. K. and J. L. Speyer 1995a. An $\mathcal{H}_\infty$ Bounded Fault Detection Filter. In *Proceedings of the American Controls Conference*.

Douglas, R. K. and J. L. Speyer 1995b. Robust Fault Detection Filter Design. In *Proceedings of the American Controls Conference*.

Douglas, R. K. and J. L. Speyer 1996. Robust Fault Detection Filter Design. *AIAA Journal of Guidance, Control, and Dynamics,* Vol. 19, No. 1:214{218.

Douglas, R. K., J. L. Speyer, D. L. Mingori, R. H. Chen, D. P. Malladi, and W. H. Chung 1995. Fault Detection and Identification with Application to Advanced Vehicle Control Systems. Technical report, University of California, Los Angeles, MANE Dept. Final Report Prepared for California PATH Program.

Doyle, J. C. 1984. Lecture Notes on Advances in Multivariable Control. ONR/Honeywell Workshop, Minneapolis, MN.

Doyle, J. C., K. Glover, P. P. Khargonekar, and B. A. Francis 1989. State-Space Solutions to Standard $\mathcal{H}_2$ and $\mathcal{H}_\infty$ Control Problems. *IEEE Transactions on Automatic Control*, Vol. AC-34, No. 8:831–847.

Emami-Naeini, A., M. M. Akhter, and S. M. Rock 1988. Effect of Model Uncertainty on Failure Detection: The Threshold Selector. *IEEE Transactions on Automatic Control*, Vol. AC-33, No. 12:1106–1115.

Francis, B. A. 1979. The Optimal Linear-Quadratic Time-Invariant Regulator with Cheap Control. *IEEE Transactions on Automatic Control*, Vol. 24, No. 4:616–621.

Frank, P. M. and J. Wünnenberg 1989. Robust Fault Diagnosis Using Unknown Input Observer Schemes. In Patton, R., P. Frank, and R. Clark, editors, *Fault Diagnosis in Dynamic Systems*. Prentice Hall.

Funahashi, K.-I. 1989. On the Approximate Realization of Continuous Mappings by Neural Networks. *Neural Networks*, Vol. 2.

Gill, R. 1983. The Influence of Correlated Random Road Excitation Processes on Vehicle Vibration. In *Proceedings of the 8th IAVSD Symposium on the Dynamics of Vehicles on Roads and Railway Tracks*, pages 449–459.

Green, M. and D. J. Limebeer 1995. *Linear Robust Control*. Prentice-Hall.

Haykin, S. S. 1994. *Neural Networks : A Comprehensive Foundation*. Maxwell Macmillan International, New York.

Hedrick, J. K., D. H. McMahon, and D. Swaroop 1993. Vehicle Modeling and Control for Automated Highway Systems. Technical report, University of California, Berkeley. UCB ITS PRR 93-24.

Hornik, K., M. Stinchcombe, and H. White 1989. Multilayer Feedforward Networks are Universal Approximators. *Neural Networks*, Vol. 2.

Jacobson, D. H. 1971. Totally Singular Quadratic Optimization Problems. *IEEE Transactions on Automatic Control*, Vol. AC-16, No. 6:651{658.

Jones, H. L. 1973. *Failure Detection in Linear Systems*. PhD thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA.

Kučera, V. 1972. A Contribution to Matrix Quadratic Equations. *IEEE Transactions on Automatic Control*, Vol. AC-17, No. 6:344{347.

Kwakernaak, H. and R. Sivan 1972. The Maximally Achievable Accuracy of Linear Optimal Regulators and Linear Optimal Filters. *IEEE Transactions on Automatic Control*, Vol. AC-17, No. 1:79{86.

Lee, G. H. 1994. *Least-Squares and Minimax Methods for Filtering, Identi¯cation and Detection*. PhD thesis, University of California, Los Angeles.

Lippmann, R. P. 1987. An Introduction to Computing with Neural Nets. *IEEE ASSP Magazine*, Vol. 4:4{22.

Lukowski, S. A., M. L. Chu, D. R. Tener, and P. W. Claar II 1990. Open-Loop, Fixed-Control Simulation of a Vehicle Undergoing Steering and Acceleration Maneuvers. pages 1{12. SAE. Paper 901658.

Lukowski, S. A. and L. Medeksza 1992. Vehicle Cornering Behavior Analysis Using a General Purpose Simulation Methodology. *Total Vehicle Dynamics*, Vol. 1:49{54.

Mangoubi, R. S., B. D. Appleby, and G. C. Verghese 1994. Stochastic Interpretation of $\mathcal{H}_\infty$ and Robust Estimation. In *Proceedings of the 33rd Conference on Decision and Control*, pages 3943{3948. IEEE.

Massoumnia, M.-A. 1986. A Geometric Approach to the Synthesis of Failure Detection Filters. *IEEE Transactions on Automatic Control*, Vol. AC-31, No. 9:839{846.

Massoumnia, M.-A., G. C. Verghese, and A. S. Willsky 1989. Failure Detection and Identi¯cation. *IEEE Transactions on Automatic Control*, Vol. AC-34, No. 3:316{321.

Moore, B. C. 1981. Principal Component Analysis in Linear Systems: Controllability, Observability, and Model Reduction. *IEEE Transactions on Automatic Control*, Vol. AC-26, No. 1.

Moore, B. C. and A. J. Laub 1978. Computation of Supremal (A, B)-Invariant and Controllability Subspaces. *IEEE Transactions on Automatic Control*, Vol. AC-23, No. 5.

Morgan, N. and H. Bourlard 1995. Continuous Speech Recognition. *IEEE Signal Processing Magazine*, Vol. 5:25{40.

Morse, A. S. 1973. Structural Invariants of Linear Multivariable Systems. *SIAM Journal of Control*, Vol. 11, No. 3:446{465.

Moylan, P. J. and J. B. Moore 1971. Generalizations of Singular Optimal Control Theory. *Automatica*, Vol. 7:591{598.

Nguyen, T. M. 1996. Modeling and Simulation of Road Vehicles. Master's thesis, Department of Mechanical and Aerospace Engineering, University of California, Los Angeles, CA.

Nikiforov, I. V. 1995. A Generalized Change Detection Algorithm. *IEEE Transactions on Information Theory*, Vol. 41, No. 1:171 { 187.

Oshman, Y. and I. Y. Bar-Itzhack 1985. Eigenfactor Solution of the Matrix Riccati Equation - A Continuous Square Root Algorithm. *IEEE Transactions on Automatic Control*, Vol. AC-30, No. 10:971{978.

Pacejka, H. B. and E. Bakker 1991. The Magic Formula Tyre Model: Tyre Models for Vehicle Dynamics Analysis. In *Proceedings of the 1st International Colloquium on Tyre Models for Vehicle Dynamics Analysis*, pages 1{18.

Peng, H. 1992. *Vehicle Lateral Control for Highway Automation*. PhD thesis, Department of Mechanical Engineering, University of California, Berkeley, CA.

Rhee, I. and J. L. Speyer 1991. A Game Theoretic Approach to a Finite-Time Disturbance Attenuation Problem. *IEEE Transactions on Automatic Control*, Vol. AC-36, No. 9:1021{1032.

Richard, M. D. and R. P. Lippmann 1991. Neural Network Classi¯ers Estimate Bayesian Aposteriori Probabilities. *Neural Computation*, Vol. 3:461{483.

Robson, J. D. 1980. Road Surface Description and Vehicle Response. *Int. Journal of Vehicle Design*, Vol. 1, No. 1:25{35.

Schumacher, J. C. 1983. The Role of the Dissapation Matrix in Singular Optimal Control. *System and Control Letters*, Vol. 2, No. 3:262{266.

Shiryayev, A. N. 1977. *Optimal Stopping Rules*. Springer-Verlag, New York.

Smith, D. E. and J. M. Starkey 1992. E®ects of Model Complexity on the Performance of Automated Vehicle Steering Controllers: Model Development, Validation, and Comparison. *Vehicle System Dynamics*, Vol. 24:163{181.

Speyer, J. L. and J. E. White 1984. Shiryayev Sequential Probability Ratio Test for Redundancy Management. *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 7, No. 5:588 { 595.

Veillette, R. J., J. V. Medani¶c, and W. R. Perkins 1992. Design of Reliable Control Systems. *IEEE Transactions on Automatic Control*, Vol. AC-37, No. 3:290{304.

White, J. E. and J. L. Speyer 1987. Detection Filter Design: Spectral Theory and Algorithms. *IEEE Transactions on Automatic Control*, Vol. AC-32, No. 7:593{603.

Willems, J. C. 1971. Least-Squares Stationary Optimal Control and the Algebraic Riccati Equation. *IEEE Transactions on Automatic Control*, Vol. AC-16, No. 6:621{634.

Willems, J. C. 1982. Almost Invariant Subspaces: An Approach to High Gain Feedback Design { Part II: Almost Conditionally Invariant Subspaces. *IEEE Transactions on Automatic Control*, Vol. AC-27, No. 5:1071{1085.

Willumeit, H. P., M. Necular, A. Vikas, and A. Wohler 1992. Mathematical Models for the Computation of Vehicle Dynamic Behavior During Development. *Total Vehicle Dynamics*, Vol. 1:41{48.

Wonham, W. M. 1985. *Linear Multivariable Control: A Geometric Approach*. Springer-Verlag, New York, 3rd edition.

Yaesh, I. and U. Shaked 1993. Game Theory Approach to Finite-Time Horizon Optimal Estimation. *IEEE Transactions on Automatic Control*, Vol. 38, No. 6:957{963.

Yip, C. K., D. N. Crolla, and D. N. L. Horton 1992. The In°uence of Aerodynamic E®ects on Car Handling. *Total Vehicle Dynamics*, Vol. 1:11{23.