

# Optimal Modular Control of Overactuated Systems - Theory and Applications

by

Christopher R. Vermillion

A dissertation submitted in partial fulfillment  
of the requirements for the degree of  
Doctor of Philosophy  
(Electrical Engineering - Systems)  
in The University of Michigan  
2009

Doctoral Committee:

Professor Jing Sun, Chair  
Professor James S. Freudenberg  
Professor Jessy W. Grizzle  
Professor Anna G. Stefanopoulou  
Kenneth R. Butts, Toyota Engineering and Manufacturing,  
North America

© Christopher R. Vermillion 2009  
All rights reserved.

To my father, Raymond Vermillion, who has from humble beginnings, through persistence and hard work, provided a better life for himself and his family. And to my mother, Linda Vermillion, who through her care and dedication has helped to make me into the person I am today.

## Acknowledgements

I first owe a great debt of gratitude to my advisor, Professor Jing Sun, who through a combination of patience and persistence has enabled me to grow more than I knew possible during the course of my Ph.D. It was largely through her insistence on my completing proofs “on the spot” on the white board in her office that I grew more comfortable with the theoretical aspects of controls. And while there were many times where I wondered why I was doing this, I now realize how these exercises contributed to my growth and ability to make contributions to the body of control theory. I don’t think I will ever quite understand how a person can be so patient through a student’s struggles and so persistent in the end goal, while remaining so confident that the student will ultimately succeed.

I am extremely thankful to Dr. Ken Butts of Toyota Technical Center, for his high level of confidence in the practical value of my theoretical work, sometimes exceeding my own level of confidence (but Ken, if you are reading this, I sincerely hope that you preserve this high level of confidence even if my own confidence in the merit of my work might drop at one point in time). I do not think there are too many practicing engineers that can so quickly relate abstract concepts to real problems. I am very grateful for the opportunity to work alongside one of only a few practicing control engineers with a true appreciation for the role of new and emerging control theory in important automotive applications.

I must thank Professor Jim Freudenberg for what is quite possibly the best graduate course in linear feedback control in the world, EECS 565: Linear Controls Boot Camp (I think that’s the name that they’re listing it under). Jim’s impressive knowledge and thorough treatment of fundamental linear control concepts has enabled me to develop theory for linear, modular control systems, with confidence. I am also thankful for Jim’s availability to meet on several occasions during the course of my Ph.D. to discuss linear modular control and MCEC results (using  $\mu$  synthesis). I can definitely state that having one of the top researchers in the field “on site” certainly makes a difference in the ability of students to produce important results and have confidence in those results.

I also must thank Professor Jessy Grizzle for his outstanding teaching (both in nonlinear systems and functional analysis), which has provided me with important fundamental knowledge relating to both of these fields. Furthermore, taking functional analysis under Jessy, I significantly increased my ability to make and prove original theory.

I owe a great debt of gratitude to Professor Anna Stefanopoulou, who, through supervising me in an undergraduate directed study where I designed the University of Michigan Solar Car Team's cruise control system, sparked my interest in obtaining a Ph.D. in control. Her optimistic personality, coupled with an impressive knowledge of control theory, provided me the excitement to pursue a career in controls.

I would like to thank Professor Le Yi Wang of Wayne State University for worthwhile and interesting discussions regarding robust control, particularly related to the development of  $H_\infty$  and  $\mu$  synthesis methods for modular control systems.

I would like to thank my lab mates, both current (Reza, Gayu, Zhen, Yanhui, and So-ryeok) and former (DK, Amey, Vasilis, Handa, Jian, and Charles) for the time they have devoted to helping me with problems and bugs, and listening to all of my presentations. I also thank them for putting up with my sense of humor, though I must say, I don't know what they're going to do for comic relief once I'm gone. I must extend special thanks to Reza Ghaemi for all his help regarding model predictive control (MPC) theory and for many trips to the fantastic restaurant, La Shish (Note: The name of this restaurant changes almost weekly, so students who stumble upon this dissertation might know it as Charlie's Mediterranean, Palm Palace, Charlie's La Shish, or "that restaurant on the corner of Washtenaw and Carpenter"). I also must extend special thanks to Gayathri Seenumani for her patience in helping me, as well as many lunches enjoyed by both of us, including lunches with Reza at La Shish (This is quite impressive since she actually hates that restaurant).

I want to thank my friends for putting up with me through the course of my Ph.D. I especially want to thank Michael Brackney for his friendship, wise advice, and many stories about our mutual acquaintance, Bill Brasky (a 10 foot tall beast man who showers in vodka and feeds his babies shrimp scampi). I also want to thank Robert Vogt, a roommate for the past 4 years, for putting up with my inability to go out to the bar on several occasions approaching the Ph.D. defense. I wish to thank Dr. Buz McCain, a friend, colleague, and, at one time, fellow intern at Toyota, for his wise advice, interesting political discussions, and a sense of humor and sarcasm only rivaled by my own. Finally, I wish to thank Samantha Stewart. Though things didn't ultimately work out for us, she remains a friend and an inspiration to me.

Finally, I must thank my family, who has supported me through my journey through academia, and who have shown great interest in hearing about my work (or at least they do a good job of smiling and nodding), even though the work itself is outside their own area of interest and expertise.

# Table of Contents

Dedication . . . . .	ii
Acknowledgements . . . . .	iii
List of Figures . . . . .	ix
List of Tables . . . . .	xii
List of Appendices . . . . .	xiii
 Chapter	
<b>1. Introduction</b> . . . . .	1
1.1 Literature Survey and Prior Art . . . . .	2
1.1.1 Centralized vs. Modular Control Strategies . . . . .	3
1.1.2 Static vs. Dynamic Control Allocation . . . . .	6
1.1.3 Open Modular Control Challenges . . . . .	7
1.1.4 Open Challenges Relating to Dynamic Control Allocation (Through MPCA) . . . . .	8
1.2 Dissertation Contributions . . . . .	9
1.2.1 Contributions to Modular Control Design Theory . . . . .	10
1.2.2 Contribution to Model Predictive Control Allocation . . . . .	11
1.2.3 Experimental Contributions on a Thermal Management System . . . . .	11
1.3 Dissertation Outline . . . . .	12
<b>2. Engine Thermal Management: A Test Bed for Control Development</b> . . . . .	13
2.1 Experimental Setup for System Identification and Rapid Controller Prototyping . . . . .	15
2.2 Dynamic Model . . . . .	17
2.3 Model Based Dynamic Analysis . . . . .	23
<b>3. Modular Control Design - Stability and Performance</b> . . . . .	26
3.1 The Notion of Relative Degree . . . . .	26
3.2 Mathematical Formulation of the Reference Model Based Modular Control Approach . . . . .	27

3.3	Stability Under the Reference Model Based Framework . . . . .	31
3.3.1	Verifying Stability Under Other Modular Control Design Approaches . . . . .	37
3.4	Performance Under the Reference Model Based Framework . . . . .	38
3.4.1	Overall System Performance Metric . . . . .	39
3.4.2	Inner Loop Reference Model Matching Conditions . . . . .	40
3.4.3	Modular vs. Centralized Performance . . . . .	44
3.4.4	Summary - Ideal Modular Control Conditions . . . . .	46
3.4.5	Performance Guarantees Under Other Modular Control Design Approaches . . . . .	47
3.5	Performance Recovery Formulation . . . . .	48
3.5.1	MCEC Design for Performance Recovery Using $\mu$ Synthesis . . . . .	51
3.6	Engine Thermal Management Application . . . . .	59
3.6.1	Modular System Description and Properties . . . . .	59
3.6.2	Inner Loop Reference Model . . . . .	60
3.6.3	Outer Loop Control Design . . . . .	60
3.6.4	Inner Loop Design . . . . .	61
3.6.5	Performance Recovery in the Presence of Uncertainty - $\mu$ Synthesis Setup . . . . .	61
3.6.6	Simulation Results and Analysis . . . . .	65
3.6.7	Experimental Results . . . . .	67
<b>4.</b>	<b>Model Predictive Control Allocation (MPCA) for Leveraging Overactuation . . . . .</b>	<b>71</b>
4.1	MPCA Essentials . . . . .	71
4.2	MPCA for the Thermal Management Application . . . . .	74
4.2.1	Control Design . . . . .	74
4.2.2	Real Time Simulation Results . . . . .	76
4.2.3	Experimental Results . . . . .	77
4.3	Stability-Constrained MPCA . . . . .	82
4.3.1	Stability Goal and Mathematical Preliminaries . . . . .	82
4.3.2	Conditions for Deriving Terminal Constraint Sets for Stabilizing MPCA . . . . .	84
4.3.3	Deriving Terminal Constraint Sets for the Thermal Management System . . . . .	86
4.3.4	Terminal Constraint Sets for Stabilizing MPCA - Generalization . . . . .	88
4.3.5	MPCA Cost Function and Constraints . . . . .	92
<b>5.</b>	<b>Combined MPCA and MCEC . . . . .</b>	<b>95</b>
<b>6.</b>	<b>Conclusions and Future Work . . . . .</b>	<b>98</b>
6.1	Conclusions . . . . .	98
6.2	Future Work . . . . .	100
6.2.1	Modular vs. Centralized Performance Comparison for Nonlinear Systems . . . . .	100
6.2.2	MPCA Stability Constraints . . . . .	101



6.2.3	Stabilization to Manifolds with MPCA . . . . .	101
6.2.4	Combined MCEC and MPCA . . . . .	101
<b>Appendices</b>	. . . . .	<b>103</b>
<b>Bibliography</b>	. . . . .	<b>114</b>

## List of Figures

1.1	Block diagram for the modular, inner loop/outer loop control system.	5
1.2	Diagram of the system (1.1) recast for the reference model based design framework. The figure reflects the distribution of design tasks (separated by dashed lines) between the inner and outer loop designs, as well as an analysis framework that is used to yield important stability and performance results. . . . .	10
2.1	Thermal management system schematic. . . . .	14
2.2	Photograph of the fluid conditioning unit for the oil system (a very similar unit exists for coolant, and both are connected to the engine block via flexible hose. . . . .	15
2.3	Schematic of the oil system, which will be the only system considered in our experimental validation. . . . .	16
2.4	Thermal management rapid prototyping configuration . . . . .	17
2.5	Diagram of the signal flow between all of the interconnected subsystems in the engine thermal management system . . . . .	20
2.6	Heat exchanger schematic. . . . .	22
2.7	Model validation results. . . . .	23
2.8	Extent of loop decoupling. . . . .	24
2.9	Comparison of transfer functions between actuator inputs and outlet temperature at various actuator settings (AS's): AS1 - $\dot{Q}_{ht} = 0$ kW, $\phi = 0.5$ ; AS2 - $\dot{Q}_{ht} = 4.5$ kW, $\phi = 0.5$ ; AS3 - $\dot{Q}_{ht} = 0$ kW, $\phi = 0.9$ . . . . .	25
3.1	Block diagram of a target calculation based approach to modular control design. . . . .	30
3.2	Block diagram of the reference model based modular control design with MCEC. . . . .	50
3.3	Generic $\mu$ synthesis formulation, without consideration of performance. . . . .	52
3.4	Diagram illustrating the setup for the D-K iteration on a generic system. For our system, $K$ is equivalent to $C_v$ . . . . .	53
3.5	$\mu$ synthesis performance formulation. Physical entities of the system are contained inside the dashed lines, whereas fictitious entities ( $k$ and $\Delta_P$ ), which are strictly introduced for $\mu$ synthesis design purposes, lie outside. . . . .	55
3.6	Flow chart of the iterative $\mu$ synthesis algorithm used for MCEC design. . . . .	58

3.7	Performance comparison with and without MCEC, for different reference model time constants, based on speculated sources of uncertainty. In parameterizing the reference model, the figure takes $\tau_f = 0.25(2^n)$ . . . . .	64
3.8	Simulation results with and without MCEC for $\tau_f = 4$ seconds. . . . .	66
3.9	Simulation results with and without MCEC for $\tau_f = 1$ second. . . . .	66
3.10	Open loop frequency response comparison with and without MCEC for $\tau_f = 4$ seconds. . . . .	67
3.11	Open loop frequency response comparison with and without MCEC for $\tau_f = 1$ second. . . . .	68
3.12	Experimental results with $\tau_f = 4$ seconds. . . . .	69
3.13	Experimental results with $\tau_f = 1$ second. . . . .	69
3.14	Experimental results with $\tau_f = 1$ second - zoomed in. . . . .	70
3.15	Quantitative performance based on the metric specified in (3.67). . . . .	70
4.1	Full system diagram, with MPCA and an outer loop predictor, recast for the reference model based design framework . . . . .	74
4.2	Setpoint tracking response - real-time simulation results . . . . .	77
4.3	Actuator responses - real-time simulation results . . . . .	78
4.4	Virtual control input response - real-time simulation results. Note that the desired virtual control here represents $v_{des}^f$ . . . . .	79
4.5	Computation time on the real-time simulator . . . . .	79
4.6	Setpoint tracking response - experimental results . . . . .	80
4.7	Actuator responses - experimental results . . . . .	80
4.8	Virtual control input response - experimental results. Note that the desired virtual control here represents $v_{des}^f$ . . . . .	81
4.9	Contours of $V_2$ , with $x_1^* = 2.5$ , $v_{des}^* = 1$ . Blue x indicates where $u^r \notin U$ , red o indicates where $V_2$ is non-decreasing. . . . .	88
4.10	Contours of $V_2$ , with $x_1^* = 2.5$ , $v_{des}^* = 2$ . Blue x indicates where $u^r \notin U$ , red o indicates where $V_2$ is non-decreasing. . . . .	89
4.11	Contours of $V_2$ , with $x_1^* = 2.5$ , $v_{des}^* = 3$ . Blue x indicates where $u^r \notin U$ , red o indicates where $V_2$ is non-decreasing. . . . .	90
4.12	Contours of $V_1$ , with $\tilde{v}^* = 0.1$ . Blue x indicates where $x_1' \notin X_1'$ , magenta + indicates where $v_{des} \notin V_{des}$ , red o indicates where $V_1$ is non-decreasing. . . . .	91
4.13	Contours of $V_1$ , with $\tilde{v}^* = 0.3$ . Blue x indicates where $x_1' \notin X_1'$ , magenta + indicates where $v_{des} \notin V_{des}$ , red o indicates where $V_1$ is non-decreasing. . . . .	92
4.14	Contours of $V_1$ , with $\tilde{v}^* = 0.5$ . Blue x indicates where $x_1' \notin X_1'$ , magenta + indicates where $v_{des} \notin V_{des}$ , red o indicates where $V_1$ is non-decreasing. . . . .	93
5.1	Experimental results with combined MCEC and MPCA with a reference model time constant of $\tau_f = 4$ seconds. . . . .	96

5.2	Experimental results with combined MCEC and MPCA with a reference model time constant of $\tau_f = 1$ second. . . . .	97
5.3	Performance with combined MCEC and MPCA. . . . .	97

## List of Tables

2.1	Model inputs . . . . .	17
2.2	Model states . . . . .	18
2.3	Model parameters . . . . .	19
3.1	Uncertain actuator model parameters . . . . .	62
4.1	Generic signals and their thermal management counterparts . . . . .	86
A.1	Model parameters . . . . .	110

## List of Appendices

A.	Thermal Management System Identification . . . . .	104
B.	Nonlinear Reference Model Matching . . . . .	111

# Chapter 1

## Introduction

This dissertation is focused on advancing the art of modular strategies for the control of overactuated systems. Here, an overactuated system is defined as one in which there are more control inputs than variables to be controlled to a setpoint. Modular control design, as it is used in this dissertation, refers to the design of different control components that are ultimately linked together in order to achieve an overall system goal. The modular control design process can be conducted by an individual, but it is often the case that several individuals, groups, or vendors design the constituent controllers in parallel. This dissertation will consider a particular inner-loop/outer-loop modular control structure that is well motivated and widely employed by overactuated systems, developing important theoretical and experimental advances for this strategy.

Overactuated systems arise in engineering applications for a variety of reasons. For example, systems may be equipped with several identical (or very similar) actuators in order to remain operable in the case of actuator failure. Most safety-critical systems have redundancy mandates that necessitate this design. In other cases, the incorporation of additional actuators is a consequence of other constraints on the system that may relate to either system efficiency or dynamic performance. In this case, while the additional actuators are not needed in order to control outputs to setpoints, they are necessary to meet other performance criteria. This is common in the automotive industry, for example, where the emissions, efficiency, and driveability criteria must all be satisfied in addition to controlling torque or speed. Additionally, overactuation may arise as part of a design when, under certain operating conditions, one or more actuators lose authority. This dissertation will examine in detail a thermal management system where an auxiliary heater is used for just this purpose. This dissertation will develop control methodologies for overactuated systems, address the stability of the closed-loop system, and explore methods that may be used to optimize

the closed-loop performance of such systems.

Overactuation gives rise to control design opportunities that are not present when the number of control inputs is equal to the number of performance variables that are controlled to a setpoint. In particular, for overactuated systems, there typically exist many combinations of control inputs that achieve the same steady-state output, or even the same dynamically evolving trajectory. These additional inputs offer the control system designer additional degrees of freedom for performance optimization.

In order to take advantage of the additional degrees of freedom, however, the control strategy must take into account important characteristics of the overactuated system. In most real systems, the actuators will possess hard saturation limits and/or different dynamic authorities (“fast” and “slow” actuators). Therefore, the goal should be to optimize performance with consideration of these features, in addition to any application-dependent performance goals. In considering all of these features, the control methodology should also be practical, not involving a large-scale optimization that will ultimately be intractable in spite of its theoretical merits.

In light of the opportunities and challenges in controlling overactuated systems, research to date has taken several approaches. In particular, a large body of literature has focused on a modular, inner loop/outer loop design framework that attempts to divide the control design between a simple outer loop controller that does not consider overactuation or constraints and an inner loop controller that optimizes performance in the presence of actuator dynamics and constraints. By pursuing the modular control strategy, the allocation of many control inputs to achieve a desired effect is simplified through a significant reduction in the number of states considered. The outer loop control design can often be accomplished using well-established techniques, since the outer loop does not possess overactuation or input constraints. This modular approach comes with significant challenges, however, and this dissertation work aims to advance the state of the art with regard to many of these challenges.

## 1.1 Literature Survey and Prior Art

Most literature to date on the control of overactuated systems arises from applications, including the robotics [1]-[2], automotive [3]-[5], aerospace [6]-[16], marine [17]-[21], and computer [22] industries. In robotics, manipulators must often be designed to be redundant in order to avoid kinematic singularities (where the actuator(s) at one or more joint(s) loses authority because of the geometric orientation of the robot).



The result is that, under all other circumstances, the robot possesses more degrees of control authority at the joints than end effector position and orientation variables to be controlled. In the marine and aerospace industry, although a ship or aircraft possesses 6 degrees of freedom, it is often the case that only a subset of these are controlled to setpoints at a given time. Under these circumstances, there exist more control inputs (control surfaces and thrust inputs) than performance outputs, and these additional degrees of freedom may be used in order to optimize maneuverability. As an additional challenge, many of the actuators may possess different dynamic authorities. For example, control surfaces are often a faster source of actuation than engines. In the automotive industry, both powertrain control systems [3]-[4] and vehicle stability control systems [5] may be viewed as overactuated. In the case of vehicle stability control, four brake pressures are used to control a single yaw moment on the vehicle, and braking can be strategically allocated to take into consideration friction limitations between the tire and the road. In the case of automotive powertrain control, the automotive engine represents a very challenging application of overactuated systems, where several control inputs (air, spark advance, fuel, variable valve timing (VVT), continuously variable transmissions (CVT), exhaust gas recirculation (EGR), and variable vane geometry for turbocharged engines) may all be used in order to control speed and air/fuel ratio to setpoints. This excess of control inputs is necessary for automotive engines in order to meet stringent emissions requirements, while achieving high performance and efficiency in order to satisfy the customer. Finally, overactuated systems are even seen in the area of magnetic disc drives [22], where fine and coarse actuators are used for positioning.

Although overactuated systems span a diverse range of applications, similar issues are faced by each. In particular, in every application that has been examined here, all actuators are subject to hard saturation limits, regardless of whether they arise from current limits on DC motors, range limitations on control surfaces, or friction limits on tires. Additionally, in many applications, the actuators possess non-negligible dynamics, and some offer faster control authority than others. The control objective always includes dynamic performance, but may also taking into account additional safety/emissions considerations.

### 1.1.1 Centralized vs. Modular Control Strategies

Existing methodologies for controlling overactuated systems can be characterized as *centralized*, which consists of a single controller, and *modular*, which separates the control design process into an inner and outer loop.

In the centralized control strategy, a single controller determines  $p$  control inputs to regulate a single output to a setpoint,  $r$ . As an application example, centralized strategies have been employed in many automotive powertrain applications, most notably idle speed control [3]-[4], which uses both spark advance and air bypass to regulate engine speed. Centralized strategies are also used in the control of magnetic disc drives in [22]. The centralized approach was also the original control strategy employed on the thermal management system modeled in [23]-[25], which is implemented on a test cell used for dynamometer-based engine testing and will be used as a test bed for much of the work in this dissertation. By using a single controller to determine all of the control inputs, the centralized strategy enables a single designer to construct a theoretically optimal controller for the whole system. However, since one controller determines the control inputs to be provided to the system, the design of this controller must consider all of the system dynamics. In addition to being a very time-consuming process, this method of a design may not lead to an effective closed-loop system in an industrial environment where different groups or vendors work on different components of the system. And if a small hardware change is made (for example, a new actuator is added or an actuator is replaced with a better version), the controller may need to be redesigned, necessitating another time-consuming design process. Finally, any optimization that is used to leverage the overactuation in the system must consider all of the system states, which can lead to an intractable problem.

When weighing the theoretical advantage of the centralized approach with the practical disadvantages, many control systems engineers have concluded that the centralized approach is indeed an unattractive one. As an alternative to the centralized approach, most applications of overactuated systems use a modular approach, depicted in Figure 1.1, where the control system is divided between an inner loop and outer loop controller [5]-[21]. At the core of the modular control approach is the identification of a *virtual control input*, which represents an overall force, moment, or generalized effect that characterizes the combined actions of all of the actuators. For example, in the vehicle stability control, flight control, and marine applications, the virtual control input is a moment (roll, pitch, or yaw), about the appropriate body-fixed axis [5]-[21]. The real control inputs, consisting of brake pressures, control surface deflections, or rudder angles/propeller inputs (depending on the application) combine effects to produce this virtual control input. It is the virtual control input that drives the plant dynamics (the vehicle dynamics, in this case), and knowledge of the virtual control input (without knowing the real control inputs) is sufficient to

characterize the behavior of the plant. By introducing this virtual control input, the state-space representation of the system can be expressed as:

$$\begin{aligned}
 \dot{x}_1 &= f_1(x_1, v), \\
 \dot{x}_2 &= f_2(x_1', x_2, u), \\
 v &= g(x_2), \\
 y &= h(x_1),
 \end{aligned} \tag{1.1}$$

with its discrete time counterpart represented by:

$$\begin{aligned}
 x_1(k+1) &= f_1^d(x_1(k), v(k)), \\
 x_2(k+1) &= f_2^d(x_1'(k), x_2(k), u(k)), \\
 v(k) &= g(x_2(k)), \\
 y(k) &= h(x_1(k)),
 \end{aligned} \tag{1.2}$$

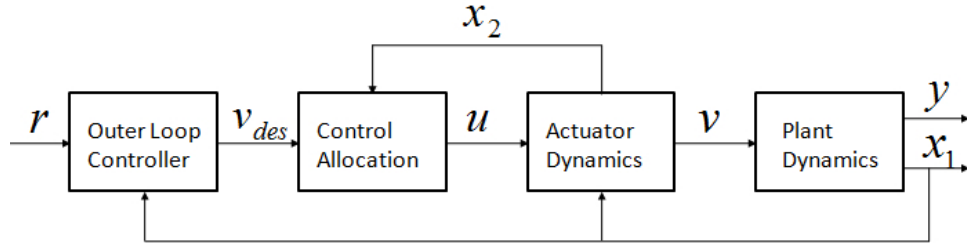


Figure 1.1: Block diagram for the modular, inner loop/outer loop control system.

where  $v \in \mathbb{R}$  represents the virtual control input,  $x_1 \in \mathbb{R}^{n_1}$  represents the plant states, which are driven by the virtual control input,  $v$ , whereas  $x_2 \in \mathbb{R}^{n_2}$  represents the actuator states, which are driven by the real control inputs,  $u$ .  $x_1' \in \mathbb{R}^{n_1'}$ , with  $n_1' \leq n_1$ , represents a subset of plant states that affect the actuator dynamics. Here,  $f_1 : (\mathbb{R}^{n_1}, \mathbb{R}) \rightarrow \mathbb{R}^{n_1}$  and  $f_2 : (\mathbb{R}^{n_1'}, \mathbb{R}^{n_2}, \mathbb{R}^p) \rightarrow \mathbb{R}^{n_2}$  represent the plant and actuator dynamics, respectively.  $g : \mathbb{R}^{n_2} \rightarrow \mathbb{R}$  represents the algebraic relationship between inner loop states and the virtual control, whereas  $h : \mathbb{R}^{n_1} \rightarrow \mathbb{R}$  represents the algebraic relationship between the outer loop states and the performance output. The key characteristic of this system representation is that the real control inputs,  $u$ , affect the plant states only through the virtual control input,  $v$ , and this virtual

control input is chosen to have the same dimension as the output,  $y$  (in this case, that dimension is 1). Therefore, the introduction of  $v$  splits the system description into an actuator dynamics subsystem containing all of the overactuation and saturation constraints and a square plant dynamics subsystem with no input constraints. This dissertation will be restricted to overactuated systems with a single performance output,  $y$ , also referred to as multi-input, single-output (MISO) systems. However, the control strategies and many of the theoretical results have a direct extension to multi-output systems.

The modular control strategy splits control design into two parts. The first part is an outer-loop controller, which determines a desired virtual control input,  $v_{des}$ . The second part is the inner-loop controller, also called the *control allocation*, which determines real control inputs,  $u$ , that will satisfy  $v = v_{des}$  closely.

This approach is currently used extensively in aerospace, marine, and automotive control design, where the design of inner and outer loop controllers are commonly done by different groups within a company or even by different companies. In these instances, one group designs the inner loop controller without intimate knowledge of the plant and another group to design the outer loop controller without intimate knowledge of the actuators. Additionally, with this approach, actuators may be interchanged without affecting the the plant dynamics. The same outer loop controller is then re-used when the actuators are replaced, upgraded, added, or removed.

### 1.1.2 Static vs. Dynamic Control Allocation

The main objective of the inner loop control allocation is to force  $v$  to track  $v_{des}$  closely and to leverage overactuation in order to do this effectively, while considering other performance criteria and constraints on the system. The method for control allocation can be classified as either *static* or *dynamic*. In static control allocation, for the purpose of inner-loop control design, all of the actuators are assumed to have an *immediate* effect on the virtual control input. This is the design method for most existing literature on overactuated systems, including [5]-[12] and [17]-[20]. Because the control inputs commands are designed with the assumption of an immediate effect on the virtual control input, it is not possible to take into consideration the fact that some actuators may, in reality, possess faster dynamic authority than others. Thus, while static control allocation is a well-established technique, its applications are limited to those in which there exists a clear bandwidth separation between the actuator dynamics and the desired closed-loop performance of the overall system.

In dynamic control allocation [13]-[16],[21], the controller design does account for

non-negligible actuator dynamics. While the strategies of [13] and [21] propose closed form control allocation schemes, [14]-[16] propose an optimization-based approach to dynamic control allocation, using *model predictive control allocation* (MPCA) on a linear flight control system (a re-entry vehicle) to optimize control inputs over a receding horizon. The use of MPCA enables the designer to impose a constraint in the optimization that explicitly accounts for hard actuator saturation limits, while the optimization over a receding horizon allows for consideration of the different dynamic actuator authorities. The authors of [14]-[16] show through simulation results that their MPCA strategy does indeed outperform static control allocation in terms of the ability to track angular velocity setpoints. This strategy is extended to a nonlinear system in the work by the author of this dissertation in [26]. [27]- [29] utilize MPC for inner loop control, and, although their work does not focus on overactuated systems, the modular framework employed in these works makes them valuable to the body of literature considered in this dissertation.

This dissertation will focus entirely on *modular* control strategies that employ *dynamic* control allocation. While dynamic control allocation combines two desirable properties, namely the ability to deal with significant actuator dynamics and modularity, it is the least understood method for controlling overactuated systems, in terms of the ability to prove rigorous mathematical properties. The following subsection describes the primary challenges in implementing modular control strategies with dynamic control allocation. Each of these challenges relates to either the modular control aspect or the overactuation aspect of the control design problem.

### 1.1.3 Open Modular Control Challenges

Several challenges exist with regard to modular control design, each of which has been only partially addressed, at best, in the literature to-date. These challenges include:

- *Guaranteeing Overall System Stability with Limited Information Exchange Between Inner and Outer Loops* - The ultimate goal is to guarantee input-to-state stability (ISS) from the setpoint ( $r$ ) to states  $x_1$  and  $x_2$ . While standard tools can be used to guarantee stability of the individual outer loop (plant and outer loop controller) and inner loop (actuator dynamics and inner loop controller) systems, challenges arise when designers are tasked to guarantee stability of the interconnected systems. Tools for verifying stability of the interconnection, such as the small gain condition, are too conservative under normal circumstances to guarantee stability of the interconnected systems, as is detailed in [26] (and

in Chapter 3 of this dissertation). Two design strategies have been proposed in the literature in an attempt to circumvent this problem, each with its benefits and pitfalls. In the first strategy, used in [21] and [15]-[16], the inner loop makes use of the desired virtual control input,  $v_{des}$ , and its derivatives (future values of  $v_{des}$  in the discrete time equivalent formulation) to compute at every instant a target inner loop state,  $x_2^t$ . The inner loop controller is designed to stabilize the origin of  $x_2 - x_2^t$ . The pitfall here is that the control strategy only works when the inner loop is provided with accurate future values of  $v_{des}$ . The second strategy, detailed in [27]-[29], employs a multi-rate strategy, in which the inner loop control is updated at least twice as frequently as the outer loop controller, thus allowing the inner loop several sampling instances to produce  $v_{des}$ . This makes guaranteeing stability possible but requires the outer loop to be updated sufficiently slowly relative to the inner loop for successful implementation.

- *Quantifying the Performance Gap Between the Optimal Modular and Centralized Control Designs* - Prior to pursuing a modular control strategy, it is of great interest to derive a bound on the performance gap between the modular control system performance and that of its centralized counterpart (without going through the process of designing a centralized controller). One can use this knowledge to decide prior to any control design work whether a modular or centralized approach is more appropriate for that application. To the author’s knowledge, no such result existed (prior to this work) in the literature for the particular modular control scheme depicted in Figure 1.1.
- *Performance Enhancement Once the Inner and Outer Loops are Integrated* - Typically, an overall performance metric exists for the system, and while the inner and outer loop controllers will be designed in parallel to meet their individual performance specifications, the overall system, once these controllers have been combined, may fall short of the overall desired performance. In this case, it is desirable to explore the possibility of designing an add-on compensator that will help to recover the desired performance of the modular control system. To the author’s knowledge, no such compensator existed in the literature prior to this work.

#### 1.1.4 Open Challenges Relating to Dynamic Control Allocation (Through MPCA)

The control allocation literature has progressed from modular, but static, control allocation, to MPCA for dynamic control allocation. MPCA has shown to be a leading inner loop control allocation method for systems with saturation constraints and significant actuator dynamics; thus, this dissertation will focus on MPCA as our control allocation “method of choice.” However, while the literature on MPCA ([14]-[16]) has resulted in significant advances, there are still several open issues:

- *Generalization to Nonlinear Actuator Dynamics* - The current literature on MPCA is applied to the special case of linear actuator dynamics.
- *Implementation and Experimental Validation* - While the current literature on dynamic control allocation provides encouraging simulation results, the strategy has not been implemented and experimentally verified. Traditionally, model predictive control (MPC) has been used in many process control applications, with slow dynamics and closed-loop requirements on the order of minutes or hours, and, more recently, as part of the outer loop in an inner-loop/outer-loop control strategy where the outer loop serves as a trajectory planner. Here, this dissertation is proposing to apply MPC to the *inner* loop of the system, where transient requirements may be on the order of seconds or fractions of a second, and computational efficiency must be addressed.
- *Deriving MPCA Constraints that Guarantee Overall System Stability* - Finally, there is no stability guarantee for the overall system with MPCA ([16] does provide a methodology to guarantee stability of the inner loop alone, for the linear case, by predetermining a target state trajectory and then stabilizing the error system).

## 1.2 Dissertation Contributions

This dissertation aims to tackle each of the open challenges of the previous section. For each challenge, the results of this dissertation either resolve the challenge or move the state of the art forward.

A central contribution of this research, which makes the other contributions possible, is a *reference model* based modular control design approach. This approach, depicted in Figure 1.2, provides a framework to tackle the open issues of the modular control design. Specifically, the approach divides the main control design into three steps:

1. Define a reference model,  $F$ , that represents a desirable and realistic performance target for the inner closed loop.
2. Design the outer loop controller with the assumption that the inner loop behavior matches that which is specified by  $F$ .
3. For the inner loop control design, instead of focusing on driving  $v$  to  $v_{des}$ , design the inner closed loop to minimize the error between  $v$  and the output of  $F$  (with input  $v_{des}$ ), which will be referred to as  $v_{des}^f$ .

The first step comes from negotiation between the inner and outer loop designers and must precede the second and third steps, whereas the second and third steps

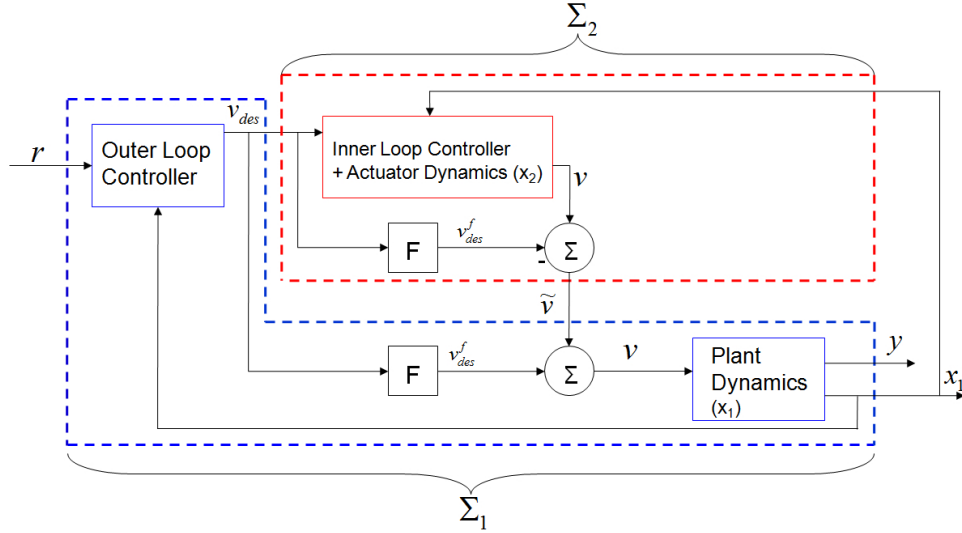


Figure 1.2: Diagram of the system (1.1) recast for the reference model based design framework. The figure reflects the distribution of design tasks (separated by dashed lines) between the inner and outer loop designs, as well as an analysis framework that is used to yield important stability and performance results.

may be conducted in parallel, by different designers. By negotiating this inner loop reference model in the preliminary design stage of control system development (prior to the design of the outer loop and inner loop controllers), the approach gives the designer of the outer loop controller a realistic performance estimate for the overall system as well as a mechanism for verifying the overall stability of the system using the small gain condition. This design framework decomposes the system into two interconnected subsystems,  $\Sigma_1$  and  $\Sigma_2$ , as shown in Figure 1.2.

The results of this dissertation will show that, in terms of resolving several open issues, the incorporation of an inner loop reference model into the control design represents an attractive alternative to the target calculation and multi-rate techniques proposed in earlier literature ([15]-[16], [21], and [27]-[29]).

### 1.2.1 Contributions to Modular Control Design Theory

The dissertation contributions that are related to the modular control design process are summarized as follows:

- *Developed tools for designing modular control systems with guaranteed stability* - This dissertation derives conditions on the reference model under which it is possible to verify overall system stability via the small gain condition, without the inner loop requiring future information regarding  $v_{des}$ .



- *Quantified the performance gap between the optimal modular and centralized control designs* - The dissertation derives conditions under which the reference model,  $F$ , can be matched exactly, as well as conditions under which the modular control performance can be made equal to its centralized counterpart. The dissertation presents a modular vs. centralized performance inequality that provides a meaningful upper bound for the modular vs. centralized performance gap without the need to design both the modular and centralized control systems. The analytical results on this topic are limited in scope to linear systems.
- *Developed a mechanism for enhancing overall system performance once the inner and outer loops have been integrated* - The dissertation proposes a novel add-on mechanism, referred to as modular control error compensation (MCEC), that recovers overall system performance once the outer and inner loop controllers have been integrated. It is shown that this compensator can be designed using  $\mu$  synthesis in order to optimize performance recovery based on two concrete metrics. The analytical results on this topic are limited to linear systems.

### 1.2.2 Contribution to Model Predictive Control Allocation

The theoretical dissertation contribution that relates to the use of MPCA for the inner loop control allocation, is given as follows:

- *Completed significant ground work in deriving MPCA constraints that will guarantee stability* - This dissertation provides a mechanism, using terminal constraint sets and control laws, for guaranteeing stability of the overall system, when MPCA is used for inner loop control and the setpoint is held constant. We also identified several challenges for the derivation of feasible, tractable, MPCA constraints, which will serve as a topic of future work.

### 1.2.3 Experimental Contributions on a Thermal Management System

In addition to advancing the theory for modular control of overactuated systems, this dissertation also validates these control strategies experimentally, resulting in the following experimental contributions:

- *Developed and validated a dynamic model for the thermal management test system* - Early in this dissertation work, a dynamic model for the thermal management system was developed and validated experimentally. The presence of a high-fidelity model of the system was essential for the ensuing dissertation contributions, which all rely on model based control design.

- *Experimentally validated modular, closed-form control with MCEC* - This dissertation demonstrates the effectiveness of the modular control design process, with MCEC, on the automotive thermal management system.
- *Generalized MPC on nonlinear actuator dynamics and implemented and experimentally validated MPC on a thermal management test bed* - MPC (with an inner loop reference model) is applied to the thermal management system detailed in Chapter 2 (with nonlinear actuator dynamics). This dissertation presents design, real-time simulation, and experimental results.

Finally, the dissertation presents experimental results that show that MCEC can be combined with MPC to yield desirable overall system performance. While the derivation of analytical performance results for combined MPC and MCEC remains an open challenge at this point, the results on the thermal management test bed provide evidence that the area of combined MPC and MCEC represents a promising and interesting avenue for future study.

### 1.3 Dissertation Outline

In order to provide the reader with both a practical and theoretical understanding of each of the dissertation contributions, Chapter 2 introduces a thermal management system that will be used subsequently as a test bed for illustrating the principles and implications of the theoretical results. As will be shown in Chapter 2, the thermal management system is overactuated and represents an excellent candidate for modular control for several reasons. Chapter 3 discusses the contributions to modular control in detail, whereas Chapter 4 focuses on the contributions to MPC. Chapter 5 examines experimental results where MCEC is combined with MPC. Finally, Chapter 6 presents conclusions and discusses opportunities for future work.

## Chapter 2

# Engine Thermal Management: A Test Bed for Control Development

In this chapter, an engine thermal management system is introduced, which possesses many features that are relevant to this dissertation and is therefore used as a test bed for studying the control strategies that are proposed in this dissertation. The system is depicted schematically in Figure 2.1 and pictorially in Figure 2.2, and is used in an automotive test cell to control coolant and oil temperatures at the outlet of an automotive engine. Here, a heat exchanger, heater, and mixing valve each reside in a *fluid conditioning unit* (one for coolant and another for oil) which is connected via flexible hose to the engine block. The mixing valve controls the distribution of flow through the heating and cooling circuits, whereas the heater is capable of providing supplemental heating, which becomes necessary when the engine itself does not provide enough heat rejection to bring the engine outlet temperature to its desired value.

The thermal management system serves an important practical purpose, since the tight control of coolant and oil temperatures facilitates and improves the process of engine mapping and calibration. Engine mapping refers to the process where models characterizing the relations between the engine inputs (e.g. spark timing, air charge, fuel, valve timing, etc., for gasoline engines) and outputs (e.g. torque, exhaust temperature, emissions, etc.) are developed and validated. A controlled environment such as an engine test cell with a dynamometer has been used for mapping to ensure that these models are accurate. Off-the-shelf temperature control strategies (for example, the baseline control strategy presented in [25]) typically do not focus on optimally leveraging both actuators in the system, but rather assume that one actuator exists for heating and the other exists for cooling, in spite of the stark contrast in the dynamic authorities of each. This has typically resulted in temperature control systems with slow responses and poor overall performance, leading either to deterio-

rated engine modeling conditions or an excessive time spent waiting for the system to settle.

In addition to posing a challenging and important control design problem, the system represents an interesting and accessible test bed that embodies both of the characteristics that we are interested in investigating in this dissertation work, specifically:

1. *Modularity* - Because the contents of the fluid conditioning unit, which contains both actuators, are separated from the engine itself, the thermal management system represents a strong candidate for modular control, taking the mixing valve outlet temperature as the virtual control input. Furthermore, the fluid conditioning units are manufactured and assembled by a different company than the engine itself. Therefore, it is of great interest to be able to design a separate inner loop controller to regulate the temperature of the fluid exiting the fluid conditioning unit and outer loop controller for controlling the fluid temperature at the engine outlet.
2. *Overactuation* - Each of the circuits (coolant and oil) has two actuators (a heater and mixing valve) for one performance variable (engine outlet temperature). Moreover, both actuators are subject to hard saturation limits, and saturation constraints are commonly active during the course of operation. Finally, the mixing valve acts as a faster degree of control authority than the heater, so the system presents a challenge of blending two actuators with different dynamic authorities.

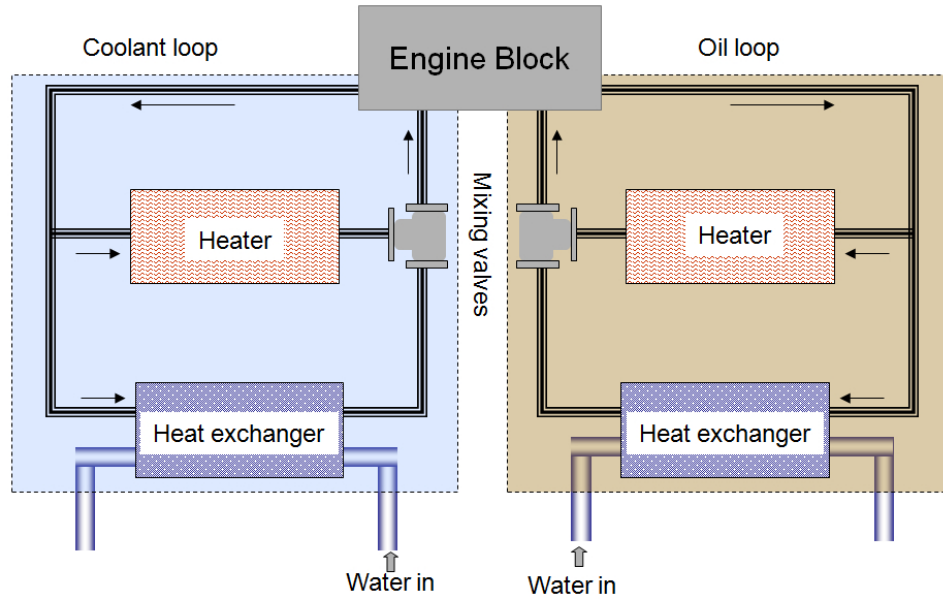


Figure 2.1: Thermal management system schematic.

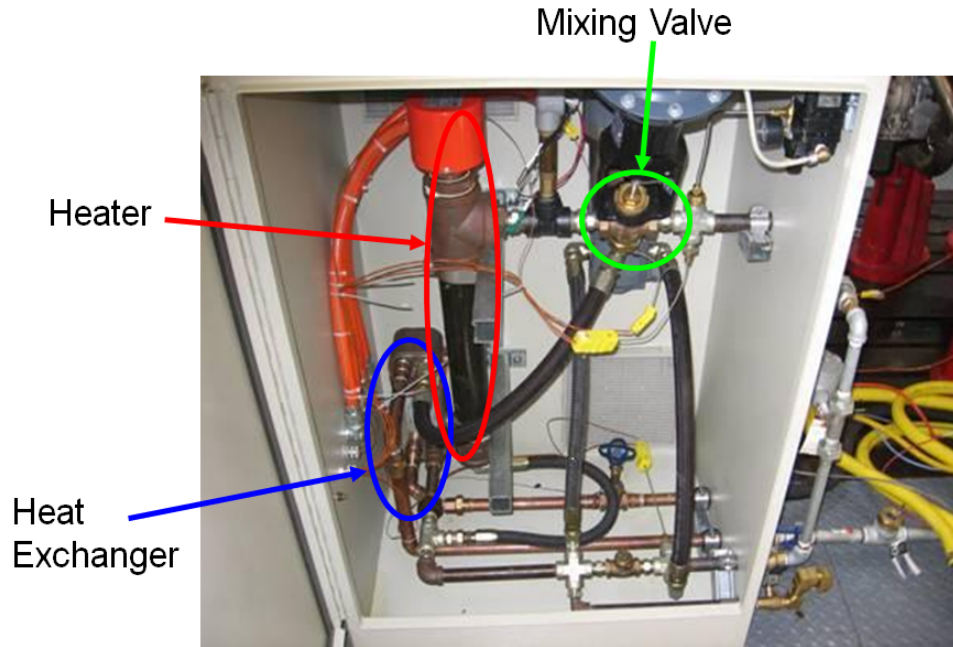


Figure 2.2: Photograph of the fluid conditioning unit for the oil system (a very similar unit exists for coolant, and both are connected to the engine block via flexible hose).

Important characteristics of the thermal management system were identified through an extensive modeling study, presented in [23]-[25], which will be summarized in this chapter. Here, the system identification setup used for modeling and the rapid prototyping setup used for control validation are presented. This system will be used in Chapters 3, 4, and 5 to illustrate important results and their applications.

## 2.1 Experimental Setup for System Identification and Rapid Controller Prototyping

For system identification, the thermal management system was instrumented as in Figure 2.3, with thermocouples at every major component outlet possible, and with flow rate sensors for the working fluid (coolant or oil) and cooling water passing through each circuit. The schematic in Figure 2.3 is given for a single circuit (oil, in this case), but an identical instrumentation is used for the other circuit as well. The large number of thermocouple measurements made it possible to decompose the full system into several simpler systems (mixing valve, heat exchanger, heater, and engine) for modeling and system identification (a detailed description of the system identification procedure is given in Appendix A), which made the process considerably less burdensome. These thermocouples remained in place for the experimental

validation of control strategies, allowing for the use of state feedback.

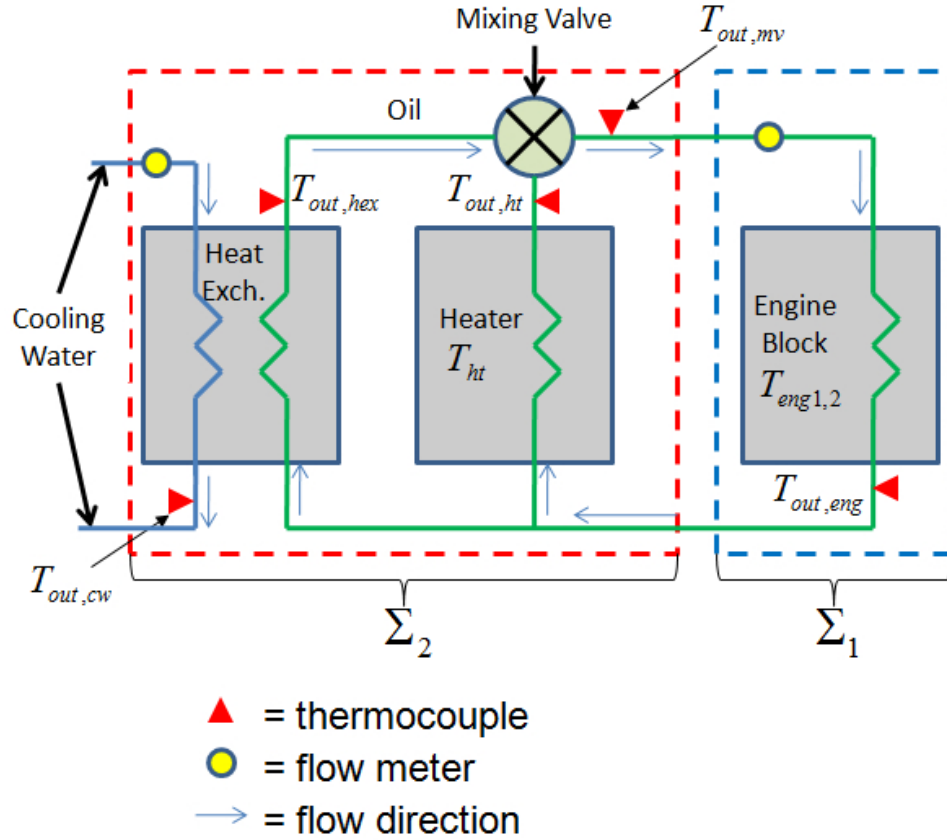


Figure 2.3: Schematic of the oil system, which will be the only system considered in our experimental validation.

In the experimental setup, depicted in Figure 2.4, the controller and hardware interface are designed using MATLAB Real Time Workshop<sup>TM</sup> and xPC Target<sup>TM</sup>. The configuration relies on a host PC where all of the control design takes place, and a target PC, which executes compiled C code and transmits/receives signals through two rapid prototyping boards. One board, the Measurement Computing PCI-DAS-TC<sup>TM</sup> board, handles all of the thermocouple measurements (with locations depicted in Figure 2.3), whereas the other board, the Quanser Q4<sup>TM</sup>, handles the analog inputs (flow rate measurements, with locations depicted in Figure 2.3) and outputs (mixing valve and heater commands).

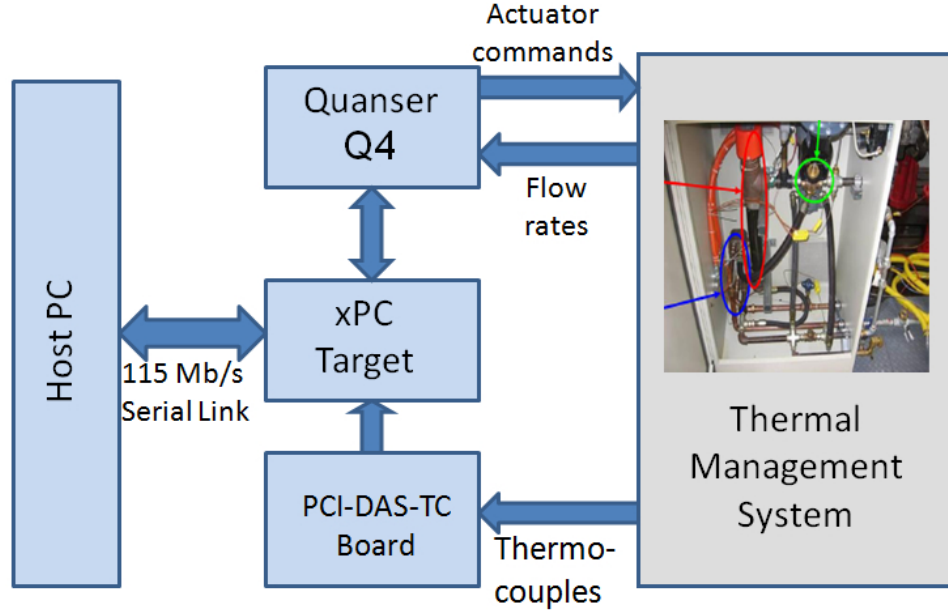


Figure 2.4: Thermal management rapid prototyping configuration

## 2.2 Dynamic Model

The thermal management model includes several components, consisting specifically of a mixing valve, heat exchanger, heater, and engine. The notation used to describe the inputs, states, and parameters of this model is provided in Tables 2.1-2.3. The experimental results obtained in Chapters 3-5 of this dissertation consider set-point tracking, with the disturbances held constant, so that they may be viewed as parameters in the model.

Note that for the mixing valve, heat exchanger, and heater, a separate set of these

Table 2.1: Model inputs

Input	Description	Control/disturbance
$\phi$	Mixing valve input (ratio of flow through heater to total flow)	Control
$\dot{Q}_{ht}$	Heater power input ( $kW$ )	Control
$\dot{Q}_{eng1,2}$	Engine heat rejection rate ( $kW$ )	Disturbance
$W$	Flow rate of coolant or oil ( $kg/s$ )	Disturbance
$W_{cw}$	Flow rate of cooling water ( $kg/s$ )	Disturbance

Table 2.2: Model states

Subsystem	State	Description
Mixing Valve	$T_{out,mv}$	Mixing valve outlet temperature
Heat Exchanger	$T_{out,hex}$	Hot side outlet temperature
	$T_{out,cw}$	Cooling water outlet temperature
Heater	$T_{out,ht}$	Heater outlet temperature (fluid)
	$T_{ht}$	Heater temperature (solid coil)
Engine	$T_{out,eng}^c$	Coolant temperature at engine outlet
	$T_{out,eng}^o$	Oil temperature at engine outlet
	$T_{eng1}$	Engine block temperature in the vicinity of coolant
	$T_{eng2}$	Engine block temperature in the vicinity of oil

states and parameters exists for both coolant and oil.

Having introduced the basic notation for the thermal management system, the detailed dynamic model is now considered. For each loop, the mixing valve, heater, and heat exchanger are considered as subsystems, using thermodynamic principles detailed in [30] along with guidance from [31]-[39] for the derivation of the dynamic model. The engine, through which both loops circulate, is modeled as a heat source, and the thermal inertia of the engine block is taken into account by the model. Figure 2.5 serves as a visual reference that shows how each of the subsystems are interconnected, in terms of the specific signals in the thermal management system. The shading around the mixing valve, heat exchanger, and heater blocks is intended to indicate that separate components exist for both coolant and oil circuits.

Without significant compromise of model accuracy, the following assumptions are made:

- A1.** Both coolant and oil have constant density and specific heat.
- A2.** The heaters and heat exchangers assume lumped parameters, with no energy storage in the heat exchanger “core.”
- A3.** The heater model assumes a single temperature state for the heater coil, allowing for energy storage in the coil.
- A4.** The engine model assumes two states for engine block temperature, reflecting different temperature distribution and heat rejection dynamics across the engine block.



Table 2.3: Model parameters

Subsystem	Parameters	Description
Mixing Valve	$\tau_t$	Mixing valve thermal time constant (s)
Heat Exchanger	$C$	Specific heat of working fluid (coolant or oil) ( $kJ/kg - K$ )
	$C_{cw}$	Specific heat of cooling water ( $kJ/kg - K$ )
	$m_{hot}$	Mass of fluid - hot side of heat exchanger ( $kg$ )
	$m_{cold}$	Mass of fluid - cold side of heat exchanger ( $kg$ )
	$hA_{hot}$	Convection coefficient times interaction area - hot side of heat exchanger ( $(kW/K)(s/kg)^{0.8}$ )
	$hA_{cold}$	Convection coefficient times interaction area - cold side of heat exchanger ( $(kW/K)(s/kg)^{0.8}$ )
	$kA_{hot}$	Conduction coefficient times interaction area - hot side of heat exchanger ( $kW/K$ )
	$kA_{cold}$	Conduction coefficient times interaction area - cold side of heat exchanger ( $kW/K$ )
Heater	$C$	Specific heat of the working fluid (coolant or oil) ( $kJ/kg - K$ )
	$m_{fluid}$	Mass of the fluid ( $kg$ )
	$mC_{ht}$	Mass times specific heat of the heater <sup>1</sup> ( $kJ/K$ )
	$hA_{ht}$	Convection coefficient times interaction area for the heater ( $(kW/K)(s/kg)^{0.8}$ )
	$kA_{ht}$	Conduction coefficient times interaction area for the heater ( $kW/K$ )
Engine	$C^{c/o}$	Specific heat of the fluid ( $kJ/kg - K$ )
	$m^{c/o}$	Mass of the fluid within engine block ( $kg$ )
	$mC_{eng1,2}$	Mass times specific heat of the engine block in the vicinity of coolant (1) and oil (2) ( $kJ/K$ )
	$hA^{c/o}$	Convection coefficient times interaction area betw. the fluid and engine block ( $(kW/K)(s/kg)^{0.8}$ )
	$kA^{c/o}$	Conduction coefficient times interaction area betw. the fluid and engine block ( $kW/K$ )
	$kA_{cp}$	Conductive thermal coupling within the engine block itself ( $kW/K$ )

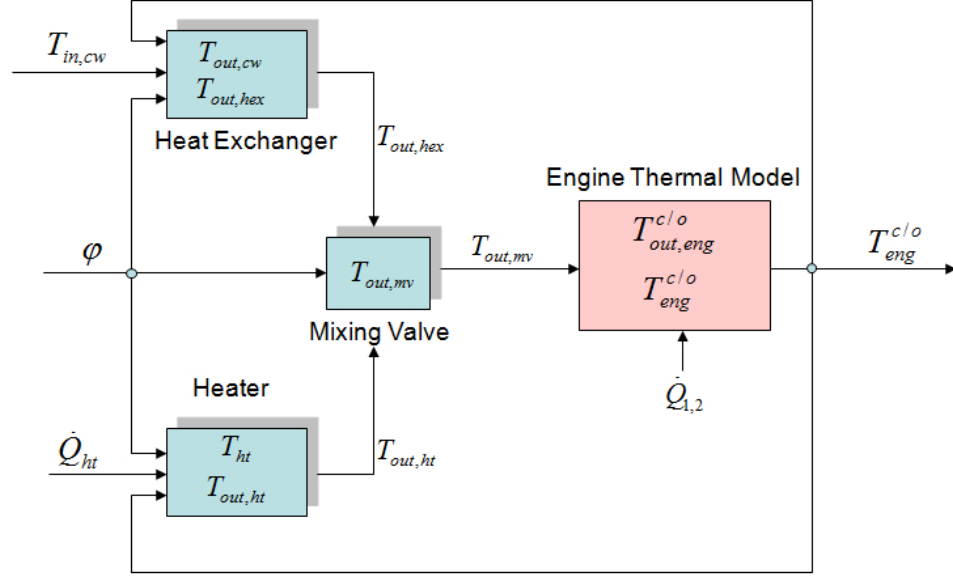


Figure 2.5: Diagram of the signal flow between all of the interconnected subsystems in the engine thermal management system

**A5.** The mixing valve assumes adiabatic mixing.

**A6.** Heat losses in pipes are neglected.

**Remark 2.2.1.** Assumption (A1) is valid throughout the operating window of the system, where the temperatures of the fluids are significantly less than their boiling points. Experimental validation has shown that other assumptions have negligible impact on the modeling results.

Taking  $\phi$  as the input to the mixing valve, as in Table 2.1, it is possible to relate total flow rate (of coolant or oil) to flow rate through the heater and heat exchanger as:

$$W_{ht} = \phi W \quad (2.1)$$

$$W_{hex} = (1 - \phi)W. \quad (2.2)$$

By conservation of energy, and assuming a mixing valve time constant,  $\tau_t$ , which is related to the amount of time required to impart a temperature change at the valve outlet, the mixing valve temperature dynamics are then given by:

$$\dot{T}_{out,mv} = \frac{1}{\tau_t}(\phi T_{out,ht} + (1 - \phi)T_{out,hex}). \quad (2.3)$$

The heat exchanger model contains temperature states for both the hot and cold flows. Heat is transferred across the heat exchanger “core” according to a logarithmic mean temperature difference law [31], yielding the equations:

$$\dot{T}_{out,hex} = \frac{1}{m_{hot}}((1 - \phi)W(T_{out,eng} - T_{out,hex}) - \frac{UA}{C}LMTD), \quad (2.4)$$

$$\dot{T}_{out,cw} = \frac{1}{m_{cold}}(W_{cw}(T_{in,cw} - T_{out,cw}) + \frac{UA}{C_{cw}}LMTD), \quad (2.5)$$

where:

- $UA$  is the overall heat transfer coefficient times the contact area - it is dependent on flow rate and may be expressed as:

$$UA = \frac{(hA_{hot}W_{hex}^{0.8} + kA_{hex})(hA_{cold}W_{cw}^{0.8} + kA_{cold})}{hA_{hot}W_{hex}^{0.8} + kA_{hot} + hA_{cold}W_{cw}^{0.8} + kA_{cold}}, \quad (2.6)$$

where  $hA_{hot/cold}$  and  $kA_{hot/cold}$  are parameters to be identified <sup>2</sup>.

- $LMTD$  is the logarithmic mean temperature difference between the hot and cold sides [31]. For a counterflow configuration, used in this system, it is defined as:

$$LMTD = \frac{(T_{out,eng} - T_{out,cw}) - (T_{out,hex} - T_{in,cw})}{\ln \frac{T_{out,eng} - T_{out,cw}}{T_{out,hex} - T_{in,cw}}}. \quad (2.7)$$

To facilitate an understanding of the heat exchanger, the counterflow arrangement of the heat exchanger (which is what is used in this application) is shown in Figure 2.6. While the LMTD is valid for both counterflow and co-flow heat exchangers, the locations in which each temperature appears in the equation is different for each arrangement. The LMTD is derived under the assumption that the heat exchanger “core” (the solid shell through which heat is transferred) has zero (minimal) thermal capacity itself and therefore can only pass fluid from one side to another without storing it in the core.

The heater model consists of a fluid temperature state and a coil temperature state, where:

---

<sup>2</sup>The power of 0.8 in the expression for convective heat transfer arises from established empirical results [30].

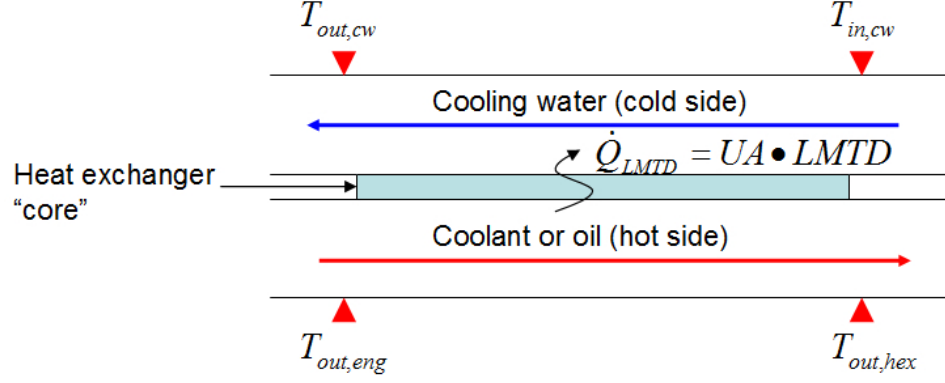


Figure 2.6: Heat exchanger schematic.

$$\dot{T}_{out,ht} = \frac{1}{m_{fluid}}(\phi W(T_{out,eng} - T_{out,ht}) \quad (2.8)$$

$$+ (\frac{hA_{ht}W_{ht}^{\cdot 8} + kA_{ht}}{C})(T_{ht} - T_{out,ht})),$$

$$\dot{T}_{ht} = \frac{1}{mC_{ht}}(\dot{Q}_{ht} - (hA_{ht}W_{ht}^{\cdot 8} + kA_{ht})(T_{ht} - T_{out,ht})). \quad (2.9)$$

In order to arrive at a representative engine model, two states are used to represent engine block temperatures, reflecting the different temperatures in the neighborhood of the coolant and oil loops circulating through the engine block. This results in the fourth order engine subsystem model:

$$\dot{T}_{out,eng}^c = \frac{1}{m^c}(W^c(T_{out,eng}^c - T_{out,mv}^c) \quad (2.10)$$

$$+ (\frac{hA^c(W^c)^{\cdot 8} + kA^c}{C^c})(T_{eng1} - T_{out,eng}^c)),$$

$$\dot{T}_{out,eng}^o = \frac{1}{m^o}(W^o(T_{out,eng}^o - T_{out,mv}^o) \quad (2.11)$$

$$+ (\frac{hA^o(W^o)^{\cdot 8} + kA^o}{C^o})(T_{eng2} - T_{out,eng}^o)),$$

$$\dot{T}_{eng1} = \frac{1}{mC_{eng1}}(\dot{Q}_1 + kA_{cp}(T_{eng2} - T_{eng1}) \quad (2.12)$$

$$- (hA^c(W^c)^{\cdot 8} + kA^c)(T_{eng1} - T_{out,eng}^c)),$$

$$\dot{T}_{eng2} = \frac{1}{mC_{eng2}}(\dot{Q}_2 + kA_{cp}(T_{eng1} - T_{eng2}) \quad (2.13)$$

$$- (hA^o(W^o)^{\cdot 8} + kA^o)(T_{eng2} - T_{out,eng}^o))$$

where  $T_{eng1,2}$  represent temperatures of the engine block states and  $kA_{cp}$  represents the heat transfer within the engine itself, due to the temperature difference between the different regions of the engine block.

The various model parameters were determined through a system identification procedure that is detailed in [25]. The resulting model was validated experimentally, with results presented in Figure 2.7, which show good correlation between the model and data.

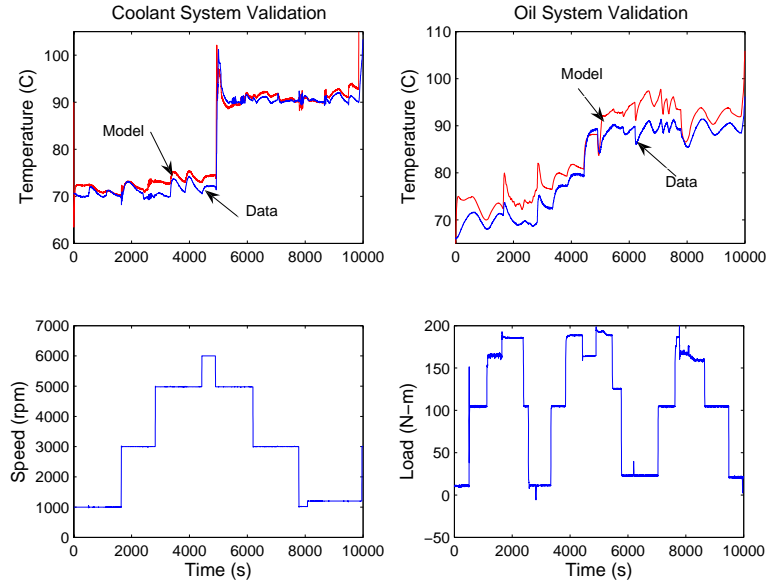


Figure 2.7: Model validation results.

## 2.3 Model Based Dynamic Analysis

Analysis of the validated model provides insights into the system dynamics that have led to an improved control design. Specifically, this analysis will show, based on the model, that:

1. The coolant and oil loops have negligible interaction at typical operating conditions, and subsequent design and analysis will consider one loop at a time, with  $T_{out,eng}$  held at setpoint for the other loop.
2. The two actuators possess very different dynamic control authorities in influencing  $T_{out,eng}$ . The mixing valve is much faster to affect the outlet temperature than the heater.

The coupling of the two loops is shown in Figure 2.8, where the Bode plots are presented for the transfer functions relating the engine inlet and outlet temperatures (a linear subsystem at constant flow rates), at typical flow rates of 5 kg/s for coolant and 0.5 kg/s for oil:

$$\begin{bmatrix} T_{out}^c \\ T_{out}^o \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \begin{bmatrix} T_{in}^c \\ T_{in}^o \end{bmatrix}. \quad (2.14)$$

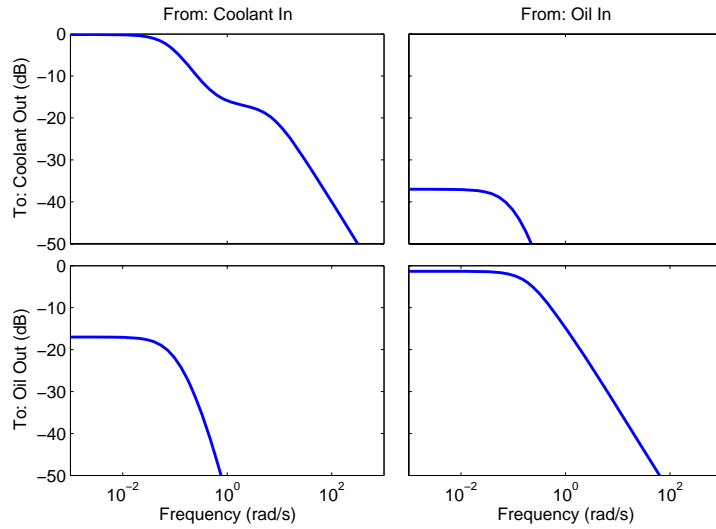


Figure 2.8: Extent of loop decoupling.

The relative magnitudes of the transfer functions on the diagonal are at least 5 times as large as the off-diagonal entries, indicating that one fluid has little thermodynamic influence on the other. Also noteworthy is the fact that the oil temperature has less effect on the coolant than vice versa. This is because the coolant loop is the dominant loop in the system, occupying a much larger volume in the engine block than the oil loop. Similar results are obtained for different operating conditions.

As a result of the decoupling analysis, and due to the situation in the test cell at the time that experimental results were obtained, the experiments presented in this dissertation are for the *oil loop only* (we would typically have a slight preference for the coolant loop, but this was operable under a different configuration at the time of testing).

With regard to the second analysis result, Figure 2.9 shows, for the coolant loop, that the transfer function between the mixing valve input and engine outlet tem-

perature has higher gain at high frequencies than that from the heater for all the actuator settings considered, suggesting that the valve possesses more transient control capability. The control inputs are normalized for the Bode plot in order to be comparable. Therefore, to reserve sufficient transient control authority, it is critical to avoid operating the mixing valve near its saturation limit, and proper setting of the heater input may help to accomplish this goal. It is also noteworthy that the system linearizations vary widely depending on the actuator settings, and this variation should be taken into account when choosing where to operate the mixing valve.

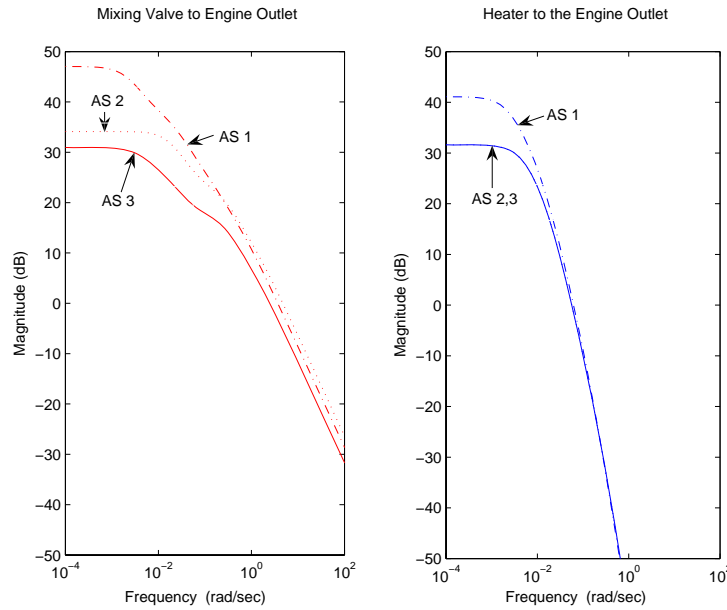


Figure 2.9: Comparison of transfer functions between actuator inputs and outlet temperature at various actuator settings (AS's): AS1 -  $\dot{Q}_{ht} = 0$  kW,  $\phi = 0.5$ ; AS2 -  $\dot{Q}_{ht} = 4.5$  kW,  $\phi = 0.5$ ; AS3 -  $\dot{Q}_{ht} = 0$  kW,  $\phi = 0.9$ .

The trends depicted in Figure 2.9 explain the difficulties that had been encountered by engineers who had tried to use the heater as one of the control inputs using a closed form feedback law. Chapter 4 will show how MPCA can be used to incorporate the use of both actuators, recognizing the differences in the dynamic authorities of each.

# Chapter 3

## Modular Control Design - Stability and Performance

The objectives of this chapter are:

1. Establishing important stability and performance properties for the reference model based modular approach;
2. Developing analytical tools to address performance issues that were previously unaddressed in the literature.

In order to achieve these goals, this chapter first formalizes the treatment of the reference model based approach and relates our notation to alternative approaches proposed in the literature. First, however, a few mathematical preliminaries must be established.

### 3.1 The Notion of Relative Degree

Prior to our formal treatment of stability and performance under the reference model based framework, it is essential to establish an understanding of the concept of *relative degree* of a system, which needs to be defined for both the vector and scalar case. For nonlinear systems, the following scalar definition of relative degree is adopted [40]:

**Definition 3.1.1.** *A continuous-time system,  $\Sigma$ , with  $p$  inputs,  $u$ , and  $q$  outputs,  $v$ , and  $n$  states,  $x$ , has (strict) relative degree  $\rho$ , well-defined everywhere, from  $u$  to  $v$  if the following hold:*

1.  $\forall i < \rho, \forall x \in \mathbb{R}^n, \frac{\partial v^{(i)}}{\partial u} = 0_{p \times q}$
2.  $\forall x \in \mathbb{R}^n, \forall u \in \mathbb{R}^p, \frac{\partial v^{(\rho)}}{\partial u} \neq 0_{p \times q}$ .

where  $v^{(i)}$  denotes the  $i^{\text{th}}$  derivative of  $v$ .



**Definition 3.1.2.** *A discrete-time system,  $\Sigma$ , with  $p$  inputs,  $u$ , and  $q$  outputs,  $v$ , and  $n$  states,  $x$ , has (strict) relative degree  $\rho$  from  $u$  to  $v$ , well-defined everywhere, if the following hold:*

1.  $\forall i < \rho, \forall x(k) \in \mathbb{R}^n, \frac{\partial v(k+i)}{\partial u(k)} = 0_{p \times q}$
2.  $\forall x(k) \in \mathbb{R}^n, \forall u(k) \in \mathbb{R}^p, \frac{\partial v(k+\rho)}{\partial u(k)} \neq 0_{p \times q}$ .

Note that this dissertation does not consider nonlinear systems whose relative degrees are not the same everywhere (for all  $x$ ); hence, we assume that for the nonlinear systems encountered here, the relative degree is given by  $\rho$ , and is well-defined everywhere.

For linear systems, for which some of our results are specialized, Definitions 3.1.1 and 3.1.2 are equivalent to the following definition of relative degree:

**Definition 3.1.3.** *A  $p \times q$  transfer function matrix,  $H(s)$ , has relative degree  $\rho(H(s)) \triangleq \min \rho(H_{i,j}(s)), 1 \leq i \leq p, 1 \leq j \leq q$ , where  $\rho(H_{i,j}(s))$  is the relative degree of a given constituent transfer function, defined in the usual way as the difference between the degree of the numerator and degree of the denominator.*

Note that the definition of relative degree for discrete-time linear systems is analogous to the continuous-time definition, simply replacing  $s$  with  $z$ .

Ultimately, the continuous-time definition of relative degree provides the number of times that one must differentiate the system output prior to the input appearing. The discrete-time definition provides the number of steps it takes for the input ( $u$ ) to affect the output ( $v$ ). For effective reference model based control, it will be essential to choose a reference model that gives the actuator dynamics enough time to produce acceptable behavior through appropriate choice of the control input,  $u$ . The relative degree of the actuator dynamics, compared with the relative degree of the reference model, will prove to be key in achieving (or failing to achieve) the rigorous stability and performance results detailed in this section.

## 3.2 Mathematical Formulation of the Reference Model Based Modular Control Approach

The reference model based modular design approach was introduced conceptually in Chapter 1 and is depicted graphically in Figure 1.2. As a review, the design approach is conducted as follows:

1. Negotiation of the reference model,  $F$ , between the inner and outer loop designers;
2. Outer loop design, assuming that the inner closed loop matches  $F$ ;
3. Inner loop design, in an attempt to yield inner closed-loop performance that matches  $F$ .

The first step must precede the second and third, but the second and third steps may be conducted in any order, or in parallel.

In order to account for the reference model, the closed-loop system (Figure 1.2) is described by the following equations:

$$\Sigma_1 := \begin{cases} \dot{x}_1^c = f_1^c(x_1^c, v_{des}^f, \tilde{v}) \\ v_{des} = c_1(x_1^c, r) \\ x_1' = d(x_1) \\ \dot{x}_{f1} = f_3(x_{f1}, v_{des}) \\ v_{des}^{f1} = g_f(x_{f1}) \end{cases} \quad (3.1)$$

$$\Sigma_2 := \begin{cases} \dot{x}_2^c = f_2^c(x_1', x_2^c, u) \\ u = c_2(x_2^c, v_{des}) \\ v = g(x_2) \\ \tilde{v} = v - v_{des}^{f2} \\ \dot{x}_{f2} = f_3(x_{f2}, v_{des}) \\ v_{des}^{f2} = g_f(x_{f2}) \end{cases} \quad (3.2)$$

where the equivalent discrete-time representation is given by:

$$\Sigma_1 : \begin{cases} x_1^c(k+1) = f_1^{dc}(x_1^c(k), v_{des}^f(k), \tilde{v}(k)) \\ v_{des}(k) = c_1(x_1^c(k), r(k)) \\ x_1'(k) = d(x_1(k)) \\ x_{f1}(k+1) = f_3^d(x_{f1}(k), v_{des}(k)) \\ v_{des}^{f1}(k) = g_f(x_{f1}(k)) \end{cases} \quad (3.3)$$

$$\Sigma_2 : \begin{cases} x_2^c(k+1) = f_2^{dc}(x_1^c(k), x_2^c(k), u(k)) \\ u(k) = c_2(x_2^c(k), v_{des}(k)) \\ v(k) = g(x_2(k)) \\ \tilde{v}(k) = v(k) - v_{des}^{f2}(k) \\ x_{f2}(k+1) = f_3^d(x_{f2}(k), v_{des}(k)) \\ v_{des}^{f2}(k) = g_f(x_{f2}(k)) \end{cases} \quad (3.4)$$

In these equations,  $x_1^c$  and  $x_2^c$  represent states of the *closed* inner and outer loop systems, which contain states of the plant and actuator, plus controller states for the outer and inner loop, respectively. Similarly,  $f_{1,2}^c$  and  $f_{1,2}^{cd}$  represent the combined controller and plant or actuator dynamics. Finally,  $x_{f1}$  and  $x_{f2}$  represent the reference model states for the outer and inner loop, respectively, which are governed by the same reference model, specified by  $f_3$  and  $g_f$ .  $c_1$  represents the outer loop control law, whereas  $c_2$  represents the inner loop control law. The chief inner loop objective with the reference model approach is to drive  $\tilde{v}$  to zero, which is accomplished under *reference model matching*. From the perspective of the outer loop,  $\tilde{v}$  enters the system as a disturbance, and the outer loop design can aim to attenuate this disturbance.

We will assume, *without loss of generality*, that:

$$g(0) = 0 \quad (3.5)$$

$$c_1(0, 0) = 0, \quad (3.6)$$

If this assumption is not satisfied in the original system coordinates, a simple coordinate translation can be performed such that it is.

**Remark 3.2.1.** *Note that an identical copy of the reference model resides in both the inner and outer loop. If the initial states for each copy of the reference model are the same, i.e.  $x_{f1}(0) = x_{f2}(0)$ , then  $x_{f1}(t) = x_{f2}(t) =: x_f(t), \forall t > 0$  and  $v_{des}^{f1}(t) = v_{des}^{f2}(t) := v_{des}^f(t), \forall t > 0$ . For our analysis results, it is assumed that this is in fact the case.*

In order to give the reader a mathematical sense of how the alternative frameworks compare in structure to the reference model based framework, this section provides a brief description of them as well. The first, employed in [15]-[16] and [21], relies on future information about  $v_{des}$  to compute a target inner loop state,  $x_2^t$ . Using  $v^{(i)}$  to represent the  $i^{th}$  derivative of  $v$ , this framework is depicted in Figure 3.1.

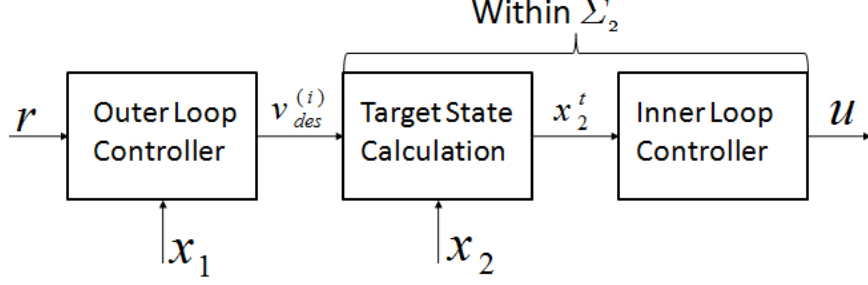


Figure 3.1: Block diagram of a target calculation based approach to modular control design.

The second alternative control strategy is a multi-rate approach, which is employed in [27]-[29]. Like the target state calculation strategy, the multi-rate strategy also does not incorporate a reference model into its design framework; however, the multi-rate strategy does not require the outer loop to make future values of  $v_{des}$  available to the inner loop. Instead, the inner loop is updated at a faster rate than the outer loop, giving the actuators enough time to produce approximately the required  $v_{des}$  signal. The mathematical description, which is exclusively applicable to the discrete-time case, is given as:

$$\Sigma_1 : \begin{cases} x_1^c(k+1) = f_1^{dc}(x_1^c(k), v_{des}(k), \tilde{v}(k)) \\ v_{des}(k) = c_1(x_1^c(k), r(k)) \\ x_1'(k) = d(x_1(k)) \end{cases} \quad (3.7)$$

$$\Sigma_2 : \begin{cases} x_2^c(h+1) = f_2^{dc}(x_1'(k), x_2^c(h), u(h)) \\ u(h) = c_2(x_2^c(h), v_{des}(k)) \\ v(h) = g(x_2(h)) \\ \tilde{v}(h) = v(h) - v_{des}(k) \end{cases} \quad (3.8)$$

where  $h = \nu k$ , and  $\nu > 1$  is an integer.

Ultimately, these alternative techniques are unified in their goal of matching  $v$  to  $v_{des}$ . While precise matching of  $v_{des}$  is desirable, it is unrealistic without information about  $v_{des}$  or a significant degree of time-scale separation between the actuator dynamics and overall closed-loop requirements of the system.

The reference model approach instead seeks to match  $v$  to the output of the reference model,  $v_{des}^f$  by designing the closed-loop behavior of the inner loop to match that of the reference model. The reference model,  $F$ , can be designed to take into account actuator limitations and produce a realistic performance target for

the closed inner loop. Ultimately, the objective with the reference model approach is to achieve *reference model matching*, wherein the inner closed loop satisfies the expression  $V(s) = F(s)V_{des}(s)$ . This will result in  $v$  approaching  $v_{des}^f$  asymptotically, and  $\tilde{v}$  approaching 0 asymptotically.

This chapter will answer the following three sets of questions regarding the modular, reference model based strategy:

1. How can one verify the input-to-state stability (ISS) of the overall system when the outer and inner loop control designers are only aware of the input-output gain of the other system, not its complete description? How does the reference model based approach enhance one's ability to verify stability, compared to the special case of  $F = 1$  and compared to other approaches proposed in the literature?
2. Under what conditions on the actuator dynamics and reference model is precise reference model matching achievable, i.e.,  $V(s) = F(s)V_{des}(s)$ ? Under what conditions on the actuator dynamics can the modular control system perform as well as the centralized control system, and how are these conditions related to the reference model matching conditions? How does the reference model based approach enhance one's ability to make these comparisons? Section 3.4 will quantify a performance metric that will help to answer these questions.
3. What mechanisms can be employed to recover ideal performance (or drive the system performance closer to its ideal level), which is defined as the performance level achieved when the reference model is matched, when the reference model is not matched precisely?

After presenting the answers to these three questions, thus completing the theoretical portion of the chapter, design, simulation, and experimental results will be presented for the thermal management application. The treatment of the thermal management system will proceed through the modular design process with closed-form outer and inner loop controllers. Finally, the thermal management application example will employ a novel performance recovery mechanism, referred to as *modular control error compensation* (MCEC), which serves as the “answer” to the third enumerated question posed above.

### 3.3 Stability Under the Reference Model Based Framework

One notion of stability, applicable to both linear and nonlinear systems, is that of input to state stability (ISS) [41]-[46], which says, qualitatively, that for inputs

whose norm is bounded, the states of the system will also possess bounded norm. Many notions of ISS have been adopted in the literature, including  $L_\infty$  to  $L_\infty$  (the traditional notion),  $L_2$  to  $L_2$  (also referred to as “integral to integral” ISS, which was shown by Sontag to be equivalent to the  $L_\infty$  to  $L_\infty$  notion), and  $L_2$  to  $L_\infty$  (also known as “integral ISS,” a weaker notion than the former). In our stability analysis, which is applicable to both continuous and discrete time representations (and both linear and nonlinear), this dissertation will consider  $L_2$  to  $L_2$  (or  $l_2$  to  $l_2$  for discrete-time) input-to-state stability (ISS) from  $r$  to  $x \triangleq \begin{bmatrix} x_1^{cT} & x_2^{cT} \end{bmatrix}^T$ , which is defined as follows [41]-[42]:

**Definition 3.3.1.** (*Input-to-State Stability*) *The continuous time system with input  $r$  and states  $x$  is ISS if  $\exists$  a class K function  $\beta$  and class  $K_\infty$  functions  $\gamma_a$  and  $\gamma_b$ <sup>1</sup> such that:*

$$\int_0^{t^f} \gamma_a(\|x(t)\|_2) dt \leq \beta(\|x(0)\|_2) + \int_0^{t^f} \gamma_b(\|r(t)\|_2) dt, \forall t^f \geq 0. \quad (3.9)$$

*The discrete time system with input  $r$  and states  $x$  is ISS if  $\exists$  a class K function  $\beta$  and class  $K_\infty$  functions  $\gamma_a$  and  $\gamma_b$  such that:*

$$\sum_{i=0}^n \gamma_a(\|x(i)\|_2) \leq \beta(\|x(0)\|_2) + \sum_{i=0}^n \gamma_b(\|r(i)\|_2), \forall n \geq 0. \quad (3.10)$$

□

Given the nature of the modular control design process, namely, the fact that the inner and outer loop control designs are conducted in parallel, it is of great interest to derive sufficient stability conditions such that one designer does not need to know the specifics of the other loop to verify stability. The well-known small gain condition provides such a set of conditions, enabling either designer (either inner or outer loop) to verify stability with knowledge only of that specific subsystem and the input-output gains of the loops. Given the modular system representation of (3.1-3.2), and from the results derived in [44] and [45] for general interconnected systems (which fit the framework depicted in Figure 1.2), the following proposition yields sufficient conditions for ISS of the composite system:

---

<sup>1</sup>A continuous function,  $g(x) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$  belongs to class  $K_\infty$  iff  $g(0) = 0$  and  $g(x)$  is strictly increasing and unbounded in  $x$ . A continuous function,  $g(x) : [0, a) \rightarrow [0, \infty)$ ,  $a > 0$  belongs to class  $K$  iff  $g(0) = 0$  and  $g(x)$  is strictly increasing (but not necessarily unbounded) in  $x$ .

**Proposition 3.3.1.** (Closed-Loop ISS Conditions with the Modular Inner-Loop/Outer-Loop Controller) Referring to Figure 1.2, suppose that  $\Sigma_1$  is ISS from inputs  $r$  and  $\tilde{v}$  to states  $x_1^c$  and  $\Sigma_2$  is ISS from inputs  $x_1'$  and  $v_{des}$  to states  $x_2^c$ . Also, suppose that there exist class K functions  $\beta_{1,2}^y$  and gains  $\gamma_1^r$ ,  $\gamma_1^y$ , and  $\gamma_2^y$  that satisfy the following small gain condition:

$$\gamma_1^y \gamma_2^y < 1, \quad (3.11)$$

where:

$$\int_0^{t^f} \left\| \begin{bmatrix} v_{des}(t) \\ x_1'(t) \end{bmatrix} \right\|_2^2 dt \leq \beta_1^y \left( \left\| \begin{bmatrix} x_1^c(0) \\ x^f(0) \end{bmatrix} \right\|_2 \right) + \gamma_1^r \int_0^{t^f} r^2(t) dt \quad (3.12)$$

$$+ \gamma_1^y \int_0^{t^f} \tilde{v}^2(t) dt, \forall t^f \geq 0,$$

$$\int_0^{t^f} \tilde{v}^2(t) dt \leq \beta_2^y \left( \left\| \begin{bmatrix} x_2^c(0) \\ x^f(0) \end{bmatrix} \right\|_2 \right) \quad (3.13)$$

$$+ \gamma_2^y \int_0^{t^f} \left( \left\| \begin{bmatrix} v_{des}(t) \\ x_1'(t) \end{bmatrix} \right\|_2 \right)^2 dt, \forall t^f \geq 0,$$

for continuous-time systems and:

$$\sum_{i=0}^n \left\| \begin{bmatrix} v_{des}(i) \\ x_1'(i) \end{bmatrix} \right\|_2^2 \leq \beta_1^y \left( \left\| \begin{bmatrix} x_1^c(0) \\ x^f(0) \end{bmatrix} \right\|_2 \right) + \gamma_1^r \sum_{i=0}^n r^2(i) \quad (3.14)$$

$$+ \gamma_1^y \sum_{i=0}^n \tilde{v}^2(i), \forall n \geq 0,$$

$$\sum_{i=0}^n \tilde{v}^2(i) \leq \beta_2^y \left( \left\| \begin{bmatrix} x_2^c(0) \\ x^f(0) \end{bmatrix} \right\|_2 \right) \quad (3.15)$$

$$+ \gamma_2^y \sum_{i=0}^n \left( \left\| \begin{bmatrix} v_{des}(i) \\ x_1'(i) \end{bmatrix} \right\|_2 \right)^2, \forall n \geq 0.$$

for discrete-time systems. Then the overall system is ISS from  $r$  to  $x$ . □

The proof follows from the one given for generic interconnected systems in [44]. This condition can be extended to one in which class  $K_\infty$  functions, rather than

gains, describe the input-output behavior of the loops, but for this work, such a generalization provides little benefit while complicating the derivations that follow.

The small gain condition given by Proposition 3.3.1 is merely a sufficient condition for stability, and one's choice of reference model,  $F$ , will impact the ability to verify stability using Proposition 3.3.1. In particular, when a reference model is not incorporated into the design, i.e., when one takes  $F = 1$ , it is generally not possible to verify stability using the small gain condition of Proposition 3.3.1. This is illustrated for discrete-time systems through the following proposition:

**Proposition 3.3.2.** (*Impact of Reference Model on Stability Verification - Discrete-time Case*) Consider the discrete-time system described by (3.3) and (3.4), where the relative degree from  $u$  to  $v$  is given by  $\rho_u^v$ , where  $\rho_u^v > 0$ , and suppose that  $F = 1$ . Furthermore, suppose that the outer loop is designed to reject disturbances from  $\tilde{v}$  at steady-state, i.e., the steady-state gain from  $\tilde{v}$  to  $v_{des}$  is equal to 1, thus  $\gamma_1 \geq 1$ . Then  $\gamma_1 \gamma_2 \geq 1$ .

□

**Proof.** Recognizing that there are no reference model states, since  $F = 1$ , the inner loop input-output relationship for discrete-time systems is given by:

$$\sum_{i=0}^n \tilde{v}^2(i) \leq \beta_2^y (\|x_2^c(0)\|_2) + \gamma_2^y \sum_{i=0}^n (\| \begin{bmatrix} v_{des}(i) & x_1'^T(i) \end{bmatrix} \|_2^2), \forall n \geq 0. \quad (3.16)$$

This condition (3.16) must hold for any values of  $x_2^c(0)$ ,  $x_1'$ , and  $v_{des}$ , including the case where  $x_2^c(0) = 0$ ,  $x_1'(0) = 0$ , and  $v_{des}(0) \neq 0$ . In this case, (3.16) degenerates to:

$$v_{des}(0)^2 \leq \gamma_2^y v_{des}(0)^2, \quad (3.17)$$

for  $n = 0$ . Thus, (3.16) can only be satisfied with  $\gamma_2^y \geq 1$  and therefore  $\gamma_1^y \gamma_2^y \geq 1$ .

Therefore, for discrete-time systems described by (3.3) and (3.4), it is impossible to satisfy the stability conditions of Proposition 3.3.1 with  $F = 1$ .

**Remark 3.3.1.** In order to reject the disturbance from  $\tilde{v}$  at steady state, it is necessary for the outer loop to produce an equal and opposite  $v_{des}^f$ , which is equal to  $v_{des}$  when  $F = 1$ . Thus, it is impossible when  $F = 1$  (and in fact for any  $F$  having a DC gain of 1) to simultaneously obtain steady-state rejection of  $\tilde{v}$  and obtain  $\gamma_1^y < 1$ .



While the continuous-time case is less straightforward, the same result is easy to obtain for the linear case, as is illustrated in the following proposition:

**Proposition 3.3.3.** (*Impact of Reference Model on Stability Verification - Continuous-time Case*) Consider a linear, continuous time system fitting the description given by (3.1) and (3.2), where the relative degree from  $u$  to  $v$  is given by  $\rho_u^v$ , with  $\rho_u^v > 0$ , and the relative degree from  $x_1$  to  $v$  is given by  $\rho_{x_1}^v$ , with  $\rho_{x_1}^v > 0$ . Suppose that  $F = 1$  and the outer loop is designed to reject disturbances from  $\tilde{v}$  at steady-state, i.e., the steady-state gain from  $\tilde{v}$  to  $v_{des}$  is equal to 1, thus  $\gamma_1 \geq 1$ . Then  $\gamma_1 \gamma_2 \geq 1$ .

□

**Proof.** A causal controller,  $u(t) = k_1 v_{des}(t) + k_2 x_2^c(t)$  will lead to a closed-loop input-output performance described by:

$$V(s) = T_1(s)V_{des}(s) + T_2(s)X_1(s), \quad (3.18)$$

where  $T_1(s)$ <sup>2</sup> has relative degree  $\rho_{T_1} \geq \rho_u^v > 0$  and  $T_2(s)$  has relative degree  $\rho_{T_2} \geq \rho_{x_1}^v > 0$ . Thus,  $T_1(s)$  and  $T_2(s)$  have high-frequency gains of 0. The relationship from  $v_{des}$  to  $\tilde{v}$  is given by:

$$\tilde{V}(s) = (T_1(s) - 1)V_{des}(s) + T_2(s)X_1(s), \quad (3.19)$$

where  $T_1(s) - 1$  has a high-frequency gain of 1. Thus,  $\gamma_2^y \geq 1$  and  $\gamma_1^y \gamma_2^y \geq 1$ .

These results have shown that without using the reference model, stability cannot, in general, be verified with the small gain condition in the discrete (3.3-3.4) or continuous time (3.1-3.2) case. However, it is important to note that Propositions 3.3.2 and 3.3.3 only show that stability *cannot* be proven without the reference model, although it is not hard to construct examples in which  $F$  is successfully used to obtain  $\gamma_1^y \gamma_2^y < 1$ . The following simple example successfully uses  $F(s)$ , in addition to a very simple inner loop control law, to obtain  $\gamma_2^y < 1$ :

**Example 3.3.1.** Consider linear plant and actuator dynamics given by:

---

<sup>2</sup> $T_{1,2}(s)$  will be used later in this dissertation to represent exactly the same transfer function and transfer function vector.

$$Y(s) = \frac{1}{5s + 1}, \quad (3.20)$$

$$V(s) = \frac{1}{2s^2 + 3s + 1}, \quad (3.21)$$

where the outer and inner loop control laws are given by:

$$V_{des}(s) = \frac{5s + .5}{s}(R(s) - Y(s)), \quad (3.22)$$

$$U(s) = 3(V_{des}(s) - V(s)). \quad (3.23)$$

Consider the reference model:

$$F(s) = \frac{1}{0.5s + 1}. \quad (3.24)$$

This combination of actuator dynamics, inner loop controller, and reference model yields  $\gamma_1^y = 1$  and  $\gamma_2^y = 0.585$ . Therefore, the small gain condition is satisfied.

This represents a very simple example, with a crude control law that is used to illustrate the fact that, with a reference model, stability can be verified using the small gain condition. The subsequent sections take matters farther, deriving conditions under which one can actually obtain  $\gamma_2^y = 0$ , which is a model-matching condition corresponding to  $V = FV_{des}$ . This ability to obtain model-matching not only results in verifiable ISS, but it also leads to the ultimate degree of separation between inner and outer loop design concerns, since the outer loop designer only needs to verify ISS of the outer loop for overall system stability (since  $\gamma_2^y = 0$ , any finite  $\gamma_1^y$  will satisfy the small gain condition). The presence of reference model matching will also lead to interesting conclusions comparing modular control performance with its centralized counterpart. However, before examining this comparison between modular and centralized performance, other modular design approaches are briefly considered, showing how the reference model approach compares insofar as proving stability is concerned.

### 3.3.1 Verifying Stability Under Other Modular Control Design Approaches

Interestingly, both alternative modular design approaches proposed in the literature (target calculation and multi-rate) were introduced chiefly in order to guarantee overall system stability. Both approaches do indeed provide the capability of guaranteeing overall system stability, albeit with some pitfalls.

In the target calculation approach, for the discrete-time case, at time 0,  $v_{des}(k+\rho_u^v)$  is calculated. The actuator, having relative degree  $\rho_u^v$  from  $u$  to  $v$ , has a sufficient number of steps available to influence  $v(\rho_u^v)$ , and the condition of (3.16) can be satisfied with  $\gamma_2^y < 1$  when the summations are indexed from  $\rho_u^v$  to  $n$ .  $v_{des}(i)$  is never actually calculated by the outer loop controller for  $0 \leq i < \rho_u^v$ , so these values of  $v_{des}(i)$  can be set arbitrarily (say, to  $v_{des}(i) \triangleq v(i), 0 \leq i < \rho_u^v$ ), such that (3.16) is satisfied from 0 to  $n$ , with  $\gamma_2^y < 1$ , as required. The idea is similar for continuous time systems, with  $v_{des}^{(\rho_u^v)}(0)$  calculated at time  $t = 0$ . The target calculation approach is equivalent to the reference model approach with  $F = 1$ , with the exception that the outer loop controller must supply  $v_{des}^{(i)}, 0 \leq i \leq \rho_u^v$  for continuous time systems,  $v_{des}(k+i), 0 \leq i \leq \rho_u^v$ , for discrete time systems, rather than just  $v_{des}$  or  $v_{des}(k)$ . In either case, future information regarding  $v_{des}$  is required in order to complete the inner loop target state calculation. In order for the outer loop controller to be causal, and therefore implementable, the input-output relationship from  $r$  to  $v_{des}$  must have relative degree greater than or equal to  $\rho_u^v$ . Thus, the design space of acceptable outer loop controllers is reduced with the target calculation based design, requiring the outer loop controller to be sufficiently slow such that the inner loop can achieve its infinitely fast  $F = 1$  objective.

The multi-rate strategy uses a faster update rate for the inner loop than the outer loop, thereby providing the inner loop several sampling instances to produce an acceptable virtual control signal,  $v$ . At step  $k = 0$ ,  $\nu$  values of  $\tilde{v}(h)$  are generated by the inner loop. Thus, the left hand side of equation (3.16) represents an average of these  $\nu$  sampling instances and can in fact be influenced by the actuators. Both the multi-rate approach and reference model based approach allow us to guarantee overall system stability, and neither offers a clear stability-related advantage or disadvantage over the other. However, the reference model approach will again be compared with the multi-rate approach when performance is considered, at which point the advantages of the reference model based approach will be clear.

### 3.4 Performance Under the Reference Model Based Framework

This section considers the performance of the overall system under the reference model based modular design framework, specifically considering:

- Conditions under which the reference model is matched;
- Conditions under which modular control performance can be made equivalent to centralized (a performance metric will be specified in order to make this comparison rigorous).

Interestingly, knowledge of conditions under which the reference model is matched will be leveraged in order to derive the modular vs. centralized equivalence conditions. The resulting equivalence conditions from this section make the reference model based approach the first to provide a clear means of guaranteeing an overall level of system performance through a modular control design.

The theory in this section is derived *exclusively* for continuous-time linear systems. Specifically, to characterize the performance achieved by the reference model based modular control design and quantify its design limitations, consider linear time invariant systems fitting the block diagram of Figure 1.1, whose dynamics are represented as follows:

$$X_1(s) = P^{x_1}(s)V(s), \quad (3.25)$$

$$Y(s) = P^y(s)V(s), \quad (3.26)$$

$$X_2(s) = A_u^{x_2}(s)U(s) + A_{x_1}^{x_2}(s)X_1(s), \quad (3.27)$$

$$V(s) = A_u^v(s)U(s) + A_{x_1}^v(s)X_1(s), \quad (3.28)$$

where  $P^y(s)$  is a scalar transfer function,  $P^{x_1}(s)$ ,  $A_u^v(s)$ , and  $A_{x_1}^v$  are  $n_1 \times 1$ ,  $1 \times p$ , and  $1 \times n_1$  transfer function vectors, respectively, and  $A_u^{x_2}(s)$  and  $A_{x_1}^{x_2}(s)$  are  $n_2 \times p$  and  $n_2 \times n_1$  transfer function matrices, respectively. The outer and inner loop control laws, respectively, are given by:

$$V_{des}(s) = C_r^o(s)R(s) + C_{x_1}^o(s)X_1(s), \quad (3.29)$$

$$U(s) = C_{v_{des}}^i(s)V_{des}(s) + C_{x_2}^i(s)X_2(s), \quad (3.30)$$

where  $C_r^o(s)$  is a scalar transfer function,  $C_{x_1}^o(s)$  and  $C_{v_{des}}^i(s)$  are  $1 \times n_1$  and  $p \times 1$  transfer function vectors, respectively, and  $C_{x_2}^i(s)$  is a  $p \times n_2$  transfer function matrix. Our results in this section are based on an inner loop controller with full inner loop state information ( $x_2$ ) and an outer loop controller with full outer loop state information ( $x_1$ ). Given an inner loop controller ( $C_{v_{des}}^i(s)$  and  $C_{x_2}^i(s)$ ), the closed inner loop input-output dynamics can be described by:

$$V(s) = T_1(s)V_{des}(s) + T_2(s)X_1(s), \quad (3.31)$$

where, suppressing the  $s$  notation for simplicity, we have:

$$T_1 = A_u^v(I - C_{x_2}^i A_u^{x_2})^{-1} C_{v_{des}}^i, \quad (3.32)$$

$$T_2 = A_u^v(I - C_{x_2}^i A_u^{x_2})^{-1} (C_{x_2}^i A_{x_1}^{x_2}) + A_{x_1}^v. \quad (3.33)$$

For the modular, reference model based control strategy, when the inner loop reference model is matched, we have:

$$T_1(s) = F(s), \quad (3.34)$$

$$T_2(s) = 0. \quad (3.35)$$

The conditions of (3.34)-(3.35) are referred to as the *nominal* closed inner loop.

### 3.4.1 Overall System Performance Metric

This subsection formalizes the notion of overall performance of the closed-loop system by proposing the following metric:

**Definition 3.4.1.** (*Overall Performance Index*) Let  $G^y(s)$  and  $G^{des}(s)$  represent the actual and desired transfer functions, respectively, from  $r$  to  $y$ , and let  $G^v(s)$  represent the actual transfer function from  $r$  to  $v$ . The performance level,  $\gamma^p$ , of the closed-loop system is defined as:

$$\gamma^p \triangleq \left\| \begin{bmatrix} W^y(s)(G^y(s) - G^{des}(s)) & W^v(s)G^v(s) \end{bmatrix} \right\|_{\infty}, \quad (3.36)$$

where  $W^y(s)$  and  $W^v(s)$  are stable weighting transfer functions that emphasize or penalize performance at different frequencies.

Thus, the outer loop controller design can be performed in an attempt to minimize

$\gamma^p$  when the reference model is matched, whereas the performance of the overall system can be analyzed by computing  $\gamma^p$  once the control design is complete. Since the definition of  $\gamma^p$  is independent of the control design approach, it can be used as the metric for evaluating the performance of both the modular and centralized control strategies.

**Remark 3.4.1.** *The performance index of Definition 3.4.1 is independent of inner loop states, except for the combined effect of these states, as expressed by  $v$ . For the MISO systems considered in this paper, it is often in fact the case that the same inner loop input-output behavior can be generated through many different control input ( $u$ ) and inner loop state ( $x_2$ ) trajectories, which is a major topic of interest in control allocation literature, but not within the scope of this work. However, the performance index,  $\gamma^p$ , can take into account any outer loop state by simply including it in the output,  $y$ .*

**Remark 3.4.2.** *Measurement noise can be augmented to the external signal (which presently includes only the setpoint,  $r$ ) without consequence to any of the remaining claims in this paper. Such an augmentation is typically necessary to satisfy well-posedness assumptions of optimal control design methodologies, such as  $H_\infty$ , that can be used to minimize  $\gamma^p$ . Incidentally, the inclusion of a penalty on  $v$  is also typically necessary for a well-posed  $H_\infty$  optimization.*

### 3.4.2 Inner Loop Reference Model Matching Conditions

In this subsection, conditions are derived under which it is possible to achieve inner loop model matching, i.e.,  $T_1(s) = F(s)$ ,  $T_2(s) = 0$ . The following proposition specifies these conditions:

**Proposition 3.4.1.** *(Model Matching Conditions)*

1. *Suppose that there exists a control input,  $u_i, i \in [1 \dots p]$ , such that:*

- $\rho(A_{u_i}^v(s)) \leq \rho(F(s))$ ,
- $\rho(A_{u_i}^v(s)) < \rho(A_{x_1}^v(s))$ , and
- $A_{u_i}^v(s)$  is minimum phase.

*Then there exists a proper transfer function  $C_{v_{des}}^i(s)$  and a proper transfer function vector  $C_{x_1}^i(s)$  (i.e., all constituent transfer functions are proper), such that the inner loop control law,  $U(s) = C_{v_{des}}^i(s)V_{des}(s) + C_{x_2}^i(s)X_2(s)$ , is stabilizing and yields  $V(s) = F(s)V_{des}(s)$ .*

2. Furthermore, no such control law exists if:

- $\rho(A_u^v(s)) > \rho(F(s))$ , or
- $\rho(A_u^v(s)) \geq \rho(A_{x_1}^v(s))$ .

□

**Proof.** Part 1: Assume, without loss of generality, that  $p = 1$  (if  $p > 1$ , one can always choose one control input,  $j$ , to use, leaving  $u_i = 0, i \neq j$ ). For notational simplicity, let  $\rho_1 \triangleq \rho(A_u^v)$ ,  $\rho_2 \triangleq \rho(A_{x_1}^v)$ . Performing an observable realization of the transfer function relationship,  $V(s) = A_u^v(s)U(s) + A_{x_1}^v(s)X_1(s)$ , where:

$$\begin{aligned}
 A_u^v(s) &= \frac{N_u(s)}{D(s)}, \\
 A_{x_1}^v(s) &= \frac{N_{x_1}(s)}{D(s)}, \\
 N_u(s) &= b_{n-\rho_1}s^{n-\rho_1} + \dots + b_0, \\
 N_{x_1}(s) &= d_{n-\rho_2}s^{n-\rho_2} + \dots + d_0, \\
 D(s) &= s^n + a_{n-1}s^{n-1} + \dots + a_0,
 \end{aligned} \tag{3.37}$$

we obtain the state-space representation:

$$\begin{aligned}
 \dot{\bar{x}}_2 &= \begin{bmatrix} -a_{n-1} & 1 & 0 & \dots & 0 \\ -a_{n-2} & 0 & 1 & & 0 \\ & & \dots & & \\ -a_1 & 0 & \dots & 0 & 1 \\ -a_0 & 0 & \dots & 0 & 0 \end{bmatrix} \bar{x}_2 + \begin{bmatrix} 0 \\ \vdots \\ b_{n-\rho_1} \\ \vdots \\ b_0 \end{bmatrix} u + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ d_{n-\rho_2} \\ \vdots \\ d_0 \end{bmatrix} x_1 \\
 v &= \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix} \bar{x}_2.
 \end{aligned}$$

Here,  $\bar{x}_2$  represents the transformed actuator states resulting from the observable realization. This state space representation can be partitioned as:

$$\begin{aligned}
 \begin{bmatrix} \dot{\bar{x}}_{21} \\ \dot{\bar{x}}_{22} \end{bmatrix} &= \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} \bar{x}_{21} \\ \bar{x}_{22} \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u + \begin{bmatrix} 0 \\ D_2 \end{bmatrix} x_1 \\
 v &= \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix} \bar{x}_{21}
 \end{aligned} \tag{3.38}$$

where  $B_1 = \begin{bmatrix} 0 & \dots & 0 & b_{n-\rho_1} \end{bmatrix}^T$ . This yields:

$$\begin{aligned}\dot{\bar{x}}_{21} &= A_{11}\bar{x}_{21} + A_{12}\bar{x}_{22} + B_1u \\ A_{12}\bar{x}_{22} &= \begin{bmatrix} 0 & \dots & 0 & \bar{x}_{22}^1 \end{bmatrix}^T,\end{aligned}$$

where  $\bar{x}_{22}^1$  is the first element in  $\bar{x}_{22}$ . Partition the control law as  $u = u_a + u_b$ , where:

$$u_a = -\frac{\bar{x}_{22}^1}{b_{n-\rho_1}}.$$

This yields:

$$\begin{aligned}\dot{\bar{x}}_{21} &= A_{11}\bar{x}_{21} + B_1u_b \\ v &= \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix} \bar{x}_{21}\end{aligned}$$

Thus, the disturbance from  $x_1$  has been completely canceled, leaving us with:

$$V(s) = A_{new}(s)U_b(s),$$

where  $A_{new}(s)$  is  $\rho_1^{th}$  order. Note that, defining  $K \triangleq \begin{bmatrix} \frac{1}{b_{n-\rho_1}} & 0 & \dots & 0 \end{bmatrix}$ , the zero dynamics are given by:

$$\dot{\bar{x}}_{22} = (A_{22} - B_2K)\bar{x}_{22} + B_2u_b + A_{21}\bar{x}_{21} + D_2x_1,$$

where:

$$A_{22} - B_2K = \begin{bmatrix} -\frac{b_{n-\rho_1-1}}{b_{n-\rho_1}} & 1 & 0 & \dots & 0 \\ \vdots & 0 & 1 & & 0 \\ \vdots & & \dots & & \\ \vdots & 0 & \dots & 0 & 1 \\ -\frac{b_0}{b_{n-\rho_1}} & 0 & \dots & 0 & 0 \end{bmatrix}. \quad (3.39)$$

Thus, the eigenvalues of  $A_{22} - B_2K$  are identical to the zeros of  $A_u^v(s)$ , which requires  $A_u^v(s)$  to be minimum phase for the zero dynamics of (3.39) to be stable. Now, given that  $A_{new}(s)$  is minimum phase (it has no zeros) and satisfies  $\rho_1 = \rho(A_{new}(s)) \leq \rho(F(s))$ , standard model reference control [47] can be used to design:



$$U_b(s) = C_1^{mrc}(s)V_{des}(s) - C_2^{mrc}(s)V(s),$$

such that  $u_b$  is stabilizing and  $V(s) = F(s)V_{des}(s)$ . Noting that the virtual control input can be given by  $v = Cx_2$  (where  $C \in \mathbb{R}^{1 \times n_2}$ ), the final inner loop control law clearly can be written as  $U(s) = C_{v_{des}}^i(s)V_{des}(s) + C_{x_2}^i(s)X_2(s)$ , as required for the inner loop controller.

Part 2: Suppose that  $\rho(A_u^v(s)) > \rho(F(s))$ . In order to achieve  $T_1(s) = F(s)$ , we require:

$$A_u^v(I - C_{x_2}^i A_u^{x_2})^{-1} C_{v_{des}}^i = F(s). \quad (3.40)$$

However, given that the inner loop controller must be causal, we have:

$$\rho(A_u^v(I - C_{x_2}^i A_u^{x_2})^{-1} C_{v_{des}}^i) \geq \rho(A_u^v) > \rho(F),$$

which contradicts (3.40), leading to the conclusion that one cannot achieve  $T_1(s) = F(s)$ .

Suppose that  $\rho(A_u^v(s)) \geq \rho(A_{x_1}^v(s))$ . In order to achieve  $T_2(s) = 0$ , we require:

$$A_u^v(I - C_{x_2}^i A_u^{x_2})^{-1} C_{x_2}^i A_{x_1}^{x_2} = -A_{x_1}^v. \quad (3.41)$$

However, given that the inner loop controller must be causal, we have:

$$\rho(A_u^v(I - C_{x_2}^i A_u^{x_2})^{-1} C_{x_2}^i A_{x_1}^{x_2}) > \rho(A_u^v) \geq \rho(A_{x_1}^v),$$

which contradicts (3.41), leading to the conclusion that one cannot achieve  $T_2(s) = 0$  and completing the proof.

**Remark 3.4.3.** The requirement that  $A_u^v(s)$  be minimum phase could be lifted if the reference model incorporated the same nonminimum phase zeros as the actuators. However, we will require  $F(s)$  to be minimum phase for future performance-related derivations.

Proposition 3.4.1 essentially requires the reference model to be sufficiently “slow” relative to the actuator dynamics from  $u$  to  $v$  ( $A_u^v(s)$ ) and the actuator’s disturbance dynamics ( $A_{x_1}^v(s)$ ) to be sufficiently “slow” relative to the actuator dynamics from  $u$  to  $v$  ( $A_u^v(s)$ ). The minimum phase requirement on  $A_u^v(s)$  ensures that the system will remain internally stable when the model-matching inner loop controller is designed. Clearly, several system features can cause these conditions to be violated, and these

features will be discussed in detail in Section 3.5. Moreover, while a simplified actuator model (discarding higher-order dynamics and the effect of  $x_1$ , for example) can be used to satisfy the reference model matching conditions, these dynamics will manifest themselves as uncertainty when the inner and outer loops are combined. The mechanisms in Section 3.5 provide a means for dealing with this as well.

**Remark 3.4.4.** *While the reference model matching conditions derived here clearly rely heavily on the relative degree of  $F(s)$ , there are other practical considerations that should go into one's choice of reference model. For example, if a relative degree 1 system is linear and minimum phase, then it is mathematically possible to derive a control law such that the closed inner loop can match an arbitrarily fast relative degree 1 reference model. However, in the presence of uncertainty and saturation limits, it may be unrealistic from an engineering perspective to choose a reference model corresponding to a very fast time constant if the actuator dynamics themselves are slow; this fact must be taken into consideration in the reference model design. This topic of reference model selection will be treated in Section 3.6 for the thermal management application example.*

### 3.4.3 Modular vs. Centralized Performance

This subsection compares the performance level of the modular control strategy to that of its centralized counterpart. For this performance comparison, the following notation is introduced:

**Definition 3.4.2.** *Let  $\gamma_o^p$ ,  $\gamma_c^p$ , and  $\gamma_m^p$  represent the values of  $\gamma^p$ , as defined in (3.36), achieved by the ideal outer loop control ( $\tilde{v} = 0$ ), centralized control, and modular control (referring to Figure 1.1), respectively.*

**Remark 3.4.5.** *It is important to realize that while  $\gamma_c^p$  and  $\gamma_m^p$  represent performance levels that are in fact attainable by their respective controllers,  $\gamma_o^p$  represents an ideal performance level that rests on the outer loop design assumption that  $\tilde{v} = 0$ , which may not be achievable.*

The first comparison of these three levels of performance is given in the following proposition:

**Proposition 3.4.2.** *(Modular vs. Centralized Performance Inequality) Suppose that  $F(s)$  is stable, minimum phase, and satisfies  $\rho(F(s)) \leq \rho(A_u^v(s))$  and  $\rho(F(s)) \leq \rho(A_{x_1}^v(s))$ . Denote the optimal levels of performance (i.e., the lowest values of  $\gamma^p$ ,*

as defined in Definition 3.4.1) achievable by the ideal outer loop design, centralized control system, and modular control system, under causal and stabilizing controllers, as  $\gamma_o^*$ ,  $\gamma_c^*$ , and  $\gamma_m^*$ , respectively. Then  $\gamma_o^* \leq \gamma_c^* \leq \gamma_m^*$ .

□

**Proof.**  $\gamma_c^* \leq \gamma_m^*$ : Since the modular control system is a special case of the centralized controller,  $\gamma_c^* \leq \gamma_m^*$  follows immediately.

$\gamma_o^* \leq \gamma_c^*$ : Consider an optimal centralized control law given by:

$$U^*(s) = C_r^c(s)R(s) + C_{x_1}^c(s)X_1(s) + C_{x_2}^c(s)X_2(s), \quad (3.42)$$

which yields  $\gamma_c^p = \gamma_c^*$ . We achieve closed-loop performance given by (suppressing the  $s$  notation):

$$\begin{aligned} V &= (1 - A_u^v(I - C_{x_2}^c A_u^{x_2})^{-1}(C_{x_1}^c + C_{x_2}^c A_{x_1}^{x_2})P^{x_1} \\ &\quad - A_{x_1}^v P^{x_1})^{-1} A_u^v(I - C_{x_2}^c A_u^{x_2})^{-1} C_r^c R, \end{aligned} \quad (3.43)$$

$$Y = P^y V. \quad (3.44)$$

For the outer loop controller in the modular control system, apply the causal and stabilizing control law:

$$V_{des}(s) = F^{-1}(s)(A_u^v(s)\hat{U}(s) + A_{x_1}^v(s)X_1(s)), \quad (3.45)$$

where:

$$\hat{U}(s) = (I - C_{x_2}^c A_u^{x_2})^{-1}(C_r^c R + (C_{x_1}^c + C_{x_2}^c A_{x_1}^{x_2})X_1). \quad (3.46)$$

Note that  $\hat{u}$  in (3.46) is identical to  $u^*$  in (3.42) when  $x_2$  is solved for in terms of  $u$  and  $x_1$ . When ideal conditions are met (i.e.,  $V(s) = F(s)V_{des}(s)$ ), this yields:

$$\begin{aligned} V &= (1 - A_u^v(I - C_{x_2}^c A_u^{x_2})^{-1}(C_{x_1}^c + C_{x_2}^c A_{x_1}^{x_2})P^{x_1} \\ &\quad - A_{x_1}^v P^{x_1})^{-1} A_u^v(I - C_{x_2}^c A_u^{x_2})^{-1} C_r^c R, \end{aligned} \quad (3.47)$$

$$Y = P^y V, \quad (3.48)$$

which is identical to (3.43), thereby yielding  $\gamma_o^p = \gamma_c^*$ . Thus, it follows that  $\gamma_o^* \leq \gamma_c^* \leq \gamma_m^*$ .

This simple modular vs. centralized performance inequality yields more profound implications when combined with the earlier reference model matching conditions of Proposition 3.4.1. This leads to the following proposition:

**Proposition 3.4.3.** (*Modular vs. Centralized Equivalence Conditions*) Suppose that  $F(s)$  is stable and minimum phase,  $\rho(F(s)) = \rho(A_u^v(s))$ ,  $\rho(A_u^v(s)) < \rho(A_{x_1}^v(s))$ , and  $A_u^v(s)$  is minimum phase. Then  $\gamma_o^* = \gamma_c^* = \gamma_m^*$ .

□

**Proof.** Suppose  $F(s)$ ,  $C_r^o(s)$ , and  $C_{x_1}^o(s)$  are chosen optimally to yield  $\gamma_o = \gamma_o^*$ . From Proposition 3.4.1, there exists a causal inner loop controller (with transfer functions  $C_{v_{des}}^i(s)$  and  $C_{x_2}^i(s)$ ) that yields  $V(s) = F(s)V_{des}(s)$ , thus yielding  $\gamma_m = \gamma_o = \gamma_o^*$ . It then follows immediately from Proposition 3.4.2 that  $\gamma_m^* = \gamma_c^* = \gamma_o^*$ .

This result states that, under a performance metric conforming to Definition 3.4.1 and specified conditions on the actuator dynamics, the centralized and modular controllers are capable of equivalent performance. While these conditions are restrictive, and are certainly not satisfied by all systems, they provide us with a basis for understanding when modular control represents an attractive control strategy and what features may lead to difficulties.

**Remark 3.4.6.** It follows from Proposition 3.4.3 that  $\gamma_o^*$  is independent of one's choice of reference model, as long as  $F(s)$  is stable and minimum phase. To gain an understanding of why this is the case, note that the outer loop controller can compensate for differences between one reference model and another (for example, the reference model  $F_1(s) = \frac{1}{\tau_s+1}$  can result in the same performance that is achieved with  $F_2(s) = \frac{\alpha s+1}{(\beta s+1)(\tau_s+1)}$  if the outer loop controller in the first case is simply augmented with  $\frac{\alpha s+1}{\beta s+1}$ ). One should be careful of this conclusion however, noting, in light of Remark 3.4.4, that the reference model will likely not be matched exactly in the presence of uncertainties and saturation limits, resulting in  $\gamma_m \neq \gamma_o$ , even though the system is designed to yield  $\gamma_m = \gamma_o$  under the nominal model. Thus, it is still advisable from an engineering perspective to explore different choices of  $F(s)$ , even when the nominal system model satisfies the conditions of Proposition 3.4.3.

### 3.4.4 Summary - Ideal Modular Control Conditions

The results up to this point provide conditions on the actuator dynamics and reference model that, when satisfied, simultaneously yield reference model matching and modular vs. centralized equivalence, which are summarized as follows:

- $F(s)$  is stable and minimum phase;
- $A_u^v(s)$  is minimum phase;
- $\rho(A_u^v(s)) < \rho(A_{x_1}^v(s))$ ;
- $\rho(F(s)) = \rho(A_u^v(s))$ .

The similarity of the reference model matching conditions and equivalence conditions indicates that the circumstances that prevent reference model matching will also prevent the modular control strategy from performing as well as its centralized counterpart. Moreover, from the modular vs. centralized performance inequality, a strategy that is aimed at bringing  $\gamma_m$  closer to  $\gamma_o^*$  will automatically bring  $\gamma_m$  closer to  $\gamma_c^*$ . In light of these observations, the remaining sections of this chapter will focus on recovering ideal performance (driving  $\gamma_m$  closer to  $\gamma_o^*$ ) when  $F(s)$  is not matched, recognizing that this will also lead to a closed-loop modular control system that more closely matches the optimal centralized performance.

### 3.4.5 Performance Guarantees Under Other Modular Control Design Approaches

In terms of stability, the reference model based approach presented no clear advantage over the target calculation or multirate approaches. However, with regard to performance, the reference model based approach is the only one that establishes an equivalence result along with a clear method for obtaining equivalent performance (when equivalence is in fact possible).

For the target calculation approach, the inner loop is actually based on matching the reference model  $F = 1$ , but with the requirement that the outer loop produces derivatives or future values of  $v_{des}$ , up to the relative degree of the actuator dynamics. One could theoretically achieve equivalent centralized vs. modular performance under the ideal actuator conditions that have been established, but there is still no means of actually designing the update law for future values of  $v_{des}$  such that these values of  $v_{des}$  can be realized by the inner loop and the equivalence result can be guaranteed.

The multi-rate approach is clearly limited to a discrete time systems, and our performance results of Propositions 3.4.1, 3.4.2, and 3.4.3 are based on continuous-time systems. This is a minor disadvantage, as it is not difficult to extend these results to discrete-time implementations. However, even under a discrete-time implementation, the literature on multi-rate modular control has not produced an equivalence result.

### 3.5 Performance Recovery Formulation

At this point, conditions have been derived under which the inner loop controller can be designed such that the inner closed-loop performance matches a suitably designed reference model,  $F(s)$ . In general, it is difficult (and sometimes impossible) to design an inner loop controller that precisely matches a prescribed reference model for at least two reasons:

1. *Non-ideal Actuator Properties* - The ideal actuator conditions require  $\rho(A_u^v(s)) < \rho(A_u^{x_1}(s))$ , which is violated in cases where the disturbance from  $x_1$  enters the actuator dynamics at the same location as the control input,  $u$ . Additionally, features such as time delays will render  $A_u^v(s)$  nonminimum phase, violating a sufficient condition for reference model matching.
2. *Actuator Modeling Uncertainties* - If the inner loop design is performed using an actuator model that possesses uncertainties, then the closed inner loop is unlikely to match  $F(s)$ .

This dissertation will use  $e'$  to denote the difference in the output of the overall system with the actual vs. ideal inner closed loop, where:

$$e' \triangleq y - y^r. \quad (3.49)$$

Here,  $y$  represents the actual output whereas  $y^r$  represents the output that is attained under the scenario given by (3.34).

In order to capture performance mismatch when  $T_1 \neq F$  and  $T_2 \neq 0$ , an uncertainty model is introduced for the closed inner loop, given by:

$$\begin{aligned} T_1 &= F + W_1 \Delta_1, \\ T_2 &= W_2 \Delta_2, \\ \|\Delta_1\|_\infty &\leq \Delta_1^{max}, \\ \|\Delta_2\|_\infty &\leq \Delta_2^{max}, \end{aligned} \quad (3.50)$$

where  $\Delta_1$  and  $\Delta_2$  are scalar and  $n_1 \times n_1$  diagonal matrix uncertainties, respectively, which represent bounded inner closed-loop reference model mismatch.  $W_{1,2}$  are known stable transfer functions (a  $1 \times n_1$  dimensional transfer function vector in the case of  $W_2$ ) that capture information known about the uncertainty. Both uncertainty sources may be cast together as:

$$\Delta = \begin{bmatrix} \Delta_1 & 0 \\ 0 & \Delta_2 \end{bmatrix}. \quad (3.51)$$

**Remark 3.5.1.** *Note that the uncertainty is given for the inner closed loop, and consequently the characteristics of the uncertainty set, such as the bound, will, in general, be affected by the inner loop control law as well as uncertainties in the system dynamics. The appropriate choice of inner loop control law and reference model ( $F(s)$ ) will have an effect on the closed inner loop uncertainty.*

**Remark 3.5.2.** *The structure of the uncertainty model (additive, multiplicative, feedback, etc.) should be chosen to be most representative for the application at hand. While an additive uncertainty structure is used for the application in this dissertation, the design procedure employed here is applicable to any other uncertainty structures in which  $\Delta$  is interconnected with the remainder of the system dynamics.*

The performance recovery results will make extensive use of the *worst-case gain*,  $\gamma^{wc}$ , from  $r$  to  $e'$  where  $\gamma^{wc}$  is defined as:

$$\gamma^{wc} \triangleq \max_{\|\Delta\|_\infty \leq \Delta_{max}} \|G^{er}\|_\infty, \quad (3.52)$$

where:

$$G^{er} \triangleq \frac{E'(s)}{R(s)}. \quad (3.53)$$

**Remark 3.5.3.** *The variable  $e'$  (3.49), and consequently the worst-case gain,  $\gamma^{wc}$ , are based on the difference between the output,  $y$ , and its nominal value,  $y^r$ , but does not consider the size of  $v$ . If the size of  $v$  is important for overall performance, then this can be augmented to  $e'$  without consequence to the performance recovery formulation.*

A novel mechanism, referred to as *modular control error compensation* (MCEC), is proposed, which is designed using  $\mu$  synthesis to manage performance degradation that is captured by  $\gamma^{wc}$ . The modular control structure with MCEC is given in Figure 3.2, where  $C_v$  represents the MCEC. The proposed MCEC design structure exploits the signal  $\tilde{v}$ , which represents the difference between the virtual control input,  $v$ , and its nominal value,  $v_{des}^f$  (the output of  $F$ ), to reduce the gap between nominal

and actual performance. While the controller structure proposed in Figure 3.2 is applicable to both linear and nonlinear systems, the scope of the analytical results presented here is limited to systems in which the blocks in Figure 3.2 are linear.

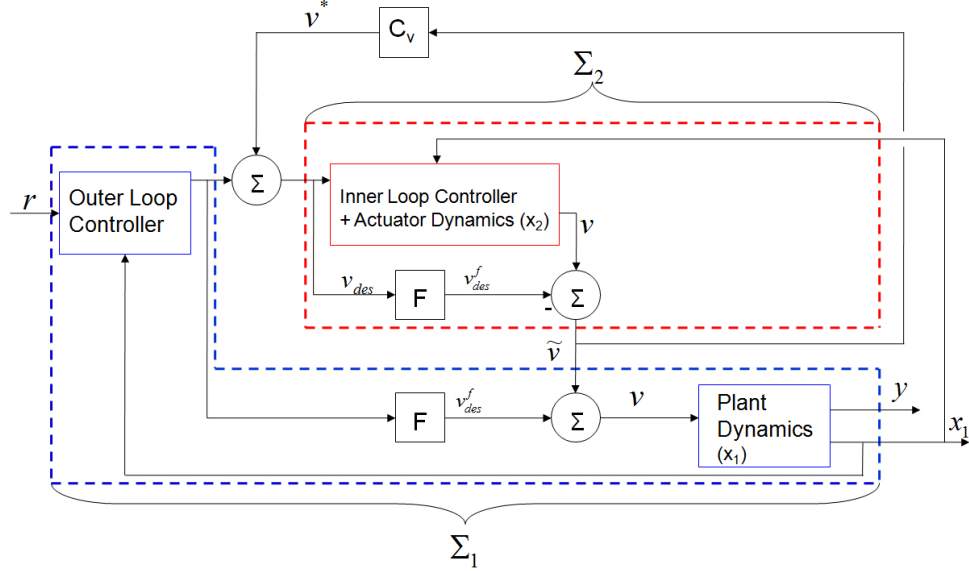


Figure 3.2: Block diagram of the reference model based modular control design with MCEC.

Because the description of the mismatch between ideal and actual inner loop performance is based on an uncertainty model, robust control tools provide the means of designing compensators to recover performance while taking this mismatch into account. Our approach to robust control design uses  $\mu$  synthesis [48]-[50] to design  $C_v$  for performance recovery. The proposed strategy differs from other  $\mu$  synthesis applications, such as [51]-[55], which design the main controller (in this case, the outer loop controller) to treat the worst-case uncertainty scenario. The designs of [51]-[55] lead to robust performance in a min-max sense but also result in a tradeoff between nominal performance and performance in the presence of uncertainty. In contrast to these other approaches, the approach proposed here leaves the outer loop controller fixed and uses  $\mu$  synthesis to design the add-on compensator,  $C_v$ , to address performance deterioration. This structure was originally inspired by a similar mechanism that was used in [56] for parametric uncertainty. The structure used here, which is an extension of the author's conference publication [57], employs  $C_v$  as an add-on compensator, resulting in two key advantages over strategies that redesign the main controller for robust performance:

1. The proposed strategy retains the modularity of the inner and outer loops in



the sense that it allows the two controllers to be designed in parallel without knowledge of each other;

2. When the inner loop reference model is matched ( $T_1(s) = F(s)$ ,  $T_2(s) = 0$ ), the nominal performance will be recovered automatically, since  $\tilde{v}$  will go to 0.

By recasting the optimization problem for the MCEC design, one can formulate a  $\mu$  synthesis problem to optimize specific worst-case performance metrics. Depending on the a priori knowledge about the uncertainties and other design constraints, two related synthesis problems can be formulated:

**S1** Given that  $\|\Delta\|_\infty < 1$ , find a stabilizing  $C_v$  that minimizes  $\gamma^{wc}$ .

**S2** Given  $\gamma^{th} > 0$  ( $\gamma^{th}$  is a performance threshold representing tolerable performance degradation), find  $C_v$  that maximizes  $\|\Delta\|_\infty$  such that the closed-loop system remains stable and  $\gamma^{wc} < \gamma^{th}$ .

**Remark 3.5.4.** *Each synthesis objective appeals to a slightly different control problem. Objective S1 aims for maximum performance recovery when given a bound on the closed inner loop uncertainty. On the other hand, objective S2 maximizes allowable uncertainty for a given performance threshold. It will be shown that the same design tool can be used to achieve both synthesis objectives.*

**Remark 3.5.5.** *Worst case performance may be more generically defined by:*

$$\gamma^{wc}(C_v) = \max_{\|\Delta\|_\infty \leq \Delta_{max}} \|W^g G^{er}\|_\infty. \quad (3.54)$$

*where  $W^g$  is a weighting function that penalizes performance degradation at particular frequencies, thereby allowing more design flexibility.*

### 3.5.1 MCEC Design for Performance Recovery Using $\mu$ Synthesis

For the purposes of designing  $C_v$  to achieve our synthesis objectives, S1 and S2, the block diagram of Figure 3.2 can be cast in the form given by Figure 3.3, where  $\bar{P}$  is a transfer function matrix containing all of the system components besides  $C_v$  and  $\Delta$  ( $\theta_{1,2}$  are fictitious signals that represent the output of the uncertainty block,  $\Delta$ ). The expression for  $\bar{P}$  is given in (3.55).

$$\begin{aligned}
\bar{P} &= [\bar{P}_1 \quad \bar{P}_2] \\
\bar{P}_1 &= \begin{bmatrix} \frac{C_{x_1}^o P^{x_1} W_1}{1-C_{x_1}^o P^{x_1} F} & \frac{C_{x_1}^o P^{x_1} W_2}{1-C_{x_1}^o P^{x_1} F} \\ (I - P^{x_1} F C_{x_1}^o)^{-1} P^{x_1} W_1 & (I - P^{x_1} F C_{x_1}^o)^{-1} P^{x_1} W_2 \\ \frac{P^y W_1}{1-F C_{x_1}^o P^{x_1}} & \frac{P^y W_2}{1-F C_{x_1}^o P^{x_1}} \\ W_1 & W_2 \end{bmatrix} \\
\bar{P}_2 &= \begin{bmatrix} \frac{C_r^o}{1-C_{x_1}^o P^{x_1} F} & \frac{1}{1-C_{x_1}^o P^{x_1} F} \\ (I - P^{x_1} F C_{x_1}^o)^{-1} P^{x_1} F C_r^o & (I - P^{x_1} F C_{x_1}^o)^{-1} P^{x_1} F \\ 0 & \frac{P^y F}{1-F C_{x_1}^o P^{x_1}} \\ 0 & 0 \end{bmatrix}
\end{aligned} \tag{3.55}$$

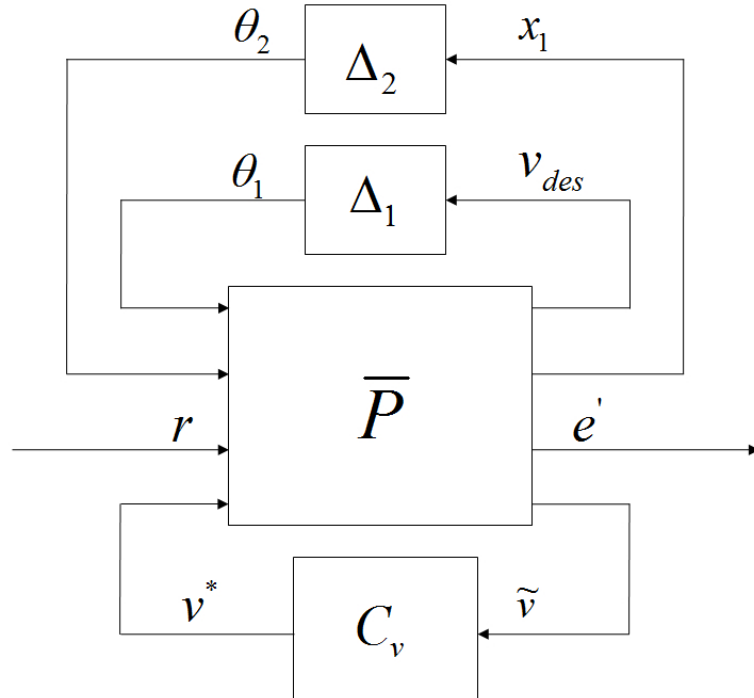


Figure 3.3: Generic  $\mu$  synthesis formulation, without consideration of performance.

Given this system representation, the objective is either to minimize the worst case gain,  $\gamma^{wc}$  (S1), or to maximize the allowable size of uncertainty under which  $\gamma^{wc} < \gamma^{th}$  (S2). Both of these objectives lend themselves to the  $\mu$  synthesis design tool, given the following proposition [48]:

**Proposition 3.5.1.** (*Interpretation of  $\mu$* ) *The structured singular value is the smallest scalar value  $\mu$  such that the system of Figure 3.3 is stable for all  $\Delta : \|\Delta\|_\infty < \frac{1}{\mu}$ . Furthermore, there exists a transfer function  $\Delta : \|\Delta\|_\infty = \frac{1}{\mu}$ , which results in an unstable closed-loop system.*

□

Given Proposition 3.5.1,  $\mu$  characterizes the allowable levels of uncertainty for closed-loop stability.  $\mu$  synthesis is the process of designing a controller to minimize  $\mu$  for a particular closed-loop system. The process is described in [48]-[50] and is based on the *D-K iteration*, which is depicted generically in Figure 3.4. Here, the controller to be synthesized is given by  $K$  (for MCEC, this is equivalent to  $C_v$ ), whereas  $D$  represents a *scaling matrix* that is used to achieve the  $\mu$  synthesis objective. For a given  $D$ , an  $H_\infty$  optimization is performed, and  $D$  is adjusted to consider the block diagonal structure of the uncertainty,  $\Delta$ . The reader is referred to [48]-[50] and references therein for the details.

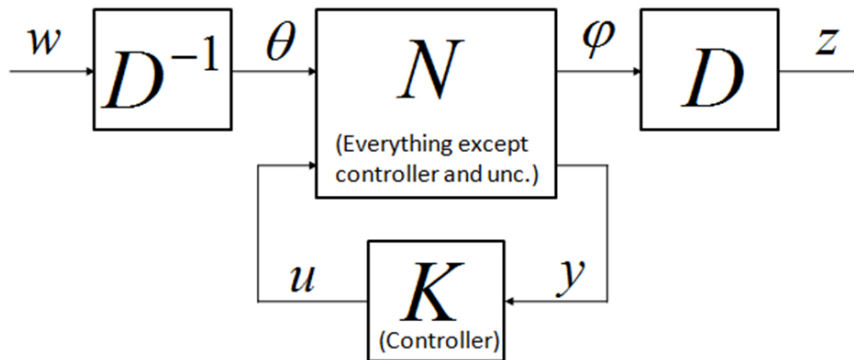


Figure 3.4: Diagram illustrating the setup for the D-K iteration on a generic system. For our system,  $K$  is equivalent to  $C_v$ .

The process of  $\mu$  synthesis has been facilitated and streamlined through the development of numerical tools such as the MATLAB Robust Control Toolbox that is used for the design example in this work.

Without further modification to the block diagram of Figure 3.3,  $\mu$  does not have a clear connection with the performance goals encompassed by our synthesis

objectives (S1 and S2). To consider performance, the uncertainty structure of Figure 3.3 is augmented such that the reference-error loop is wrapped around a fictitious uncertainty block,  $\Delta_P$ , and a scalar gain block,  $k$ , as in Figure 3.5. Under this configuration,  $\mu$  synthesis considers an uncertainty structure given by:

$$\Delta_{aug} \triangleq \begin{bmatrix} \Delta & 0 \\ 0 & \Delta_P \end{bmatrix}. \quad (3.56)$$

This leads to the following result [48], which follows from (3.56) and the small gain condition:

**Proposition 3.5.2.** *(Robust performance interpretation of  $\mu$ )* Let  $\mu$  be the structured singular value for the system shown in Figure 3.5. The following two properties hold for  $\mu$ :

1. The closed-loop system of Figure 3.3 is stable and  $k\|G^{er}\|_\infty < \mu$  for all uncertainties satisfying  $\|\Delta\|_\infty < \frac{1}{\mu}$ .
2. There exists a perturbation  $\Delta : \|\Delta\|_\infty = \frac{1}{\mu}$  for which the closed-loop system of Figure 3.3 is unstable or  $k\|G^{er}\|_\infty = \mu$ .

□

The adjustable parameter  $k$  weights the relative importance of robust performance vs. stability and is used here to provide a mechanism such that the synthesis objectives S1 and S2 can be achieved. The following propositions allow us to develop a design algorithm to achieve the synthesis objectives by iteratively adjusting  $k$  and carrying out  $\mu$  synthesis to design  $C_v$ .

**Proposition 3.5.3.** *(Synthesis Objective (S1))* Assume that S1 is feasible. Let  $\mu^*(k, \bar{P})$  be the structured singular value of the system shown in Figure 3.5 where  $C_v$  has been designed using  $\mu$  synthesis, for a given constant  $k$ . Furthermore, let  $k_1^*$  be the maximum value of  $k$  such that  $\mu^*(k, \bar{P}) = 1$ . Then  $k_1^*$  exists. Furthermore, if  $C_v$  is designed using  $\mu$  synthesis, with  $k = k_1^*$ , then  $\gamma^{wc}$  is minimized and the closed loop is stable, subject to  $\|\Delta\|_\infty < 1$ .

□

**Proof.** Existence of  $k_1^*$ : Given that S1 is feasible, there exists a stabilizing  $C_v$  when  $\|\Delta\|_\infty < 1$ , which results in a finite  $\gamma^{wc}$ . Taking  $k = \frac{1}{\gamma^{wc}}$ , it follows from Proposition

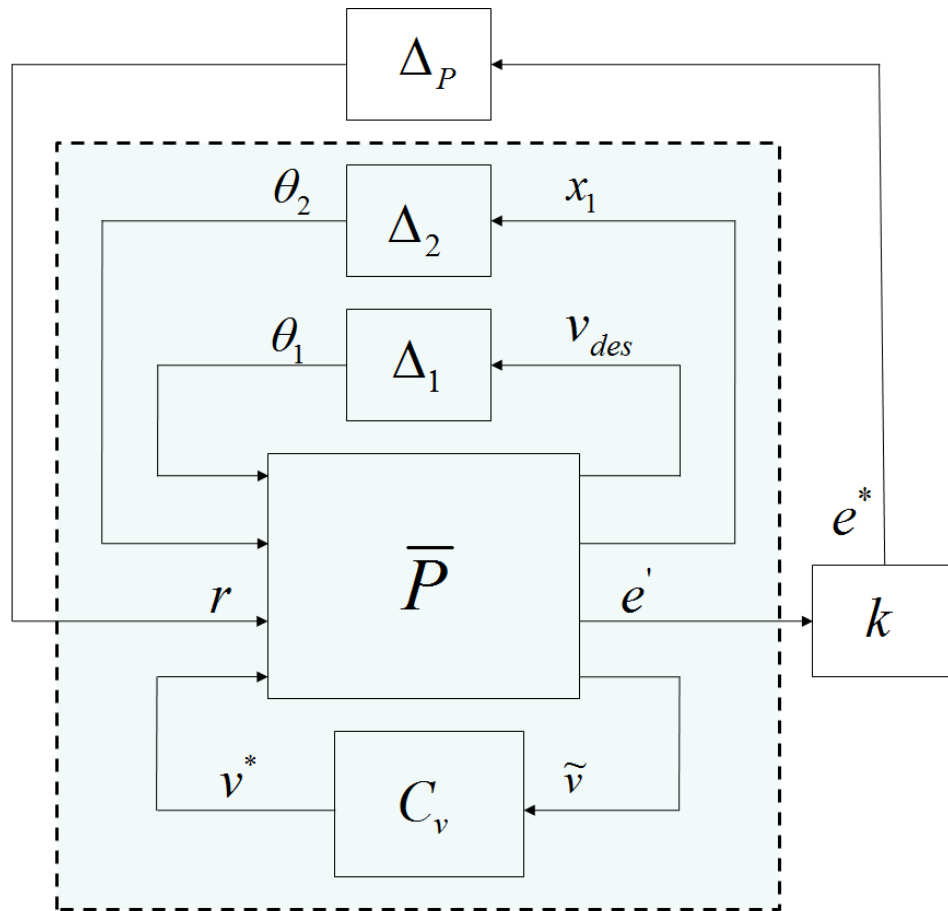


Figure 3.5:  $\mu$  synthesis performance formulation. Physical entities of the system are contained inside the dashed lines, whereas fictitious entities ( $k$  and  $\Delta_P$ ), which are strictly introduced for  $\mu$  synthesis design purposes, lie outside.

3.5.2 that  $\mu(C_v, k, \bar{P}) = 1$ . Because the set  $\{k : \mu^*(k, \bar{P}) = 1\}$  is closed, it has a maximum, thus  $k_1^*$  exists.

When  $C_v$  is designed with  $k = k_1^*$ , the closed loop is stable and  $\gamma^{wc}$  is minimized, subject to  $\|\Delta\|_\infty < 1$ : Closed-loop stability when  $\mu^* = 1$  follows directly from property (1) of Proposition 3.5.2. To show that  $\gamma^{wc}$  is minimized, we first show by contradiction that  $\gamma^{wc} = \frac{\mu^*}{k_1^*}$ . Suppose that  $\gamma^{wc} < \frac{\mu^*}{k_1^*}$  ( $\gamma^{wc} > \frac{\mu^*}{k_1^*}$  violates Proposition 3.5.2, property (1), and need not be considered). Then there exists  $k > k_1^*$  that also yields  $\mu^*(k, \bar{P}) = 1$ , which contradicts the fact that  $k_1^*$  is the maximum value of  $k$  yielding  $\mu^*(k, \bar{P}) = 1$ . Therefore,  $\gamma^{wc} = \frac{\mu^*}{k_1^*} = \frac{1}{k_1^*}$ . Because  $k$  is maximized and  $\mu$  synthesis minimizes  $\mu$ ,  $\gamma^{wc}$  is minimized for all  $C_v, k$  that yield  $\mu(C_v, k, \bar{P}) = 1$ . Finally, note that for all  $C_v$  that yield a stable closed loop when  $\|\Delta\|_\infty < 1$ , there exists a  $k$  such that  $\mu(C_v, k, \bar{P}) = 1$ . Therefore, minimizing  $\gamma^{wc}$  over all  $C_v, k$  that yield  $\mu(C_v, k, \bar{P}) = 1$  is equivalent to minimizing  $\gamma^{wc}$  over all  $C_v$  that yield a stable closed loop when  $\|\Delta\|_\infty < 1$ .

**Proposition 3.5.4.** (Synthesis Objective (S2)) Let  $\mu^*(k, \bar{P})$  be the structured singular value of the system shown in Figure 3.5 where  $C_v$  has been designed using  $\mu$  synthesis, for a given constant  $k$ . Let  $k_2^*$  be the minimum value of  $k$  such that  $\mu^*(k, \bar{P}) = k\gamma^{th}$ . Then  $k_2^*$  exists, and if  $C_v$  is designed with  $k = k_2^*$ , then  $\mu$  synthesis maximizes the value of  $\|\Delta\|_\infty$  such that  $\gamma^{wc} < \gamma^{th}$  and the closed-loop system remains stable.

□

**Proof.** Existence of  $k_2^*$ : S2 is always feasible since it always is possible to achieve  $\gamma^{wc} < \gamma^{th}$  for sufficiently small  $\Delta$ . Take  $k = \frac{1}{\Delta^* \gamma^{th}}$ , which will result in  $\mu(C_v, k, \bar{P}) = \frac{1}{\Delta^*}$ . Therefore, we have  $\mu(C_v, k, \bar{P}) = k\gamma^{th}$ , thereby proving existence. Because the set  $\{k : \mu^*(k, \bar{P}) = k\gamma^{th}\}$  is closed,  $k_2^*$  exists.

When  $C_v$  is designed with  $k_2 = k_2^*$ , tolerable uncertainty is maximized: To show that tolerable uncertainty is maximized, we first show, by contradiction, that there exists  $\Delta : \|\Delta\|_\infty = \frac{1}{\mu^*}$  that leads to closed-loop instability. Suppose that for all  $\Delta : \|\Delta\|_\infty = \frac{1}{\mu^*}$ , the closed loop is stable. Then there exists  $k < k_2^*$  that yields  $\mu^*(k, \bar{P}) = k\gamma^{th}$ , which contradicts the fact that  $k_2^*$  is the minimum value of  $k$  yielding  $\mu^*(k, \bar{P}) = k\gamma^{th}$ . Therefore, there does exist  $\Delta : \|\Delta\|_\infty = \frac{1}{\mu^*}$  that leads to closed-loop instability. Since  $k_2^*$  is minimized and  $\mu$  synthesis minimizes  $\mu$ , tolerable uncertainty is maximized for all  $C_v, k$  yielding  $\mu(C_v, k, \bar{P}) = k\gamma^{th}$ . Finally, it follows from the existence proof that there always exists a  $k$  that yields  $\mu(C_v, k, \bar{P}) = k\gamma^{th}$ . Therefore, tolerable uncertainty is maximized over all  $C_v$  such that  $\gamma^{wc} < \gamma^{th}$  and the closed-loop system remains stable.

Using Propositions 3.5.3 and 3.5.4 and treating  $k$  as an adjustable parameter, the following iterative algorithm is proposed for the design of the MCEC controller  $C_v$  to achieve objectives S1 and S2; for S1, take  $\mu^o = 1$ , for S2, take  $\mu^o = k\gamma^{th}$ :

1. (a) Initialize  $k_{low}$  to any value that is known to satisfy  $k_{low} < k_i^*$  ( $i = 1$  or  $i = 2$  depending on the objective). Take  $k_{low} = 0$  if no other lower bound is known. Proceed to step (1b).
- (b) Initialize  $k_{high}$  to any value that is known to satisfy  $k_{high} > k_i^*$ , and proceed to step (2). If no upper bound on  $k$  is known, make an initial guess,  $k_{init}$ , and carry out  $\mu$  synthesis for  $k = k_{init}$ .
  - i. If  $\mu^*(k_{init}, \bar{P}) - \mu^o > \epsilon$ , take  $k_{high} = k_{init}$  and continue with step (2).
  - ii. If  $\mu^*(k_{init}, \bar{P}) - \mu^o \leq \epsilon$ , increase  $k_{init}$  and repeat step (1b).
2. Carry out  $\mu$  synthesis for  $k = \frac{1}{2}(k_{high} + k_{low})$ , which will return  $C_v$  and  $\mu^*$ . Proceed to (3).
3. (a) If  $|\mu^* - \mu^o| < \epsilon$ , move to (4) for S1 and (5) for S2.
- (b) If  $\mu^* - \mu^o \leq -\epsilon$ , set  $k_{low} = k$  and repeat (2).
- (c) If  $\mu^* - \mu^o \geq \epsilon$ , set  $k_{high} = k$  and repeat (2).
4. (a) If  $|k_{high} - k| < \epsilon_k$ , where  $\epsilon_k$  is a user-defined threshold, terminate the algorithm.
- (b) If  $|k_{high} - k| \geq \epsilon_k$ , take  $k_{low} = k$  and move to (4c).
- (c) Take  $k = \frac{1}{2}(k_{low} + k_{high})$ , and carry out  $\mu$  synthesis.
  - i. If  $|\mu^*(k, \bar{P}) - \mu^o| \geq \epsilon$ , set  $k_{high} = k$  and repeat (4c) until  $|k_{high} - k| < \epsilon_k$ .
  - ii. Otherwise, take  $k_{low} = k$  and repeat (4c) until  $|k_{high} - k| < \epsilon_k$ .
5. (a) If  $|k - k_{low}| < \epsilon_k$ , where  $\epsilon_k$  is a user-defined threshold, terminate the algorithm.
- (b) If  $|k - k_{low}| \geq \epsilon_k$ , take  $k_{high} = k$  and move to (5c).
- (c) Take  $k = \frac{1}{2}(k_{low} + k_{high})$ , and carry out  $\mu$  synthesis.
  - i. If  $|\mu^*(k, \bar{P}) - \mu^o| \geq \epsilon$ , set  $k_{low} = k$  and repeat (5c) until  $|k - k_{low}| < \epsilon_k$ .
  - ii. Otherwise, take  $k_{high} = k$  and repeat (5c) until  $|k - k_{low}| < \epsilon_k$ .

This algorithm is illustrated in the flowchart of Figure 3.6.

**Remark 3.5.6.** *As an alternative to the iterative procedure proposed here, a skew  $\mu$  synthesis procedure may be employed, in which the size of one uncertainty block is held fixed while the other(s) are allowed to vary [58]-[59]. This framework requires additional mathematical tools to set up, but may result in reduced computational effort once implemented.*

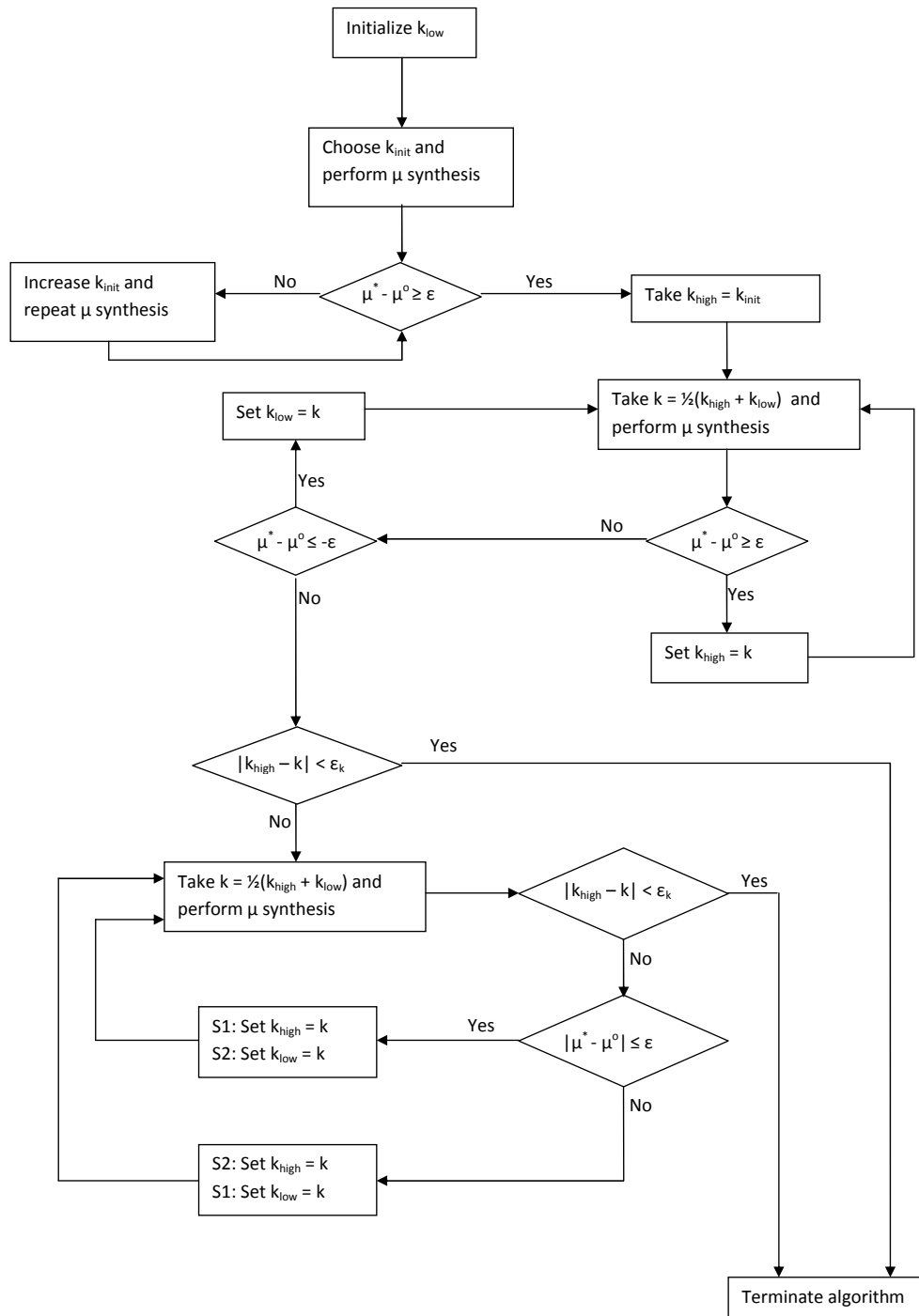


Figure 3.6: Flow chart of the iterative  $\mu$  synthesis algorithm used for MCEC design.



## 3.6 Engine Thermal Management Application

The proposed modular control design approach is now applied to the engine thermal management system, described in detail in Chapter 2.

### 3.6.1 Modular System Description and Properties

While there are two actuators available (heater and mixing valve), only the valve will be active in the setup for this chapter (leveraging overactuation is deferred to Chapter 4), and will have control input  $\phi$ , where  $\phi$  represents the ratio of flow through the heater to total flow rate, commonly referred to as the *bypass ratio*. Because linear design techniques are being used, a linearization is performed about  $T_{out,eng} = 80$  degrees C (and the corresponding steady-state), and translated coordinates, i.e.,  $\delta T_{out,eng} = T_{out,eng} - T_{out,eng}^{ss}$ , etc., as well as a translated control input,  $\delta\phi$ , are used for the control design. The ultimate objective is to control engine outlet temperature ( $T_{out,eng}$ ) to a setpoint, with the virtual control input taken as  $T_{out,mv}$ , yielding the modular system description:

$$\begin{aligned}
 x_1 &= \begin{bmatrix} \delta T_{out,eng} \\ \delta T_{eng1} \\ \delta T_{eng2} \end{bmatrix}, x_2 = \begin{bmatrix} \delta T_{out,mv} \\ \delta T_{out,ht} \\ \delta T_{ht} \\ \delta T_{out,hex} \\ \delta T_{out,cw} \end{bmatrix}, \\
 y &= \delta T_{out,eng}, \\
 v &= \delta T_{out,mv}, \\
 u &= \delta\phi.
 \end{aligned}$$

As it turns out, for the linearized actuator dynamics,  $A_u^v(s)$  is minimum phase,  $\rho(A_u^v(s)) = 1$ , and  $\rho(A_{x_1}^v(s)) = 2$ . Thus, the modeled actuator dynamics satisfy the requirements for equivalence between optimal modular and centralized system performance, making this system an excellent candidate for modular control. Nevertheless, the system model is not perfect, and modeling uncertainties will lead to a mismatch between nominal and actual inner loop performance. This makes the use of MCEC an attractive strategy for the thermal management system, and, as such, it will be the main focus of the experimental results.

### 3.6.2 Inner Loop Reference Model

Given that  $A_u^v(s)$  is minimum phase and  $\rho(A_u^v(s)) = 1$ , an appropriate reference model should be stable, minimum phase, and have relative degree 1. For this system, the simplest such reference model is chosen, namely:

$$F(s) = \frac{1}{\tau_f s + 1}. \quad (3.57)$$

The system analysis and experimental results will explore the effect of adjusting  $\tau_f$ , both with and without MCEC in place.

**Remark 3.6.1.** *It is important to explore the effects of adjusting the reference model time constant,  $\tau_f$ , in light of Remark 3.4.6, which points out that even though the ideal actuator properties are satisfied for the system (which imply, through Proposition 3.4.3 that the performance level will not depend on  $\tau_f$  if the model is perfect), the performance level is likely to differ for different reference models when uncertainty is present. Alternatively, the reference model could be chosen with any number of stable, minimum phase lead-lag elements appended to the structure of  $F(s)$  considered in (3.57). This has not been done for the thermal management application and therefore remains a possible topic of further investigation. However, for the thermal management system, it will be shown that good performance is attained with the simple reference model structure of (3.57), particularly when MCEC is used.*

### 3.6.3 Outer Loop Control Design

The objective of the outer loop controller is to yield desirable overall system performance when the inner closed loop matches the reference model, i.e.,  $V(s) = F(s)V_{des}(s)$ . For the thermal management system, a simple outer loop PI controller, given by:

$$V_{des} = \frac{2s + 0.05}{s}(R(s) - Y(s)), \quad (3.58)$$

or, in terms of temperatures,

$$\delta T_{out,mv}^{des} = \frac{2s + 0.05}{s}(\delta T_{out,eng}^{des}(s) - \delta T_{out,eng}(s)), \quad (3.59)$$

yields excellent tracking performance when the reference model is matched. This can be seen in Figs. 3.12-3.13, which show the predicted performance of the system when  $F(s)$  is matched, in addition to experimental results.

### 3.6.4 Inner Loop Design

Given the actuator model, it is possible to design the control law for  $\delta\phi$  such that  $\delta T_{out,mv}(s) = F(s)\delta T_{out,mv}^{des}(s)$  by taking:

$$\begin{aligned}
 \delta\phi &= k_1\delta T_{out,mv}^{des} - k_2\delta T_{out,mv} - k_3\delta T_{out,ht} - k_4\delta T_{out,hex}, \\
 k_1 &= \frac{\tau_t}{d}, \\
 k_2 &= \frac{\tau_t - 1}{d}, \\
 k_3 &= \frac{\phi^{ss}}{d}, \\
 k_4 &= \frac{1 - \phi^{ss}}{d}, \\
 d &= T_{out,ht}^{ss} - T_{out,hex}^{ss},
 \end{aligned} \tag{3.60}$$

where  $\tau_t$  is the modeled time constant associated with the temperature dynamics of the mixing valve.

### 3.6.5 Performance Recovery in the Presence of Uncertainty - $\mu$ Synthesis Setup

Given that the actuator model is not perfect, MCEC design is now proposed to keep the performance of the system within 10 percent of the ideal performance. This corresponds to synthesis objective S2, where, to approximate the 10 percent requirement,  $\gamma^{th} = 0.1$ .

To construct weighting functions, likely sources of parametric uncertainty are considered, and these uncertainties are “lumped” into  $W_1\Delta_1$  and  $W_2\Delta_2$ . Specifically, three potential sources of uncertainty are considered:

- Flow dynamics, where the flow distribution through the heater and heat exchanger does not respond immediately, but rather through a time constant,  $\tau_{flow}$ ;
- Uncertain time constant,  $\tau_t$ , associated with mixing valve thermal mixing dynamics;
- Inaccurate steady-state estimates,  $\phi^{ss}$ ,  $T_{out,ht}^{ss}$ , and  $T_{out,hex}^{ss}$ , which impact our linearized dynamics. An uncertain value of  $\phi^{ss}$  is especially realistic for two reasons. First, the valve is not position controlled (it is pressure-controlled and

Table 3.1: Uncertain actuator model parameters

Parameter	Nominal Value	Range
$\tau_{flow}$	0 s	0.1-2 s
$\tau_t$	8 s	6-10 s
$\phi^{ss}$	0.75	0.7-0.8
$T_{out,ht}^{ss} - T_{out,hex}^{ss}$	40.8 degrees C	35-45 degrees C

sometimes yields inconsistent response). Secondly, the ambient temperature affects the required value of  $\phi^{ss}$  to achieve a particular steady-state (through the effect of ambient temperature on the cooling water temperature supplied to the system).

When augmented to include flow dynamics (3.61), the inner loop dynamics that are affected by the uncertainty are given by:

$$\delta\dot{\phi} = \frac{1}{\tau_{flow}}(-\delta\phi + \delta u_{mv}), \quad (3.61)$$

$$\begin{aligned} \delta\dot{T}_{out,mv} = & \frac{1}{\tau_t}(-\delta T_{out,mv} + \phi^{ss}\delta T_{out,ht} \\ & + (1 - \phi^{ss})\delta T_{out,hex} + (T_{out,ht}^{ss} - T_{out,hex}^{ss})\delta\phi) \end{aligned} \quad (3.62)$$

where the mixing valve command,  $u_{mv}$ , does not immediately affect the valve position (and flow distribution), represented by  $\phi$ . In characterizing the uncertain inner loop dynamics, uncertainties appearing in the other four inner loop state equations (for the heater and heat exchanger) are not considered, largely because these can be compensated for through state feedback of  $T_{out,ht}$  and  $T_{out,hex}$ . The uncertain parameters, their nominal values, and an approximate estimate of their ranges, are given in Table 3.1. Note that estimating the ranges of the uncertain parameters is done strictly for the purpose of generating weighting transfer functions ( $W_1(s)$  and  $W_2(s)$ ) that characterize the effect of this uncertainty, and synthesis objective S2 has been chosen precisely because there is a lack of knowledge of exactly how large these uncertain parameters can become. Moreover, the list of uncertain parameters given in Table 3.1 is not an exclusive list of all possible sources of uncertainty. However, because we are employing  $\mu$  synthesis as our design approach, the final design will be robust for *all* uncertainties within a bound, not just parametric uncertainties.

For the thermal management system, although there are 3 plant states, only one

of these states has any effect on the actuator dynamics; thus, 2 of the entries in  $W_2(s)$  will be equal to 0. To simplify matters, we drop these transfer functions from the formulation entirely (the resulting  $\Delta$  is  $2 \times 2$ ). Analyzing the closed-loop effects of these uncertainties through Bode plots of pseudo-random uncertain systems (via MATLAB), it has been concluded that the following weighting functions capture the frequency-dependent impact of the expected sources of system uncertainty:

$$W_1(s) = \frac{.2(.1s + 1)(100s + 1)}{(.3s + 1)(20s + 1)}, \quad (3.63)$$

$$W_2(s) = 0.1.$$

Nonzero DC content is essential to capture the true system uncertainty, since there is no integrator in the inner loop (nor is one necessary, due to the outer loop integrator), and the uncertain elements of Table 3.1 (especially the third) will contribute to a DC error.

$\mu$  synthesis was performed for several reference models (with different time constants,  $\tau_f$ ), each leading to a different compensator,  $C_v(s)$ . In spite of the differences, and the high orders of the resulting compensators, each reflects a somewhat de-tuned and causal approximation of the compensator  $C_v(s) = \frac{1}{F(s)}$ , which results from setting  $G^{er}(s) = 0$ , and solving for  $C_v(s)$ . In light of this fact, the resulting compensators were parameterized as:

$$C_v(s) = \frac{b_1s + b_0}{a_2s^2 + a_1s + a_0}, \quad (3.64)$$

where the transfer function coefficients were determined through a balanced truncation. The resulting worst-case performance levels,  $\gamma^{wc}$ , under these (reduced-order) compensators, are given in Figure 3.7, for different values of  $\tau_f$ .

Figure 3.7 shows that while MCEC provides benefit for all reference models, it is most helpful for aggressive reference models (i.e., fast time constants), where the inner loop designer is pursuing a higher-bandwidth reference model which exacerbates the effect of uncertainties on inner closed-loop performance. Although the reference model is typically negotiated in the beginning stages of a control design project, the graph of Figure 3.7 can only be generated after the outer loop controller has been designed and closed inner loop uncertainty has been characterized (through weighting functions). Therefore, while it is evident from Figure 3.7 that  $F(s) = \frac{1}{4s+1}$  represents

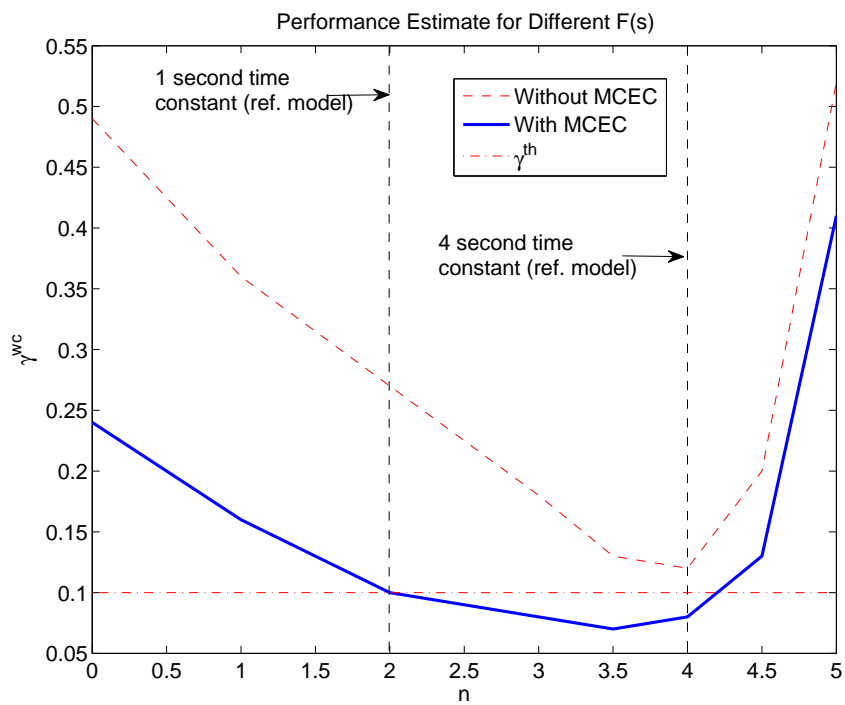


Figure 3.7: Performance comparison with and without MCEC, for different reference model time constants, based on speculated sources of uncertainty. In parameterizing the reference model, the figure takes  $\tau_f = 0.25(2^n)$ .

a good reference model choice both with and without MCEC, this may not be known to a designer a priori; it is therefore advantageous to have MCEC available since it results in a system whose performance is more robust to one's choice of reference model.

As a final note, for very slow reference models, overall behavior degrades, since the inner loop is designed to a bandwidth specification that is not well separated from that of the outer loop. This lack of bandwidth separation makes the overall system performance more sensitive to any mismatch between the reference model and actual inner loop performance; thus, control performance both with and without MCEC is deteriorated.

### 3.6.6 Simulation Results and Analysis

Prior to acquiring experimental data, MCEC was simulated for the thermal management system, under one of the worst possible cases of uncertainty in terms of mixing valve time constants, namely:

$$\tau_{flow} = 2 \tag{3.65}$$

$$\tau_t = 10 \tag{3.66}$$

This reflects much slower actuator dynamics than nominal. Additionally, in order to create errors in  $\phi^{ss}$  and  $T_{out,ht}^{ss} - T_{out,hex}^{ss}$  (in reality, these errors arise from other factors, such as ambient conditions, and are not directly adjustable as parameters in the physical thermal management model), cooling water inlet temperature was adjusted downwards to 10 degrees C (from 20 degrees C).

Figures 3.8 and 3.9 indicate marked performance improvement with the use of MCEC with a fast ( $\tau_f = 1$  second) reference model. With  $\tau_f = 4$  seconds, performance is good both with and without MCEC. These findings are consistent with Figure 3.7, which indicates that performance is helped most by MCEC in cases of aggressive reference models.

To gain deeper insight into the mechanisms by which MCEC improves performance, one can examine the Bode plots in Figures 3.10 and 3.11, where the frequency response of the open loop transfer function is plotted, breaking the loop at the outer loop controller output. The figures indicate improved phase margins with the use of MCEC, particularly for the case of  $\tau_f = 1$  second. This result makes intuitive sense, since MCEC designs  $C_v$  to maximize the size of uncertainty for which the system is

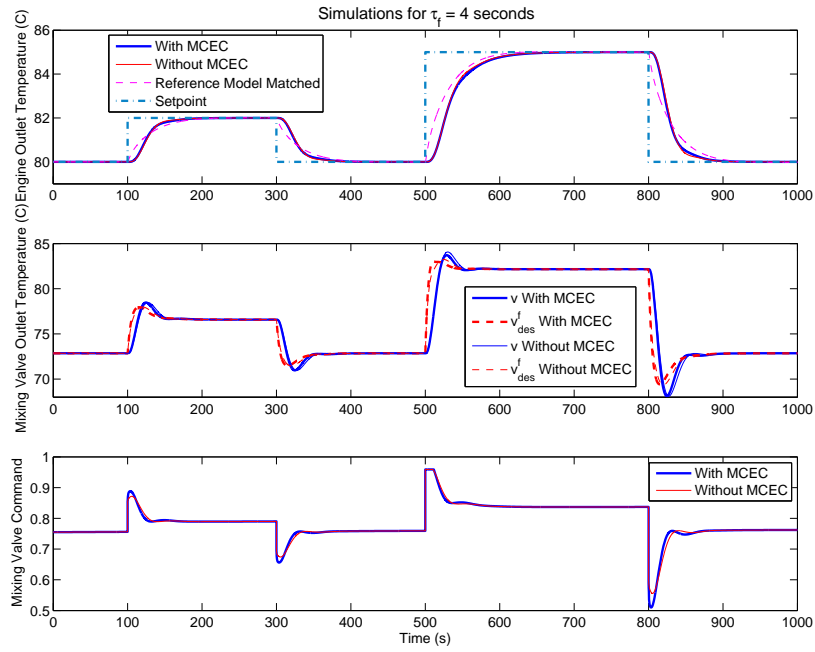


Figure 3.8: Simulation results with and without MCEC for  $\tau_f = 4$  seconds.

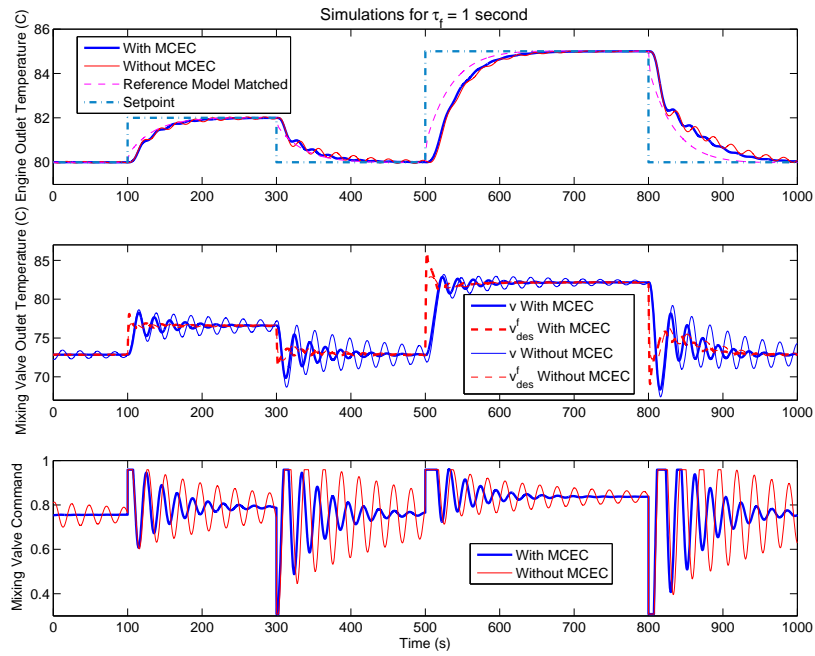


Figure 3.9: Simulation results with and without MCEC for  $\tau_f = 1$  second.



stable and satisfies the predetermined performance threshold. However, it is important to note that if the performance threshold,  $\gamma^{th}$ , is taken to be very small, then  $\mu$  synthesis may result in a compensator that only leads to closed-loop stability for very small  $\Delta$ , even though allowable uncertainty was indeed maximized subject to the performance threshold. In this case, one would say that the MCEC problem, through one's choice of  $\gamma^{th}$ , has led to a poor balance of performance vs. robust stability. The results of Figures 3.10 and 3.11 indicate that the MCEC design pursued here, with our choice of  $\gamma^{th} = 0.1$ , has led to an appropriate balance of performance vs. robust stability.

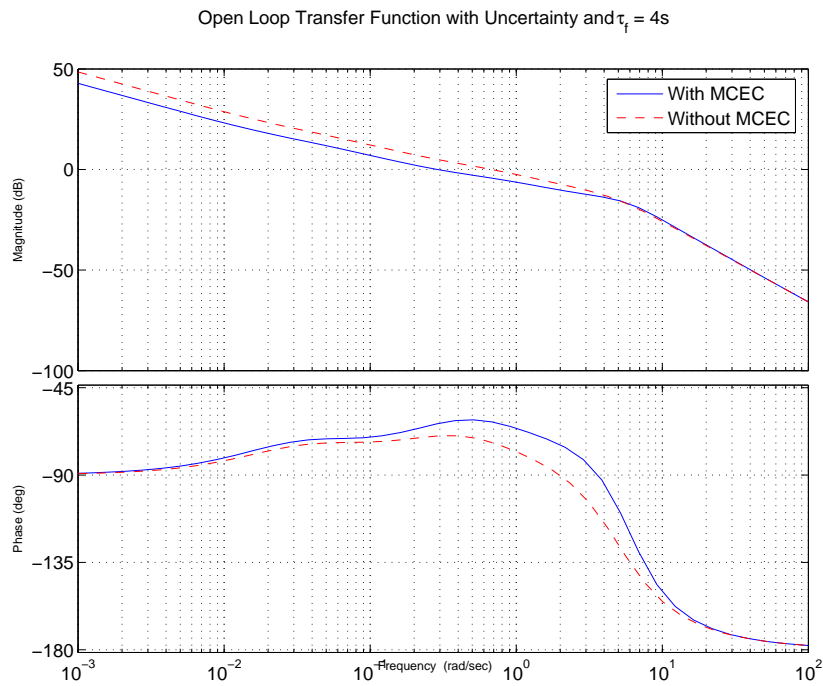


Figure 3.10: Open loop frequency response comparison with and without MCEC for  $\tau_f = 4$  seconds.

### 3.6.7 Experimental Results

The performance of the modular control strategy and MCEC was verified in an engine test cell, with the engine running at 2000 rpm and a 75 N-m load. These experimental results examined 2 degree setpoint changes where the linearized model is highly accurate, as well as 5 degree changes, which begin to excite nonlinearities (including saturation constraints on the upward step).

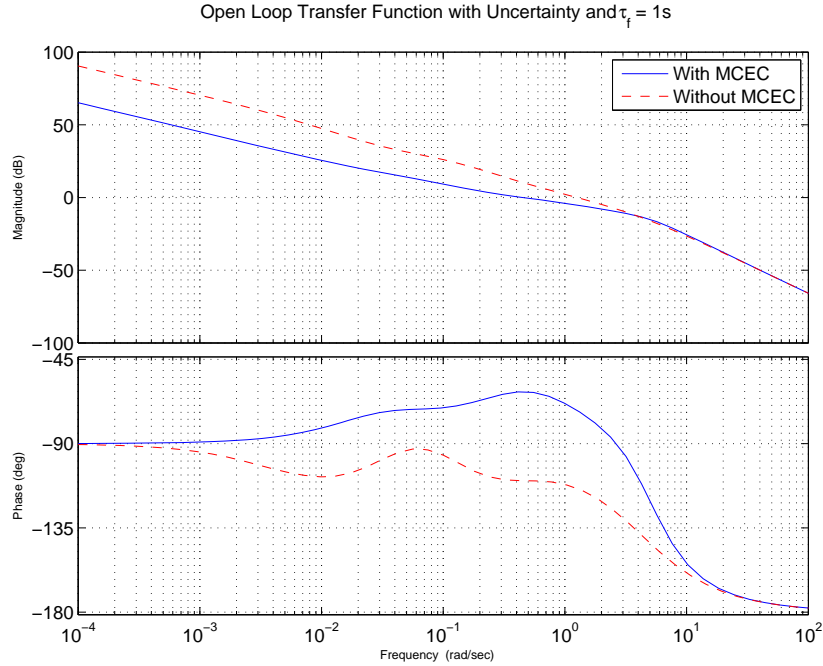


Figure 3.11: Open loop frequency response comparison with and without MCEC for  $\tau_f = 1$  second.

The plots given in Figures 3.12-3.14 show an improvement in performance with MCEC, with both  $\tau_f = 1$  and  $\tau_f = 4$ . As expected, the performance improvement is more pronounced for the faster reference model,  $\tau_f = 1$ , where the system without MCEC exhibits oscillations throughout the temperature responses.

In order to compare performance quantitatively, consider the following metric, which approximately captures, for our discrete-time data sets, the gain from  $r$  to  $e'$ :

$$\gamma^{exper} = \frac{(\sum_{i=0}^n ((\delta T_{out,eng}(i) - \delta T_{out,eng}^r(i))^2))^{1/2}}{(\sum_{i=0}^n ((\delta T_{out,eng}^{des}(i))^2))^{1/2}}. \quad (3.67)$$

The results are shown in Figure 3.15. As expected,  $\gamma^{exper}$  is lower with MCEC than without under both reference models, but the difference is much more pronounced when  $\tau_f = 1$ . Also, with MCEC,  $\gamma^{wc} < \gamma^{th}$  with both reference models, indicating that acceptable performance has been achieved with MCEC in both cases. The experimental results correlate well with the worst-case analysis results, which show that the performance gap between ideal and actual performance becomes larger for the faster ( $\tau_f = 1$  second) reference model and that MCEC has more of an effect in this case.

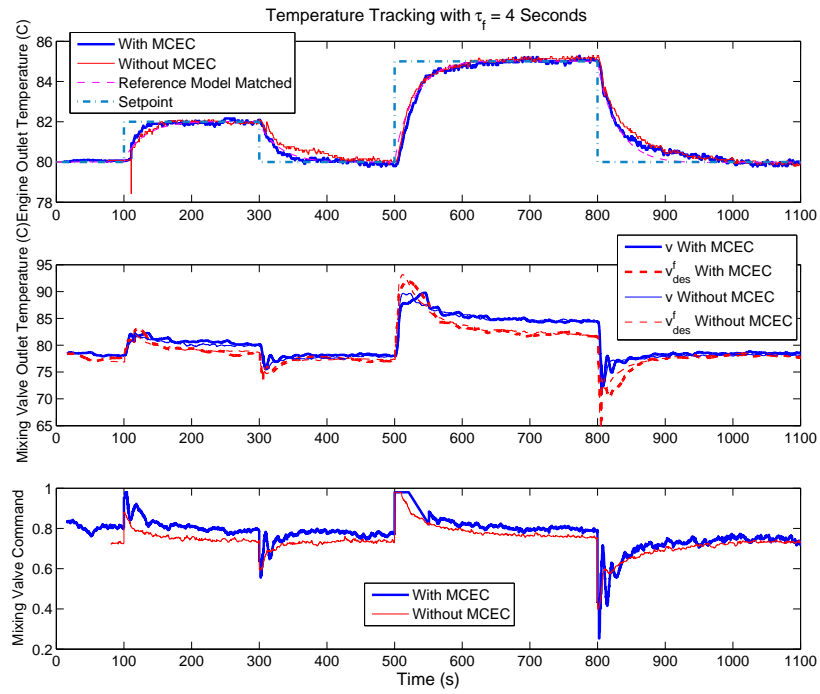


Figure 3.12: Experimental results with  $\tau_f = 4$  seconds.

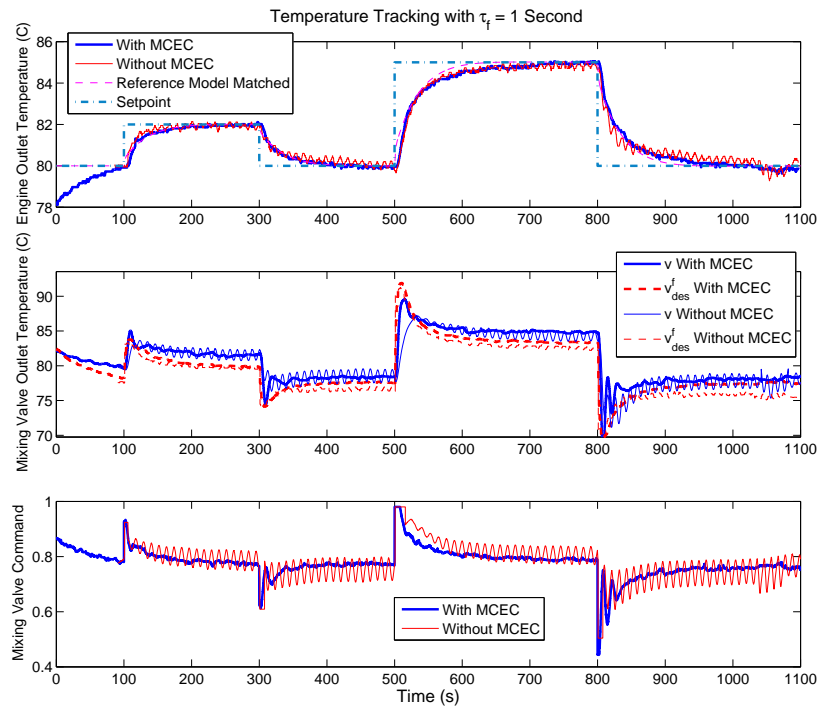


Figure 3.13: Experimental results with  $\tau_f = 1$  second.

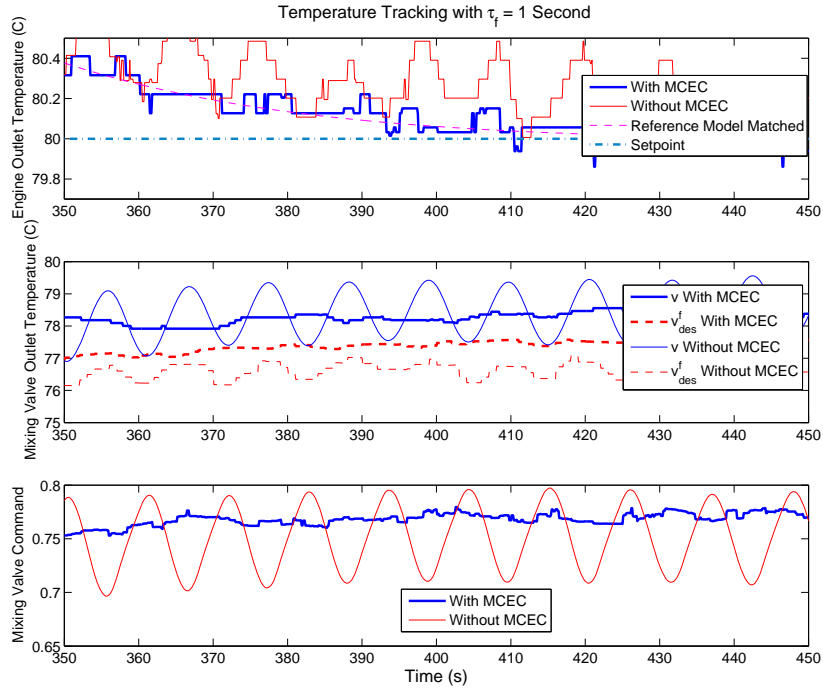


Figure 3.14: Experimental results with  $\tau_f = 1$  second - zoomed in.

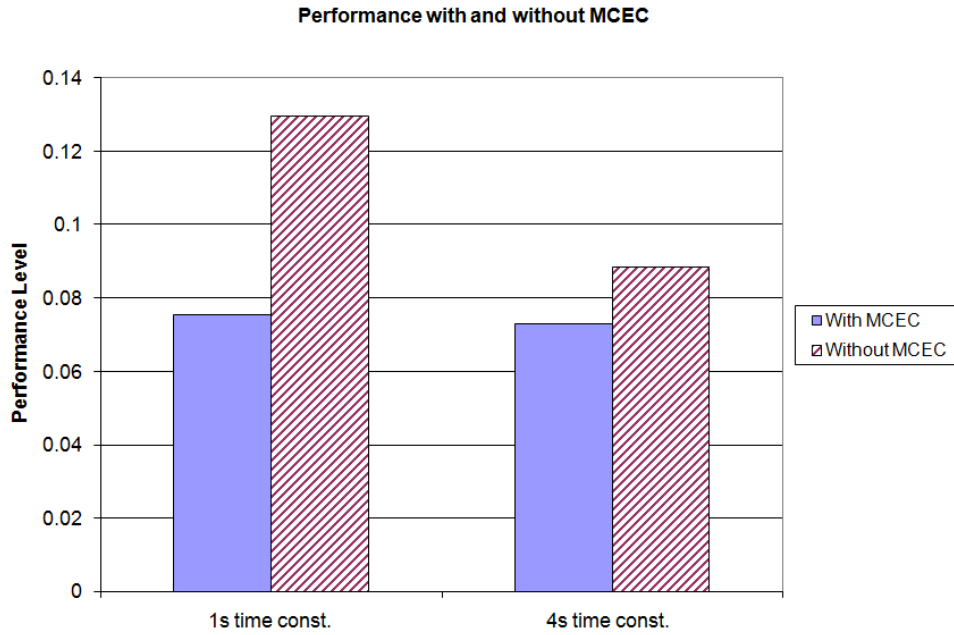


Figure 3.15: Quantitative performance based on the metric specified in (3.67).

## Chapter 4

# Model Predictive Control Allocation (MPCA) for Leveraging Overactuation

While the previous chapter provides a reference model based strategy that results in new theoretical advances in modular control, it does not discuss the details of the control allocation performed by the inner loop. In particular, the results in the previous chapter provide no guidance in terms of selecting an inner loop controller to deliver the desired virtual control input,  $v_{des}^f$ , in the presence of multiple sources of actuation and hard saturation limits on the actuators. This chapter is specifically dedicated to addressing this challenge through a mechanism known as *model predictive control allocation* (MPCA). Specifically, this chapter will show how MPCA can be integrated with the reference model based modular control approach, in order to:

- Optimize inner loop performance in the presence of multiple actuators with hard saturation limits;
- Simultaneously guarantee overall system stability through an appropriate set of MPCA constraints (this chapter will provide the ground work for this development; the complete formulation for stability-constrained MPCA, however, remains a topic of future work).

The development and results for MPCA will consider the discrete-time framework.

### 4.1 MPCA Essentials

MPCA optimizes control inputs over a receding horizon in order to track the desired virtual control input as closely as possible while simultaneously considering actuator saturation constraints, different dynamic authorities, and additional application-dependent performance criteria.

The purpose of the inner loop control allocation is to generate closed-loop behavior for  $\Sigma_2$  that closely matches the inner loop reference model in the presence of saturation constraints and different dynamic actuator authorities. In order to do this, the inner loop controller is formulated as a standard model predictive control (MPC) problem, as described in detail in [60]. The model predictive control *allocation* (MPCA) problem consists of an optimization in which the following cost function is optimized over a receding horizon:

$$\begin{aligned} J(x_2(k), \mathbf{u}(k)) &= \sum_{i=k}^{k+N-1} l(\hat{x}_2(i|k), \hat{v}_{des}^f(i|k), u(i|k)), \\ \mathbf{u}(k) &= [ u(k|k) \quad \dots \quad u(k+N-1|k) ]^T \end{aligned} \quad (4.1)$$

subject to the constraints:

$$u(i|k) \in U = \{u : u_{min,j} \leq u_j \leq u_{max,j}, 1 \leq j \leq p\}, \quad (4.2)$$

where:

$$l(\hat{x}_2(i|k), \hat{v}_{des}^f(i|k), u(i|k)) = (\hat{v}(i|k) - \hat{v}_{des}^f(i|k))^2 + P(\hat{x}_2(i|k), u(i|k)), \quad (4.3)$$

where  $N$  is the length of the prediction horizon. The first control input of  $\mathbf{u}(k)$  is implemented, and the optimization is repeated at step  $k+1$ . Thus, the MPCA control law is:

$$u(k) = u^o(k|k), \quad (4.4)$$

where:

$$\begin{aligned} \mathbf{u}^o(k) &\triangleq \arg \min J(x_2(k), \mathbf{u}(k)) \\ &= [ u^o(k|k) \quad \dots \quad u^o(k+N-1|k) ]^T. \end{aligned} \quad (4.5)$$

The notation  $\hat{v}$  (where  $v$  could be replaced with other variables) denotes a prediction, rather than the actual value, and the notation  $(i|k)$  denotes the value at step  $i$ , where the prediction, or optimization in the case of  $u(i|k)$ , is being made at step  $k$ .

For this work, the  $x_2$  states are not augmented with any additional integrator states (or anything else, for that matter), so we have  $x_2^c = x_2$  and will use  $x_2$  throughout this chapter for simplicity of notation.

The first term in the incremental cost penalizes deviation from the inner loop reference model, whereas the second can be used to shape the response of the closed-loop system. The optimization requires a prediction of  $\hat{v}_{des}^f(i|k), i = 0 \dots N - 1$ , which is accomplished through an outer loop predictor, based on a specific model of the closed outer loop, and given by:

$$\begin{aligned}
 \hat{x}_1(i+1|k) &= f_1(\hat{x}_1(i|k), \hat{v}_{des}^f(i|k), \hat{v}(i|k)), \\
 \hat{v}_{des}(i|k) &= c_1(\hat{x}_1(i|k), \hat{r}(i|k)), \\
 \hat{x}'_1(i|k) &= d(\hat{x}_1(i|k)), \\
 \hat{x}_f(i+1|k) &= f_3(\hat{x}_f(i|k), \hat{v}_{des}(i|k)), \\
 \hat{v}_{des}^f(i|k) &= g_f(\hat{x}_f(i|k)).
 \end{aligned} \tag{4.6}$$

Note that MPCA by its nature automatically derives a prediction of the states over the receding horizon, i.e.,  $\hat{v}(i|k), i = 0 \dots N$ , so there is no need for an additional inner loop predictor. The incorporation of an outer loop predictor results in a modified system diagram, given by Figure 4.1, which preserves the structure of Figure 1.2 but incorporates prediction of  $v_{des}$  over the future horizon. The design of this predictor is within the design scope of the outer loop (hence, it lies within the dashed lines belonging to  $\Sigma_1$ ), because it requires knowledge of the plant and outer loop controller dynamics.

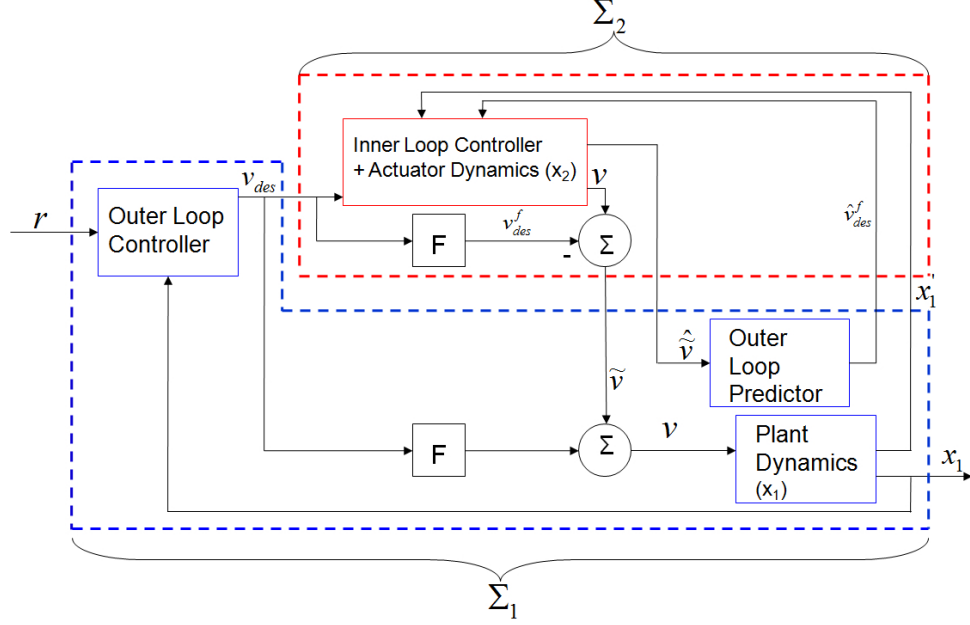


Figure 4.1: Full system diagram, with MPCA and an outer loop predictor, recast for the reference model based design framework

## 4.2 MPCA for the Thermal Management Application

This section considers the thermal management system that was introduced in Chapter 2 and studied in Chapter 3. In this application, however, we will actively use both control inputs,  $\phi$  and  $\dot{Q}_{ht}$ , thus leveraging the system's overactuation via MPCA.

### 4.2.1 Control Design

The controller design is divided into three components, namely the design of the inner loop reference model, the design of the outer loop controller, and the selection of the cost function and horizon length for the MPCA optimization.

#### Inner Loop Reference Model Design

The inner loop reference model is designed, as in Chapter 3, as:

$$\begin{aligned}
 F(s) &= \frac{1}{\tau_r s + 1} = \frac{1}{4s + 1}, \\
 F(z) &= \frac{.22}{z - .78},
 \end{aligned} \tag{4.7}$$



where the implementation sampling time is 1 second.

### Outer Loop Control Design

For the outer loop control design, the closed inner loop dynamics are replaced with our choice of  $F(z)$ , and the control design proceeds using linear system design and analysis tools. Simplicity is an important factor in choosing the outer loop controller, and linear design tools (Root loci and Bode plots) indicate that a PI controller, given by:

$$\begin{aligned} C(s) &= \frac{2s + 0.05}{s}, \\ C(z) &= \frac{2z - 1.95}{z - 1}, \end{aligned} \tag{4.8}$$

(where the implementation sampling time is again 1 second) yields desirable results, as demonstrated earlier in Chapter 3.

### MPCA Optimization

For this application, the inner loop sampling time is taken as 1 second (same as the outer loop), with the cost function specified by (4.1) with  $P(x_2, u) = 0$ . The horizon length for the MPCA optimization is chosen to be 30 seconds (30 steps), which is determined by considering the time constants associated with the inner loop reference model as well as the heater and heat exchanger.

In order to implement the proposed algorithm in real time, it is necessary to choose, among many available MPCA optimization methods (such as sequential quadratic programming (SQP), dynamic programming (DP), or a gradient-based approach), an optimization that is tractable (given the computation and time constraints of the problem, namely a 2 GHz processor and 1s sampling time) and can consider the appropriate constraints (in this case, saturation constraints on  $u$ ). The MPCA optimization used in this paper relies on the computation of the *sensitivity function*, as in [61], which captures the sensitivity of the cost function to the control inputs. This method has been shown in [61] to be more efficient than DP or SQP methods (for a particular large-scale ship application) and is capable of handling saturation constraints. The sensitivity function provides a search direction along which a one-dimensional optimization is performed to determine the minimum value of the cost function. The process is carried out iteratively until the optimal cost converges,

based on a prescribed convergence criterion, or a maximum number of iterations has been reached. Readers are referred to [61] for the details of implementation.

## 4.2.2 Real Time Simulation Results

Prior to implementing the proposed controller on the thermal management system, simulations were performed on the University of Michigan Real Time and Adaptive Control Engineering RACE Lab's OpalRT<sup>TM</sup> real time simulator, in order to:

1. Verify the functionality of the outer loop controller, predictors, and MPCA;
2. Assess the computational efficiency and feasibility of the sensitivity function optimization method on the MPCA problem.

In order to provide a benchmark control strategy against which to compare the MPCA strategy, we consider a closed-form controller, given by:

$$\begin{aligned}\phi(k) &= \text{sat}\left(\frac{1}{T_{out,ht} - T_{out,hex}}\left(\left(1 - \frac{\tau_t}{\tau_r}\right)(T_{out,mv}^{des} - T_{out,mv}) + T_{out,mv}^{des} - T_{out,hex}\right)\right), \\ \dot{Q}_{ht}(k) &= 2.25,\end{aligned}\tag{4.9}$$

which results in closed inner loop performance that *matches* the reference model exactly when saturation is not active. Here, the heater is held at a constant power (one that is desirable for the engine speed and load conditions), which has been employed in previous thermal management strategies due to the difficulty in effectively incorporating the heater into the controller [23].

Real-time simulation results are shown in Figures 4.2-4.4, with the trajectories of the performance variable ( $T_{out,eng}$ ), the control inputs ( $\phi$  and  $\dot{Q}_{ht}$ ), and the virtual control input ( $T_{out,mv}$ ). These simulations are based on a test condition with an engine speed of 2000 rpm and load of 75 N-m. Results demonstrate that reference model based MPCA uses both actuators effectively in order to provide more accurate tracking than under the benchmark (closed form) controller. However, in the case when no reference model is used (where  $F = 1$  and hence  $v_{des}^f = v_{des}$ ), although setpoint tracking is not significantly inhibited, the consequences can begin to be seen at the virtual control input level, and more so at the control input level. Without the reference model, MPCA produces erratic (wildly varying) control inputs in an effort to track achieve an objective that is unrealistic for the system at hand. The effect

can be seen in the longer amount of time required for the virtual control input,  $v$ , to catch up with  $v_{des}$ . While the consequences of not incorporating a reference model into the inner loop design will likely vary by application, this particular case study shows that a properly designed reference model-based MPCA algorithm can alleviate these negative consequences without hindering any aspect of system performance.

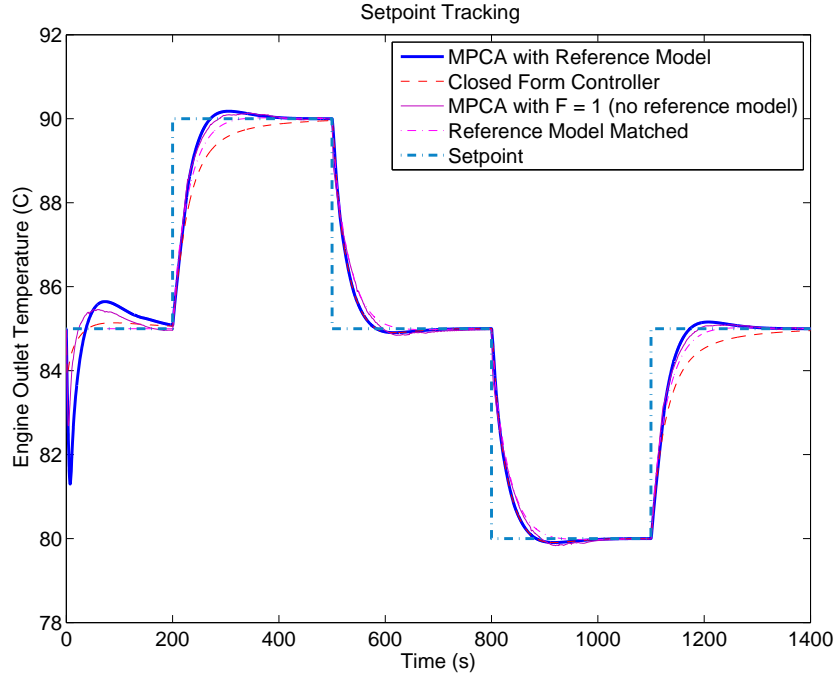


Figure 4.2: Setpoint tracking response - real-time simulation results

Finally, using the real time simulator with the plant model and controller running on separate targets allows us to determine the computational burden seen by the MPCA algorithm (in addition to the outer loop controller and predictors, which contribute a relatively small burden). Figure 4.5 shows the computational time required for the controllers, which is far less than the 0.1s base sample time used for simulation and 1s base controller implementation time, indicating the feasibility of real-time implementation on the actual thermal management system.

### 4.2.3 Experimental Results

Experimental results were acquired for identical test cases as those used in simulation (engine speed of 2000 rpm and load of 75 N-m) and are provided in Figures 4.6-4.8. These experimental results show that MPCA does indeed outperform the

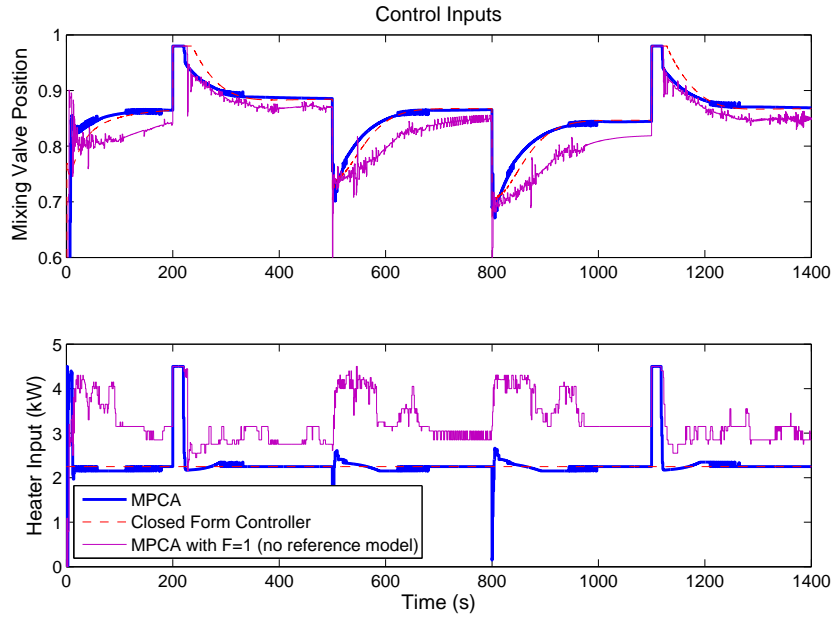


Figure 4.3: Actuator responses - real-time simulation results

benchmark, closed-form controller by making effective use of the heater, particularly in temperature increase responses where the mixing valve saturates. The slight offset between  $v$  and  $v_{des}^f$  throughout Figure 4.8 is due to the fact that there is no integrator for the inner loop, and the inner loop model is not perfect. In this system, the integrator present in the outer loop controller is sufficient for steady-state tracking of the desired engine outlet temperature.

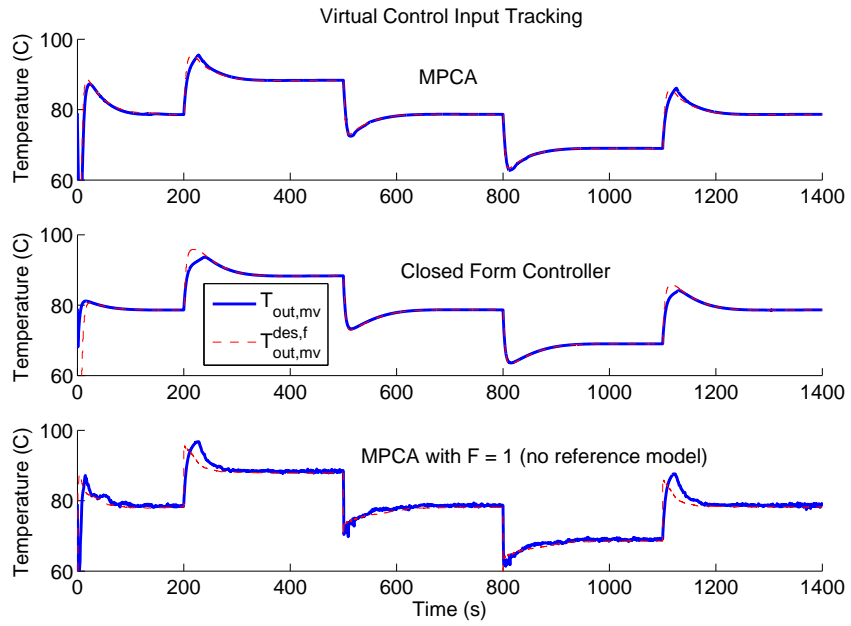


Figure 4.4: Virtual control input response - real-time simulation results. Note that the desired virtual control here represents  $v_{des}^f$ .

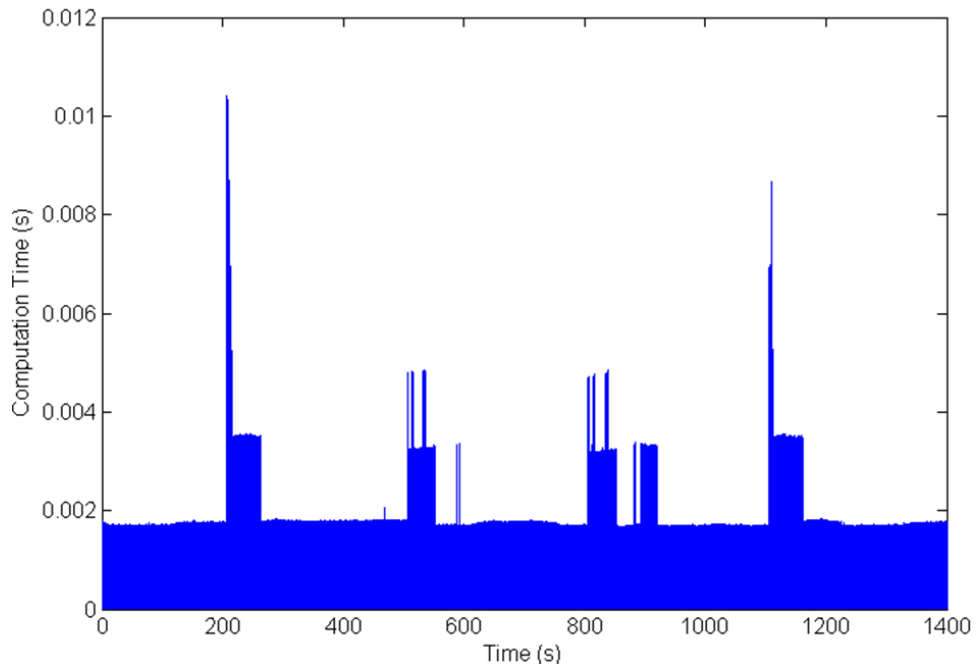


Figure 4.5: Computation time on the real-time simulator

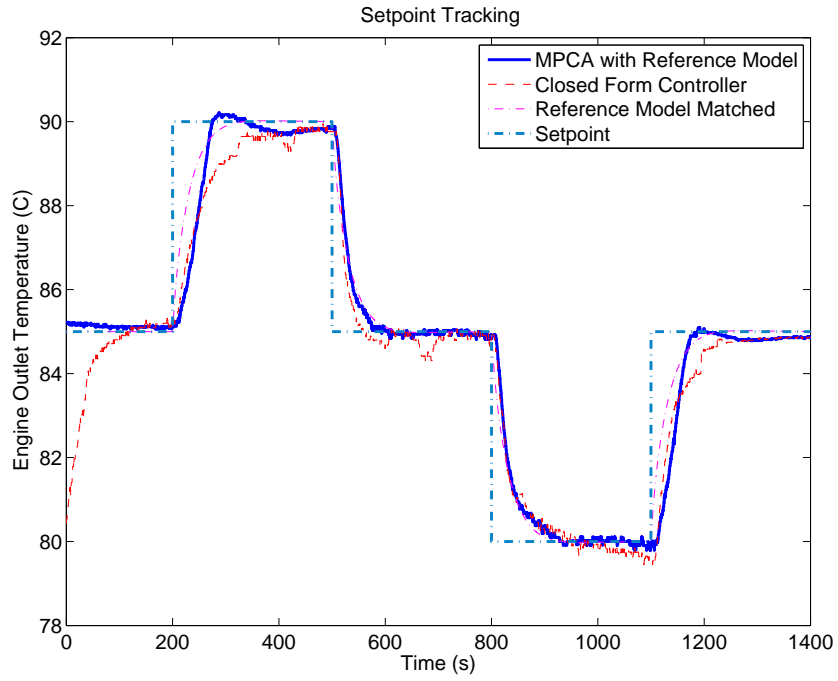


Figure 4.6: Setpoint tracking response - experimental results

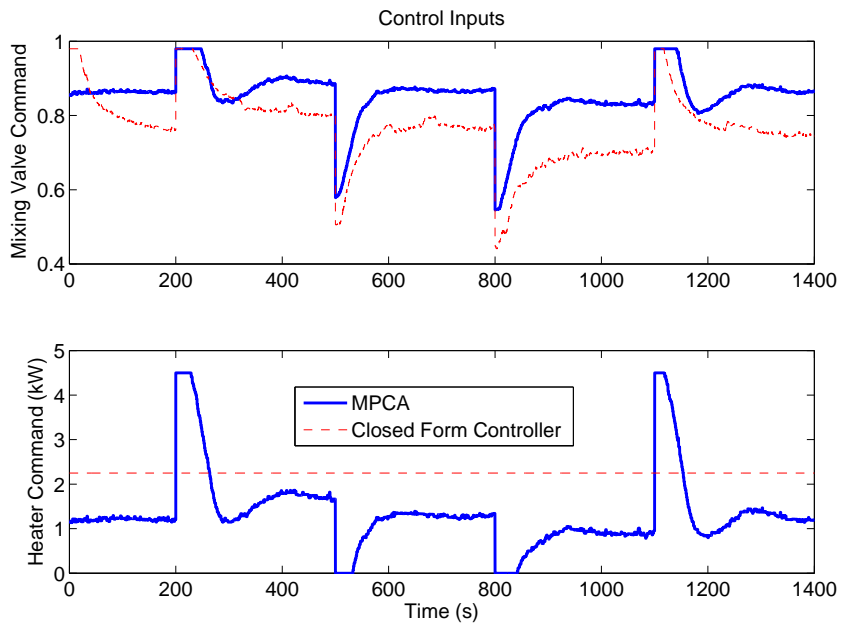


Figure 4.7: Actuator responses - experimental results

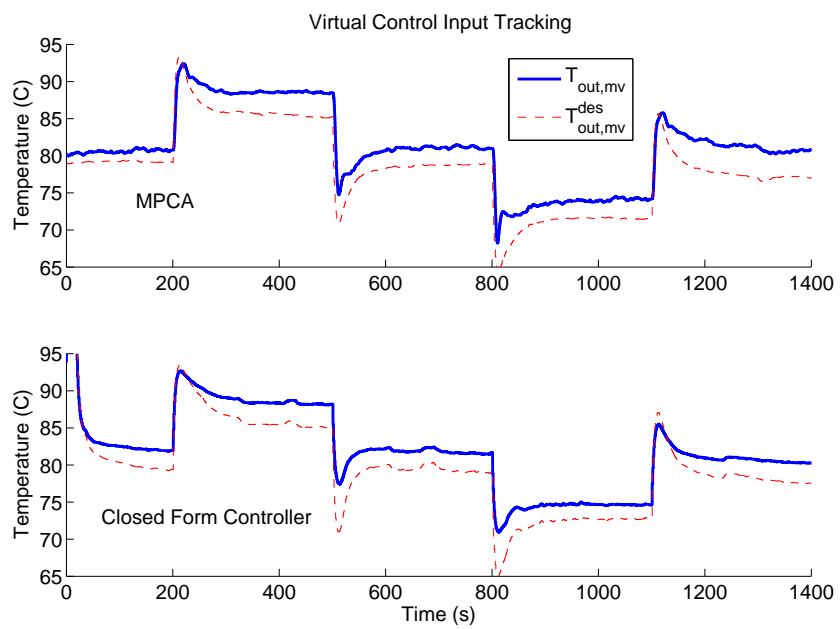


Figure 4.8: Virtual control input response - experimental results. Note that the desired virtual control here represents  $v_{des}^f$ .

## 4.3 Stability-Constrained MPCA

While the results of Section 4.2 suggest that the thermal management system is stable (likely ISS), the results are only presented for a single setpoint trajectory and therefore do not in fact *guarantee* stability of the overall system. Furthermore, a different system may not be stable under MPCA without any additional stability-related constraints. The objective of this section is to lay out the ground work for deriving sufficient, tractable MPCA constraints that will ultimately guarantee overall system stability. In particular, the ultimate objective is to specify the form of the cost function (4.1) and add additional constraints to the saturation constraints of (4.2) which, when enforced, will guarantee overall system stability. While the full result remains an open research topic, this dissertation lays out important ground work that will serve as the foundation for the development of further stability constraints and results.

After establishing preliminary mathematical notions, the thermal management system will be considered. Following this, the ideas from the thermal management system will be extended to a more general framework, and the open issues will become clear.

### 4.3.1 Stability Goal and Mathematical Preliminaries

This dissertation considers only the case of a constant setpoint,  $r$ , and assumes, without loss of generality, that  $r = 0$ , with a corresponding steady-state given by  $x_1 = 0, x_2 = 0$ . The case of varying setpoints is left as an important topic of future work. The primary interest will be to guarantee *asymptotic stability of the origin*, which is defined as follows [62]:

**Definition 4.3.1.** (*Asymptotic Stability*) *The origin,  $x = 0$ , is asymptotically stable if for every  $\epsilon > 0$ , there exists  $\delta_1 > 0$  such that:*

$$\|x(0)\|_2 < \delta_1 \Rightarrow \|x(k)\|_2 < \epsilon, \forall k \geq 0, \quad (4.10)$$

*and there exists  $\delta_2 > 0$  such that:*

$$\|x(0)\|_2 < \delta_2 \Rightarrow \lim_{k \rightarrow \infty} x(k) = 0. \quad (4.11)$$

□

Because asymptotic stability is defined locally in Definition 4.3.1, it is important



to understand how far away from the origin the initial conditions may lie while still guaranteeing convergence to the origin. After all, for initial conditions that are extremely close to the origin, a closed form control law can often be used to guarantee asymptotic stability, without violating saturation constraints. Therefore, it is important to define the notion of *region of attraction*, as follows [62]:

**Definition 4.3.2.** (*Region of Attraction*) *The region of attraction of  $x = 0$  is the largest set,  $R \subset \mathbb{R}^n$ , such that if  $x(0) \in R$ , then  $\lim_{k \rightarrow \infty} x(k) = 0$ .*

□

Given these definitions of asymptotic stability and region of attraction, the goal of stability-constrained MPCA is to *derive additional constraints on the MPCA optimization that guarantee that the origin,  $x = 0$ , is asymptotically stable for a larger region of attraction,  $R$ , than would be possible with a closed form inner loop controller.*

Because  $\Sigma_1$  and  $\Sigma_2$  are interconnected systems, and the designer of one controller may not know the specifics of the other system, it is essential to define conditions for asymptotic stability of  $x = 0$  that are contingent on the individual properties of  $\Sigma_1$  and  $\Sigma_2$ , viewing them as two interconnected systems with “interaction variables” passed between them. Specifically, even though we are considering internal stability of the overall system, each individual subsystem ( $\Sigma_1$  and  $\Sigma_2$ ) *does* in fact have external inputs. Because ISS is equivalent to asymptotic stability of the origin when  $r = 0$ , the ISS conditions of Proposition 3.3.1, taking  $r = 0$ , serve as sufficient conditions for asymptotic stability of the origin. For notational convenience in this section, we will lump  $x$  and  $x_f$  together in the following manner:

$$x_1^{aug} = \begin{bmatrix} x_1^c \\ x_f \end{bmatrix}, x_2^{aug} = \begin{bmatrix} x_2 \\ x_f \end{bmatrix}.$$

The primary tool for verifying stability in the literature is a *Lyapunov function*, and ISS is no exception, where the following proposition yields necessary and sufficient conditions for ISS in terms of a suitably designed Lyapunov function,  $V(x)$  [62]:

**Proposition 4.3.1.** *Considering the system of Proposition 3.3.1, let  $V : \mathbb{R}^n \rightarrow \mathbb{R}$  be a continuously differentiable function such that:*

$$\alpha_1(\|x\|_2) \leq V(x) \leq \alpha_2(\|x\|_2), \tag{4.12}$$

for some class  $K_\infty$  functions  $\alpha_{1,2}$ . Suppose that:

$$\begin{aligned} V(x(k+1)) - V(x(k)) &< -W(x(k)) < 0, \\ \forall \|x(k)\|_2 &\geq \xi(\|u(k)\|_2) > 0, \end{aligned} \tag{4.13}$$

for some continuous positive definite function  $W$  and class  $K$  function  $\xi$ . Then the system is ISS from  $u$  to  $x$ . Furthermore, if the system is ISS, then there exist  $\alpha_{1,2}$ ,  $\xi$ , and  $W$  that satisfy (4.12) and (4.13).

### 4.3.2 Conditions for Deriving Terminal Constraint Sets for Stabilizing MPCA

The use of a terminal constraint set, which is detailed in [60], is a popular method of guaranteeing stability for model predictive control schemes. The idea is to derive a set, containing the origin, which is invariant under a closed form *terminal control law*, then switch from MPC to this control law once the set has been reached. The following definition specifies what is meant by an invariant set:

**Definition 4.3.3.** (Invariant Set): *A set  $G$  is invariant if and only if for every  $x(k) \in G$ ,  $x(k+1) \in G$ .*

By requiring the final state (at step  $k+N$ ) in the MPC optimization lie inside this set, the optimal value of the cost function can be used as a Lyapunov function, and it can be shown that this Lyapunov function is decreasing in time, thereby proving convergence to the terminal constraint set, and, consequently, to the origin.

Because the modular control design is split between an inner and outer loop design, neither of which possesses full state information ( $x_1$  and  $x_2$ ), the formulation for modular control is more complicated. Here, two terminal constraint sets,  $G_1 \in \mathbb{R}^{n_1}$  and  $G_2 \in \mathbb{R}^{n_2}$  will be derived, and simultaneous convergence to each set will ensure stability of the overall system. Additionally, when the inner loop switches to the terminal control law, the states must remain in  $G_{1,2}$  for all future instances. Because the evolution of  $x_1$  depends on  $\tilde{v}$  (an inner loop variable) and the evolution of  $x_2$  depends on  $v_{des}$  and  $x_1'$ , it is typically the case that *any*  $G_{1,2}$  will only be invariant under some restrictions on  $\tilde{v}$ ,  $v_{des}$ , and  $x_1'$ . Therefore,  $G_{1,2}$  will need to be *robustly invariant*, that is, invariant under some restrictions on the disturbances to the subsystems. Robust invariance is defined as follows [63]:

**Definition 4.3.4.** (Robust Invariant Set): Consider a generic system with states  $x(k) \in \mathbb{R}^n$  and exogenous inputs  $d(k) \in D \subset \mathbb{R}^r$ , whose evolution is given by:

$$x(k+1) = f(x(k), d(k)) \quad (4.14)$$

A set  $G$  is robustly invariant if and only if for every  $x(k) \in G$  and  $d(k) \in D$ , we have  $x(k+1) \in G$ .

Thus, the idea here is that if the exogenous inputs are sufficiently small (specifically, that they lie within  $D$ ), then the state will stay within  $G$ . In the case of modular control, the exogenous inputs to one system are outputs of the other system; thus, they are not exogenous to the overall system, and it will be up to the controller (and MPCA constraints) to ensure that these signals eventually become sufficiently small.

Denoting the terminal control law by  $u^r(k)$ , the full inner loop controller can be expressed as:

$$u(k) = \begin{cases} u^r(k) & x_1(k) \in G_1, x_2(k) \in G_2, \\ u^o(k|k) & \text{otherwise.} \end{cases} \quad (4.15)$$

Here,  $u^r(k)$  is the output of a pre-determined closed form control law. We will see that if we can derive this closed form control law, along with sets  $G_1$  and  $G_2$ , such that the following conditions hold, then asymptotic stability of the origin will follow from convergence to these sets:

- *Condition 1* - There exist scalars  $v_{des}^* > 0$  and  $x_1'^* > 0$  such that if  $x_2(k) \in G_2$ ,  $\|x_1'(k)\|_2 < x_1'^*$ , and  $|v_{des}(k)| < v_{des}^*$ , then  $x_2(k+1) \in G_2$ ,  $u^r(k) \in U$ , and the conditions of Proposition 4.3.1 are satisfied for the inner loop.
- *Condition 2* - If  $x_1 \in G_1$ , then  $\|x_1'\|_2 < x_1'^*$  and  $|v_{des}| < v_{des}^*$ .
- *Condition 3* - There exists a scalar  $\tilde{v}^* > 0$ , such that if  $x_1(k) \in G_1$  and  $|\tilde{v}(k)| < \tilde{v}^*$ , then  $x_1(k+1) \in G_1$  and the conditions of Proposition 4.3.1 are satisfied for the outer loop.
- *Condition 4* - When  $u(k) = u^r(k)$  and  $x_2 \in G_2$ , the small gain condition of (3.18)-(3.19) is satisfied.

Condition 1 essentially states that it is possible to derive a robust invariant set for the inner loop with the additional property that ISS conditions must hold within this set. Condition 3 requires a similar outer loop robust invariant set. Referring back to Definition 4.3.4, for the inner loop,  $D$  is given by:

Table 4.1: Generic signals and their thermal management counterparts

Generic Signal	Thermal Management Signal
$v$	$T_{out,mv}$
$v_{des}$	$T_{out,mv}^{des}$
$\tilde{v}$	$T_{out,mv} - T_{out,mv}^{des,f}$
$x'_1$	$T_{out,eng}$

$$D = \{(v_{des}, x'_1) : |v_{des}| < v_{des}^*, \|x'_1\|_2 < x'_1\} \quad (4.16)$$

and for the outer loop,  $D$  is given by:

$$D = \{\tilde{v} : |\tilde{v}| < \tilde{v}^*\} \quad (4.17)$$

Condition 2 states that if  $G_1$  is reached, then outer loop interaction variables are guaranteed to be “small enough.” It will be up to the MPCA optimization to ensure that  $\tilde{v}$  becomes “small enough.” Finally, Condition 4 ensures that that small gain condition holds locally, which is a sufficient condition for asymptotic stability of the origin.

If one can construct  $G_1$  and  $G_2$  so that these conditions all hold, then simultaneous convergence of  $x_1^c$  and  $x_2$  to  $G_1$  and  $G_2$ , respectively, in addition to convergence of  $\tilde{v}$  (such that  $|\tilde{v}(i)| < \tilde{v}^*$  after  $x_{1,2}$  have converged to  $G_{1,2}$ ), will guarantee that the  $x_{1,2}$  remain in  $G_{1,2}$ .

### 4.3.3 Deriving Terminal Constraint Sets for the Thermal Management System

The next task at hand is to propose mechanisms by which  $G_{1,2}$  and the terminal control law,  $u^r$ , can be generated such that conditions 1-4 hold. This is first examined for the special case of the thermal management system. Following the derivation for the thermal management system, a general procedure for constructing  $G_{1,2}$  is provided, noting at that point the roadblocks involved in deriving these constraint sets and in using them as MPCA constraints to guarantee stability.

To begin, Table 4.1 establishes equivalences between generic terms (like  $v_{des}$ ,  $x'_1$ , and  $\tilde{v}$ ) and their specific thermal management signals.

For the outer loop control, the control law of (4.8) is used, whereas the control law of (4.9) will serve as the inner loop terminal control law. To construct  $G_1$  and  $G_2$  such that conditions 1-4 hold, the inner loop is first considered, using a Lyapunov function whose level sets will serve as candidate boundaries for  $G_2$ . This Lyapunov function is chosen as:

$$V_2(x_2) = x_2^T P x_2, \quad (4.18)$$

where:

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 8 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 4 \end{bmatrix} \quad (4.19)$$

which yields  $V_2(x_2(k+1)) - V_2(x_2(k)) < 0$  locally when  $v_{des} = 0$  and  $x_1 = 0$ . Therefore, if  $v_{des}^*$  and  $x_1'^*$  are chosen sufficiently small, then  $G_2$  can be constructed as the interior of a valid level set of  $V_2$ . However, to avoid being burdensome to the outer loop, it is desirable to make  $x_1^*$  and  $v_{des}^*$  as large as possible such that one can still construct  $G_2$  (otherwise  $G_1$  will have to be very small). This can be accomplished by cycling through different bounds on  $|v_{des}|$ , keeping the bound on  $\|x_1\|_2$  fixed at  $x_1'^* = 2.5$ . The results can be seen in Figures 4.9-4.11. These figures show a “slice” of the  $x_2$  state space (so that ellipsoids reproduce as ellipses that can be viewed on a flat page). The boundary of  $G_2$  can be taken as any level set of the Lyapunov function  $V_2$  that does not enter into the “forbidden” region that is marked by  $x$ 's. Note that the actual construction of  $G_2$  must search the entire  $x_2$  space, not just the slice shown in Figures 4.9-4.11.

From the results of Figures 4.9-4.11 (and also considering additional dimensions), the best choice is to take  $v_{des}^* = 2$  degrees, since this results in a relatively large invariant set where saturation constraints are satisfied on the terminal control law. Doing this, the outer loop invariant set,  $G_1$ , is considered next, which is constructed by imposing a bound on  $\tilde{v}$  of  $\tilde{v}^*$ . Figures 4.12-4.14 consider different bounds on  $\tilde{v}^*$ , leading to a conclusion that a bound of  $\tilde{v}^* = 0.3$  degrees is the largest one can obtain, using the proposed quadratic outer loop Lyapunov function,  $V_1(x_1^c) = x_1^{cT} P_1 x_1^c$ .

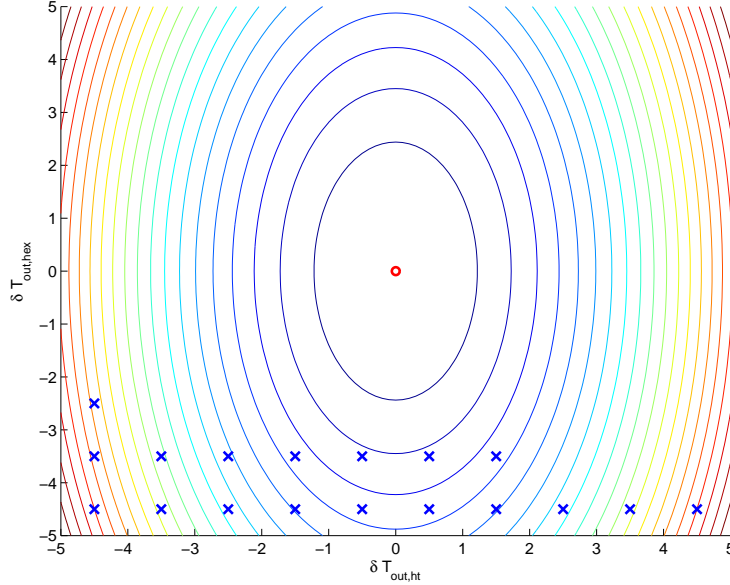


Figure 4.9: Contours of  $V_2$ , with  $x_1^* = 2.5$ ,  $v_{des}^* = 1$ . Blue x indicates where  $u^r \notin U$ , red o indicates where  $V_2$  is non-decreasing.

### 4.3.4 Terminal Constraint Sets for Stabilizing MPCA - Generalization

The previous subsection shows that for the thermal management system, it is possible to derive terminal constraint sets that are invariant under restrictions on the interaction variables,  $v_{des}$ ,  $x_1'$ , and  $\tilde{v}$  (thereby satisfying the requirements of conditions 1-4). In this section, a general procedure is proposed for constructing terminal sets ( $G_1$  and  $G_2$ ), which, when followed, guarantees that conditions 1-4 are satisfied. The procedure is given as follows and is a generalization of what was done for the thermal management system:

- *Step 1 (Design a stabilizing outer loop control)* - Design an outer loop controller,  $v_{des} = c_1(x_1^c(k), r(k))$ , that renders  $\Sigma_1$  ISS.
- *Step 2 (Design a stabilizing inner loop control in the neighborhood of  $x_2 = 0$ )* - Design a control law,  $u^r(k) = c_2^r(v_{des}(k), x_2(k))$ , such that there exists a set  $S$  containing  $x_2 = 0$ , such that when  $x_2(k) \in S$  and  $u(k) = u^r(k)$ , the following are true:
  1.  $u(k) \in U$
  2. There exist a Lyapunov function,  $V_2(x_2)$ , class  $K_\infty$  functions  $\alpha_{1,2}$ , a class  $K$  function  $\xi$ , and a positive definite function  $W$ , such that conditions of (4.12) and (4.13) are fulfilled;
  3. The small gain conditions of (3.18)-(3.19) are fulfilled.

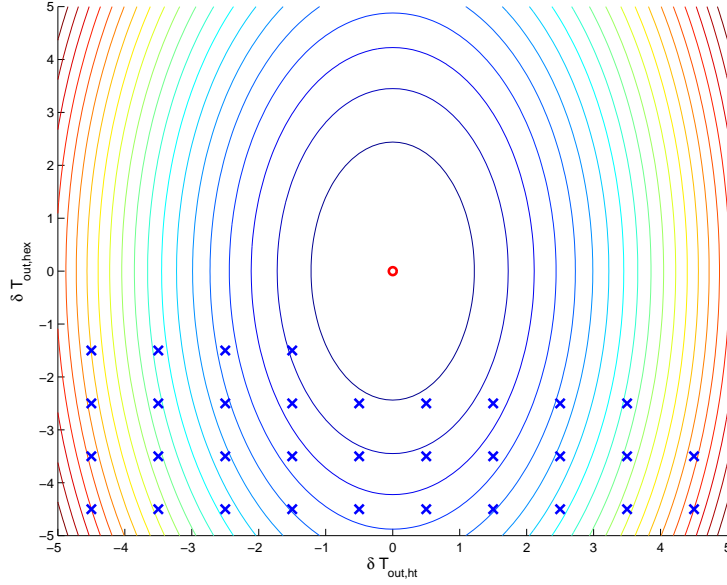


Figure 4.10: Contours of  $V_2$ , with  $x_1'^* = 2.5$ ,  $v_{des}^* = 2$ . Blue x indicates where  $u^r \notin U$ , red o indicates where  $V_2$  is non-decreasing.

In addition, the following assumption is made regarding the outer loop control law and inner loop terminal control law of Steps 1 and 2:

- *Small Control Assumption (small control property of  $u^r$  and  $v_{des}$ )* - There exist scalars  $v_{des}^* > 0$  and  $x_2^* > 0$  such that if  $|v_{des}(k)| < v_{des}^*$  and  $\|x_2(k)\|_2 < x_2^*$ , then  $u^r(k) \in U$ . Furthermore, for every  $\epsilon_1 > 0$ , there exists  $\delta_1 > 0$  such that  $\|x_1^c\|_2 < \delta_1 \Rightarrow |v_{des}(0, x_1^c)| < \epsilon_1$ .

Both steps in the procedure outlined here come with the implicit assumption that the step is in fact feasible. Feasibility of Step 1 requires that a controller can be designed to ensure outer loop ISS, whereas feasibility of Step 2 ultimately requires that there exists a closed-form inner loop controller that satisfies the inner loop ISS condition and small gain condition locally (near the origin). The Small Control Assumption simply requires that  $u^r$  satisfies saturation constraints for sufficiently small  $v_{des}$  and  $x_2$ . Of all of these, Step 2 is the most challenging to verify feasibility for, particularly due to the second part, which requires one to construct several functions to verify local ISS under the terminal control law. The third part of Step 2 is difficult to verify in circumstances where reference model matching cannot be achieved, but is trivial when reference model matching is achieved locally (like the thermal management system), since this yields  $\gamma_2^y = 0$ .

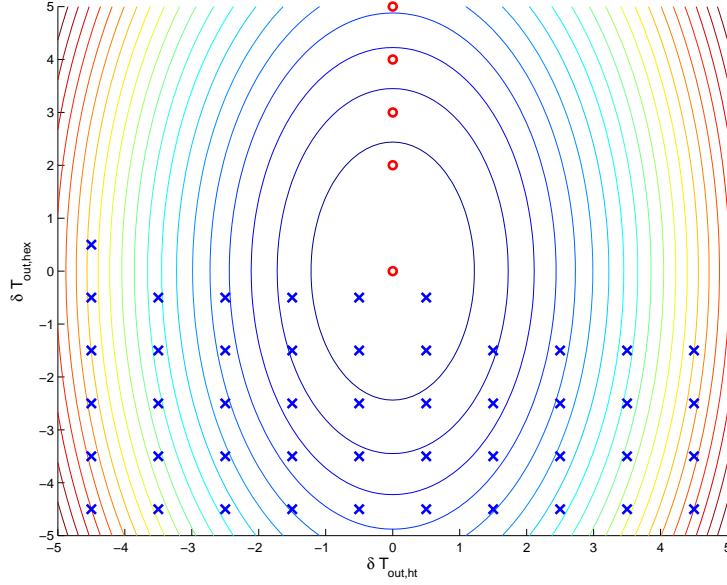


Figure 4.11: Contours of  $V_2$ , with  $x_1'^* = 2.5$ ,  $v_{des}^* = 3$ . Blue x indicates where  $u^r \notin U$ , red o indicates where  $V_2$  is non-decreasing.

By following Steps 1-2, satisfying the small control assumption, and invoking some properties and tools relating to ISS (as established in Proposition 4.3.1), it is now possible to state the following proposition:

**Proposition 4.3.2.** (*Inner Loop Invariant Set*) *Suppose that Steps 1-2 are followed and that the small control assumption holds. Then there exist positive scalars  $V_2^*$ ,  $v_{des}^*$ , and  $x_1'^*$  such that if  $V_2(x_2(k)) \leq V_2^*$ ,  $|v_{des}(k)| \leq v_{des}^*$ ,  $\|x_1'(k)\|_2 \leq x_1'^*$ , and  $u(k) = u^r(k)$ , then  $u(k) \in U$  and  $V_2(x_2(k+1)) \leq V_2^*$ .*

□

**Proof.** *From the small control assumption, there exist  $v_{des}^{**} > 0$  and  $x_2^* > 0$  such that if  $|v_{des}(k)| \leq v_{des}^{**}$  and  $\|x_2(k)\|_2 \leq x_2^*$ , then  $u^r(k) \in U$ . Fixing  $v_{des}^{**}$ , choose  $V_2^*$  such that  $\|x_2\|_2 \leq x_2^*$  whenever  $V(x_2) \leq V_2^*$ . From Step 2, combined with Proposition 4.3.1, there exists  $w > 0$  such that if  $\xi(\| \begin{bmatrix} v_{des}(k) & x_1'^T \end{bmatrix} \|_2) \leq w$  and  $V(x_2(k)) \leq V_2^*$ , then  $V(x_2(k+1)) \leq V_2^*$ . To complete the proof, choose  $v_{des}^*$  and  $x_1'^*$  such that  $v_{des}^* \leq v_{des}^{**}$  and if  $|v_{des}(i)| \leq v_{des}^*$  and  $\|x_1'(i)\|_2 \leq x_1'^*$ , then  $\xi(\| \begin{bmatrix} v_{des}^* & x_1'^*T \end{bmatrix} \|_2) \leq w$ .*

Thus, while the thermal management application example showed one particular case where it was possible to find bounds on  $v_{des}$ ,  $x_1'$ , and  $\tilde{v}$ , such that one could construct an inner loop robust invariant set, Proposition 4.3.2 shows how, in general, this robust invariant set can be constructed.  $G_2$  is constructed as:



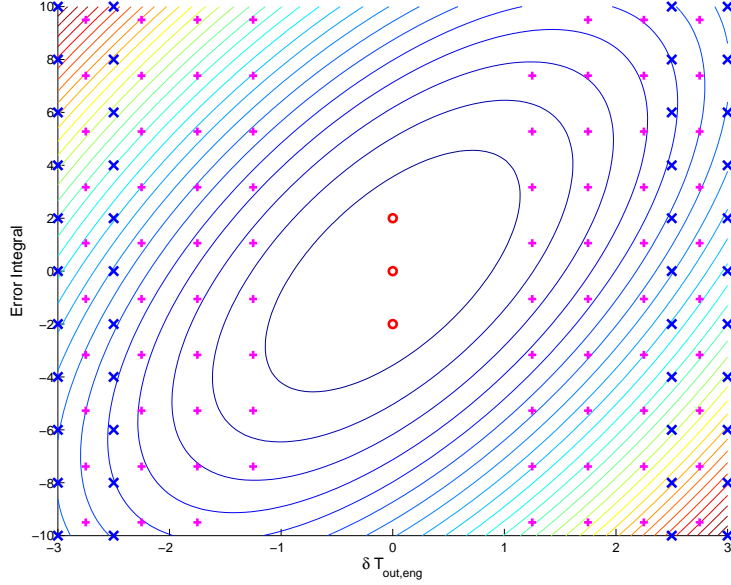


Figure 4.12: Contours of  $V_1$ , with  $\tilde{v}^* = 0.1$ . Blue x indicates where  $x_1' \notin X_1'$ , magenta + indicates where  $v_{des} \notin V_{des}$ , red o indicates where  $V_1$  is non-decreasing.

$$G_2 \triangleq \{x_2 : V_2(x_2) \leq V_2^*\}. \quad (4.20)$$

Because both  $v_{des}$  and  $x_1'$  are functions of  $x_1^c$ , it is possible to construct an analogous set to  $G_2$ , which we refer to as  $G_1$ , as illustrated by the following proposition:

**Proposition 4.3.3.** (*Outer Loop Invariant Set*) *Suppose that Steps 1-2 are followed and the small control assumption holds. Then there exist positive scalars  $V_1^*$  and  $\tilde{v}^*$  such that if  $V_1(x_1^c(k)) \leq V_1^*$  and  $|\tilde{v}(k)| \leq \tilde{v}^*$ , then  $|v_{des}(k)| < v_{des}^*$ ,  $\|x_1'(k)\|_2 < x_1'^*$ , and  $V_1(x_1^c(k+1)) \leq V_1^*$ .*

□

**Proof.** *From the small control assumption, there exists a scalar  $V_1^* > 0$  such that when  $V_1(x_1^c) \leq V_1^*$ , we have  $|v_{des}| < v_{des}^*$  and  $\|x_1'\|_2 < x_1'^*$ . From Step 1, combined with Proposition 4.3.1, there exists  $w > 0$  such that if  $\xi(|\tilde{v}(k)|) \leq w$  and  $V_1(x_1^c(k)) = V_1^*$ , then  $V_1(x_1^c(k+1)) \leq V_1^*$ . We then can choose  $\tilde{v}^*$  to be any positive scalar that satisfies  $\xi(\tilde{v}^*) \leq w$  to complete the proof.*

$G_1$  is then defined as:

$$G_1 \triangleq \{x_1 : V_1(x_1) \leq V_1^*\}. \quad (4.21)$$

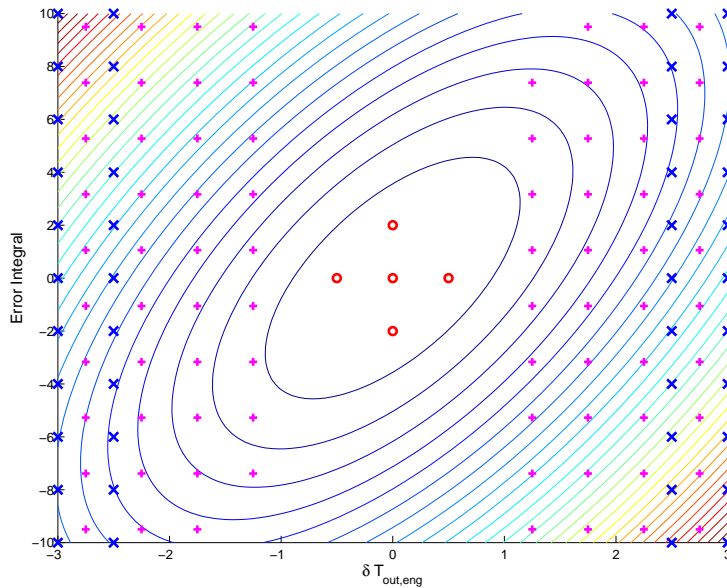


Figure 4.13: Contours of  $V_1$ , with  $\tilde{v}^* = 0.3$ . Blue x indicates where  $x_1' \notin X_1'$ , magenta + indicates where  $v_{des} \notin V_{des}$ , red o indicates where  $V_1$  is non-decreasing.

If initial conditions are taken sufficiently close to the origin, then we have  $x_1^c(0) \in G_1$ ,  $x_2(0) \in G$ , and  $k_{conv} = 0$ , in which case the terminal control law is always active. The purpose of stability-constrained MPCA, however, is to enlarge the region of attraction of  $x = 0$ , such that for initial conditions far from the origin, there exists an instant,  $k_{conv}$  when the states enter into their respective constraint sets and asymptotic stability of the origin follows.

Results at this point only provide conditions under which terminal constraint sets can be *constructed*, as well as a means for constructing them. These terminal sets, however, must be *reached* via an appropriate set of MPCA constraints that are both feasible and tractable. This represents a significant challenge and topic of future work.

### 4.3.5 MPCA Cost Function and Constraints

In the model predictive control literature (as a famous example, [60]), imposing a constraint that  $\hat{x}(k+N) \in G$ , along with a suitable cost function, is sufficient to ensure stability of the origin. Stability is shown by taking the optimal value of the cost function,  $J^o(x(k))$ , as a Lyapunov function, then showing that when we enforce the constraint that  $\hat{x}(k+N) \in G$ , we can guarantee that:

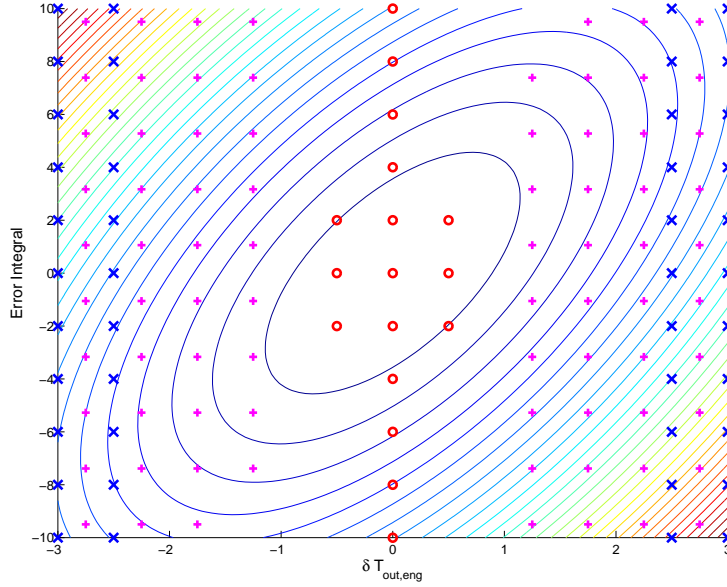


Figure 4.14: Contours of  $V_1$ , with  $\tilde{v}^* = 0.5$ . Blue x indicates where  $x'_1 \notin X'_1$ , magenta + indicates where  $v_{des} \notin V_{des}$ , red o indicates where  $V_1$  is non-decreasing.

$$J^o(x(k+1)) - J^o(x(k)) < 0, \quad (4.22)$$

from which asymptotic stability of the origin follows. The modular control system in question here suffers from two complications, which make the use of a terminal constraint less straightforward in guaranteeing asymptotic stability of  $x = 0$  and ultimately lead us to require additional constraints:

1. The outer loop controller is a closed-form control law. Therefore, all of the constraints must be imposed at the inner loop level, even though we do in fact require convergence to an outer loop constraint set.
2. The predicted trajectory,  $\hat{x}_2$ , is influenced by the predictions of  $v_{des}^f$  and  $x'_1$ . Therefore, if we take  $\mathbf{u}(k+1) = \mathbf{u}(k)$ , we will not necessarily achieve  $\hat{x}_2(k+i|k+1) = \hat{x}_2(k+i|k)$ ,  $i = 1 \dots N-1$  unless  $\hat{v}_{des}^f(k+i|k+1) = \hat{v}_{des}^f(k+i|k)$  and  $\hat{x}'_1(k+i|k+1) = \hat{x}'_1(k+i|k)$  for  $i = 1 \dots N-1$ . One idea for addressing this issue is to use principles from output feedback MPC [64], where if the estimation error is driven to zero, then standard tools can be used to verify stability. In this application, estimation error would be replaced with prediction error (for example, between  $v_{des}^f$  and  $\hat{v}_{des}^f$ ).

Additionally, although convergence of  $x_1$  to  $G_1$  guarantees that  $v_{des}$  and  $x'_1$  will be sufficiently small, convergence of  $x_2$  to  $G_2$  *does not* guarantee that  $\tilde{v}$  will become

sufficiently small; this must be achieved through MPCA constraints. Because these problems have not been fully solved, the remainder of the discussion is deferred to the Future Work Chapter (Chapter 6).

# Chapter 5

## Combined MPCA and MCEC

Chapter 3 presented significant results relating to the reference model based control strategy, including modular control error compensation (MCEC), which significantly improved modular control performance on the thermal management test bed when a closed form, single loop (mixing valve only) inner loop control law was in place. Meanwhile, Chapter 4 presented successful experimental results on the thermal management system using model predictive control allocation (MPCA), in conjunction with the reference model based design approach, to optimally use both actuators in controlling temperature. From a practical perspective, there is no reason why the add-on MCEC compensator cannot be applied when an MPCA based inner loop is in place; this is precisely the topic of this chapter.

While it is theoretically difficult to perform an uncertainty characterization with an optimization-based inner loop, and it is furthermore difficult, if not impossible, to construct weighting functions that capture the effect of saturation on the mismatch between the ideal and actual inner loop, it is still possible to apply MCEC. In the experimental results presented here, we consider the *exact* same compensator,  $C_v$ , that was used in Chapter 3, but now incorporate the MPCA inner loop controller as detailed in Chapter 4. These results are shown in Figures 5.1-5.2, with the analysis of performance, based on the metric:

$$\gamma^{exper} = \frac{(\sum_{i=0}^n ((\delta T_{out,eng}(i) - \delta T_{out,eng}^r(i))^2))^{1/2}}{(\sum_{i=0}^n ((\delta T_{out,eng}^{des}(i))^2))^{1/2}}.$$

given in Figure 5.3. This is the same performance metric that was used in Chapter 3. Results indicate good performance of the combined MCEC and MPCA. These results also indicate the incremental improvement from applying a closed form controller, to using MPCA, to combining MCEC with MPCA.

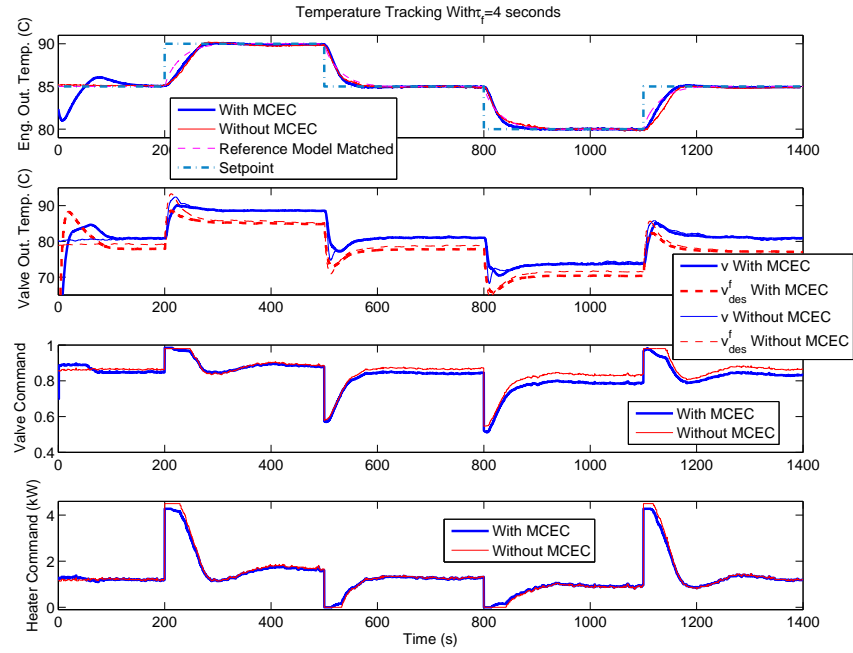


Figure 5.1: Experimental results with combined MCEC and MPCA with a reference model time constant of  $\tau_f = 4$  seconds.

These results indicate that the combination of these control techniques holds promise.

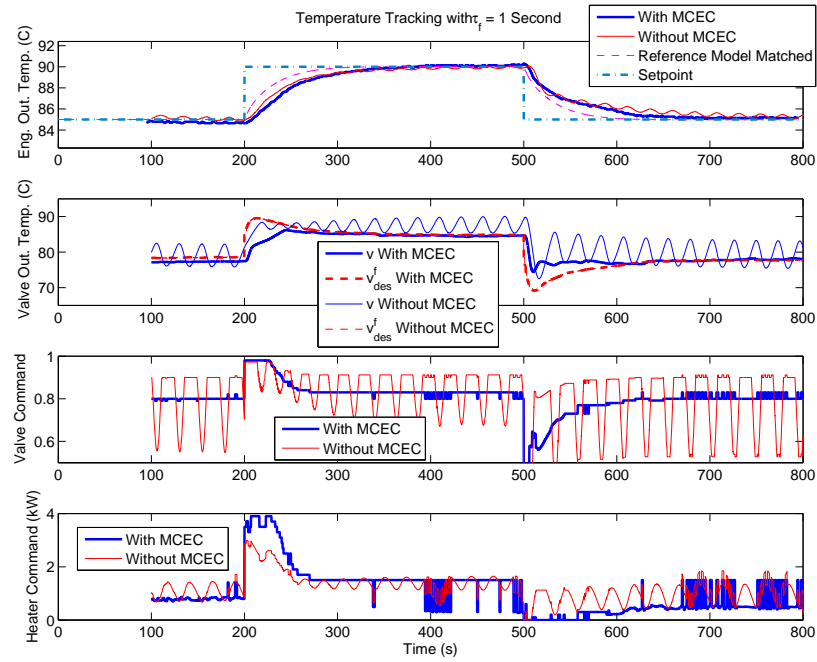


Figure 5.2: Experimental results with combined MCEC and MPCA with a reference model time constant of  $\tau_f = 1$  second.

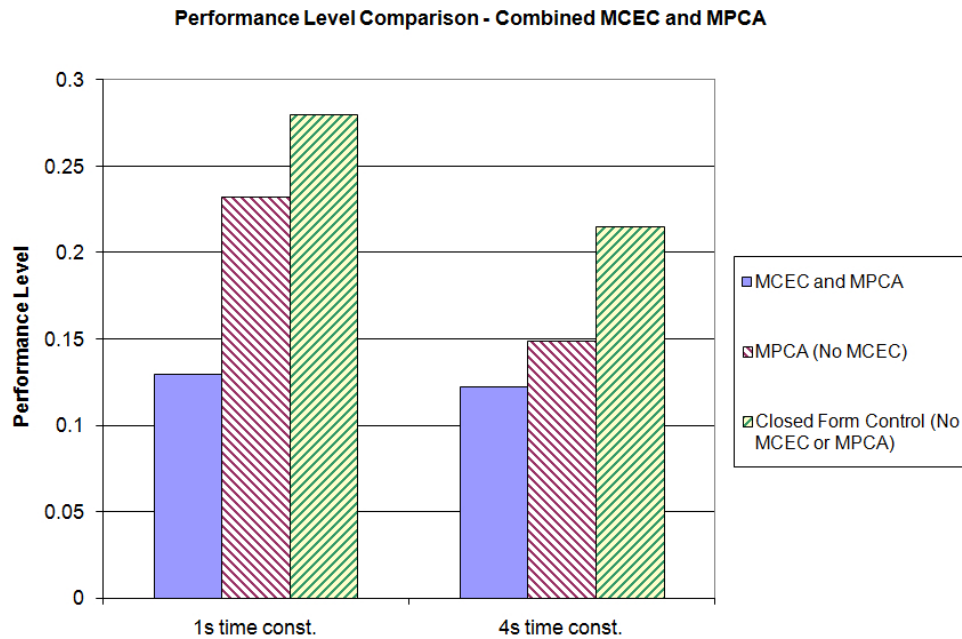


Figure 5.3: Performance with combined MCEC and MPCA.

# Chapter 6

## Conclusions and Future Work

This dissertation has developed a new design framework for modular control of overactuated systems, using an inner loop reference model as an integration mechanism. The reference model based strategy proposed in this dissertation has made it possible to prove theoretical performance properties of the modular control system that have not been derived for previously proposed integration approaches. On top of this strategy, the dissertation has shown the effectiveness, both theoretically and through experimental results, of modular control error compensation (MCEC) on recovering ideal performance of the overall system when the inner and outer loops do not initially work as well as intended when combined. It has also shown the effectiveness of model predictive control allocation (MPCA) as an optimal inner loop control strategy for dealing with overactuation and hard saturation limits. The combination of MPCA with MCEC has shown strong promise through experimental results demonstrated in Chapter 5. Finally, this dissertation established significant ground work in the derivation of stability constraints for MPCA.

### 6.1 Conclusions

We now return to the dissertation contributions claimed in Chapter 1, citing and summarizing the important results related to each, which have been obtained through the course of this dissertation:

- *Developed and validated a dynamic model for the thermal management test system* - A thermal management system was used as a test bed for the experimental validation of all of the proposed control methodologies in this dissertation. Because these methods are all inherently model based, a high fidelity dynamic model is important to their successful implementation. The validation results show, typically, less than 5 degrees (C) variation between predicted and actual



temperature traces, indicating high model accuracy. Furthermore, Chapter 2 has shown that the dynamic analysis of this model has led to the simplification of the experimental setup (considering a single loop, oil in our case) and the identification of MPCA as a strong candidate control methodology for optimally regulating both control inputs.

- *Developed tools for designing modular control systems with guaranteed stability* - Chapter 3 showed how the small gain condition can be combined with the reference model approach to provide a mechanism for verifying overall system input to state stability (ISS) from a setpoint,  $r$ , to the states,  $x$ .
- *Quantified the performance gap between the optimal modular and centralized control designs* - Chapter 3 presented and derived a performance inequality that compares optimal modular control performance to optimal centralized performance. Taking matters further, conditions were derived under which the centralized controller could achieve equivalent performance to its modular counterpart. Specifically, the equivalence approach showed that if the actuator dynamics satisfied certain conditions such that a suitable reference model could be matched by the inner loop, then equivalence followed. Because of the involvement of the reference model in the equivalence proof, the result not only gives conditions on the actuator dynamics under which equivalent performance can be attained; it also provides guidelines on how the reference model based strategy can be used for parallel outer and inner loop controller design, in order to achieve this performance.
- *Developed a mechanism for enhancing overall system performance once the inner and outer loops have been integrated* - Chapter 3 proposed a novel mechanism, referred to as modular control error compensation (MCEC), that exploits the inner loop error signal,  $\tilde{v}$ , in order to adjust the input to the inner loop,  $v_{des}$ , to improve overall system performance. The design methodology was presented, based on  $\mu$  synthesis, which optimizes performance based on two concrete worst case metrics.
- *Experimental validation of modular, closed-form control with MCEC* - The modular control design approach, with and without MCEC, was experimentally validated on the thermal management system described in Chapter 2. Results show strong performance of the overall system when MCEC is in place, for both a typical and aggressive reference model. Results also show a strong benefit of incorporating MCEC for the aggressive reference model, which are demonstrated both qualitatively and quantitatively through simulation and experimental validation.
- *Generalized MPCA to nonlinear actuator dynamics and implemented and experimentally validated MPCA on a thermal management test bed* - Chapter 4 showed, through experimental results on the thermal management system, how model predictive control allocation (MPCA) could be used to optimally modulate both actuators (heater and mixing valve), which are subject to hard saturation constraints and possess significantly different dynamic authorities. Results demonstrate the advantage of using MPCA vs. a single loop controller (mixing valve only).

- *Completed significant ground work in deriving MPCA constraints that will guarantee stability* - Chapter 4 illustrates how terminal constraint sets can be used to guarantee stability of the overall system, with MPCA in place, under a constant setpoint. Further work is required to translate the construction of these terminal constraint sets into a set of feasible, tractable MPCA constraints.
- *Experimentally demonstrated the merits of combined MCEC and MPCA* - The results in Chapter 5 indicate through experimental results on the thermal management system that MCEC can be combined with MPCA to yield better performance than what would be obtained if MPCA were in place alone. This is demonstrated both qualitatively and quantitatively.

## 6.2 Future Work

The material studied in this dissertation has provided new advances in control and also a great deal of future opportunities. Nearly every analysis result and design technique presented in this dissertation, while contributing new results, also leads to the possibility for future related avenues of study. This section will examine each of the major results for which the body of knowledge can be significantly advanced, presenting the major open questions and challenges relating to each.

### 6.2.1 Modular vs. Centralized Performance Comparison for Nonlinear Systems

The modular vs. centralized performance inequality, as well as the equivalence conditions under which the two could yield equal performance, were based on linear systems theory. Specifically, the results of Chapter 3 showed that if the actuator dynamics had the right properties (minimum phase and correct relative degree), then block diagram manipulations could be used in conjunction with an appropriately designed reference model to bound the gap between modular and centralized performance, and, in some cases, prove equivalence. While the concepts of relative degree and stable zero dynamics (minimum phase) exist for nonlinear systems (see Appendix B for treatment of nonlinear reference model matching), the ability to manipulate block diagrams as we had for the linear case goes away. This makes the nonlinear extension of modular vs. centralized results a significant challenge, but an important one.

### 6.2.2 MPCA Stability Constraints

The stability constrained MPCA results of Chapter 4 left off by showing that if inner and outer loop states converge to their respective terminal sets, then asymptotic stability of the origin would follow. We identified two significant challenges that make the derivation of constraints to guarantee entry into these sets very difficult, namely:

1. Deriving sufficient, feasible constraints on the inner loop “interaction variable,”  $\tilde{v}$  that ensure that the outer loop states will converge to a terminal set and therefore guarantee convergence of  $v_{des}$  and  $x'_1$ .
2. Deriving sufficient, feasible constraints on the MPCA optimization such that the *predictions* of the interaction variables,  $\hat{\tilde{v}}$ ,  $\hat{v}_{des}^f$ , and  $\hat{x}'_1$ , converge to their true values.

In addition, future work following the solution to these problems should involve the derivation of stability results for varying setpoints.

### 6.2.3 Stabilization to Manifolds with MPCA

Typically, with overactuated systems, several combinations of steady state control inputs, and consequently several steady states,  $x^{ss}$ , exist for the same performance output,  $y$ . Thus, considering the tracking of a piecewise constant setpoint, steady-state tracking can be obtained by converging to any one of these steady-states, which may lie on a manifold. Because stability-constrained MPCA relies on the convergence of  $x$  to a particular value of  $x^{ss}$ , it is restrictive; other steady-states may be superior, for example, in tracking subsequent setpoints, but have been eliminated from consideration due to the insistence on stabilizing to a point. By stabilizing to a manifold instead, greater flexibility would be allowed, letting the state drift as long as the output stayed constant, potentially putting the system in a better place for subsequent setpoint changes. Therefore, as a stretch goal, it would be of great interest in stability-constrained MPCA to stabilize to a manifold rather than simply a single steady state.

### 6.2.4 Combined MCEC and MPCA

While this dissertation does not provide rigorous stability and performance results for the case when MCEC and MPCA are used together, experimental results do show that the combined use of MCEC and MPCA yields favorable results. These favorable

results came in spite of certain features that could lead to problems. Future work in the area of combining MCEC and MPCA should be aimed at analyzing these features and developing physical insights and design guidelines for the integrated MPCA and MCEC. Specifically, future research should address the following issues:

- *Saturation Limits* - Any time saturation limits exist (with or without MPCA), the inner loop exhibits a nonlinearity, and it is very difficult, if not impossible, to bound the closed inner loop behavior with the uncertainty structure proposed in Chapter 3. Thus, the uncertainty characterization of  $T_1 = F + W_1\Delta_1$ ,  $T_2 = W_2\Delta_2$  fails to characterize the true mismatch between the actual behavior and the reference model.
- *Optimization-based Control* - Since MPCA relies on an optimization, rather than a closed-form control law, it is difficult to characterize the uncertainty in the inner closed loop even without saturation limits. Recall that in Chapter 3, we generated  $W_1$  and  $W_2$  by hypothesizing sources of parametric uncertainty, then generating pseudo-random Bode plots of the inner closed loop under various values of these parameters. This was possible since the inner loop control law was given in closed form. However, this process of generating weighting functions cannot be used if the inner loop controller is instead an optimization. Thus, the design of weighting functions  $W_1$  and  $W_2$  for an optimization based inner loop design remains an open challenge.

## Appendices

# Appendix A

## Thermal Management System Identification

Parameters in the dynamic model were identified experimentally. It should be noted that the fraction of flow passing through the heater and heat exchanger and the temperature of the heater coil or engine block were not measured. In order to identify the model parameters with available measurements, we manipulate the state equations for each component to derive a parametric model in order to use limited measurements. Here, we elaborate on the identification approach that was applied to each component.

### A.1 Heat Exchanger Parameter Identification

For the heat exchanger, the availability of both  $T_{out,hex}$  and  $T_{out,cw}$  as measured variables leads to a simple two-step identification process in which we use steady-state data to identify the specific heats,  $C_{hot/cold}$ , convection coefficients,  $hA_{hot/cold}$ , and conduction coefficients,  $kA_{hot/cold}$ , then use transient data to identify the fluid masses,  $m_{hot/cold}$ . Even though the parameters do not appear linearly in the steady-state model, constrained optimization can be used to identify the parameters by minimizing steady-state modeling error. For this application, this constrained optimization was cast as a minimization of the cost function:

$$J(\hat{\theta}_{hex}) = \sum_{i=1}^N (\epsilon_{i,hot}^2 + \epsilon_{i,cold}^2), \quad (\text{A.1})$$

where:

$$\epsilon_{i,hot} = W_{hot}(T_{out,eng} - T_{out,hex}) - \frac{\hat{U}A}{\hat{C}_{hot}}LMTD, \quad (\text{A.2})$$

$$\epsilon_{i,cold} = W_{cw}(T_{in,cw} - T_{out,cw}) + \frac{\hat{U}A}{\hat{C}_{cold}}LMTD, \quad (\text{A.3})$$

and

$$\hat{\theta}_{hex} = \begin{bmatrix} \hat{h}A_{hot} & \hat{h}A_{cold} & \hat{k}A_{hot} & \hat{k}A_{cold} & \hat{C}_{hot} & \hat{C}_{cold} \end{bmatrix}. \quad (\text{A.4})$$

Once  $\hat{\theta}_{hex}$  is identified, we proceed with the identification of the mass parameters  $m_{hot}$  and  $m_{cold}$  by substituting  $\hat{\theta}_{hex}$  for  $\theta_{hex}$  in the heat exchanger dynamic equations to obtain:

$$\dot{T}_{out,hex} = \frac{1}{m_{hot}}(W_{hot}(T_{out,eng} - T_{out,hex}) - \frac{\hat{U}A}{\hat{C}_{hot}}LMTD), \quad (\text{A.5})$$

$$\dot{T}_{out,cw} = \frac{1}{m_{cold}}(W_{cw}(T_{in,cw} - T_{out,cw}) + \frac{\hat{U}A}{\hat{C}_{cold}}LMTD), \quad (\text{A.6})$$

where the unknown parameters,  $\frac{1}{m_{hot}}$  and  $\frac{1}{m_{cold}}$  appear linearly. Therefore, a standard least squares algorithm can be used from this point forward to identify the unknown masses. To avoid using signal derivatives and reduce noise sensitivities, we apply a first order filter to both sides of these equations before deriving the linear parametric model for identification.

## A.2 Heater Parameter Identification

For a given flow rate, the heater subsystem, with inputs  $T_{out,eng}$  and  $\dot{Q}_{ht}$  and output  $T_{out,ht}$ , may be modeled as a second-order linear system, taking  $\phi W$  as a parameter. This leads to a 2x1 transfer function matrix, which has the following properties:

1. The heater subsystem is linear and second order with respect to the inputs,  $\dot{Q}_{ht}$  and  $T_{out,eng}$ .
2. The transfer functions from  $(\dot{Q}_{ht})$  and  $(T_{out,eng})$  to  $(T_{out,ht})$  have relative degrees 2 and 1, respectively.

3. The DC gain from the input  $T_{out,eng}$  to the heater outlet temperature,  $T_{out,ht}$ , is 1.

Therefore, the heater can be represented by the input-output relation:

$$Y(s) = \frac{b_0}{s^2 + d_1s + d_0}U_1(s) + \frac{c_1s + d_0}{s^2 + d_1s + d_0}U_2(s), \quad (\text{A.7})$$

where  $Y$  is the outlet temperature,  $T_{out,ht}$ , and  $U_1$  and  $U_2$  represent  $\dot{Q}_{ht}$  and  $T_{out,eng}$ , respectively. Well-established methods [47] can be applied to identify  $d_0$ ,  $d_1$ ,  $b_0$ , and  $c_1$ .

The four transfer function coefficients are related to the physical heater model parameters by:

$$d_0 = \frac{m_{fluid}(hA_{ht}W_{ht}^8 + kA_{ht})}{W_{ht}mC_{ht}} \quad (\text{A.8})$$

$$d_1 = \frac{W_{ht}}{m_{fluid}} + \frac{hA_{ht}W_{ht}^8 + kA_{ht}}{mC_{ht}} + \quad (\text{A.9})$$

$$b_0 = \frac{\frac{hA_{ht}W_{ht}^8 + kA_{ht}}{m_{fluid}C_{fluid}}}{mC_{ht}(hA_{ht}W_{ht}^8 + kA_{ht})} \quad (\text{A.10})$$

$$c_1 = \frac{m_{fluid}}{W_{ht}}. \quad (\text{A.11})$$

In the linear heater model, the flow rate through the heater,  $W_{ht}$ , is treated as a parameter that affects the transfer function coefficients. In order to use the linear model to identify heater parameters, identification was performed at various constant  $W_{ht}$  (requiring a constant  $W$  and  $\phi$ ), while engine load,  $L$ , was varied in order to excite  $T_{out,eng}$ , thereby providing the needed excitation for parameter identification. By relying on data collected at different values of  $W_{ht}$ , it was possible to distinguish  $hA_{ht}$  and  $kA_{ht}$  (note that these parameters are indistinguishable through data collected at a single flow rate).

### A.3 Engine Parameter Identification

Although the engine subsystem sees heat rejection,  $\dot{Q}_{eng1,2}$  as disturbance inputs, these disturbances cannot be directly measured and therefore could not be used for



engine parameter identification. As an alternative, the heat rejection rates,  $\dot{Q}_{1,2}$ , were parameterized in terms of engine speed ( $N$ ) and load ( $L$ ), as:

$$\dot{Q}_{eng1} = a_{0,1} + a_{1,1}N + a_{2,1}L + a_{3,1}NL \quad (\text{A.12})$$

$$\dot{Q}_{eng2} = a_{0,2} + a_{1,2}N + a_{2,2}L + a_{3,2}NL. \quad (\text{A.13})$$

A two-step engine parameter identification procedure was followed here. First, steady-state data was used to identify heat transfer parameters and the coefficients for the laws governing heat rejection from the engine ( $\dot{Q}$ ). Then transient data was used to identify mass parameters.

We cast the steady-state and transient parameters in vectors  $\theta_{eng,ss}$  and  $\theta_{eng,trans}$ :

$$\theta_{eng,ss} = \left[ hA^{c/o} \quad kA^{c/o} \quad kA_{cp} \quad a_{i,j} \right], \quad (\text{A.14})$$

and

$$\theta_{eng,trans} = \left[ m^{c/o} \quad mC_{eng1,2} \right]. \quad (\text{A.15})$$

At steady-state, the equations associated with the engine subsystem become:

$$0 = W^c(T_{out,eng}^c - T_{out,mv}^c) + \left( \frac{hA^c(W^c)^{\cdot 8} + kA^c}{C^c} \right) (T_{eng1} - T_{out,eng}^c), \quad (\text{A.16})$$

$$0 = W^o(T_{out,eng}^o - T_{out,mv}^o) + \left( \frac{hA^o(W^o)^{\cdot 8} + kA^o}{C^o} \right) (T_{eng2} - T_{out,eng}^o), \quad (\text{A.17})$$

$$0 = \dot{Q}_1 - (hA^c(W^c)^{\cdot 8} + kA^c)(T_{eng1} - T_{out,eng}^c) + kA_{cp}(T_{eng2} - T_{eng1}), \quad (\text{A.18})$$

$$0 = \dot{Q}_2 - (hA^o(W^o)^{\cdot 8} + kA^o)(T_{eng2} - T_{out,eng}^o) + kA_{cp}(T_{eng1} - T_{eng2}). \quad (\text{A.19})$$

We may express the engine states,  $T_{eng1}$  and  $T_{eng2}$ , in terms of the measured states and  $\theta_{ss}$  as:

$$\begin{bmatrix} T_{eng1} \\ T_{eng2} \end{bmatrix} = \begin{bmatrix} b_1 + b_2 & -b_1 \\ -b_1 & b_1 + b_3 \end{bmatrix}^{-1} \begin{bmatrix} \dot{Q}_1 + b_2 T_{out,eng}^c \\ \dot{Q}_2 + b_3 T_{out,eng}^o \end{bmatrix}, \quad (\text{A.20})$$

where:

$$b_1 = kA_{cp} \quad (\text{A.21})$$

$$b_2 = kA^c + hA^c(W^c)^{.8} \quad (\text{A.22})$$

$$b_3 = kA^o + hA^o(W^o)^{.8} \quad (\text{A.23})$$

We then have 2 steady-state equations that relate  $\dot{Q}_1$ ,  $\dot{Q}_2$ , and  $T_{out,mv}^{c/o}$  to  $T_{eng,out}^{c/o}$  through 13 parameters (5 heat transfer parameters and 8 heat rejection parameters). The steady-state equations do not lead to a linear parametric model. As we did for the heat exchanger validation, off-line optimization was used to identify unknown parameters.

Having identified the coefficients to the heat rejection law, we were able to estimate  $\dot{Q}$  in terms of engine speed and load. The availability of this input and the knowledge of system flow rates ( $W$ ) allows us to treat the engine subsystem as a linear system, with  $W^{c/o}$  taken as parameters, where the transient parameters contained in  $\hat{\theta}_{trans}$  may be identified.

This linear engine subsystem model may be represented in the form of a 4x2 transfer function matrix, partitioned as:

$$\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = \begin{bmatrix} G(\theta_{ss,eng}, \theta_{trans,eng}) \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix}, \quad (\text{A.24})$$

where  $Y_1$  and  $Y_2$  represent the coolant and oil engine outlet temperatures, respectively,  $G$  is a 4x2 transfer function matrix,  $U_1$  and  $U_2$  represent the coolant and oil engine inlet temperatures, and  $U_3$  and  $U_4$  represent  $\dot{Q}_{1,2}$ . The transfer function matrix,  $G$ , contains a total of 24 coefficients which may be identified using a standard least squares technique. We cast these coefficients into a vector,  $c$ . Each element in  $c$  is related to the original physical parameters in  $\theta_{trans,eng}$  by known but nonlinear

relations. Note that  $c$  has dimension of 24 while  $\theta_{trans,eng}$  has a dimension of 4. Therefore, there is no one-to-one mapping between  $c$  and  $\theta_{trans,eng}$  from which  $\theta_{trans,eng}$  can be determined. For our application,  $\theta_{trans,eng}$  is determined by minimizing:

$$J = \|c - c(\hat{\theta}_{trans,eng})\|, \quad (\text{A.25})$$

where  $c$  is the vector resulting from applying standard least squares identification to Equation (A.24).

## A.4 Model Parameters

The system model parameters identified and used for simulation and control development are given in Table A.1. Descriptions of the variable names, with units, are given in Table 2.3.

Table A.1: Model parameters

Subsystem	Parameters	Coolant Value	Oil Value
Mixing Valve	$\tau_t$	5	8
Heat Exchanger	$C$	4.0	3.0
	$C_{cw}$	4.2	4.2
	$m_{hot}$	5.0	1.6
	$m_{cold}$	5.2	1.6
	$hA_{hot}$	1.0	0.1
	$hA_{cold}$	2.0	0.12
	$kA_{hot}$	0	0.01
	$kA_{cold}$	0.25	0.025
Heater	$C$	4.0	3.0
	$m_{fluid}$	10.1	1.7
	$mC_{ht}$	10.2	10.2
	$hA_{ht}$	2.0	0.6
	$kA_{ht}$	0	0.03
Engine	$C^{c/o}$	4.0	3.0
	$m^{c/o}$	25.0	16.7
	$mC_{eng1,2}$	22	3.3
	$hA^{c/o}$	8.2	0.81
	$kA^{c/o}$	0	0.1
	$kA_{cp}$	0.64	0.64

# Appendix B

## Nonlinear Reference Model Matching

This chapter shows how a linear reference model,  $F(s)$  (or a nonlinear one, for that matter), can be matched if the actuator dynamics are nonlinear, but feedback linearizable. The results follow those derived in [62].

Consider inner-loop systems having the following open-loop structure:

$$\begin{aligned}\dot{x}_2 &= f_{2x}(x_2) + f_{2u}(x_2)u, \\ v &= g(x_2).\end{aligned}\tag{B.1}$$

Such systems are said to be *affine* in the control input ( $u$ ) such that feedback linearization may be applied. We extend the standard methods of SISO feedback linearization [62] to the MISO systems considered in this work through the following design algorithm:

1. Design feedback control laws,  $u_i = c_{2i}(x_2, v_{des})$  for  $p - 1$  control inputs, then substitute these feedback laws into the state equations (B.1);
2. Derive a coordinate transformation and feedback linearizing control law with the remaining control input (to be described).

Upon completion of the first step in the design algorithm and substitution of these  $p - 1$  control laws into the original state equations (B.1), we obtain:

$$\begin{aligned}\dot{x}_2 &= f'_{2x}(x_2) + f_{2u}(x_2)u', \\ v &= g(x_2).\end{aligned}\tag{B.2}$$

In order to proceed with the design of a feedback linearizing controller and coordinate

transformation, there must exist a diffeomorphism,  $T(x_2) = [ T_1^T(x_2) \ T_2^T(x_2) ]^T$ , where for  $\Sigma_2$  having relative degree  $\rho_2$  from  $u$  to  $v$ :

$$T_1^T(x_2) = [ \phi_1(x_2) \ \dots \ \phi_{n-\rho_2}(x_2) ]^T, \quad (\text{B.3})$$

$$T_2^T(x_2) = [ g(x_2) \ L_{f_{2x}}' g(x_2) \ \dots \ L_{f_{2x}}'^{\rho_2-1} g(x_2) ]^T. \quad (\text{B.4})$$

The diffeomorphism,  $T(x_2)$  transforms the original system, in  $x_2$  coordinates, into the system:

$$\begin{aligned} \dot{\eta} &= g_0(\eta, \xi), \\ \dot{\xi} &= A_c \xi + B_c \gamma'(x_2)(u' - \alpha(x_2)), \\ v &= \xi_1, \end{aligned} \quad (\text{B.5})$$

where:

$$\begin{aligned} \gamma'(x_2) &= L_{f_{2u}} L_{f_{2x}}'^{\rho_2-1} g(x_2), \\ \alpha(x_2) &= -\frac{L_{f_{2x}}'^{\rho_2} g(x_2)}{L_{f_{2u}} L_{f_{2x}}'^{\rho_2-1} g(x_2)}, \end{aligned}$$

and

$$A_c = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix},$$

$$B_c = [ 0 \ 0 \ 0 \ \dots \ 1 ]^T.$$

These transformed state equations are in the *normal form*, as given by [62], which allows for a straightforward design of a  $u'$  that results in a virtual control input ( $v$ ) that *asymptotically tracks*  $v_{des}$ , so long as the relative degree of the reference model is at least equal to  $\rho_2$ . The discussion of feedback linearizing controller design will

focus on the case where  $\rho_2 = \rho(F(s))$  here.

We may recast the system (B.5) as an error system by defining error coordinates as:

$$\begin{aligned} e &\triangleq [\xi_1 - v_{des}^f \quad \dots \quad \xi_{\rho_2} - v_{des}^{f(\rho_2-1)}]^T \\ &= [\tilde{v} \quad \dots \quad \tilde{v}^{(\rho_2-1)}]^T, \end{aligned} \quad (\text{B.6})$$

such that the system description in error coordinates becomes:

$$\begin{aligned} \dot{\eta} &= g_0(\eta, \xi), \\ \dot{e} &= A_c e + B_c(\gamma'(x_2)(u' - \alpha(x_2)) - v_{des}), \\ \tilde{v} &= e_1. \end{aligned} \quad (\text{B.7})$$

From the results in [62], through the use of a feedback linearizing control law,

$$\begin{aligned} u' &= \alpha(x_2) + \frac{1}{\gamma'(x_2)}(u'' + v_{des}^{(\rho_2)}), \\ u'' &= -Ke, \end{aligned} \quad (\text{B.8})$$

the error state equations reduce to:

$$\begin{aligned} \dot{\eta} &= g_0(\eta, \xi), \\ \dot{e} &= (A_c - B_c K)e, \\ \tilde{v} &= e_1. \end{aligned} \quad (\text{B.9})$$

Note that the state equations in  $e$  are linear, and do not involve an input term. This implies that, through design of a (perfect) feedback linearizing controller,  $v$  will asymptotically track  $v_{des}$  with an  $l_2$  input-output gain of 0, hence achieving reference model matching!

## Bibliography



- [1] M. Naghshineh and M. Keshumiri, "Actuator Saturation Avoidance in Overactuated Systems," *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots*.
- [2] K. Ohishi, H. Nozawa, T. Miyazaki, "Redundant Manipulator Control with Autonomous Consideration Algorithm of Torque Saturation," ISIE 1999.
- [3] K. Butts, N. Shivashankar, J. Sun, "Feedforward and Feedback Design for Engine Idle Speed Control Using  $l_1$  Optimization," *Proceedings of the American Control Conference*, 1995.
- [4] M. Thornhill, S. Thomson, H. Sindano, "A Comparison of Idle Speed Control Schemes," *Control Engineering Practice*, Vol. 8, No. 5, May, 1995.
- [5] J. H. Plumlee, D. M. Bevly, A. S. Hodel, "Control of a Ground Vehicle Using Quadratic Programming Based Control Allocation Techniques," *Proceedings of the American Control Conference*, Boston, MA, 2004.
- [6] O. Harkegard, "Resolving Actuator Redundancy - Control Allocation vs. Linear Quadratic Control," *Proceedings of the European Control Conference*, Cambridge, UK, 2003.
- [7] M. Bodson, "Evaluation of Optimization Methods for Control Allocation," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 4, July-August 2002, pp. 703-711.
- [8] W. Durham, "Constrained Control Allocation," *Journal of Guidance, Control, and Dynamics*, 1993.
- [9] D. Doman, M. Oppenheimer, "Improving Control Allocation Accuracy for Non-linear Aircraft Dynamics," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Monterey, CA, 2002.
- [10] W. Durham, "Computationally Efficient Control Allocation," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 3, 2001.
- [11] J. Buffington, D. Enns, A. Teel, "Control Allocation and Zero Dynamics," *AIAA Guidance, Navigation, and Control Conference*, San Diego, CA, 1996.
- [12] W. Durham, J. Bolling, K. Bordignon, "Minimum Drag Control Allocation," *AIAA Atmospheric Flight Mechanics Conference*, San Diego, CA, 1996.
- [13] O. Harkegard, "Dynamic Control Allocation Using Constrained Quadratic Programming," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 6, 2004.
- [14] Y. Luo, A. Serrani, S. Yurkovich, D. Doman, M. Oppenheimer, "Model Predictive Dynamic Control Allocation with Actuator Dynamics," *Proceedings of the American Control Conference*, Boston, MA, 2004.

- [15] Y. Luo, A. Serrani, S. Yurkovich, D. Doman, M. Oppenheimer, "Dynamic Control Allocation with Asymptotic Tracking of Time-Varying Control Input Commands," *Proceedings of the American Control Conference*, Portland, OR, 2005.
- [16] Y. Luo, A. Serrani, S. Yurkovich, M. Oppenheimer, D. Doman, "Model Predictive Dynamic Control Allocation Scheme for Reentry Vehicles," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 1, 2007, pp. 100-113.
- [17] K. Lindegaard, T. Fossen, "Fuel Efficient Rudder and Propeller Control Allocation for Marine Craft: Experiments with a Model Ship," *IEEE Transactions on Control Systems Technology*, Vol. 11, No. 6, 2003, pp. 850-862.
- [18] T. Johansen, T. Fuglseth, P. Tondel, T. Fossen, "Optimal Constrained Control Allocation in Marine Surface Vessels with Rudders," *Control Engineering Practice*, Vol. 16, No. 4, 2008, pp. 457-464.
- [19] T. Johansen, T. Fossen, S. Berge, "Constrained Nonlinear Control Allocation with Singularity Avoidance Using Sequential Quadratic Programming," *IEEE Transactions on Control Systems Technology*, Vol. 12, No. 1, 2004, pp. 211-216.
- [20] O. J. Sordalen, "Optimal Thrust Allocation for Marine Vessels," *Control Engineering Practice*, Vol. 5, No. 9, 1997, pp. 1223-1231.
- [21] J. Tjonnas, T. Johansen, "Optimizing Adaptive Control Allocation with Actuator Dynamics," *Proceedings of the IEEE Conference on Decision and Control*, New Orleans, LA, 2007.
- [22] S. Schroeck, W. Messner, "On Controller Design for Linear Time-Invariant Dual-Input Single-Output Systems," *Proceedings of the American Control Conference*, Pittsburgh, PA, 1999.
- [23] C. Vermillion, J. Sun, K. Butts, A. Hall, "Modeling and Analysis of a Thermal Management System for Engine Calibration," *Proceedings of the IEEE Conference on Control Applications*, 2006.
- [24] C. Vermillion, J. Sun, K. Butts, F. Biens, "Control Strategies and Experimental Results of an Engine Thermal Management System for Fast and Accurate Dynamometer Engine Mapping," *Proceedings of the Fourth IAV Conference on Design of Experiments*, 2007.
- [25] C. Vermillion, J. Sun, K. Butts, "Modeling, Control Design, and Experimental Validation for an Overactuated Engine Thermal Management System for Engine Dynamometer Applications," *IEEE Journal on Control Systems Technology*, 2009.
- [26] C. Vermillion, J. Sun, K. Butts, "Model Predictive Control Allocation for Overactuated Systems - Stability and Performance," *Proceedings of the IEEE Conference on Decision and Control*, New Orleans, LA, 2007.

- [27] P. Falcone, F. Borrelli, H. Tseng, J. Asgari, D. Hrovat, "A Hierarchical Model Predictive Control Framework for Autonomous Ground Vehicles," *Proceedings of the American Control Conference*, Seattle, WA, 2008.
- [28] R. Scattolini, P. Colaneri, "Hierarchical Model Predictive Control," *Proceedings of the IEEE Conference on Decision and Control*, New Orleans, LA, 2007.
- [29] R. Scattolini, P. Colaneri, D. Vito, "A Switched MPC Approach to Hierarchical Control," *Proceedings of the International Federation of Automatic Control (IFAC) World Congress*, Seoul, Korea, 2008.
- [30] R. Sonntag, C. Borgnakke, *Introduction to Engineering Thermodynamics*, 2002.
- [31] V. Tsourapas, "Modeling, Optimization, and Control of a Combined Heat and Power Fuel Processor for Hydrogen Generation From Natural Gas," Master's Thesis in Mechanical Engineering, University of Michigan, 2005.
- [32] P. Setlur, J. Wagner, D. Dawson, J. Chen, "Nonlinear Controller for Automotive Thermal Management Systems.," *Proceedings of the American Control Conference*, 2003.
- [33] H. Couetouse, D. Gentile, "Cooling System Control in Automotive Engines," SAE paper no. 920788, 1992.
- [34] J. Wagner, I. Paradis, E. Marotta, D. Dawson, "Enhanced Automotive Engine Cooling Systems - A Mechatronics Approach," *Int. Journal of Vehicle Design*, vol. 28, no. 1-3, 2002.
- [35] J. Wagner, M. Ghone, D. Dawson, E. Marotta, "Coolant Flow Control Strategies for Automotive Thermal Management Systems", SAE paper 2002-01-0713, 2002.
- [36] P. Setlur, J. Wagner, D. Dawson, E. Marotta, "An Advanced Engine Thermal Management System: Nonlinear Control and Test", *IEEE/ASME Transactions on Mechatronics*, Vol. 10, No. 2, April 2005.
- [37] X. Zou, J.A.Jordan, M. Shillor, "A Dynamic Model for a Thermostat," *Journal of Eng. Math.*, vol. 36, pp. 291-310, 1999.
- [38] G. Einaudi, W. Mortara, "Engine Cooling Electronic Control System," SAE Paper 885085, 1988.
- [39] A. Kenny, C. Bradshaw, B. Creed, "Electronic Thermostat System for Automotive Engines," SAE Paper 880265, 1988.
- [40] D. Liberzon, A. Morse, E. Sontag, "Output-input Stability and Minimum-phase Nonlinear Systems," *IEEE Transactions on Automatic Control*, Vol. 47, No. 3, pp. 422-436, 2002.

- [41] D. Angeli, E. Sontag, Y. Wang, "Further Equivalences and Semiglobal Versions of Integral Input to State Stability," *Journal of Dynamics and Control*, Vol. 10, 2000.
- [42] Z. Jiang, Y. Wang, "Input-to-state Stability for Discrete-time Nonlinear Systems," *Automatica*, Vol. 37, pp. 857-869, 2001.
- [43] E. Sontag, "On the Input-to-State Stability Property," *European Journal of Control*, 1995.
- [44] Z.-P. Jiang, A. R. Teel, L. Praly, "Small-Gain Theorem for ISS Systems and Applications," *Mathematics of Control, Signals, and Systems*, pp. 95-120, 1994.
- [45] D. Nesić, P. Dower, "A Note on Input-to-State Stability and Averaging of Systems with Inputs," *IEEE Transactions on Automatic Control*, Vol. 46, No. 11, 2001.
- [46] S. Huang, M. James, D. Nesić, P. Dower, "Analysis of Input to State Stability for Discrete Time Nonlinear Systems via Dynamic Programming," *Proceedings of the IEEE Conference on Decision and Control*, 2003.
- [47] P. Ioannou, J. Sun, *Robust Adaptive Control*, Prentice Hall, 1996.
- [48] G. Balas, J. Doyle, K. Glover, A. Packard, R. Smith,  *$\mu$  Synthesis and Analysis Toolbox*, The Math Works, 2001.
- [49] K. Zhou, J. Doyle, and K. Glover, *Robust and Optimal Control*, Prentice Hall, 1995.
- [50] K. Zhou, J. Doyle, *Essentials of Robust Control*, Prentice Hall, 1997.
- [51] B. Boulet, B. A. Francis, P. C. Hughes, T. Hong, " $\mu$  Synthesis for a Large Flexible Space Structure Experimental Testbed," *Journal of Guidance, Control, and Dynamics*, Vol. 24, 2001.
- [52] M. Steinbuch, P. Van Gross, G. Schootstra, P. Wortelboer, O. Bosgra, " $\mu$  Synthesis of a Compact Disc Player," *International Journal of Robust and Nonlinear Control*, Vol. 8, 1998.
- [53] J. Bibel, H. Stalford, " $\mu$  Synthesis Autopilot Design for a Flexible Missile," *Aerospace Sciences Meeting*, Reno, NV, 1991.
- [54] J. Freudenberg, B. Morton, "Robust Control of a Booster Vehicle Using H and SSV Techniques," *Proceedings of the IEEE Conference on Decision and Control*, Tucson, AZ, 1992.
- [55] R. Adams, S. Banda, "Robust Flight Control Design Using Dynamic Inversion and Structured Singular Value Synthesis," *IEEE Transactions on Control Systems Technology*, Vol. 1, 1993, pp. 80-92.

- [56] J. Sun, "A Modified Model Reference Adaptive Control Scheme for Improved Transient Performance," *IEEE Transactions on Automatic Control*, Vol. 38, No. 8, 1993.
- [57] C. Vermillion, J. Sun, K. Butts, "Performance Enhancement of Modular Control Systems Using  $\mu$  Synthesis," *Proceedings of the IEEE Conference on Decision and Control*, Cancun, MX, 2008.
- [58] R. Holland, P. Young, and C. Zhu, "Development of a Skew  $\mu$  Upper Bound," *Proceedings of the European Control Conference*, Denver, CO, 2003.
- [59] G. Ferreres, J. Biannic, J. Magni, "A Skew  $\mu$  Toolbox for Robustness Analysis," *Proceedings of the IEEE Symposium on Computer Aided Control Systems Design*, Taipei, Taiwan, 2004.
- [60] D. Q. Mayne, J. B. Rawlings, C. V. Rao, P. O. M. Scokaert, "Constrained Model Predictive Control: Stability and Optimality," *Automatica*, Vol. 36, pp. 789-814, 2000.
- [61] G. Seenumani, J. Sun, H. Peng, "A Numerically Efficient Iterative Procedure for Hybrid Power System Optimization Using Sensitivity Functions," *Proceedings of the American Control Conference*, 2007.
- [62] H. Khalil, *Nonlinear Systems*, Prentice Hall, 2002.
- [63] F. Blanchini, "Set Invariance in Control," *Automatica*, Vol. 35, pp. 1747-1767, 1999.
- [64] R. Findeisen, L. Immanuel, F. Allgower, B. Foss, "State and Output Feedback Nonlinear Model Predictive Control: An Overview," *European Journal of Control*, Vol. 9, pp. 190-206, 2003.