Air-Fuel Ratio Control in Spark Ignition Internal Combustion Engines Using Switching LPV Techniques

by

Marius Postma

B.A.Sc., University of British Columbia, 2008

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Applied Science

in

THE FACULTY OF APPLIED SCIENCE
(Mechanical Engineering)

The University Of British Columbia
(Vancouver)

December 2010

Abstract

The Three-Way Catalytic Converter (TWC) is a critical component for the mitigation of tailpipe emissions of modern Internal Combustion (IC) engines. Because the TWC operates effectively only when a stoichiometric ratio of air and fuel is combusted in the engine, accurate control of the air-fuel ratio is required. To track the desired ratio, a switching Linear Parameter Varying (LPV) air-fuel ratio feedback controller, scheduled based on engine speed and air flow, and providing guaranteed L_2 performance, is introduced. The controller measures the air-fuel ratio in the exhaust flow using a Universal Exhaust Gas Oxygen (UEGO) sensor and adjusts the amount of fuel injected accordingly.

A detailed model of the air-fuel ratio control problem is developed to demonstrate the non-linear and parameter-dependent nature of the plant, as well as the presence of pure delays. The model's dynamics vary considerably with engine speed and air flow. A simplified model, widely used in literature and known as a First Order Plus Dead Time (FOPDT) model, is then derived. It effectively captures the control problem using a model which is linear but parameter-varying with engine speed and air flow.

Large variation of the FOPDT model across the engine's operating range has led to conservative LPV controllers in previous literature. For this reason, the operating range is divided into smaller subregions, and an individual LPV controller is designed for each subregion. The LPV controllers are then switched based on the current engine speed and air flow and are collectively referred to as a switching LPV controller. The controller design problem is expressed as a Linear Matrix Inequality (LMI) convex optimization problem which can be efficiently solved using available LMI techniques.

Simulations are performed and the air-fuel ratio tracking performance of the switching LPV controller is compared with that of conventional controllers including, H_∞ and LPV, as well as a novel adaptive controller. The switching LPV controller achieves improved performance over the complete operating range of the engine.

Table of Contents

List of Tables

List of Figures

Glossary

ECU Engine Control Unit

FOPDT First Order Plus Dead Time

HEGO Heated Exhaust Gas Oxygen

IC Internal Combustion

IMLLPC Incremental Mode Linear Laguerre Predictive Control

LMI Linear Matrix Inequality

LNT Lean NOx Trap

LPV Linear Parameter Varying

MVM Mean Value Modeling

MAF Manifold Air Flow

MAP Manifold Absolute Pressure

SI Spark Ignition

TWC Three-Way Catalytic Converter or Three-Way Catalyst

UEGO Universal Exhaust Gas Oxygen

Acknowledgments

I wish to expresses my sincere appreciation to my devoted supervisor and teacher, Dr. Ryozo Nagamune, for giving me the wonderful opportunity to grow intellectually under his guidance. For his inspiration, encouragement, patience, friendship, and faith in me, I am forever indebted. I know few people who put their heart and soul into their work like Dr. Nagamune does, and I have enjoyed following his example. Furthermore, I am grateful for his support in the form of a Research Assistantship as well as the equipment necessary for conducting my research.[1]

Special thanks go to both my committee members, Dr. Clarence de Silva and Dr. Farrokh Sassani, whose knowledge and expertise I have benefitted from as early as my undergraduate degree.

Thanks go, also, to Dr. Greg Stewart at Honeywell for donating his personal time and for taking an interest in the current research. His advice, feedback, and assistance has been of great value.

I would like to thank my very dear friends and colleagues in the Control Engineering Laboratory: Mr. Ehsan Azadi Yazdi, Mr. Masih Hanifzadegan, and

[1]This work is supported through a Discovery Grant awarded to Dr. Ryozo Nagamune by the Natural Sciences and Engineering Research Council of Canada.

Mr. Mohammad Sepasi. The assistance, encouragement, and camaraderie they have provided during the period of my research is deeply valued and appreciated. To my other friends at UBC, Jeswin, Sina, and Susana, as well as Hamed and Mahkameh, whom I have had the pleasure of studying with since the very beginning of my career, I extend my deepest gratitude for the support and friendship. To all my friends at home, I extend thanks for patience and understanding while I disappeared into the laboratory for indefinite periods of time.

Finally, I wish to acknowledge and thank my family. To my parents, who have loved me, inspired me, given me so many opportunities and, through example, nurtured a faith in Christ, I owe nothing less than all my life. I have been blessed with two beautiful sisters who have always encouraged me and, when necessary, endured me, and I thank and cherish them both.

Dedication

To my Parents,

an insufficient token of my appreciation of their unwavering love and faithfulness

Chapter 1

Introduction

1.1 Motivation

Modern automotive engines have benefited from the use of the Three-Way Catalytic Converter (TWC) since it was introduced in production automotive vehicles in 1981 as a response to increasingly stringent tailpipe emission standards. The TWC is situated in the exhaust line of an automotive engine and improves the quality of exhaust gasses by reducing nitrogen oxides to nitrogen and oxygen, oxidizing carbon monoxide to carbon dioxide, and oxidizing hydrocarbons to carbon dioxide and water [16]. More generally, the TWC can be thought of as a buffer to small air and fuel flow variations in the exhaust gas as it is capable of rejecting small disturbances to the air-fuel ratio. The TWC achieves this function by storing oxygen on the catalytic surface when an excess is present and donating it back to the exhaust gas when a deficiency exists.

The effective operation of the TWC relies heavily on two factors: Firstly, the oxygen storage level on the catalytic surface should be near half of capacity and, secondly, the air-fuel ratio of the incoming exhaust gasses should be close to stoichiometric. The air-fuel ratio of the exhaust gas is said to be stoichiometric when the correct proportion of air and fuel was combusted in the engine, yielding an exhaust gas with neither an excess or deficiency of oxygen. Accurately maintaining the desired air-fuel ratio in the engine is therefore essential to ensure good health and operation of the TWC and, consequently, low tailpipe emissions.

When the driver gives a throttle command by lowering the throttle pedal, the throttle valve is opened either mechanically or by the Engine Control Unit (ECU) if the vehicle uses a drive-by-wire throttle system. The opening throttle valve allows more air to be drawn into the engine's cylinders. In order to compensate for the changing air flow and maintain a desired air-fuel ratio, the ECU adjusts the fuel flow by changing the duration of the fuel injection pulses.

Conventional control strategies have relied heavily on static maps and feed-forward controllers to determine the correct fueling rate. These controllers maintain the desired air-fuel ratio by estimating the air flow into the cylinders and simply commanding a proportional fuel flow. A shortcoming of the feed-forward controller architecture is that it does not compensate for the following situations that result in unknown disturbances on the actual air-fuel ratio achieved:

- Aging of the sensors used in the estimation of the air flow, causing a loss of accuracy.

- Inaccuracy of the modeled dynamics, stored in the ECU's memory, used to estimate the air flow into the cylinders due to engine aging and plant identification deficiencies.

- Variations in the oxygen concentration of the air flow due to altitude, weather conditions, and fuel canister purge cycles (which is when fuel vapors trapped inside the fuel tank are periodically released into the air flowing into the engine).

- Clogging or wear of the fuel injectors, causing the actual amount of fuel being injected to differ from the amount commanded by the ECU.

- Variations in the quality and composition of fuel.

The introduction of the Universal Exhaust Gas Oxygen (UEGO) sensor has made it possible to measure the air-fuel ratio in the exhaust flow and rely on feedback [4]. Using a feedback controller in conjunction with a feedforward controller enables fast and accurate air-fuel ratio control. The feedback controller takes a measurement of the air-fuel ratio in the exhaust flow and rejects disturbances to the desired air-fuel ratio. The feed-forward controller estimates the air flow into the cylinders.

1.2 Problem Statement

The current research focuses specifically on the normally aspirated, port injected, Spark Ignition (SI) Internal Combustion (IC) engine which is commonly used in personal vehicles. Figure 1.1 shows the air and fuel paths through one cylinder of

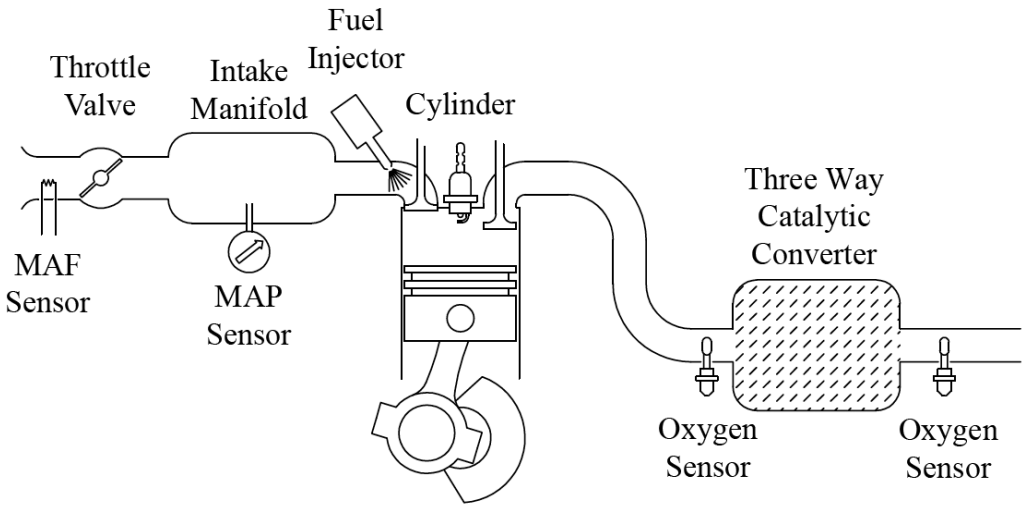


Figure 1.1: Diagram of Engine Showing Air and Fuel Path

the engine. The driver varies the air flow allowed into the system by opening and closing the throttle valve. In the intake manifold, the flow of air is divided into a separate stream for each cylinder. Fuel is injected separately into each air stream before it enters the cylinder, the amount injected being controlled by the ECU through varying the length of the injection pulses. After the air and fuel mixture have completed the combustion stroke in each cylinder, the separate flows are combined in the exhaust manifold before traveling to the TWC. A measurement of the air-fuel ratio is taken using UEGO sensors mounted in the exhaust line before and after the TWC.

The air mass flow rate, which is controlled by the driver, is considered an uncontrolled input to the plant but is assumed to be available. Although this thesis does not discuss them, methods have been developed to measure or estimate the air flow into the cylinders of normally aspirated IC engines based on Manifold

Air Flow (MAF) and Manifold Absolute Pressure (MAP) sensor readings in e.g. [9], [14], [34], [35], and [36], and range from simple schemes like the speed density method to advanced observers that compensate for sensor and air flow dynamics. Because the fuel mass flow rate is closely proportional to the width of the fuel injection pulses which are controlled by the ECU, the rate of fuel injected is considered as the controlled input to the plant. An air-fuel ratio measurement is available in the exhaust line both before and after the TWC. The method in this thesis uses the air-fuel ratio measurement of the first UEGO sensor as the output of the plant. Control schemes exist that use the measurement of the second UEGO sensor installed after the TWC to estimate the oxygen storage level in the TWC (see e.g. [3], [7], [10], [11], and [25]) but are beyond the scope of this thesis. These methods may build on the control loop presented in this thesis as the inner of two cascaded loops.

The closed-loop air-fuel ratio control problem is non-trivial and a poor fit for simple linear control techniques for three reasons. Firstly, the discrete strokes of the IC engine cause a pure delay in the system, meaning that a rational transfer function for the plant does not exist. Secondly, the engine dynamics are time-varying. The aforementioned pure delay, for example, varies considerably as a function of engine speed and air flow. Finally, because the air-fuel ratio is a fraction consisting of the IC engine's two inputs, air and fuel, it is impossible to express the output, the air-fuel ratio, as a linear transfer function of both inputs, and the control problem is therefore non-linear.

1.3 Literature Review

Previously developed methods addressing the air-fuel ratio control problem in SI IC engines are discussed in Section 1.3.1. Literature concerning controller design techniques influencing the current research is given in Section 1.3.2.

1.3.1 Air-Fuel Ratio Controllers

Current methods that address the air-fuel control problem in SI IC engines include PI-Control [3], H_∞ Robust Control [24], Kalman Filter [26], [37], Adaptive Control [19], [31], [32], [41], [42], Linear Parameter Varying (LPV) Control [44], [45], [47], and others [22], [39]. Robust control methods treat the variations of the IC engine as uncertainty and employ time-invariant controllers with guaranteed performance. However, because the variation in plant dynamics is substantial over the operating range of a normal engine, robust control methods yield undesirably conservative controllers.

Adaptive controllers are able to accommodate the time-varying engine dynamics. An attractive feature of adaptive controllers is that no *a priori* information of the relationship between the engine's operating point and the plant dynamics is necessary. However, adaptation is typically slow and may lose performance and even stability during sudden changes in the engine operating point. A combination between adaptation and gain-scheduling is therefore often used to adress a rapidly moving operating point as is done in [26] and [42]. Adaptive methods generally lack a theoretical guarantee for stability and performance.

Gain-scheduling controllers, on the other hand, are time-varying controllers

that adjust for the variations in the dynamics using *a priori* knowledge of the plant dynamics throughout the operating range. Gain-scheduling methods based on interpolation also share the shortcoming that no theoretical guarantee of stability and performance exists. Considerable advances have been made, however, and in [5], [6], [33], gain-scheduling methods are presented with guaranteed performance. The LPV methods in [47] and [48] use Lyapunov theory developed in [5] to create gain-scheduled controllers with guaranteed L_2 performance. These methods use a single parameter dependent Lyapunov function to guarantee performance over the operating range of the plant. Because the variation of the plant dynamics can be extensive across the full operating range of an SI IC engine, a single Lyapunov function may not necessarily exist or, if it does exist, may lead to an unacceptably conservative controller.

The following subsections take a more detailed look at literature that have addressed the air-fuel ratio control problem in SI IC engines and includes methods based on inverse dynamics, H_∞, LPV-based gain-scheduling, and adaptive control. Similar approaches have been used in Diesel engines (see e.g. [1], [2], and [8]) but are beyond the scope of this thesis.

Detailed Model-Based and Inverse Dynamics Control

One of the early papers on model-based IC engine air-fuel ratio control, [4], gives an introduction to the control problem. The author compares two control architectures: Feed-forward control with emphasis on static maps, and a closed-loop, model-based controller which is obtained from inverting the dynamics of the en-

gine. The dynamics of the SI, IC engine are presented and include many non-linear relationships and time-varying coefficients.

Another model-based controller is presented in [29]. The method uses an observer to estimate the air flow into the cylinders with great accuracy so that a proportional amount of fuel can be injected. Air-fuel ratio measurements from oxygen sensors mounted in the exhaust line serve to improve the accuracy of the estimated observer model.

In [27], a detailed non-linear model is used to capture the dynamics of the SI IC engine and includes air flow dynamics, wall wetting dynamics, engine inertia, process delays and UEGO sensor dynamics. A sliding mode controller is used to regulate the air-fuel ratio.

A disadvantage of closed-loop controllers based on detailed engine models is that their effectiveness heavily rely on accurate identification of the time-varying engine parameters.

H_∞ Control

An SI IC engine with drive-by-wire throttle is considered in [24]. A feedforward controller is used to calculate the base fueling while an H_∞ preview controller provides an adjustment based on feedback from the UEGO sensor mounted in the exhaust line. Because the ECU has control of the position of the throttle valve, and consequently the air flow, in a drive-by-wire configuration, predicted future air flow values are available. The H_∞ preview controller takes advantage of this fact to remove the effect of delay in the closed loop plant.

Adaptive Control

An adaptive controller which compensates for sensor aging is presented in [32]. Because the time constant of a UEGO sensor may increase from about 50*msec* up to 1*sec* during its lifetime, the authors recommend using adaptation law in order to maintain performance of the closed loop controller including the sensor. The variation in sensor dynamics occurs very gradually over the life of the sensor, however, meaning that the adaptation rate may be very slow. The IC engine dynamics are assumed to be known and are modeled as a First Order Plus Dead Time (FOPDT). The time-varying nature of the engine dynamics is beyond the scope of the paper.

Methods presented in [31], [41] and [19] use adaptation to compensate for the time-varying engine dynamics. In each method the engine dynamics are simplified and modeled using an FOPDT transfer function. Adaptation is then used to estimate some or all of the model parameters and a suitable controller is calculated in real-time. Feed-forward controllers are used to augment the adaptive controllers by compensating for the rapidly changing low frequency gain of the plant due to changes in air flow. An adaptive Smith predictor is used in [19] to remove the effect of the time-delay in the closed-loop.

A structure similar to that of a Smith predictor is also used in [42] but, rather than using adaptation, the time-varying delay is computed using *a priory* knowledge and sensor measurements and is scheduled accordingly. The authors introduce an adaptive controller which uses adaptation in both the feed-forward controller and feedback loop for the regulation of both the air-fuel ratio and the TWC

oxygen storage level. A UEGO sensor mounted upstream of the TWC and a Heated Exhaust Gas Oxygen (HEGO) sensor installed downstream of the TWC are used to estimate the oxygen storage level inside the TWC. Adaptation is used to identify the model of the IC engine including wall wetting, exhaust dynamics, and sensor dynamics.

Adaptive Kalman filters are used in [37], [25], and [26]. The method presented in [37] uses an indirect model-reference adaptive controller but assumes that the UEGO sensor dynamics, as well as all system delays in the engine, are known and available in real-time from lookup tables. The parameter identification is only used to estimate the wall wetting dynamics of the engine. In [25], on the other hand, the Kalman filter is used to estimate the time-varying delay and disturbance dynamics and, in [26], it is used to estimate the parameters of a FOPDT model of the IC engine. While the low frequency gain and time constant of the plant dynamics in [26] are automatically estimated using the Kalman filter, the delay is explicitly computed at each time-step and expressed in terms of sample periods. The authors present experimental data confirming the dependence of the delay on the engine speed and air flow.

LPV-Based Control

LPV-based control is used in [44] and [45] to track a reference air-fuel ratio in a lean burn IC engine. Unlike conventional IC engines, lean burn engines do not operate in the narrow band around stoichiometry but instead vary the air-fuel ratio considerably in order to achieve the effective operation of both the TWC and

the Lean NOx Trap (LNT), creating a more challenging tracking problem. The method proposed in [44] uses input shaping on the reference air-fuel ratio in order to improve transient tracking responses. Air flow and engine speed are measured using a hot wire MAF sensor and engine speed sensor, respectively, and are both used to calculate the time-varying delay of the plant in real-time and schedule the controller accordingly.

An air-fuel ratio controller for SI IC engines using LPV gain-scheduling techniques is presented in [47]. The controller takes feedback from the UEGO sensor mounted in the exhaust line. The plant to be controlled, including engine and sensor dynamics, is modeled as an FOPDT which contains a time-varying delay and time constant. A feed-forward controller calculates a base fueling rate which is then adjusted by the feedback controller in a multiplicative manner. The controller design method used is developed in [5] and uses Lyapunov theory to guarantee performance over the operating range of the engine. The method is revisited in [48] and a state-delayed LPV controller is introduced. Rather than using a Padé approximation to model the delay in the plant for controller design, the delay is removed from the dynamics and reintroduced into the LPV form of the plant explicitly as an additional state. The authors of both papers assume that the engine dynamics, specifically the delay and time constant, are only affected by engine speed. The effects of air flow are ignored. This omission causes poor system performance in some regions of the engine's operating range and voids any guarantee of stability that the method provides.

1.3.2 Controller Design Theory

Supporting theory for the development of the current research is presented in the following subsections. The first two topics, H_∞, and adaptive control, are explored for comparative purposes. The last topic, LPV-based control, is the emphasis of the current research and is applied to the air-fuel ratio control problem in a novel way. An overview of gain-scheduling approaches to non-linear control is presented in [21] including methods of linearization and LPV formulations.

H_∞ Control

The H_∞ controller design method used in Section 3.3 is developed and presented in [13] and benefits from existing Linear Matrix Inequality (LMI) techniques. The controller design problem is formulated in terms of LMIs which yield a convex optimization problem and can be efficiently solved using existing LMI techniques. An alternative but similar method is given in [33] and is capable of addressing multi-objective H_2 and H_∞ problems.

Adaptive Control

An adaptive method using a series of orthonormal functions, known as a Laguerre network, to model the plant dynamics is presented in [43]. The Laguerre network is a set of time-invariant or time-varying delay elements placed in series. The output of the network is given as a vector of coefficients, called the spectral gain, multiplied by the state vector of the network. The overall transfer function of the network is very similar to a Padé approximation, making it a very suitable

model structure for plants containing a pure delay. The value of the spectral gain vector is estimated using a recursive least squares algorithm. The method is revisited in [46] to improve tracking performance and provide necessary and sufficient conditions for stability and performance.

LPV-Based Control

The LPV-based gain-scheduling techniques used in this research are first developed in [6]. The plant is written in an LPV form where the plant state-space matrices are assumed to depend affinely on a vector, θ, called the scheduling parameter. Methods for expressing non-linear plants as LPV systems are given in [18] and [30]. The scheduling parameter is assumed to be available in real-time and is used to update the controller. The resulting controller is time-varying and automatically scheduled within the bounds of θ, known as the operating space. The controller design problem is formulated in terms of LMIs, yielding a convex optimization problem. Given that the plant state-space matrices can be accurately modeled as affine functions of the scheduling parameter, the method provides a guarantee of H_∞ performance for any operating point and for any movement of the scheduling parameter.

In [5] the method is developed further and the requirement that the state-space matrices of the plant vary affinely with the scheduling parameter is removed. This is done by sampling the operating space using a grid consisting of a finite number of points. LMIs are used to define constraints at each point and are computed together in order to calculate the optimal parameter-dependent controller matri-

ces. A function is introduced so that it mimics the dependence of the Lyapunov function, and consequently of the controller matrices, on the scheduling parameter. This function then allows the assembly of the controller matrices at any point within the operating space, including but not limited to the sampled points on the grid. Furthermore, the controller design technique yields a less conservative controller because, unlike [6], a limit can be imposed on the rate of movement of the scheduling parameter within the operating range.

The method is advanced further in [23], where switching is introduced. The operating space is divided into overlapping subregions, each subregion having a controller developed using the the techniques in [5]. The overlap in subregions is used to create hysteresis switching. Additional LMIs are introduced for points along the switching surfaces and ensure that the value of the Lyapunov function is decreasing at each switching event. Because each subregion can have its own parameter-dependent Lyapunov function, the resulting switching controllers can be much more aggressive than a single controller with a single parameter-varying Lyapunov function that covers the entire operating range.

1.4 Research Objectives

Multiple authors have proposed excellent methods for addressing the air-fuel ratio control problem in SI IC engines, as discussed in the previous section. However, because authors have focussed on specific aspects of the air-fuel ratio problem, currently no single controller design technique exists which does:

- Address the time-varying and non-linear nature of the SI IC engine by ap-

propriately compensating for the engine's operating point.

- Appropriately address the parameter-dependent pure delay in the plant dynamics.
- Incorporate the design of a feed-forward controller or remove the need to design it separately from the feedback controller.
- Guarantee stability and performance over the entire operating range of the engine.

The overall objective of the current research is to apply advanced controller design techniques to the air-fuel ratio control problem and thereby address the above shortcomings with a single air-fuel ratio controller design technique. The LPV-based controller design technique developed in [5] and [23] is used. A switching LPV air-fuel ratio controller which is scheduled for both engine speed and air flow is designed and solved using LMI techniques. The proposed air-fuel ratio controller design technique will:

- Use a model of the SI IC engine which is realistic but simple enough for controller design purposes.
- Appropriately model the pure delay present in the engine dynamics.
- Treat non-linear engine dynamics as linear but parameter-dependent dynamics.
- Compensate for variations in the engine dynamics due to both air flow and engine speed.

- Remove the need for a separately designed feed-forward controller and instead use only the output of an air flow estimator.
- Track a reference air-fuel ratio and reject disturbances.
- Guarantee performance over the operating range of the engine using Lyapunov theory.
- Reduce the conservativeness of the resultant controller as compared to existing methods.
- Be practically useful, meaning that all assumptions are valid for a real SI IC engine and only available or measurable information is used by the controller.

Simulations are performed using a model of an SI IC engine to demonstrate the performance of the controller designed using the proposed technique.

1.5 Organization of Thesis

This thesis is organized as follows. Chapter 2 presents the SI IC engine plant model in two parts. First a detailed model of the plant is developed. The model serves to demonstrate its parameter-dependent and non-linear nature. A simple model, known as an FOPDT model, is then derived so that it can be used for controller design.

In Chapter 3, the proposed switching LPV controller design technique is introduced. Two current air-fuel ratio controller design techniques, H_∞, and LPV, are

also presented as well as a novel adaptive air-fuel ratio controller.

The controllers are simulated in closed loop and the results are shown in Chapter 4. Multiple controllers are developed using each technique in Chapter 3, resulting in a large number of controllers. Chapter 4 is therefore divided into three sections and the results are compared in meaningfully selected groups rather than simultaneously.

Conclusions drawn from the results are presented in Chapter 5 and the contributions of the current research are discussed. Outstanding and possible future developments are also briefly discussed.

Controller design techniques are deferred to the Appendices. The switching LPV controller design technique and the adaptive control algorithm are given in Appendix A and Appendix B, respectively.

Chapter 2

Modeling

2.1 A Measure of Air-Fuel Ratio

A measurement of the air-fuel ratio is commonly given by the variable λ, which is defined as

$$\lambda := \frac{1}{R_{stoich}} \cdot \frac{\dot{m}_{air}}{\dot{m}_{fuel}}, \tag{2.1}$$

where $\dot{m}_{fuel}$ and $\dot{m}_{air}$ are the fuel and air mass flows, respectively, and R_{stoich} is the stoichiometric ratio which is approximately 14.7 parts air to one part fuel by mass. Equation (2.1) is referred to as the *combustion equation* in this thesis.

When $\lambda < 1$ the air-fuel mixture is said to be *rich*, meaning that insufficient oxygen was supplied for complete combustion to occur, thereby causing uncombusted fuel to remain in the exhaust gas. Conversely, when $\lambda > 1$ the mixture is *lean*, meaning that insufficient fuel was supplied and that oxygen remains in the exhaust gas.

2.2 Mean Value Modeling

Although the four-stroke IC engine has discrete events making it suitable for discrete time modeling, it is common practice to model it in continuous time using Mean Value Modeling (MVM) [15]. Transient responses within each event are ignored by MVM, while the mean value over each event is captured well. This is sufficient because the period of each discrete event is much smaller than the time scale of interest. MVM is an especially good fit for four-stroke engines with four cylinders or more because of the mechanical design of these engines. The cycles of the cylinders are shifted in phase such that, at any given time, at least one cylinder is performing each of the four strokes: induction, compression, power, and exhaust. Because the engine is performing each of the four strokes at any given time, it can be thought of as a continuous process where the strokes are occurring concurrently and continuously. More importantly, air and fuel are continuously entering the engine and exhaust gas is continuously produced. The delay introduced by the engine will be discussed in Sections 2.3.3.

2.3 Detailed Model of Plant

A detailed model of the air-fuel ratio dynamics in a normally aspirated SI IC engine is developed in this section. The air flow dynamics, fuel flow dynamics, exhaust mixing, system delays, and sensor dynamics are considered. The purpose of developing this model is to achieve a complete understanding of the plant dynamics and to demonstrate the non-linear and parameter-dependent nature of the engine as well as the source of delays. The model is not used for controller design

or closed-loop simulations. Instead, a simplified but widely accepted model is presented in Section 2.4.

2.3.1 Air Flow Dynamics

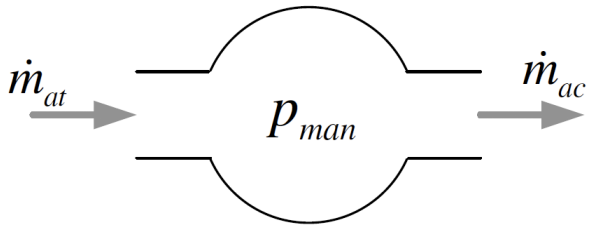


Figure 2.1: Intake Manifold Model

Due to the compressibility of the air in the intake manifold, the air flow through the throttle valve is not necessarily the same as the air flowing into the cylinders, as depicted in Figure 2.1. Assuming the air flow is isentropic, $\dot{m}_{at}$, the air flow through the throttle valve, which is measured by the MAF sensor, is given in [4] as

$$\dot{m}_{at} = \frac{p_a}{\sqrt{RT_a}} Q(\alpha)\Psi(\frac{p_{man}}{p_a}), \tag{2.2}$$

where p_a is the ambient air pressure, p_{man} is the pressure inside the intake manifold and is measured by the MAP sensor, R is the ideal gas constant, T_a is the ambient air temperature, and α is the angle of the throttle plate and determines the opening through the throttle valve. The functions $Q(\alpha)$ and $\Psi(\frac{p_{man}}{p_a})$ are nonlinear but static functions which limit the air flow into the intake manifold and are experimentally determined.

The air flow entering the cylinders is

$$\dot{m}_{ac} = \frac{\eta_{vol}(N, p_{man}) p_{man} N V_d}{n_r R T_{man}}, \tag{2.3}$$

where η_{vol} is the volumetric efficiency and is a function of engine speed, N, and manifold pressure, p_{man}. The cylinder's volumetric displacement is denoted V_d and the number of crankshaft rotations per cycle is denoted by n_r. For a four-stroke engine, $n_r = 2$. If a measurement of the manifold temperature, T_{man}, is not available, it can reasonably be assumed to be the same as the ambient temperature, $T_{man} \approx T_a$.

The rate of change of the pressure in the manifold is given by

$$\frac{dp_{man}}{dt} = K \cdot (\dot{m}_{at} - \dot{m}_{ac}), \tag{2.4}$$

where the constant, K, is related to the size of the intake manifold.

2.3.2 Wall Wetting Dynamics

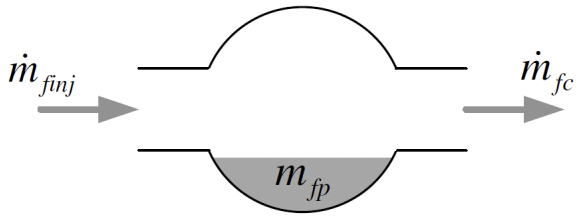


Figure 2.2: Wall Wetting Model

Fuel is injected in the intake manifold, upstream of the intake valve, in liquid

form but is partially vaporized. Of the fuel injected, $\dot{m}_{finj}$, a fraction is deposited on the manifold walls as fuel puddles while the remaining fuel immediately enters the cylinders [4]. Fuel in the puddles evaporates and also enters the cylinders. Let x_{fp} denote the fraction of fuel injected that is deposited on the manifold walls. Then $1 - x_{fp}$ of the fuel injected goes directly to the cylinders. The rate of fuel evaporating is proportional to the mass of fuel in the puddles, m_{fp}, and is dictated by τ_{fp}, the evaporation time constant. The rate of change of the mass of fuel in the puddles is given by

$$\frac{dm_{fp}}{dt} = x_{fp}\dot{m}_{finj} - \frac{1}{\tau_{fp}} m_{fp}, \tag{2.5}$$

and the fuel flow into the cylinders is

$$\dot{m}_{fc} = (1 - x_{fp})\dot{m}_{finj} + \frac{1}{\tau_{fp}} m_{fp}. \tag{2.6}$$

2.3.3 Four-Stroke Engine Cycle Delays

As stated in Section 2.2, the IC engine can be modeled as a continuous system using MVM, knowing that at any given time it is concurrently receiving fuel and air and expelling exhaust gas. It is important to consider, however, that the air and fuel mixture entering the engine at a given moment is not the same as that leaving the engine because the mixture remains in the engine while a cycle is completed. This dwell time, which is a function of the engine speed, can be accurately modeled as a pure delay, also known as dead time.

The time that fuel resides in the engine, T_{fuel}, given in seconds, is

$$T_{fuel} = \frac{60 \cdot n_r \, n_{inj-ex}}{n_{stroke} \, N}, \tag{2.7}$$

where n_{inj-ex} denotes the number of strokes completed between the rising edge of the fuel injection pulse and the start of the exhaust stroke. For a modern four-stroke engine, $n_r = 2$ and denotes the number of crankshaft revolutions per cycle, while $n_{stroke} = 4$ and denotes the number of stokes per complete cycle. The engine speed, N, is measured in revolutions per minute.

The dwell time for air and fuel are not necessarily equal. The air dwell time is given by

$$T_{air} = \frac{60 \cdot n_r \, n_{ind-ex}}{n_{stroke} \, N}, \tag{2.8}$$

where n_{ind-ex}, the number of strokes from induction to exhaust, is always equal to three for a four-stroke engine. The values of n_{inj-ex} and n_{ind-ex} are typically not equal because most port-injected IC engines inject fuel into the intake port one or even up to three strokes prior to the intake valve opening during the induction stroke.

Figure 2.3 serves as an example and shows the timing for a 2.0L 1992 Pontiac Sunbird engine, which is a four-cylinder SI IC engine, over two complete cycles. The figure displays four complete revolutions of the crankshaft, meaning two complete cycles. Stroke names, intake and exhaust valve positions, spark ignition, and fuel injection pulses are shown. This engine's ECU injects fuel in two equal pulses for each cycle. The decision of how much fuel to give is made before

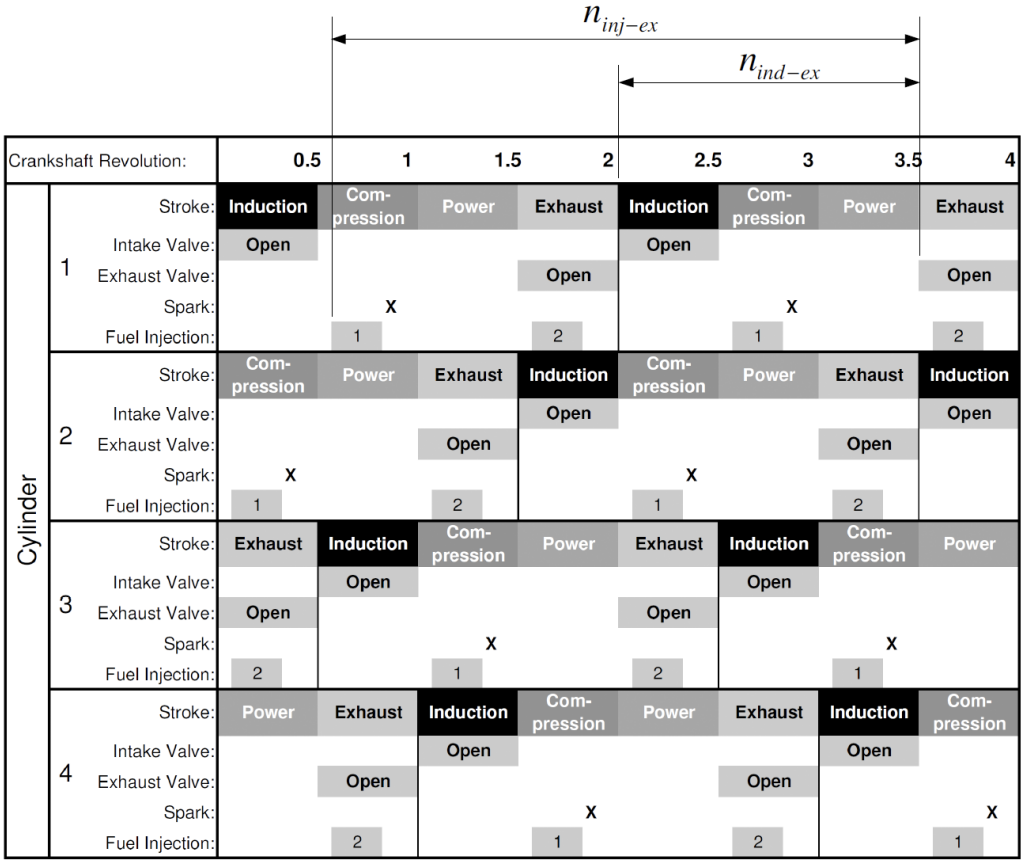


Figure 2.3: Timing of a Four-Cylinder Four-Stroke Engine

the first pulse. The first pulse occurs three strokes before the intake valve opens during the induction stroke. Fuel therefore remains in the engine for six strokes, from injection to exhaust, and therefore $n_{inj-ex} = 6$ as labeled for the first cylinder in the figure.

2.3.4 Exhaust Mixing

Individual flows from the cylinders are recombined in the exhaust manifold. Because the cycles of the cylinders in an IC engine are shifted in phase, the cylinders expel their contents into the exhaust manifold sequentially. A step change to the air-fuel ratio entering the engine will therefore result in a staircase-shaped output consisting of n_{cyl} smaller steps where n_{cyl} is the number of cylinders in the engine [20]. As an example, the output of a four-cylinder engine is shown in Figure 2.4. Each step corresponds to one cylinder completing the exhaust stroke and expelling its gasses. The duration of each step, T_{s-s}, represents to the time passing from one cylinder completing the exhaust stroke until the next cylinder does so, and is related to the engine's speed as follows:

$$T_{s-s} = \frac{60 \cdot n_r}{N \, n_{cyl}} \tag{2.9}$$

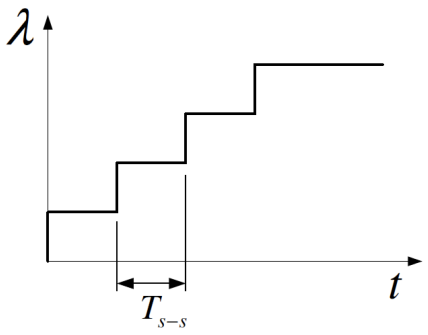


Figure 2.4: Response of a Four-Cylinder Engine to a Step Input

Recall that N is the engine speed in revolutions per minute and n_r denotes the number of revolutions per complete engine cycle.

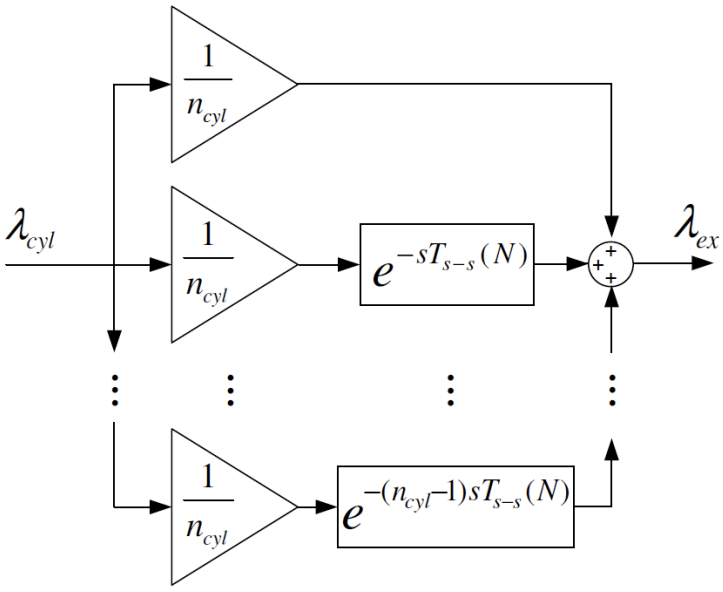


Figure 2.5: Model of Exhaust Mixing

Figure 2.5 shows a block diagram of a model for the exhaust mixing. The air-fuel ratio of the mixture inside the engine is denoted by λ_{cyl} and the air fuel ratio in the exhaust manifold is denoted by λ_{ex}. The staircase-shaped output is achieved using delays of duration kT_{s-s} placed in parallel, where $k = [0, 1, \ldots, n_{cyl} - 1]$.

2.3.5 Exhaust Transport Delay

Another delay, T_{ex}, arises from the time it takes the exhaust gas to travel from the exhaust manifold to the UEGO sensor mounted upstream of the TWC. Intuitively, T_{ex} is inversely proportional to the exhaust gas volumetric flow rate. Because the

exhaust gas volumetric flow rate is proportional to the air mass flow rate for a constant engine temperature, the exhaust gas transport delay can be expressed as a function of air mass flow rate,

$$T_{ex} = \frac{c}{\dot{m}_{air}}. \tag{2.10}$$

The proportionality constant, c, in (2.10) is chosen such that T_{ex} varies between the values of $20msec$ and $500msec$ over the engine's full operating range as recommended in [20].

2.3.6 Sensor Dynamics

Sensor dynamics are modeled as first order transfer functions (in [4], [11], [12], etc.) such that

$$\tilde{y} = \frac{1}{s\tau_y + 1}\, y, \tag{2.11}$$

where y is the measurand, i.e. the measured quantity, $\tilde{y}$ is the measurement, i.e. the quantity reported by the sensor, and τ_y is the time constant of the sensor. Typical time constants for the sensors used in the air-fuel ratio control problem are given in Table 2.1.

Table 2.1: Sensor Time Constant Values

Sensor Name	Time Constant
MAF	$10msec - 60msec$ [17]
MAP	$3msec - 20msec$ [17]
UEGO	$50msec$ [32]

2.3.7 Complete Model

Figure 2.6 shows the completed model of the air-fuel ratio dynamics discussed throughout Section 2.3. The inputs to the model are the throttle angle, α, and fuel injection rate, $\dot{m}_{finj}$, and the outputs include the MAF and MAP sensor readings, $\tilde{\dot{m}}_{at}$ and $\tilde{p}_{man}$ respectively, and the measured air-fuel ratio, $\tilde{\lambda}_{sensor}$. The model

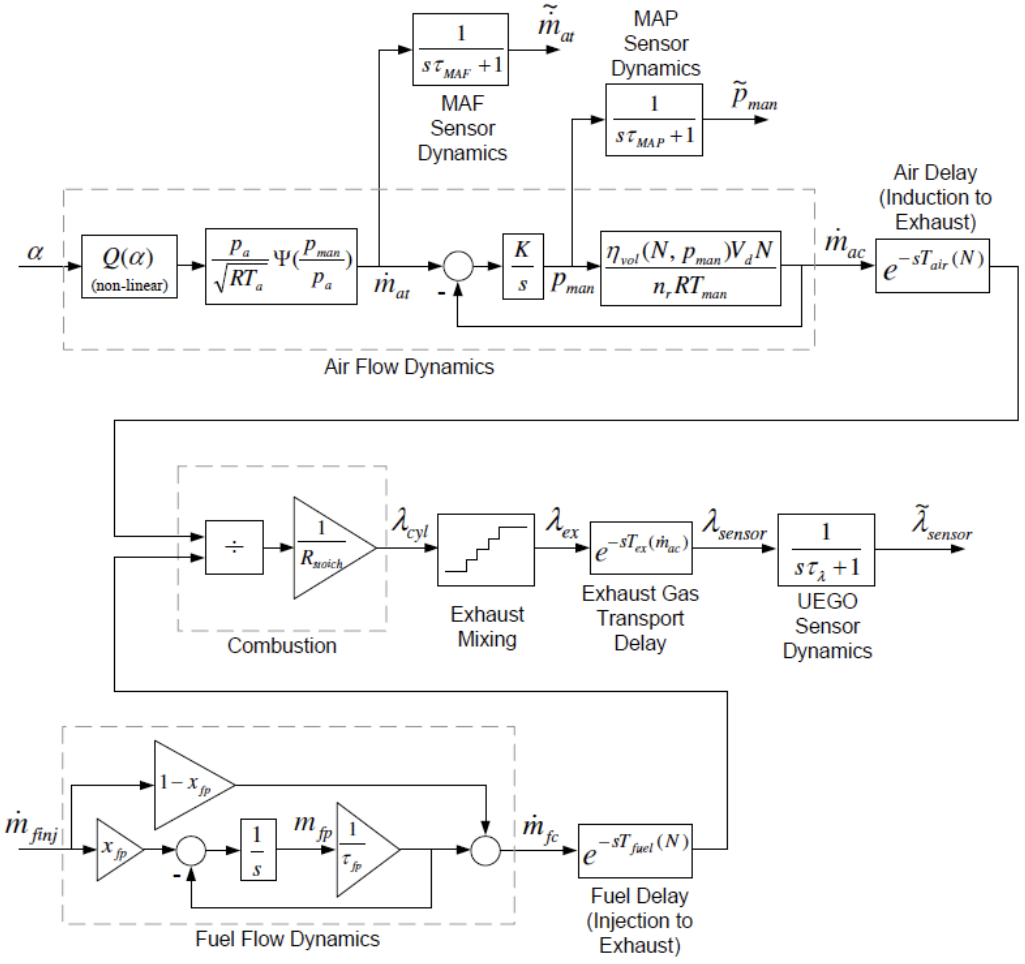


Figure 2.6: Complete Detailed SI IC Engine Model

includes the air flow dynamics, given in (2.2), (2.3), and (2.4), which relate the throttle angle to the airflow into the cylinders, $\dot{m}_{ac}$. Wall wetting dynamics defined by (2.5) and (2.6), relating the fuel flow injected, $\dot{m}_{finj}$, to the fuel flow into the cylinders, $\dot{m}_{fc}$, are also included. The air and fuel dwell times inside the cylinders, presented in Section 2.3.3, are represented by $e^{-sT_{air}}$ and $e^{-sT_{fuel}}$, respectively. The air-fuel ratio in the cylinders, λ_{cyl}, is a function of the air and fuel flows entering the cylinders according to the combustion equation, (2.1). The exhaust mixing, which relates the air-fuel ratio in the cylinders to the air-fuel ratio of the combined exhaust flow in the exhaust manifold, λ_{ex}, is shown in Figure 2.5. The exhaust gas transport delay, discussed in Section 2.3.5, is represented by $e^{-sT_{ex}}$. Finally, the first order sensor dynamics for the MAF, MAP, and UEGO sensors are also included.

2.4 Reduction to First Order Plus Dead Time

The overall dynamics of the plant described in Section 2.3 include manifold air flow dynamics, wall wetting effects, combustion cycle and exhaust gas transport delays, exhaust gas mixing, as well as sensor dynamics. However, the model describing these dynamics precisely is non-linear and too complex for controller design. The problem is aggravated by the fact that this model contains many parameter-dependent values. Constants and parameter-dependent values are experimentally identified and are unique to each engine. The identification of engine parameters is beyond the scope of the current research. The detailed model is therefore not used for controller design or closed-loop simulations.

A model which accurately captures the overall dynamics of the plant, as well as its dependence on the operating point, but is simple enough for controller design purposes is therefore necessary. A popular option is to model the overal dynamics of the IC engine as an FOPDT model. A controller which was designed for an FOPDT plant has been successfully implemented on an actual IC engine in [26]. The FOPDT model is defined as

$$\frac{g}{s\tau+1}\, e^{-sT}, \tag{2.12}$$

which includes the following three time-varying parameters:

g: The steady state gain of the system.

τ: The time constant of the first order component.

T: The duration of the pure delay.

This section uses the notation $\dot{m}_{air}$ and $\dot{m}_{fuel}$ to denote air flow and fuel flow, respectively. Note that $\dot{m}_{air}$ refers to the air flow into the engine's cylinders,

$$\dot{m}_{air} = \dot{m}_{ac}. \tag{2.13}$$

Methods have been developed to measure or estimate the air flow into the cylinders of normally aspirated IC engines based on MAF or MAP sensor readings and air flow dynamics in e.g. [9], [14], [34], [35], and [36]. Likewise, $\dot{m}_{fuel}$ refers to the fuel flow into the cylinders. Since the FOPDT model omits wall wetting

dynamics,

$$\dot{m}_{fuel} = \dot{m}_{finj} = \dot{m}_{fc}. \tag{2.14}$$

2.4.1 Steady State Gain

The air-fuel ratio, λ, is given as a non-linear function of air flow and fuel flow by the combustion equation (2.1). Notice that the uncontrolled input, air flow, enters (2.1) linearly while the controlled input, fuel flow, appears in the denominator. This non-linear relationship is udesirable for control purposes. A simple and commonly used remedy is to define a new measure of air-fuel ratio, the *equivalence ratio*, as $\phi = \frac{1}{\lambda}$. Now only the controlled input enters linearly:

$$\phi := R_{stoich} \cdot \frac{\dot{m}_{fuel}}{\dot{m}_{air}}. \tag{2.15}$$

When $\phi > 1$ the air-fuel mixture is said to be *rich* and when $\phi < 1$ the mixture is *lean*.

By considering $\dot{m}_{air}$ as a time-varying parameter, rather than an input, the plant is expressed as a single-input-single-output system that can be characterized by a pseudo-linear but parameter-dependent transfer function. Equation (2.15) then represents the steady state gain of the transfer function of the plant as it relates the plant's output, ϕ, to its input, $\dot{m}_{fuel}$, but ignores any dynamics or transient responses of the engine. It can be rewritten as

$$g = \frac{\phi}{\dot{m}_{fuel}} = \frac{R_{stoich}}{\dot{m}_{air}}. \tag{2.16}$$

Equation (2.16) shows the steady state gain of the FOPDT model as a function of the air flow.

2.4.2 Time Constant

In Section 2.3.4, it was shown that a step in the air-fuel ratio inside the engine results in a staircase-shaped response of the air-fuel ratio in the exhaust manifold due to the fact that the cylinders sequentially expel their contents. The staircase-shaped output can be approximated as a first order response. The time constant is given in [20] as

$$\tau = \frac{60 \cdot n_r(n_{cyl} - 1)}{N \cdot n_{cyl}}. \tag{2.17}$$

Recall that n_r denotes the number of revolutions the crankshaft makes to complete a cycle, n_{cyl} denotes the number of cylinders in the engine, and N is the engine speed in revolutions per minute. A measurement of the engine speed is readily available on modern engines. The time constant's dependence on the engine speed was confirmed using experimental data in [26].

2.4.3 Pure Delay

Because fuel flow is chosen as the input to the plant in Section 2.4.1, we are interested in the delay in the fuel path. The total delay from fuel injection to the air-fuel ratio measurement, T, is therefore the summation of the time that fuel remains in the engine and the travel time to the sensor:

$$\begin{aligned} T &= T_{fuel} + T_{ex}, \\ &= \frac{60 \cdot n_r \, n_{inj-ex}}{n_{stroke} \, N} + \frac{c}{\dot{m}_{air}}, \end{aligned} \tag{2.18}$$

where T_{fuel} is given in (2.7) and T_{ex} is given in (2.10). Notice that the total delay depends on both the engine speed and air flow.

2.5 Rational Plant Model

In order to perform controller design using the method developed in [23], the FOPDT plant model which is presented in Section 2.4 needs to be expressed as a rational transfer function. The pure delay, e^{-sT}, can either be removed from the transfer function and be reintroduced using additional states in the state-space realization [48] or it can be approximated using a Padé approximation. The latter is used in this research. A modified Padé approximation with a second order denominator but first order numerator is employed because, as shown in [38], it exhibits better approximation. The plant transfer function is then given by

$$\frac{\phi}{\dot{m}_{fuel}} = \frac{g}{s\tau + 1} \cdot \frac{6 - 2sT}{6 + 4sT + (sT)^2}, \tag{2.19}$$

where g, τ, and T are time-varying and depend on the engine speed and air flow as shown in (2.16), (2.17), and (2.18), respectively. When realized into the state-

space form, the plant is given as

$$\begin{aligned} \dot{x}_p &= A_p x_p + B_p \dot{m}_{fuel}, \\ \phi &= C_p x_p, \end{aligned} \tag{2.20}$$

where x_p is the plant state vector and

$$\begin{aligned} A_p &= \begin{bmatrix} \frac{-1}{\tau} & \frac{6}{T^2} & \frac{-2}{T} \\ 0 & 0 & 1 \\ 0 & \frac{-6}{T^2} & \frac{-4}{T} \end{bmatrix}, \quad B_p = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \\ C_p &= \begin{bmatrix} \frac{g}{\tau} & 0 & 0 \end{bmatrix}. \end{aligned} \tag{2.21}$$

Chapter 3

Controller Design

In this chapter, the controller design technique is presented. Four techniques are given and include LPV control, switching LPV control, H_∞ control, and adaptive control. The LPV controller is presented first in Section 3.1 and is created in order to validate the choice of scheduling parameter and compare the results with that of an existing LPV-based gain-scheduling air-fuel ratio controller. The primary contribution of the current research, an LPV-based gain-scheduling air-fuel ratio controller which also includes switching, is then described in Section 3.2. Finally, an H_∞ and adaptive controller, both of which are designed for comparison, are explained in Sections 3.3 and 3.4, respectively.

As discussed in Chapter 1, the controller design objective is to track a reference air-fuel ratio and reject disturbances. A reference air-fuel ratio other than stoichiometric, meaning an equivalence ratio other than unity, may be desired during engine warmup or may be required by the TWC oxygen storage level controller

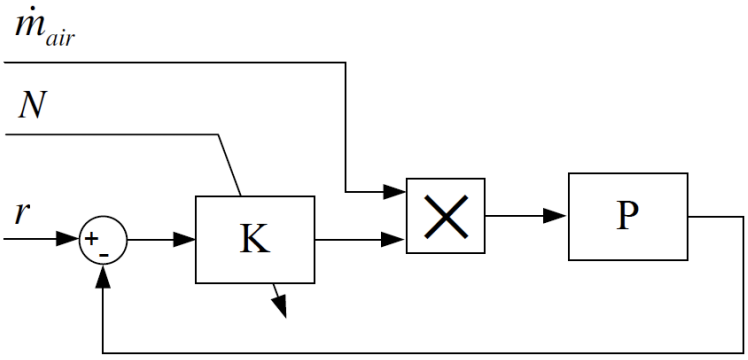
when a correction in storage level is necessary. Each controller designed therefore includes an integrator in order to improve the reference tracking performance.

The controllers are designed for the simplified model of the SI IC engine given in Section 2.4. Each controller attempts to track a reference equivalence ratio by adjusting $\dot{m}_{fuel}$, the amount of fuel injected. Accurate measurements of the engine speed, N, air flow, $\dot{m}_{air}$, and equivalence ratio, ϕ, are assumed to be available.

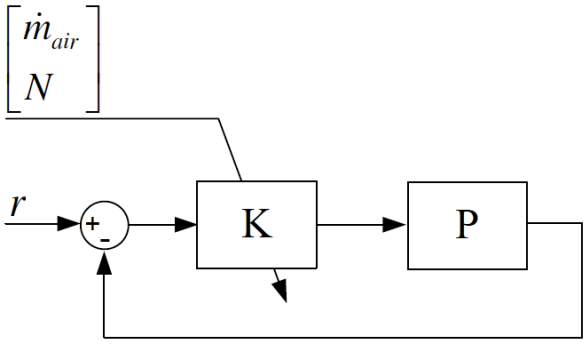
3.1 LPV Controller

An LPV-based gain-scheduled air-fuel ratio controller is designed in this section.[1] The controller design method uses LPV techniques presented in [5] and is similar to the air-fuel controller presented in [47]. The authors of [47] ignore the effect of air flow on the dynamics of the SI IC engine and present an LPV-based controller which is scheduled for engine speed only. As shown in Chapter 2, the air flow affects both the delay, T, and the steady state gain, g, of the plant, however. In order to cope with the large variations in gain due to varying air flow, the authors fix the gain in the plant transfer function of the plant model but use a feed-forward controller to calculate a base fueling rate using a measurement of the airflow. The LPV-based closed-loop controller then adjusts the fueling rate in a multiplicative manner. Figure 3.1(a) shows the structure of this method where K represents the controller and P is the time-varying plant. This explicit multiplicative controller scheduling based on air flow is not necessary for an LPV-based gain-scheduling controller which includes both air flow and engine speed in its scheduling param-

[1] The controller presented in this section has been published in [28] and presented at the 2010 ASME Dynamic Systems and Control Conference.



(a) Controller Scheduled for Engine Speed Only



(b) Controller Scheduled for Air Flow and Engine Speed

Figure 3.1: Comparison of LPV Controller Scheduling Structures

eter as shown in Figure 3.1(b).

The closed-loop system used for controller design takes the form shown in Figure 3.2. The matrices A_p, B_p, C_p, and D_p are the state-space realization of the FOPDT plant given in Chapter 2 in (2.21) where $D_p = 0$. K represents the parameter-dependent controller. The integrator placed in series with the controller improves the tracking performance. The disturbance, d, is modeled on the output

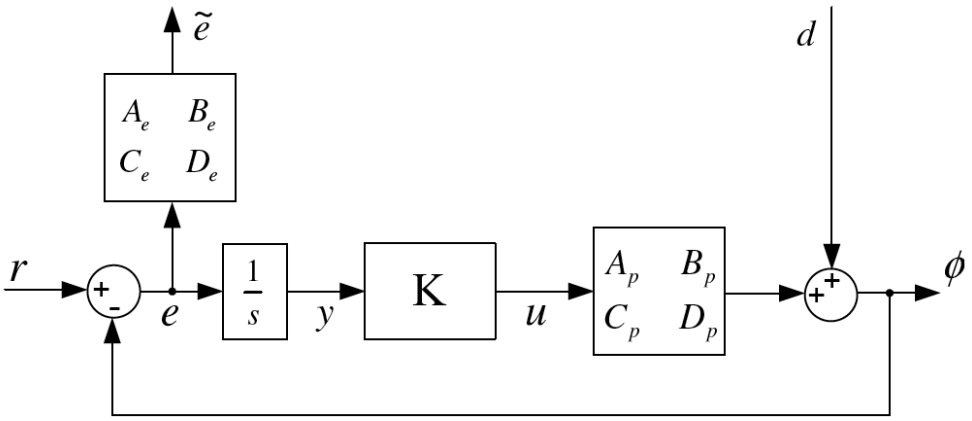


Figure 3.2: Closed-Loop System With LPV Controller

of the plant. A weighting function defined by state-space matrices A_e, B_e, C_e, and D_e filters the error and outputs a weighed error, $\tilde{e}$, which will be minimized by the controller design technique.

3.1.1 Linear Parameter Varying Form of the Plant

The plant can be written in the LPV form as

$$\begin{aligned} \dot{x} &= A(\theta)x + B_1(\theta)w + B_2(\theta)u, \\ z &= C_1(\theta)x + D_{11}(\theta)w + D_{12}(\theta)u, \\ y &= C_2(\theta)x + D_{21}(\theta)w, \end{aligned} \tag{3.1}$$

where θ is a vector known as the scheduling parameter.

The design method minimizes the L_2 norm from the exogenous input, w, to the performance channel, z. Since we wish to minimize the error due to disturbances

or reference changes, we choose

$$w = \begin{bmatrix} d \\ r \end{bmatrix} \quad and \quad z = \tilde{e}, \tag{3.2}$$

where $\tilde{e}$ is the weighed error. The weighting function placed on the error can be used to tune the controller during the design stage. The states, controller input, and controller output are

$$x = \begin{bmatrix} x_p \\ x_e \\ \int e \end{bmatrix}, \qquad y = \int e, \qquad and \;\; u, \tag{3.3}$$

respectively. For the plant shown in in Figure 3.2 expressed in LPV form, the state-space matrices are given by

$$\begin{aligned}
A &= \begin{bmatrix} A_p & 0 & 0 \\ -B_e C_p & A_e & 0 \\ -C_p & 0 & 0 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0 & 0 \\ -B_e & B_e \\ -1 & 1 \end{bmatrix}, \quad B_2 = \begin{bmatrix} B_p \\ -B_e D_p \\ -D_p \end{bmatrix}, \\
C_1 &= \begin{bmatrix} -D_e D_p & C_e & 0 \end{bmatrix}, \quad D_{11} = \begin{bmatrix} -D_e & D_e \end{bmatrix}, \quad D_{12} = \begin{bmatrix} -D_e D_p \end{bmatrix}, \\
C_2 &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}, \quad D_{21} = \begin{bmatrix} 0 & 0 \end{bmatrix}.
\end{aligned} \tag{3.4}$$

3.1.2 Scheduling Parameter

Although most SI IC engines can run at up to $6000rpm$ and wide open throttle, they spend very little time in that range. The normal use operating range of most SI IC engines is $800rpm$ to $3500rpm$ with an air flow of 10% to 50% of maximum, wide open throttle. In order to be comparable with the previous literature, which consider only the normal operating range of the engine, the controller in this section is also designed for only the normal operating range.

The proposed gain-scheduling controller adjusts for both engine speed, N, and air flow, $\dot{m}_{air}$. Noting that engine speed and air flow enter the parameters of the FOPDT plant in the denominator in (2.16), (2.17), and (2.18), we select the scheduling parameter, θ, as

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{\dot{m}_{air}} \\ \frac{1}{N} \end{bmatrix}. \tag{3.5}$$

3.1.3 LPV Controller Design

The controller design method seeks to find a controller of the form

$$\begin{aligned} \dot{x}_K &= A_K(\theta,\dot{\theta})x_K + B_K(\theta,\dot{\theta})y, \\ u &= C_K(\theta,\dot{\theta})x_K + D_K(\theta,\dot{\theta})y, \end{aligned} \tag{3.6}$$

for the LPV plant expressed in (3.1). The plant's dependence on the scheduling parameter, θ, should be mimicked by a function, denoted by $\rho(\cdot)$, which enters the

plant affinely. For the LPV plant described by (3.4) and the choice of scheduling parameter given in (3.5), $\rho(\cdot)$ can simply be selected as $\rho_1(\theta) = \theta_1$ and $\rho_2(\theta) = \theta_2$. The time-varying Lyapunov variables then also mimic the plant's dependence on θ and are given by

$$\begin{aligned} X &= X_0 + \rho_1(\theta)X_1 + \rho_2(\theta)X_2, \\ Y &= Y_0, \end{aligned} \tag{3.7}$$

which yields a practically valid solution. This implies that a measurement of $\dot{\theta}$ is not required to reconstruct the controller's state space matrices in (3.6) from the Lyapunov variables which are solved using a family of LMIs.

For comparison, the LPV-based controller presented in [47] is also developed. The controller design technique is identical to that presented above but does not include the air flow, $\dot{m}_{air}$, in the scheduling parameter, θ. Therefore, $\theta = \frac{1}{N}$, which means that $\rho(\theta) = \theta$. The Lyapunov function in (3.7) becomes

$$\begin{aligned} X &= X_0 + \rho(\theta)X_1, \\ Y &= Y_0. \end{aligned} \tag{3.8}$$

The varying steady state gain of the plant is addressed by multiplying the controller's output by the air flow as shown in Figure 3.1(a).

The LPV controllers developed in this section are referred to as LPV Controller 1 and 2 in the results section of this thesis (Chapter 4). LPV Controller 1 is scheduled for only engine speed as discussed in the above paragraph while

LPV Controller 2 is scheduled for both engine speed and air flow.

3.2 Switching LPV Controller

In this section the primary contribution of the current research, a switching LPV-based gain-scheduling controller, is presented.[2] The present method is a continuation of that used in Section 3.1 and reduces conservatism by applying the switching LPV controller design methods developed in [23] to the air-fuel ratio control problem in order to compensate for time-varying engine speed and air flow. The proposed method divides the engine's operating range into multiple subregions. Each subregion has its own parameter dependent Lyapunov function used to create an LPV controller for that subregion. The controllers are then switched in real-time so that the controller for the subregion that encompasses the current operating point is always active. By overlapping the subregions, hysteresis is introduced to the switching, thereby making it possible to guarantee switching performance by ensuring that the Lyapunov function decreases for any switching event. As well as being more aggressive, the switching LPV controller developed in this paper is able to cover the entire operating range of the engine while still providing a theoretical guarantee of performance.

The following five controllers are developed using the method presented in this section. The controllers' performance is compared in Chapter 4.

[2]The Switching LPV-based gain-scheduling controller developed in this section has been submitted for publication in the IEEE Transactions on Control Systems Technology.

S-LPV Controller 1: Single region LPV controller.

S-LPV Controller 2: Switching LPV controller with two subregions (Switched along the axis of N).

S-LPV Controller 3: Switching LPV controller with two subregions (Switched along the axis of $\dot{m}_{air}$).

S-LPV Controller 4: Switching LPV controller with four subregions.

S-LPV Controller 5: Switching LPV controller with nine subregions.

3.2.1 Linear Parameter Varying Form of Plant

Like the LPV-based controller, the switching LPV-based gain-scheduling controller design process requires the open-loop plant expressed in an LPV form as

$$\begin{aligned} \dot{x} &= A(\theta)x + B_1(\theta)w + B_2(\theta)u, \\ z &= C_1(\theta)x + D_{11}(\theta)w + D_{12}(\theta)u, \\ y &= C_2(\theta)x + D_{21}(\theta)w, \end{aligned} \tag{3.9}$$

where θ is a time-varying vector known as the scheduling parameter and x is the state vector of the LPV plant. A parameter dependent controller, also a function of the scheduling parameter, will be found such that it minimizes the worst-case L_2-gain from the input, w, to the performance channel, z, and has the form

$$\begin{aligned} \dot{x}_K &= A_K(\theta)x_K + B_K(\theta)y, \\ u &= C_K(\theta)x_K + D_K(\theta)y, \end{aligned} \tag{3.10}$$

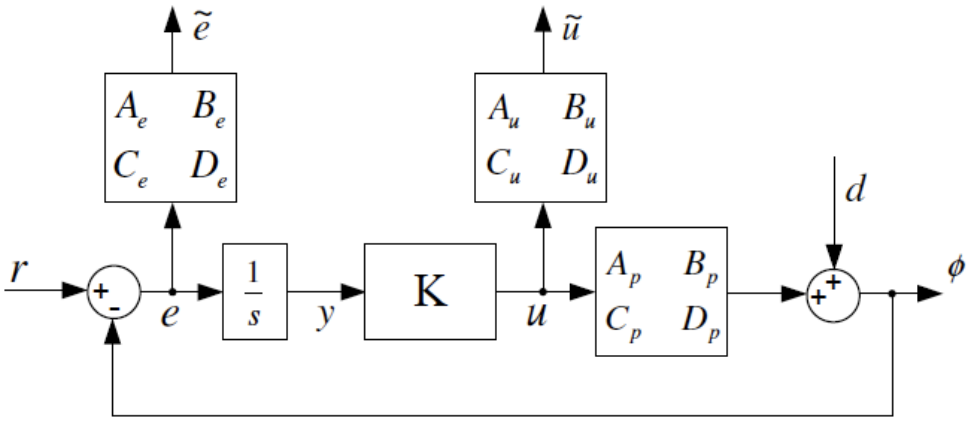


Figure 3.3: Closed-Loop System With Switching LPV Controller

with controller states x_K. The controller takes the measurement, y, and gives output, u, where y is the integral of the error as can be seen in Figure 3.3 and u is the fuel flow, $\dot{m}_{fuel}$. A disturbance, d, is modeled on the output of the plant. The exogenous input and the performance channels are

$$w = \begin{bmatrix} d \\ r \end{bmatrix} \quad \text{and} \quad z = \begin{bmatrix} \tilde{e} \\ \tilde{u} \end{bmatrix}. \tag{3.11}$$

Second order weighting functions W_e and W_u are included on the error signal and the controller output, respectively, to produce the performance channels, $\tilde{e}$ and $\tilde{u}$, and allow controller tuning. The weighting functions, W_n, where $n := \{e,u\}$, are realized using the state-space form,

$$\begin{aligned} \dot{x}_n &= A_n x_n + B_n n, \\ \tilde{n} &= C_n x_n + D_n n, \end{aligned} \tag{3.12}$$

with the state vector x_n. The weighting function parameters are chosen such that low frequency errors and high frequency controller outputs are prevented. Table 3.1 shows values for the weighting functions used in this thesis.

Table 3.1: Switching LPV Controller Design Weighting Functions

	W_e	W_u
Low Frequency Gain	10	10^3
High Frequency Gain	10^4	10^{-1}
Crossover Frequency	$5 \times 10^3 rad/sec$	$10^5 rad/sec$

The states, controller input, and controller output of the LPV system are

$$x = \begin{bmatrix} x_p \\ x_e \\ x_u \\ \int e \end{bmatrix}, \qquad y = \int e, \qquad \text{and} \qquad u, \tag{3.13}$$

respectively. The state space matrices in (3.9) are

$$
\begin{aligned}
A &= \begin{bmatrix} A_p & 0 & 0 & 0 \\ -B_e C_p & A_e & 0 & 0 \\ 0 & 0 & A_u & 0 \\ -C_p & 0 & 0 & 0 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0 & 0 \\ -B_e & B_e \\ 0 & 0 \\ -1 & 1 \end{bmatrix}, \quad B_2 = \begin{bmatrix} B_p \\ 0 \\ B_u \\ 0 \end{bmatrix}, \\
C_1 &= \begin{bmatrix} 0 & C_e & 0 & 0 \\ 0 & 0 & C_u & 0 \end{bmatrix}, \quad D_{11} = \begin{bmatrix} -D_e & D_e \\ 0 & 0 \end{bmatrix}, \quad D_{12} = \begin{bmatrix} 0 \\ D_u \end{bmatrix}, \\
C_2 &= \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}, \quad D_{21} = \begin{bmatrix} 0 & 0 \end{bmatrix}.
\end{aligned}
\tag{3.14}
$$

3.2.2 Scheduling Parameter

Although not explicitly shown in (3.14), the plant is parameter-varying due to the dependence of A_p, B_p, and C_p on the parameters g, τ and T which are, in turn, dependent on the engine speed and air flow as shown in Section 2.4. The engine speed can vary from $800rpm$ to $6000rpm$ and the air flow can vary from 10% to 100%. Air flow is normalized and given as a percentage of maximum flow in this thesis. The engine speed and air flow ranges are collectively referred to as the engine's operating range. Furthermore, the rate of change of the operating point can be up to $6000rpm$ per second for the engine speed and 100% per second for the air flow. Noting that air flow, $\dot{m}_{air}$, and engine speed, N, enter (2.16), (2.17), and

(2.18) in the denominator, we select the scheduling parameter θ as

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{\dot{m}_{air}} \\ \frac{1}{N} \end{bmatrix}. \tag{3.15}$$

Although the LPV controller, presented in the previous section, is only designed for the normal use operating range of the engine, the added performance of the switching LPV controllers allows them to remain effective over the full operating range. From the limits on the operating range of the engine, we can calculate the bounds on the scheduling parameter as well as the bounds on its derivative, denoted by Θ and Θ_d, respectively:

$$\begin{aligned} \Theta &= \left\{ (\theta_1, \theta_2) \mid \tfrac{1}{100\%} < \theta_1 < \tfrac{1}{10\%},\ \tfrac{1}{6000rpm} < \theta_2 < \tfrac{1}{800rpm} \right\}, \\ \Theta_d &= \left\{ (\dot{\theta}_1, \dot{\theta}_2) \mid \tfrac{-100\%/sec}{(10\%)^2} < \dot{\theta}_1 < \tfrac{100\%/sec}{(10\%)^2},\ \tfrac{-6000rpm/sec}{(800rpm)^2} < \dot{\theta}_2 < \tfrac{6000rpm/sec}{(800rpm)^2} \right\}. \end{aligned} \tag{3.16}$$

3.2.3 Switching LPV Controller Design

A detailed review of the LPV controller design method used is given in Appendix A. The parameter space, Θ, is divided into R overlapping subregions $\Theta^{(r)}, r = 1, ..., R$. A superscript in parenthesis refers to the subregion index in this thesis. The method uses a separate parameter dependent Lyapunov function to provide an

LPV controller for each subregion with guaranteed performance of the closed-loop plant within the limits $\Theta^{(r)}$ and Θ_d. The controllers are then switched depending on which subregion the scheduling parameter, θ, belongs to at each point in time. In order to guarantee switching performance, it must be ensured that the Lyapunov function decreases at each switching event. In other words, when the scheduling parameter moves from one subregion to another, the Lyapunov function defined within the subregion it is leaving must be greater than the Lyapunov function defined within the subregion it is entering. By increasing the overlap of the subregions, hysteresis is increased and this constraint becomes easier to meet.[3]

The dependence of the Lyapunov function on the scheduling parameter, θ, is mimicked by a set of functions, $\rho(\cdot)$, which are differentiable but not necessarily linear functions of the scheduling parameter. The function $\rho(\cdot)$ is introduced and explained in greater detail in [40]. For the plant given in (3.14), $\rho(\cdot)$ is simply

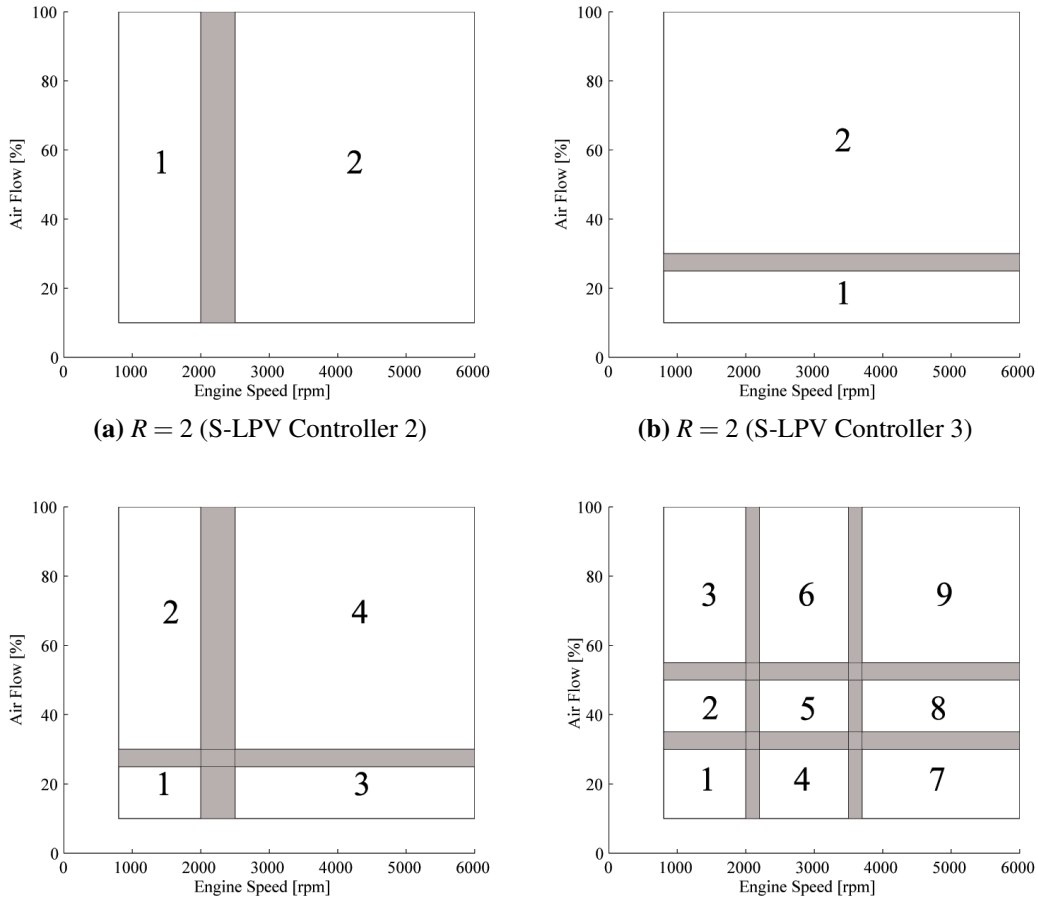
$$\rho(\cdot) = \begin{bmatrix} \rho_1(\cdot) \\ \rho_2(\cdot) \end{bmatrix} = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} \tag{3.17}$$

and is used to define the LMI variables. For example, the parameter dependent Lyapunov variables are

$$\begin{aligned} X &= X_0, \\ Y^{(r)}(\theta) &= Y_0^{(r)} + \rho_1(\theta) Y_1^{(r)} + \rho_2(\theta) Y_2^{(r)}. \end{aligned} \tag{3.18}$$

[3]Figure A.1 in Appendix A shows the use of subregion overlap to create hysteresis switching.

Note that X is a constant while $Y^{(r)}(\theta)$ is a function of the scheduling parameter. A practically valid controller (See Appendix A) can also be designed with $X^{(r)}(\theta)$ as a function of the scheduling parameter and Y held constant, but yields more conservative results in our problem setting. Because no analytical method exists to choose which Lyapunov variable to fix, both options must be tried and compared in order to find the least conservative resultant controller.



(a) $R = 2$ (S-LPV Controller 2)

(b) $R = 2$ (S-LPV Controller 3)

(c) $R = 4$ (S-LPV Controller 4)

(d) $R = 9$ (S-LPV Controller 5)

Figure 3.4: Operating Space Divided Into R Subregions

The five controllers developed using the LPV controller design method are referred to as S-LPV Controller 1 to 5 throughout this thesis. S-LPV Controller 1 uses one subregion to cover the entire operating range of the engine, $\Theta^{(1)} = \Theta$, and therefore does not actually involve any switching. For S-LPV Controller 2 and S-LPV Controller 3, the parameter space is divided into two subregions as shown in Figure 3.4(a) and Figure 3.4(b), respectively. The shaded areas denote the overlap of the subregions. Recall that the parameter space is given in θ, which is actually the inverse of engine speed and air flow as shown in (3.15). However, graphical representations of the subregions in this thesis use the axes air flow and engine speed rather than θ_1 and θ_2 because their meanings are more intuitive. S-LPV Controller 4 and S-LPV Controller 5 are developed for four subregions and nine subregions shown in Figure 3.4(c) and Figure 3.4(d), respectively.

The benefit of increasing the number of subregions is that the parameter variation within each region becomes smaller, consequently reducing the variation of the Lyapunov function and yielding a more aggressive controller for each subregion. The drawback, however, is the added memory and time requirements to solve and store the increasing number of LMI variables. Table 3.2 shows the number of LMIs and LMI variables used for each switching LPV controller as well as the time taken to solve for the LMIs yielding the controllers presented. Computations were performed on the same computer and under similar conditions so that times can be meaningfully compared.

Table 3.2: Complexity and Computing Time of Switching LPV Controllers

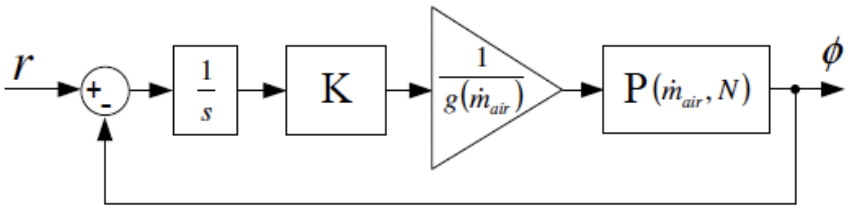
Name	No. of Subregions	No. of LMI's	LMI Variables	Time to Solve
S-LPV Controller 1	1	32	17	1*min* 12*sec*
S-LPV Controller 2	2	68	32	1*min* 38*sec*
S-LPV Controller 3	2	68	32	1*min* 38*sec*
S-LPV Controller 4	4	144	62	7*min* 0*sec*
S-LPV Controller 5	9	336	137	58*min* 37*sec*

3.3 H_∞ Controller

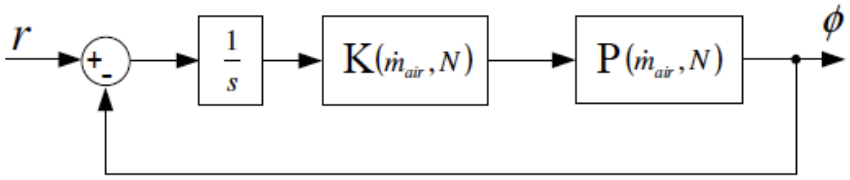
Two H_∞ controllers are developed for comparison with the LPV and switching LPV controllers using the LPV plants defined by (3.4) and (3.14), respectively, with LMI techniques described in [13]. Unlike the LPV and switching LPV controllers developed in the last section, the usual H_∞ controller design method assumes a time-invariant plant and produces a single time-invariant controller.

The H_∞ controllers in this paper are modified to compensate for variations in g, the parameter dependent steady state gain of the system, using the following method: First the values of τ and T are sampled at a nominal engine speed and air flow and the steady state gain is given the value $g = 1$. A time-invariant H_∞ controller for the plant with these values is then generated using the method in [13]. A parameter dependent gain is placed on the output of the controller. This gain is equal to the inverse of the steady state gain of the time-varying plant and is calculated in real-time. In this way, the parameter dependent gain on the output of the controller cancels out the time-varying steady state gain of the plant.

Figure 3.5(a) shows the closed-loop system with the parameter dependent



(a) Gain-Scheduled H_∞ Controller



(b) LPV or Switching LPV Controller

Figure 3.5: Comparison of H_∞ Controller Scheduling With LPV Controller

plant, time-invariant controller, and parameter dependent gain. Figure 3.5(b) shows the equivalent closed-loop system with a parameter dependent LPV or switching LPV controller for comparison. Since the gain-scheduled H_∞ controller does not adjust for variations of the time constant, τ, or pure delay, T, performance cannot be guaranteed at any operating point other than the nominal point that it is designed for.

The first H_∞ controller, developed for comparison with the LPV controllers of Section 3.1, is designed for a nominal operating point of 30% air flow and $1500rpm$ engine speed. The controller is carefully tuned such that the closed loop will remain stable over the normal operating range of 10% to 50% air flow and $800rpm$ to $3500rpm$ engine speed because this is the operating range that the LPV controllers are designed for. It is important to keep in mind that, though the H_∞

controller is tuned to work over the operating range, no guarantee of performance exists. The second H_∞ controller, which is developed for comparison with the switching LPV controllers of Section 3.2, is designed for a nominal operating point of 80% air flow and $4000rpm$ engine speed and is tuned to remain effective over the entire operating range of the engine.

3.4 Adaptive Controller

As an alternative to the gain-scheduling approach to the air-fuel ratio problem, an adaptive controller is also developed. The appeal of the adaptive approach is that no *a priori* knowledge of the parameter variations of the plant is required, meaning that the development of an accurate model of the plant is not necessary. The controller identifies the plant dynamics in real-time using a recursive least squares algorithm and provides the appropriate control input, needing only measurements of the plant's input and output. Furthermore, the adaptive controller is able to compensate not only for variations in the dynamics due to engine speed and air flow, but also for variations due to engine and sensor aging and environmental factors.

An adaptive predictive controller using a Laguerre network is chosen for this application.[4] The controller design method is introduced in [43] and has had much success in controlling pH levels in an industrial bleach plant extraction stage as well as other systems with large delays. The authors develop a model predictive controller which looks beyond the delay time in the plant. An attractive feature of

[4]The Laguerre network is discussed in greater detail in Section B.1 in the Appendix.

using a Laguerre network to model the plant is that no assumptions about the plant order or time delay are required, making it ideal for the air-fuel ratio control problem since the dynamics of the plant are dominated by a time-varying delay. The method is improved in [46] where it is given the name Incremental Mode Linear Laguerre Predictive Control (IMLLPC) and is modified by inserting an integrating action into the controller to provide better tracking performance. Necessary and sufficient conditions for stability and steady-state performance are also presented but are useful for analysis only and are not used during the design stage.

The detailed control algorithm is reviewed in Appendix B and is not presented here for brevity. Table 3.3 shows values that act as the tuning parameters and are set by the designer before creating the controller. The values were found mostly through an iterative tuning procedure.

Table 3.3: Tuning Parameters of the IMLLPC Algorithm

Symbol	Description	Value
T_s	Sampling time	$0.1sec$
p	Time-scale	60
N	Order of Laguerre network	8
d	Prediction horizon	7
M	Control horizon	1
Q	Weighting matrix on error	$I_{d\times d}$
R	Weighting matrix on control input	$I_{M\times M}$
α	Softening factor on reference model output	0.2
λ	Forgetting factor for recurser	0.96

In order to simulate the adaptive controller's performance, a discrete-time

model of the plant is required. We use a state-space model of the form

$$\begin{aligned} x[k+1] &= A_p x[k] + B_p \dot{m}_{fuel}[k-n_T], \\ \phi[k] &= C_p x[k] + D_p \dot{m}_{fuel}[k-n_T], \end{aligned} \tag{3.19}$$

where n_T is the delay time of the plant, T, expressed in number of sample periods, $n_T = \frac{T}{Ts}$, and A_p, B_p, C_p, and D_p are matrices of the discretized state-space realization of the transfer function

$$\frac{\phi}{\dot{m}_{fuel}} = \frac{g}{s\tau+1}, \tag{3.20}$$

which is simply the first order component of the FOPDT model given in (2.19) in Chapter 2. This model is then used to simulate the closed-loop of the system and iteratively tune the controller. The controller's closed-loop simulation results are shown and discussed in Chapter 4.

Chapter 4

Simulation Results

In this chapter the controllers are validated and their reference tracking and disturbance rejection performances are compared using simulations performed using MATLAB and Simulink software. The chapter is divided into three sections. In Section 4.1, the LPV controllers are compared with each other as well as an H_∞ controller in order to support the decision to consider both engine speed and air flow in the scheduling parameter. In Section 4.2, the performance improvements of the switching LPV controllers are shown. Finally, the performance of the adaptive controller is compared with that of a switching LPV controller in Section 4.3.

Each section is further divided into two subsections titled Time-Invariant Simulations and Time-Varying Simulations. In the former, the dynamic model of the plant is sampled and held constant for the duration of each simulation, consequently producing a time-invariant plant. The time-invariant simulations are used to compare the response time of the controllers at different points within the en-

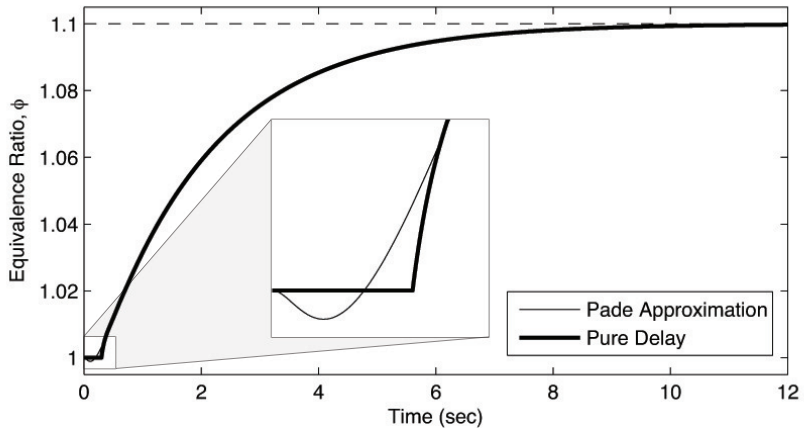


Figure 4.1: Comparison of Pure Delay and Padé Approximation

gine's operating range. During the time-varying simulations, on the other hand, the engine speed and air flow follow realistic trajectories while the controllers are performing reference tracking and disturbance rejection. The performance of the controllers is thereby validated for realistic operating conditions.

Although the plant includes a Padé approximation during controller design, a non-rational pure delay is employed during validation. Figure 4.1 compares the effect of a Padé approximation and a pure delay on the step response at an engine speed of $800rpm$, and an air flow of 50%. Both curves are acceptably similar, validating the choice of order of the Padé approximation given in Chapter 2.5. This holds true at all engine operating conditions.

Remark

In order to ensure stability of the controller, which was designed for a plant containing a Padé approximation, with a plant containing a pure delay, either of two

methods can be used.

The first method is to place a low-pass filter on the output of the controller. The bandwidth of the filter should be equal to or less than the frequency at which the phase of the Padé approximation's frequency response plot starts differing from that of the actual pure delay. Because the pure delay, T, is time-varying this frequency will vary also. Either a low-pass filter with a time-varying bandwidth or a time-invariant filter, with a bandwidth that corresponds to the worst case of the delay (which happens at $800rpm$ and 10% air flow) can be used. Addition of the filter effectively modifies the FOPDT plant's frequency response so that it is similar to that of an FOPDT approximated as a rational transfer function.

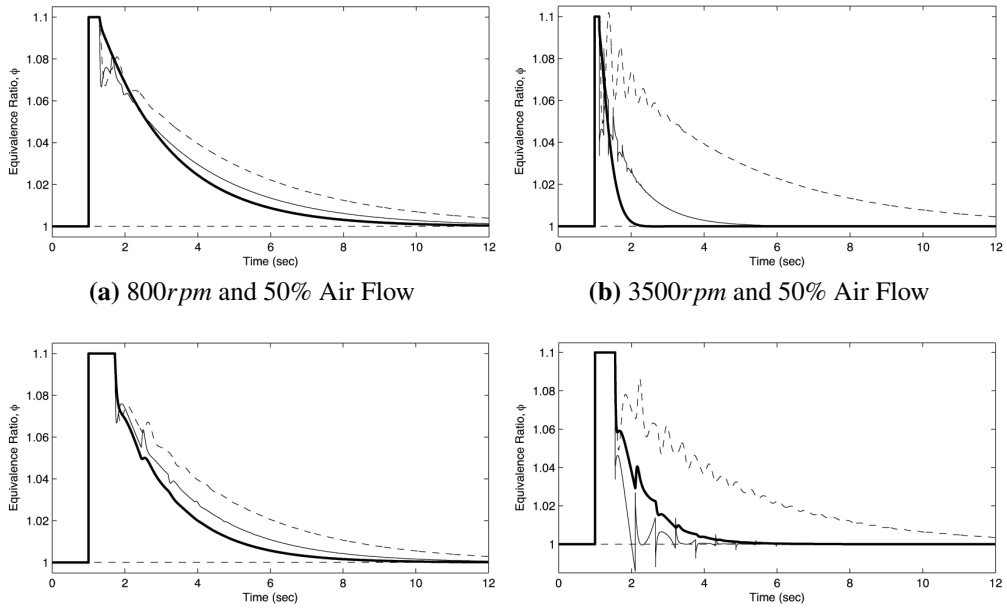
An alternative method is to use the weighting function on the controller output to tune the controller during controller design such that high frequency controller outputs are avoided. The controller should not output signals with any frequencies higher than the frequency where the phase of the Padé approximation starts differing from that of the pure delay.

Both methods have been attempted and have yielded similar results. Results shown in this chapter are generated using aggressively tuned controllers with time-invariant low pass filters on their outputs.

4.1 LPV Controllers

The H_∞ controller developed in Section 3.3 and LPV Controller 1 and LPV Controller 2, both developed in Section 3.1, are compared in this section.[1] The pur-

[1]The results presented in this section have been published in [28] and presented at the 2010 ASME Dynamic Systems and Control Conference.



(a) 800rpm and 50% Air Flow

(b) 3500rpm and 50% Air Flow

(c) 800rpm and 10% Air Flow

(d) 3500rpm and 10% Air Flow

Figure 4.2: Disturbance rejection of H_∞ Controller (Dashed), LPV Controller 1 (Thin), and LPV Controller 2 (Thick) at Fixed Operating Points

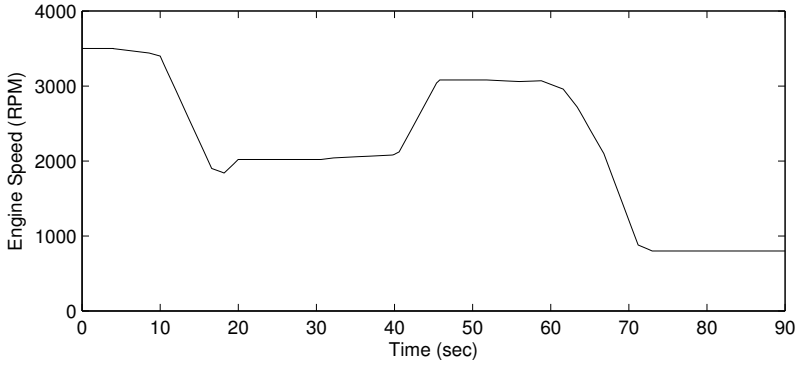
pose of comparing these controllers is to show the performance improvement achieved by scheduling the controller for both engine speed and air flow.

4.1.1 Time-Invariant Simulations

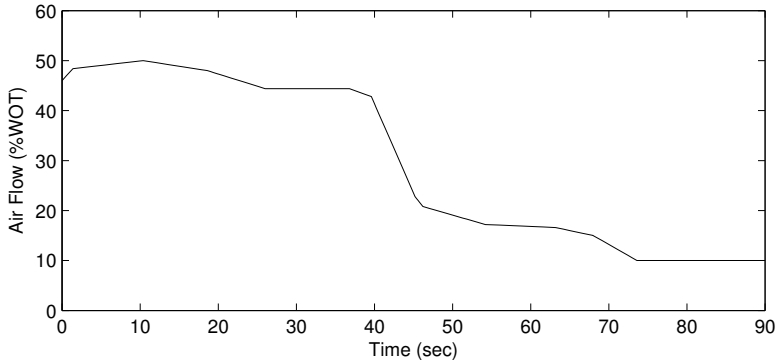
Figure 4.2 displays the system's response to a 10% step disturbance applied at time $t = 1sec$ while operating at each of the four extremes of the operating range. Figures 4.2(a) and 4.2(c) show that both LPV Controller 1 and LPV Controller 2 have comparable results with the H_∞ controller at low engine speed, when the open-loop plant is at its slowest. At high engine speeds, however, the LPV controllers adjust for the faster plant, yielding a much faster closed-loop response

than that of the fixed controller as can be seen in Figures 4.2(b) and 4.2(d). Figure 4.2(d) shows that LPV Controller 1, which does not compensate for variations in air flow, is too aggressive for the plant at 10% air flow while Figure 4.2(b) shows it to be slow for the plant at 50% air flow. LPV Controller 2, which is scheduled for both air flow and engine speed, maintains acceptable responses for both cases.

4.1.2 Time-Varying Simulations

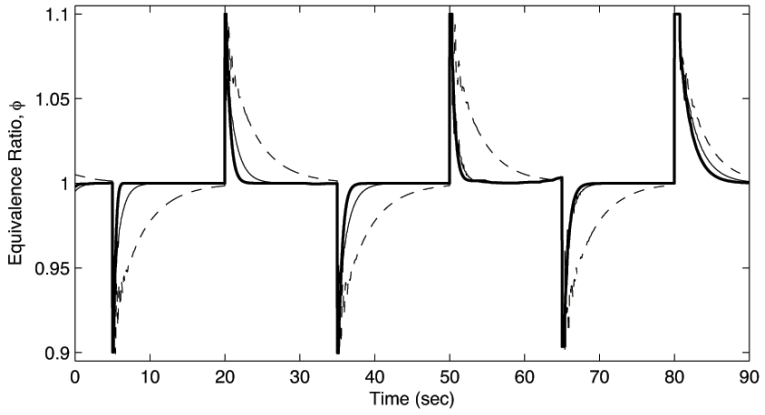


(a) Engine Speed

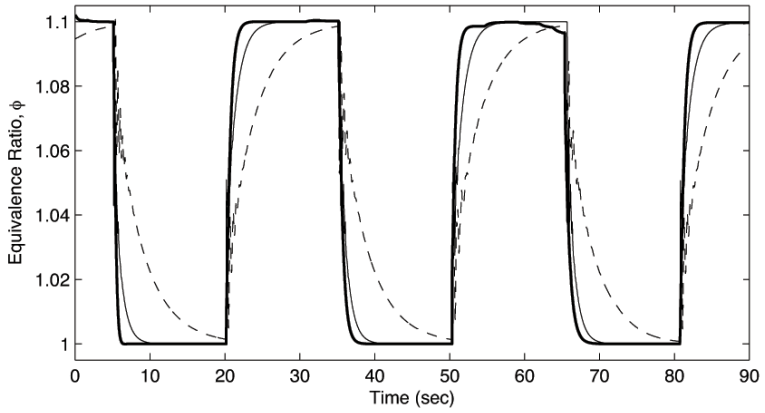


(b) Air Flow

Figure 4.3: Engine Speed and Air Flow Profiles



(a) Disturbance Rejection



(b) Reference Tracking

Figure 4.4: Performance of H_∞ Controller (Dashed), LPV Controller 1 (Thin), and LPV Controller 2 (Thick) With Time-Varying Plant Dynamics

In order to demonstrate the system's stability and performance while the engine operating point is changing, realistic profiles for engine speed and air flow are used. Figure 4.3 shows the engine speed and air flow over a 90*sec* simulation. The profiles represent normal engine usage and encompass all four extremes of

the normal use operating range. In Figure 4.4(a) a disturbance having the shape of a square wave with a period of 30*sec* and an amplitude of 10% of the reference is applied to the system. A square reference signal varying from 1 to 1.1 with a period of 30*sec* is applied to the system in Figure 4.4(b). LPV Controller 2, compensated for both air flow and engine speed, shows some improvement over LPV Controller 1, and very significant improvement over the H_∞ controller in both reference tracking and disturbance rejection performance.

4.2 Switching LPV Controllers

The switching LPV controllers developed in Section 3.2 as well as an H_∞ controller developed in Section 3.3 are evaluated in this section.[2] The purpose of comparing these controller is to show the improvement in performance which can be achieved by dividing the operating space into subregions and to demonstrate that stability is maintained even for rapidly moving operating points.

Recall that the naming convention of the switching LPV controllers is:

S-LPV Controller 1:	Single region LPV controller.
S-LPV Controller 2:	Switching LPV controller with two subregions (Switched along the axis of engine speed, N).
S-LPV Controller 3:	Switching LPV controller with two subregions (Switched along the axis of air flow, $\dot{m}_{air}$).
S-LPV Controller 4:	Switching LPV controller with four subregions.
S-LPV Controller 5:	Switching LPV controller with nine subregions.

[2]The results presented in this section have been submitted for publication in the IEEE Transactions on Control Systems Technology.

4.2.1 Time-Invariant Simulations

The controllers are tested at nine different points within the operating range of the engine in this section. Because the scheduling parameter, θ, remains fixed throughout the duration of each simulation, the parameter dependent plant is also time-invariant during each simulation. A step disturbance is given at time $t = 1sec$ and the response is plotted. The responses of the H_∞ controller and of S-LPV Controller 1 can be seen in Figure 4.5. At higher engine speed and air flow, the LPV controller takes advantage of the plant's faster open-loop response and yields improved closed-loop response while the H_∞ controller's response time remains similar throughout the operating range. Both controllers have poor and oscillatory responses at low engine speed and air flow when the plant dynamics

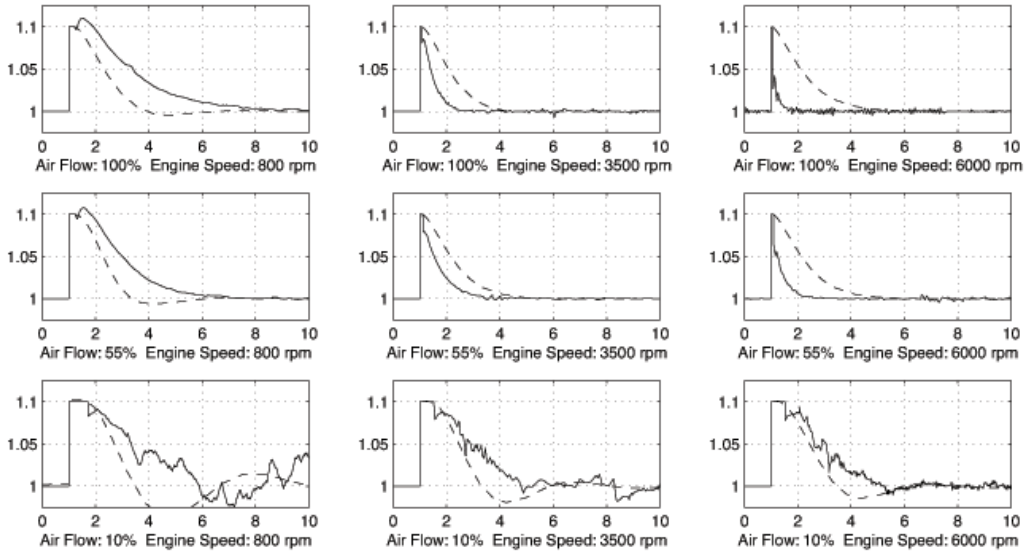


Figure 4.5: Disturbance Rejection of H_∞ Controller (Dashed) and S-LPV Controller 1 (Solid)

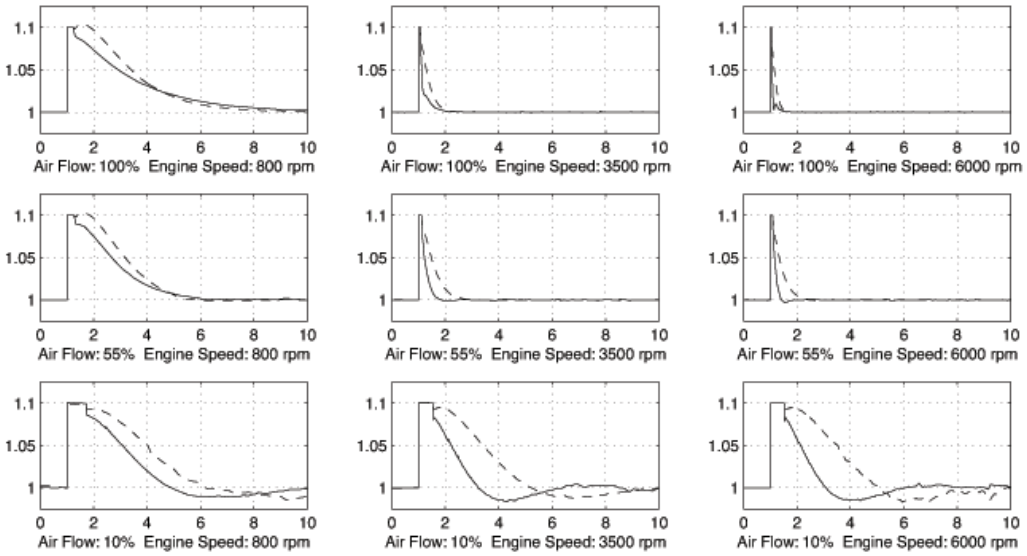


Figure 4.6: Disturbance Rejection of S-LPV Controller 2 (Dashed) and S-LPV Controller 3 (Solid)

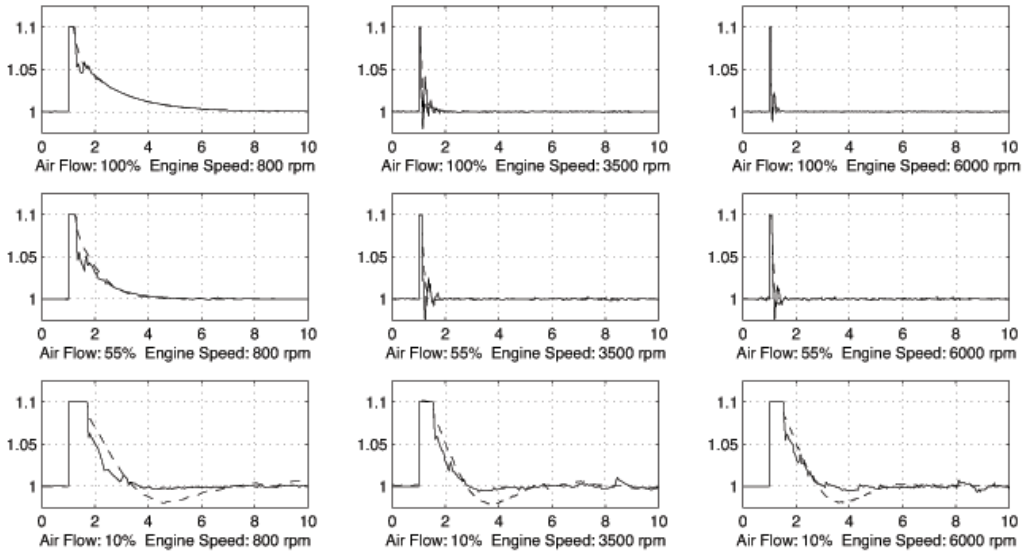


Figure 4.7: Disturbance Rejection of S-LPV Controller 4 (Dashed) and S-LPV Controller 5 (Solid)

are very slow.

Figure 4.6 shows the closed-loop responses of S-LPV Controllers 2 and 3, both having two subregions. S-LPV Controller 3 is switched along the air flow axis and yields improved disturbance rejection at all operating points. This demonstrates the importance of carefully choosing the subregions for the switching controller. Intuitively, the subregions should be chosen such that the open-loop plant's variation due to the movement of the scheduling parameter is minimum within each subregion. An analytical method for finding the optimal subregions is beyond the scope of this research, however. Recall that the subregions used for the switching LPV controllers are given in Figure 3.4 in Section 3.2 and have been chosen through trial and error.

Figure 4.7 shows the disturbance rejection of S-LPV Controllers 4 and 5, having four and nine subregions, respectively. Due to the decreased parameter variation within each subregion, these two controllers outperform the single region LPV and two subregion switching LPV controllers at all operating points. The performance improvement of S-LPV Controller 5 compared to S-LPV Controller 4 is small while it requires a considerable increase in memory and computation time as shown in Table 3.2 in Section 3.2. S-LPV Controller 4, having four subregions, is therefore recommended as the controller of choice for this particular plant.

4.2.2 Time-Varying Simulations

An engine speed and air flow profile representing realistic engine use are employed to demonstrate the system's stability and performance while the engine

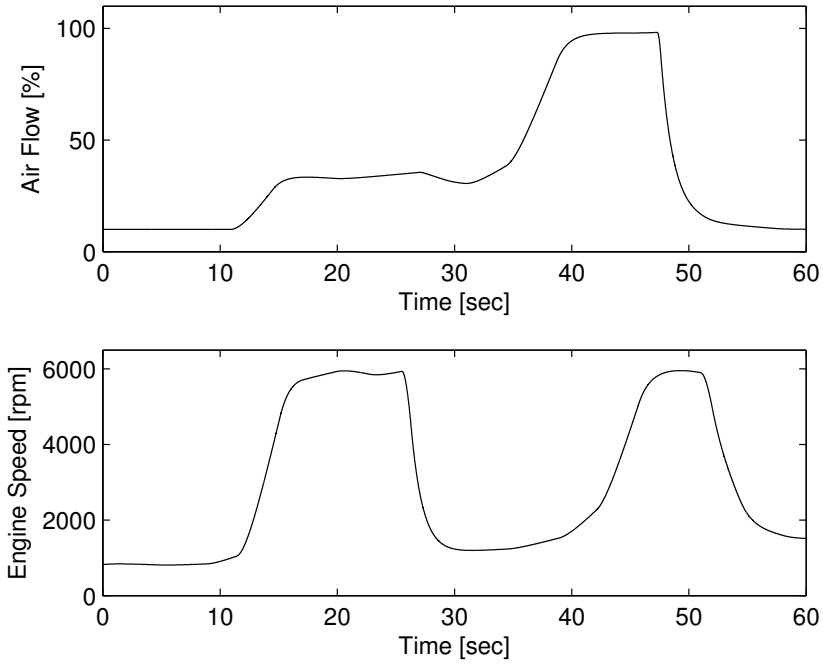


Figure 4.8: Air Flow and Engine Speed Profiles

operating point is varying. Figure 4.8 shows the engine speed and air flow over the 60*sec* simulation. Idling, "revving", high load, and engine breaking are all represented in the engine profiles. The resulting trajectory of the scheduling parameter within the operating space is depicted in Figure 4.9. The derivative of the scheduling parameter remains within in the bounds Θ_d, set in (3.16) in Section 3.2. The H_∞ controller, S-LPV Controller 1, and S-LPV Controller 4 are compared using the time-varying simulation. The switching signal generated for S-LPV Controller 4 can be seen in Figure 4.10 and confirms that the parameter trajectory passes through all four subregions.

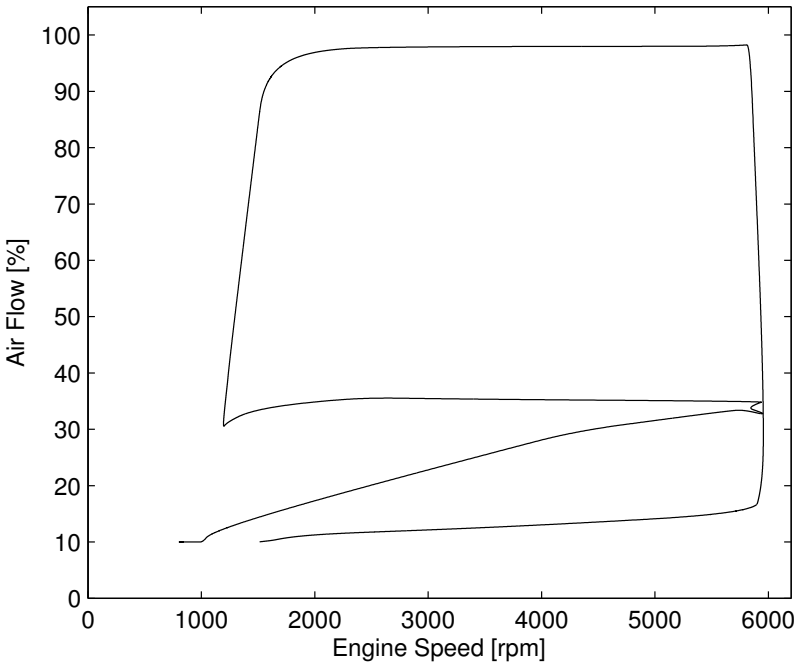


Figure 4.9: Engine Operating Point Trajectory

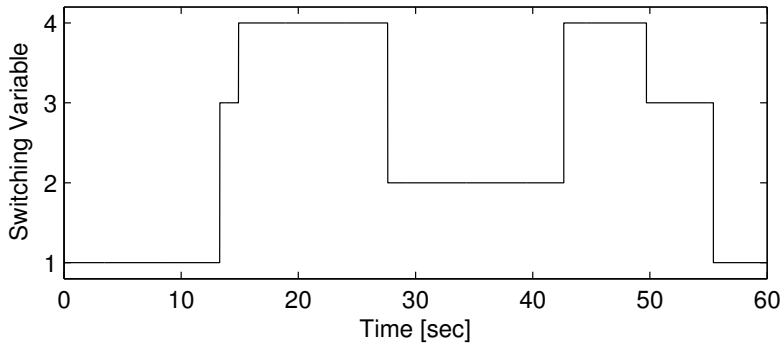
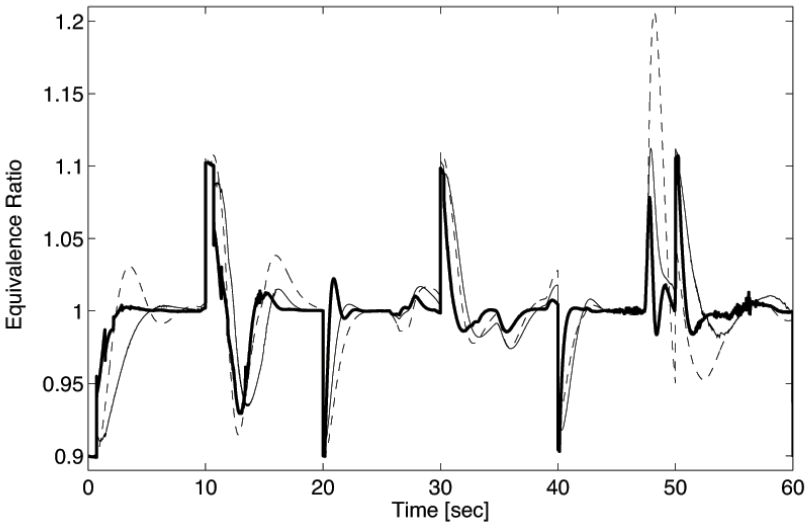
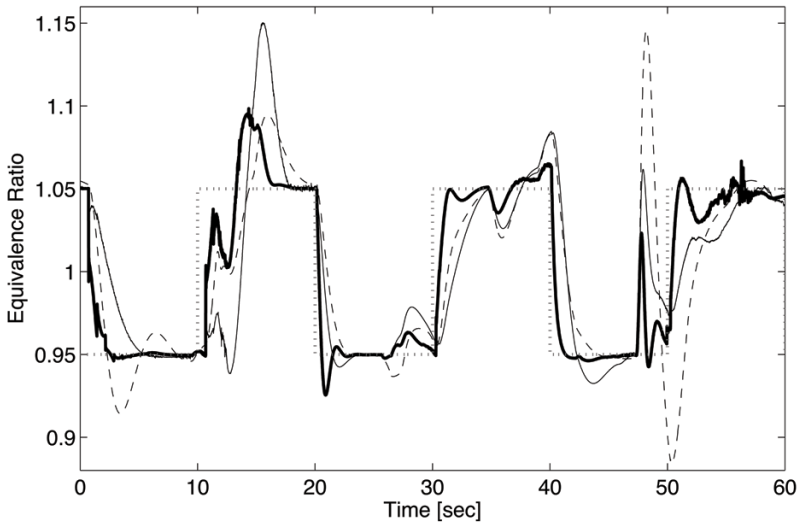


Figure 4.10: Time-Varying Switching Variable



(a) Disturbance Rejection



(b) Reference Tracking

Figure 4.11: Performance of H_∞ Controller (Dashed), S-LPV Controller 1 (Thin), and S-LPV Controller 4 (Thick) With Time-Varying Plant Dynamics

The disturbance rejection of the controllers with the time-varying plant is seen in Figure 4.11(a). A constant reference equivalence ratio, $r = 1$, is given while a square-wave shaped disturbance with 10% amplitude and a period of $20sec$ is applied. S-LPV Controller 4 provides reduced overshoot and improved settling time. The large overshoot at time $t = 12sec$ and spike at time $t = 47sec$ are due to fast parameter movements at those times. Figure 4.11(b) shows the reference tracking performance of the controllers, given a square-wave shaped reference signal, represented by the dotted line. Once again, S-LPV Controller 4 achieves reduced overshoot and improved settling time.

4.3 Adaptive Controller

The simulation results of the adaptive controller obtained in Section 3.4 are presented in this section. Whereas the H_∞, LPV, and switching LPV controllers are developed in continuous time, the adaptive controller is developed and simulated completely in discrete time domain. The discrete time state-space model of the FOPDT plant given in (3.19) is used for simulations and iterative tuning.

4.3.1 Time-Invariant Simulations

In order to demonstrate the adaptive controller's adaptation capability, simulations are first performed with the plant dynamics remaining fixed. At the start of the simulation, at $t = 0$, the estimate vector, known a the spectral gain, is near zero. A non-zero, initial estimate is selected because a zero-estimate results in divergence. An explanation of the adaptive controller design technique, including the Laguerre

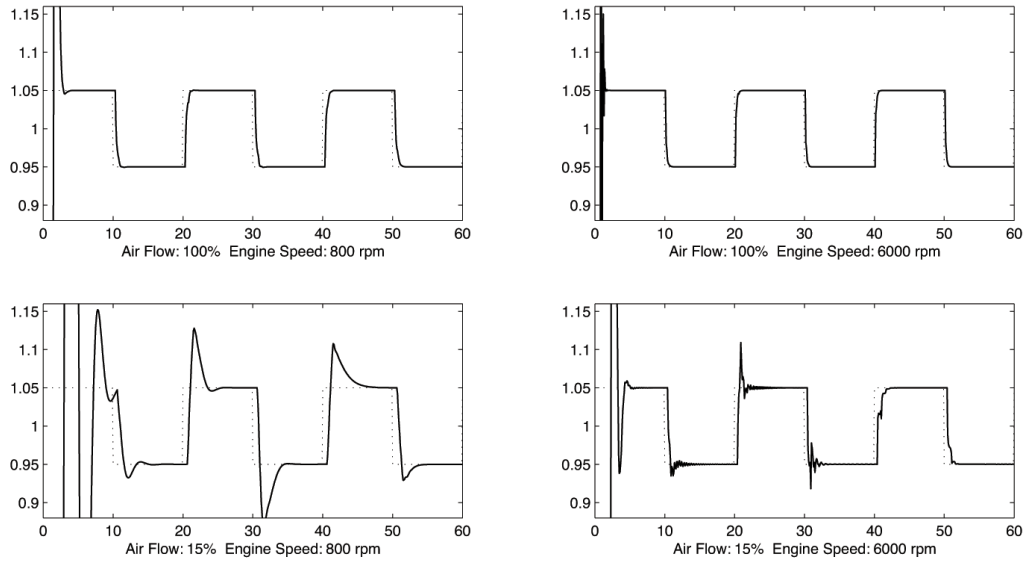


Figure 4.12: Adaptive Controller Reference Tracking for Fixed Operating Points

network and the definition of the spectral gain, is given in Appendix B. As the simulation progresses, the recursive least squares algorithm improves the estimate and the internal model of the plant becomes more accurate.

Figure 4.12 shows the tracking performance of the controller at four constant operating points of the engine. At low engine speed and air flow, when the plant dynamics are at their slowest, the adaptation rate appears to be very slow and the reference tracking performance only becomes acceptable around $t = 40sec$. When either or both engine speed and air flow are high, on the other hand, the adaptation rate is fast, causing the controller to show acceptable reference tracking within the first $10sec$. Recall that no *a priory* information of the plant dynamics is available to the controller, meaning that it is completely 'learning' the dynamics of the plant

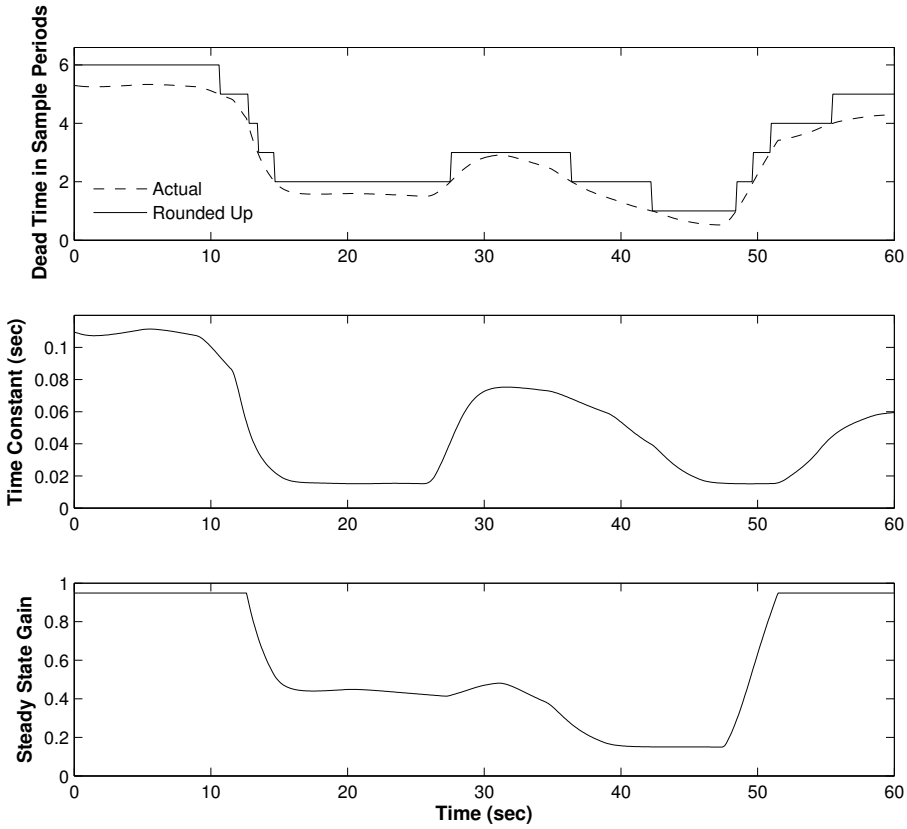


Figure 4.13: Time-Varying FOPDT Parameters

during the simulation.

4.3.2 Time-Varying Simulations

Engine speed and air flow profiles representing realistic engine use are once again employed to validate the system's stability and performance while the engine operating point is varying. The engine speed and air flow profiles used are identical to those used for the switching LPV controller in Section 4.2.2 and are shown in

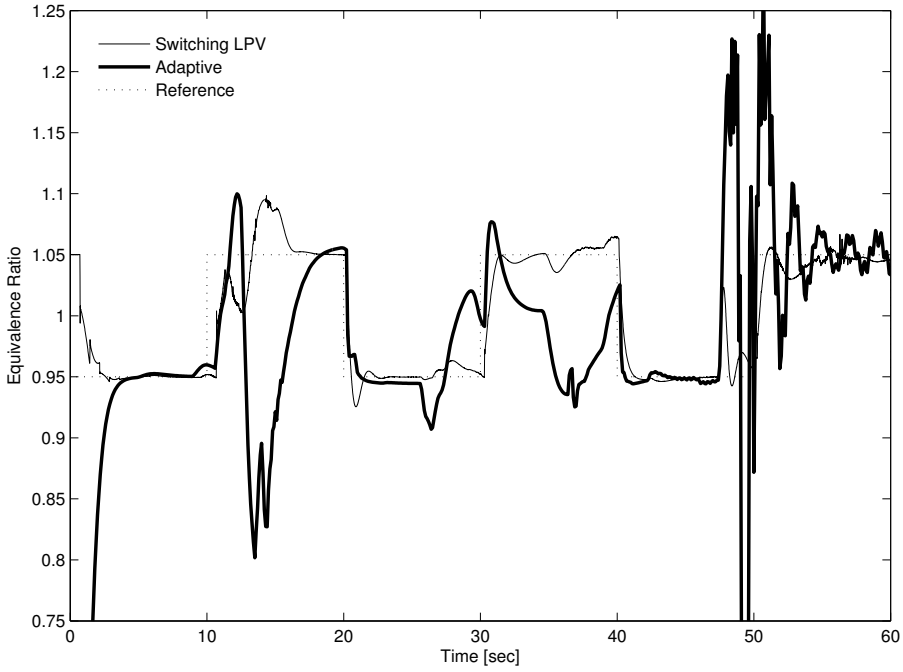


Figure 4.14: Tracking Performance of S-LPV Controller 4 (Thin) and the Adaptive Controller (Thick)

Figure 4.8. The resulting FOPDT parameter values, τ, T, and g, can be seen in Figure 4.13. Because a discrete time model is used to simulate the FOPDT plant, the time delay must be expressed in terms of sample periods, T_s, meaning that it cannot be continuously variable. In order to create a conservative model, the delay is always rounded up. With the sampling interval of $T_s = 0.1sec$ used in these simulations, the resultant variation of the delay is between two and six sampling intervals as seen in Figure 4.13. Using a smaller sampling interval reduces rounding errors but increases the order of the discrete time model.

Figure 4.14 shows the tracking performance of the adaptive controller compared with that of the recommended switching LPV controller using four subregions (S-LPV Controller 4). A reference signal around unity with a period of $20sec$ is given as represented by the dotted line. The adaptive controller exhibits very poor reference tracking performance between $10sec$ and $20sec$ into the simulation, again between $30sec$ and $40sec$, and once again around $50sec$. At each of these times, the engine's operating point is rapidly moving, as can be seen from the air flow and engine speed profiles in Figure 4.8, causing very fast changes in the plant dynamics. Because the dynamics are changing fast, the adaptive controller is unable to keep up, fails to accurately track the reference, and may even become unstable. Using a smaller sampling interval may improve the estimation rate and mitigate this problem but, as has already been stated, increases the order and the complexity of the model and adaptive controller. Further research is required to improve the adaptation rate and robustness of the adaptive controller.

Chapter 5

Conclusions

5.1 Summary

The objective of this research was to develop an air-fuel ratio controller technique for the Spark Ignition (SI) Internal Combustion (IC) engine which appropriately addresses the engine's non-linear and parameter-varying nature as well as its time delays and provides guaranteed closed-loop performance over the engine's entire operating range.

A detailed model of the air-fuel ratio dynamics was developed in Chapter 2. The model demonstrated that the dynamics are non-linear and depend on engine speed, air flow, and manifold pressure, which is related to air flow. Time delays in the system were examined, and are also dependent on engine speed and air flow. The detailed model was not used for controller design purposes. Instead, a widely accepted simplified model of the engine dynamics, known as a First Order

Plus Dead Time (FOPDT) was used. The FOPDT model approximates the combined dynamics as a first order transfer function coupled in series with a pure delay. The model can be completely defined using only three time-varying parameters: a steady state gain, g, first order time constant, τ, and delay time, T. These three parameters were shown to be functions of the engine speed and air flow into the cylinders. An estimate of the air flow into the cylinders was assumed to be available. Methods that successfully estimate the air flow into the cylinders have been demonstrated in prior literature. By selecting air flow as a scheduling parameter, rather than an input to the plant, the engine model was expressed as a pseudo-linear but parameter-dependent transfer function.

Four different controller design techniques were used to develop air-fuel ratio feedback controllers to track reference air-fuel ratios and reject disturbances in Chapter 3. A novel air-fuel ratio controller using switching Linear Parameter Varying (LPV) controller design techniques was introduced. In this method, the operating range of the engine is divided into multiple subregions, and an LPV controller which guarantees performance within its respective subregion using Lyapunov stability theory is developed for each subregion. Switching stability is also guaranteed. The switching LPV controller takes both air flow and engine speed as the scheduling parameter and provides a guarantee of performance over the operating range of the engine. Three other design techniques, H_∞ control, LPV-based control, and adaptive control, were also used to develop controllers for comparative purposes.

Simulations performed in Chapter 4 were used to demonstrate the switching

LPV controller's performance. Firstly, the results were used to validate the choice of air flow and engine speed as the scheduling parameter. An LPV controller scheduled for both engine speed and air flow was compared against an LPV controller which was compensated for only engine speed as well as an H_∞ controller. The LPV controller scheduled for air flow and engine speed outperformed both the LPV controller scheduled for only engine speed and the H_∞ controller. Secondly, switching LPV controllers using two, four, and nine subregions were compared against an LPV controller and an H_∞ controller. Simulations confirmed that using a greater number of subregions, resulting in smaller parameter variations within each subregion, reduced the controller's conservativeness and therefore improved the closed-loop performance.

Finally, the performance of a novel adaptive air-fuel ratio controller was compared with that of the recommended switching LPV controller. Simulations showed that the adaptive controller could not adapt to the rapid changes in the engine's operating point fast enough to be competitive with the switching-LPV controller. This is attributed to the fact that the switching LPV controller, like any gain-scheduled controller, uses *a priory* information on the parameter-dependent dynamics of the plant to compensate for changes in the operating point instantaneously while the adaptive controller needs to dynamically estimate the plant dynamics in real-time.

5.2 Contributions

The main contribution of the current research is the development of a switching LPV air-fuel ratio controller. Although the controller was developed specifically for a port injected, normally aspirated, SI IC engine in this research, it can be applied to air-fuel ratio control problems in other engines, including forced induction, lean burn, and direct injection, as well. The contributions of this thesis are outlined below:

- A detailed model of the SI IC engine's air and fuel dynamics is developed and confirms the engine's non-linear nature and dependency on parameters such as air flow, engine speed, and manifold pressure.

- The delays that the four-stroke engine introduces to the air and fuel flow dynamics are considered in detail. Previous authors have assumed both the air and fuel dwell times to be equal to the time that it takes the engine to complete three strokes: induction, power, and exhaust. Because fuel is injected between one and three strokes before the intake valve opens, the fuel dwell time is actually between four and six strokes.

- By taking air flow as a time-varying parameter rather than an input to the plant, the non-linear combustion equation is expressed in pseudo-linear but parameter-dependent form.

- The FOPDT model of the engine is improved by including a parameter-dependent low frequency gain. The low frequency gain is derived from

the combustion equation and is a function of the air flow.

- The effect of air flow on the delay in the FOPDT model is considered and compensated for.

- The air-fuel ratio control problem is expressed in an LPV form and the switching controller design technique developed in [5] and [23] is applied. The resultant switching-LPV controller has reduced conservativeness, compared to previous methods, and guarantees L_2 performance over the entire operating range of the engine.

5.3 Future Work

The air-fuel ratio control problem in IC engines is a mature problem. However, it is evident, even from related papers published in 2010 only, that the air-fuel ratio continues to benefit from recent and increasingly advanced controller design techniques. In future work, the research presented in this thesis can be expanded in four areas discussed below.

State-Delayed Switching LPV Controller

At the same time that the current research was improving on the LPV-based air-fuel ratio controller presented in [47] by applying switching LPV techniques, the authors of [47] were improving the method and introducing a state-delayed LPV-based controller in [48]. The benefit of using a state-delayed LPV controller over the LPV controller is that the delay in the plant does not need to be modeled by

a Padé approximation. Using a state-delayed LPV controller can also result in closed-loop performance improvements. The same method used to express the LPV plant as a state-delayed LPV plant in [48] can be applied to the current research, and a state-delayed switching LPV controller can be produced.

Optimal Selection of Subregions

In Chapter 3, it is mentioned that the division of the engine's operating space into subregions was found through trial and error. Considerable improvements can be made to the process of dividing the operating range into subregions. An optimal selection of subregions does exist and can be found analytically.

Implementation on Physical IC Engine

The recommended switching LPV controller was simulated on a realistic FOPDT model of the SI IC engine. Once the required hardware is available, a real IC engine's dynamics can be identified and the controller can be implemented.

Improvement to Adaptive Controller Performance

Finally, the novel adaptive air-fuel ratio controller developed in Chapter 3 as an alternative to gain-scheduling lacked the performance to compete with the switching LPV controller. Additional attention to the controller design and tuning method may greatly improve the controller's adaptation rate and consequently its closed-loop performance.

Bibliography

[1] D. Alberer, M. Hirsch, and L. del Re. A virtual references design approach for diesel engine control optimization. *Control Engineering Practice*, 18 (11):1263 – 1271, 2010. ISSN 0967-0661. Special Issue on Automotive Control Applications, 2008 IFAC World Congress. → pages 7

[2] E. Alfieri, A. Amstutz, and L. Guzzella. Gain-scheduled model-based feedback control of the air/fuel ratio in diesel engines. *Control Engineering Practice*, 17(12):1417 – 1425, 2009. ISSN 0967-0661. Special Section: The 2007 IFAC Symposium on Advances in Automotive Control. → pages 7

[3] P. Andersson and L. Eriksson. Three way catalyst control using a PI-style controller with HEGO sensor feedback. CCSSE, Norrköping, Sweden, October 2002. → pages 5, 6

[4] P. Andersson, L. Eriksson, and L. Nielsen. Modeling and architecture examples of model based engine control. In *Proceedings of the Second Conference on Computer Science and Systems Engineering*, Norrköping, Sweden, 1999. → pages 3, 7, 20, 22, 27

[5] P. Apkarian and R. Adams. Advanced gain-scheduling techniques for uncertain systems. *IEEE Transactions on Control Systems Technology*, 6(1): 21–32, 1998. → pages 7, 11, 13, 14, 15, 36, 78, 86, 91

[6] P. Apkarian, P. Gahinet, and G. Becker. Self-scheduled H_∞ control of linear parameter-varying systems: A design example. *Automatica*, 31(9):1251 – 1261, 1995. → pages 7, 13, 14

[7] M. Balenovic, J. Edwards, and T. Backx. Vehicle application of model-based catalyst control. *Control Engineering Practice*, 14(3):223 – 233, 2006. ISSN 0967-0661. Advances in Automotive Control (AC'04). → pages 5

[8] J. Chauvin, G. Corde, N. Petit, and P. Rouchon. Motion planning for experimental airpath control of a diesel homogeneous charge-compression ignition engine. *Control Engineering Practice*, 16(9):1081 – 1091, 2008. ISSN 0967-0661. → pages 7

[9] J. Cook and B. Powell. Modeling of an internal combustion engine for control analysis. *IEEE Control Systems Magazine*, 8(4):20 –26, aug 1988. ISSN 0272-1708. → pages 5, 30

[10] G. Fiengo, J. Cook, and J. Grizzle. Fore-aft oxygen storage control. In *Proceedings of the 2002 American Control Conference*, volume 2, pages 1401 – 1406 vol.2, Anchorage, Alaska, May 2002. → pages 5

[11] G. Fiengo, J. Grizzle, J. Cook, and A. Karnik. Dual-UEGO active catalyst control for emissions reduction: Design and experimental validation. *IEEE Transactions on Control Systems Technology*, 13(5):722–736, 2005. → pages 5, 27

[12] G. Franklin and A. Powell, J.D. Emami-Naeini. *Feedback Control of Dynamic Systems - Sixth Edition*. Prentice-Hall, 6 edition, 2010. → pages 27

[13] P. Gahinet and P. Apkarian. A linear matrix inequality approach to H_∞ control. *International Journal of Robust & Nonlinear Control*, 4(4):421–448, 1994. → pages 12, 51

[14] J. Grizzle, J. Cook, and W. Milam. Improved cylinder air charge estimation for transient air fuel ratio control. In *Proceedings of the 1994 American Control Conference*, Baltimore, Maryland, June 1994. Citeseer. → pages 5, 30

[15] L. Guzzella and C. Onder. *Introduction to modeling and control of internal combustion engine systems*. Springer Verlag, 2004. → pages 19

[16] R. Heck and R. Farrauto. Automobile exhaust catalysts. *Applied Catalysis A: General*, 221(1-2):443–457, 2001. → pages 1

[17] E. Hendricks, M. Jensen, A. Chevalier, and T. Vesterholm. Conventional event base engine control. *SAE International*, 103(3):470 – 489, 1994. → pages 27

[18] Y. Huang and A. Jadbabaie. Nonlinear H_∞ control: An enhanced quasi-LPV approach. In *Proceedings of the 1999 IFAC World Congress*, pages 85–90, Beijing, China, July 1999. → pages 13

[19] N. Kahveci and M. Jankovic. Adaptive controller with delay compensation for air-fuel ratio regulation in SI engines. In *Proceedings of the 2010 American Control Conference*, pages 2236–2241, Baltimore, Maryland, July 2010. → pages 6, 9

[20] U. Kiencke and L. Nielsen. Automotive control systems: For engine, driveline, and vehicle. *Measurement Science and Technology*, 11, 2000. → pages 25, 27, 32

[21] D. J. Leith and W. E. Leithead. Survey of gain-scheduling analysis and design. *International Journal of Control*, 73(11):1001–1025(25), 10 July 2000. → pages 12

[22] B. Lennox, G. Montague, A. Frith, and A. Beaumont. Non-linear model-based predictive control of gasoline engine air-fuel ratio. *Transactions of the Institute of Measurement & Control*, 20(2):103–112, 1998. → pages 6

[23] B. Lu and F. Wu. Switching LPV control designs using multiple parameter-dependent Lyapunov functions. *Automatica*, 40(11):1973–1980, 2004. → pages 14, 15, 33, 42, 78, 86, 93

[24] L. Mianzo, H. Peng, and I. Haskara. Transient air-fuel ratio H_∞ preview control of adrive-by-wire internal combustion engine. In *Proceedings of the 2001 American Control Conference*, volume 4, Arlington, Virginia, June 2001. → pages 6, 8

[25] K. Muske and J. Peyton Jones. Multi-objective model-based control for an automotive catalyst. *Journal of Process Control*, 16(1):27–36, 2006. → pages 5, 10

[26] K. Muske, J. Jones, and E. Franceschi. Adaptive analytical model-based control for SI engine air-fuel ratio. *IEEE Transactions on Control Systems Technology*, 16(4):763–768, 2008. → pages 6, 10, 30, 32

[27] J. Pieper and R. Mehrotra. Air/fuel ratio control using sliding mode methods. In *Proceedings of the 2010 American Control Conference*, San Diego, California, June 1999. → pages 8

[28] M. Postma and R. Nagamune. LPV-based air-fuel ratio control of spark ignition engines using two gain scheduling parameters. In *Proceedings of the ASME 2010 Dynamic Systems and Control Conference*, Cambridge, Massachusetts, September 2010. → pages 36, 58

[29] D. Powell, N. Fekete, and C. Chang. Observer-based air-fuel ratio control. *IEEE Control Systems Magazine*, 18(5):72–83, 1998. → pages 8

[30] W. J. Rugh. Research on gain scheduling. *Automatica*, 36(10):1401–1425, 1999. → pages 13

[31] D. Rupp and L. Guzzella. Iterative tuning of internal model controllers with application to air/fuel ratio control. In *Proceedings of the 2010 American Control Conference*, Zurich, Switzerland, June 2010. → pages 6, 9

[32] D. Rupp and L. Guzzella. Adaptive internal model control with application to fueling control. *Control Engineering Practice*, 18(8):873 – 881, 2010. → pages 6, 9, 27

[33] C. Scherer, P. Gahinet, and M. Chilali. Multi-objective output-feedback control via lmi optimization. 1997. → pages 7, 12

[34] A. G. Stefanopoulou, O. F. Storset, and R. Smith. Pressure and temperature-based adaptive observer of air charge for turbocharged diesel engines. *International Journal of Robust and Nonlinear Control*, 14(6):722–736, 2004. → pages 5, 30

[35] O. F. Storset, A. G. Stefanopoulou, and R. Smith. Adaptive air charge estimation for turbocharged diesel engines without exhaust gas recirculation. *Journal of Dynamic Systems, Measurement, and Control*, 126(3):633–643, 2004. → pages 5, 30

[36] A. Stotsky and I. Kolmanovsky. Application of input estimation techniques to charge estimation and control in automotive engines. *Control Engineering Practice*, 10(12):1371–1383, 2002. → pages 5, 30

[37] R. Turin and H. Geering. Model-reference adaptive A/F-ratio control in an SI engine based on Kalman-filtering techniques. In *Proceedings of the 1995 American Control Conference*, volume 6, Seattle, Washington, June 1995. → pages 6, 10

[38] M. Vajta. Some remarks on Padé-approximations. In *Proc. Third TEMPUS-INTCOM Symposium*, pages 53–58, Veszprém, Hungary, September 2000. Citeseer. → pages 33

[39] M. Won, S. Choi, and J. Hedrick. Air-to-fuel ratio control of spark ignition engines using gaussian network sliding control. *IEEE Transactions on Control Systems Technology*, 6(5):678 –687, 1998. ISSN 1063–6536. → pages 6

[40] F. Wu, X. Yang, A. Packard, and G. Becker. Induced L2-norm control for LPV systems with bounded parameter variation rates. In *Proceedings of the 1995 American Control Conference*, pages 2379–2383, Seattle, Washinton, June 1995. → pages 48, 94

[41] Y. Yildiz, A. Annaswamy, D. Yanakiev, and I. Kolmanovsky. Adaptive air fuel ratio control for internal combustion engines. In *Proceedings of the 2008 American Control Conference*, Seattle, Washington, June 2008. → pages 6, 9

[42] Y. Yildiz, A. M. Annaswamy, D. Yanakiev, and I. Kolmanovsky. Spark ignition engine fuel-to-air ratio control: An adaptive control approach. *Control Engineering Practice*, 18(12):1369 – 1378, 2010. → pages 6, 9

[43] C. Zervoz and G. Dumont. Deterministic adaptive control based on Laguerre series representation. *International Journal of Control*, 48(6):2333–2359, 1988. → pages 12, 53, 98, 101

[44] F. Zhang, K. Grigoriadis, M. Franchek, and I. Makki. Transient lean burn air-fuel ratio control using input shaping method combined with linear parameter-varying control. In *Proceedings of the 2006 American Control Conference*, Minneapolis, Minesota, June 2006. → pages 6, 10, 11

[45] F. Zhang, K. M. Grigoriadis, M. A. Franchek, and I. H. Makki. Linear parameter-varying lean burn air-fuel ratio control for a spark ignition engine. *Journal of Dynamic Systems, Measurement, and Control*, 129(4):404–414, 2007. → pages 6, 10

[46] H. Zhang, Z. Chen, Y. Wang, M. Li, and T. Qin. Adaptive predictive control algorithm based on Laguerre functional model. *International Journal of Adaptive Control and Signal Processing*, 20:53–76, 2006. → pages 13, 54, 98, 100, 101

[47] R. Zope, J. Mohammadpour, K. Grigoriadis, and M. Franchek. Air-fuel ratio control of spark ignition engines with TWC using LPV techniques. In *Proceedings of the ASME 2009 Dynamic Systems and Control Conference*, Hollywood, California, October 2009. → pages 6, 7, 11, 36, 41, 78

[48] R. Zope, J. Mohammadpour, K. Grigoriadis, and M. Franchek. Robust fueling strategy for an SI engine modeled as an linear parameter varying time-delayed system. In *Proceedings of the 2010 American Control Coference*, Baltimore, Maryland, July 2010. → pages 7, 11, 33, 78, 79

Appendix A

Switching LPV Controller Synthesis With Guaranteed L_2 Performance

In this appendix, a method of designing a switching LPV controller with guaranteed L_2 performance that is utilized to design the gain-scheduled air-fuel ratio controllers throughout Section 3.2 is presented. The method is a combination of results in [5] and [23]. Although the body of the thesis addresses the development of a switching LPV controller specifically for air-fuel ratio regulation of SI IC engines, the method described in this appendix is not specific and can be applied to any plant which can be expressed in an LPV form.

A.1 Controller Design Problem

Consider a plant expressed in an LPV form as

$$\begin{aligned} \dot{x} &= A(\theta)x + B_1(\theta)w + B_2(\theta)u, \\ z &= C_1(\theta)x + D_{11}(\theta)w + D_{12}(\theta)u, \\ y &= C_2(\theta)x + D_{21}(\theta)w, \end{aligned} \tag{A.1}$$

where x is the state vector, w is the disturbance input, u is the controlled input, z is the performance channel, and y is the measured output of the plant, and all the matrices have compatible dimensions. The scheduling-parameter, θ, is a column vector containing measurable time-varying parameters that affect the plant dynamics. The bounds on θ as well as $\dot{\theta}$, its rate of change, must be known and are given as

$$\begin{aligned} \theta(t) &\in \Theta, \qquad \forall\, t \geq 0, \\ \dot{\theta}(t) &\in \Theta_d, \qquad \forall\, t \geq 0, \end{aligned} \tag{A.2}$$

where Θ and Θ_d are hyperrectangles defining the space of the scheduling parameters and their derivatives, respectively. The parameter space Θ is divided into R overlapping subregions $\Theta^{(r)},\ r = 1, ..., R$. The superscript in parenthesis refers to the index of a subregion. Variables with a superscript, r, are only defined within the subregion $\Theta^{(r)}$ while variables without it are shared throughout the parameter space Θ.

An LPV controller of the form

$$\begin{aligned} \dot{x}_K &= A_K^{(r)}(\theta,\dot{\theta})x_K + B_K^{(r)}(\theta,\dot{\theta})y, \\ u &= C_K^{(r)}(\theta,\dot{\theta})x_K + D_K^{(r)}(\theta,\dot{\theta})y, \end{aligned} \tag{A.3}$$

is connected to the LPV plant. A set of controller matrices is designed for each subregion and is switched depending on the value of θ as discussed below.

A.2 Switching Variable

At any given time, the scheduling parameter, θ, falls within one of the subregions, $\Theta^{(r)}$, of the parameter space, Θ. We let $\Theta^{(\sigma)}$ denote this subregion where σ is a piecewise-constant function, $\sigma(t) \in \{1,2,...,R\},\ \forall t \geq 0$, and is called the switching variable. Whenever θ leaves the current subregion, $\Theta^{(\sigma)}$, a switching event occurs and the value of σ changes to the index of the subregion that θ is entering. Because switching only occurs when θ leaves its current subregion, letting subregions overlap introduces hysteresis. Figure A.1 explains hysteresis switching graphically and shows two arbitrary overlapping and neighboring subregions including an example trajectory of θ that crosses the two switching surfaces. The points where switching events occur are marked with crosses. The overlap of the two subregions is shaded. At time $t = 0$, θ falls within the subregion $\Theta^{(i)}$ and therefore $\sigma = i$. When the trajectory of θ passes through the switching surface, $S^{(ij)}$, switching occurs and the value of σ becomes j. Switching occurs again only once the trajectory passes through the switching surface, $S^{(ji)}$, and σ equals i from then on. Notice that switching from subregion i to j does not occur along the same

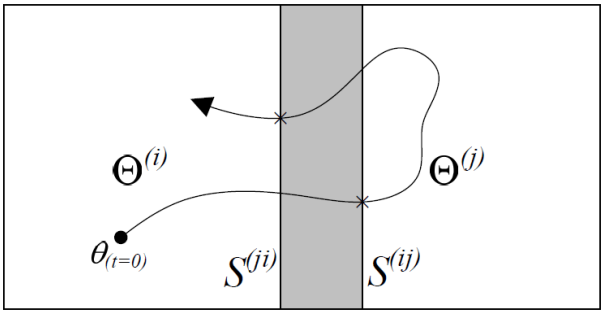


Figure A.1: An Example Parameter Trajectory Causing Switching Between Two Subregions

surface where switching occurs from subregion j to i.

A.3 Controller Design Method

A switching LPV controller of the form (A.3) is developed so that it guarantees internal stability of the closed-loop as well as a bound of the L_2-gain for the closed-loop with (A.1) and (A.3) from w, the disturbance signal, to z, the error signal, for any trajectory of θ satisfying (A.2). The LPV controller matrices are computed by expressing the problem as an optimization problem using a family of Linear Matrix Inequality (LMI)s which are solved using available LMI techniques. The following subsections are organized as follows: The LMIs to guarantee performance within each subregion are first given in A.3.1. The controller's dependence on $\dot{\theta}$, the rate of change of the scheduling parameter, is then removed in A.3.2 giving a *practically valid* controller. Additional LMIs are introduced to guarantee switching performance in A.3.3, and finally the problem is reduced to finite dimensions by gridding the parameter space in A.3.4.

A.3.1 Controller Performance

Within each subregion, closed-loop stability and an L_2-gain bound, denoted by γ, from the disturbance signal to the error signal are guaranteed when the LMI constraints (A.4) and (A.5) hold at all values of θ within the subregion $\Theta^{(r)}$ and the rate of parameter change, $\dot{\theta}$, within Θ_d.

$$\begin{bmatrix} \dot{X}^{(r)} + X^{(r)}A + \hat{B}_K^{(r)}C_2 + (*) & * & * & * \\ (\hat{A}_K^{(r)})^T + A + B_2 D_K^{(r)} C_2 & -\dot{Y}^{(r)} + AY^{(r)} + B_2\hat{C}_K^{(r)} + (*) & * & * \\ (X^{(r)}B_1 + \hat{B}_K^{(r)}D_{21})^T & (B_1 + B_2 D_K^{(r)} D_{21})^T & -\gamma I & * \\ C_1 + D_{12}D_K^{(r)}C_2 & C_1 Y^{(r)} + D_{12}\hat{C}_K^{(r)} & D_{11} + D_{12}D_K^{(r)}D_{21} & -\gamma I \end{bmatrix} < 0, \tag{A.4}$$

$$\begin{bmatrix} X^{(r)} & I \\ I & Y^{(r)} \end{bmatrix} > 0. \tag{A.5}$$

The dependencies on the scheduling parameter, θ, is hidden in (A.4) and (A.5) for brevity. All matrices are functions of θ, however. The LMI variables include the symmetric matrices $X^{(r)}(\theta)$ and $Y^{(r)}(\theta)$, known as the Lyapunov variables, as well as $\hat{A}_K^{(r)}(\theta)$, $\hat{B}_K^{(r)}(\theta)$, $\hat{C}_K^{(r)}(\theta)$, and $D_K^{(r)}(\theta)$, the *quadruple state space matrices*. The bound of the L_2-gain, γ, can be chosen as a constant or can be included as an LMI variable and be minimized, thereby yielding an optimal controller. Matrix terms with an asterisk (*) should be completed such that the LMI matrix (A.4) is symmetric.

Once the LMI variables are obtained, the controller matrices in (A.3) can be computed as

$$\begin{aligned} A_K^{(r)}(\theta) &= [N^{(r)}(\theta)]^{-1}[X^{(r)}(\theta)\dot{Y}^{(r)}(\theta) + N^{(r)}(\theta)[\dot{M}^{(r)}(\theta)]^T + \hat{A}_K^{(r)}(\theta) \\ &\quad -X^{(r)}(\theta)[A(\theta) - B_2(\theta)D_K^{(r)}(\theta)C_2(\theta)]Y^{(r)}(\theta) \\ &\quad -\hat{B}_K^{(r)}(\theta)C_2(\theta)Y^{(r)}(\theta) - X^{(r)}(\theta)B_2(\theta)\hat{C}_K^{(r)}(\theta)][M^{(r)}(\theta)], \\ B_K^{(r)}(\theta) &= [(N^{(r)}(\theta)]^{-1}[\hat{B}_K^{(r)}(\theta) - X^{(r)}(\theta)B_2(\theta)D_K^{(r)}(\theta)], \\ C_K^{(r)}(\theta) &= [\hat{C}_K^{(r)}(\theta) - D_K^{(r)}(\theta)C_2(\theta)Y^{(r)}(\theta)][M^{(r)}(\theta)]^T. \end{aligned} \tag{A.6}$$

Computation of the matrices $M^{(r)}(\theta)$ and $N^{(r)}(\theta)$ is shown below.

A.3.2 Practical Validity

The controller shown in (A.3) is not a usual gain-scheduling controller because of its dependence on $\dot{\theta}$, the rate of change of the scheduling parameter. Since a real-time measurement of $\dot{\theta}$ is usually not available and can be difficult to compute, this dependence is undesirable. Controllers that do not rely on a measurement or estimate of $\dot{\theta}$ are known as *practically valid* controllers [5]. The controller's dependence on $\dot{\theta}$ results from the presence of $\dot{X}^{(r)}(\theta)$ and $\dot{Y}^{(r)}(\theta)$ in (A.4) and of $\dot{Y}^{(r)}(\theta)$ and $\dot{M}^{(r)}(\theta)$ in (A.6). These terms can be eliminated by fixing either one of the Lyapunov variables, $X^{(r)}(\theta)$ or $Y^{(r)}(\theta)$. There is no analytical method for choosing which variable should be constant and which should remain a function of θ. Both cases should be attempted and the resultant closed-loop performance compared in order to find the choice that yields the least conservative controller.

Depending on the choice of Lyapunov variables, the matrices $M^{(r)}(\theta)$ and

$N^{(r)}(\theta)$, used for controller reconstruction in (A.6), are computed as follows: If we choose $X^{(r)}(\theta) = X_0$, a constant, then we have

$$\begin{aligned} M^{(r)}(\theta) &= X_0^{-1} - Y^{(r)}(\theta), \\ N^{(r)}(\theta) &= X_0. \end{aligned} \tag{A.7}$$

Alternatively, for $Y^{(r)}(\theta) = Y_0$, we have

$$\begin{aligned} M^{(r)}(\theta) &= Y_0, \\ N^{(r)}(\theta) &= Y_0^{-1} - X^{(r)}(\theta). \end{aligned} \tag{A.8}$$

A.3.3 Switching Performance

The LMI constraints in (A.4) and (A.5) guarantee closed-loop stability and L_2 performance for the LPV plant and controller combination within each subregion $\Theta^{(r)}$. In order to ensure switching performance, the Lyapunov function must be forced to decrease at any switching event along the switching surface $S^{(ij)}$. Recall that $S^{(ij)}$ denotes the switching surface from an arbitrary subregion $\Theta^{(i)}$ to a neighboring and overlapping subregion $\Theta^{(j)}$. This condition is satisfied when either

$$\begin{aligned} X^{(i)}(\theta) &\geq X^{(j)}(\theta), \\ Y^{(i)}(\theta) - (X^{(i)}(\theta))^{-1} &\leq Y^{(j)}(\theta) - (X^{(j)}(\theta))^{-1}, \end{aligned} \tag{A.9}$$

or equivalently,

$$\begin{aligned} Y^{(i)}(\theta) &\leq Y^{(j)}(\theta), \\ X^{(i)}(\theta) - (Y^{(i)}(\theta))^{-1} &\geq X^{(j)}(\theta) - (Y^{(j)}(\theta))^{-1}, \end{aligned} \tag{A.10}$$

hold for all values of θ along the switching surface. A further explanation of (A.9) and (A.10), as well as the proof, is given in [23] although different notation is used. In order to synthesize a controller subject to the switching constraints given above, the LMI family consisting of (A.4) and (A.5) is augmented with

$$Y^{(i)}(\theta) - Y^{(j)}(\theta) \leq 0 \tag{A.11}$$

if the Lyapunov variable X is chosen to remain constant, or

$$X^{(i)}(\theta) - X^{(j)}(\theta) \geq 0 \tag{A.12}$$

if the Lyapunov variable Y is chosen to remain constant. While the LMIs (A.4) and (A.5) must hold for all values of the scheduling parameter, θ, within each subregion, (A.11) and (A.12) must only hold true for values of θ along the two switching surfaces, $S^{(ij)}$ and $S^{(ji)}$, for each neighboring subregion pair, $\Theta^{(i)}$ and $\Theta^{(j)}$.

A.3.4 Reduction to Finite Dimensional

Because the LMIs (A.4) and (A.5) must hold for all values of the scheduling parameter, θ, within each subregion, and the LMI (A.11) or (A.12) must hold for all

values of θ along the switching surfaces, there exists an infinite number of operating points where the LMI variables need to be solved subject to the constraints. In order to make this family of LMIs solvable, it must therefore be reduced to a finite dimensional problem. This can be done by sampling each subregion with a finite number of points. Let $G^{(r)}$ denote a grid within the space of $\Theta^{(r)}$ and let G_d denote a grid within the space of Θ_d. Since $\dot{\theta}$ enters the LMIs linearly in (A.4) and (A.5), the LMI constraints need to be checked only at its extreme values. G_d therefore only needs to include the vertices of the space Θ_d. Furthermore, let $G_S^{(ij)}$ and $G_S^{(ji)}$ denote grids along the switching surfaces between each neighboring subregion pair. A finite family of LMIs can then be set up for all the subregions, including LMIs (A.4) and (A.5) for all combinations of $G^{(r)} \times G_d$ and LMI (A.11) or (A.12) for each point in $G_S^{(ij)}$ and in $G_S^{(ji)}$, and the LMI variables can be calculated simultaneously.

In order to calculate the controller matrices for any value of θ within Θ while the LMI variables are only solved at a finite number of points, the nature of their parameter dependence needs to be known. A simple solution, proposed by [40], is to mimic the parameter dependence using a series of functions denoted by $\rho(\cdot)$ such that $\rho_i(\cdot),\ i = 1,...,N_\rho$ are differentiable functions of θ. N_ρ represents the number of individual functions required to capture the dependence on θ and can be larger than or equal to the number of elements in θ. Copies of the function $\rho(\cdot)$

are then introduced into the LMI variables as

$$\begin{aligned}
\hat{A}_K^{(r)}(\theta) &= \hat{A}_{K0}^{(r)} + \sum_{i=1}^{N_\rho} \rho_i(\theta)\hat{A}_{Ki}^{(r)}, \\
\hat{B}_K^{(r)}(\theta) &= \hat{B}_{K0}^{(r)} + \sum_{i=1}^{N_\rho} \rho_i(\theta)\hat{B}_{Ki}^{(r)}, \\
\hat{C}_K^{(r)}(\theta) &= \hat{C}_{K0}^{(r)} + \sum_{i=1}^{N_\rho} \rho_i(\theta)\hat{C}_{Ki}^{(r)}, \\
D_K^{(r)}(\theta) &= D_{K0}^{(r)} + \sum_{i=1}^{N_\rho} \rho_i(\theta)D_{Ki}^{(r)},
\end{aligned} \tag{A.13}$$

and

$$\begin{aligned}
X^{(r)}(\theta) &= X_0^{(r)} + \sum_{i=1}^{N_\rho} \rho_i(\theta)X_i^{(r)}, \\
Y^{(r)}(\theta) &= Y_0^{(r)} + \sum_{i=1}^{N_\rho} \rho_i(\theta)Y_i^{(r)}.
\end{aligned} \tag{A.14}$$

The LMI variables now become γ and $\hat{A}_{Kj}^{(r)},\ \hat{B}_{Kj}^{(r)},\ \hat{C}_{Kj}^{(r)},\ D_{Kj}^{(r)},\ X_j^{(r)},\ Y_j^{(r)}$ where $j = 0, 1, ..., N_\rho$ and $r = 1, 2, ..., R$. After the LMI variables are computed subject to the LMI constraints, the quadruple state space matrices and the Lyapunov variables are reconstructed as functions of θ using (A.13) and (A.14) before the controller matrices are calculated using (A.6).

A.4 Recapitulative Procedure

The overall procedure for finding the switching LPV controller can be described as follows:

1. Set up a finite system of LMIs:

(a) Define $G^{(r)}$, a grid within each parameter subspace $\Theta^{(r)}$ as well as G_d, the limits of the parameter's rate of change.

(b) Define $G_S^{(ij)}$, the grids along the switching surfaces between each pair of neighboring subregions.

(c) Define the function $\rho(\cdot)$ so that it mimics the dependence on the scheduling parameter, θ, and use it to define the quadruple state space matrices and the Lyapunov variables as functions of θ in (A.13) and (A.14).

(d) Create a family of LMIs such that an instance of (A.4) and (A.5) is present for each pair $G^{(r)} \times G_d$ of each subregion and an LMI (A.11) or (A.12) is present for each point in $G_S^{(ij)}$ for each neighboring pair of subregions.

2. Solve for variables subject to LMI constraints:

(a) Minimize γ subject to the LMI constraints and compute the values of the LMI variables.

(b) Use the solution of the LMI variables to obtain the quadruple state space data matrices and the Lyapunov variables as functions of θ using (A.13) and (A.14).

3. Check LMI constraints on a denser grid:

(a) Define another grid in each subregion $\Theta^{(r)}$ such that it is denser than $G^{(r)}$.

(b) Define another grid of each switching surface such that it is denser than $G_S^{(ij)}$ and $G_S^{(ji)}$.

(c) Calculate the values of the LMI variables at each point of the denser grid using (A.13) and (A.14) and check that the LMI constraints hold at each point.

(d) If Step 3.c fails, increase the grid density of $G^{(r)}$, $G_S^{(ij)}$, and $G_S^{(ji)}$ and restart Step 3.

4. Compute controller matrices:

(a) In real-time, calculate the controller matrices using (A.6).

(b) Switch the controller matrices in real-time using the switching variable σ such that

$$\begin{aligned} A_K(\theta) &= A_K^{(\sigma)}(\theta), \\ B_K(\theta) &= B_K^{(\sigma)}(\theta), \\ C_K(\theta) &= C_K^{(\sigma)}(\theta), \\ D_K(\theta) &= D_K^{(\sigma)}(\theta). \end{aligned} \tag{A.15}$$

5. Perform Steps 1 through 4 twice, fixing either one of the two Lyapunov variables X and Y each time, and choose the least conservative controller.

Appendix B

Adaptive Predictive Control Algorithm using Laguerre Network

In this appendix, the algorithm used in Section 3.4 to create an adaptive air-fuel ratio controller is presented. The method is introduced in [43] and is further developed in [46] where it is given the title Incremental Mode Linear Laguerre Predictive Control (IMLLPC). The method is suitable for any control problem where the plant dynamics are dominated by a delay. This appendix only serves as an overview of the algorithm and does not show the derivation or any proofs associated with the method.

B.1 Laguerre Network

The Laguerre function is defined as the series

$$\Phi_i(t) = \sqrt{2p}\frac{e^{pt}}{(i-1)!}\cdot\frac{d^{i-1}}{dt^{i-1}}\left[t^{i-1}\cdot e^{2pt}\right], \qquad i = 1,2,\ldots,\infty \tag{B.1}$$

where p is a constant, called the time-scale and $t \in [0,\infty)$ is the time variable. The Laplace transform of the Laguerre Function is

$$\Phi_i(s) = L[\Phi_i(t)] = \sqrt{2p}\frac{(s-p)^{i-1}}{(s+p)i}, \qquad i = 1,2,\ldots,\infty \tag{B.2}$$

Figure B.1 shows the structure of an open-loop stable system approximated by a Laquerre series with order N. The output of the model is

$$Y_m(s) = \sum_{i=1}^{N} C_i\Phi_i(s)U(s) = \sum_{i=1}^{N} C_i l_i(s) \tag{B.3}$$

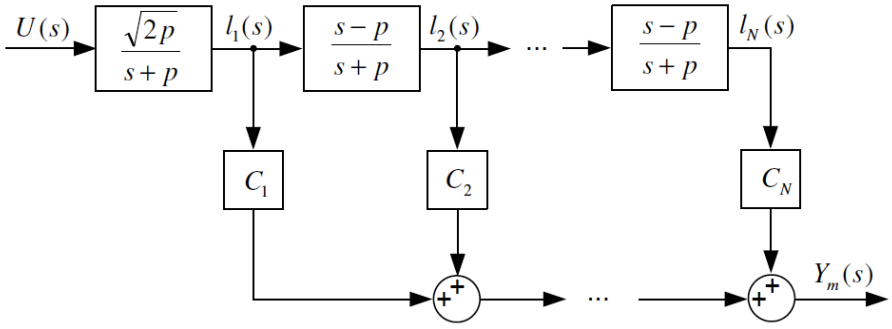


Figure B.1: Laguerre Network Model Structure

where $C = [C_1, \ldots, C_N]$ and is called the Laguerre network's spectral gain and $L = [l_1, \ldots, l_N]$ is the Laguerre network's state vector. The Laguerre network can be expressed in a state-space form as

$$\begin{aligned} L(k+1) &= AL(k) + bu(k), \\ Y_m(k) &= C^T L(k), \end{aligned} \tag{B.4}$$

with the model's input, $u(k)$, and output, $Y_m(k)$.

B.2 IMLLPC Algorithm

The IMLLPC algorithm in [46] makes use of the Laguerre network given in (B.4) as the model of the plant in a model reference adaptive approach. Rather than using the values of the input and output directly, IMLLPC uses incremental values, $\Delta u(k)$, $\Delta Y_m(k)$, and $\Delta L(k)$. The state-space form in (B.4) therefore becomes

$$\Delta L(k+1) = A\Delta L(k) + b\Delta u(k), \tag{B.5}$$

$$\Delta Y_m(k) = C^T \Delta L(k). \tag{B.6}$$

By using incremental values, integration is effectively added to the resulting controller and reference tracking performance is improved. The IMLLPC algorithm is given below.

B.2.1 User-Defined Parameters

The following values are defined by the designer and serve as tuning parameters. For considerations and instructions on the selection of parameters, see [43] and [46].

- T_s Sampling time
- p Time-scale
- N Order of Laguerre network
- d Prediction horizon
- M Control horizon
- Q Weighting matrix on error
- R Weighting matrix on control input
- α Softening factor on reference model output
- λ Forgetting factor for recurser

B.2.2 Constant Matrices

The A and b state-space matrices of the Laguerre network are constants and are given as

$$A = \begin{bmatrix} \tau_1 & 0 & \cdots & 0 \\ \frac{-(\tau_1\tau_2+\tau_3)}{T_s} & \tau_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \frac{(-1)^{N-1}\tau_2^{N-2}(\tau_1\tau_2+\tau_3)}{T_s} & \cdots & \frac{-(\tau_1\tau_2+\tau_3)}{T_s} & \tau_1 \end{bmatrix}, \tag{B.7}$$

and

$$b^T = \begin{bmatrix} \tau_4 & (\frac{-\tau_2}{T_s})\tau_4 & \cdots & (\frac{-\tau_2}{T_s})^{N-1}\tau_4 \end{bmatrix}, \tag{B.8}$$

where

$$\begin{aligned}
\tau_1 &= e^{-pT_s}, \\
\tau_2 &= T_s + \tfrac{2}{p}(e^{-pT_s} - 1), \\
\tau_3 &= -T_s e^{-pT_s} - \tfrac{2}{p}(e^{-pT_s} - 1), \\
\tau_4 &= \sqrt{2p}\tfrac{1-\tau_1}{p}.
\end{aligned} \tag{B.9}$$

B.2.3 Recursive Estimation

The C matrix of the Laguerre network's state-space form is estimated in a recursive loop such that

$$\hat{C}(k) = \hat{C}(k-1) + \frac{P(k-1)\Delta L(k)}{\lambda + \Delta L^T(k)P(k-1)\Delta L(k)} \cdot \left[\Delta y(k) - \hat{C}^T(k)\Delta L(k-1)\right], \tag{B.10}$$

and the covariance matrix is updated using

$$P(k) = \frac{1}{\lambda}\left[P(k-1) - \frac{P(k-1)\Delta L(k)\Delta L^T(k)P(k-1)}{\lambda + \Delta L^T(k)P(k-1)\Delta L(k)}\right]. \tag{B.11}$$

B.2.4 Control Move Calculation

The current output of the Laguerre network is

$$y_m(k) = \hat{C}^T(k) \cdot \Delta(k), \tag{B.12}$$

and the estimated future plant output is

$$\hat{Y}_p(k+1) = SH_l\Delta L(k) + \Phi y_m(k) + K\left[y(k) - y_m(k)\right], \tag{B.13}$$

with $y(k)$, the measured output of the plant. The matrices, S, H_l, Φ, and K are constants and are defined as

$$S = \begin{bmatrix} 1 & \cdots & 0 & 0 \\ \vdots & & & 0 \\ 1 & & & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}_{d\times d} \quad H_l = \begin{bmatrix} \hat{C}^T(k)A \\ \hat{C}^T(k)A^2 \\ \vdots \\ \hat{C}^T(k)A^d \end{bmatrix}_{d\times N} \quad \Phi = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}_{d\times 1} \quad K = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}_{d\times 1} \tag{B.14}$$

respectively.

The controller outputs, including the current and future outputs up to $k+M-1$ are calculated using

$$\Delta U_M(k) = (H_u^T S^T Q S H_u + R)^{-1} H_u^T S^T Q\left[Y_r(k+1) - \hat{Y}_p(k+1)\right], \tag{B.15}$$

where $Y_r(k+1)$, the reference output is computed from

$$y_r(k+i) = \alpha^i y(k) + (1-\alpha^i)w, \qquad i = 1,2,\cdots,d \tag{B.16}$$

and w is the setpoint. H_u is a constant matrix:

$$H_u = \begin{bmatrix} \hat{C}^T(k)b & 0 & \cdots & 0 \\ \hat{C}^T(k)Ab & \hat{C}^T(k)b & & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \hat{C}^T(k)A^{M-1}b & \hat{C}^T(k)A^{M-2}b & \cdots & \hat{C}^T(k)b \\ \vdots & \vdots & & \vdots \\ \hat{C}^T(k)A^{d-1}b & \hat{C}^T(k)A^{d-2}b & \cdots & \hat{C}^T(k)A^{d-M}b \end{bmatrix}_{d\times M} \tag{B.17}$$

The current control output is then simply extracted from the results of (B.15) using the equation

$$\Delta u(k) = D\Delta U_M(k), \tag{B.18}$$

where

$$D = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}_{1\times M}. \tag{B.19}$$

Finally, the Laguerre network states are updated using (B.5).