Robust Lateral Control of Highway Vehicles

Raymond H. Byrne

B.S.E.E., University of Virginia, 1987

M.S.E.E., University of Colorado, Boulder, 1989

Ph.D., University of New Mexico, 1995

Abstract

This dissertation explores robust control topics related to the automatic lateral control of highway vehicles. The main topic is the application of robust control to the lateral control problem where the model uncertainty is characterized as additive unstructured perturbations in the frequency-domain. This approach relies on interpolation theory to construct the stabilizing controller. New interpolation techniques are presented which allow interpolation on the jw axis and with points that have multiplicity. Test results comparing the robust controller and a PID controller are presented. Two other robust control approaches, simultaneous stabilization of a family of plants and a multiobjective design, are also discussed in the context of the lateral control problem.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Automatic control of highway vehicles has been of great interest since the concept was first introduced by General Motors Corporation at the 1939 World's Fair [34]. Many advantages to automatic operation of highway vehicles have been claimed. These include increased freeway throughput, increased highway capacity, increased safety and reduced congestion. Automated highways would also free humans of the driving task. The expected benefits of automated highways are described in more detail in [42, 32]. This thesis ignores many of the larger non-technical institutional issues, and focuses on the lateral control of highway vehicles. The goal of the lateral control system is to keep the vehicle centered in the lane. In addition, the lateral control system might be required to perform lane changes or evasive maneuvers for collision avoidance.

In this introductory chapter, some of the difficulties associated with the lateral control of highway vehicles are examined. These difficulties provide the motivation for applying robust control techniques. Then, contributions of this thesis are summarized and an outline of the document is provided.

1.1 Lateral Control of Highway Vehicles

The goal of the lateral control system is to keep the vehicle on the desired trajectory. Usually this is centered in the lane, but may also include lane changes and evasive collision avoidance maneuvers. Therefore, one of the design objectives is to minimize the lateral path error between the actual trajectory and the desired trajectory. A two degree-of-freedom (DOF) linearized model is typically used to design the lateral control system. This model is discussed in more detail in Chapter 2. However, it is important to note that it is a linearized model, and that many of the parameters are subject to change or are unknown. For instance, the vehicle mass changes as passengers and cargo are loaded and unloaded. As tire pressure and road conditions change, the cornering stiffness of the tires can change dramatically. The linearized model is also highly dependent on the vehicle's velocity, which has a wide range of possible values. Therefore, another design criterion is that the lateral control system must function over a wide range of parameter changes.

In addition to providing performance over a wide range of parameter changes, the lateral control system must also ensure that the driver is comfortable under automatic control. This is usually translated into minimizing the lateral acceleration and jerk that is experienced under automatic control. Finally, the lateral control system must exhibit some robustness to sensor, actuator, and controller faults which results in a graceful degradation as systems malfunction. The fact that the vehicle model is time-varying and subject to many variations and uncertainties lends itself to the application of robust control techniques, which are introduced in the next section.

1.2 Robust Control

Robust control can be defined as the control of uncertain systems with fixed controllers [27]. Generally, the goal is to design a fixed compensator which guarantees stability, and hopefully some performance, as the plant varies. The design of the robust controller is governed by the model of the system uncertainty. The uncertainty is typically classified as structured or unstructured, depending on the amount of information available. The uncertainty can be expressed in the time domain via a state-space model, or in the frequency domain with a transfer function model. This leads to the following four common classifications for the modeling of the system uncertainty.

- time-domain, unstructured uncertainty
- time-domain, structured uncertainty
- frequency-domain, unstructured uncertainty
- frequency-domain, structured uncertainty

Frequency-domain, unstructured uncertainty can be further broken down into additive perturbation, input-multiplicative perturbation, and output-multiplicative perturbation. Examples of each are shown in (1.1 -1.3), where $G_0(s)$ represents the transfer function of the nominal plant.

$$\text{Additive perturbation} \quad G(s) = G_0(s) + \delta G(s) \tag{1.1}$$

$$\text{Input multiplicative perturbation} \quad G(s) = G_0(s)(I + R(s)) \tag{1.2}$$

$$\text{Output multiplicative perturbation} \quad G(s) = (I + L(s))G_0(s) \tag{1.3}$$

This dissertation deals with the application of robust control to the lateral control problem using frequency-domain additive unstructured uncertainty to model the unknown and time-varying parameters. Although this approach is not specifically designed to accommodate time-varying systems, the assumption that the variations will occur slowly allows time variations to be included. Robust control is an attractive technique to deal with the lateral control problem for several reasons. First, a fixed controller can be used, which eliminates the need for any gain-scheduling or adaptation algorithms. A robust controller also theoretically guarantees stability over a wide range of parameter changes. Finally, because the vehicle model and uncertainty are treated in the frequency domain, actual test data on the vehicle response can be gathered by injecting sinusoids of different frequencies. The only drawback of this approach is that a solution is not guaranteed to exist, but this is discussed in more detail in Chapter 3. The next section presents the contributions of this thesis.

1.3 Contributions of the Thesis

The contributions of this thesis are made in two areas. The first contribution is in robust control of systems with frequency-domain unstructured additive perturbations. The second contribution is in applying this theory to the lateral control of automobiles and comparing it to several other techniques. This includes field testing of the robust control algorithm on an experimental vehicle.

Robust control of systems with additive unstructured perturbations was first presented by Kimura in [46]. This thesis extends the results of Kimura by allowing multiplicity of unstable poles, as well as poles on the jw axis. These extensions are based on new

interpolation techniques for handling interpolation points on the jw axis and points with multiplicity. One technique for handling poles on the jw axis was developed by Li in [48], but handling a multiplicity of poles on the jw axis was not discussed. The interpolation techniques presented in this thesis are also applicable to other types of robust control theory.

The motivation for extending the work of Kimura was to allow the application of this type of robust control theory to the lateral control of highway vehicles. This thesis also presents experimental results from the implementation of the robust control algorithm on a testbed vehicle. The performance of the robust control algorithm is compared to a proportional-integral-derivative (PID) controller. In addition, several other control techniques applicable to the lateral control problem are discussed. These include simultaneous stabilization, as well as minimizing the sensitivity and the derivative of the control sensitivity to improve driver ride comfort.

1.4 Thesis Outline

An outline of the remaining chapters is as follows:

- In Chapter 2, a more detailed review of previous lateral control research is provided.
- In Chapter 3, Robust control theory for plants with unstructured additive perturbations is reviewed. This includes a discussion of interpolation theory, which is the basic tool used to arrive at the robustly stabilizing controller. The limitations of interpolation theory are discussed, and techniques which overcome these limitations are proposed.

- In Chapter 4, the results presented in Chapter 3 are used to design a robustly stabilizing compensator for the lateral control system.

- In Chapter 5, the control system designed in Chapter 4 is implemented on a test vehicle. The performance of the robust controller is compared to a proportional-integral-derivative (PID) controller during field testing.

- In Chapter 6, several other promising control techniques for lateral control are discussed. These include minimum-phase simultaneous stabilization, and minimizing the complimentary sensitivity function and its derivative to improve ride comfort.

- In Chapter 7, our conclusions and recommendations for future work are presented.

- Appendix A describes the testbed vehicle in more detail.

- Appendix B contains MATLAB programs developed for vehicle control simulations.

- Appendix C contains vehicle control software used in the field testing.

Chapter 2

Review of Lateral Vehicle Control Research

One of the fundamental goals of ITS (Intelligent Transportation Systems) is to develop automated highways, where vehicles are capable of automatic operation. The most popular concept is platoons of 10-15 vehicles, separated by 1 meter, traveling down the highway in a special lane at normal highway speeds. Automatic operation of highway vehicles is usually broken down into lateral and longitudinal control. At a higher level, lateral and longitudinal control must be coordinated to perform maneuvers like a lane change. Lateral and longitudinal dynamics are also related – lateral models are a function of longitudinal velocity, and lateral steering causes torque disturbances in the engine and powertrain. However, lateral and longitudinal control are usually treated separately in the literature, especially when the necessary low-level control systems are addressed. While other survey papers have mentioned vehicle control topics [34, 68, 51], this chapter reviews the different approaches

to lateral control system design taken over the last 40 years. The chapter is organized in a roughly chronological order, but some discussions are kept together and out of order when they are performed by the same researcher or organization.

2.1 Early Research

The automated highway system concept (AHS) was first introduced in the General Motors Pavilion at the 1939 World's Fair [34]. Initial research efforts were conducted by RCA Laboratories in the 1950's [84]. Lateral control was achieved by following an electrified cable buried in the center of the lane. The sensing system consisted of two coils at the front of the vehicle which detected the magnetic field caused by the alternating current flowing through the wire. The control algorithm used in the early tests is summarized by "In a dynamic situation with properly stabilized servo controls the car will proceed down the cable with no wavering" [84]. A 300 foot section of highway near Lincoln, Nebraska was used for a full-scale working demonstration in September of 1957 [84]. Other demonstrated technologies included loop vehicle detectors, and a radio warning system for following vehicles controlled by the detectors [84].

RCA and General Motors demonstrated an automated highway system at an electronic test track developed by RCA in the late 1950's [39]. The same concept of following a buried wire for lateral control was applied. The control algorithm developed was a function of velocity, otherwise "the car could operate successfully at only one speed" [39].

Extensive research efforts in vehicle control were conducted at the Ohio State University (OSU) from 1964 until 1980 [34]. In 1961, Barrick used the transfer function shown in

equation (2.1) to model the lateral dynamics of a vehicle [12].

$$\frac{E_f(s)}{\delta_f(s)} = \frac{K(V_x)(s + a(V_x))}{s^2} \tag{2.1}$$

where:

$E_f(s)$ = lateral deviation from lane center of the front center of the vehicle,

$\delta_f(s)$ = the front-wheel angle,

V_x = the vehicle forward speed,

$K(V_x)$ and $a(V_x)$ = velocity dependent parameters.

After further tests at OSU, this model was determined to be inaccurate and more suitable models were developed. One of the simplest models for the lateral motion of a vehicle is a two degree-of-freedom bicycle model. This is shown in Figure 2.1 and described by equations (2.2) and (2.3) [79].

$$m\dot{V}_y + \left[mV_x + \frac{2l_1C_{\alpha f} - 2l_2C_{\alpha r}}{V_x}\right]\Omega_z + \left[\frac{2C_{\alpha f} + 2C_{\alpha r}}{V_x}\right]V_y = 2C_{\alpha f}\delta_f(t) \tag{2.2}$$

$$I_z\dot{\Omega}_z + \left[\frac{2{l_1}^2C_{\alpha f} - 2{l_2}^2C_{\alpha r}}{V_x}\right]\Omega_z + \left[\frac{2l_1C_{\alpha f} - 2l_2C_{\alpha r}}{V_x}\right]V_y = 2l_1C_{\alpha f}\delta_f(t) \tag{2.3}$$

The quantities shown in Figure 2.1 and equations (2.2) and (2.3) are described in Table 2.1. Rearranging equations (2.2) and (2.3) yields the following state-space system:

$$\dot{x} = Ax + Bu \tag{2.4}$$

where $A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$, $B = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$, $u = \delta_f$, and $x = \begin{bmatrix} V_y \\ \Omega_z \end{bmatrix}$.

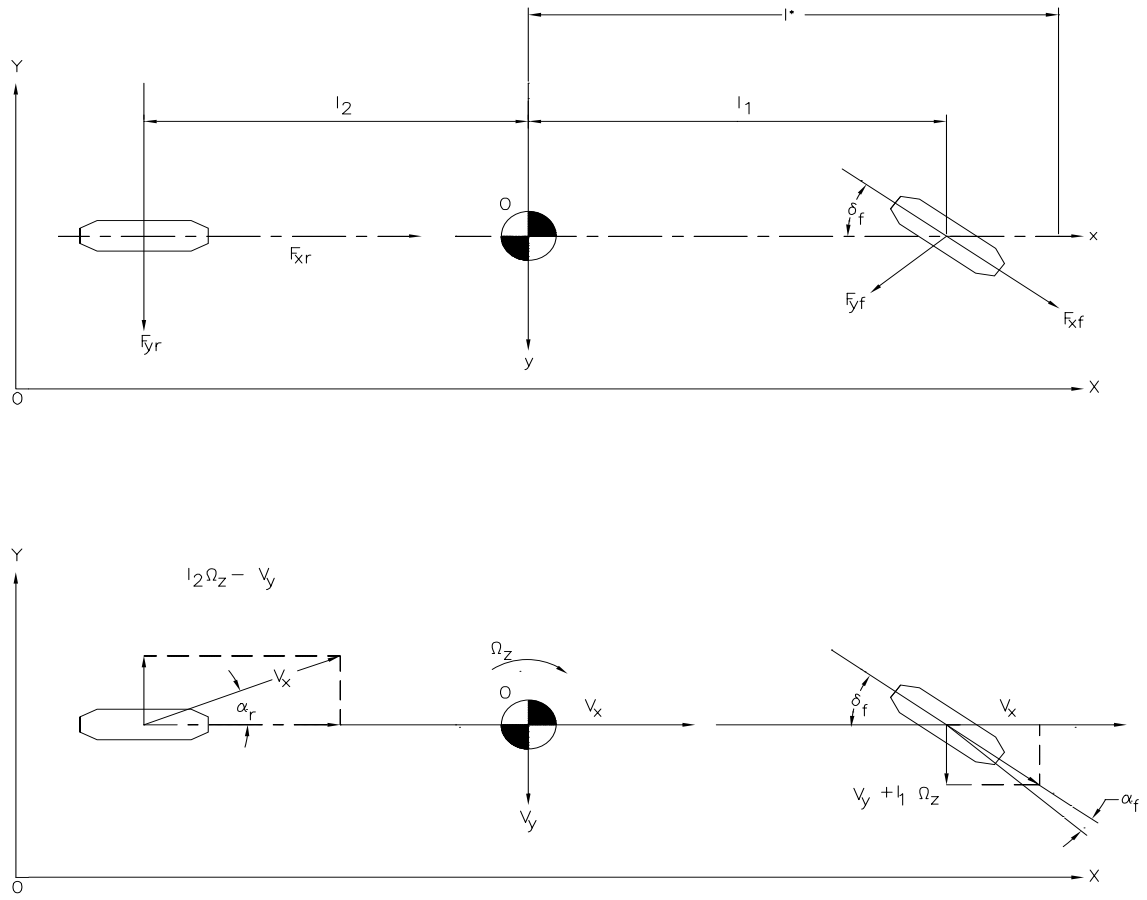


Figure 2.1: Simplified Bicycle Model for the Car

Parameter	**Description**
m	Vehicle mass (kg)
V_x	Longitudinal velocity (vehicle coordinates) (m/s)
V_y	Lateral velocity (vehicle coordinates) (m/s)
l_1, l_2	Distance from front and rear axles to c.g. (m)
Ω_z	Yaw-rate about the c.g. (rad/s)
δ_f	Front steering angle $(rads)$
C_r, C_f	Front and rear tire cornering stiffness (Kn/rad)
I_z	Inertia about the z-axis $(kg \cdot m^2)$
l^*	Distance from the lateral sensor to the c.g. (m)

Table 2.1: Summary of Coefficients in Bicycle Model

$$\begin{array}{rclrcl} a_{11} &=& -\left[\ \frac{2C_{\alpha f}+2C_{\alpha r}}{V_x m}\ \right] & a_{12} &=& -\left[\ V_x + \frac{2l_1C_{\alpha f}-2l_2C_{\alpha r}}{V_x m}\ \right] \\ a_{21} &=& -\left[\ \frac{2l_1C_{\alpha f}-2l_2C_{\alpha r}}{V_x I_z}\ \right] & a_{22} &=& -\left[\ \frac{2{l_1}^2C_{\alpha f}+2{l_2}^2C_{\alpha r}}{V_x I_z}\ \right] \\ b_1 &=& \frac{2C_{\alpha f}}{m} & b_2 &=& \frac{2l_1C_{\alpha f}}{I_z} \end{array} \tag{2.5}$$

The two state variables are the yaw-rate Ω_z and the lateral velocity V_y. The lateral path error E_{cg} of the center of gravity ($c.g.$) is a function of the lateral velocity V_y, the heading Θ, and the longitudinal velocity V_x. This relation is shown in equations (2.6) and (2.7).

$$\dot{E}_{cg} = V_y + V_x\Theta \tag{2.6}$$

$$\dot{\Theta} = \Omega_z \tag{2.7}$$

If the lateral error is sensed at the front of the vehicle a distance l^* from the $c.g.$, then equation (2.6) must be changed as shown in (2.8). The augmented state-space model is shown in equation (2.9). E_f is the path error sensed at the front of the vehicle.

$$\dot{E}_f = V_y + V_x\Theta + l^*\Omega_z \tag{2.8}$$

$$\begin{bmatrix} \dot{V}_y \\ \dot{\Omega}_z \\ \dot{\Theta} \\ \dot{E}_f \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & 0 & 0 \\ a_{21} & a_{22} & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & l^* & V_x & 0 \end{bmatrix} \begin{bmatrix} V_y \\ \Omega_z \\ \Theta \\ E_f \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ 0 \\ 0 \end{bmatrix} \delta_f(t) \tag{2.9}$$

The bicycle model for lateral motion described by (2.9) has four eigenvalues, with two located at the origin. The remaining two eigenvalues, describing the lateral dynamics of

the vehicle, are usually a stable complex pair. For constant velocities and the lateral error as the only output ($C = [0001]$), the state-space model described by (2.9) takes the form of (2.10) when expressed as a transfer function,

$$\frac{E_f(s)}{\delta_f(s)} = \frac{q_1 s^2 + q_2 s + q_3}{s^2(q_4 s^2 + q_5 s + q_6)} \tag{2.10}$$

The origin of the "bicycle" model is apparently unknown [34]. A modified version of the bicycle model was used at OSU for lateral controller designs in the 1970's [34]. The OSU model is described by equation (2.11).

$$\begin{aligned} E_f(s) &= K_1(V_x)\frac{s^2 + a_1(V_x)s + a_2(V_x)}{s^2 \triangle(s)}\delta_f(s) - K_2(V_x)\frac{s + a_3(V_x)}{s^2}\Omega_z(s) + \\ &\quad K_3(V_x)\frac{s^2 + a_4(V_x)s + a_5(V_x)}{s^2 \triangle(s)}F_d(s) + K_4(V_x)\frac{s^2 + a_6(V_x)s + a_7(V_x)}{s^2 \triangle(s)}F_e(s) \end{aligned} \tag{2.11}$$

where:

$\triangle(s) = s^2 + a_8(V_x)s + a_9(V_x)$

$\Omega_z(s) =$ yaw-rate command

$F_d(s) =$ disturbance input

$F_e(s) =$ road elevation input

$K_i(V_x) =$ velocity dependent gains (i=1,2,....9)

$a_j(V_x) =$ velocity dependent parameters and/or a collection of parameters ($j = 1, 2, ..., 9$).

Equation (2.11) is basically a two degree-of-freedom bicycle model plus a roll-steer correction term [36]. The $K_i(V_x)$ and $a_i(V_x)$ parameters were identified by field tests and from manufacturer-supplied data for a 1965 Plymouth sedan [36, 55]. Note that the four

poles of the OSU model correspond to the four poles of the simple bicycle model described by equation (2.11). The additional terms in the OSU model correspond to a commanded yaw-rate term, a disturbance term, and a superelevation term (road elevation angle causes roll steering).

An early wire-following system developed at OSU in 1967 used two buried wires to overcome some of the problems associated with a single cable and relied on mechanically moving the sensor coils as a linear function of the front wheel angle to improve system damping. This did not yield satisfactory results at velocities greater than 30 mph, so a lead-compensator was tried. The lateral transient response, with and without the moving coils, was improved at higher velocities by this lead-compensator [33]. A phase-lead filter was then used in experiments between 1967 and 1969 [55].

Work in the early 1970's at OSU used the model described by (2.11), and classical control theory to design lateral controllers [36]. Both single-loop (lateral-error (E_f) feedback) and dual-loop (lateral error (E_f) and yaw-rate (Ω_z) feedback) controllers were examined. The single-loop compensator was designed using a settling time criterion, and limits on E_f for maximum disturbance conditions. In order to meet these requirements, a lead-lag compensator was employed. Root locus techniques were used to analyze the compensator at two different speeds [36]. The performance of the dual-loop compensator was evaluated experimentally, and gave good results at lower speeds. However, a satisfactory means of obtaining the yaw-rate Ω_z was not available. Both controllers were implemented on-board the vehicle with an analog computer containing 20 op-amps.

The fixed-gain compensator designed at OSU performed satisfactorily at lower speeds, but encountered difficulties at higher speeds. Problems included decreased damping with

increased speed, and large errors for large yaw-rate inputs associated with small radius turns at higher speeds [36]. In order to overcome these problems, velocity-dependent compensators were designed [18, 35].

The velocity-dependent compensator consisted of a $G_v(s)$ term added to the lead-lag compensator shown in (2.12). The goal of adding $G_v(s)$ was to cancel out the velocity-dependent poles and zeros of the system.

$$G_c(s) = K_1 G_v(s) \left[\frac{s+\alpha}{s+\beta}\right] \left[\frac{s+\gamma}{s}\right] \tag{2.12}$$

The lag compensator was used to reduce the steady-state tracking error. The lead compensator was used to overcome the destabilizing affect of the phase lag [18]. The car model used is shown in equation (2.13).

$$E_f(s) = 114.5 \frac{\left(s^2 + 81.65\frac{s}{V_x} + 17.95\right)}{s^2\left(s^2 + 156.1\frac{s}{V_x} + 6063\frac{1}{{V_x}^2} + 26.93\right)} \delta_f(s) \tag{2.13}$$

The transfer function $G_v(s)$ chosen to cancel out the velocity-dependent modes of the vehicle is shown in (2.14).

$$G_v(s) = \left(\frac{s^2 + 2a_1 s + {a_1}^2 + {b_1}^2}{s^2 + 81.65\frac{s}{V_x} + 17.95}\right)\left(\frac{s^2 + 156.1\frac{s}{V_x} + 6063\frac{1}{{V_x}^2} + 29.63}{s^2 + 2hs + h^2 + k^2}\right) \tag{2.14}$$

$G_c(s)$ is then called a velocity-adaptive controller. A lag compensator was also tried (without lead compensation). Root locus techniques and analog computer simulations, along with field tests, were used to evaluate the performance of the velocity-adaptive controllers. Note however, the use of the Laplace Transform in (2.13) and (2.14) is not valid

because the system is time varying. Therefore, although the system worked in practice, all theoretical guarantees of stability are lost when such an approach is taken.

A digital version of the velocity-adaptive controller was later implemented using an Intel 8086 microprocessor. The same compensator described by (2.12) and (2.14) was implemented in digital form, using a bilinear transformation [37, 53].

$$s = \frac{2}{T}\frac{z-1}{z+1}, \quad T = 120ms \tag{2.15}$$

The performance of the digital controller very closely matched the performance of the analog controller, which was expected.

A two-frequency radar which sensed the distance to a barrier was later tried at OSU as an alternative to the wire-following approach [51]. However, the only mention of the controller is that the velocity-adaptive one designed in [18] was used.

While the early efforts were being conducted at OSU, similar wire-following experiments were also being conducted in Japan and England [45, 16]. In Japan, automatic control of the steering, brake and throttle was demonstrated with a wire-following system [45]. One wire, excited at 1200 Hz was used as the lateral reference system. The on-board sensors measured the signal amplitude to determine the vehicle's lateral position. A proportional-derivative (PD) control law was used in the lateral control system. In order to improve performance on curved sections, a nonlinear gain was applied to increase the gain as the lateral error increased. The shape of this gain is shown in Figure 2.2.

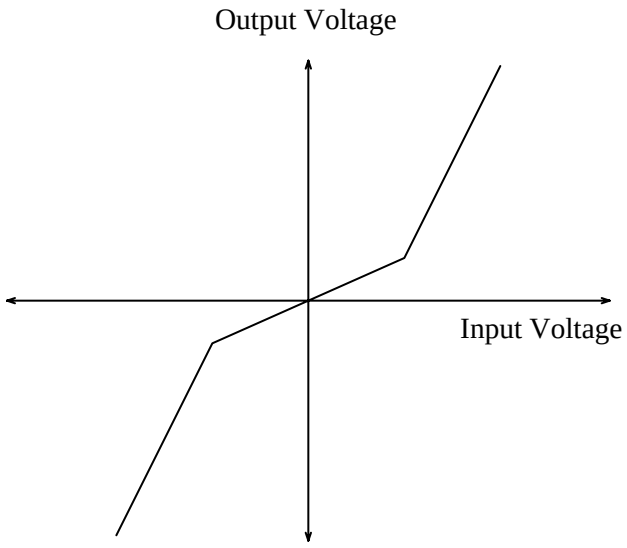


Figure 2.2: Nonlinear Gain Curve

2.2 Modern Control Theory

Modern control theory, including state-space approaches and optimal control, was applied to vehicle control as early as 1971 [57]. However, inadequate sensor technology did not allow practical implementations until much later. Therefore, frequency-domain and classical control techniques, which are more oriented towards input-output transfer function analysis, remained prevalent in the early hardware implementations. This section reviews the application of modern control theory to lateral control of highway vehicles.

Pasternack in 1971 discussed the observability and controllability of the linearized lateral dynamics of a rubber-tired vehicle [57]. He also discussed the application of optimal control to the lateral control problem. In 1974, Bonderson proposed using optimal control with a quadratic performance index that penalizes the lateral displacement, lateral acceleration, and yaw acceleration [13].

In the mid-1980's an optimal controller was developed at OSU [31]. The optimal control

design involved choosing a weighting matrix for the cost function, and solving the algebraic Riccati equation (ARE). The vehicle dynamics are described by (2.13). The steering actuator dynamics were also modeled, yielding a 5th-order state-space model. In order to take care of the velocity dependence of the model, the ARE was solved at different velocities so that $K = K(V_x)$ (gain-scheduling). The velocity increment $\triangle V_x$ used for gain-scheduling was 1 m/s, and gains were calculated as V_x varied from 10 m/s to 35 m/s. Since all 5 states were not available, a reduced-order observer was designed to estimate the states. The poles of the observer were placed at the roots of $(s + 6.2)^2(s + 29)^2$. The design was carried out for a continuous-time system, and then discretized using the bilinear transformation for computer implementation. A sampling period of $T = 120ms$ was chosen, which is faster than ten times the closed-loop bandwidth (ten times closed-loop bandwidth $\sim 256ms$). The optimal controller was implemented on an Intel 8086 microprocessor, and simulated on an analog computer representing the vehicle dynamics. Although an optimal controller was implemented (Linear Quadratic Regulator - LQR design), a traditional observer was used to estimate the states. If a Kalman Filter had been used in conjunction with Loop Transfer Recovery (LTR), some robustness properties could have been added to the system at fixed velocities. However, no robustness properties can be guaranteed for the optimal approach taken.

Research on Automated Guideway Transit Vehicles conducted in the 1970's was very much concerned with the lateral control of rubber-tired vehicles. Automated Guideway Transit (AGT) was a class of proposed urban transportation systems in which vehicles operated under automatic longitudinal and lateral control on exclusive guideways [69]. Several lateral control strategies were presented in [67]. These included an optimal Wiener filter, an

LQR design, and a suboptimal reduced-state-feedback controller. A two degree-of-freedom bicycle model was used to design and evaluate the different control strategies. The performance index for the Wiener filter consisted of mean-square acceleration and mean-square tracking error at a given point on the vehicle. A fourth-order compensator was designed which cancels unwanted vehicle poles and zeros to yield the optimal desired transfer function. The high-order of the compensator and pole-zero cancellation made the Wiener filter implementation unattractive at the time.

A full-state-feedback Linear Quadratic Regulator (LQR) design was also presented in [69], with the detailed calculations appearing in [67]. Although the LQR design and Wiener filter are very similar, the LQR approach does not rely on pole-zero cancellations to yield the desired transfer function. However, the LQR design does rely on full-state feedback. A Kalman filter was proposed for cases when full-state feedback was not available. Because of the added complexity of a Kalman filter design, a reduced-state feedback controller which used only lateral position and yaw angle error was designed to evaluate the performance of a relatively simple controller.

The reduced-state-feedback proposed in [69] can be implemented with one or two sensors, depending on the sensor location, with the following control law:

$$\delta_f = -K_y(y - y_0) - K_\theta \theta \tag{2.16}$$

where θ represents the yaw angle and $(y - y_0)$ is the lateral path error. K_y is the gain which multiplies the path error, while K_θ is the gain which multiplies the yaw angle error. As noted in [69], if the lateral error sensor is located a distance l^* in front of the vehicle center of mass, the error sensed is a function of the vehicle lateral position error as measured at

the center of mass, $(y - y_0)$, and of the yaw angle, θ, for non-zero l^*.

$$E_f = (y + l^*\theta) - y_0 \tag{2.17}$$

Taking advantage of (2.17), the simple steering controller described by (2.18) was examined in [69]. The gain K_c represents the overall controller gain, which multiplies the lateral error and the yaw angle estimate $l^*\theta$.

$$\delta_f = -K_c\left[(y + l^*\theta) - y_0\right] \tag{2.18}$$

The performance of the simple controller described by (2.18) was compared to the performance of the baseline Wiener optimal design. The results indicated that the performance of the simple controller closely matched the optimum design, and the loading of the vehicle did not significantly affect performance. Much of this was attributed to the design of the vehicle — well-damped open-loop dynamics and nearly neutral-steer.

The effects of steering actuator dynamics on vehicle performance were also examined in [69]. The steering actuator dynamics were modeled by a first-order lag. For all of the conditions studied, vehicle dynamics were unaffected by actuators with responses flat up to 16 Hz. With the simple proportional controller studied, actuator time-constants greater than about 0.025 seconds caused instability. Other results discussed in [69] include the effects of vehicle speed on controller performance and occupant comfort. The ISO (International Organization for Standardization) passenger comfort specifications were used to evaluate root-mean-square (rms) lateral acceleration data. The performance of the simple proportional controller was found to degrade quickly with increasing speed, while the full-state-feedback controller was more robust with respect to velocity changes. This is reasonable because LQR designs with full state feedback have inherent robustness properties,

$(0.5, \infty)$ gain margin and $(+60^\circ, -60^\circ)$ phase margin [47].

In the mid 1970's, Daimler-Benz conducted research on automatic lateral control of buses [17, 19]. State-space design methods were applied. The stability limitations of a simple proportional controller with output feedback (front error, front and rear error) were discussed. Based on the limitations of proportional control, an optimal LQR approach was suggested. An attempt was made to find a fixed optimal controller which performed well over a wide range of varying parameters. A fixed optimal controller with low parameters sensitivity was not found, so efforts shifted to designing a pole-placement controller. An observer was designed to estimate the states. Although the observer had a fixed gain while parameters were varying, the performance of the observer was fairly robust to parameter changes, especially when compared to a simple differentiating filter for state estimation.

The largest AHS effort is currently being undertaken at PATH (Program for Advanced Technology for the Highway) in California. The PATH approach to automatic lateral control relies on sensing the location of buried magnets spaced 1 meter apart. In addition to providing lateral position error data, the magnets are also coded using polarity changes to provide road curvature information to the vehicle lateral control system. The use of preview control was deemed necessary to improve performance on sharp curves, especially at high speeds [56]. The magnetic reference system is described in great detail in [83].

The overall lateral controller designed at PATH consists of a feedback and a feedforward component. The feedback component of the control system was designed using frequency-shaped linear quadratic (FSLQ) control theory, while the feedforward controller was designed to generate a preview steering command based on the road curvature information [63]. A six degree-of-freedom complex model was used for simulations, while a simplified

two degree-of-freedom model was used for the controller design. Open-loop simulations were conducted to verify that the simplified and complex models performed similarly under typical operating conditions.

FSLQ control theory was chosen for lateral vehicle control at PATH because frequency-dependent weighting factors can be placed on the tracking error, lateral acceleration, and control effort [63]. In addition, weighting factors on the tracking error terms can be shaped in the frequency-domain to increase the robustness of the plant to high frequency noise and unmodeled dynamics. The weighting factor on the acceleration term can also be weighted in the frequency domain to ensure good ride quality in the frequency range sensitive to passengers [63].

FSLQ theory is an extension of linear-quadratic gaussian methods [40]. Traditional linear quadratic regulator (LQR) designs are based on the linear model (2.19) and the quadratic performance index (2.20).

$$\dot{x} = Ax + Bu,\ x(0) = x_0,\ (0 \leq t \leq t_f) \tag{2.19}$$

$$J = \int_0^{t_f} \left(x^T Q x + u^T R u \right) dt + x^T(t_f) M x(t_f) \tag{2.20}$$

The dynamics of the plant are described by (A, B), and the design objective is to minimize the cost index J , with Q positive semidefinite ($Q \geq 0$), R positive definite ($R > 0$), and M positive semidefinite ($M \geq 0$). If the standard LQR cost function is written in the frequency domain using Parseval's theorem with the weighting matrices Q and R functions of frequency, and with no weighting on the final state, the result is (2.21)

[40].

$$J = \frac{1}{2}\int_{-\infty}^{+\infty} \left[x^H(jw)Q(jw)x(jw) + u^H(jw)R(jw)u(jw)\right] dw \tag{2.21}$$

$Q(jw)$ and $R(jw)$ must be Hermitian [1] at all frequencies, and a solution is guaranteed if $R(jw)$ is positive definite and $Q(jw)$ is positive semidefinite at all but a finite number of discrete frequencies [40]. The dynamic equation (2.19) and the cost function (2.21) may be written in terms of an extended state vector. The control law is then obtained by solving a modified algebraic Riccati equation [40].

Experimental results for the FSLQ controller approach are described in [62]. Sensitivity reduction was used to lessen the effects of perturbations in velocity and cornering stiffness. Open-loop tests were conducted to determine tire model parameters, steering actuator parameters, and overall vehicle response. The magnetic sensor system was calibrated using a video stripe sensing system. A proportional-integral-derivative (PID) controller was also tested for comparison purposes. Test speeds were relatively slow, $20 - 50$ $kmph$ (12-30 mph), and the control system could not keep the vehicle within the range of the magnetic marker system (± 25 cm) in the high speed tests of 50 $kmph$. In subsequent testing, the performance was improved at higher speeds, and the maximum velocity was increased to 60 $kmph$ [43].

Other lateral control research conducted by PATH researchers has analyzed the use of lateral velocity and yaw-rate as two independent control inputs [50]. This approach requires an extra input in addition to front steering angle, which might consist of rear independent

[1] def. of Hermitian is $A^H(jw) = A(jw)\ \forall w$ where A^H is the complex conjugate transpose of A

steering, front differential torque, or rear differential torque [50]. These combinations would allow independent control of lateral and yaw motions. The range of steady-state lateral velocity and yaw-rate were analyzed for three different cases: conventional front-wheel steering, front-wheel steering and front-differential force, and front and rear independent steering. The case of independent front and rear wheel steering allowed widest range of reachable steady-state yaw-rate and lateral velocity. An optimal controller was developed for the front and rear independent steering system, and was evaluated using simulations as well as experiments on a small model car. An estimator was also designed to estimate the cornering stiffness of the tires for use in the control algorithm. Gain-scheduling was used to account for varying velocities and cornering stiffness [50]. The control law also incorporated a feedforward term to compensate for centrifugal force. In the actual experiments, yaw-rate was measured by an angular velocity sensor, and lateral velocity was obtained by integrating data from an accelerometer [68]. The use of gain-scheduling and sub-optimal estimation without loop transfer recovery results in the loss of any inherent robustness properties in the optimal controller design.

A topic related to lateral control, robust fault detection in lateral control systems, was also analyzed as part of the PATH program [59]. The approach taken involves designing three observers, each of which uses two out of the three available sensor measurements. Should one of the sensors fail, only one observer would be affected. The observer was designed by writing the system dynamics in the form:

$$\dot{x} = Ax + Bu + Ed \tag{2.22}$$

$$y = Cx + Du + f_s \tag{2.23}$$

where the E matrix accounts for the parameter uncertainties, d is the disturbance input, and f_s represents the sensor fault vector. Defining an observer as:

$$\dot{\hat{x}} = A\hat{x} + Bu + K(y - \hat{y}) \tag{2.24}$$

$$\hat{y} = C\hat{x} + Du \tag{2.25}$$

The observer error dynamics become:

$$\dot{e} = (A - KC)e + Ed + Kf_s,\ F = (A - KC) \tag{2.26}$$

Defining the residue r as $r = He$, yields the following transfer function from uncertainty d to r:

$$G_{rd}(s) = H(sI - F)^{-1}E \tag{2.27}$$

$$\text{where } F = (A - KC)$$

For the residue to be insensitive to uncertainties, G_{rd} should be zero, as shown in (2.28).

$$G_{rd}(s) = H(sI - F)^{-1}E = WC(sI - F)^{-1}E = 0 \tag{2.28}$$

The K and W matrices are then chosen using eigenstructure assignment as described in [59, 58].

Simulation and experimental results of the sensor fault detection algorithm are discussed in [59]. The same model car used in [50] was used for testing the fault detecting observer scheme. As in [50], noisy acceleration data was a problem, and lateral acceleration sensor faults were not easily detected.

A nonlinear combined lateral and longitudinal controller for lateral control proposed by PATH researchers is described in [66]. A nonlinear model, which takes into account

engine dynamics, steering actuator dynamics, and lateral dynamics was used. The proposed nonlinear control law is very similar to computed-torque control laws in robotics. This approach is outlined by equations (2.29-2.30).

$$M\ddot{q} + N(q) + kq = u(t) \tag{2.29}$$

$$u(t) = N(q) + L(\dot{q}) \tag{2.30}$$

If the dynamics of a system are described by (2.29), where $N(q)$ is a known nonlinearity, a feedback law of the form of (2.30) can be used to cancel the nonlinearity and place the poles of the remaining linear system. However, the robustness of the system is affected by inexact cancellation of the nonlinearities. This approach is taken in [66], but approximations of the sine and cosine terms are made in order to arrive at the control law. The control law was simulated for a platoon of 5 vehicles accelerating on a curved road section. The control system performed well in simulation, and further work will address the robustness of the controller to measurement noise and communication delays between vehicles [66].

Fuzzy logic has also been applied to the lateral control problem by researchers at PATH [43]. Fuzzy logic was used to develop feedback, preview, and gain scheduling rule bases. The fuzzy logic controller was then tested on a Toyota Celica, developed by IMRA America. Testing was performed at 30 $kmph$, 40 $kmph$, 50 $kmph$, and 60 $kmph$. The fuzzy logic controller was capable of maintaining lateral errors of less than 10 cm at speeds up to 50 $kmph$. The experimental performance of the fuzzy logic controller was comparable to the PID and FSLQ controllers previously tested by PATH.

Robust control techniques applied to vehicle lateral control are described in [3, 6, 5, 2, 1]. In [3], a gain-scheduled controller for four-wheel steering that yields velocity-independent

yaw eigenvalues is presented. The bicycle model is used to model vehicle lateral dynamics. A result presented in [1], which states that a feedback compensator $H_f(s) = \frac{1}{s}$ for the front wheel steering angle decouples the yaw mode from the lateral mode of the front axle, is used in [3]. Using this compensator, the yaw-rate and front steering angle are unobservable from the acceleration of the front axle, and the acceleration of the front axle is not controllable from the rear steering angle [3]. A controller described by (2.31) is then proposed for the rear steering which results in robust yaw damping by yaw-rate feedback.

$$\delta_r = \left(\frac{l}{V_x} - k_D \right) (w_r - \Omega_z) \tag{2.31}$$

where:

δ_r = rear wheel steering angle

l = wheelbase

V_x = longitudinal velocity

k_D = controller parameter adjusted to obtain the desired damping

w_r = commanded rear yaw-rate

Ω_z = vehicle yaw-rate

Note that the controller described by (2.31) requires gain-scheduling for the velocity term. The controller does however eliminate the velocity dependence of the characteristic polynomial of the closed-loop system [3]. Decoupling feedback ($H_f(s) = \frac{1}{s}$) yields a velocity-independent natural frequency, but low damping. Adding rear-wheel steering with the yaw-rate feedback shown in (2.31) yields velocity independent eigenvalues and better damping, comparable to the eigenvalue region of a conventional car [3].

In [6], yaw-rate feedback in conjunction with simultaneous pole region assignment for four extremal plants is applied to improve the robustness of a bus lateral control system. The four extremal plants consist of combinations of the maximum/minimum velocity and mass: (m^+, v^-), (m^+, v^+), (m^-, v^-), (m^-, v^+). A hyperbola was chosen as the stability boundary. The closed-loop poles of the system were placed to the left of the hyperbola for the four extremal plants. Simulation results showed that the robust controller designed with yaw-rate feedback performed better than a feedback system which only uses lateral error. The improvements included reduced maximal displacements, reduced oscillatory tendencies, and more uniform responses over the range of operation conditions (m, v) [6].

In [2], sliding mode control is applied to the lateral control problem. A bicycle model was used for the controller design, but the simulations were conducted with an extended nonlinear two-track model. Two different approaches were taken for the sliding mode controller design. The first assumed lateral displacement and steering angle as the two measurable states. The second approach assumed that lateral displacement and yaw-rate were the two measurable states. The robustness of this approach comes from the application of sliding mode control, but the controller parameters were hand-tuned to obtain a suitable trade-off between tracking performance and ride quality [2].

Research on passenger ride comfort during lane changes has been studied by Mitsubishi [41]. Experimental tests were conducted to determine the profile of typical and emergency lane changes. Based on the lateral acceleration and jerk profile of a typical lane change, a lateral controller and trajectory generator were designed. The trajectory generator commands the steering actuator position and velocity to obtain the symmetric acceleration and jerk trajectories seen in the test data. The steering position control is only used during

lane changes, not during normal lateral control, and involves the use of a feedforward term to the steering actuator. Both simulations and field tests were used to evaluate the control system performance.

The application of model-reference adaptive control (MRAC) to the lateral control problem is described in [14]. Adaptive control is promising because it allows for adaptation as parameters change (velocity, cornering stiffness, mass, etc.). Although model-reference adaptive control guarantees stability and performance after adaptation, performance is not guaranteed during adaptation. There are also several limitations to practical implementation. These include instability caused by steering actuator saturation during adaptation, and a tradeoff between control amplitude and frequency content during adaptation. These two issues require further research before model-reference adaptive control is suitable for lateral control of vehicles. One possible approach for handling actuator saturation is described in [71, 80]. This involves using the sum of several saturation functions to obtain the control input.

There has also been a significant amount of research conducted on vision-based lateral and longitudinal control of vehicles [49, 52, 65, 78, 21, 20, 24, 75, 76, 22, 9, 10, 74, 61, 25]. Much of this work has focused on the vision aspects, and little attention has been paid to the actual control algorithm. In other cases, the lateral control algorithm is "hidden" in a neural network vision algorithm and is difficult to analyze [61]. Much of the vision-based navigation work done in the 1980's was part of the Autonomous Land Vehicle (ALV) project [49, 52, 65, 78, 21, 20, 24, 75, 76, 22, 9]. The most successful vision-based road following research is currently being conducted at Carnegie Mellon University (CMU) and in Germany [61, 25]. The CMU approach uses neural networks while the Germans are using more classical vision

techniques (line and stripe following, edge detection). Both have demonstrated successful lateral control at highway speeds (55 *mph* plus). The National Institute of Standards (NIST) has also conducted some vision-based road following research, including experimental work with a High Mobility Multipurpose Wheeled Vehicle (HMMWV – the Army jeep).

2.3 Summary and Conclusions

A summary of the history of lateral control research appears in Table 2.3. Many of the concepts popular today, optimal control and velocity gain-scheduling, have their origins in the early research. This is somewhat driven by the requirements and problems encountered when designing a lateral control system. Table 2.2 summarizes the basic requirements and problems.

As seen in Table 2.2, there are many unsolved problems. These include dealing with nonlinearities and time varying parameters. Although gain-scheduling is often used to solve the velocity dependent problems, there are no guarantees of stability. Therefore extensive simulation and field testing must be conducted to ensure the stability of a gain-scheduling control system. Adaptive control can be applied to solve some of the time varying issues, but there are no performance guarantees while the system is adapting. This can lead to oscillations, and in the case of saturation of the steering actuator, instability. Robust control techniques can guarantee stability for a wide range of parameter changes, but performance may vary greatly. In summary, although there has been a significant amount of research on lateral control over the last 40 years, the "ideal" solution still does not exist. Therefore, this dissertation makes a contribution by exploring the merits of robust controllers which

Design Requirements

- minimize lateral error
- function over a wide range of parameter changes (velocity, cornering stiffness, mass, etc.)
- good ride comfort
- graceful degradation and robustness to sensor, actuator, and controller faults

Difficulties

- developing a suitable model is tough, most controller designs are performed using a linearized two DOF bicycle model. Higher order, nonlinear models are used for simulation purposes. These include terms for roll steering and tire models. Still, these models are not valid for all modes of operation. The nonlinearities are also difficult to deal with using modern control techniques.
- system is time varying (parameters like velocity, cornering stiffness, mass, etc.).
- some parameters are not easily measured, like cornering stiffness and lateral velocity, requiring some sort of estimation.

Table 2.2: Summary of Controller Design Requirements and Difficulties

can overcome many of the problems associated with other controllers, while still using a fixed control law.

The next chapter discusses robust control of plants with unstructured additive perturbations in the frequency-domain.

Program & Researcher	**YEAR**	**Sim/HW**	**Control Theory Used**
RCA	1957	HW	not available
GM & RCA,Gardels	late 1950's	HW	velocity dependent controller
Japan	1973	HW	PD lateral control, non-linear gain curve to handle curves
Britain, Giles	mid 1960's	HW	not available
S. Pasternak	1971		controllability, observability, optimal control.
Bonderson	1974		optimal control, performance index penalizes lateral displacement, lateral acceleration, and yaw acceleration
OSU, Fenton	1967-69	HW	lead compensator, bicycle model + roll steer term
OSU, Fenton	early 1970's	HW	single and dual loop compensators, root locus techniques
OSU, Fenton	late 1970's	HW & Sim	velocity-dependent controller, lead-lag compensator, root locus techniques, digital version later implemented (discretized analog version), analog computer simulations
OSU, Fenton	mid 1980's	Sim	optimal (LQR) controller with gain-scheduling, traditional observer
Daimler-Benz AG/H. Christ	mid 1970's	Sim & HW	lateral control of a bus, investigated proportional control (lateral error - both front & rear), discuss LQR (too difficult to implement), implement pole placement with observer
MIT, Shladover	1976-78	Sim	Wiener optimal filter, optimal LQR, reduced state feedback, bicycle model, AGT research
PATH, Peng & Tomizuka	1989- present	Sim & HW	frequency-shaped linear quadratic, sensitivity reduction, feedforward preview control (road curvature), PID for comparison purposes
PATH, Matsumoto & Tomizuka	1990	Sim & HW	lateral velocity & yaw-rate as control inputs, 4 wheel steering, optimal control w/gain-scheduling, feedforward term for centrifugal force

Program & Researcher	**YEAR**	**Sim/HW**	**Control Theory Used**
PATH, Patwardhan & Tomizuka	1992	Sim & HW	sensor fault detection with observers
PATH, Sheikhoslam & Desoer	1992	Sim	nonlinear controller, similar to computed torque control
PATH, Hessburg & Tomizuka	1994	Sim & HW	Fuzzy Logic
J. Ackermann	1990	Sim	decoupling yaw and lateral acceleration modes at front axle with yaw-rate feedback, bicycle model
J. Ackermann	1990	Sim	simultaneous pole placement for extremal plants, bicycle model
J. Ackermann	1992	Sim	velocity independent yaw eigenvalues via gain-scheduling feedback to rear wheels
J. Ackermann	1994	Sim	sliding mode control
Mitsubishi, Hayafune	1990	Sim & HW	trajectory generator (steering angle and velocity) for lane changes which ensures good passenger ride comfort
CMU, Pommerlau	1990-present	Sim & HW	vision-based neural networks, implemented on HMMWV
E.D. Dickmann	1990- present	Sim & HW	vision-based road following using classical vision techniques (edge detection, stripe following)
NIST	1992- present	Sim & HW	vision-based neural networks, implemented on HMMWV
Sandia, Byrne	1993	Sim	model-reference adaptive control

Table 2.3: Summary of Lateral Control Techniques

Chapter 3

Robust Control

This chapter deals with the design of robust controllers in the face of frequency-domain additive unstructured uncertainty. However, many of the techniques involving interpolation for additive perturbations may also be applied to systems with multiplicative perturbations. The limitations of the current theory are discussed, and methods to extend it are presented. New methods are presented for handling plants with poles on the jw axis and plants with unstable poles with multiplicity greater than 1. The next section reviews the basic theory of robust control given unstructured additive perturbations in the frequency-domain.

3.1 The Robust Stability Condition

Consider the class of single-input single-output (SISO) plants, in the simple closed-loop system shown in Figure 3.1. The transfer function of the controller is denoted by $c(s)$ while the transfer function of the SISO plant is denoted by $p(s)$. If the uncertainty of the plant is modeled as unstructured additive perturbations, the following definition is a natural way

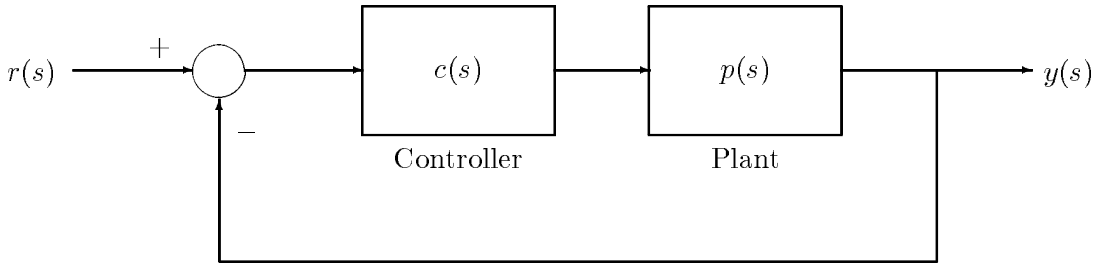


Figure 3.1: Closed-Loop System

to describe the variations of the frequency response of the system.

Definition 3.1 [46]: A transfer function $p(s)$ is said to be in the class $C(p_0(s), r(s))$ if

1. $p(s)$ has the same number of unstable poles as that of $p_0(s)$,

2. $|p(jw) - p_0(jw)| \leq |r(jw)|, |r(jw)| > 0, \forall w \in \mathbb{R}$

In definition 3.1, $p_0(s)$ represents the model of the nominal plant dynamics, while $p(s)$ denotes the actual plant dynamics. Therefore, $r(s)$ characterizes the uncertainty of the nominal plant model. It is assumed that $r(s)$ is a stable proper rational function. Given a compensator $c(s)$ which stabilizes $p_0(s)$, the conditions for $c(s)$ to be a robust stabilizer for all the plants in the class $C(p_0(s), r(s))$ can be derived. The first step is to define the plant as the sum of the nominal plant and a perturbation $\delta p(s)$.

$$p(s) = p_0(s) + \delta p(s);\ |\delta p(jw)| \leq |r(jw)| \quad \forall w \in \mathbb{R} \tag{3.1}$$

From the assumption that $c(s)$ stabilizes $p_0(s)$, one can write

$$p_0(jw)c(jw) + 1 \neq 0 \, \forall w \in \mathbb{R} \tag{3.2}$$

Because the closed-loop system is stable, the Nyquist stability criterion states that $c(s)p_0(s)$ has the correct encirclements of the -1 point. If $p_0(s)$ and $p(s)$ have the same number of unstable poles and thus the same number of encirclements, then

$$1 + p(jw)c(jw) \tag{3.3}$$

can be written as

$$1 + [p_0(jw) + \delta p(jw)] \, c(jw) \tag{3.4}$$

$$1 + p_0(jw)c(jw) + \delta p(jw)c(jw) \tag{3.5}$$

Factoring out the term $(1 + p_0(jw)c(jw))$ yields

$$[1 + p_0(jw)c(jw)] \left[1 + (1 + p(jw)c(jw))^{-1} c(jw)\delta(jw)\right] \tag{3.6}$$

From the small gain theorem [30], one is guaranteed that the number of encirclements of the -1 point will not change with the perturbed plant as long as

$$\sup_w \left| (1 + p_0(jw)c(jw))^{-1} c(jw)p(jw) \right| < 1 \tag{3.7}$$

$$\text{where } \sup_w \text{ is the supremum for all } w$$

A sufficient condition for robust stability is then described by (3.8).

$$\left\| [1 + p_0(s)c(s)]^{-1} \, c(s)r(s) \right\|_\infty < 1 \tag{3.8}$$

$$\text{where } \|F(s)\|_\infty \equiv \sup_w |F(jw)|$$

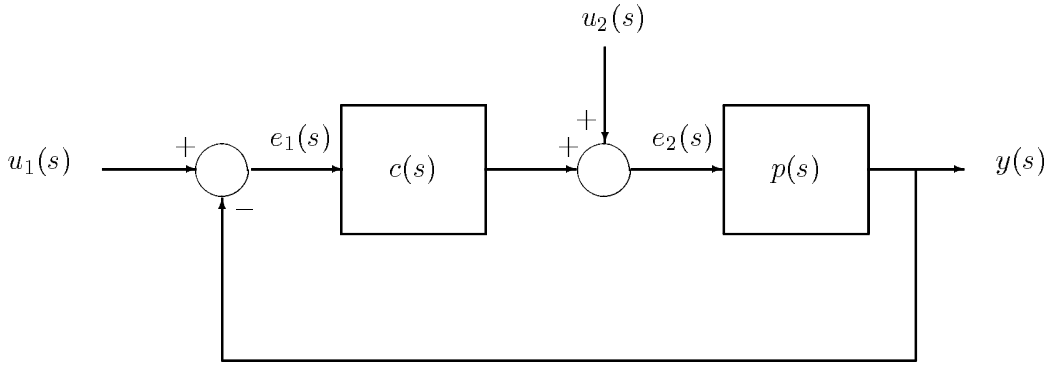


Figure 3.2: General closed-loop system

If the q parameterization, originally described in [82], is introduced, where

$$q(s) = \frac{c(s)}{1 + p_0(s)c(s)} \tag{3.9}$$

The robust stability condition can be written as

$$\|q(s)r(s)\|_\infty < 1 \tag{3.10}$$

In addition to the robust stability condition described by (3.10), it is also desirable to guarantee internal stability described next.

The standard feedback configuration of a general closed-loop system is shown in Figure 3.2. The transfer matrix between the two inputs and two outputs is shown below.

$$\begin{pmatrix} e_1(s) \\ e_2(s) \end{pmatrix} = \begin{pmatrix} [1 + p(s)c(s)]^{-1} & -p(s)[1 + p(s)c(s)]^{-1} \\ c(s)[1 + p(s)c(s)]^{-1} & [1 + p(s)c(s)]^{-1} \end{pmatrix} \begin{pmatrix} u_1(s) \\ u_2(s) \end{pmatrix} \tag{3.11}$$

The closed-loop system is internally stable if all of the transfer functions in (3.11) are bounded-input bounded-output stable (BIBO stable $\Leftrightarrow$ H^∞ functions). This assures that any bounded signal injected at $u_1(s)$ or $u_2(s)$ leads to a bounded response at any other point in the system. Applying the q parameterization to (3.11) yields

$$\begin{pmatrix} e_1(s) \\ e_2(s) \end{pmatrix} = \begin{pmatrix} 1 - p(s)q(s) & -p(s)\left[1 - p(s)q(s)\right] \\ q(s) & 1 - p(s)q(s) \end{pmatrix} \begin{pmatrix} u_1(s) \\ u_2(s) \end{pmatrix} \tag{3.12}$$

Based on (3.12), the conditions on $q(s)$ for the internal stability of the system are:

1. $q(s) \in H^\infty$
2. $q(s)$ must have zeros at the poles of $p(s)$ in the right-half-plane (RHP)
3. $p(s)q(s)$ must interpolate to 1 at the poles of $p(s)$ in the RHP

The first two conditions can be easily met by introducing the Blaschke product $B(s)$, where

$$B(s) = \prod_{i=1}^{m} \left(\frac{\alpha_i - s}{\bar{\alpha}_i + s} \right), |B(jw)| \equiv 1 \; \forall w, \; i = 1, ..., m \tag{3.13}$$

For this application, the α_i are the poles of $p(s)$ in Re $s > 0$. Also assume that all the unstable poles are simple, and $p(s)$ has no poles on the jw axis. Both these assumptions will be relaxed later. Defining $q(s)$ as

$$q(s) = \tilde{q}(s)B(s) \text{ where } \tilde{q}(s) \in H^\infty \tag{3.14}$$

ensures that the first two conditions are met.

If $\tilde{p}(s)$ is defined as

$$\tilde{p}(s) = p_0(s)B(s) \tag{3.15}$$

then

$$p_0(s)q(s) = \tilde{p}_0(s)\tilde{q}(s) \tag{3.16}$$

To ensure that condition 3 for internal stability is met, $\tilde{q}(s)$ must satisfy the interpolation conditions

$$\tilde{q}(\alpha_i) = \frac{1}{\tilde{p}_0(\alpha_i)} \tag{3.17}$$

Because the magnitude of $B(s)$ is equal to 1 for all w, the robust stability condition (3.10) can be written as

$$\|\tilde{q}(s)r_m(s)\|_\infty < 1 \tag{3.18}$$

where $r_m(s)$ is a minimum phase H^∞ function such that

$$|r_m(jw)| = |r(jw)| \tag{3.19}$$

Introduce the function

$$u(s) = \tilde{q}(s)r_m(s) \tag{3.20}$$

and the robust stability condition can be written as

$$\|u(s)\|_\infty < 1 \tag{3.21}$$

Equation (3.21) can only be satisfied if $u(s)$ is a strictly bounded real (SBR) function. The interpolation conditions on $u(s)$ are

$$u(\alpha_i) = \tilde{q}(\alpha_i)r_m(\alpha_i) = \frac{r_m(\alpha_i)}{\tilde{p}_0(\alpha_i)} \tag{3.22}$$

The robust stability problem for additive unstructured perturbations reduces to an equivalent interpolation problem of finding a strictly bounded real (SBR) function which interpolates at the unstable poles of the nominal plant in the RHP. This interpolation problem

is often referred to as the Nevanlinna-Pick interpolation problem, and is discussed in great detail in the next section.

The interpolation conditions described by equation (3.22) are for the case with no multiplicity. For interpolation conditions with multiplicity m, constraints are put on the derivatives of $u(s)$ at α_i. These constraints are derived below:

$$1 - \tilde{q}(s)\tilde{p}_0(s) = (s - \alpha_i)^m \frac{n(s)}{d(s)} \tag{3.23}$$

The first $(m-1)$ derivatives of equation (3.23) are equal to zero at $s = \alpha_i$. Therefore, equation (3.23) may be written as

$$\frac{d^i}{ds^i}\left[1 - \tilde{q}(s)\tilde{p}_0(s)\right]\Big|_{s=\alpha_i} = 0 \text{ where } i = 1, 2, 3, ..., (m-1) \tag{3.24}$$

Reorganizing equation (3.24) yields

$$\frac{d^i}{ds^i}\left[\tilde{q}(s)\right]\Big|_{s=\alpha_i} = \frac{d^i}{ds^i}\left[\frac{1}{\tilde{p}_0(s)}\right]\Big|_{s=\alpha_i} \text{ where } i = 1, 2, 3, ..., (m-1) \tag{3.25}$$

$u(s)$ is defined as

$$u(s) = \tilde{q}(s) r_m(s) \tag{3.26}$$

Therefore,

$$\frac{d^i}{ds^i}\left[u(s)\right]\Big|_{s=\alpha_i} = \frac{d^i}{ds^i}\left[\tilde{q}(s) r_m(s)\right]\Big|_{s=\alpha_i} \text{ where } i = 1, 2, 3, ..., (m-1) \tag{3.27}$$

Using equation (3.25), equation (3.27) may be written as

$$\frac{d^i}{ds^i}\left[u(s)\right]\Big|_{s=\alpha_i} = \frac{d^i}{ds^i}\left[\frac{r_m(s)}{\tilde{p}_0(s)}\right]\Big|_{s=\alpha_i} \text{ where } i = 1, 2, 3, ..., (m-1) \tag{3.28}$$

Equation (3.28) is important because it allows the calculation of the desired derivatives of $u(s)$ at the interpolation point from known information ($\tilde{q}(s)$ has not been constructed yet).

In addition to the interpolation conditions placed on the SBR function $u(s)$, constraints may also be placed on the relative degree of $u(s)$ to make the compensator proper. From (3.9), the controller can be expressed in terms of $q(s)$.

$$c(s) = \frac{q(s)}{1 - p_0(s)q(s)} \tag{3.29}$$

The controller is proper only if $q(s)$ is proper. From (3.14) and (3.20),

$$q(s) = \frac{B(s)u(s)}{r_m(s)} \tag{3.30}$$

In order for $q(s)$ to be proper, the relative degree of $u(s)$ must be greater than or equal to the relative degree of $r_m(s)$. The Nevanlinna-Pick interpolation discussed in the next section generally yields a function with relative degree 0. Therefore, if the relative degree of $r_m(s)$ is greater than 0, interpolation conditions at ∞ are required for $u(s)$ to have a relative degree greater than or equal to the relative degree of $r_m(s)$. These interpolation conditions at ∞ ensure that $q(s)$ is proper.

3.2 Interpolation Theory

Interpolation theory is used to generate a function $u(s)$ which meets the interpolation conditions $u(\alpha_i) = \beta_i$ for $i = 1...m$. Examples include Lagrange interpolation of rational transfer functions [38], interpolation of positive real functions [70, 81], and interpolation of rational Schur functions – the classical Nevanlinna-Pick problem [23, 54, 60]. Applications of the Nevanlinna-Pick interpolation problem in circuit and system theory are described in [23]. The robust control problem presented in the previous section requires interpolation with SBR functions. When the interpolation points are on the real axis, the Nevanlinna-

Pick approach yields SBR functions. Otherwise, equation (3.31) may be used to obtain bounded real functions from Schur functions.

$$w_{BR}(s) = \frac{1}{2}\left[w_{Schur}(s) + \bar{w}_{Schur}(\bar{s})\right] \tag{3.31}$$

A modified version of the Nevanlinna-Pick algorithm which directly gives bounded real functions is described in [28].

The interpolation of functions analytic in the unit circle with a modulus not greater than M is described in great detail in [77]. From [77], interpolation in the unit circle is reduced from the problem of finding a function $w(z)$ that takes on m given values, to finding a function $w_1(z)$ that takes on $m-1$ given values using the mapping below. The variable z will be used for functions analytic in the unit circle, while the variable s will be used for functions analytic in the right-half-plane.

$$w_1(\alpha_k) = w_k^{(1)} \equiv M^2 \frac{w_k^{(0)} - w_1^{(0)}}{M^2 - \bar{w}_1^{(0)} w_k^{(0)}} \left(\frac{1 - \bar{\alpha}_1 \alpha_k}{\alpha_k - \alpha_1}\right),\ k = 2, 3, ..., m \tag{3.32}$$

$$\text{where } w_0(\alpha_i) = w_i^{(0)}$$

This gives a recurrent method for the general solution to the problem shown by (3.33) and (3.34).

$$w_k^{(v)} \equiv w_v(\alpha_k) = M^2 \frac{w_k^{(v-1)} - w_v^{(v-1)}}{M^2 - \bar{w}_v^{(v-1)} w_{v-1}(z)} \left(\frac{1 - \bar{\alpha}_v z}{z - \alpha_v}\right),\ k = v+1, v+2, ..., m \tag{3.33}$$

$$w_v(z) \equiv M^2 \frac{w_{v-1}(z) - w_v^{(v-1)}}{M^2 - \bar{w}_v^{(v-1)} w_{v-1}(z)} \left(\frac{1 - \bar{\alpha}_v z}{z - \alpha_v}\right),\ w_0(z) \equiv w(z) \tag{3.34}$$

Equation (3.33) is used to reduce the interpolation problem recursively until there is only one interpolation point. Then equation (3.34) is used to back out to the solution, $w(z)$. Equations (3.32), (3.33), and (3.34) are concerned with interpolation of functions analytic

and with modulus not greater than M inside the unit circle. In order to generate bounded real functions analytic in the right-half-plane, the bilinear mapping

$$z = \frac{s-1}{s+1} \tag{3.35}$$

may be used to map the unit circle into the right-half-plane. Also, the bound M in equations (3.32), (3.33), and (3.34) equals 1.

Using (3.35) to modify (3.32) changes the analytic region to the right-half-plane and yields (3.36).

$$u_1(s) \equiv \frac{u_0(s) - u_1^{(0)}}{1 - \bar{u}_1^{(0)} u_0(s)} \left(\frac{s + \bar{\alpha}_1}{s - \alpha_1} \right) \left(\frac{\alpha_1 + 1}{\bar{\alpha}_1 + 1} \right), \; s = \alpha_k, \; k = 2, 3, ..., m \tag{3.36}$$

or

$$u_1(s) \equiv \frac{u_0(s) - u_1^{(0)}}{1 - \bar{u}_1^{(0)} u_0(s)} \left(\frac{s + \bar{\alpha}_1}{s - \alpha_1} \right) e^{j\theta}, \; s = \alpha_k, \; k = 2, 3, ..., m \tag{3.37}$$

$$\text{where } \theta = 2 \tan^{-1} \frac{\text{Im } \alpha}{1 + \text{Re } \alpha}$$

The $e^{j\theta}$ does not affect the magnitude of $u_1(s)$. Therefore, (3.37) may be written as (3.38) while still guaranteeing that $u_1(s)$ is a Schur function.

$$u_1(s) \equiv \frac{u_0(s) - u_1^{(0)}}{1 - \bar{u}_1^{(0)} u(s)} \left(\frac{s + \bar{\alpha}_1}{s - \alpha_1} \right) \tag{3.38}$$

The simpler form of equation (3.38) is typically used. The equations that characterize the general solution which is analytic inside the unit circle, (3.33) and (3.34), map to equations (3.39) and (3.40) for the general solution which is analytic in the right-half-plane.

$$u_k^{(v)} \equiv u_v(s)\big|_{s=\alpha_k} = \frac{u_{v-1}(s) - u_v^{(v-1)}}{1 - \bar{u}_v^{(v-1)} u_{v-1}(s)} \left(\frac{s + \bar{\alpha}_v}{s - \alpha_v} \right) \Bigg|_{s=\alpha_k}, \; k = v+1, v+2, ..., m \tag{3.39}$$

$$u_v(s) \equiv \frac{u_{v+1}^{(v)} + u_{v+1}(s) \left(\frac{s - \alpha_{v+1}}{s + \bar{\alpha}_{v+1}} \right)}{1 + \bar{u}_{v+1}^{(v)} \left(\frac{s - \alpha_{v+1}}{s + \bar{\alpha}_{v+1}} \right) u_{v+1}(s)} \tag{3.40}$$

α_1	α_2	$\cdots$	α_m	
$u_1^{(0)} = u_0(\alpha_1) = \beta_1$	$u_2^{(0)} = u_0(\alpha_2) = \beta_2$	$\cdots$	$u_m^{(0)} = u_0(\alpha_m) = \beta_m$	$u_0(s)$
	$u_2^{(1)} = u_1(\alpha_2)$	$\cdots$	$u_m^{(1)} = u_1(\alpha_m)$	$u_1(s)$
		$\ddots$	$\vdots$	$\vdots$
			$u_m^{(m-1)} = u_{m-1}(\alpha_m)$	$u_{m-1}(s)$

Figure 3.3: Fenyves Array

The interpolation problem bookkeeping is simplified by using the so called "Fenyves array" shown in Figure 3.3 [46].

Lemma 3.1 [27] There is a strictly bounded real (SBR) solution to the Nevanlinna-Pick problem if and only if all the elements of the Fenyves array are strictly less than one, i.e.

$$\left| u_k^{(v)} \right| < 1 \tag{3.41}$$

□

Lemma 3.2 [27] If an element of the Fenyves array is equal to one, then a bounded real (BR) solution exists only if the whole row of the Fenyves array has terms equal to one.

□

An equivalent method for determining the existence of a solution to the Nevanlinna-Pick problem is to look at the Pick matrix P, shown below [7].

$$P = \begin{bmatrix} \frac{1-\beta_1\bar{\beta}_1}{\alpha_1+\bar{\alpha}_1} & \cdots & \frac{1-\beta_1\bar{\beta}_m}{\alpha_1+\bar{\alpha}_m} \\ \vdots & \ddots & \vdots \\ \frac{1-\beta_m\bar{\beta}_1}{\alpha_m+\bar{\alpha}_1} & \cdots & \frac{1-\beta_m\bar{\beta}_m}{\alpha_m+\bar{\alpha}_m} \end{bmatrix} \tag{3.42}$$

Lemma 3.3 [7] The Nevanlinna-Pick problem is solvable for SBR functions, if and only if the matrix P is positive definite.

□

An application of the Nevanlinna-Pick interpolation theory applied to the robust control problem is described in Example 3.1.

Example 3.1 – Robust Control of Plant with Relative Degree 0

Given the following unstable plant,

$$p_0(s) = \frac{(1+s)(5+s)}{(2-s)(3-s)} \tag{3.43}$$

with the uncertainty of the plant given by

$$r(s) = 0.5\frac{(1+s)(5+s)}{(2+s)(3+s)} \tag{3.44}$$

Find the robustly stabilizing compensator.

Step 1 – Compute $r_m(s)$, $\tilde{p}_0(s)$, and $B(s)$. Looking at (3.43), the unstable poles are $\alpha_1 = 2$ and $\alpha_2 = 3$, therefore $B(s)$ can be written

$$B(s) = \frac{(2-s)(3-s)}{(2+s)(3+s)} \tag{3.45}$$

Given $B(s)$, $\tilde{p}_0(s)$ can be written

$$\tilde{p}_0(s) = p_0(s)B(s) = \frac{(1+s)(5+s)}{(2+s)(3+s)} \tag{3.46}$$

Since $r(s)$ is a stable minimum phase proper transfer function, $r_m(s) = r(s)$.

Step 2 – Find an SBR function $u(s)$ such that $u(\alpha_i) = \beta_i = \frac{r_m(\alpha_i)}{\tilde{p}_0(\alpha_i)}$ where α_i are the poles of $p_0(s)$ in the RHP. Solve the Nevanlinna-Pick problem. First calculate $u(\alpha_i)$ and then

$\alpha_1 = 2$	$\alpha_2 = 3$	
0.5	0.5	$u_0(s)$
	0	$u_1(s)$

Figure 3.4: Fenyves Array for Example 3.1

construct the Fenyves array.

$$u(\alpha_i) = \frac{r_m(\alpha_i)}{\tilde{p}_0(\alpha_i)} = 0.5 \tag{3.47}$$

Note that $u(\alpha_i)$ is a constant for this case because of the choice of $r_m(s)$. The corresponding Fenyves array constructed using (3.39) is shown in Figure 3.4. The SBR function $u_1(s)$ must interpolate to 0 at $s = 3$. Therefore, a suitable, yet arbitrary, choice of $u_1(s)$ is

$$u_1(s) = \frac{s-3}{4+s} \tag{3.48}$$

Using (3.40), $u_0(s)$ is

$$u(s) = u_0(s) = \frac{-s^2 + 16s - 4}{s^2 + 17s + 10} \tag{3.49}$$

Another possible solution for $u_0(s)$ which meets all of the interpolation conditions is $u_0(s) = 0.5$. This choice is obtained if $u_1(s) = 0$ is used in the Fenyves Array, or by inspection. This solution however yields an infinite gain compensator, which stabilizes the nominal plant, but is impractical to implement.

Step 3 – Compute $q(s)$.

$$q(s) = \frac{B(s)u(s)}{r_m(s)} = 2\frac{(2-s)(3-s)}{(1+s)(5+s)}\left(\frac{-s^2 + 16s - 4}{s^2 + 17s + 10}\right) \tag{3.50}$$

Step 4 – Compute $c(s)$.

$$c(s) = \frac{q(s)}{1 - p_0(s)q(s)} = \frac{2}{3}\frac{(-s^2 + 16s - 4)}{(1+s)(5+s)} \tag{3.51}$$

The characteristic polynomial of the nominal closed-loop system is $s^2 + 17s + 10$, which is Hurwitz, as expected.

□

In the previous example, the relative degree of $r_m(s)$ was equal to 0. Therefore, no interpolation points at infinity were required. The case requiring an interpolation point at infinity is shown in Example 3.2.

Example 3.2 – Robust Control with Relative Degree of $r_m(s) = 1$

Given the following unstable plant,

$$p_0(s) = \frac{(1+s)}{(2-s)(3-s)} \tag{3.52}$$

with the uncertainty of the plant given by

$$r(s) = 0.5\frac{(1+s)}{(2+s)(3+s)} \tag{3.53}$$

Find the robustly stabilizing compensator.

Step 1 – Compute $r_m(s)$, $\tilde{p}_0(s)$, and $B(s)$. Looking at (3.52), the unstable poles are $\alpha_1 = 2$, $\alpha_2 = 3$, therefore $B(s)$ can be written

$$B(s) = \frac{(2-s)(3-s)}{(2+s)(3+s)} \tag{3.54}$$

Given $B(s)$, $\tilde{p}_0(s)$ can be written

$$\tilde{p}_0(s) = p_0(s)B(s) = \frac{(1+s)}{(2+s)(3+s)} \tag{3.55}$$

Since $r(s)$ is a stable minimum phase proper transfer function, $r_m(s) = r(s)$. Note that the relative degree of $r_m(s)$ is 1, which adds an additional interpolation condition at infinity.

$\alpha_1 = 2$	$\alpha_2 = 3$	$\alpha_3 = \infty$	
0.5	0.5	0	$u_0(s)$
	0	-0.5	$u_1(s)$
		-0.5	$u_2(s)$

Figure 3.5: Fenyves Array for Example 3.2

Step 2 – Find an SBR function $u(s)$ such that $u(\alpha_i) = \beta_i = \frac{r_m(\alpha_i)}{\tilde{p}_0(\alpha_i)}$ where α_i are the poles of $p_0(s)$ in the RHP. Solve the Nevanlinna-Pick problem. First calculate $u(\alpha_i)$ and then construct the Fenyves array.

$$u(\alpha_i) = \frac{r_m(\alpha_i)}{\tilde{p}_0(\alpha_i)} = 0.5 \tag{3.56}$$

Note that $u(\alpha_i)$ is a constant for this case because of the choice of $r_m(s)$. Also, because $r_m(s)$ has relative degree 1, there is an additional interpolation point at infinity. The corresponding Fenyves array is shown in Figure 3.5. The SBR function $u_2(s)$ must interpolate to -0.5 at infinity. Therefore, a suitable, yet arbitrary, choice of $u_2(s)$ is

$$u_2(s) = \frac{-s-1}{2s+3} \tag{3.57}$$

Using (3.40), $u_1(s)$ is

$$u_1(s) = \frac{(1+s)(3-s)}{(2s+3)(s+3)} \tag{3.58}$$

Using (3.40), $u_0(s)$ is

$$u(s) = u_0(s) = \frac{21s^2 + 25s + 6}{3s^3 + 30s^2 + 53s + 30} \tag{3.59}$$

Note that the relative degree of $u_0(s)$ is 1 which is a result of the interpolation condition at infinity, ensuring that $q(s)$ is proper.

Step 3 – Compute $q(s)$.

$$q(s) = \frac{B(s)u(s)}{r_m(s)} = 2\frac{(2-s)(3-s)}{(1+s)}\left(\frac{21s^2+25s+6}{3s^3+30s^2+53s+30}\right) \tag{3.60}$$

Step 4 – Compute $c(s)$.

$$c(s) = \frac{q(s)}{1-p_0(s)q(s)} = \frac{2}{3}\frac{(21s^2+25s+6)}{(1+s)(1+s)} \tag{3.61}$$

The characteristic polynomial of the nominal closed-loop system is $3s^3+30s^2+53s+30$, which is Hurwitz, as expected.

□

The previous two examples have shown how Nevanlinna-Pick interpolation can be used to solve the robust control problem with unstructured additive perturbations. Limitations of the approach include not being able to handle poles on the jw axis and poles with multiplicity. An approach for handling one type of pole on the jw axis, specifically a single integrator, is outlined in [46]. This involves dividing the integrator out of the nominal plant and the plant uncertainty, and adding an interpolation point at the origin. The interpolation point at the origin must be handled like the point at infinity in Example 3.2. The interpolation points at zero and infinity must be in the last two columns of the Fenyves array so that the last function, $u_{n-2}(s)$, can be crafted using the equation below.

$$u_{n-2}(s) = \frac{\delta s+\lambda}{s+1} \tag{3.62}$$

By selecting δ and λ, the interpolation conditions for u_{n-2} at zero and infinity can be met, provided there is a solution to the Nevanlinna-Pick problem.

One approach for handling interpolation points on the jw axis is described in [48]. This involves using a mapping which interpolates positive-real functions on the jw axis. Then

the positive real function is mapped to a bounded real function. A method for handling interpolation points with multiplicity on the jw axis was not discussed.

A general approach for handling interpolation points on the jw axis is outlined in the next section. This approach is also able to deal with multiplicity, which is described in a subsequent section.

3.3 Interpolation Points on the jw Axis

The classical solution of the Nevanlinna-Pick interpolation problem is concerned with interpolating functions analytic (having no poles) in the unit circle or the right-half-plane. Functions analytic in the unit circle are useful in discrete-time problems, while functions analytic in the right-half-plane are useful for continuous-time problems. The bilinear mapping below is typically used to map the unit circle into the right-half-plane.

$$z = \frac{s-1}{s+1} \tag{3.63}$$

From the previous section, the mapping used for interpolation of functions analytic in the right-half-plane is given by (3.64).

$$u_1(s) = \frac{u_0(s) - u_1^{(0)}}{1 - \bar{u}_1^{(0)} u_0(s)} \left(\frac{s + \bar{\alpha}_1}{s - \alpha_1} \right) \tag{3.64}$$

Although this mapping is very useful, it has several shortcomings. It cannot be used to interpolate points with multiplicity without modification. It also cannot be used to interpolate on the jw axis. Finally, although the functions generated are analytic in the right-half-plane, it is sometimes desirable to generate functions that are analytic in a region other than the right-half-plane. Based on this motivation, a modified mapping is proposed

to overcome these shortcomings. Instead of mapping the unit circle into the right-half-plane, a modified version of (3.63) is used to map the unit circle into a region D which includes the right-half-plane. This involves modifying the bilinear mapping (3.63) as shown below.

$$z = \frac{s - (1 - \epsilon)}{s + (1 + \epsilon)} \tag{3.65}$$

Applying (3.65) to (3.64) yields

$$u_1(s) = \frac{u_0(s) - u_1^{(0)}}{1 - \bar{u}_1^{(0)} u_0(s)} \left(\frac{s + \bar{\alpha}_1 + 2\epsilon}{s - \alpha_1} \right) \tag{3.66}$$

Lemma 3.4 If $u_0(s)$ is a Schur function and $u_0(\alpha_1) = u_1^{(0)}$, then $u_1(s)$ is a Schur function.

□

Proof of Lemma 3.4 Because of the Maximum Modulus Theorem [64], the maximum value of the function occurs on the boundary of the analytic region. This occurs at $s = jw - \epsilon$ for (3.66). Therefore, it is necessary to define a new norm, the H_ϵ^∞ norm, as shown below

$$\|u(s)\|_\infty^\epsilon \equiv \sup |u(s)|\,,\ s = jw - \epsilon,\ \forall w \tag{3.67}$$

Because of the Maximum Modulus Theorem, the following relationship between the H_ϵ^∞ norm and the standard H^∞ norm exists for functions analytic in the region D (see Figure 3.6).

$$\|u(s)\|_\infty \leq \|u(s)\|_\infty^\epsilon \tag{3.68}$$

The use of the H_ϵ^∞ norm makes the robust stability condition more conservative as ϵ grows in magnitude. By increasing ϵ, the likelihood of a solution existing is decreased.

$u_1(s)$ is a Schur function if and only if

1. $u_1(s) \in H^\infty$

2. $1 - \bar{u}_1(s)u_1(s) \geq 0,\ s = jw - \epsilon;\ \forall w \in \mathbb{R}$

By assumption $|\bar{u}_1^{(0)}| < 1$ and $|u_0(s)| \leq 1$. Therefore, $1 - \bar{u}_1^{(0)}u_0(s) \neq 0$ for $s = jw - \epsilon$, and $\forall w$. The pole $s = \alpha_1$ in the RHP is canceled by the interpolation condition at α_1. This proves $u_1(s) \in H^\infty$.

Next, for $s = jw - \epsilon$, $1 - \bar{u}_1(s)u_1(s)$ can be written as

$$1 - \bar{u}_1(s)u_1(s) = \frac{\left(1 - |u_1^{(0)}|^2\right)\left(1 - |u_0(s)|^2\right)}{\left|1 - \bar{u}_1^{(0)}u_0(s)\right|^2} \tag{3.69}$$

because

$$\left|\frac{s + \bar{\alpha}_1 + 2\epsilon}{s - \alpha_1}\right| \equiv 1 \text{ when } s = jw - \epsilon;\ \forall w \in \mathbb{R} \tag{3.70}$$

Equation (3.69) is nonnegative since by assumption $|u_1^{(0)}| < 1$ and $|u_0(s)| \leq 1$.

□

The inverse mapping is given by (3.71)

$$u_0(s) = \frac{u_1^{(0)} + u_1(s)\left(\frac{s - \alpha_1}{s + \bar{\alpha}_1 + 2\epsilon}\right)}{1 + u_1(s)\bar{u}_1^{(0)}\left(\frac{s - \alpha_1}{s + \bar{\alpha}_1 + 2\epsilon}\right)} \tag{3.71}$$

Lemma 3.5 If $u_1(s)$ is a Schur function, then $u_0(s)$ is also a Schur function. In addition, $u_0(\alpha_1) = u_1^{(0)}$ independently of $u_1(s)$.

□

Proof of Lemma 3.5 $u_0(s)$ is a Shur function if and only if:

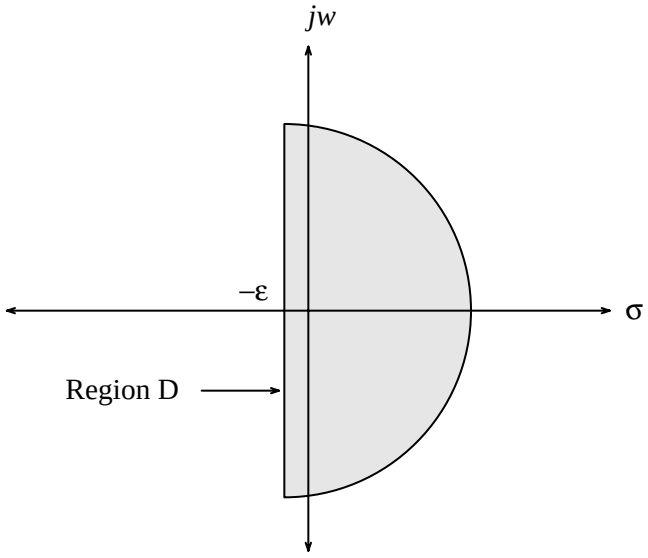


Figure 3.6: Analytic Region D

1. $u_0(s) \in H^\infty$

2. $1 - \bar{u}_0(s)u_0(s) \geq 0,\ s = jw - \epsilon;\ \forall w \in \mathbb{R}$

From the fact that $u_1(s)$ is Schur, $|\bar{u}_1^{(0)}| < 1$, and the Maximum Modulus Theorem,

$$1 + u_1(s)\bar{u}_1^{(0)}\left(\frac{s - \alpha_1}{s + \bar{\alpha}_1 + 2\epsilon}\right) \neq 0,\ s = jw - \epsilon;\ \forall w \in \mathbb{R} \tag{3.72}$$

Therefore, $u_0(s) \in H^\infty$.

Next, for $s = jw - \epsilon$, $1 - \bar{u}_0(s)u_0(s)$ can be written

$$1 - \bar{u}_0(s)u_0(s) = \frac{\left(1 - |\bar{u}_1^{(0)}|^2\right)\left(1 - |u_1(s)|^2\right)}{\left|1 + \bar{u}_1^{(0)}u_1(s)\right|^2},\ s = jw - \epsilon;\ \forall w \in \mathbb{R} \tag{3.73}$$

which is nonnegative because $|u_1^{(0)}| < 1$ and $u_1(s) \leq 1$ by assumption.

□

An application of the mapping described by (3.66) to the robust control problem is shown in Example 3.3. By using this mapping, it is now possible to handle poles on the jw axis.

Example 3.3 – Robust Control of a Plant with Poles on the jw Axis

Given the following plant,

$$p_0(s) = \frac{(s+3)(s+4)(s+5)}{(s^2+1)(s+2)} \tag{3.74}$$

with the uncertainty of the plant given by

$$r(s) = \frac{(s+1)(s+3)(s+4)(s+5)}{(s^2+1)(s+2)(s+2)} \tag{3.75}$$

Find the robustly stabilizing compensator. It is assumed that the pole on the jw axis is preserved under perturbations.

Step 1 – Compute $r_m(s)$, $\tilde{p}_0(s)$, and $B(s)$. Write

$$r(s) = \frac{r'_m(s)}{(s^2+1)} = \frac{(s+1)(s+3)(s+4)(s+5)}{(s^2+1)(s+2)(s+2)} \tag{3.76}$$

$$r'_m(s) = \frac{(s+1)(s+3)(s+4)(s+5)}{(s+2)(s+2)} \tag{3.77}$$

where $r'_m(s)$ is a minimum phase function. Then, $\tilde{p}_0(s)$ can be written

$$\tilde{p}_0(s) = (s^2+1)p_0(s) = \frac{(s+3)(s+4)(s+5)}{(s+2)} \tag{3.78}$$

Because the plant has no poles in the open RHP, $B(s) = 1$.

Step 2 – Find an SBR function $u(s)$ such that $u(\alpha_i) = \beta_i = \frac{r'_m(\alpha_i)}{\tilde{p}_0(\alpha_i)}$ where α_i are the poles of $p_0(s)$ in the closed RHP. Solve the Nevanlinna-Pick problem.

For this example, there are no interpolation points in the open RHP because the plant has no poles in the open RHP. However, there are two interpolation points on the jw axis, $\alpha_1 = j$ and $\alpha_2 = -j$.

$$u(\alpha_i) = \frac{\alpha_i + 1}{\alpha_i + 2} \tag{3.79}$$

$\alpha_1 = j$	$\alpha_2 = -j$	
$\frac{3+j}{5}$	$\frac{3-j}{5}$	$u_0(s)$
	$\frac{10(\epsilon-j)}{17+j6}$	$u_1(s)$

Figure 3.7: Fenyves Array for Example 3.3

evaluating (3.79), one obtains

$$u(j) = \frac{3+j}{5}, \;\; u(-j) = \frac{3-j}{5} \tag{3.80}$$

Using (3.80) and (3.66), the corresponding Fenyves array is shown in Figure 3.7. In order for a solution to exist, all the elements of the Fenyves array must be less than or equal to 1 in magnitude. This puts the following constraint on the size of ϵ.

$$\epsilon \leq \frac{3}{2} \tag{3.81}$$

Solving for $u_1(s)$ in terms of ϵ, one obtains

$$u(s) = u_0(s) = \frac{s[45 + 50\epsilon - j15] + [90\epsilon - j45 - 15 + j20\epsilon]}{s[75 + 30\epsilon - j10\epsilon] + [160\epsilon - j75 + j30\epsilon]} \tag{3.82}$$

Using (3.81), one can safely choose an $\epsilon = 0.1$ and be guaranteed that a solution to the Nevanlinna-Pick interpolation problem exists. This yields the following $u(s)$.

$$u(s) = \frac{s[50 - j15] + [-6 - j43]}{s[78 - j] + [16 - j72]} \tag{3.83}$$

Because $u(s)$ is a Schur function, equation (3.31) must be used to generate an SBR function. This yields

$$u_0(s) = \frac{3915s^2 + 1455s + 3000}{6085s^2 + 2640s + 5440} \tag{3.84}$$

Step 3 – Compute $q(s)$ using equation (3.85).

$$q(s) = \frac{B(s)u(s)}{r(s)} = \frac{(3915s^2 + 1455s + 3000)(s^2 + 1)(s + 2)^2}{(6085s^2 + 2640s + 5440)(s + 1)(s + 3)(s + 4)(s + 5)} \tag{3.85}$$

Step 4 – Compute $c(s)$. The compensator is shown in equation (3.86).

$$c(s) = \frac{q(s)}{1 - p_0(s)q(s)} = \frac{(3915s^2 + 1455s + 3000)(s + 2)^2}{(2170s - 560)(s + 3)(s + 4)(s + 5)} \tag{3.86}$$

Note that the stabilizing compensator is unstable, due to the pole located at $s = 0.258$. However, if one checks the four transfer functions in (3.11), the system will still be internally stable as expected. The poles of the closed-loop nominal system are at -1 and $-0.2169 \pm j0.9203$.

□

The next section presents an approach for handling plants with a multiplicity of poles in the right-half-plane.

3.4 Interpolation Points with Multiplicity

For the case where the α_i's are distinct, equations (3.36) and (3.37) are easily applied. However, for the case where the α_i's are not distinct, equations (3.36) and (3.37) may not be used. The case of multiplicity equal to two is described by Walsh in [77]. When the α_i are not distinct, constraints are forced on the derivatives of $u(s)$. A simple approach to handle points with multiplicity using a modified Fenyves array, is described by Theorem 3.1.

Theorem 3.1 The solution to the Nevanlinna-Pick interpolation problem with multiplicity m is obtained by constructing a modified Fenyves array shown in Figure 3.8 whose

terms are given by equation (3.87).

$$u_k(\alpha_1) = \frac{(\bar{\alpha}_1 + \alpha_1)}{1 - |u_{k-1}(\alpha_1)|^2} u'_{k-1}(\alpha_1),\ k = 1, 2, 3, ..., m-1 \tag{3.87}$$

where $u'_{k-1}(\alpha_1)$ represents the first derivative of $u_{k-1}(s)$ evaluated at α_1.

Once the Fenyves array is constructed, the desired function $u(s)$ is obtained in the usual manner.

□

Note: Although (3.87) gives the formula for constructing the modified Fenyves Array, the functions $u_k(s)$ are not constructed until after the Fenyves Array. However, (3.87) may be evaluated by applying the mapping (3.39) and differentiating. Interpolation with multiplicity m requires matching the first $m-1$ derivatives of $u_0(s)$ with the derivatives of the function that $u_0(s)$ will multiply. In other words, each additional interpolation point with multiplicity places constraints on the next higher order derivative of $u_0(s)$.

Proof of Theorem 3.1: Two different methods may be used for the proof by induction. The first approach involves assuming that the interpolation points are separated by a small increment ϵ, and then taking the limit as ϵ approaches zero. The second approach involves applying L'Hopital's Rule to evaluate terms in the Fenyves Array. The proof by induction using the limit approach is outlined in Figure 3.8, and is discussed next.

The mapping typically used to construct the Fenyves array is

$$u_v(s)\big|_{s=\alpha_i} = \frac{u_{v-1}(s) - u_v^{(v-1)}}{1 - \bar{u}_v^{(v-1)} u_{v-1}(s)} \left(\frac{s + \bar{\alpha}_v}{s - \alpha_v}\right)\Bigg|_{s=\alpha_i} \tag{3.88}$$

Assume that the α_i are separated by a small increment ϵ, so that

$$\alpha_{j+1} - \alpha_j = \epsilon,\ j = 1, 2, ..., (m-1) \tag{3.89}$$

α_1	$\cdots$	α_j	α_{j+1}	$\cdots$	α_m	
$u_0(\alpha_1)$					$u_0(\alpha_m)$	$u_0(s)$
	$\ddots$					$\vdots$
		$u_{j-1}(\alpha_j)$	$u_{j-1}(\alpha_j+\epsilon)$			$u_{j-1}(s)$
			$u_j(\alpha_j+\epsilon)$			$u_j(s)$
				$\ddots$		$\vdots$
					$u_{m-1}(\alpha_m)$	$u_{m-1}(s)$

where $\alpha_{j+1} - \alpha_j = \epsilon$ and $\alpha_m = \alpha_1 + (m-1)\epsilon$

Figure 3.8: Modified Fenyves Array

Then, any diagonal element of the Fenyves array, $u_j^{(j-1)}$, can be evaluated using equation (3.88),

$$u_j(\alpha_j+\epsilon) = \frac{u_{j-1}(\alpha_j+\epsilon) - u_{j-1}(\alpha_j)}{1 - \bar{u}_{j-1}(\alpha_j)u_{j-1}(\alpha_j+\epsilon)} \left(\frac{\alpha_j + \epsilon + \bar{\alpha}_j}{\alpha_j + \epsilon - \alpha_j} \right) \tag{3.90}$$

$$u_j(\alpha_j+\epsilon) = \frac{u_{j-1}(\alpha_j+\epsilon) - u_{j-1}(\alpha_j)}{\epsilon} \left(\frac{\alpha_j + \epsilon + \bar{\alpha}_j}{1 - \bar{u}_{j-1}(\alpha_j)u_{j-1}(\alpha_j+\epsilon)} \right) \tag{3.91}$$

Taking the limit as ϵ approaches zero in equation (3.91) and noting that all $\alpha_i = \alpha_1$ yields

$$u_j(\alpha_1) = \frac{\bar{\alpha}_1 + \alpha_1}{1 - |u_{j-1}(\alpha_1)|^2} u'_{j-1}(\alpha_1) \tag{3.92}$$

which is the same result stated by equation (3.87).

□

It should be noted that this approach can only applied when the derivatives of $u_0(s)$ have meaning, i.e. are finite. Because $u_0(s)$ is analytic in the RHP, this should never be a

problem. The case where a derivative is zero is easily handled. This case yields a row of the modified Fenyves Array where all of the terms are zero. A bounded real function with a multiplicity of zeros at the same point can be constructed by inspection. This is shown in example 3.4.

Example 3.4 – Robust Control with Unstructured Uncertainty, $m = 3$

Given the following unstable plant,

$$p_0(s) = \frac{(s+3)(s+2)^2}{(s-1)^3} \tag{3.93}$$

with the uncertainty of the plant given by

$$r(s) = \frac{1}{2}\frac{(s+3)(s+2)^2}{(s+1)^3} = \frac{1}{2}\tilde{p}_0(s) \tag{3.94}$$

Find the robustly stabilizing compensator.

Step 1 – Compute $r_m(s)$, $\tilde{p}_0(s)$, and $B(s)$. Looking at (3.93), the unstable poles are $\alpha_1 = 1$, $\alpha_2 = 1$, and $\alpha_3 = 1$. Therefore, $B(s)$ can be written

$$B(s) = \left(\frac{s-1}{s+1}\right)^3 \tag{3.95}$$

Given $B(s)$, $\tilde{p}_0(s)$ can be written

$$\tilde{p}_0(s) = \frac{(s+3)(s+2)^2}{(s+1)^3} \tag{3.96}$$

Since $r(s)$ is a minimum phase proper transfer function, $r_m(s) = r(s)$.

Step 2 – Find an SBR function $u(s)$ such that $u(\alpha_i) = \frac{r_m(\alpha_i)}{\tilde{p}_0(\alpha_i)}$ where α_i are the poles of $p_0(s)$ in the RHP. Solve the Nevanlinna-Pick problem. First calculate $u(\alpha_i)$, $u'(\alpha_i)$, and $u''(\alpha_i)$. Then construct the Fenyves Array.

$$u(\alpha_i) = \frac{r_m(\alpha_i)}{\tilde{p}_0(\alpha_i)} = 0.5 \tag{3.97}$$

$\alpha_1 = 1$	$\alpha_2 = 1$	$\alpha_3 = 1$	
0.5	0.5	0.5	$u_0(s)$
	0	0	$u_1(s)$

Figure 3.9: Fenyves Array for Example 3.4

$$u'(\alpha_i) = 0, u''(\alpha_i) = 0 \tag{3.98}$$

The corresponding Fenyves Array constructed using (3.97) is shown in Figure 3.9.

Looking at the Fenyves array in Figure 3.9, there are two possible solutions. First, by inspection, $u_0(s) = 0.5$ satisfies the interpolation conditions and the constraints on the first and second derivatives of $u(s)$. However, this choice of $u(s)$ yields an infinite gain compensator. In theory, this stabilizes the plant, but such a controller is difficult to implement. The second possible solution may be obtained from the second row of the modified Fenyves Array. If one were to continue the array to the third row, the last term in the array would also be zero. Using equation (3.87), the constraint on the first derivative of $u_1(s)$ at $s = \alpha_1$ is

$$u_1'(\alpha_1) = 0 \tag{3.99}$$

Therefore, $u_1(s)$ may be constructed by inspection, any bounded real function with two zeros at 1 will suffice. For this example, the $u_1(s)$ below will be chosen.

$$u_1(s) = \frac{(s-1)^2}{(s+2)^2} \tag{3.100}$$

It should be noted that if the Fenyves Array is completed to the third row, the inverse mapping yields the constant solution $u(s) = 0.5$. To get the other solution with the case of zero derivatives, the construction of the Fenyves Array must be stopped at the first row of

all zeros, where the derivative constraints can be easily incorporated into a rational function by inspection. Using (3.100), $u_0(s)$ is

$$u_0(s) = \frac{1.5s^3 - 0.5s^2 + 7s + 1}{1.5s^3 + 3.5s^2 + 9.5s + 3.5} \tag{3.101}$$

Step 3 – Compute $q(s)$.

$$q(s) = \frac{B(s)u(s)}{r_m(s)} = \left(\frac{s-1}{s+1}\right)^3 \left(\frac{1.5s^3 - 0.5s^2 + 7s + 1}{1.5s^3 + 3.5s^2 + 9.5s + 3.5}\right) \frac{2}{\tilde{p}_0(s)} \tag{3.102}$$

Step 4 – Compute $c(s)$.

$$c(s) = \frac{q(s)}{1 - p_0(s)q(s)} = \frac{-4(1.5s^3 - 0.5s^2 + 7s + 1)}{3(s+3)(s+2)^2} \tag{3.103}$$

The poles of the nominal closed-loop system are located at $-0.9556 \pm j2.1479$ and -0.4222.

□

The case of non-zero derivatives in the interpolation conditions is examined in the next example.

Example 3.5 – Interpolation with Non-zero Derivatives

Given the following unstable plant,

$$p_0(s) = \frac{(s+2)(s+3)(s+4)}{(s-1)^3} \tag{3.104}$$

with the uncertainty of the plant given by

$$r(s) = \frac{(s+3)(s+4)}{2(s+1)^2} \tag{3.105}$$

Find the robustly stabilizing compensator.

Step 1 – Compute $r_m(s)$, $\tilde{p}_0(s)$, and $B(s)$. Looking at (3.104), the unstable poles are $\alpha_1 = 1$, $\alpha_2 = 1$, and $\alpha_3 = 1$. Therefore, $B(s)$ can be written

$$B(s) = \left(\frac{s-1}{s+1}\right)^3 \tag{3.106}$$

Given $B(s)$, $\tilde{p}_0(s)$ can be written

$$\tilde{p}_0(s) = \frac{(s+2)(s+3)(s+4)}{(s+1)^3} \tag{3.107}$$

Since $r(s)$ is a minimum phase proper transfer function, $r_m(s) = r(s)$.

Step 2 – Find an SBR function $u(s)$ such that $u(\alpha_i) = \beta_i = \frac{r_m(\alpha_i)}{\tilde{p}_0(\alpha_i)}$ where α_i are the poles of $p_0(s)$ in the RHP. Solve the Nevanlinna-Pick problem. First calculate $u(\alpha_i)$. Because of the multiplicity, the first two derivatives of $u(s)$ must also match. These interpolation conditions are met when the modified Fenyves Array is constructed using (3.87). The Fenyves Array is shown in Figure 3.10. Using (3.87), the calculations required to construct the Fenyves array are outlined below.

$$u(\alpha_1) = \tfrac{(\alpha_i+1)}{2(\alpha_i+2)} \quad u'(\alpha_i) = \tfrac{1}{2(\alpha_i+2)^2} \quad u''(\alpha_i) = -\tfrac{1}{(\alpha_i+2)^3} \tag{3.108}$$

$$u_0(1) = \tfrac{1}{3} \quad u_0'(1) = \tfrac{1}{18} \quad u_0''(1) = -\tfrac{1}{27} \tag{3.109}$$

Using (3.87),

$$u_1(\alpha_1) = \frac{\bar{\alpha}_1 + \alpha_1}{1 - |u_0(\alpha_1)|^2} u_0'(\alpha_1) \tag{3.110}$$

$$u_1(\alpha_1) = \frac{1+1}{1 - |\frac{1}{3}|^2} \left(\frac{1}{18}\right) \tag{3.111}$$

Once again, applying (3.87) yields

$$u_2(\alpha_1) = \frac{\bar{\alpha}_1 + \alpha_1}{1 - |u_1(\alpha_1)|^2} u_1'(\alpha_1) \tag{3.112}$$

However, $u_1'(\alpha_i)$ must be expressed in terms of $u_0(\alpha_i)$. This is obtained by differentiating the mapping (3.39), which yields

$$u_2(s) = \left(\frac{1}{2}\right)\frac{s+\bar{\alpha}_1}{1-|u_1(\alpha_1)|^2}\left[\frac{u_0''(s)}{1-|u_0(\alpha_1)|^2}\frac{(s+\bar{\alpha}_1)^2}{(\alpha_1+\bar{\alpha}_1)}+ 2\frac{u_1(s)}{s+\bar{\alpha}_1}+2\frac{u_0(\alpha_1)u_1^2(s)(\alpha_1+\bar{\alpha}_1)}{(s+\bar{\alpha}_1)^2}\right]\Bigg|_{s=\alpha_i} \tag{3.113}$$

Substituting $s=\alpha_1$ into (3.113) yields

$$u_2(\alpha_1) = \frac{1}{21} \tag{3.114}$$

Using the inverse mapping, and arbitrarily choosing $u_2(s)=\frac{1}{21}$, gives the following solution for $u_0(s)$.

$$u_0(s) = \frac{32s^2+36s+16}{67s^2+124s+61} \tag{3.115}$$

Step 3 – Compute $q(s)$.

$$q(s) = \frac{B(s)u(s)}{r_m(s)} = \frac{32s^2+36s+16}{67s^2+124s+61}\frac{2(s-1)^3}{(s+1)(s+3)(s+4)} \tag{3.116}$$

Step 4 – Compute $c(s)$.

$$c(s) = \frac{q(s)}{1-p_0(s)q(s)} = \left(\frac{2}{3}\right)\frac{(32s^2+36s+16)}{(s+3)(s+4)} \tag{3.117}$$

The poles of the closed-loop system with the nominal plant $p_0(s)$ are located at -1 and $-0.9254 \pm j0.2327$. Although this approach can be applied for cases of higher multiplicity, the algebra required to determine the terms of the Fenyves Array becomes rather tedious for $m > 3$.

□

$\alpha_1 = 1$	$\alpha_2 = 1$	$\alpha_3 = 1$	
$u_0(1) = \frac{1}{3}$	$u_0(s) = \frac{1}{3}$	$u_0(1) = \frac{1}{3}$	$u_0(s)$
	$u_1(1) = \frac{1}{8}$	$u_1(1) = \frac{1}{8}$	$u_1(s)$
		$u_2(1) = \frac{1}{21}$	$u_2(s)$

Figure 3.10: Fenyves Array for Example 3.5

3.5 Summary and Conclusion

This chapter has reviewed the basic theory of robust control using interpolation for plants with additive unstructured perturbations. There are limitations to the current theory. Plants with poles on the jw axis and unstable poles with multiplicity greater than 1 cannot be handled, with the exception of a single pole at the origin. This chapter extended the theory by allowing the approach to include plants with poles on the jw axis as well as plants with a multiplicity of unstable poles.

Poles on the jw axis are handled by using a slightly modified mapping. The classical Nevanlinna-Pick algorithm relies on a mapping which produces SBR functions analytic in the right-half-plane. Because the jw axis lies on the boundary of the analytic region, the mapping cannot be used with poles on the jw axis. The mapping introduced in this chapter modifies the analytic region of the mapping to include some of the left-half-plane, which allows interpolation on the jw axis.

The case of unstable poles with multiplicity is handled by using a modified Fenyves array. Interpolation at poles with multiplicity places constraints on the derivatives of $u(s)$. These are incorporated in the modified Fenyves array. Once the modified Fenyves array is

constructed, the function $u(s)$ is obtained in the usual manner.

In summary, this chapter has presented the current theory for robust control of systems with additive unstructured perturbations, and extended the theory to include plants with a multiplicity of unstable poles and poles on the jw axis. The next chapter applies these techniques to the lateral control problem.

Chapter 4

The Lateral Control Problem

This chapter applies the robust control techniques developed in the previous chapter to the lateral control of highway vehicles. A bound on the plant uncertainty is obtained by simulating the car model as the velocity and cornering stiffness are varied. The 2 DOF linearized bicycle model is used to represent the lateral dynamics of the test vehicle, a GMC S-15 Blazer. From Chapter 2, the linearized model has a relative degree of 2, with two poles at the origin. The two poles at the origin may be handled using the interpolation techniques developed in the previous chapter. However, the relative degree of the plant will require two interpolation conditions at ∞. An approach for handling the two interpolation points at infinity is outlined in this chapter. The next section describes the process used to determine the bound on the plant uncertainty.

Parameter	**Nominal Value**	**Range**
m	1590 kg	constant
V_x	8 m/s (28.8 $kmph$)	$5-10$ m/s ($18-36$ $kmph$)
l_1, l_2	1.17 m, 1.42 m	constant
C_r, C_f	42000 Kn/rad	0.85 to 1.15
I_z	3200 $kg \cdot m^2$	constant
l^*	2 m	constant

Table 4.1: Estimated Parameters for S-15 Blazer

4.1 Vehicle Modeling and Uncertainty

The two degree-of-freedom linearized bicycle model for a vehicle's lateral dynamics presented in Chapter 2 will be used in this section to model the test vehicle, a GMC S-15 Blazer. The estimated nominal parameters for the Blazer and the expected range of values are listed in Table 4.1. The nominal speed of 8 m/s corresponds to 28.8 $kmph$, while the extreme speeds correspond to 18 $kmph$ and 36 $kmph$ respectively. These values were chosen because the initial field testing will be conducted at relatively low speeds. Simulation results for highway speeds appear in [15].

Using the values in Table 4.1, the nominal car model is given by equation (4.1).

$$\frac{E_f(s)}{\delta_f(s)} = \frac{114.2552(s^2 + 13.4391s + 31.4366)}{s^2(s^2 + 24.3156s + 151.9179)} \tag{4.1}$$

A MATLAB® program (see Appendix B) was written to determine the bound on the frequency-domain uncertainty of the nominal plant as the velocity and cornering stiffness

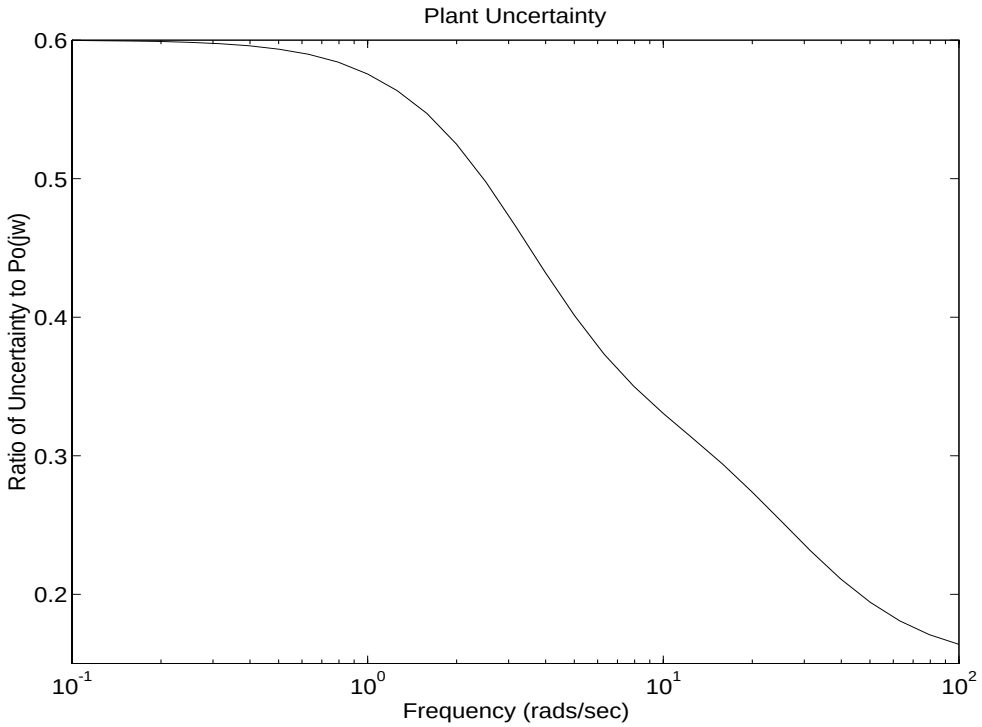


Figure 4.1: Simulation of Plant Uncertainty

vary over the ranges in Table 4.1. The program finds the magnitude of

$$|\delta p(jw)| = |p_0(jw) - p(jw)| \tag{4.2}$$

where $p_0(s)$ is the transfer function of the nominal plant and $p(s)$ is the transfer function of the actual plant as the parameters are varied. The results of the computer simulation are shown in Figure 4.1. The results are plotted as the ratio $|\frac{\delta p(jw)}{p_0(jw)}|$. This format of data presentation was chosen because it facilitates the choice of $r(s)$ as a function of $p_0(s)$. This simplifies the calculations required to arrive at the robust controller in the next section.

4.2 Controller Design

Using the data in Figure 4.1, a conservative bound on the plant uncertainty can be expressed as

$$r(s) = 0.6p_0(s) \tag{4.3}$$

With this bound, a robustly stabilizing controller can be designed if there exists an SBR solution to the interpolation problem. It should be noted that this choice of uncertainty is equivalent to the uncertain gain problem, whose solution is described in [72, 73]. However, a less conservative gain could be chosen which would require the use of the techniques described in Chapter 3.

The nominal plant described by equation (4.1) has two poles at the origin and a relative degree of 2. Therefore, the uncertainty bound also has a relative degree of 2. The two poles at the origin can be handled by using the techniques outlined in Chapter 3. The two interpolation points at infinity must be handled differently. One interpolation point at infinity will be handled using the approach described in Chapter 3. The second interpolation point at infinity will be added using equation (4.4). An alternate method to handle the multiple interpolation points at infinity would be to map the problem to the z-domain where zeros at infinity are represented as $z = 1$.

$$u^*(s) = u(s)\Omega(s) \tag{4.4}$$

By properly choosing $\Omega(s)$, the second interpolation point at infinity can be achieved, while still keeping $u^*(s)$ SBR.

Step 1 – Compute $r_m(s)$, $\tilde{p}_0(s)$, and $B(s)$. Define

$$p_0(s) = \frac{\tilde{p}_0(s)}{s^2} \tag{4.5}$$

Using equation (4.1), $\tilde{p}_0(s)$ may be written

$$\tilde{p}_0(s) = \frac{114.2552(s^2 + 13.4391s + 31.4366)}{(s^2 + 24.3156s + 151.9179)} \tag{4.6}$$

Similarly, define

$$r(s) = \frac{r_m^*(s)}{s^2} \tag{4.7}$$

where

$$r_m^*(s) = 0.6\tilde{p}_0(s) \tag{4.8}$$

$B(s) = 1$ because the nominal plant has no poles in the open right-half-plane.

Step 2 – Find an SBR function $u(s)$ such that $u(\alpha_i) = \beta_i = \frac{r_m^*(\alpha_i)}{\tilde{p}_0(\alpha_i)}$ where α_i are the poles of $p_0(s)$ in the closed RHP. Solve the Nevanlinna-Pick interpolation problem.

For the car problem, there are no interpolation points in the open RHP because the plant has no poles in the RHP. However, there are two interpolation points on the jw axis, $\alpha_1 = 0$ and $\alpha_2 = 0$. There is also one interpolation point at infinity (the other one will be handled later). Because of the choice of r_m^*, $u(\alpha_i)$ is a constant, shown in equation (4.9).

$$u(\alpha_i) = 0.6,\ u'(\alpha_i) = 0 \tag{4.9}$$

Using the results from the previous chapter, the modified Fenyves array for the car problem is shown in Figure 4.2.

Note that for this particular case, the elements of the Modified Fenyves array are not a function of ϵ. A suitable, yet arbitrary, choice for $u_2(s)$ is given in equation (4.10).

$$u_2(s) = -0.6 \tag{4.10}$$

$\alpha_1 = 0$	$\alpha_2 = 0$	$\alpha_3 = \infty$	
0.6	0.6	0	$u_0(s)$
	0	-0.6	$u_1(s)$
		-0.6	$u_2(s)$

Figure 4.2: Modified Fenyves Array for Car Problem

Using the modified mapping, from Chapter 3, $u_1(s)$ is shown in equation (4.11).

$$u_1(s) = -\frac{0.6s}{s + 2\epsilon} \tag{4.11}$$

Finally, $u_0(s)$ is given in equation (4.12).

$$u_0(s) = \frac{0.6(4\epsilon s + 4\epsilon^2)}{0.64s^2 + 4\epsilon s + 4\epsilon^2} \tag{4.12}$$

Note that $u_0(s)$ has a relative degree of 1, as expected. In order to obtain the second interpolation point at infinity, the $\Omega(s)$ in equation (4.13) will be chosen of the form,

$$\Omega(s) = \frac{(s + a)}{(s + b)(s + c)} \tag{4.13}$$

The coefficients a, b, and c must be chosen to ensure that $u^*(s)$ is SBR, and that the interpolation conditions at the origin are still met. One possible choice of $\Omega(s)$ is shown in equation (4.14).

$$\Omega(s) = \frac{(s + 0.25)}{(s + 0.5)^2} \tag{4.14}$$

This yields the following $u^*(s)$.

$$u^*(s) = \frac{0.6(4\epsilon s + 4\epsilon^2)(s + 0.25)}{(0.64s^2 + 4\epsilon s + 4\epsilon^2)(s + 0.5)^2} \tag{4.15}$$

The choice of ϵ in equation (4.15) affects the natural frequency w_0 of the closed-loop nominal system. Therefore, $\epsilon = 0.5$ results in a natural frequency of approximately 1.25 rad/sec. Substituting $\epsilon = 0.5$ in equation (4.15) yields equation (4.16).

$$u^*(s) = \frac{0.6(2s+1)(s+0.25)}{(0.64s^2+2s+1)(s+0.5)^2} \tag{4.16}$$

Step 3 – Compute $q(s)$. $q(s)$ can be found using equation (4.17).

$$q(s) = \frac{B(s)u^*(s)}{r(s)} = \frac{(2s+1)(s+0.25)}{(0.64s^2+2s+1)(s+0.5)^2 p_0(s)} \tag{4.17}$$

Step 4 – Compute $c(s)$. The compensator is shown in equation (4.18).

$$c(s) = \frac{(2s^2+1.5s+0.25)(s^2+24.3156s+151.9179)}{114.2552(0.64s^2+2.64s+1.16)(s^2+13.4391s+31.4366)} \tag{4.18}$$

The poles of the nominal-closed loop system are -2.5, -0.625, and -0.5. These are the poles of $u^*(s)$. The step response of the closed-loop system is shown in Figure 4.3.

The lane change trajectory shown in Figure 4.4 was used to simulate the family of plants as the cornering stiffness and velocity were varied. The lane change performance of the family of plants is shown if Figure 4.5. The lane change response for the family of plants varies as the model changes, but all responses are stable and slightly underdamped. The maximum percent overshoot is approximately 20 percent. Modifications in the response of the system may be made by choosing a different $u_2(s)$, and by choosing a different ϵ.

4.3 Summary and Conclusions

This chapter has presented the design of a robust lateral control system which guarantees stability as the cornering stiffness and velocity vary over a prescribed range. The car lateral

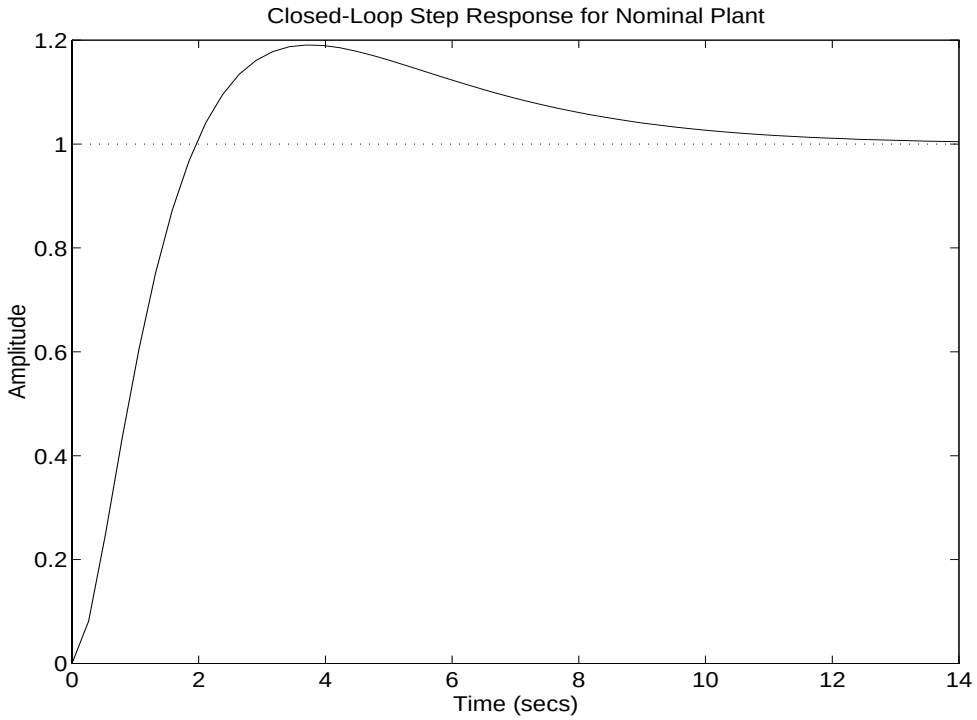


Figure 4.3: Step Response of Closed-Loop System with the Nominal Plant

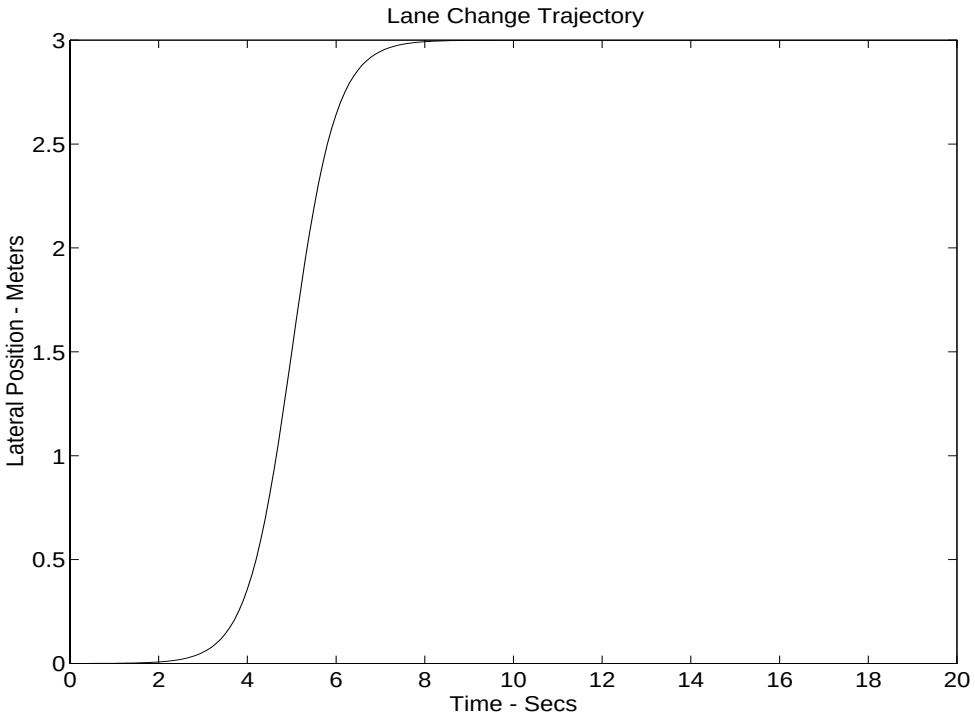


Figure 4.4: Lane Change Trajectory

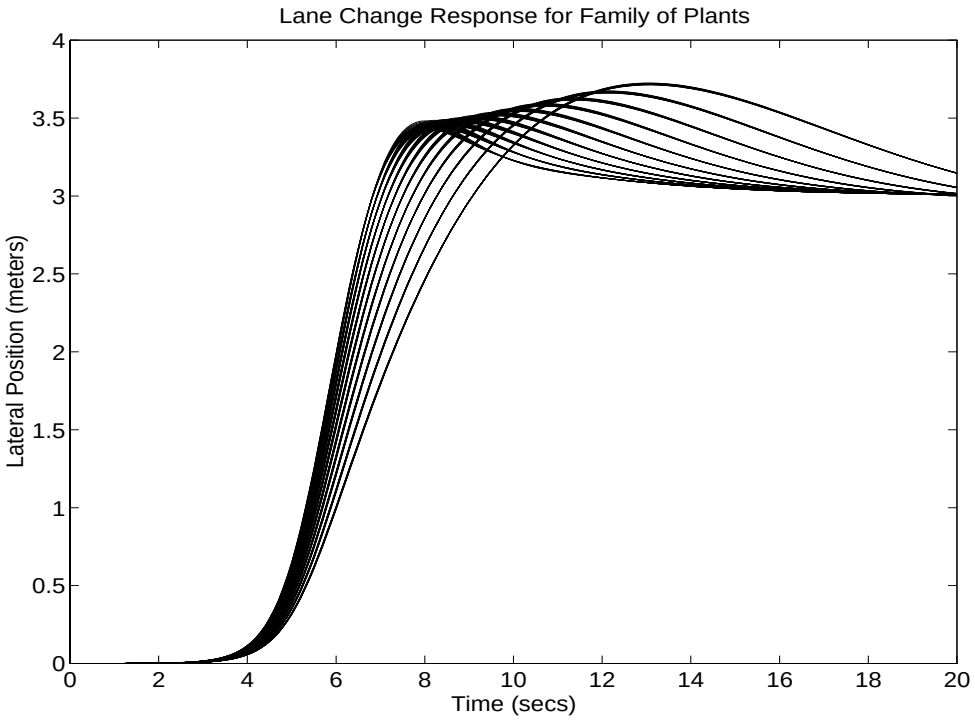


Figure 4.5: Lane Change Response, Family of Plants

control problem becomes an interpolation problem with two interpolation points at the origin and two interpolation points at infinity. The techniques developed in Chapter 3 are used to handle the two interpolation points at the origin and one interpolation point at infinity. The second interpolation point at infinity is achieved by multiplying the function $u(s)$ by another function $\Omega(s)$ which results in the correct relative degree.

Simulation results are presented for the family of plants as the cornering stiffness and velocity are varied. The performance of the closed-loop system varies as the plant parameters change, but all of the responses are stable for the fixed controller. Performance of the closed-loop system may be modified by changing the poles of $u(s)$, or by varying ϵ. The next chapter discusses the implementation of this control algorithm on a test vehicle.

Chapter 5

System Implementation and Experimental Results

This chapter presents experimental results from the field testing of the robust lateral control algorithm designed in the previous chapter. An overview of the test vehicle, a 1989 GMC Jimmy 4x4, is also provided. The motivation for implementing the robust controller was to determine the performance of the controller on the actual system, and determine deficiencies of this robust control approach which might require further research. The continuous-time robust controller was discretized using the bilinear transformation for computer implementation. A proportional-integral-derivative (PID) was also tested for comparison to the robust controller. The next section provides an overview of the testbed vehicle.

5.1 Testbed System Overview

The test vehicle used to verify the robust control algorithm is a 1989 GMC Jimmy. A sketch of the vehicle is shown in Figure 5.1. For the purposes of this research, the Jimmy has been modified for drive-by-wire operation, where there are no mechanical linkages between the driver's steering command and the motion of the wheels. A hydraulic control system is used to move the wheels under electronic control. A block diagram of the vehicle system is shown in Figure 5.2. Two parallel valves are used to control the steering actuator. This provides some redundancy in case of a valve failure. A hydraulic accumulator, charged to approximately 1000 psi, stores energy in case of a pump failure. An analog back-up system was included to allow steering in the event of a computer failure. The analog back-up system is shown in Figure 5.3.

A proportional-integral (PI) control law was used for the low-level steering actuator control. The sampling period for the steering actuator control was chosen to be 10 ms. This was the fastest periodic interrupt available from the real-time clock on the control computer, and is 5 times faster than the 20 Hz bandwidth of the hydraulic valves. The PI controller was empirically tuned using step functions to examine the response of the steering actuator system. The response of the closed-loop steering actuator with the controller described by equation (5.1) is shown in Figure 5.4.

$$C_{steering}(z) = 6 + \frac{0.01}{1 - z^{-1}} \tag{5.1}$$

The output of the low-level steering controller was bounded to limit the speed of the hydraulic actuator. It also should be noted that there is approximately 1 degree of backlash

Figure 5.1: Test Vehicle (Reprinted with Permission from General Motors Corp.)

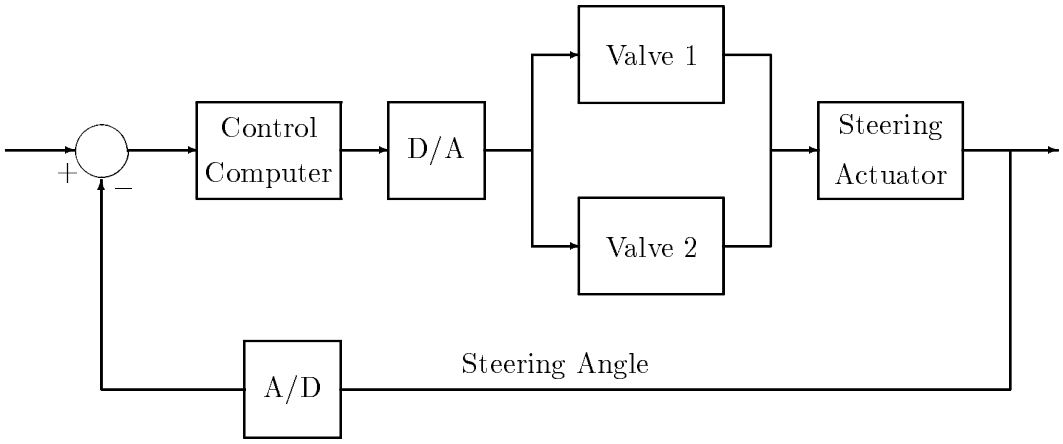


Figure 5.2: Jimmy Steering Block Diagram

in the steering actuator. Because of the actuator design, the actuator can move very small amounts before the wheels actually begin to turn. This is caused by movement of the actuator mounting plate, as well as by movement of the actuator attachment point to the tie rod. This will result in a slight modification of the robust controller design implementation in the next section.

The vehicle control experiments were conducted on an instrumented 1 mile track. A 16 gauge, teflon coated wire was placed in a 3/8 inch deep slot located in the center of the outside lane. The wire was then covered with hot asphalt sealer to prevent damage to the track. A 9.6 kHz signal in the wire is sensed at the front of the test vehicle to determine the lateral position. A more detailed description of the wire reference system and sensing electronics appears in Appendix A.

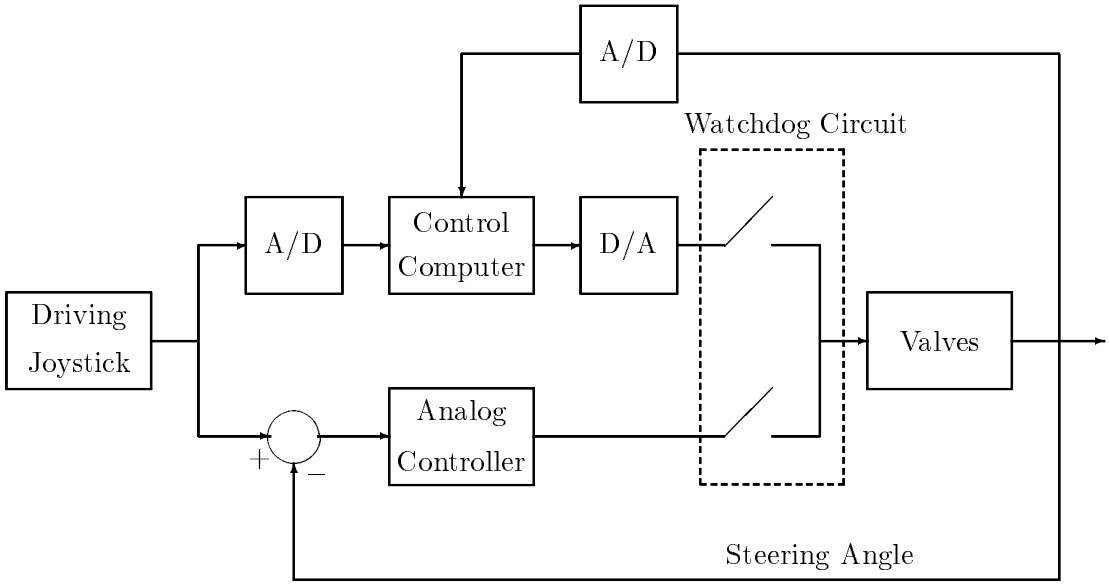


Figure 5.3: Analog Backup Steering Block Diagram

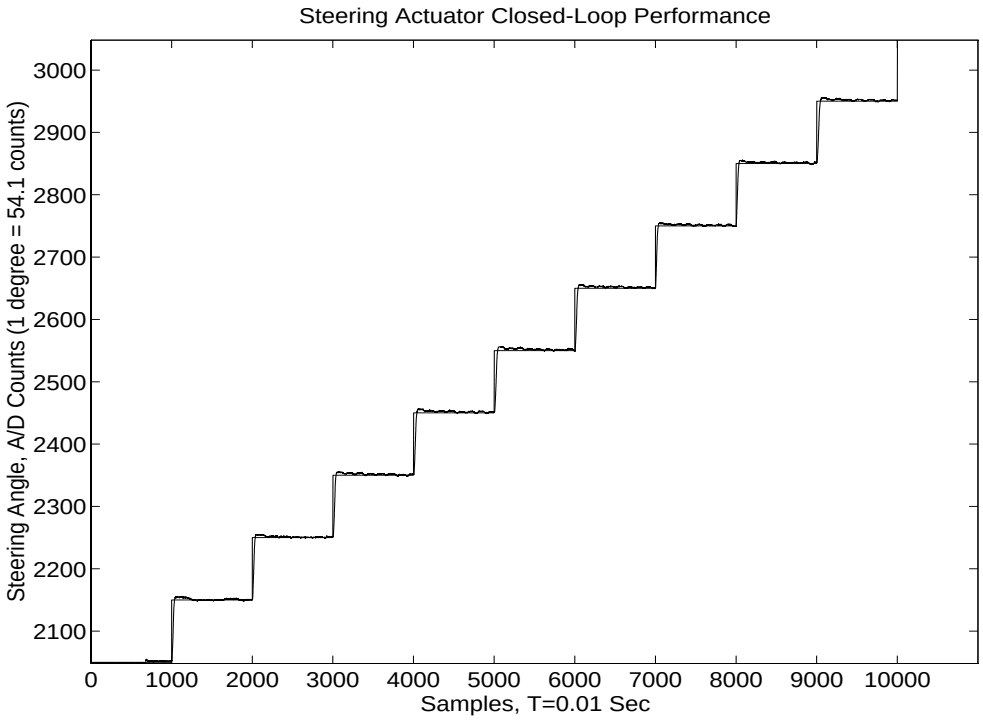


Figure 5.4: Steering Actuator Closed-Loop Response

5.2 Algorithm Implementation

From the previous chapter, the robustly stabilizing compensator is given by

$$c(s) = \frac{(2s^2 + 1.5s + 0.25)(s^2 + 24.3156s + 151.9179)}{114.2552(0.64s^2 + 2.64s + 1.16)(s^2 + 13.4391s + 31.4366)} \tag{5.2}$$

The compensator was implemented on a single board computer with a 286-compatible processor. The sampling period $T = 100\ ms$ was chosen because it is approximately 10 times faster than the closed-loop bandwidth of the system. Using the bilinear transformation

$$s = \frac{2}{T}\frac{z-1}{z+1},\ T = 100\ ms \tag{5.3}$$

The discretized version of the robust controller is

$$\frac{\delta_f(z)}{E_f(z)} = \frac{0.0321 - 0.0939z^{-1} + 0.1000z^{-2} - 0.0462z^{-3} + 0.0079z^{-4}}{1 - 2.8973z^{-1} + 3.2034z^{-2} - 1.6189z^{-3} + 0.3164z^{-4}} \tag{5.4}$$

The units of $\delta_f(z)$ are radians, while the units of $E_f(z)$ are meters. The computer implementation requires a scale factor to changes the units to A/D and D/A counts. This is calculated below.

$$\left(\frac{\text{radians}}{\text{meters}}\right)\left(\frac{180\text{ deg}}{3.14159\text{ rad}}\right)\left(\frac{54.115\text{ counts}}{1\text{ deg}}\right)\left(\frac{1\text{ meter}}{4.94291\text{ volts}}\right)\left(\frac{1\text{ volt}}{204.8\text{ counts}}\right) = \tag{5.5}$$
$$3.06286\left(\frac{\text{D/A counts}}{\text{A/D counts}}\right)$$

Using equation (5.5), the robust controller, expressed in terms of D/A and A/D counts is given by

$$\frac{\delta_f(z)}{E_f(z)} = \frac{0.09845 - 0.2877z^{-1} + 0.3064z^{-2} - 0.1415z^{-3} + 0.02435z^{-4}}{1 - 2.8973z^{-1} + 3.2034z^{-2} - 1.6189z^{-3} + 0.3164z^{-4}} \tag{5.6}$$

Writing equation (5.6) as a difference equation yields

$$\delta_f(k) = 2.8973\delta_f(k-1) - 3.2034\delta_f(k-2) + 1.6189\delta_f(k-3) - \tag{5.7}$$

$$0.3164\delta_f(k-4) + 0.09845E_f(k) - 0.2877E_f(k-1) +$$

$$0.3064E_f(k-2) - 0.1415E_f(k-3) + 0.02435E_f(k-4)$$

The controller described by equation (5.7) outputs steering commands on the order of 1 degree when simulated with typical error signals. Because the steering actuator has approximately 1 degree of backlash, the controller's output was multiplied by 2 in the actual implementation to overcome some of the effects of the backlash. The controller described by equation (5.7) also does not have any integration action for zeroing out steady-state errors. Sources of such errors include steering offsets and road superelevation. Therefore, the robust controller was modified to include a small integration term to reduce the steady state-error. A decaying integration term was used to reduce the chances of creating instability. The integration term is described by equation (5.8).

$$\frac{I(z)}{E_f(z)} = \frac{K_I}{1 - 0.95z^{-1}},\ K_I = 0.001 \tag{5.8}$$

The actual robust controller implemented on the test vehicle is then given by equation (5.9).

$$\begin{aligned} \delta_f(k) \quad = \quad & 2[2.8973\delta_f(k-1) - 3.2034\delta_f(k-2) + 1.6189\delta_f(k-3) - \\ & 0.3164\delta_f(k-4) + 0.09845E_f(k) - 0.2877E_f(k-1) + \\ & 0.3064E_f(k-2) - 0.1415E_f(k-3) + 0.02435E_f(k-4)] + \\ & I(k) \end{aligned} \tag{5.9}$$

The robust controller described by equation (5.9) was tested on a track instrumented with a wire reference system. All test runs were conducted on a straight section of the test

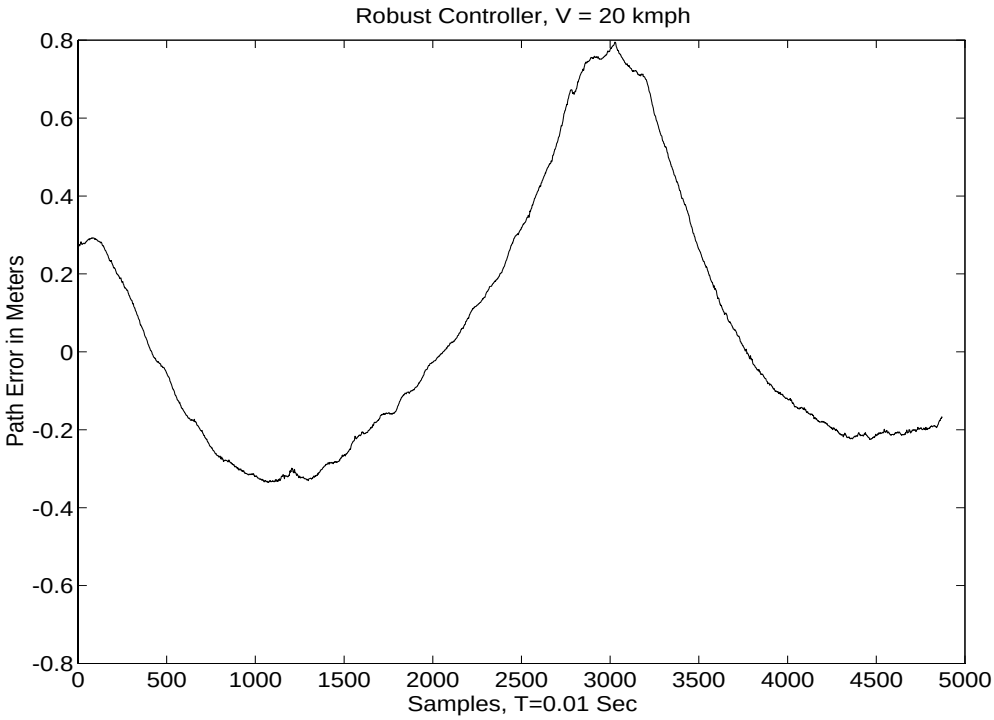


Figure 5.5: Robust Controller, $V_x = 20\ kmph$

track. The test results for 20 $kmph$, 30 $kmph$, and 40 $kmph$ test runs are shown in Figures 5.5 - 5.7.

A proportional-integral-derivative (PID) control algorithm was also implemented for comparison to the robust controller. The PID control algorithm is described by equation (5.10).

$$P(z) = 0.3 * E(z),\ I(z) = \frac{0.0005E(z)}{1 - 0.99z^{-1}},\ D(z) = 6(z^{-5} - 1)E(z) \tag{5.10}$$

$$U(z) = P(z) + I(z) + D(z) \tag{5.11}$$

The performance of the PID algorithm at the same three speeds is shown in Figures 5.8 - 5.10. A summary of the rms error for each test run appears in Table 5.1.

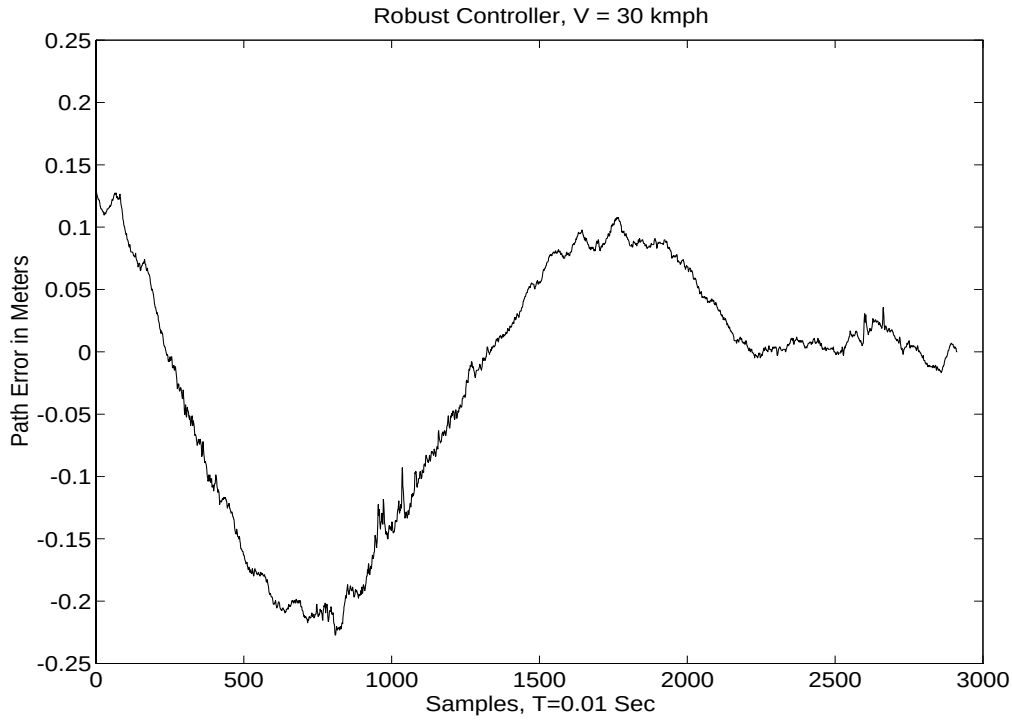


Figure 5.6: Robust Controller, $V_x = 30\ kmph$

Speed $(kmph)$	Robust Controller rms error	PID Controller rms error
20	0.1085 m	0.0857 m
30	0.0751 m	0.0858 m
40	0.0953 m	0.0779 m

Table 5.1: RMS Error for PID and Robust Controllers

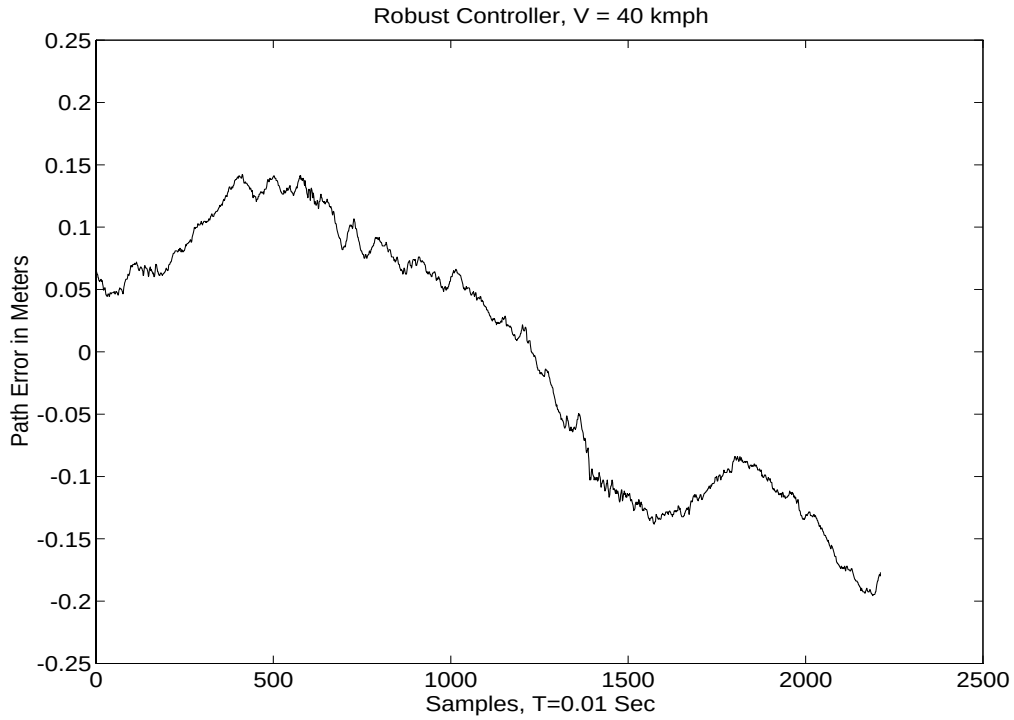


Figure 5.7: Robust Controller, $V_x = 40\ kmph$

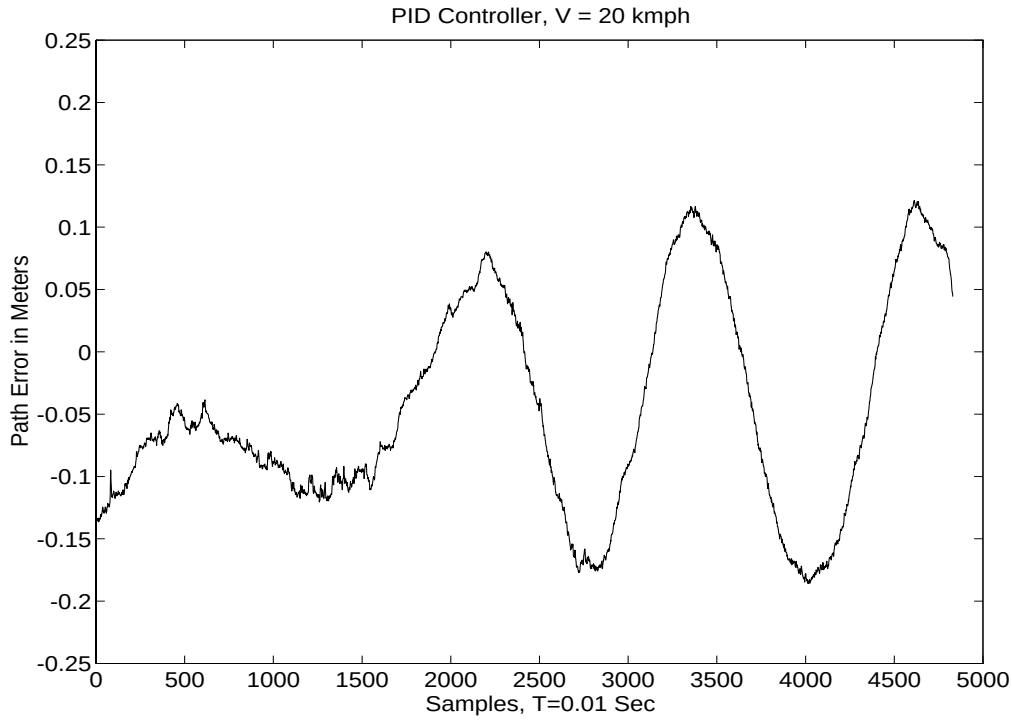


Figure 5.8: PID Controller, $V_x = 20\ kmph$

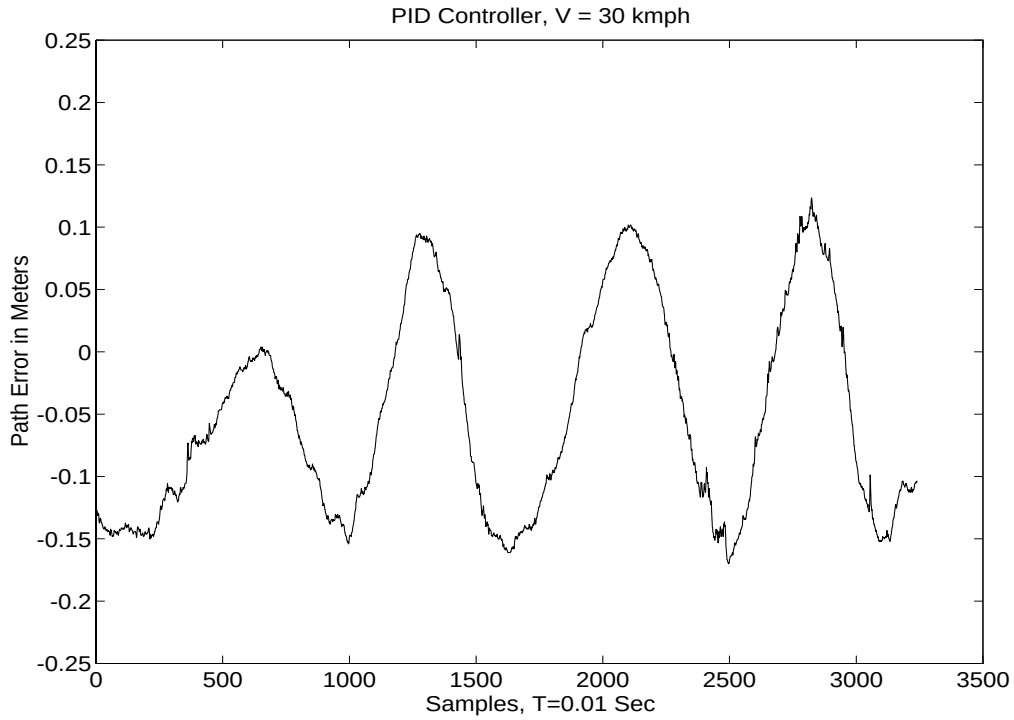


Figure 5.9: PID Controller, $V_x = 30\ kmph$

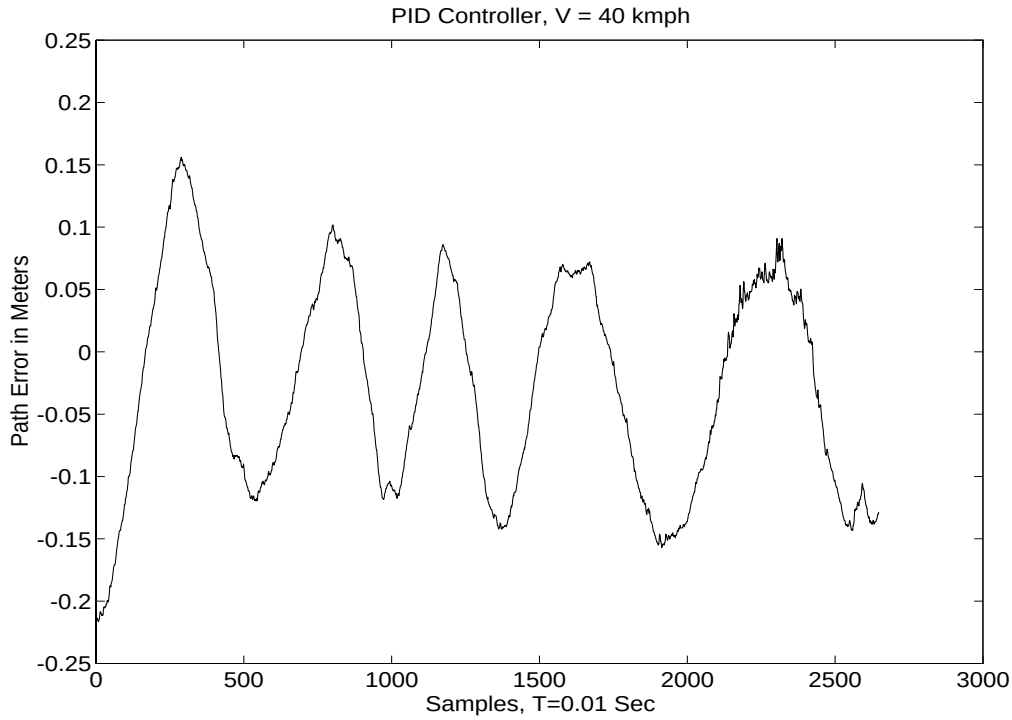


Figure 5.10: PID Controller, $V_x = 40\ kmph$

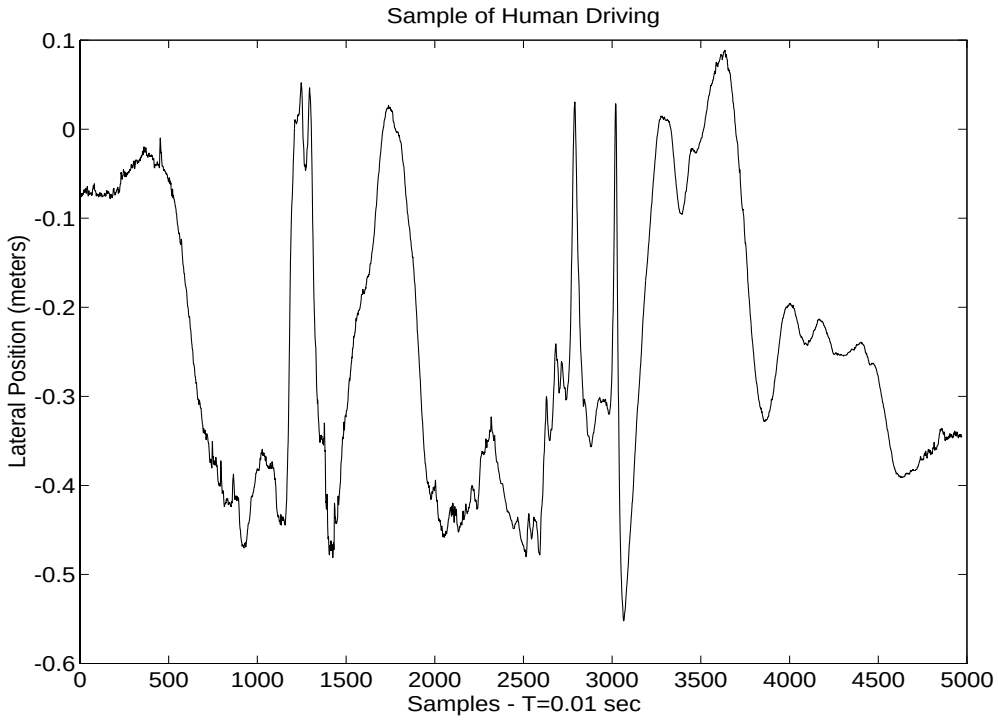


Figure 5.11: Human Driving Sample

5.3 Discussion of Test Results

Looking at the test results in the previous section, and at the rms error summarized in Table 5.1, both controllers performed satisfactorily at the speeds tested. To allow a fair comparison of the automatic controllers, a sample of human driving is provided in Figure 5.11. This data was taken as the speed varied between 30 $kmph$ and 60 $kmph$. The human driving sample shows that the performance of the automatic control systems is equal to or better than the human driving.

It should be noted that the performance measurement, rms error, was not incorporated into the controller design (robust or PID). However, it is customary to speak about the rms error (or maximum error), when discussing test results. The U-Parameter design would be

one way to incorporate an rms error specification into the robust controller design [48]. This approach was not pursued because of difficulty parameterizing the free parameter $U(s)$ with the interpolation conditions at the origin.

The rms error for the PID controller is better at 20 $kmph$ and 40 $kmph$. The nominal plant for the robust control design was at 28.8 $kmph$, so it would seem reasonable that the best performance of the robust controller would be at 30 $kmph$. At this speed, the performance of the robust controller was slightly better than the PID controller. There are several difficulties with the robust controller implementation however. The robust controller was very sensitive to the accuracy of the center steering position as well as to the initial orientation of the vehicle. The test data presented for the robust controller was obtained with the vehicle starting centered and headed in the proper direction. If the vehicle was started in a different orientation, headed slightly to the left or right, the robust controller had difficulty correcting. In addition, adding an offset to the center steering position will cause the robust controller to have difficulty keeping the vehicle centered in the lane. Both of these difficulties represent unmodeled dynamics that were not taken into account in the robust controller design and uncertainty modeling. Therefore, it is reasonable that the robust controller should have difficulty with these conditions, even with the addition of a small integration term.

Comparing the robust controller to the PID controller, the robust controller results in less oscillations. The performance of the PID controller could almost be described as steady state, slow oscillation. The period of the oscillations is very large, and is almost imperceptible by the driver. The higher gain and integration term in the PID controller contribute to the oscillations, but also help the controller overcome some of the unmodeled disturbances

like steering angle offsets and actuator backlash. Because of this, the PID controller is able to handle deviations in initial orientation much better than the robust controller. The PID controller also operates successfully up to 60 $kmph$ while the maximum speed of the robust controller is approximately 40 $kmph$. One of the most useful results of the hardware implementation is the identification of steering actuator backlash, center steering offsets, and road disturbances as problematic areas for control system implementation. These items, in addition to parameter variations, must also be considered when designing a truly robust control system.

5.4 Summary and Conclusions

This section has described the hardware implementation of the robust controller designed in the previous chapter. Test results for the robust controller and a PID controller were presented for three test speeds: 20 $kmph$, 30 $kmph$, and 40 $kmph$. The performance of both controllers was satisfactory. The PID controller performed better at 20 $kmph$ and 40 $kmph$, but resulted in steady-state slow oscillations at all three speeds. The robust controller exhibited the best performance at 30 $kmph$, which was reasonable considering the nominal robust design was performed for a speed of 28.8 $kmph$. The robust controller did not exhibit oscillations at any velocity.

Several deficiencies in the robust controller design were revealed in the hardware implementation. These included difficulty handling steady state error, caused by center steering offsets and road superelevation. In addition, the robust controller had difficulty with backlash in the steering actuator, which in effect neglected the small steering corrections from

the robust controller. This was remedied by increasing the gain of the robust controller by a factor of two. The steady state error was improved by adding a small integration term to the robust controller. Incorporating the steering center offset uncertainty and steering actuator backlash in the uncertainty modeling will probably result in better controller performance.

The next chapter discusses several recent control techniques that can be applied to the lateral control problem.

Chapter 6

Simultaneous Stabilization & Sensitivity Minimization

This chapter explores two additional robust control techniques that may be useful for the lateral control problem. The first is simultaneous stabilization of a family of plants, which uses the results described in [11]. Rather than describing the plant uncertainty in the frequency domain, this technique applies Kharitonov's Theorem to analyze stability as the coefficients of the transfer function model are varied. The second approach described involves minimizing the sensitivity function $S(s)$ and the derivative of the control sensitivity $R(s)$ in order to improve the performance of the lateral control system. Minimizing $S(s)$ improves the disturbance rejection, while minimizing the derivative of $R(s)$ minimizes the frequency content of the steering commands from the control system.

6.1 Simultaneous Stabilization

Given the family of plants described by equation 6.1, simultaneous stabilization can be guaranteed when the following assumptions are met [11].

$$P(s,q) = \frac{\sum_{i=0}^{m(q)} b_i(q) s^{m(q)-i}}{s^{n(q)} + \sum_{i=1}^{n(q)} a_i(q) s^{n(q)-i}} = \frac{N_p(s,q)}{D_p(s,q)} \tag{6.1}$$

Assumptions:

1. For each fixed $q \in Q$, $P(s,q)$ is strictly proper. $m(q) < n(q)$.

2. There exists some positive integer $n_{max} < \infty$ such that $n(q) \leq n_{max}$ for all $q \in Q$.

3. For all allowable values of the subscript i, $a_i(q)$ and $b_i(q)$ are continuous over Q.

4. Q is a compact subset of $\mathbb{R}^{\mathrm{P}}$.

5. Let $m_{max} = \max_{q \in Q}\ m(q)$ and $n_{max} = \max_{q \in Q}\ n(q)$. Then for each $k < m_{max}, l \leq n_{max}$, the set

$$Q_{kl} = \{q \in Q : m(q) = k, n(q) = l\}$$

6. For each fixed $q \in Q$, $P(s,q)$ is minimum phase.

7. The sign of the high frequency gain $b_0(q)$ is invariant over Q.

The coefficients of the transfer function model of a car's lateral dynamics may be described by

$$\frac{E_f(s)}{\delta_f(s)} = \frac{q_1 s^2 + q_2 s + q_3}{s^2(q_4 s^2 + q_5 s + q_6)} \tag{6.2}$$

coefficient	max	min
q_1	131.3935	97.1169
q_2	3249.1	887.514
q_3	4750.1	2595.1
q_4	1	1
q_5	44.7406	16.5346
q_6	499.6620	72.7904

Table 6.1: Summary of Coefficient Changes

Equation (6.2) meets all of the assumptions required for simultaneous stabilization of the family of plants as the parameters vary.

A MATLAB program was written to determine the minimum and maximum values of the coefficients in the transfer function model as the velocity and cornering stiffness vary. The values in Table 4.1, describing the parameters for the GMC Blazer, were used in the simulations. The MATLAB program appears in Appendix B. The results of the simulation are summarized in Table 6.1.

Using the results described in [11], a compensator of the form

$$C(s) = \frac{\delta_1 s + \delta_0}{\epsilon_2 s^2 + \epsilon_1 s + \epsilon_0};\ \delta_1 > 0 \tag{6.3}$$

can be constructed to stabilize the family of plants as the parameters vary. In order to ensure that $N_c(s)$ is Hurwitz, we will choose $\delta_0 = \delta_1 = 1$. This yields

$$N_c(s) = s + 1 \tag{6.4}$$

The next step in constructing the compensator to stabilize the family of plants is to construct

$$\triangle_0(s,q) = N_c(s)N_p(s,q) \tag{6.5}$$

Equation (6.5) must always be Hurwitz for a simultaneously stabilizing controller to exist. The coefficients $q_1 - q_3$ are always positive, ensuring that $\triangle_0$ is always stable. In order to determine the coefficients of the simultaneously stabilizing controller, the following equations must be iteratively solved, making sure that each is always Hurwitz for the expected parameter changes. In [11], the existence of a stable $\triangle_i(s,q)$ is guaranteed.

$$\triangle_1(s,q) = \triangle_0(s,q) + \epsilon_0 D_p(s,q) \tag{6.6}$$

$$\triangle_2(s,q) = \triangle_1(s,q) + \epsilon_1 s D_p(s,q) \tag{6.7}$$

$$\triangle_3(s,q) = \triangle_2(s,q) + \epsilon_2 s^2 D_p(s,q) \tag{6.8}$$

One way of proving that a Hurwitz solution to equations (6.6-6.8) exists is to write the equations in the following form.

$$\triangle_{i+1}(s,q) = \triangle_i(s,q) + \epsilon_i s^i D_p(s,q) \tag{6.9}$$

Let $\epsilon_i = \frac{1}{K_i}$. Then equation(6.9) can be written

$$\frac{K_i \triangle_{i+1}(s,q)}{s^i D_p(s,q)} = 1 + K_i \frac{\triangle_i(s,q)}{s^i D_p(s)} \tag{6.10}$$

Taking the limit as K_i approaches ∞, the zeros of $\triangle_{i+1}(s,q)$ approach the zeros of $\triangle_i(s,q)$, which is stable by construction. One zero of $\triangle_{i+1}(s,q)$ approaches $-\infty$, which is also

stable, because the relative degree of $\frac{\triangle_i(s,q)}{s^i D_p(s)}$ is equal to 1. Therefore, as ϵ_i approaches zero, a Hurwitz $\triangle_{i+1}$ is always guaranteed to exist.

The task of ensuring that equations (6.6-6.8) are Hurwitz is greatly simplified by applying Kharitonov's Theorem [11, 4]. Kharitonov's Theorem is stated below.

Theorem 6.1 [4] The polynomial family

$$P(s,\mathcal{A}) = \{p(s,a) = a_0 + a_1 s + ... + a_n s^n | a \in \mathcal{A}\}, a_n > 0 \tag{6.11}$$

is stable if and only if the following polynomials are stable:

$$p^{+-}(s) = a_0^+ + a_1^- s + a_2^- s^2 + a_3^+ s^3 + a_4^+ s^4 + \ldots \tag{6.12}$$

$$p^{++}(s) = a_0^+ + a_1^+ s + a_2^- s^2 + a_3^- s^3 + a_4^+ s^4 + \ldots \tag{6.13}$$

$$p^{-+}(s) = a_0^- + a_1^+ s + a_2^+ s^2 + a_3^- s^3 + a_4^- s^4 + \ldots \tag{6.14}$$

$$p^{--}(s) = a_0^- + a_1^- s + a_2^+ s^2 + a_3^+ s^3 + a_4^- s^4 + \ldots \tag{6.15}$$

□

The polynomials (6.12-6.15) are called Kharitonov polynomials. The superscripts on a indicate the upper and lower bounds on the coefficients.

A MATLAB program was written to verify Kharitonov stability as the controller was constructed iteratively using equations (6.6-6.8). The following values for ϵ_i were chosen:

$$\epsilon_0 = 1,\ \epsilon_1 = 0.05,\ \epsilon_2 = 0.00001 \tag{6.16}$$

ϵ_0 was selected to arrive at a dc gain of 1. ϵ_1 and ϵ_2 were chosen to guarantee Kharitonov stability. The simultaneously stabilizing controller is shown in equation (6.17).

$$c(s) = \frac{s+1}{0.00001s^2 + 0.05s + 1} \tag{6.17}$$

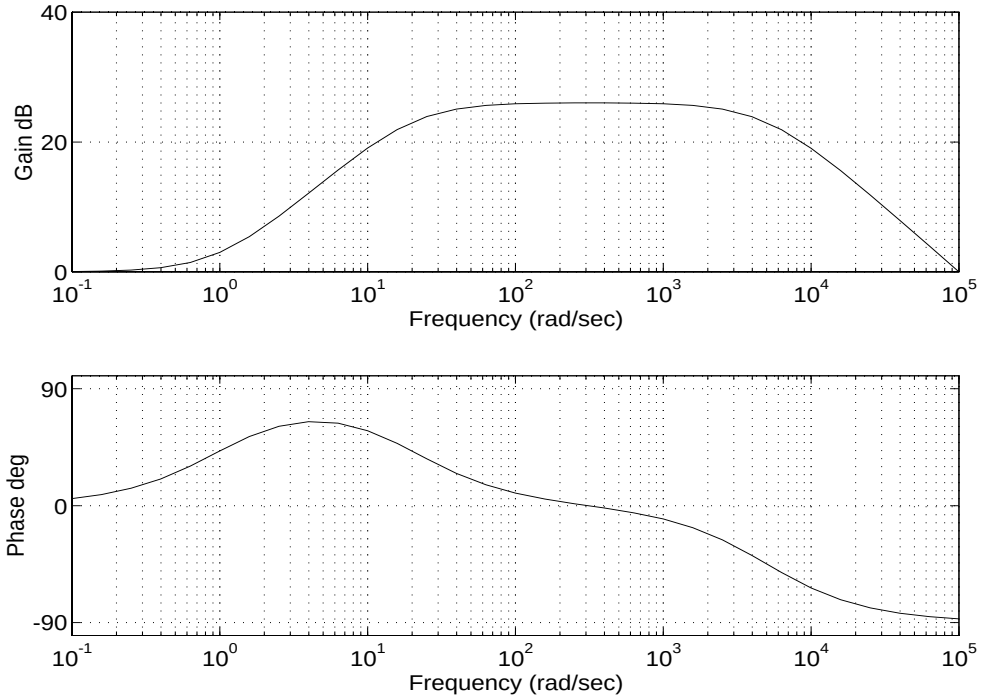


Figure 6.1: Bode Plot of Simultaneously Stabilizing Controller

A Bode plot of the simultaneously stabilizing controller appears in Figure 6.1. The robust controller is basically a lead compensator, ignoring the higher frequency roll-off. The lane change response for the family of plants with this robust controller in shown in Figure 6.2. Note that the lane change response for the family of plants is significantly better than for the robust controller designed in Chapter 4. There is an interesting synergy between the simultaneous stabilization approach and classical frequency domain methods. It is possible to design a lead compensator using classical methods, taking into account the performance of the nominal closed-loop system, while still guaranteeing stability for the family of plants. This would be an iterative trial and error process though, involving the design of the lead compensator for nominal plant performance, and then the construction of a similar compensator using the simultaneous stabilization approach.

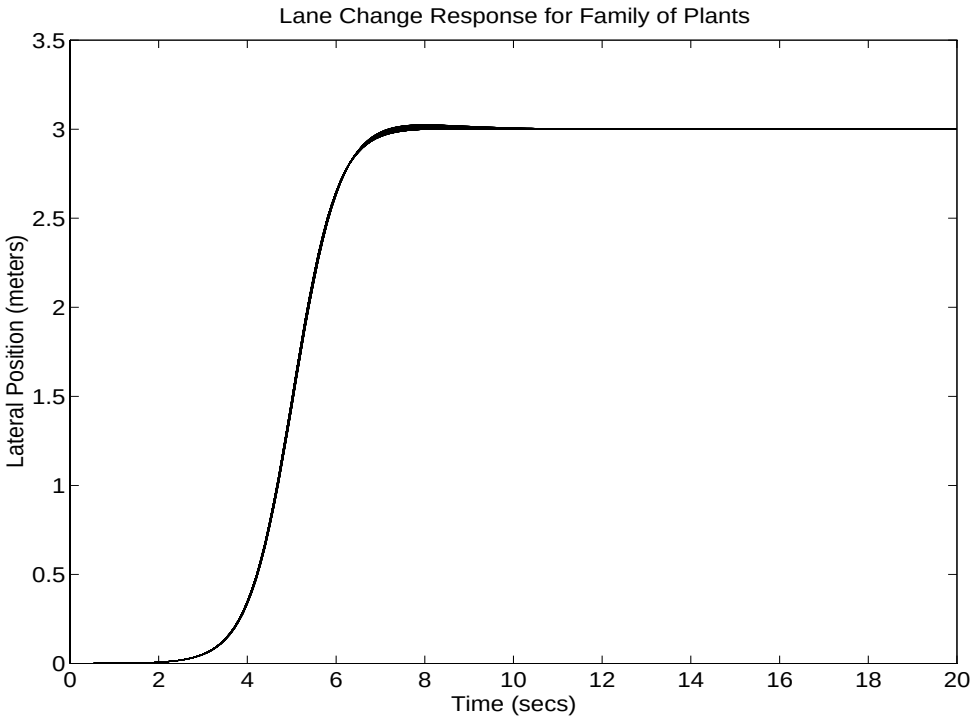


Figure 6.2: Lane Change Response for Family of Plants

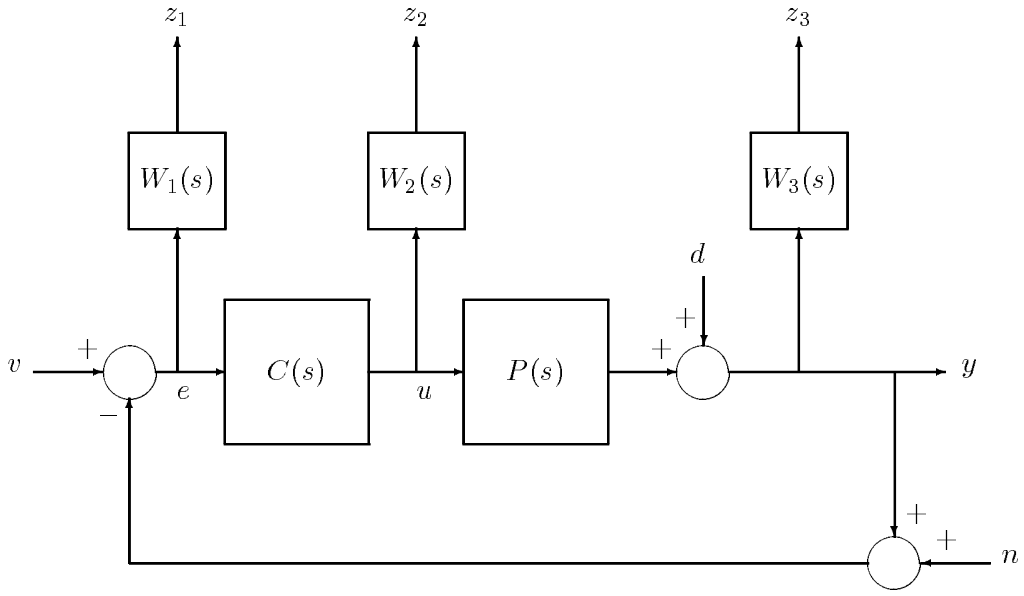


Figure 6.3: Feedback System with Weighting

6.2 Sensitivity Minimization

This section explores the application of multiobjective design to the lateral control problem. Given the feedback system with weightings $W_1 - W_2$ shown in Figure 6.3, it is possible to design a stabilizing compensator which also minimizes the following H_∞ norm.

$$\left\| \begin{array}{c} W_1(s)S(s) \\ W_2(s)R(s) \\ W_3(s)T(s) \end{array} \right\|_\infty \leq 1 \tag{6.18}$$

where

$$S(s) = \frac{1}{1+C(s)P(s)} \quad \text{Sensitivity} \tag{6.19}$$

$$R(s) = \frac{C(s)}{1+C(s)P(s)} \quad \text{Control Sensitivity} \tag{6.20}$$

$$T(s) = \frac{C(s)P(s)}{1+P(s)C(s)} \quad \text{Complementary Sensitivity} \tag{6.21}$$

The weighting function $W_1(s)$ is a mirror image of the desired sensitivity function $S(s)$. It is desirable to have the magnitude of $S(s)$ small at low frequencies, and large at high frequencies to obtain good disturbance rejection [8]. The actuator weight $W_2(s)$ is used to penalize actuator signals. An appropriate $W_2(s)$ should have a small magnitude at low frequencies and a large magnitude at high frequencies to cost for saturation [8]. $W_2(s)$ may also contain a constant component to reflect the costs of constant actuation. It is desirable to penalize the complementary sensitivity at high frequencies to minimize the influence of high frequency measurement noise. This yields a high pass filter shape for $W_3(s)$. The closed-loop bandwidth can also be limited by $W_3(s)$, since the cut-off frequency of $W_3(s)$ limits the system bandwidth [8].

A solution to the robust minimization problem described by equation (6.18) can be obtained by using an augmented plant (shown in Figures 6.3-6.4) and solving two Riccati equations [29]. The equations describing the augmented system are given by:

$$\dot{x} = Ax + B_1 u_1 + u_2 \tag{6.22}$$

$$y_1 = C_1 x + D_{11} u_1 + D_{12} u_2 \tag{6.23}$$

$$y_2 = C_2 x + D_{21} u_1 + D_{22} u_2 \tag{6.24}$$

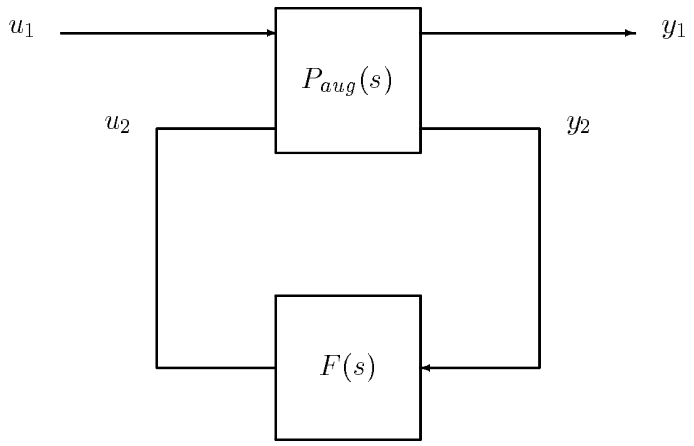


Figure 6.4: Augmented Model

where the augmented plant $P_{aug}(s)$ is

$$P_{aug}(s) = \left[\begin{array}{c|c} W_1 & -W_1 P \\ 0 & W_2 \\ 0 & W_3 P \\ \hline I & -P \end{array}\right] \tag{6.25}$$

with the state-space realization

$$P(s) := \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{array}\right] \tag{6.26}$$

The MATLAB command **hinf** was used to solve the multiobjective minimization problem with the following weighting matrices. Once again, the nominal car model described by the

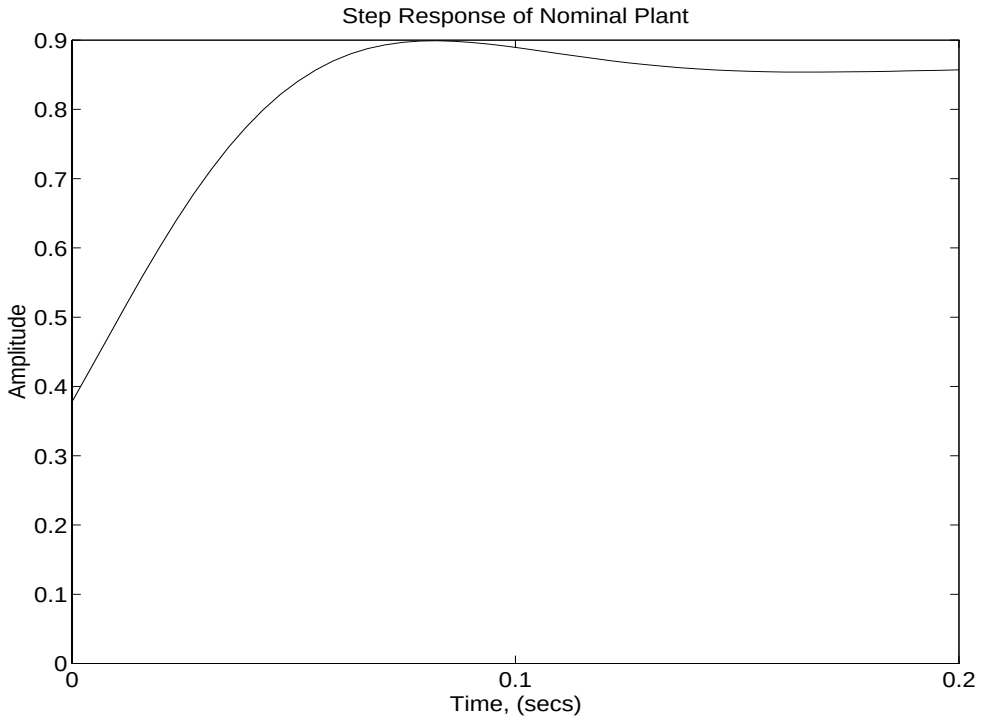


Figure 6.5: Nominal Plant Step Response

parameters in Table 4.1 was used.

$$W_1(s) = 8500\frac{(s/15+1)(s/15+1)}{(s/0.1+1)(s/0.1+1)} \tag{6.27}$$

$$W_2(s) = 0.001\frac{s}{s/100+1} \tag{6.28}$$

$$W_3(s) = 0 \tag{6.29}$$

The simulation results for the robustly stabilizing controller appear in Figure 6.5. The performance of the controller is acceptable, but the controller has two zeros very close to the origin which might cause problems in an actual implementation.

6.3 Summary and Conclusions

This chapter has explored two additional robust control techniques which are applicable to the lateral control problem. The first approach, simultaneous stabilization of a family of plants, is possible because the system model meets the minimum phase requirement. This approach is potentially very useful, because the structure of the controller is known in advance. This allows tweaking the controller with frequency-domain trial-and-error methods to obtain the desired performance of the nominal plant, as well as stability for the family of plants.

The second approach described in this chapter is a multiobjective design where the sensitivity $S(s)$ and the control sensitivity $R(s)$ are minimized. Minimizing the sensitivity reduces the affects of disturbances while minimizing the derivative of the control sensitivity reduces the frequency content of the control effort. The simulated step response for the nominal plant with the robust controller was satisfactory. However, the controller obtained by solving 2 Riccati equations contains two zeros very close to the origin. These might cause difficulties in a practical implementation.

Chapter 7

Summary and Conclusions

This chapter provides a summary and recommendations for future research.

7.1 Summary

The main focus of this dissertation has been the automatic lateral control of highway vehicles, and the application of robust control techniques to this problem. In Chapter 2, a review of lateral control research was presented. Many of the control techniques being applied to the lateral control problem today, like optimal control and gain-scheduling, have their origins in the early research. The largest automated highway research effort is currently being conducted at PATH. Recent results have included field testing of frequency-shaped-linear-quadratic optimal controllers, PID controllers, and fuzzy logic controllers. Test speeds are relatively slow, ranging from 20 $kmph$ to 60 $kmph$. Parametric robust control techniques have also been proposed by J. Ackermann in several recent publications.

The robust control approach used in this dissertation models the plant uncertainty as

unstructured additive perturbations in the frequency-domain. Chapter 3 contains a review of robust control in the face of additive unstructured perturbations in the frequency-domain. There are several limitations to the current theory. These include problems with handling poles on the jw axis and unstable poles with a multiplicity greater than 1. The model for the lateral dynamics of an automobile contains 2 poles located at the origin. In Chapter 3, the current theory was extended to allow dealing with poles on the jw axis and unstable poles with a multiplicity greater than 1.

Using the interpolation techniques developed in Chapter 3, a robust lateral controller was designed in Chapter 4. The uncertainty was characterized by simulating a linearized bicycle model of the lateral dynamics as the velocity and cornering stiffness were varied. The uncertainty was then expressed as a fraction of the nominal plant transfer function. The robust lateral controller designed in Chapter 3 was implemented on a test vehicle, and the test results are summarized in Chapter 5. A PID controller was also tested for comparison purposes. Both controllers performed satisfactorily, but several difficulties were found with the robust control approach employed. The test results were very comparable to the results being obtained at PATH [66].

The robust controller had difficulty zeroing out steady state-error because of a lack of integration action. Steady-state errors were caused by center steering offsets, backlash in the steering actuator, and road disturbances. These types of uncertainties were not included in the robust controller design, so therefore it is reasonable that the controller would have difficulty handling them. In addition, the uncertainty bound used for the robust controller design was rather conservative. This conservative modeling of parameter uncertainties (velocity and cornering stiffness) reduced the gain of the robust controller, which led to the

difficulty in handling other types of uncertainties, nonlinearities, and disturbances.

In Chapter 6, two other robust control techniques were applied to the lateral control problem. The first, simultaneous stabilization of a family of plants, used Kharitonov's Theorem to help construct a stabilizing compensator. This approach yielded a lead compensator, which could be tweaked using classical frequency-domain techniques. The simulated performance of the simultaneously stabilizing controller was better than the robust controller described in Chapter 4. The second controller designed in Chapter 6 involved a multiobjective design. The sensitivity function $S(s)$ was minimized to improve disturbance rejection, while the derivative of the control sensitivity $R(s)$ was minimized to reduce the frequency content of the steering actuator. Although this approach used a conservative bound, the simulation results were also acceptable.

The contributions of this dissertation are made in two areas. First, Kimura's robust control approach for systems with additive unstructured perturbations in the frequency domain was extended to include unstable poles with multiplicity, and poles on the jw axis. A modified mapping for the Nevanlinna-Pick interpolation algorithm was developed to handle poles on the jw axis. A technique for handling interpolation points with multiplicity was also presented. These new interpolation techniques are applicable to other robust control approaches involving interpolation. The second contribution of this dissertation is the exploration of implementation issues with this robust control approach applied to the lateral control of highway vehicles.

The next section makes recommendations for future research.

7.2 Recommendations for Future Research

The motivation for applying robust control techniques to the lateral control problem was the nature of the control problem. Parameters were subject to change, and there was some uncertainty to the parameter changes. In addition, using a linearized model for the lateral dynamics added more uncertainty to the system modeling. From previous experience with adaptive controllers [14], there was also some concern about the magnitude of the steering command causing saturation, as well as driver discomfort. These concerns are not isolated to the lateral control of highway vehicles. Modeling uncertainty, nonlinearities, parameter changes, and saturation appear when almost every theoretical algorithm is used on a physical system.

Some of these difficulties are embodied in Ackermann's "*Three Basic Rules of Robust Control*" shown below [4].

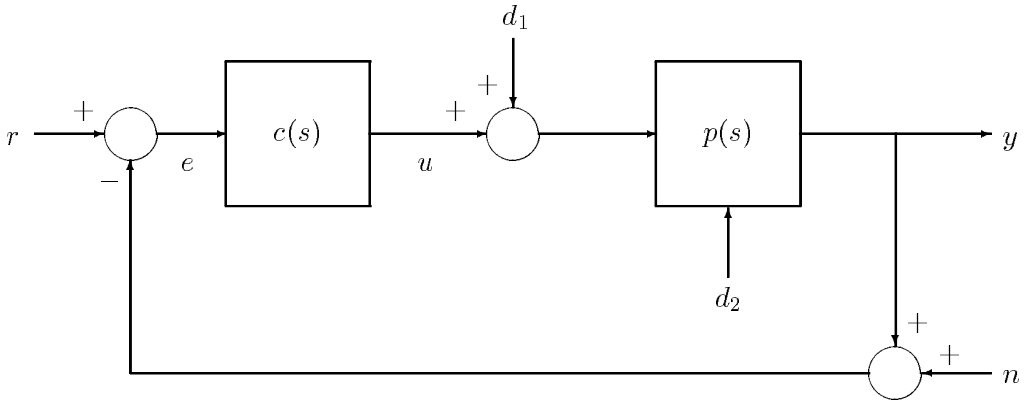
- **Rule 1** - Require robustness of a control system only for physically motivated parameter values and not with respect to arbitrarily assumed uncertainties of the mathematical model.
- **Rule 2** - When you close a loop with actuator constraints, leave a slow system slow and leave a fast system fast.
- **Rule 3** - Be pessimistic in analysis, then you can afford to be an optimist in the design.

Modeling the system uncertainty in the frequency-domain as unstructured additive perturbations as the velocity and cornering stiffness varied obeyed Rules 1 and 3. The robust-

ness was motivated by physical parameter changes, the velocity of a car changes, as does the cornering stiffness. The uncertainty bound chosen was also very conservative. Rule 2 was also followed in that we kept a slow system slow, therefore avoiding any actuator saturation problems. Even though Ackermann's Basic Rules of Robust control were not violated, the performance of the robust controller still left some room for improvement when implemented on the physical system. The problems with the robust controller arose from actuator backlash (a nonlinearity), center steering offsets (a modeling error), and road disturbances. For the most part, nonlinearities like backlash are not a frequency-domain phenomenon. Therefore, one area requiring more research is how to incorporate nonlinearities like backlash into a frequency-domain design. One method for handling nonlinearities is to use Quantitative Feedback Theory (QFT), but this approach relies on finding an "equivalent linear time-invariant" plant set whose responses follow the nonlinear system [44].

For the lateral control problem, road disturbances, center steering offsets, and actuator backlash are structured uncertainty now that they have been identified as problematic. These disturbances are shown in Figure 7.1. The performance deficiencies of the robust controller designed in Chapter 4 can probably be eliminated by using a less conservative description of the plant uncertainty incorporating some knowledge of the structure of the uncertainty into the design. This is another area requiring further research.

The simultaneous stabilization technique used in Chapter 6 could be developed into a powerful design tool for a limited class of minimum phase systems. The structure of the controller is known at the start of the design, so it is possible to incorporate other design criteria into the controller design. For example, the lead compensator designed for the lateral control problem could be tweaked using frequency-domain techniques looking at



where d_1 represents center steering offsets, actuator backlash, and measurement noise, d_2 represents road disturbances, and n represents measurement noise

Figure 7.1: Lateral Control System Model

the phase margin of the nominal plant, while still guaranteeing stability for the family of plants. Although this would be an iterative trial and error process, many control engineers are already familiar with performance constraints related to frequency-domain information. By incorporating the simultaneous stabilization approach into the design, more robustness properties of the control system could be incorporated (or known in advance).

One topic that was highlighted while preparing this dissertation is the difference between analytic feedback design approaches (i.e., interpolation) and trial-and-error design approaches (i.e., Nyquist stability criterion). Analytic design techniques are attractive because they include the following two elements: conditions for existence of a solution, and an algorithm which is guaranteed to find the solution when it exists [26]. However, when implementing a control algorithm on a physical system (or even in simulation), the first cut at a controller rarely meets all of the design requirements. Therefore, some modification or tweaking is required. For trial-and-error methods, this is relatively easy because the original design was reached by trial-and-error. For some analytic designs though, it sometimes is not intuitively obvious what to change to try and improve the performance. This is especially true for interpolation techniques. On the other hand, linear-quadratic theory is an example of an analytic approach where is it relatively straightforward and intuitive to tweak the controller by changing the weighting matrices. Therefore, more research is needed on the "what to do next" step in tweaking analytic design methods, especially those involving interpolation, to improve system performance.

Based on the results of this research, several recommendations for further study of the lateral control problem are outlined below:

- Develop better analytical methods to include both performance and robustness in the

controller design.

- Include nonlinearities like steering actuator backlash in the design to improve robustness.

- Use more knowledge of the plant uncertainty in the controller design to improve system performance.

- Although a fixed controller for all velocities and operating conditions is probably impossible, the use of several fixed controllers for the various operating conditions might be preferable to gain-scheduling as a function of many parameters. This would reduce the sensor requirements for the gain-scheduling.

- More research is needed to investigate the tradeoffs between increased system complexity (i.e., more sensors) for improved performance, and the increased chances of failure from greater system complexity.

Appendix A

Hardware Documentation

**

Appendix B

MATLAB Programs

```
% M-FILE FOR CAR BODE PLOTS / ROBUST DESIGN
% PROGRAM DETERMINES PLANT UNCERTAINTY BOUND
% Define Nominal Values
L1=1.17;        % does not change
L2=1.42;        % does not change
LS=2;           % distance to sensor from cg
V=8;            % 5 m/s to 10 m/s, 18 kmph to 36 kmph
C=42000;        % 0.85 to 1.15
M=1590;         % constant
I=3200;         % constant

a11=-(4*C)/V/M;
a12=-(V+(2*C*(L1-L2)/V/M));
a21=-(2*C*(L1-L2)/V/I);
a22=-(2*C*(L1^2+L2^2)/V/I);
b1=2*C/M;
b2=2*L1*C/I;

A=[a11 a12 0 0; a21 a22 0 0; 0 1 0 0; 1 LS V 0];
B=[b1;b2;0;0];
C=[0 0 0 1];
D=0;

r=zeros(31,1);
phase=zeros(31,1);
ratio=zeros(31,1);

pause;

w=logspace(-1,2,31);

Gnom=freqresp(A,B,C,D,1,sqrt(-1)*w);

Ci=35700:1260:48300;
Vx=5:0.5:10;

for i=1:11,
i,
for x=1:11,
C=Ci(i);
I=I;
M=M;
V=Vx(x);

a11=-(4*C)/V/M;
a12=-(V+(2*C*(L1-L2)/V/M));
a21=-(2*C*(L1-L2)/V/I);
a22=-(2*C*(L1^2+L2^2)/V/I);
b1=2*C/M;
b2=2*L1*C/I;

A=[a11 a12 0 0; a21 a22 0 0; 0 1 0 0; 1 LS V 0];
B=[b1;b2;0;0];
C=[0 0 0 1];
G=freqresp(A,B,C,D,1,sqrt(-1)*w);
for y=1:31,
if abs(G(y)-Gnom(y))>r(y),
temp = G(y)-Gnom(y);
r(y)=abs(temp);
ratio(y)=r(y)/abs(Gnom(y));
phase(y)=(180/pi)*atan2(imag(temp),real(temp));
end,    % if
end,    % y loop
end     % x loop
end     % i loop
```

```
% M-FILE FOR CAR BODE PLOTS / ROBUST DESIGN
% PROGRAM PLOTS FAMILY OF LANE CHANGE RESPONSES
% Define Nominal Values
L1=1.17;        % does not change
L2=1.42;        % does not change
LS=2;           % distance to sensor from cg
V=8;            % 5 m/s to 10 m/s, 18 kmph to 36 mph
C=42000;        % 0.85 to 1.15
M=1590;         % constant
I=3200;         % constant

a11=-(4*C)/V/M;
a12=-(V+(2*C*(L1-L2)/V/M));
a21=-(2*C*(L1-L2)/V/I);
a22=-(2*C*(L1^2+L2^2)/V/I);
b1=2*C/M;
b2=2*L1*C/I;

A=[a11 a12 0 0; a21 a22 0 0; 0 1 0 0; 1 LS V 0];
B=[b1;b2;0;0];
C=[0 0 0 1];
D=0;

t=(0:0.1:20)';
u=1.5*(1.+tanh(t-5));

plot(t,u);

pause;

numc=conv([2 1.5 .25],[1 12.9683 47.9080]);
denc=114.2552*conv([0.5644 2.5644 1.1411],[1 7.1675 31.4366]);

[num,den]=ss2tf(A,B,C,D);
num=[num(3) num(4) num(5)];
[numsys,densys]=cloop(conv(num,numc),conv(den,denc),-1);
[y,x]=lsim(numsys,densys,u,t);
plot(t,y);
hold on

Ci=35700:1260:48300;
Vx=5:0.5:10;

for i=1:11,
i,
for x=1:11,

C=Ci(i);
I=I;
M=M;
V=Vx(x);

a11=-(4*C)/V/M;
a12=-(V+(2*C*(L1-L2)/V/M));
a21=-(2*C*(L1-L2)/V/I);
a22=-(2*C*(L1^2+L2^2)/V/I);
b1=2*C/M;
b2=2*L1*C/I;

A=[a11 a12 0 0; a21 a22 0 0; 0 1 0 0; 1 LS V 0];
B=[b1;b2;0;0];
C=[0 0 0 1];

[num,den]=ss2tf(A,B,C,D);
num=[num(3) num(4) num(5)];
[numsys,densys]=cloop(conv(num,numc),conv(den,denc),-1);
[y,x]=lsim(numsys,densys,u,t);
plot(t,y);

end     % x loop
end     % i loop
```

```
hold off
```

```
%m-file to determine coefficents of simultaneously stabilizing
%controller using Kharitonov Test

%coefficient variations
q1 = [ 131.3935 97.1169 ];
q2 = [ 3249.1 887.514 ];
q3 = [4750.1 2595.1 ];
q4 = [ 1 1 ];
q5 = [ 44.7406 16.5346 ];
q6 = [ 499.662 72.7904 ];

plus=1;
minus=2;
e0=10;
e1=0.05;
e2=0.001;

%check e0
d1pm=[e0 (q1(plus)+e0*q5(plus)) (q1(minus)+q2(minus)+e0*q6(minus))
(q2(minus)+q3(minus)) q3(plus) ]

d1pp=[e0 (q1(minus)+e0*q5(minus)) (q1(minus)+q2(minus)+e0*q6(minus))
(q2(plus)+q3(plus)) q3(plus) ]

d1mp=[e0 (q1(minus)+e0*q5(minus)) (q1(plus)+q2(plus)+e0*q6(plus))
(q2(plus)+q3(plus)) q3(minus) ]

d1mm=[e0 (q1(plus)+e0*q5(plus)) (q1(plus)+q2(plus)+e0*q6(plus))
(q2(minus)+q3(minus)) q3(minus) ]

roots(d1pm)
roots(d1pp)
roots(d1mp)
roots(d1mm)

%check e1
d2pm=[e1*q4(minus) (e0+e1*q5(plus)) (q1(plus)+e0*q5(plus))
(q1(minus)+q2(minus)+e0*q6(minus)) (q2(minus)+q3(minus)) q3(plus) ]

d2pp=[e1*q4(plus) (e0+e1*q5(plus)) (q1(minus)+e0*q5(minus))
(q1(minus)+q2(minus)+e0*q6(minus)) (q2(plus)+q3(plus)) q3(plus) ]

d2mp=[e1*q4(plus) (e0+e1*q5(minus)) (q1(minus)+e0*q5(minus))
(q1(plus)+q2(plus)+e0*q6(plus)) (q2(plus)+q3(plus)) q3(minus) ]

d2mm=[e1*q4(minus) (e0+e1*q5(minus)) (q1(plus)+e0*q5(plus))
(q1(plus)+q2(plus)+e0*q6(plus)) (q2(minus)+q3(minus)) q3(minus) ]

roots(d2pm)
roots(d2pp)
roots(d2mp)
roots(d2mm)

%check e2
d3pm=[e2 (e1*q4(minus)+e2*q5(minus)) (e0+e1*q5(plus)+e2*q6(plus))
(q1(plus)+e0*q5(plus)) (q1(minus)+q2(minus)+e0*q6(minus))
(q2(minus)+q3(minus)) q3(plus) ]

d3pp=[e2 (e1*q4(plus)+e2*q5(plus)) (e0+e1*q5(plus)+e2*q6(plus))
(q1(minus)+e0*q5(minus)) (q1(minus)+q2(minus)+e0*q6(minus))
(q2(plus)+q3(plus)) q3(plus) ]

d3mp=[e2 (e1*q4(plus)+e2*q5(plus)) (e0+e1*q5(minus)+e2*q6(minus))
(q1(minus)+e0*q5(minus)) (q1(plus)+q2(plus)+e0*q6(plus))
(q2(plus)+q3(plus)) q3(minus) ]

d3mm=[e2 (e1*q4(minus)+e2*q5(minus)) (e0+e1*q5(minus)+e2*q6(minus))
(q1(plus)+e0*q5(plus)) (q1(plus)+q2(plus)+e0*q6(plus))
(q2(minus)+q3(minus)) q3(minus) ]

roots(d3pm)
```

```
roots(d3pp)
roots(d3mp)
roots(d3mm)
```

Appendix C

Vehicle Software

```
/*****************************************/
/*  JIMMY data acquisistion/control program */
/*  started 10/10/94                       */
/*  last modified   12/16/94               */
/*  written by: Ray Byrne, 9616            */
/*****************************************/

/****************** INCLUDES ****************************/

#include <stdio.h>
#include <conio.h>
#include <dos.h>
#include <io.h>

/****************** DEFINES ****************************/

/* MATRIX A/D BOARD */
#define AD_CH_SELECT    0xFE0B
#define AD_DATA         0xFE0C

/* RTC defines */
#define MAIN_STATUS     0xFDC0
#define INTERRUPT_0     0xFDC3
#define Periodic_Flag   0xFDC3

/* control defines */
#define valve_max_p     2298
#define valve_max_n     1798
/****************** GLOBAL VARIABLES ******************/

FILE *data_file;

unsigned char   Flag = 0,
dummy;

unsigned int    i,
count = 0,
count1 = 0,
data = 0,
raw_sensor = 2048,
prev_sensor = 2048,
prev_prev_sensor = 2048,
raw_joystick = 2048,
raw_pressure,
prev_joystick,
raw_steering = 2048,
prev_steering = 2048,
center_steer,
prev_steering;

int     steer_error,
downcount = 0,
sensor,
joystick,
comm_steering = 2048,
steering,
road_error,
error_hist[5],
valve_out;

long int        steer_integral = 0;

float   valve_out_fp,
comm_steering_fp,
road_prop,
road_deriv,
road_integral,
sensor_hist[4],
output_hist[5],
filt_sensor[100];       /* array of filtered sensor data */

/****************** FUNCTION PROTOTYPES ****************/

void interrupt (*old_RTC_vector) ();
```

```
void interrupt RTC_service (void);
void initVectors (void);
void restoreVectors (void);
void Init_RTC (void);
void create_files (void);
void close_files (void);

/****************** FUNCTIONS ************************/
void initVectors (void)
{
disable();      /* disable interrupts */
old_RTC_vector = getvect (0x0F);
setvect (0x0F, RTC_service);
dummy = inportb (0x21);
outportb (0x21, (dummy & 0x7f));        /* unmask IR 7 */
enable();       /* enable interrupts */
return;
}       /* end of initVectors() */

void restoreVectors (void)
{
disable();      /* disable interrupts */
setvect (0x0F, old_RTC_vector);
dummy = inportb (0x21);                 /* get interrupt mask */
outportb (0x21, (dummy | 0x80));        /* mask out RTC IR 7 */
outportb (0xFDC0, 0x44);        /* select Int 0 Register */
outportb (0xFDC3, 0);           /* clear timer interupts */
outportb (0xFDC0, 0x44);        /* clear interupt in main register */
enable();       /* enable interrupts */
return;
}       /* end of restoreVectors */

void interrupt RTC_service (void)
{
Flag = 1;
outportb (MAIN_STATUS, 0x04);   /* set bit 6 = 0, bit 7 = 0 */
dummy = inportb(Periodic_Flag);
outportb (AD_CH_SELECT, 0x00);  /* select CH 0 */
do
{
raw_sensor = inport (AD_DATA);
}       while (raw_sensor >= 0x8000);

outportb (AD_CH_SELECT, 0x02);  /* select CH 2 */
do
{
raw_steering = inport (AD_DATA);
}       while (raw_steering >= 0x8000);
if (abs(raw_steering - prev_steering) > 300)
raw_steering = prev_steering;
road_error = 2200 - filt_sensor[0];
road_integral = 0.95*road_integral + road_error;
downcount++;
if(downcount > 9)
{
error_hist[4] = error_hist[3];
```

```
error_hist[3] = error_hist[2];
error_hist[2] = error_hist[1];
error_hist[1] = error_hist[0];

error_hist[0] =  road_error;

output_hist[4] = output_hist[3];
output_hist[3] = output_hist[2];
output_hist[2] = output_hist[1];
output_hist[1] = output_hist[0];


output_hist[0] =  2.6971*output_hist[1] - 2.6226*output_hist[2] +
1.076*output_hist[3] - 0.1532*output_hist[4] +
0.10646*error_hist[0] - 0.256*error_hist[1] +
0.2035*error_hist[2] - 0.060158*error_hist[3] +
0.006241*error_hist[4];

downcount = 0;

}       /* end of 0.1 sec loop */

comm_steering = 2140 + 2*output_hist[0] + 0.001*road_integral;

steer_error = raw_steering - comm_steering;
steer_integral = steer_integral + steer_error;
valve_out_fp = 6 * steer_error + 0.01 * steer_integral;
valve_out = (valve_out_fp + 2048);

if (valve_out > valve_max_p) valve_out = valve_max_p;   /* limit speed */
if (valve_out < valve_max_n) valve_out = valve_max_n;   /* of steering */

outportb (0xfe10, (valve_out & 0xff));
outportb (0xfe11, (valve_out >> 8));

outportb (0x1f8, 0xff);
outportb (0x1f8, 0x00); /* toggle watchdog */

prev_steering = raw_steering;

outportb (0x20, 0x20);  /* non-specific EOI */
return;

}       /* end of interrupt service routine */

void Init_RTC (void)
{
outportb (MAIN_STATUS, 0x00);   /* set bit 6 = 0, bit 7 = 0 */
dummy=inportb (Periodic_Flag);  /* read clears flag register */
outportb (MAIN_STATUS, 0x44);   /* set bit 6 = 1, bit 7 = 0 */
outportb (INTERRUPT_0, 0x10);   /* generate interrupt @ 0.01 sec */

return;

}       /* end of Init_RTC() */

void create_files (void)
{
data_file = fopen("r:data.m", "a");

return;

}       /* end of create_files() */

void close_files (void)
{
fclose (data_file);

return;

}       /* end of close_files() */

/******************* MAIN PROGRAM ***********************/
```

```
void main()
{
create_files();
Init_RTC();             /* initialize real time clock */
initVectors();  /* initialize ISR and IR mask */
for (i=0;i<100;i++) filt_sensor[i] = 2048;      /* init filter */
for (i=0;i<6;i++) error_hist[i] = 0;
for (i=0;i<6;i++) output_hist[i]= 0;
outportb (AD_CH_SELECT, 0x02);  /* select CH 0 */
do
{
center_steer = inport (AD_DATA);
}   while (center_steer >= 0x8000);

while (!kbhit())
{
if (Flag == 1)
{
count++;
Flag = 0;
for(i=1;i<100;i++)
{
filt_sensor[100-i]=filt_sensor[99-i];
}       /* end of for loop */

filt_sensor[0] = (7.064076e-4)*(raw_sensor + 2*prev_sensor
+ prev_prev_sensor)
+ 1.907232*filt_sensor[1] -
0.9100573*filt_sensor[2];
prev_prev_sensor = prev_sensor;
prev_sensor = raw_sensor;

fprintf(data_file,"%d %d %d %f %f \n",
raw_sensor, road_error, raw_steering, output_hist[0],
filt_sensor[0]);
}       /* end of if statement */
}       /* end of while loop */

FINISH:
restoreVectors();
close_files();

}       /* end of main */
```

Bibliography

[1] J. Ackermann. Robust car steering by yaw rate control. In *Proc. of the 1990 CDC*, pages 2033–2034, Honolulu, 1990.

[2] J. Ackermann. Robust car steering by yaw rate feedback. In *Control of Uncertain Dynamic Systems*, pages 125–139, CRC Press, Boca Raton, 1991.

[3] J. Ackermann. Velocity-independent yaw eigenvalues of four wheel-steering automobiles. In M. Mansour, S. Balemi, and W. Truol, editors, *Robustness of Dynamic Systems with Parameter Uncertainties*, pages 291–302, Birkhauser Verlag, Basel, 1192.

[4] J. Ackermann, A. Bartlett, D. Kaesbauer, W. Sienel, and R. Steinhauser. *Robust Control, Systems with Uncertain Physical Parameters*. Springer Verlag, London, 1993.

[5] J. Ackermann, J. Guldner, and V. I. Utkin. A robust nonlinear control approach to automatic path tracking of a car. In *Proc. of the International Conference on Control*, pages 196–201, Coventry, England, 1994.

[6] J. Ackermann and W. Sienel. Robust control for automatic steering. In *Proc. of the 1990 ACC*, pages 795–800, San Diego, 1990.

[7] N. I. Akhiezer. *The Classical Moment Problem*. Oliver and Boyd, London, 1965.

[8] A. H. A. H. Ali. *A Survey of Weighting Functions for H_∞ Control.* Technical Report 782, Control System Centre, UMIST, Manchester, UK, May 1993.

[9] F. P. Andresen and L. S. Davis. *Visual Position Determination for Autonomous Land Vehicle Navigation.* Technical Report CAR-TR-100, University of Maryland, College Park, MD, 1984.

[10] D. Aubert and C. E. Thorpe. *Color Image Processing for Navigation:Two Road Trackers.* Technical Report CMU-RI-TR-90-09, Carnegie Mellon University, Pittsburg, 1990.

[11] B. R. Barmish and K. H. Wei. Simultaneous stabilizability of single input-single output systems. In *Proc. of the 7th Intl. Symposium on Mathematical Theory of Networks and Systems*, pages 431–443, Stockholm, 1985.

[12] D. E. Barrick. *Automatic Steering Techniques.* Technical Report EES 202A-2, Ohio State University Engr. Exp. Station, Columbus, OH, Dec. 1961.

[13] L. S. Bonderson. Optimum lateral control for dual-mode vehicles. In *Proc. of the Int. Conf. on Dual-Mode Transportation*, page 98, 1974.

[14] R. H. Byrne and C. T. Abdallah. Design of a model reference adaptive controller for vehicle road following. Accepted in Network, Control, Communication and Computing Technologies for Intelligent Vehicle Highway Systems, 1994.

[15] R. H. Byrne and C. T. Abdallah. Robust lateral control of highway vehicles. To appear in the Proc. of the Intelligent Vehicles '94 Symposium, 1994.

[16] K. H. F. Cardew. *The Automatic Steering of Vehicles - An Experimental System Fitted to a DS 19 Citrogen Car*. Technical Report LR340, Road Research Lab, Great Britain, 1970.

[17] H. Christ, W. Darenberg, F. Panik, and W. Weidemann. Automatic track control of vehicles, theory and experiment. In *Proc. of the 5th VSD - 2nd IUTAM Symposium*, pages 145–164, Vienna, 1977.

[18] W. H. Cormier and R. E. Fenton. On the steering of automated vehicles - a velocity adaptive controller. *IEEE Trans. on Veh. Tech.*, 375–385, Nov. 1980.

[19] W. Darenberg, F. Panik, and W. Weidemann. Design of automatic lateral vehicle controllers. In *Proc. of the 7th IFAC World Congress*, pages 2545–2550, Helsinki, 1978.

[20] L. S. Davis and T. R. Kushner. Road boundary detection for autonomous vehicle navigation. In *Intelligent Robots and Computer Vision*, pages 362–366, 1985.

[21] L. S. Davis, T. R. Kushner, J. L. LeMoigne, and A. M. Waxman. Road boundary detection for autonomous vehicle navigation. *Optical Engineering*, 25(3):409–414, 1986.

[22] L. S. Davis and J. L. LeMoigne. Visual navigation of roadways. In *Proc. of the Intl. Conf. on Intelligent Systems*, pages 21–30, Amsterdam, 1986.

[23] P. H. Delsarte, Y. Genin, and Y. Kamp. On the role of the Nevanlinna-Pick problem in circuit and system theory. *Int. J. Circuit Theory Application*, 9:177–187, 1981.

[24] S. J. Dickinson and L. S. Davis. A flexible tool for prototyping ALV road following algorithms. *IEEE Trans. of Robotics and Automation*, 6(2):232–242, 1990.

[25] E. D. Dickmans and A. Zapp. Autonomous high speed road vehicle guidance by computer vision. In *Proc. of the IFAC 10th Annual World Congress*, pages 221–226, Munich, 1987.

[26] P. Dorato. *Analytic Feedback System Design: An Interpolation Approach.* Technical Report, University of New Mexico, 1994.

[27] P. Dorato, L. Fortuna, and G. Muscato. *Robust Control for Unstructured Perturbations - An Introduction.* Springer Verlag, New York, 1992.

[28] P. Dorato and Y. Li. A modification of the classical Nevanlinna-Pick interpolation algorithm with applications to robust stabilization. *IEEE Trans. Automat. Contr.*, AC-31:645–648, 1986.

[29] J. C. Doyle, K. Glover, P. P. Khargonekar, and B. A. Francis. State-space solutions to standard H^2 and H^∞ control problems. *IEEE Trans. Automat. Contr.*, AC-34:831–847, 1989.

[30] J. C. Doyle and G. Stein. Multivariable feedback design: concepts for classical/modern synthesis. *IEEE Trans. Automat. Contr.*, AC-26:4–16, 1981.

[31] R. Fenton and I. Selim. On the optimal design of an automatic lateral controller. *IEEE Trans. on Veh. Tech.*, 37(2):108–113, 1988.

[32] R. E. Fenton. IVHS/AHS: driving into the future. *IEEE Control Systems Magazine*, 14(6):13–20, December 1994.

[33] R. E. Fenton, R. L. Cosgriff, K. W. Olson, and L. M. Blackwell. One approach to highway automation. *Proc. of the IEEE*, 56(4):556–566, 1968.

[34] R. E. Fenton and R. J. Mayhan. Automated highway studies at the ohio state university - an overview. *IEEE Trans. on Veh. Tech.*, 40(1):100–113, 1991.

[35] R. E. Fenton, R. J. Mayhan, G. M. Takasaki, and J. Glimm. *Fundamental Studies in Automatic Vehicle Control.* Technical Report 784712-2, Ohio State University, Columbus, OH, May 1979.

[36] R. E. Fenton, G. C. Melocik, and K. W. Olson. On the steering of automated vehicles - theory and experiment. *IEEE Trans. Automat. Contr.*, AC-21:306–315, 1976.

[37] R. E. Fenton and S. S. Murphy. A microprocessor-based vehicle lateral controller. In *Proc. of the 31st Veh. Tech. Soc. Conf.*, pages 381–386, Washington, DC, 1981.

[38] F. R. Gantmacher. *The Theory of Matrices.* Volume 1, Chelsea, New York, 1960.

[39] K. Gardels. *Automatic Car Controls for Electronic Highways.* Technical Report GMR-276, GM Research Labs, Warren, MI, 1960.

[40] N. K. Gupta. Frequency-shaped cost functionals: extension of linear-quadratic-gaussian design methods. *J. Guidance and Control*, 3(6):529–535, 1980.

[41] K. Hayafune and H. Yoshida. Control method of autonomous vehicle considering compatability of riding comfort and vehicle controllability. In *Automated Highway/Intelligent Vehicle Systems: Technology and Socioeconomic Aspects*, pages 35–40, Society of Automotive Engineers, 1990.

[42] J. K. Hedrick, M. Tomizuka, and P. Varaiya. Control issues in automated highway systems. *IEEE Control Systems Magazine*, 14(6):21–32, December 1994.

[43] T. Hessburg and M. Tomizuka. Fuzzy logic control for lateral vehicle guidance. *IEEE Control Systems Magazine*, 14(4):55–63, August 1994.

[44] I. Horowitz. Quantitative feedback theory. *IEE Proc.*, 129(6):215–226, 1982.

[45] T. Ito, M. Furumata, F. Harashima, H. Inaba, and S. Matsumoto. An automatic driving system of automobiles by guidance cables. In *Proc. of the Int. Automotive Engineering Conference*, Detroit, 1973.

[46] H. Kimura. Robust stability for a class of transfer functions. *IEEE Trans. Automat. Contr.*, AC-29(9):788–793, 1984.

[47] N. A. Lehtomaki, N. R. Sandell, and M. Athans. Robustness results in the linear-quadratic gaussian based multivariable control designs. *IEEE Trans. Automat. Contr.*, AC-26(1):75–92, 1981.

[48] Y. Li. *U-Parameter Design: feedback System Design with Guaranteed Robust Stability.* PhD thesis, University of New Mexico, Albuquerque, NM, 1989.

[49] J. W. Lowrie, M. Thomas, K. Gremban, and M. Turk. The autonomous land vehicle (ALV) preliminary road following demonstration. In *Proc. of the SPIE*, pages 336–350, 1985.

[50] N. Matsumoto and M. Tomizuka. Vehicle lateral velocity and yaw rate control with two independent control inputs. In *Proc. of the 1990 ACC*, pages 1868–1875, San Diego, 1990.

[51] R. J. Mayhan and R. A. Bishel. A two frequency radar for vehicle automatic lateral control. *IEEE Trans. on Veh. Tech*, VT-31(1):32–39, Feb. 1982.

[52] D. Morgenthaler, S. J. Hennessy, and D. DeMenthon. Range-video fusion and comparison of inverse perspective algorithms in static images. *IEEE Trans. on Systems, Man and Cybernetics*, 20(6):1301–1312, 1990.

[53] S. S. Murphy. *A Microprocessor-Based Lateral Controller for an Automated Vehicle.* Master's thesis, Ohio State University, Columbus, OH, 1980.

[54] R. Nevanlinna. Uber beschrankte funktionen die in gegeben punkten vorgeschriedene werte annehmen. *Ann. Acad. Sci. Fenn.*, 9:177–187, 1918.

[55] K. W. Olson, E. R. Sapp, and T. L. Flaig. A system for automatic vehicle guidance. In *Highway Research Record, Proc. of the 48th Annual Meeting*, pages 1–11, 1969.

[56] R. E. Parsons and W. B. Zhang. Program on advanced technology for the highway lateral guidance/control. In *Proc. of the 1st Int'l Conf., Applications of Advanced Technology in Transportation Engineering*, pages 275–280, San Diego, CA, 1989.

[57] S. Pasternack. *Modern Control Aspects of Automatically Steered Vehicles.* Technical Report DOT-TSC-OST-72-3, Transportation Systems Center, Cambridge, MA, December 1971.

[58] R. J. Patton and J. Chen. Robust fault detection of jet engine sensors using eigenstructure assignment. *AIAA Journal of Guidance, Control, and Dynamics*, 1666–1675, 1991.

[59] S. Patwardhan and M. Tomizuka. Robust failure detection in lateral control for IVHS. In *Proc. of the 1992 ACC*, pages 1768–1772, Chicago, 1992.

[60] G. Pick. Uber die beschrankungen analytischer funktionen, welche durch vorgegebenen werte annehmen. *Math Ann.*, 77:7–23, 1916.

[61] D. A. Pommerleau. Progress in neural network-based vision for autonomous robot driving. In *Proc. of the Intelligent Vehicles '92 Symposium*, pages 391–396, Detroit, 1992.

[62] H. Pueng, T. Hessburg, M. Tomizuka, W. B. Zhang, Y. Lin, P. Devlin, S. E. Shladover, and A. Arai. A theoretical and experimental study on vehicle lateral control. In *Proc. of the 1992 ACC*, pages 1738–1742, Chicago, 1992.

[63] H. Pueng and M. Tomizuka. Vehicle lateral control for highway automation. In *Proc. 1990 ACC*, pages 788–794, San Diego, 1990.

[64] E. B. Saff and A. D. Snider. *Fundamentals of Complex Analysis for Mathematics and Science*. Prentice Hall, Englewood Cliffs, NJ, 1976.

[65] S. Seida. ALV closeup: finding the road when offroad. *Proceedings of the SPIE*, 1007:49–55, 1988.

[66] S. Sheikholeslam and C. A. Desoer. Combined longitudinal and lateral control of a platoon of vehicles. In *Proc. of the 1992 ACC*, pages 1763–1767, Chicago, 1992.

[67] S. Shladover, R. Fish, H. Richardson, and D. Wormley. *Analysis and Design of Steering Controllers for Automated Guideway Transit Vehicles*. Technical Report UMTA-RD-MA-11-0023, Massachusets Institute of Technology, Cambridge, MA, Sept. 1976.

[68] S. E. Shladover, C. A. Desoer, J. K. Hedrick, M. Tomizuka, J. Walrand, W. B. Zhang, D. H. McHahon, H. Pueng, S. Sheikholeslam, and N. McKeown. Automatic vehicle

control developments in the PATH program. *IEEE Trans. on Veh. Tech.*, 40(1):114–130, 1991.

[69] S. E. Shladover, D. N. Wormley, H. H. Richardson, and R. Fish. Steering controller design for automated guideway transit vehicle. *Journal of Dynamic Systems, Measurement, and Control*, 100:1–8, March 1978.

[70] L. I. Smilen. Interpolation on the real frequency axis. *IEEE Internat. Conv. Record*, 13:42–50, 1965.

[71] H. Sussmann, E. Sontag, and Y. Yang. A general result on the stabilization of linear systems using bounded controls. In *IEEE Conf. Decision and Control*, pages 1802–1807, San Antonio, 1993.

[72] A. Tannenbaum. Feedback stabilization of linear dynamical plants with uncertainty in the gain factor. *Int. Jour. Control*, 32(1):1–16, 1980.

[73] A. Tannenbaum. Modified Nevalinna-Pick interpolation and feedback stabilization of linear plants with uncertainty in the gain factor. *Int. Jour. Control*, 36(2):331–336, 1982.

[74] C. E. Thorpe and T. Kanade. *1987 Year End Report for Road Following at Carnegie Mellon.* Technical Report CMU-RI-TR-88-4, Carnegie Mellon University, Pittsburg, 1988.

[75] M. Turk, D. G. Morgenthaler, K. D. Gremban, and M. Marra. Video road-following for the autonomous land vehicle. In *Proc. of the 1987 Intl. Conf. on Robotics and Automation*, pages 273–280, Raleigh, 1987.

[76] R. S. Wallace. Robot road following by adaptive color classification and shape tracking. In *Proc. of the 1987 Intl. Conf. on Robotics and Automation*, pages 258–263, Raleigh, 1987.

[77] J. L. Walsh. *Interpolation and Approximation.* Volume XX, American Mathematical Society, Providence, RH, 1956.

[78] A. M. Waxman, J. L. LeMoigne, L. S. Davis, B. Srinivasan, T. R. Kushner, E. Liang, and T. Siddalingaiah. A visual navigation system for autonomous land vehicles. *IEEE Journal of Robotics and Automation*, RA-3(2):124–141, 1987.

[79] J. Y. Wong. *Theory of Ground Vehicles.* John Wiley & Sons, New York, 2 edition, 1993.

[80] Y. Yang, H. J. Sussmann, and E. D. Sontag. Stabilization of linear systems with bounded controls. In *IFAC Nonlinear Control Systems*, pages 51–56, Bordeaux, France, 1992.

[81] D. C. Youla and M. Saito. Interpolation with positive real functions. *J. Franklin Inst.*, 284:77–108, 1967.

[82] G. Zames and B. A. Francis. Feedback, min max sensitivity, and optimal robustness. *IEEE Trans. Automat. Contr.*, AC-28:585–601, 1983.

[83] W. Zhang, R. E. Parsons, and T. West. An intelligent roadway reference system for vehicle lateral guidance/control. In *Proc. of the 1990 ACC*, pages 281–286, San Diego, 1990.

[84] V. K. Zworykin and L. Flory. Electronic control of motor vehicles on the highway. In *Proc. of the 37th Annual Meeting of the Highway Research Board*, 1958.