Composite Adaptive Internal Model Control: Theory and Applications to Engine Control

by

Zeng Qiu

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering: Systems)
in The University of Michigan
2017

Doctoral Committee:

 Professor Jing Sun, Co-Chair
 Doctor Mrdjan Jankovic, Ford Motor Company, Co-Chair
 Professor Kira Barton
 Professor Jessy Grizzle

Zeng Qiu

connieqz@umich.edu

ORCID iD: 0000-0002-5907-2741

To my family, for their love and support.

And to John, for always believing in me.

ACKNOWLEDGEMENTS

I feel fortunate for having spent the last few years with my advisors, teachers, friends, and colleagues in the University of Michigan, Ann Arbor. The relationships that we built together will forever be my treasure.

This dissertation would not have been possible without my advisor Prof. Jing Sun. She taught me Adaptive Control, led me into the world of research, and challenged me to be a better researcher. Her work ethics will continue to be an inspiration for my future. I thank my advisor Dr. Mrdjan Jankovic from Ford Motor Company. He is not only a brilliant control theorist, but also a master of automotive control applications, who led a great career path for me to follow. Technical discussions with him always spurred inspirations.

I am very grateful to Prof. Jessy Grizzle, who was the best and most friendly teacher that I had. I am especially thankful to him for giving me opportunity to be the graduate student instructor of Nonlinear System Theory. I would like to thank my committee member Prof. Kira Barton, for her interest and encouragement in my work. I am thankful to Prof. Johanna Mathieu, for offering me life and work advices in the gEECS Mentor/Mentee program. I thank Prof. Semyon Meerkov, for advising me on research in Production System Analysis.

I am grateful to the RACE lab members: Caihao, Dave, Esteban, Hao, Hyeongjun, Jun, Kai, Richard, Yujia, and Zhenzhong. I was fortunate for having had the summer internship position at Ford Motor Company, where I had the privilege of implementing my control algorithms on real vehicles. I want to thank my brilliant colleagues at Ford:

Amey, Baitao, Erik, Hamid, Henry, Julie, and Mike. I especially want to thank my summer internship advisor Dr. Mario Santillo, who was also a great friend. Without him, I would not have been able to obtain the test results on time.

Yash, Umair, and Jay, I immensely enjoyed the time when we studied together as control theory beginners. Kangwei, Wei, Shengyin, Jianfeng, Yuxi, Chenyang, Shiyang, Yang, and Yue, thanks for being fantastic roommates and/or "meal partners". Thanks to Yelin, Yi-Chin, Parinaz, Mai, Zhaojian, Pengchuan, Yao, Tru, Xinyi, Wubing, Shankar, Yousaf, and so many other friends and colleagues. Thanks also to the Shenmue crew: Alex L., Jon, Greg, Alex B., Max, and Juice, for showing me fun on the streets of Hong Kong.

I would like to thank John Forrest, for being a wonderful companion through my PhD journey, for sharing the joy of my achievements, for offering encouragements during my struggles, and for all his love. My PhD experience would not have been so pleasant without him. I appreciate the support from the Forrest family.

Last but not least, I would like to thank my family: my mother Shuying Yang, my father Yuzhong Qiu, my brother Zhi Qiu, and my beloved grandparents Fengling Zhao and Hongji Yang, for their unconditional love and support. I wish Zhi will enjoy his school time in America as much as I did.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

LIST OF APPENDICES

Appendix

LIST OF ABBREVIATIONS

AIMC: Adaptive internal model control

CAIMC: Composite adaptive internal model control

CAIMC-LI: CAIMC with left inverse

CAIMC-RI: CAIMC with right inverse

IMC: Internal model control

LHP: Left half plane

LMI: Linear matrix inequality

LPV: Linear parameter varying

LTI: Linear time invariant

LTV: Linear time varying

MIMO: multiple-input, multiple-output

MP: Minimum phase

NMP: Non-minimum phase

PCM: Powertrain control module

PE: Persistently exciting

RHP: Right half plane

SISO: Single-input, single-output

ABSTRACT

Composite Adaptive Internal Model Control: Theory and Applications to Engine Control

by

Zeng Qiu

Co-Chairs: Prof. Jing Sun and Dr. Mrdjan Jankovic

To meet customer demands for vehicle performance and to satisfy increasingly stringent emission standard, powertrain control strategies have become more complex and sophisticated. As a result, controller development and calibration have presented a time-consuming and costly challenge to the automotive industry. This thesis aims to develop new control methodologies with reduced calibration effort. Internal model control (IMC) lends itself to automotive applications for its intuitive control structure with simple tuning philosophy. A few applications of IMC to the boost-pressure control problem have been reported, however, none offered an implementable and easy-to-calibrate solution. Motivated by the need to develop robust and easily calibratable control technologies for boost-pressure control of turbocharged gasoline engines, this thesis developed new control design methodologies in the IMC framework. Two directions are pursued: adaptive IMC (AIMC) and nonlinear IMC.

A plant model and a plant inverse are explicit components of IMC. In the presence of plant-model uncertainty, combining the IMC structure with parameter identification through the certainty equivalence principle leads to adaptive IMC (AIMC), where

the plant model is identified and the plant inverse is derived by inverting the model. We propose the composite AIMC (CAIMC), which identifies the model and the inverse in parallel, and reduces the tracking error through the online identification. "Composite" refers to the simultaneous identifications. The constraint imposed by the stability of an n-th order model is nonconvex, and it is re-parameterized as a linear matrix inequality. The parameter identification problem with the stability constraint is reformulated as a convex programming problem. Stability proof and asymptotic performance are established for CAIMC of a general n-th order plant.

CAIMC is applied to the boost-pressure control problem of a turbocharged gasoline engine. It is first validated on a physics-based high-order and nonlinear proprietary turbocharged gasoline engine Simulink model, and then validated on a turbocharged $2L$ four-cylinder gasoline engine on a Ford Explorer EcoBoost. Both simulations and experiments show that CAIMC is not only effective, but also drastically reduces the calibration effort compared to the traditional PI controller with feedforward.

Nonlinear IMC is presented in the context of the boost-pressure control of a turbocharged gasoline engine. To leverage the available tools for linear IMC design, the quasi-linear parameter varying (quasi-LPV) models are explored. A new approach for nonlinear inversion, referred to as the structured quasi-LPV model inverse, is developed and validated. A fourth-order nonlinear model which sufficiently describes the dynamic behavior of the turbocharged engine is used as the design model, and the IMC controller is derived based on the structured quasi-LPV model inverse. The nonlinear IMC is applicable when the nonlinear system has a special structural property and has not been generalized yet. Simulations on a high-fidelity turbocharged engine model are carried out to show the feasibility of the proposed nonlinear IMC.

CHAPTER I

Introduction

1.1 Motivation

The automotive industry had made tremendous progress over the last few decades in terms of vehicle performance and energy efficiency. As shown in Fig. 1.1, while the average weight of a vehicle is similar between model year 2016 and 1975, the horsepower and fuel economy are improved significantly. For more efficient vehicles, many improved powertrain technologies have been adopted, for example, variable valve timing, direct fuel injection, engine down-sizing with turbochargers [2]. As the powertrain complexity increases, in order to meet the legislative emission standards, the number of calibration parameters of the powertrain control increases exponentially as shown in Fig. 1.2. The "curse of dimensionality" drives the calibration process lengthier, more expensive, and labor-intensive [4]. The calibration process typically starts with many months of experimental work in engine test cells and experimental vehicles. The finalization of the control and diagnosis can easily take two to three years [5].

The automotive control problem is out-pacing the traditional control design technique, which poses challenges and opportunities for control technology development [5]. Control frameworks that provide easy calibratability are essential for the way of working in the automotive industry [6]. Internal model control (IMC) lends itself
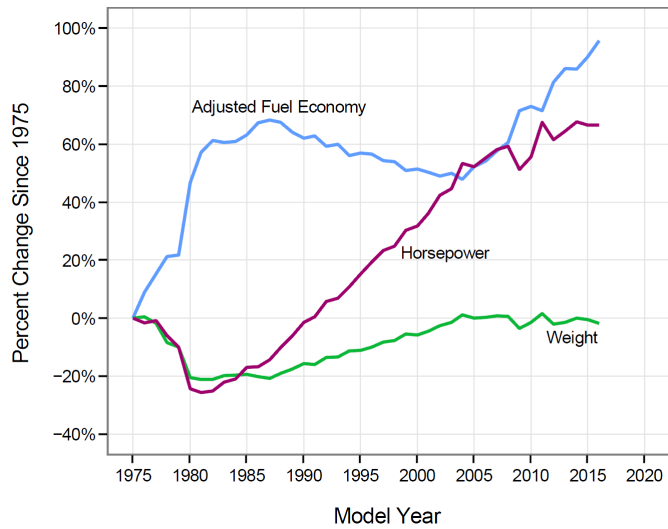
Figure 1.1: Average change in adjusted fuel economy, weight, and horsepower of a vehicle since 1975 [1].
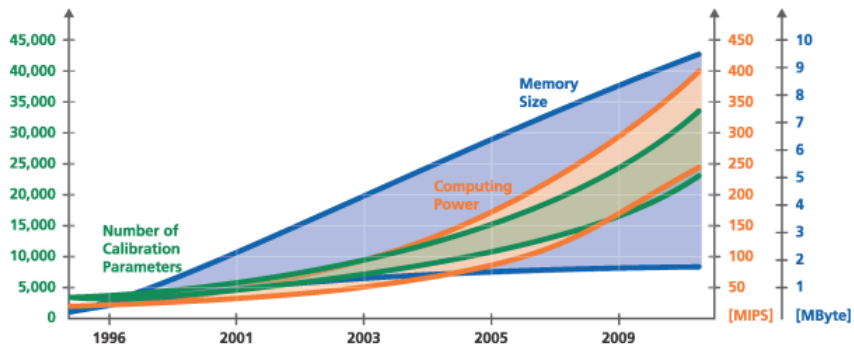
Figure 1.2: Progress of memory size (in MByte), calibration parameters (number), and calculation power (in mega instructions per second / MIPS) of engine control units since 1995 [3].
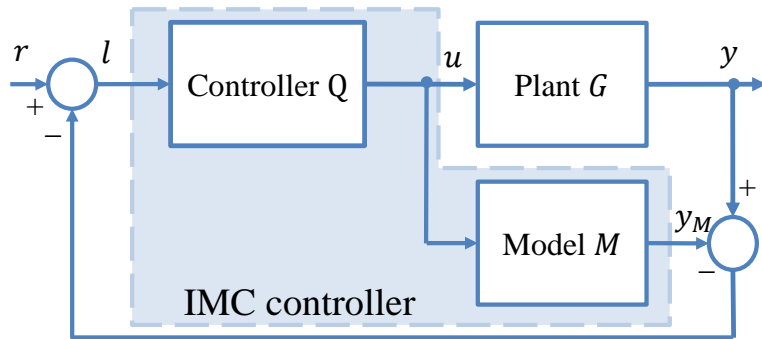
Figure 1.3: Internal model control structure.

to automotive industry for its intuitive control structure with simple tuning philosophy [7,8].

As shown in Fig. 1.3, IMC incorporates the plant model as an explicit part of the IMC controller, and the feedback is the difference between the plant and the plant model responses. When the model is the same as the plant, the feedback will cease to have an effect and the IMC structure will be equivalent to an open-loop feedforward control [9]. The controller Q in Fig. 1.3 can be simply chosen as the approximate inverse of the model. IMC offers several salient features that make it attractive, especially from the controller tuning point of view. When the plant is stable and minimum phase, this inverse can be obtained by inverting the plant model M and augmenting it with a low pass filter, where the latter is used to assure causality of the controller. The time constant of the filter, as the only tuning parameter in this design, can be calibrated to achieve the desired bandwidth of the control system, thereby ensuring robustness [10, 11]. IMC has been applied to many automotive applications, such as boost-pressure control [12–17], air/fuel ratio control [18–20], idle speed control [21], throttle control [22], traction control [23], and vehicle yaw control [24].

1.2 Background on Internal Model Control

IMC was first introduced by Garcia and Morari in 1982 [9], but similar concepts were used previously and independently by other researchers. Richalet et al. [25] and Cutler and Ramaker [26] proposed similar but heuristic control algorithms in the process industries, which were applied successfully to complex process control applications. Morari and Zafiriou put IMC on a firmer footing in [8].

The design, analysis, and implementation of IMC for linear systems have been well explored [8]. It has numerous successful applications [27–32]. Besides being directly applied, many linear IMCs are shown to lead to PID controllers, occasionally augmented with a first-order lag filter. The superiority of using IMC for PID tuning is demonstrated in terms of closed-loop performance and robustness [33, 34].

For nonlinear plants, the results and tools for IMC design and implementation are very limited. Particularly, the online nonlinear inversion presents a great challenge. Hirschorn studied the invertibility condition, inverse structure, and derivation for nonlinear dynamic system inverse [35], but his derivation of the nonlinear inverse involved higher-order derivatives and caused problems when noises and disturbances were present in the system. A few nonlinear IMC algorithms are proposed based on Hirschorn nonlinear inverse [13–15, 36, 37]. Some nonlinear IMCs assumes the higher-order derivatives to be known, which is unrealistic. Others solve the nonlinear inverse numerically using the contraction principle method or Newton's method. It is prone to noise and has high algorithmic and computational complexity.

The nonlinear model is often linearized to exploit the linear IMC design tools. Feedforward/feedback linearization approach was adopted by Calvet and Arkun to derive the model for the nonlinear plant in IMC [38]. Toivonen et al. derived the linear model based on the velocity-based linearization, then developed the IMC controller based on linear IMC theory [39]. It was only applicable when there were a small number of scheduling parameters. Another possible avenue to exploit the lin-

ear IMC design tools for nonlinear systems would be through the linear parameter varying (LPV) model [40]. It is, however, shown that such a treatment only works in limited cases, and when it works, the derivation is computationally expensive and the subsequent design in the LPV framework is very demanding. Mohammapour et al. applied IMC on a quasi-LPV model with two approaches [41]. In the first approach, the IMC controller parameters were scheduled based on the LPV model parameters which were assumed to be known in real time and not vary rapidly. In the second approach, the design problem was formulated in the H_∞ framework as a set of linear matrix inequalities (LMI). Solving the associated LMI problem, however, was computationally intensive.

In this thesis, we showed that for some specific nonlinear systems, the structure of the nonlinearity can be explored so that the inversion can be decomposed into cascaded inversions of several first-order systems [42, 43], thereby alleviating the difficulty in high-order nonlinear system inversion. However, this is only applicable if the nonlinear system has the special structural property that lends itself for such a decomposition.

In addition to nonlinearity, a challenge in IMC design is dealing with variations in the underlying plant and its operating environment. Many identification techniques have been exploited for adaptive IMC (AIMC). Applying the artificial neural networks on IMC has been well studied, where the feasibility of simultaneous identifying the model and its inverse with neural networks was demonstrated [44–49]. Application of IMC with neural networks have been documented in a few publications [50, 51], where the identifications of the neural networks were performed off-line due to its computational complexity. IMC has been explored with fuzzy logic [52–54], where the dynamic model and its inverse were represented by fuzzy logic, and found success in a few applications [55, 56]. A Bayesian Gaussian process was exploited to identify the model in IMC [57]. A few kernel-based identification techniques were combined

with IMC [58–60]. AIMC with an adaptive inverse control strategy was investigated in [61,62], in which the inverse is approximated by an adaptive finite impulse response (FIR) filter as presented in [63]. However, for neural network, fuzzy logic, Gaussian process, kernel-based model, and adaptive FIR filter, their black-box identifications make it difficult to incorporate physical knowledge about the plant in the controller design.

AIMC with the certainty equivalence principle and parameter estimation incorporated physical knowledge about the plant [64–67]. By certainty equivalence design principle, the model was identified based on known structure of the plant, and the inverse was derived by inverting the estimated model. Stability, robustness, and performance issues are addressed in [64]. AIMC with parameter estimation is intuitive and simple to implement. Many successful applications have been discussed [68–70].

For AIMC designed with the certainty equivalence approach, simultaneous identification of the model and the inverse is very tempting: IMC performs better with a more accurate model and a more accurate inverse. Intuitively by identifying the inverse directly, as opposed to calculating the inverse model from the identified plant model, one can achieve a more accurate inverse dynamic model. Moreover, the direct identification of the inverse avoids online inversion of the identified model, which might be non-invertible under unspecified excitation conditions. This inspired us to design composite AIMC (CAIMC) as presented in this thesis, where the model and the inverse are identified simultaneously [71–73].

1.3   Background on Boost-Pressure Control of Turbocharged Gasoline Engines

In this thesis, we choose the boost-pressure control problem of turbocharged gasoline engines to illustrate the design and calibration challenges as well as to demon-
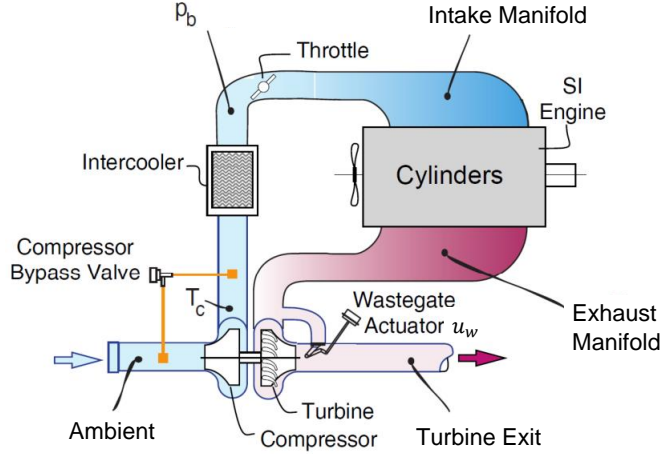
Figure 1.4: Turbocharged gasoline engine structure [75].

strate the proposed solutions. As shown in Fig. 1.3, the turbocharger extracts the energy from the exhaust gas to pressurize the ambient air, which increases the power density and fuel efficiency of the engine [74]. The air is compressed by the compressor, and passes through an intercooler and a throttle before entering the engine intake port. The engine exhaust port is connected to the turbine, which is mechanically connected to the compressor. An wastegate actuator controls the opening of the turbine bypass path in this application [17], affecting the compressor speed and therefore the boost-pressure. While providing adequate boost at low speed and load, the turbocharger system also has to avoid over-boost situation at high speed and load [76]. Traditionally, the boost-pressure is measured and PI control with feedforward on the wastegate is used to regulate the boost-pressure to desired set-point [74, 76, 77]. A few applications of IMC to boost-pressure control of turbocharged gasoline engines emerged in the last few years.

Thomasson et al. utilized IMC for PID tuning of wastegate control in turbocharged gasoline engines [16]. Karnik et al. [17] applied IMC directly to wastegate control for a turbocharged gasoline engine. They used a first-order gain-scheduled linear model which was simplified from a fourth-order nonlinear model using singular perturbation. While the simplicity of the first-order model-based design was an ad-

vantage for implementation, its performance was limited by the linear approximation, as it is defined for a particular operating point. Overall, none of these applications has offered a general, implementable, and easy-to-calibrate solution.

Motivated by the need for an easy-to-calibrate control framework, we first explored the application of nonlinear IMC to the boost-pressure control problem. A first principle fourth-order nonlinear model is derived, and its inverse is derived from the quasi-linear parameter varying version of the nonlinear model. It is presented in the context of the boost-pressure control, and it is hard to generalize due to its requirement for a special structural property of the nonlinear model. We then applied CAIMC to the boost-pressure control problem. The design of CAIMC assumes that the plant model and its inverse are represented by the first-order linear dynamics. The simultaneous identification of the model and the inverse reduces the tracking error through the online identifications. CAIMC is validated on the boost-pressure control problem in both simulations and experiments, which shows that CAIMC is not only effective, but also reduces the calibration effort significantly.

1.4 Contributions

This thesis aims at developing an analytical framework and associated tools to facilitate the design and implementation of easy-to-calibrate control solutions. We proposed a general control framework CAIMC, and developed new technologies for the boost-pressure control problem: CAIMC and nonlinear IMC. Therefore, the contributions are two-fold.

The first contribution is the development of new and improved control methodologies in the IMC framework.

- CAIMC, where the forward plant model and the plant inverse are identified simultaneously, is developed. It is an improvement upon AIMC, where the

model is obtained from system identification of a plant, and the inverse is the inversion of the identified model.

- For the first time, the IMC tracking error is represented as the sum of the forward plant modeling error and the right-inverse modeling error, based on which the advantage of CAIMC can be established analytically.

- A general approach to identify the parameter with the constraint imposed by the stability of an n-th order model is developed. The constrained parameter identification is reformulated as a convex programming problem.

- The stability proof and asymptotic performance are established for CAIMC. All the signals in the closed-loop system are uniformly bounded, and the tracking error converges to zero when there is no unmodeled dynamics.

The second contribution is the development of new control technologies for the boost-pressure control problem of turbocharged gasoline engines.

- CAIMC is applied to the boost-pressure control problem of a turbocharged gasoline engine. It is first applied to a physics-based high-order and nonlinear Ford proprietary turbocharged gasoline engine model, and then validated on a turbocharged $2L$ four-cylinder gasoline engine on a Ford Explorer EcoBoost with vacuum actuated wastegate. It is shown to be effective and robust and it drastically reduces the calibration effort.

- The feasibility, performance, advantages, and limitations of a nonlinear IMC for the boost-pressure control are explored. The challenges for inverting the nonlinear model are addressed by representing the nonlinear dynamics with a quasi-LPV model. Simulations on a high-fidelity turbocharged engine model are carried out to show the feasibility of the proposed nonlinear IMC. The nonlinear IMC is applicable only when the nonlinear system has a special structural

property, and it remains an open problem to generalize the nonlinear IMC and develop a guideline for calibration.

1.5   Outline

This thesis reports the comprehensive research efforts on IMC and boost-pressure control of turbocharged gasoline engines. Specifically, we first present the control methodology development of CAIMC, then we presented the technology development for boost-pressure control, including CAIMC and nonlinear IMC. This thesis is organized as follows:

Chapter I is an introduction. **Chapter II** presents what initiates and motivates this work: internal model control(IMC) and adaptive IMC (AIMC).

Chapter III presents the composite AIMC (CAIMC) for a first-order plant. The tracking error for CAIMC is expressed as the sum of the forward modeling error and inverse modeling error to justify the simultaneous identifications in CAIMC. Two different designs of CAIMC: CAIMC with left-inverse (CAIMC-LI) and CAIMC with right-inverse (CAIMC-RI) are presented for the first-order model and compared. The stability proof in the ideal case without unmodeled dynamics is presented.

Chapter IV generalizes the first-order CAIMC presented in Chapter III to n-th order. The CAIMC design procedure and stability proof are both presented. The challenge when extending CAIMC from first-order to n-th order lies in handling the constraint imposed by the stability of an n-th order model. A constrained parameter identification algorithm is reformulated as a convex programming problem to tackle the challenge.

Chapter V presents the detailed design procedures of applying the first-order CAIMC to the boost-pressure control of a turbocharged gasoline engine. The resulting CAIMC is first applied to a physics-based high order and nonlinear proprietary turbocharged gasoline engine model, and then validated on a turbocharged $2L$ four-

cylinder gasoline engine on a Ford Explorer EcoBoost.

Chapter VI present the application of a nonlinear IMC to the boost-pressure control problem. A fourth-order nonlinear model which sufficiently describes the dynamic behavior of the turbocharged engine is used as the design model in the IMC structure. A new approach for nonlinear inversion, referred to as the structured quasi-LPV model inverse, is developed and validated. Finally, simulations on a validated high-fidelity model are carried out to show the feasibility of the proposed IMC. Its closed-loop performances are compared with a well-tuned PI controller with extensive feedforward and anti-windup built in. Robustness of the nonlinear IMC design is analyzed using simulations.

Chapter VII draws the conclusions.

CHAPTER II

Adaptive Internal Model Control

2.1 Internal Model Control (IMC)

Internal model control (IMC) is a control structure as shown in Fig. 1.3, where G, M, and Q represent the plant, model, and inverse, respectively. It is called internal model control because the model M is an explicit component in the controller, which offers an alternative to the classical feedback control structure.

The difference between the responses of G and M, $y - y_M$ is fed back to Q. When the model is exact, i.e., $M = G$, the feedback signal is zero and IMC structure becomes open-loop. This open-loop feedforward structure obviously requires that G and Q are stable. We assume that the plant of interest is **stable** for the rest of the thesis. Q can be designed as a stable open-loop feedforward controller. Therefore, one approach to IMC design is to optimize the tracking error in the H_2 sense, that is

$$\underset{Q}{\text{minimize}} \ \|e\|_2 = \underset{Q}{\text{minimize}} \ \|r - y\|_2 = \underset{Q}{\text{minimize}} \ \|\{1 - GQ\}r\|_2, \qquad (2.1)$$

subject to the constraint that Q is stable and causal. The optimization (2.1) reaches the minimum 0 at $Q = G^{-1}$ without constraints. For a stable and minimum phase(MP) plant, Q can be designed as G^{-1} appended with a filter to make it causal. But for a stable and non-minimum phase (NMP) plant, $Q = G^{-1}$ is unstable. For a particular

reference r, an inverse of $G = M$ that minimize the tracking error in the H_2 sense can be found through the following lemma [8].

Lemma II.1. *Assume that M is stable. Factor M into an allpass portion M_A and a MP portion M_M, i.e.,*

$$M = M_A M_M, \tag{2.2}$$

so that M_A includes all the RHP zeros and delays of M and

$$|M_A(i\omega)| = 1, \forall \omega. \tag{2.3}$$

In general, M_A has the form

$$M_A(s) = e^{-s\theta} \prod_i \frac{-s + \zeta_i}{s + \zeta_i^H}, \quad Re(\zeta_i) > 0, \ \theta > 0. \tag{2.4}$$

where the superscript H denotes complex conjugate.

Factor R, the Laplace transformation of the reference signal r similarly

$$R = R_A R_M. \tag{2.5}$$

The controller Q which solves (2.1) is given by

$$Q = (M_M R_M)^{-1} \{M_A^{-1} R_M\}_*. \tag{2.6}$$

where the operator $\{\cdot\}_*$ denotes performing a partial fraction expansion of the term $(\cdot)$, and then omitting all components that involve the poles of M_A^{-1} [8].

With (2.6), an approximate inverse Q of the model M is derived. In general, Q is not proper, and it is to be augmented by a filter.

13

IMC has many appealing properties [8], which were shown to carry over to IMC with nonlinear models [36] and adaptive IMC [64], including:

- **_Zero offset_** The controller will lead to zero steady-state error as long as the steady-state gain of Q is the inverse of the steady-state gain of M.

- **_Simple tuning philosophy_** The only tuning parameter in the IMC structure is the time constant of the filter augmented to the inverse. It can be calibrated to achieve the desired bandwidth of the closed-loop system.

2.2 Adaptive IMC (AIMC)

In practical applications, the plant parameters are often unknown or vary with time. The plant uncertainty might be too large to be handled by a fixed controller, making adaptive control a desirable solution. Adaptive control structure is usually formulated by combining online parameter estimation with a control law that is motivated from the known parameter case [78]. The estimated parameters are treated as if they are the true parameters. It is referred to as the certainty equivalence principle, and is the key to adaptive control design.

Applying adaptive control to IMC structure with certainty equivalence principle leads to Adaptive IMC (AIMC). It is shown in Fig. 2.1, where the model M is derived using online plant parameter identification, and the inverse Q is derived from inverting M following Lemma II.1 [64]. The design details of AIMC including the online identification of the parameters and the controller design are presented.

We will use a linear model to approximate the plant. For a fixed nonlinear plant, the parameters of its linearization vary with different operating points. An n-th order linear dynamic model can be assumed to have the general form of [78]

$$y = \left\{ \frac{Z_M(s)}{R_M(s)} \right\} u = \frac{b_0^* s^n + b_1^* s^{n-1} + \cdots + b_n^*}{s^n + a_1^* s^{n-1} + \cdots + a_n^*} u, \tag{2.7}$$

14
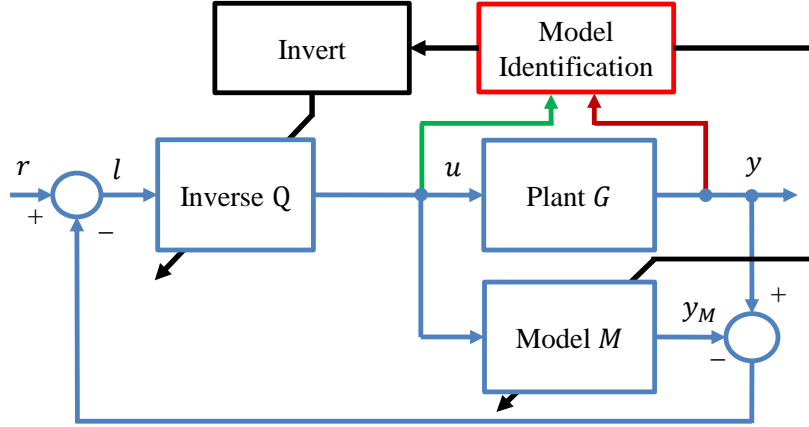
Figure 2.1: AIMC illustration.

where y and u are the output and input of the plant, $\theta^* = [b_0^*, \ldots, b_n^*, a_1^*, \ldots, a_n^*]$ are unknown. Throughout the thesis, $\{\cdot\}$ represents the dynamic operators, whose transfer function is $(\cdot)$. A general certainty equivalence AIMC scheme can be designed following the following steps:

1. Formulate a plant model structure based on the plant dynamics, i.e., define the order and the relative degree of the plant model, while taking the accuracy and complexity into consideration.

2. Derive the parametric model from (2.7) and estimate the unknown parameters using an appropriate adaptive law. (This step is presented in Section 2.2.1.)

3. Calculate the appropriate Q from the estimated plant model M, using Lemma II.1.

4. Using the certainty equivalence principle to implement Q. (This step is presented in detailed in Section 2.2.2.)

Note: the estimation of the plant parameters and the calculation of Q occur online. Therefore, AIMC is capable of capturing the plant parameter variation caused by aging and variation in the operating environment.

15

2.2.1 Plant Model and its Estimation

2.2.1.1 Linear Parametric Model

If one can obtain the linear expression

$$z = \theta^{*T}\phi, \tag{2.8}$$

where the parameter vector θ^* contains all the unknown parameters in the dynamic equation (2.7), z and ϕ are available to measure, then θ^* can be easily estimated with standard estimation approaches, such as the least-square and the gradient method. Such linear expression (2.8) is referred to as a linear parametric model. z is referred to as the observation, and ϕ is referred to as the regressor.

To derive the linear parametric model, (2.7) can be rewritten as

$$s^n y = b_0^* s^n u + b_1^* s^{n-1} u + \cdots + b_n^* u - a_1^* s^{n-1} y - \cdots - a_n^* y, \tag{2.9}$$

$$\left\{\frac{s^n}{\Lambda(s)}\right\} y = b_0^* \left\{\frac{s^n}{\Lambda(s)}\right\} u + \cdots + b_n^* \left\{\frac{1}{\Lambda(s)}\right\} u - a_1^* \left\{\frac{s^{n-1}}{\Lambda(s)}\right\} y - \cdots - a_n^* \left\{\frac{1}{\Lambda(s)}\right\} y,$$

where $\frac{1}{\Lambda(s)}$ is introduced to avoid derivatives in generating signals needed for parameter estimation and to filter out the noise. $\Lambda(s)$ is an n-th order Hurwitz polynomial [78]. The observation z and regressor ϕ in (2.10) are defined as

$$z = \left\{\frac{s^n}{\Lambda(s)}\right\} y, \tag{2.10}$$

$$\phi = \left[\left\{\frac{s^n}{\Lambda(s)}\right\} u, \left\{\frac{s^{n-1}}{\Lambda(s)}\right\} u, \dots, \left\{\frac{1}{\Lambda(s)}\right\} u, -\left\{\frac{s^{n-1}}{\Lambda(s)}\right\} y, -\left\{\frac{s^{n-2}}{\Lambda(s)}\right\} y, \dots, -\left\{\frac{1}{\Lambda(s)}\right\} y\right]^T.$$

2.2.1.2 Continuous-time Adaptive Law

The normalized gradient algorithm with projection is presented here to identify the unknown parameter vector θ^* [78]. The gradient algorithm identifies the parameter θ by minimizing certain performance cost w.r.t. θ. Projection algorithm can be used

with the gradient algorithm to handle the constraints on θ.

Continuous-time Normalized Gradient Algorithm We will first discuss the identification of the unknown parameters θ^* without constraints on θ. Here the gradient algorithm minimizes a quadratic cost function of the normalized estimation error

$$\epsilon = \frac{z - \theta^T \phi}{1 + \phi^T \phi} = \frac{z - \theta^T \phi}{m^2}, \tag{2.11}$$

where $m^2 = 1 + \phi^T \phi$ is the normalizing term. The quadratic cost function of ϵ is defined as

$$J(\theta) = \frac{\epsilon^2 m^2}{2} = \frac{(z - \theta^T \phi)^2}{2m^2}. \tag{2.12}$$

$J(\theta)$ is convex over the space of θ. Applying the gradient method, we have

$$\dot{\theta} = -\Gamma \nabla J(\theta) = \Gamma \epsilon \phi, \tag{2.13}$$

where $\Gamma = \Gamma^T$ is a scaling positive definite matrix that affects how rapid θ converges. The properties of the normalized gradient algorithm are summarized as:

Lemma II.2. *The adaptive law (2.13) applied to the linear parametric model (2.8) guarantees that [78]*

(i) $\epsilon, \epsilon m, \theta, \dot{\theta} \in \mathcal{L}_\infty$.

(ii) $\epsilon, \epsilon m, \dot{\theta} \in \mathcal{L}_2$ *independent of the boundedness of the signal vector ϕ, and*

(iii) *if $m, \phi \in \mathcal{L}_\infty$ and ϕ is persistently exciting (PE), then $\theta(t)$ converges exponentially to θ^*.*

17

When there is unmodeled dynamics, instead of (2.8),

$$z = \theta^{*T}\phi + \eta, \tag{2.14}$$

where η is the unmodeled dynamics term. From (2.11), $\epsilon = \frac{(\theta^* - \theta)^T \phi + \eta}{m^2}$ is used to "drive" the adaptive law to estimate θ in the case of the gradient algorithm. When ϵ is small, η may be more dominant than $(\theta^* - \theta)^T\phi$ in ϵ, and θ will drift unnecessarily. To make the adaption law robust w.r.t. the unmodeled dynamics η, a deadzone is often applied on $\epsilon(i)$ [78]. The deadzone with size g_0 can be implemented as a function

$$g(\epsilon) = \begin{cases} \epsilon - g_0, & \text{if } \epsilon > g_0, \\ \epsilon + g_0, & \text{if } \epsilon < -g_0, \\ 0, & \text{otherwise.} \end{cases}$$

With the deadzone, the adaptation occurs only when the estimation error ϵ is large relative to the modeling error η.

Continuous-time Normalized Gradient Algorithm with Projection The normalized gradient algorithm (2.13) minimizes the cost function $J(\theta)$ with no constraints, i.e., it allows θ to lie anywhere in $\mathcal{R}^l$, where l is the dimension of θ. When there is priori knowledge about θ^*, or when θ has to be bounded for stability or safety reasons, projection algorithm can be applied to the normalized gradient algorithm to handle the constraints. Suppose the constraint is $\theta \in \mathcal{S}$, where $\mathcal{S}$ is a convex set with a smooth boundary. The problem now is to minimize the cost function $J(\theta)$ subject to $\theta \in \mathcal{S}$. Let $\mathcal{S}$ be given by

$$\mathcal{S} = \{\theta \in \mathcal{R}^m | g(\theta) \leq 0\} \tag{2.15}$$

where $g : \mathcal{R}^m \to \mathcal{R}$ is a smooth function. The normalized gradient algorithm with

projection is given by

$$\dot{\theta} = Pr_{\mathcal{S}}(-\Gamma\nabla J)$$

$$:= \begin{cases} -\Gamma\nabla J, \text{ if } \theta \in \mathcal{S}^0, \text{ or if } \theta \in \delta(\mathcal{S}) \text{ and } -(\Gamma\nabla J)^T\nabla g \le 0 \\ -\Gamma\nabla J + \Gamma\frac{\nabla g\nabla g^T}{\nabla g^T\Gamma\nabla g}\Gamma\nabla J, \text{ otherwise.} \end{cases} \quad (2.16)$$

where $\mathcal{S}^0$ is the interior of $\mathcal{S}$, $\delta(\mathcal{S})$ is the boundary of $\mathcal{S}$, and the initial condition $\theta(0) \in \mathcal{S}$. The intuitive interpretation of the gradient algorithm with projection (2.16) is as follows: θ is usually updated using the gradient algorithm, but when θ is going to move to any $\tilde{\theta}$ outside $\mathcal{S}$, $\dot{\theta}$ is adjusted such that θ moves to the projection of $\tilde{\theta}$ on the boundary $\delta(\mathcal{S})$. Therefore, θ never leaves the subset $\mathcal{S}$, i.e., the constraints are always satisfied.

Lemma II.3. *The gradient adaptive law of (2.13) with the projection modification given by (2.16) retains all their properties that are established in the absence of projection and in addition guarantees that $\theta \in \mathcal{S}$, $\forall t \ge 0$ provided $\theta(0) \in \mathcal{S}$ and $\theta^* \in \mathcal{S}$ [78].*

2.2.1.3   Hybrid Adaptive Law

For computational and robustness reasons, the updates of the identified parameters can be performed at specific instants of time kT_s, where T_s is the sampling time. The updates of the identified $\theta(k)$ at the k-th sample will take the general form of

$$\theta(k+1) = \theta(k) + f(\phi(t), \epsilon(t)), t \in [kT_s, (k+1)T_s], \quad (2.17)$$

where f is the correction term, and the normalized estimation error

$$\epsilon(t) = \frac{z(t) - \theta^T(k)\phi(t)}{m(t)^2}, \forall kT_s \le t < (k+1)T_s, \quad (2.18)$$

where $m(t)^2 = 1 + \phi(t)^T \phi(t)$ is the normalizing term. (2.17) is referred to as hybrid adaptive law.

Hybrid Normalized Gradient Algorithm Hybrid version of the normalized gradient algorithm is presented here to obtain the estimated θ of the unknown parameter vector θ^* in (2.8) [78]. Again we consider a quadratic cost function of ϵ: $J(\theta) = \frac{\epsilon^2 m^2}{2}$. Applying the gradient method, we have

$$\theta(k+1) = \theta(k) + \Gamma \int_{kT_s}^{(k+1)T_s} \epsilon(t)\phi(t)dt, \tag{2.19}$$

where $\Gamma = \Gamma^T$ is a positive-definite matrix that affects how rapid θ converges.

Lemma II.4. *[78] Let $2 - T_s\lambda_{max}(\Gamma) \geq c$ for some $c > 0$. The adaptive law (2.19) for (2.8) guarantees that*

(i) $\theta \in l_\infty$.

(ii) $\Delta\theta \in l_2$, where $\Delta\theta(k) = \theta(k+1) - \theta(k)$.

(iii) $\epsilon, \epsilon m \in \mathcal{L}_2 \cap \mathcal{L}_\infty$.

(iv) If $m, \phi \in \mathcal{L}_\infty$ and ϕ is persistently exciting, then $\theta(k) \to \theta^*$ as $k \to \infty$ exponentially fast.

Hybrid Normalized Gradient with Projection Similarly with the continuous-time case, projection algorithm can be applied to the hybrid normalized gradient algorithm to handle the constraints given by $\theta \in \mathcal{S}$ in (2.15). The normalized gradient algorithm with projection is given by

$$\theta(k+1) = Pr_{\mathcal{S}}\left(\theta(k) + \Gamma \int_{kT_s}^{(k+1)T_s} \epsilon(t)\phi(t)dt\right), \tag{2.20}$$

and the initial condition $\theta(0) \in \mathcal{S}$. Projection of a point onto a convex set $\mathcal{S}$ always exists and is unique.

Lemma II.5. *The gradient adaptive law of (2.19) with the projection modification given by (2.20) retains all their properties that are established in the absence of projection and in addition guarantees that $\theta \in \mathcal{S}$, $\forall k \geq 0$ provided $\theta(0) \in \mathcal{S}$ and $\theta^* \in \mathcal{S}$ [78].*

2.2.2 Controller/Inverse Design

From Fig. 1.3, we see that the control input u in the IMC structure for a LTI model is given by

$$u = \{Q(s)\}l = \{Q(s)\}(r - (y - y_M)). \tag{2.21}$$

An alternate approach to implement $Q(s)$ that is well-suited for varying coefficients is introduced as follows. Q can be expressed as $Q = \frac{Q_n}{Q_d}$, where $Q_n(s)$ and $Q_d(s)$ are polynomials with $Q_d(s)$ being Hurwitz. Choose $\Lambda(s)$ to be an arbitrary monic Hurwitz polynomial of order n with leading coefficient 1. Then the control law (2.21) can also be implemented as

$$
\begin{aligned}
u &= \left\{ \frac{\Lambda(s) - Q_d(s)}{\Lambda(s)} \right\} u + \left\{ \frac{Q_n(s)}{\Lambda(s)} \right\} (r - (y - y_M)) \\
&= q_d^T \left\{ \frac{\xi_{n-1}(s)}{\Lambda(s)} \right\} u + q_n^T \left\{ \frac{\xi_n(s)}{\Lambda(s)} \right\} (r - (y - y_M))
\end{aligned} \tag{2.22}
$$

where $\xi_n(s) = [1, s, \ldots, s^n]^T$, q_d is the vector of the coefficients of $\Lambda(s) - Q_d(s)$, and q_n is the vector of the coefficients of $Q_n(s)$.

With the parametric model (2.10) and the adaptive law (2.16) or (2.20), the estimates of the numerator $\hat{Z}_M$ and denominator $\hat{R}_M$ of the plants are obtained. Using the estimated frozen-time plant model, we can derive $Q(s, t) = \frac{\hat{Q}_n(s,t)}{\hat{Q}_d(s,t)}$ following Lemma II.1.

With (2.22), the certainty equivalence principle leads to the control law

$$
\begin{aligned}
u &= \left\{ \frac{\Lambda(s) - \hat{Q}_d(s,t)}{\Lambda(s)} \right\} u + \left\{ \frac{\hat{Q}_n(s,t)}{\Lambda(s)} \right\} (r - (y - y_M)) \\
&= \hat{q}_d^T \left\{ \frac{\xi_{n-1}(s)}{\Lambda(s)} \right\} u + \hat{q}_n^T \left\{ \frac{\xi_n(s)}{\Lambda(s)} \right\} (r - (y - y_M))
\end{aligned}
\tag{2.23}
$$

where $\hat{q}_d$ is the vector of the coefficients of $\Lambda(s) - \hat{Q}_d(s,t)$, and $\hat{q}_n$ is the vector of the coefficients of $\hat{Q}_n(s,t)$.

CHAPTER III

First-order Composite Adaptive Internal Model Control

The IMC structure explicitly includes a model and an inverse of the plant. In Chapter II, we presented adaptive IMC, where the model is identified, and the inverse is derived by inverting the estimated model. However, the inverse of the estimated model does not necessarily represent a good inverse of the plant, especially when there is no persistent excitation. To form a more accurate inverse, identifying the inverse directly is very appealing, which yields the composite adaptive IMC (CAIMC) as shown in Fig. 3.1. It is based on a standard IMC structure with two simultaneous identifications, where the model identification estimates the unknown parameters of
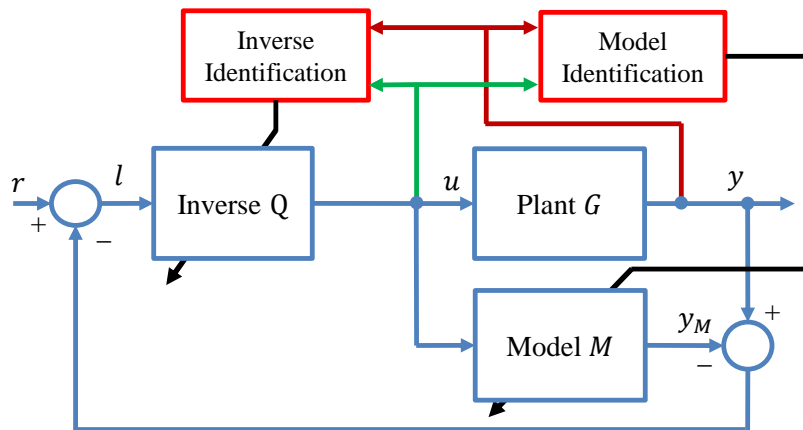
Figure 3.1: CAIMC illustration.

the model, and the inverse identification estimates the unknown parameters of the inverse. "Composite" refers to the simultaneous identifications of the plant model and the plant inverse. The design procedure of CAIMC, following the certainty equivalence principle, is described in steps as follows:

1. Formulate a forward plant model and an inverse model structure based on the plant dynamics, i.e., define the order and the relative degree of the forward plant model and the plant inverse model, while taking the accuracy and complexity into consideration.

2. Derive the parametric models of the proposed forward plant model and inverse model structures respectively, identify the unknown parameters using an appropriate adaptive law.

3. Treat the identified model and inverse as the true plant and plant inverse and embed them into M and Q in the IMC structure.

In this chapter, we will focus on the design details of CAIMC for a first-order plant. In Section 3.1, we first demonstrate the advantages of CAIMC analytically. The tracking error is represented as the sum of the forward modeling error and the inverse modeling error. If the identifications can minimize the modeling errors, then CAIMC can reduce the tracking error through the simultaneous identifications of the model and the inverse. In Section 3.2, CAIMC for a first-order plant, including the parametric models, the parameter identification schemes, and the controller realization, are explained in detail. In Section 3.3, the stability proof of the first-order CAIMC structure without unmodeled dynamics and noises is presented.

3.1 Tracking Error Representation of CAIMC

Intuitively simultaneous identifications of the model and the inverse will yield better IMC performance. Here we will discuss the advantage of simultaneous identifi-

cations analytically. The tracking error of IMC is presented as $e = \{1-GQ\}r$ in (2.1). It is based on the assumption that there is no modeling error, which is unrealistic in practical applications. When $M \neq G$, the tracking error of IMC can be represented as

$$e = e_Q + e_M, \tag{3.1}$$

where

$$e_M = y - y_M, \tag{3.2}$$

$$e_Q = l - y. \tag{3.3}$$

y_M is the model response, and $l = r - y + y_M$ is the input to the inverse Q as shown in Fig. 1.3. (3.1) can be derived by noting that $e = r - y = (r - y + y_M - y) + (y - y_M) = (l - y) + (y - y_M)$.

Note that e_M is the difference between the plant and the model responses, which is referred to as the **forward modeling error**, and e_Q is the difference between the input to Q and the plant response, which is referred to as the **inverse modeling error**.

With (3.1), the tracking error e has been represented as the sum of the forward modeling error e_M and the inverse modeling error e_Q. To the best of our knowledge, this expression has not been presented in the literature. With the triangle inequality,

$$|e| \leq |e_M| + |e_Q|, \tag{3.4}$$

The tracking error can be bounded from above by the sum of $|e_M|$ and $|e_Q|$. This expression inspires and justifies the separate estimations of M and Q. Intuitively the forward modeling error e_M should be related to the forward model estimation error
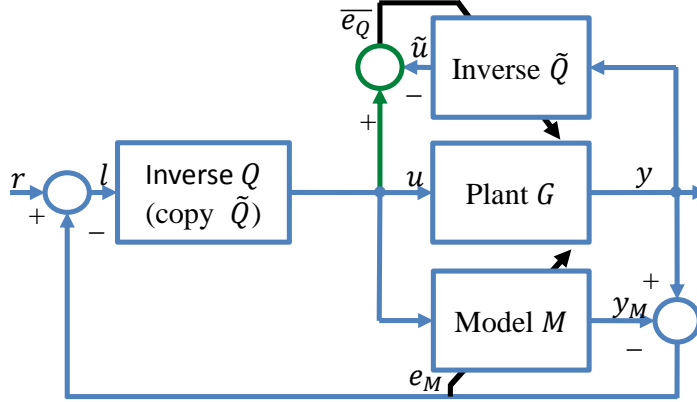
Figure 3.2: CAIMC-LI illustration.

ϵ_M, and the inverse modeling error e_Q should be related to the inverse estimation error ϵ_Q. Recall that the gradient method is based on minimizing the quadratic cost functions of the estimation error ϵ. Therefore, the minimization of the quadratic cost functions of ϵ_M and ϵ_Q will contributes to reducing the tracking error.

Specifically, e_Q is the right-inverse modeling error. For an operator G that maps the input space U to the output space Y, it is often possible to find operators Q_L and Q_R with the property

$$\{Q_L G\}u = u, u \in U, \tag{3.5}$$

$$\{G Q_R\}y = y, y \in Y. \tag{3.6}$$

Q_L and Q_R are called left and right-inverse operators of G, respectively [36]. For a SISO LTI operator G, $Q_R = Q_L$, whereas they are generally different for nonlinear or time-varying operators. $l = \{G Q_R\}l$ matches the form in (3.6). Therefore $e_Q = l - y = \{G Q_R\}l - \{G Q\}l$ characterizes how well Q represents Q_R, the right-inverse of G.

Therefore, in step 2 of CAIMC design procedure, the identification of the inverse model can be carried out in two different ways by assuming different inverses, leading to two different CAIMC schemes: CAIMC with left-inverse (CAIMC-LI) and
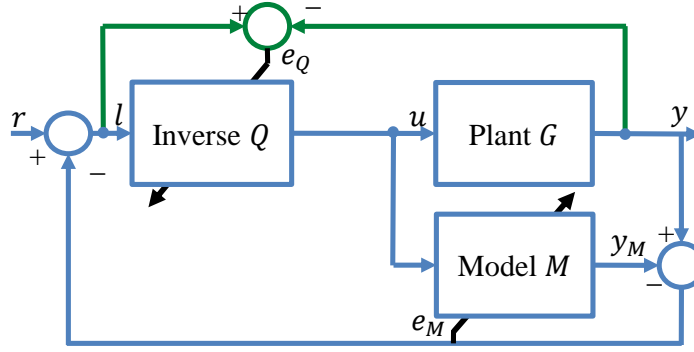
Figure 3.3: CAIMC-RI illustration.

CAIMC with right-inverse (CAIMC-RI). They have the same structure as shown in Fig. 3.1, the same overall design procedure following step 1 - 3, and the same model identification for M. The differences are the identifications of the inverse Q.

CAIMC-LI and CAIMC-RI are illustrated in Fig. 3.2 and 3.3, where the forward modeling error

$$e_M = y - y_M \tag{3.7}$$

drives the model identification for both CAIMC-LI and CAIMC-RI. For CAIMC-LI in Fig. 3.2, the left-inverse modeling error signal

$$\overline{e_Q} = u - \tilde{u} = \{Q_L G\}u - \{\tilde{Q}G\}u \tag{3.8}$$

which characterizes how well $\tilde{Q}$ represent Q_L drives the left-inverse identification, and the identified $\tilde{Q}$ is copied to Q. The left-inverse identification is simple and intuitive. Its design is very similar to the forward model identification. However, it does not directly minimize the right-inverse modeling error e_Q.

For CAIMC-RI in Fig. 3.3, the right-inverse modeling error signal

$$e_Q = l - y = \{GQ_R\}l - \{GQ\}l \tag{3.9}$$

27

which characterizes how well Q represent Q_R drives the right-inverse identification. The right-inverse identification directly minimizes the right-inverse modeling error e_Q, which is of interest.

In the next section, the details of CAIMC-LI and CAIMC-RI are discussed for a first-order linear plant. Their forward model and inverse parametric models, parameter identifications, and the controller realization are discussed.

3.2   CAIMC Design for a First-order Plant

For simplicity and clarity, CAIMC for a first-order linear plant is first introduced. Later in Chapter IV, we will generalize it to n-th order systems. Here the plant is assumed to have the first-order linear model

$$y = \left\{ \frac{k_1^*}{\tau^* s + 1} \right\} u \qquad (3.10)$$

and inverse

$$u = \left\{ \frac{1}{k_2^*} \frac{b^* s + 1}{a^* s + 1} \right\} y, \qquad (3.11)$$

where y and u are the output and input of the plant, τ^*, k_1^*, k_2^*, a^*, and b^* are unknown. Note that k_1^* and k_2^* should be equivalent, but they are treated as independent parameters for the model and inverse identification. Following step 2 of CAIMC design procedure, the parametric models will be discussed for the model and the inverse, and the normalized gradient algorithm with projection presented in Chapter II will be exploited to identify the unknown parameters. Then following step 3, the identified parameters are treated as the real ones to implement M and Q in the IMC structure.

3.2.1 CAIMC-LI for a First-order Plant

CAIMC-LI is demonstrated in Fig. 3.2, where the model identification is driven by minimizing the forward modeling error $e_M = y - y_M$ as in (3.7), and the inverse identification is driven by minimizing the left-inverse modeling error $\overline{e_Q} = u - \tilde{u}$ as in (3.8).

Forward model parametric model and identification

The first-order linear model is assumed to have the structure (3.10). The model identification minimizes a quadratic cost function of $\epsilon_M = \frac{z_M - \theta_M^T \phi_M}{m_M^2}$ as in (2.11). The goal of the model identification design is to design z_M and ϕ_M, such that the forward modeling error $e_M = y - y_M$ drives the identification, i.e., $\epsilon_M m_M^2 = z_M - \theta_M^T \phi_M = e_M$. With simple manipulation of (3.10), we have

$$\tau^* \left\{ \frac{s}{\tau_M s + 1} \right\} y + \left\{ \frac{1}{\tau_M s + 1} \right\} y = k_1^* \left\{ \frac{1}{\tau_M s + 1} \right\} u,$$

$$y = (\tau_M - \tau^*) \left\{ \frac{s}{\tau_M s + 1} \right\} y + k_1^* \left\{ \frac{1}{\tau_M s + 1} \right\} u,$$

where $\frac{1}{\tau_M s + 1}$ is the regressor filter. Therefore, for the plant model identification, the associated signals of the parametric model $z_M = \theta_M^{*T} \phi_M$ can be defined as

$$z_M = y, \theta_M^* = [\tau_M - \tau^*, k_1^*]^T,$$

$$\phi_M = [\left\{ \frac{s}{\Lambda_M(s)} \right\} y, \left\{ \frac{1}{\Lambda_M(s)} \right\} u]^T, \text{ where } \Lambda_M(s) = \tau_M s + 1, \qquad (3.12)$$

$$\epsilon_M = \frac{z_M - \theta_M^T \phi_M}{m_M^2}, m_M^2 = 1 + \phi_M^T \phi_M.$$

Then, the continuous-time normalized gradient algorithm (2.16) that minimizes $J(\theta_M)$ $= \frac{\epsilon_M^2 m_M^2}{2}$ can be adopted to estimate $\theta_M = [\tau_M - \tau, k_1]^T$. τ has to be positive to ensure the stability of the implemented model. Therefore the identification has to satisfy the constraint $\tau_M - \tau < \tau_M$. The detail of the normalized gradient algorithm with

projection according to (2.16) is:

$$(\tau_M \overset{\cdot}{-} \tau) = \begin{cases} \gamma_1 \epsilon_M \phi_{M1}, \text{if } \tau_M - \tau < c_1, c_1 < \tau_M, \text{or if } \tau_M - \tau = c_1, \epsilon_M \phi_{M1} < 0, \\ \\ 0, \text{otherwise}. \end{cases}$$

$$\dot{k}_1 = \gamma_2 \epsilon_M \phi_{M2},$$

$$\epsilon_M = \frac{z_M - \theta_M^T \phi_M}{m_M^2}, m_M^2 = 1 + \phi_M^T \phi_M. \tag{3.13}$$

Model realization and forward modeling error e_M

If M is designed such that

$$y_M = \theta_M^T \phi_M = (\tau_M - \tau) \left\{ \frac{s}{\Lambda_M(s)} \right\} y + k_1 \left\{ \frac{1}{\Lambda_M(s)} \right\} u, \tag{3.14}$$

where τ and k_1 are the identified parameters, we have

$$\epsilon_M m_M^2 = z_M - \theta_M^T \phi_M = y - y_M = e_M. \tag{3.15}$$

The forward modeling error e_M equals to the forward model estimation error $\epsilon_M m_M^2$, which drives the identification of M.

Left-inverse parametric model and identification

By simply swapping the roles of the input and output of plant model, we have the first-order inverse model as (3.11). This inverse is referred to as the left-inverse, in the sense that if $\tilde{Q} = Q_L$ in (3.8), $\overline{e_Q} = 0$. Similarly to the model identification, the goal of the left-inverse identification design is to design z_{QL} and ϕ_{QL}, such that the left-inverse modeling error $\overline{e_Q}$ drives the identification, i.e., $\epsilon_{QL} m_{QL}^2 = z_{QL} - \theta_Q^T \phi_{QL} = \overline{e_Q}$. To formulate all the signals used in the inverse identification, we write the inverse

model (3.11) as

$$a^* \left\{ \frac{s}{\tau_Q s + 1} \right\} u + \left\{ \frac{1}{\tau_Q s + 1} \right\} u = \left(\frac{b}{k_2}\right)^* \left\{ \frac{s}{\tau_Q s + 1} \right\} y + \left(\frac{1}{k_2}\right)^* \left\{ \frac{1}{\tau_Q s + 1} \right\} y,$$

where $\frac{1}{\tau_Q s + 1}$ is the regressor filter. It can be further expressed as

$$u = \left(\frac{b}{k_2}\right)^* \left\{ \frac{s}{\tau_Q s + 1} \right\} y + \left(\frac{1}{k_2}\right)^* \left\{ \frac{1}{\tau_Q s + 1} \right\} y + (\tau_Q - a^*) \left\{ \frac{s}{\tau_Q s + 1} \right\} u.$$

For the left-inverse identification, the associated signals of the parametric model $z_{QL} = \theta_Q^{*T} \phi_{QL}$ are defined as

$$z_{QL} = u, \theta_Q^* = [(\frac{b}{k_2})^*, (\frac{1}{k_2})^*, \tau_Q - a^*]^T,$$

$$\phi_{QL} = [\left\{ \frac{s}{\Lambda_Q(s)} \right\} y, \left\{ \frac{1}{\Lambda_Q(s)} \right\} y, \left\{ \frac{s}{\Lambda_Q(s)} \right\} u]^T, \text{ where } \Lambda_Q(s) = \tau_Q s + 1, \quad (3.16)$$

$$\epsilon_{QL} = \frac{z_{QL} - \theta_Q^T \phi_{QL}}{m_{QL}^2}, m_{QL}^2 = 1 + \phi_{QL}^T \phi_{QL}.$$

With the continuous-time normalized gradient algorithm (2.16), the unknown parameters $(\frac{b}{k_2})^*$, $(\frac{1}{k_2})^*$, and $\tau_Q - a^*$ can be identified by minimizing $J(\theta_Q) = \frac{\epsilon_{QL}^2 m_{QL}^2}{2}$. $\theta_Q = [\frac{b}{k_2}, \frac{1}{k_2}, \theta_{Q3}]^T$ represent the estimated parameters, where $\theta_{Q3} = \tau_Q - a$. Not only does a has to be positive to assure the stability of the plant, a has to be bigger than a certain constant to limit the bandwidth of the overall control system. Therefore, the identification has to satisfy the constraint $\theta_{Q3} < c_2$. The detail of applying the normalized gradient algorithm with projection according to (2.16) is

$$\left(\frac{\dot{b}}{k_2}\right) = \gamma_1 \epsilon_{QL} \phi_{QL1},$$

$$\left(\frac{\dot{1}}{k_2}\right) = \gamma_2 \epsilon_{QL} \phi_{QL2},$$

(3.17)

$$\dot{\theta}_{Q3} = \begin{cases} \gamma_3 \epsilon_{QL} \phi_{QL3}, & \text{if } \theta_{Q3} < c_2, \text{or if } \theta_{Q3} = c_2, \epsilon_{QL} \phi_{QL3} < 0, \\ 0, \text{otherwise.} \end{cases}$$

$$\epsilon_{QL} = \frac{z_{QL} - \theta_Q^T \phi_{QL}}{m_{QL}^2}, m_{QL}^2 = 1 + \phi_{QL}^T \phi_{QL}.$$

Left-inverse realization and inverse modeling error e_Q

Similarly to the model realization (3.14), the left-inverse $\tilde{Q}$ can be designed as

$$\tilde{u} = \theta_Q^T \phi_{QL} = \frac{b}{k_2} \left\{ \frac{s}{\tau_Q s + 1} \right\} y + \frac{1}{k_2} \left\{ \frac{1}{\tau_Q s + 1} \right\} y + (\tau_Q - a) \left\{ \frac{s}{\tau_Q s + 1} \right\} u. \quad (3.18)$$

The left-inverse estimation error

$$\epsilon_{QL} m_{QL}^2 = z_{QL} - \theta_Q^T \phi_{QL} = u - \tilde{u} = \overline{e_Q}. \quad (3.19)$$

Note that $\tilde{Q}$ is copied to Q in Fig. 3.2, and the control is actually implemented differently. For a fixed inverse $Q = \frac{1}{k_2^*} \frac{b^* s + 1}{a^* s + 1}$ with input $l = r - y + y_M$, $u = Ql = \{\frac{1}{k_2^*} \frac{b^* s + 1}{a^* s + 1}\}(r - y + y_M)$. A filter $\frac{1}{\tau_Q s + 1}$ is added for implementation.

$$a^* \left\{ \frac{s}{\tau_Q s + 1} \right\} u + \left\{ \frac{1}{\tau_Q s + 1} \right\} u = \frac{b^*}{k_2^*} \left\{ \frac{s}{\tau_Q s + 1} \right\} l + \frac{1}{k_2^*} \left\{ \frac{1}{\tau_Q s + 1} \right\} l,$$

which can be further expressed as

$$\frac{a^*}{\tau_Q}u = \left\{\frac{\frac{a^*}{\tau_Q}-1}{\tau_Q s + 1}\right\}u + \frac{b^*}{k_2^*}\left\{\frac{s}{\tau_Q s + 1}\right\}l + \frac{1}{k_2^*}\left\{\frac{1}{\tau_Q s + 1}\right\}l,$$

$$u = \left\{\frac{1-\frac{\tau_Q}{a^*}}{\tau_Q s + 1}\right\}u + \frac{\tau_Q b^*}{a^* k_2^*}\left\{\frac{s}{\tau_Q s + 1}\right\}l + \frac{\tau_Q}{a^* k_2^*}\left\{\frac{1}{\tau_Q s + 1}\right\}l.$$

Using certainty equivalence principle, the identified parameters a, b, and k_2 are treated as if they are the true parameters,

$$u = \left\{\frac{1-\frac{\tau_Q}{a}}{\tau_Q s + 1}\right\}u + \frac{\tau_Q b}{a k_2}\left\{\frac{s}{\tau_Q s + 1}\right\}l + \frac{\tau_Q}{a k_2}\left\{\frac{1}{\tau_Q s + 1}\right\}l. \tag{3.20}$$

Then the left-inverse estimation error can be expressed as

$$\begin{aligned}
\epsilon_{QL} m_{QL}^2 &= z_{QL} - \theta_Q^T \phi_{QL} = u - \tilde{u} \\
&= \left\{\frac{b}{k_2}s + \frac{1}{k_2}\right\}\left\{\frac{1}{\Lambda_Q(s)}\right\}(l-y) = \left\{\frac{b}{k_2}s + \frac{1}{k_2}\right\}\left\{\frac{1}{\Lambda_Q(s)}\right\}e_Q,
\end{aligned}$$

or equivalently

$$e_Q = (\left\{\frac{b}{k_2}s + \frac{1}{k_2}\right\}\left\{\frac{1}{\Lambda_Q(s)}\right\})^{-1}\epsilon_{QL}m_{QL}^2 = \{X\}\epsilon_{QL}m_{QL}^2, \tag{3.21}$$

where $\{X\}$ is a operator whose transfer function is $(\{\frac{b}{k_2}s + \frac{1}{k_2}\}\{\frac{1}{\Lambda_Q(s)}\})^{-1}$. The left-inverse identification derivation is simple and straightforward, however, the left-inverse identification minimizes the quadratic cost function of $\epsilon_{QL}m_{QL}^2 = \overline{e_Q}$, which is not exactly e_Q.

3.2.2 CAIMC-RI for a First-order Plant

CAIMC-RI is demonstrated in Fig. 3.3, where the model identification is driven by the forward modeling error $e_M = y - y_M$ as in (3.7), and the inverse identification is driven by $e_Q = l - y$ as in (3.9). The model identification is exactly the same as

CAIMC-LI. The inverse identification poses the biggest challenge.

Right-inverse parametric model and identification

The goal of the right-inverse identification design is to minimize e_Q, i.e., the right-inverse estimation error $\epsilon_{QR} m_{QR}^2 = e_Q$ as in (3.9). It is essentially designing an optimal feedforward controller that can minimize the quadratic cost function of the tracking error resulting from the feedforward control structure. For the first-order linear plant, the inverse is still assumed to be $u = \{\frac{1}{k_2^*} \frac{b^* s + 1}{a^* s + 1}\} l$ in (3.11).

The right-inverse identification scheme can be designed based on the left-inverse identification. To use e_Q as the inverse estimation error, the associated signals of the parametric model $z_{QR} = \theta_Q^{*T} \phi_{QR}$ has to be rewritten as

$$
z_{QR} = \{X_{i-1}\} \left\{ \frac{1}{\Lambda_Q(s)} \right\} u, \theta_Q^* = [(\frac{b}{k_2})^*, (\frac{1}{k_2})^*, \tau_Q - a^*]^T,
$$

$$
\phi_{QR} = \{X_{i-1}\}[\{ \frac{s}{\Lambda_Q(s)} \} y, \{ \frac{1}{\Lambda_Q(s)} \} y, \{ \frac{s}{\Lambda_Q(s)} \} u]^T, \Lambda_Q(s) = \tau_Q s + 1, \quad (3.22)
$$

$$
\epsilon_{QR} = \frac{z_{QR} - \theta_Q^T \phi_{QR}}{m_{QR}^2}, m_{QR}^2 = 1 + \phi_{QR}^T \phi_{QR}.
$$

where $\{X_{i-1}\} = (\{\frac{b}{k_2}(i-1)s + \frac{1}{k_2}(i-1)\}\{\frac{1}{\Lambda_Q(s)}\})^{-1}$ is added to the observation z_{QL} and regressor vector ϕ_{QL} given in (3.16). X_{i-1} is the transfer function given in (3.21) with $\frac{b}{k_2}(i-1)$ and $\frac{1}{k_2}(i-1)$ be the identified $\frac{b}{k_2}$ and $\frac{1}{k_2}$ from the previous sample time $(i-1)$. Note that the continuous-time adaptive law is adopted in identifying the parameters, so θ is a continuous-time signal. $\theta(i-1)$ is defined as $\theta(t_i - T_s)$, where T_s is the sampling time. $\{X_{i-1}\}$ is adopted because the identified parameters for the current time are not available for calculation. Similarly to the left-inverse identification, the identification has to satisfy the constraint $\theta_{Q3} < c_2$. The detail of applying the normalized gradient algorithm with projection according to (2.16) is

$$\left(\frac{\dot{b}}{k_2}\right) = \gamma_1 \epsilon_{QR} \phi_{QR1},$$

$$\left(\frac{\dot{1}}{k_2}\right) = \gamma_2 \epsilon_{QR} \phi_{QR2}, \tag{3.23}$$

$$\dot{\theta}_{Q3} = \begin{cases} \gamma_3 \epsilon_{QR} \phi_{QR3}, & \text{if } \theta_{Q3} < c_2, \text{or if } \theta_{Q3} = c_2, \epsilon_{QR} \phi_{QR3} < 0, \\ 0, & \text{otherwise} \end{cases}$$

$$\epsilon_{QR} = \frac{z_{QR} - \theta_Q^T \phi_{QR}}{m_{QR}^2}, m_{QR}^2 = 1 + \phi_{QR}^T \phi_{QR}.$$

Right-inverse realization and modeling error e_Q

The right-inverse is still designed as in (3.20). With this estimation reformulation, the right-inverse estimation error can be expressed as

$$\begin{aligned} \epsilon_{QR} m_{QR}^2 &= z_{QR} - \theta_Q^T \phi_{QR} \\ &= \{X_{i-1}\} z_{QL} - \theta_Q^T \{X_{i-1}\} \phi_{QL} = \{X_{i-1}\}(z_{QL} - \theta_Q^T \phi_{QL}) + \epsilon_1 \\ &= \{X_{i-1}\}\{X_i\}^{-1} e_Q + \epsilon_1 = e_Q + \epsilon_2, \end{aligned} \tag{3.24}$$

where ϵ_1 and ϵ_2 are residues from swapping the dynamic operators, and they are bounded by $\dot{\theta}_Q$ [78]. When θ_Q vary slowly, for the right-inverse, $\epsilon_{QR} m_{QR}^2 \approx e_Q$.

3.2.3 Comparison of CAIMC-LI and CAIMC-RI

CAIMC-LI is simple and intuitive to derive. Its tracking error based on (3.1), (3.15), and (3.21) is

$$e = e_M + e_Q = \epsilon_M m_M^2 + \{X\} \epsilon_{QL} m_{QL}^2, \tag{3.25}$$

where e_Q is represented as a function of $\epsilon_{QL} m_{QL}^2$. However, the time-varying operator X is hard to analyze. The reduction of tracking error is not directly related to the

minimization of the inverse identification error $\epsilon_{QL}m_{QL}^2$.

For CAIMC-RI, assuming the identified parameters vary slowly, based on (3.1), (3.15), and (3.24), the tracking error is

$$e = e_M + e_Q \approx \epsilon_M m_M^2 + \epsilon_{QR} m_{QR}^2. \tag{3.26}$$

With the tracking error representation (3.26), CAIMC-RI eases the performance analysis of CAIMC, and the direct minimization of e_M and e_Q will further improve the CAIMC performance.

3.3 Stability Proof of First-order CAIMC in the Ideal Case

In this section, the stability and asymptotic performance of the first-order CAIMC for the ideal case when there is no unmodeled dynamics is established.

Remark III.1. The inverse (3.11) is an approximate inverse of (3.10), because the plant is strictly proper. For the simplicity of the presentation, the stability proof and asymptotic performance are established for the ideal case without unmodeled dynamics. The proof of a general CAIMC and effects of the unmodeled dynamics are discussed in Chapter IV.

Theorem III.1. *Consider the plant (3.10), (3.11) subject to the CAIMC-LI or CAIMC-RI schemes without unmodeled dynamics. For any bounded reference r, all the signals in the closed-loop system are uniformly bounded. The tracking error $e = r - y$ converges to zero as $t \to \infty$.*

Proof: τ_M and τ_Q are design constants for the identifications. For simplicity of the proof, let $\tau_M = \tau_Q = \tau_0$. Note that the same analysis can be carried out with arbitrary choice of $\tau_M > 0$ and $\tau_Q > 0$ at the expense of some additional algebra. We

define that

$$\dot{x}_1 = -\frac{1}{\tau_0}x_1 + \frac{1}{\tau_0}y, \, x_1 = \left\{\frac{1}{\tau_0 s + 1}\right\}y$$

$$\dot{x}_2 = -\frac{1}{\tau_0}x_2 + \frac{1}{\tau_0}u, \, x_2 = \left\{\frac{1}{\tau_0 s + 1}\right\}u \qquad (3.27)$$

$$\dot{x}_3 = -\frac{1}{\tau_0}x_3 + \frac{1}{\tau_0}\epsilon_M m_M^2, \, x_3 = \left\{\frac{1}{\tau_0 s + 1}\right\}(\epsilon_M m_M^2),$$

where $\epsilon_M m_M^2 = y - y_M$ as shown in (3.15).

Step 1

Correlate u and y to the estimation error:

For the model estimation, from (3.14), (3.15), and (3.27), we have

$$\epsilon_M m_M^2 = \tau_0 \dot{x}_1 + x_1 - k_1 x_2 - (\tau_M - \tau)\dot{x}_1,$$

$$\dot{x}_1 = \frac{1}{\tau}(-x_1 + k_1 x_2 + \epsilon_M m_M^2). \qquad (3.28)$$

For the control Law (3.20) and (3.27), let $\bar{r} = \frac{\tau_Q b}{ak_2}\{\frac{s}{\tau_Q s + 1}\}r + \frac{\tau_Q}{ak_2}\{\frac{1}{\tau_Q s + 1}\}r$,

$$u = \tau_0 \dot{x}_2 + x_2 = (1 - \frac{\tau_0}{a})x_2 + \bar{r} - \frac{b}{ak_2}\epsilon_M m_M^2 - \frac{\tau_0 - b}{ak_2}x_3.$$

$$\dot{x}_2 = -\frac{1}{a}x_2 - \frac{\tau_0 - b}{ak_2 \tau_0}x_3 + \frac{1}{\tau_0}\bar{r} - \frac{b}{ak_2 \tau_0}\epsilon_M m_M^2. \qquad (3.29)$$

From (3.27),

$$\dot{x}_3 = -\frac{1}{\tau_0}x_3 + \frac{1}{\tau_0}\epsilon_M m_M^2. \qquad (3.30)$$

Let $x = [x_1, x_2, x_3]^T$, from (3.28), (3.29), and (3.30), we have

$$\dot{x} = \begin{bmatrix} -\frac{1}{\tau} & \frac{k_1}{\tau} & 0 \\ 0 & -\frac{1}{a} & -\frac{\tau_0 - b}{ak_2\tau_0} \\ 0 & 0 & -\frac{1}{\tau_0} \end{bmatrix} x + \begin{bmatrix} \frac{1}{\tau} \\ -\frac{b}{ak_2\tau_0} \\ \frac{1}{\tau_0} \end{bmatrix} \epsilon_M m_M^2 + \begin{bmatrix} 0 \\ \frac{1}{\tau_0} \\ 0 \end{bmatrix} \bar{r} \qquad (3.31)$$

$$\begin{bmatrix} u \\ y \\ \epsilon_M m_M^2 \end{bmatrix} = \tau_0 \dot{x} + x$$

Note: The left-inverse and the right-inverse are implemented with the same structure. Therefore, (3.31) applies to both CAIMC-LI and CAIMC-RI.

Step 2

Establish the exponential stability of the homogeneous part of (3.31):

All the elements of $A(t) = \begin{bmatrix} -\frac{1}{\tau} & \frac{k_1}{\tau} & 0 \\ 0 & -\frac{1}{a} & \frac{\tau_0 - b}{ak_2\tau_0} \\ 0 & 0 & -\frac{1}{\tau_0} \end{bmatrix}$ are obviously differentiable. For each

fixed time t, $A(t)$ has eigen-values at $-\frac{1}{\tau}$, $-\frac{1}{a}$, and $-\frac{1}{\tau_0}$, which are all negative $\forall t \geq 0$ because of projection (3.13), (3.17), and (3.23).

By Lemma II.2 and II.3, the adaptive law guarantees that $\theta_M, \theta_Q \in \mathcal{L}_\infty$, $\epsilon_M, \epsilon_M m_M$, $\dot{\theta}_M, \epsilon_Q, \epsilon_Q m_Q, \dot{\theta}_Q \in \mathcal{L}_\infty \cap \mathcal{L}_2$. $-\frac{1}{\tau}, -\frac{1}{a} \in \mathcal{L}_\infty$ because of projection. Since $\dot{\theta}_M, \dot{\theta}_Q \in \mathcal{L}_\infty \cap \mathcal{L}_2$, $\|\dot{A}(t)\| \in \mathcal{L}_2$. Lemma B.1 implies that the state transition matrix $\Phi(t, \tau)$ associated with $A(t)$ satisfies $\|\Phi(t, \tau)\| \leq \kappa_1 e^{-\kappa_2(t-\tau)}$, $\forall t \geq \tau \geq 0$ for some constant $\kappa_1, \kappa_2 > 0$.

Step 3

Establish signal boundedness: The $\mathcal{L}_{2\delta}$ norm $\|(\bullet)_t\|_{2\delta}$ for some $\delta > 0$ is the exponentially weighted $\mathcal{L}_2$ norm defined as $\|x_t\|_{2\delta} := (\int_0^t e^{-\delta(t-\tau)} x^T(\tau) x(\tau) d\tau)^{\frac{1}{2}}$. Ap-

plying Lemma B.2 to the state space equation (3.31), we can obtain

$$\|x_t\|_{2\delta} \leq c\|(\hat{\epsilon}_M m_M^2)_t\|_{2\delta} + c, \tag{3.32}$$

$$|x(t)| \leq c\|(\hat{\epsilon}_M m_M^2)_t\|_{2\delta} + c,$$

where $|\bullet|$ is a vector norm, for any $\delta \in [0, \delta_1)$ where $\delta_1 > 0$ is any constant less than $2\kappa_2$ and some finite constant $c \geq 0$. For simplicity of the representation, in this thesis, c is used to represent a generic constant.

We define the fictitious normalizing signal $m_f^2 := 1 + \|u_t\|_{2\delta}^2 + \|y_t\|_{2\delta}^2$. From the state space equation, we have $\|u_t\|_{2\delta} + \|y_t\|_{2\delta} \leq c\|x_t\|_{2\delta} + c\|(\hat{\epsilon}_M m_M^2)_t\|_{2\delta} + c$. With equation (3.32), we have $\|u_t\|_{2\delta} + \|y_t\|_{2\delta} \leq c\|(\hat{\epsilon}_M m_M^2)_t\|_{2\delta} + c$, implying

$$m_f^2 \leq c\|(\hat{\epsilon}_M m_M^2)_t\|_{2\delta}^2 + c.$$

From (3.12), applying Lemma B.3,

$$m_M = \sqrt{1 + \phi_M^T \phi_M} \leq cm_f, \tag{3.33}$$

$$m_f^2 \leq c\|(\tilde{g}m_f)_t\|_{2\delta}^2 + c,$$

where $\tilde{g} = \hat{\epsilon}_M m_M \in \mathcal{L}_{2e}$. Or

$$m_f^2 \leq c \int_0^t e^{-\delta(t-\tau)} \tilde{g}^2(\tau) m_f^2(\tau) d\tau + c,$$

where $0 < \delta \leq \delta^*$ and $\delta^* = min[2\lambda, \delta_1]$, $\delta_1 \in (0, 2\kappa_2)$.

Applying Lemma B.4 the B-G lemma, we can establish that $m_f \in \mathcal{L}_\infty$. Then with (3.33), we have $m_M \in \mathcal{L}_\infty$ and therefore $\phi_M, x, \dot{x}, u, y \in \mathcal{L}_\infty$, $\epsilon_Q m_Q^2 \in \mathcal{L}_\infty$, $m_Q = \sqrt{1 + \phi_Q^T \phi_Q} = \sqrt{1 + x_1^2 + x_2^2} \in \mathcal{L}_\infty$.

Step 4:

Establish that the tracking error e converges to zero:

Since the tracking error e can be expressed as the sum of $e_M = y - y_M$ and $e_Q = l - y$. We can demonstrate the convergence of e by demonstrating the convergence of e_M and e_Q respectively.

First, we consider the forward model estimation error equation

$$\epsilon_M m_M^2 = y - y_M. \tag{3.34}$$

$\epsilon_M m_M \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ based on the adaptive law. $m_M \in \mathcal{L}_\infty$ from Step 3. Therefore, $\epsilon_M m_M^2 \in \mathcal{L}_2 \cap \mathcal{L}_\infty$. Operating on each side of $\epsilon_M m_M^2 = y - y_M = y - \theta_M^T \phi_M$ with $s := \frac{d}{dt}$, we obtain

$$\frac{d}{dt}(y - y_M) = \dot{y} - \dot{\theta}_M^T \phi_M - \theta_M^T \dot{\phi}_M.$$

With (3.12),

$$
\begin{aligned}
\frac{d}{dt}(y - y_M) =& \dot{y} + \dot{\tau} \left\{ \frac{s}{\Lambda_M(s)} \right\} y - \dot{k}_1 \left\{ \frac{1}{\Lambda_M(s)} \right\} u - (\tau_0 - \tau)(\frac{1}{\tau_0}\dot{y} - \left\{ \frac{1}{\tau_0}\frac{s}{\Lambda_M(s)} \right\} y) \\
& - k_1 \left\{ \frac{s}{\Lambda_M(s)} \right\} u \\
=& \frac{\tau}{\tau_0} \dot{y} + \dot{\tau} \left\{ \frac{s}{\Lambda_M(s)} \right\} y - \dot{k}_1 \left\{ \frac{1}{\Lambda_M(s)} \right\} u + (\tau_0 - \tau)\frac{1}{\tau_0} \left\{ \frac{s}{\Lambda_M(s)} \right\} y \\
& - k_1 \left\{ \frac{s}{\Lambda_M(s)} \right\} u.
\end{aligned}
$$

With (3.27),

$$\frac{d}{dt}(y - y_M) = \frac{\tau}{\tau_0}\dot{y} + \dot{\tau}\dot{x}_1 - \dot{k}_1 x_2 + (\tau_0 - \tau)\frac{1}{\tau_0}\dot{x}_1 - k_1\dot{x}_2.$$

$\dot{y} \in \mathcal{L}_\infty$, $x, \dot{x}, \theta_M, \dot{\theta}_M \in \mathcal{L}_\infty$. Therefore,

$$\frac{d}{dt}(\epsilon_M m_M^2) \in \mathcal{L}_\infty.$$

Since $\epsilon_M m_M^2 \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\frac{d}{dt}(\epsilon_M m_M^2) \in \mathcal{L}_\infty$, it follows that $\epsilon_M m_M^2 \to 0$ as $t \to \infty$. Similarly, we can show that $\epsilon_{QL} m_{QL}^2 \to 0$ and $\epsilon_{QR} m_{QR}^2 \to 0$ as $t \to \infty$.

For CAIMC-LI, $e = e_M + e_Q = \epsilon_M m_M^2 + \{X\}\epsilon_{QL} m_{QL}^2$, where $\{X\}$ is a stable transfer function. Therefore, $e \to 0$ as $t \to \infty$. For CAIMC-RI, $e_Q = \epsilon_Q m_Q^2 - \epsilon_2$, and $\epsilon_2 \to 0$ since $\dot{\theta}_Q \to 0$. Therefore, $e_Q \to 0$. $e = e_M + e_Q$.

Therefore, $e \to 0$ as $t \to \infty$.

CHAPTER IV

Generalized n-th Order CAIMC

This chapter generalizes the first-order CAIMC as presented in Chapter III to n-th order. With the generalization, CAIMC can be applied to a SISO n-th order plant. We consider the n-th order stable plant

$$y = \left\{ \frac{Z_M(s)}{R_M(s)} \right\} u, \tag{4.1}$$

whose stable and proper inverse is represented by

$$u = \left\{ \frac{Z_Q(s)}{R_Q(s)} \right\} y, \tag{4.2}$$

where y and u are the output and input of the plant respectively. $R_M(s)$, $R_Q(s)$, and $Z_Q(s)$ are n-th order Hurwitz polynomials. Z_M is a polynomial with order no greater than n. The leading coefficients of $R_M(s)$ and $R_Q(s)$ are 1. The other coefficients are unknown.

The design procedure of CAIMC presented in Chapter III applies to n-th order plant. The tracking error $e = e_M + e_Q$ as presented in Section 3.1 also holds for n-th order CAIMC. One fundamental assumption of the IMC design is that both the plant and the inverse models are stable. The stability of a first-order and second-order model yields linear constraints in the parameter space, whereas for an n-th order
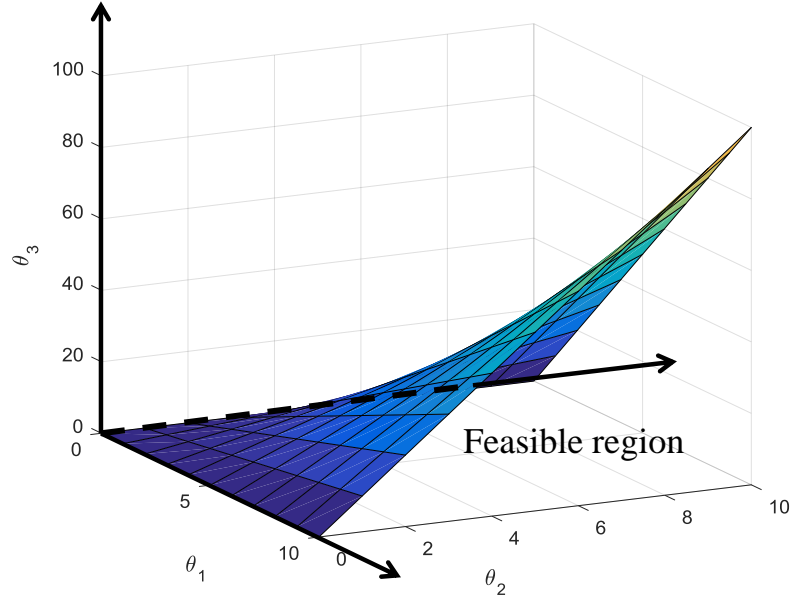
Figure 4.1: Feasible region constrained by the stability of a third-order model.

model ($n > 2$), the stability constraint yields nonlinear and nonconvex constraints with non-smooth boundaries in the original parameter space. Shown in Fig. 4.1 is the feasible region constrained by the stability of a third-order transfer function $\frac{1}{s^3+\theta_1 s^2+\theta_2 s+\theta_3}$. The constraints include $\theta_1 > 0$, $\theta_2 > 0$, $\theta_3 > 0$, and $\theta_1 \theta_2 > \theta_3$. The feasible region is the area indicated in Fig. 4.1 in the first quadrant under the curved surface, which is obviously nonconvex with non-smooth boundaries.

The main challenge in generalizing CAIMC is to develop an effective constrained parameter identification algorithm with the following properties:

- It can handle the stability constraint, which yields nonlinear and nonconvex constraints with non-smooth boundaries.

- It can produce a reasonable approximation for the unconstrained parameter identification.

- It is computationally efficient as the identification is executed online.

- It can assure the closed-loop stability of the control system.

In this chapter, the detail of n-th order CAIMC is discussed in Section 4.1, including the parametric models and controller realization. In Section 4.2, the constrained parameter identification problem is formulated as a convex programming problem, with established properties for the identified parameters. In Section 4.3, the closed-loop stability and asymptotic performance are established for the ideal case when there are no unmodeled dynamics, and the effect of unmodeled dynamics is discussed. In Section 4.4, CAIMC is applied to a third-order LTI plant.

4.1   CAIMC Design for an n-th Order Plant

Following is the development detail of CAIMC-LI and CAIMC-RI applied to the plant (4.1, 4.2). Following step 2 of CAIMC design procedure, the parametric models will be discussed for the model and the inverse, and the adaptive law will be discussed in Section 4.2 to identify the unknown parameters. Then following step 3, the identified parameters are treated as the real ones to realize M and Q in the IMC structure.

4.1.1   CAIMC-LI for an n-th Order Plant

CAIMC-LI is demonstrated in Fig. 3.2, where the model identification is driven by the forward modeling error $e_M = y - y_M$ as in (3.7), and the inverse identification is driven by the left-inverse modeling error $\overline{e_Q} = u - \tilde{u}$ as in (3.8).

Forward model parameterization and identification

The goal of the model identification is to define z_M and ϕ_M, such that a parametric model $z_M = \theta_M^{*T} \phi_M$ can be used to identify θ_M^* which includes the unknown parameters of (4.1). Since the parametric model in the form of (2.8) for a given physical process is not unique, we attempt to find the particular one with the property of $\epsilon_M m_m^2 = e_M$,

in light of the discussion in Section 3.1, so that e_M drives the model identification as shown in Fig. 3.2.

With simple manipulation of (4.1), and introducing a regressor filter, we have

$$y = \left\{ \frac{\Lambda_M - R_M(s)}{\Lambda_M} \right\} y + \left\{ \frac{Z_M(s)}{\Lambda_M} \right\} u, \tag{4.3}$$

where Λ_M is an n-th order Hurwitz Polynomial. We then define

$$z_M = y, \theta_M^* = [\theta_a^{*T}, \theta_b^{*T}]^T, \tag{4.4}$$

$$\phi_M = [\left\{ \frac{1}{\Lambda_M} \right\} y, ..., \left\{ \frac{s^{n-1}}{\Lambda_M} \right\} y, \left\{ \frac{1}{\Lambda_M} \right\} u, ..., \left\{ \frac{s^{n-1}}{\Lambda_M} \right\} u]^T,$$

where θ_a^* is the coefficients vector of $\Lambda_M - R_M(s)$, and θ_b^* is the coefficients vector of $Z_M(s)$. (4.3) can be expressed as $z_M = \theta_M^{*T} \phi_M$ and the estimation error ϵ_M can be defined as

$$\epsilon_M = \frac{z_M - \theta_M^T \phi_M}{m_M^2}, m_M^2 = 1 + \phi_M^T \phi_M, \tag{4.5}$$

where θ_M is the estimation of θ_M^*, ϵ_M is the normalized estimation error, and m_M^2 is the normalizing term. Without constraints, the hybrid normalized gradient algorithm (2.18) that minimizes $J(\theta_M) = \frac{\epsilon_M^2 m_M^2}{2}$ can be adopted to identify θ_M.

Forward model realization and modeling error

Once the estimates θ_M are obtained, M can be designed as

$$y_M = \theta_M^T \phi_M, \tag{4.6}$$

then

$$e_M = y - y_M = z_M - \theta_M^T \phi_M = \epsilon_M m_M^2. \tag{4.7}$$

Consequently, the estimation algorithm that minimizes the estimation error ϵ_M also minimizes e_M.

Left-inverse parametrization and identification

The goal of the left-inverse identification is to define z_{QL} and ϕ_{QL}, such that a parametric model $z_{QL} = \theta_Q^{*T} \phi_{QL}$ can be used to identify θ_Q^* which includes the unknown parameters of (4.2). For left-inverse, we can follow a similar parametric model design with the model. From (4.2), we have

$$u = \left\{ \frac{\Lambda_Q - R_Q(s)}{\Lambda_Q} \right\} u + \left\{ \frac{Z_Q(s)}{\Lambda_Q} \right\} y,$$

where Λ_Q is an n-th order Hurwitz polynomial that serves as a filter. For the left-inverse identification, the associated signals of the parametric model $z_{QL} = \theta_Q^{*T} \phi_{QL}$ are defined as

$$z_{QL} = u, \theta_Q^* = [\theta_c^{*T}, \theta_d^{*T}]^T,$$
$$\phi_{QL} = [\left\{ \frac{1}{\Lambda_Q} \right\} y, \left\{ \frac{s}{\Lambda_Q} \right\} y, ..., \left\{ \frac{s^{n-1}}{\Lambda_Q} \right\} y, \left\{ \frac{s^n}{\Lambda_Q} \right\} y, \left\{ \frac{1}{\Lambda_Q} \right\} u, \left\{ \frac{s}{\Lambda_Q} \right\} u, ..., \left\{ \frac{s^{n-1}}{\Lambda_Q} \right\} u]^T,$$
$$\epsilon_{QL} = \frac{z_{QL} - \theta_Q^T \phi_{QL}}{m_{QL}^2}, m_{QL}^2 = 1 + \phi_{QL}^T \phi_{QL}, \tag{4.8}$$

where θ_c^* is the vector of coefficients of $Z_Q(s)$ whose size is $n + 1$, θ_d^* is the vector of coefficients of $\Lambda_Q - R_Q(s)$ whose size is n. Without constraints, the hybrid normalized gradient algorithm (2.18) that minimizes $J(\theta_Q) = \frac{\epsilon_{QL}^2 m_{QL}^2}{2}$ can be adopted to identify θ_Q.

Left-inverse realization and modeling error

Similarly to the model realization (4.6), the left-inverse $\tilde{Q}$ can be designed as

$$\tilde{u} = \theta_Q^T \phi_{QL} = \left\{ \hat{Z}_Q(s) \frac{1}{\Lambda_Q} \right\} y + \left\{ (\Lambda_Q - \hat{R}_Q(s)) \frac{1}{\Lambda_Q} \right\} u. \tag{4.9}$$

The left-inverse estimation error

$$\epsilon_{QL} m_{QL}^2 = z_{QL} - \theta_Q^T \phi_{QL} = u - \tilde{u} = \overline{e_Q}. \tag{4.10}$$

Note that $\tilde{Q}$ is copied to Q in Fig. 3.2, and the control is actually realized differently. For a fixed inverse $Q = \frac{Z_Q(s)}{R_Q(s)}$ with input $l = r - y + y_M$, $u = Ql = \left\{ \frac{Z_Q(s)}{R_Q(s)} \right\} (r - y + y_M)$.

$$\left\{ \frac{R_Q(s)}{\Lambda_Q} \right\} u = \left\{ \frac{Z_Q(s)}{\Lambda_Q} \right\} l = \left\{ \frac{Z_Q(s)}{\Lambda_Q} \right\} (r - y + y_M),$$

which can be further expressed as

$$u = \left\{ \frac{\Lambda_Q - R_Q(s)}{\Lambda_Q} \right\} u + \left\{ \frac{Z_Q(s)}{\Lambda_Q} \right\} l$$

$$= \left\{ \frac{\Lambda_Q - R_Q(s)}{\Lambda_Q} \right\} u + \left\{ \frac{Z_Q(s)}{\Lambda_Q} \right\} (r - y + y_M).$$

Using certainty equivalence principle,

$$u = \left\{ (\Lambda_Q - \hat{R}_Q(s)) \frac{1}{\Lambda_Q} \right\} u + \left\{ \hat{Z}_Q(s) \frac{1}{\Lambda_Q} \right\} (r - y + y_M) \tag{4.11}$$

$$= \left\{ (\Lambda_Q - \hat{R}_Q(s)) \frac{1}{\Lambda_Q} \right\} u + \left\{ \hat{Z}_Q(s) \frac{1}{\Lambda_Q} \right\} r - \left\{ \hat{Z}_Q(s) \frac{1}{\Lambda_Q} \right\} y + \left\{ \hat{Z}_Q(s) \frac{1}{\Lambda_Q} \right\} y_M$$

$$= \theta_d^T \left\{ \frac{\xi_{n-1}(s)}{\Lambda_Q} \right\} u + \theta_c^T \left\{ \frac{\xi_n(s)}{\Lambda_Q} \right\} r - \theta_c^T \left\{ \frac{\xi_n(s)}{\Lambda_Q} \right\} y + \theta_c^T \left\{ \frac{\xi_n(s)}{\Lambda_Q} \right\} y_M,$$

where $\xi_n(s) = [1, s, \dots, s^n]^T$, $[\theta_c^T, \theta_d^T]^T = \theta_Q$.

From (4.9) and (4.11), the left-inverse estimation error can be expressed as

$$\epsilon_{QL}m_{QL}^2 = u - \tilde{u} = \left\{ \hat{Z}_Q(s)\frac{1}{\Lambda_Q} \right\} (l - y) = \left\{ \hat{Z}_Q(s)\frac{1}{\Lambda_Q} \right\} e_Q,$$

or equivalently

$$e_Q = (\left\{ \hat{Z}_Q(s)\frac{1}{\Lambda_Q} \right\})^{-1}\epsilon_{QL}m_{QL}^2 = \{X\}\epsilon_{QL}m_{QL}^2, \tag{4.12}$$

where $\{X\}$ is a operator whose transfer function is $(\left\{ \hat{Z}_Q(s)\frac{1}{\Lambda_Q} \right\})^{-1}$. The left-inverse identification derivation is simple and straightforward, however, the left-inverse identification minimizes the quadratic cost function of $\epsilon_{QL}m_{QL}^2 = \overline{e_Q}$, which is not exactly e_Q.

4.1.2   CAIMC-RI for an n-th Order Plant

CAIMC-RI is demonstrated in Fig. 3.3, where the model identification is driven by the forward modeling error $e_M = y - y_M$ as in (3.7), and the inverse identification is driven by $e_Q = l - y$ as in (3.9). The forward model identification is exactly the same as CAIMC-LI. The inverse identification poses the biggest challenge.

Right-inverse parameterization and identification

The key challenge in the inverse identification is to define a parametric model that directly relates to the inverse modeling error so that $\epsilon_{QR}m_{QR}^2 = e_Q$, where $e_Q = l - y$ as defined in (3.3). Then e_Q drives the inverse identification as shown in Fig. 3.3. Note that from (4.12)

$$\begin{aligned} e_Q &= \{X\}(z_{QL} - \theta_Q^T\phi_{QL}) = \{X\}z_{QL} - \{X\}\theta_Q^T\phi_{QL} \\ &= \{X\}z_{QL} - \theta_Q^T\{X\}\phi_{QL} + \epsilon_1, \end{aligned} \tag{4.13}$$

48

where the last equation is obtained by applying the Swapping Lemma [84], and ϵ_1 is the residual term resulting from swapping. If we redefine the inverse parametric model such that the new observation is $\{X\}z_{QL}$ and the new regressor is $\{X\}\phi_{QL}$, then the new estimation error is $e_Q - \epsilon_1$. Since X depends on the identified parameters whose value at current sample time are not available, z_{QR} and ϕ_{QR} are defined as

$$
\begin{aligned}
z_{QR} &= \{X_{i-1}\}u, \theta_Q^* = [\theta_c^{*T}, \theta_d^{*T}]^T, \\
\phi_{QR} &= \{X_{i-1}\}[\left\{\frac{1}{\Lambda_Q}\right\}y, ..., \left\{\frac{s^n}{\Lambda_Q}\right\}y, \left\{\frac{1}{\Lambda_Q}\right\}u, ..., \left\{\frac{s^{n-1}}{\Lambda_Q}\right\}u]^T,
\end{aligned}
\tag{4.14}
$$

where X_{i-1} is a transfer function with $\hat{Z}_Q(s)$ having the parameters identified at the previous sample time $(i-1)$ using the hybrid adaptive law (2.18). Now we can define

$$
\epsilon_{QR} = \frac{z_{QR} - \theta_Q^T \phi_{QR}}{m_{QR}^2}, m_{QR}^2 = 1 + \phi_{QR}^T \phi_{QR},
\tag{4.15}
$$

ϵ_{QR} is the normalized estimation error, and m_{QR}^2 is the normalizing term. Without constraints, the normalized gradient algorithm that minimizes $J(\theta_Q) = \frac{\epsilon_{QR}^2 m_{QR}^2}{2}$ can be adopted to identify θ_Q.

Right-inverse realization and modeling error

Same as (4.11), the inverse model Q is designed as

$$
\begin{aligned}
u &= \left\{(\Lambda_Q - \hat{R}_Q(s))\frac{1}{\Lambda_Q}\right\}u + \left\{\hat{Z}_Q(s)\frac{1}{\Lambda_Q}\right\}l \\
&= \theta_d^T \left\{\frac{\xi_{n-1}(s)}{\Lambda_Q}\right\}u + \theta_c^T \left\{\frac{\xi_n(s)}{\Lambda_Q}\right\}r - \theta_c^T \left\{\frac{\xi_n(s)}{\Lambda_Q}\right\}y + \theta_c^T \left\{\frac{\xi_n(s)}{\Lambda_Q}\right\}y_M.
\end{aligned}
\tag{4.16}
$$

To establish the connection between e_Q and ϵ_Q, from (4.13) note that the inverse estimation error can be expressed as

$$
\begin{aligned}
\epsilon_{QR}m_{QR}^2 &= z_{QR} - \theta_Q^T \phi_{QR} = \{X_{i-1}\}z_{QL} - \theta_Q^T\{X_{i-1}\}\phi_{QL} \\
&= \{X_{i-1}\}(z_{QL} - \theta_Q^T\phi_{QL}) + \epsilon_2 \\
&= \{X_{i-1}\}\{X\}^{-1}e_Q + \epsilon_2 = e_Q + \epsilon_3.
\end{aligned}
\tag{4.17}
$$

where ϵ_2 and ϵ_3 are residues from swapping the dynamic operators, they are bounded by $\theta_Q(k) - \theta_Q(k-1)$ [84]. For the inverse, we therefore have $\epsilon_Q m_Q^2 \approx e_Q$.

4.2 Parameter Identification with Stability and Bandwidth Constraints

The design procedure discussed in Section 4.1 does not consider constraints on the identified parameters θ_M or θ_Q. Constraints have to be imposed on θ_M and θ_Q to assure stability and to limit the bandwidth of the closed-loop system, because even if θ_M^* and θ_Q^* satisfy the constraints, θ_M and θ_Q may not due to the transients or the lack of excitations. Besides, the projection method adopted in Chapter III is not applicable to an n-th order model with nonconvex constraints with non-smooth boundaries. In this section, the stability constraints of M and Q are enforced as a linear matrix inequality (LMI). The constrained parameter identification is formulated as a convex optimization problem, which is solved at each sampling time to assure the stability of the estimated M and Q.

4.2.1 Stability and Bandwidth Constraints

Similarly to the first-order case, one of the sufficient conditions for establishi the CAIMC stability is the frozen-time stability of M and Q, namely their denominators $\hat{R}_M(s)$ and $\hat{R}_Q(s)$ have to be Hurwitz for all time. To have a stable implementation

of $\{X\}$ in (4.14), $\hat{Z}_Q$ has to be Hurwitz as well. Moreover, to limit the bandwidth of the closed-loop IMC, the real part of the solutions of $\hat{R}_Q(s) = 0$ have to satisfy $-\lambda_b < Re(s) < 0$, where λ_b is a constant. Therefore, an algorithm to constrain the locations of the roots of an n-th order polynomial is required. For simplicity, we will focus on constraining the solutions on the left-half plane (LHP), as the other cases can be dealt with using a simple linear transformation.

4.2.1.1 Routh-Hurwitz Criterion

Routh-Hurwitz criterion can be used to describe the stability constraint. As the order of the polynomial increases $(n > 2)$, the constraint becomes nonconvex with non-smooth boundaries. Besides, the Routh-Hurwitz criterion introduces a set of complicated high-order polynomial constraint that is hard to generalize for an n-th order plant. To represent the constraint in a more general and compact form, we seek Lyapunov inequality to represent the constraint.

4.2.1.2 Lyapunov Inequality

Given a transfer function whose denominator is $s^n + \sum_1^n \hat{\theta}_i s^{n-i}$ and the corresponding controllable canonical form of its state-space realization has its A matrix in the form of

$$A = \begin{bmatrix} -\hat{\theta} \\ I_{n-1} \quad 0 \end{bmatrix},$$

where $\hat{\theta} = [\hat{\theta}_1, ..., \hat{\theta}_n] \in \mathcal{R}^n$. The stability condition can be expressed as the non-emptiness of the set defined by $\mathcal{P} = \{P | P \succ 0, AP + PA^T \prec 0\}$.

Note that AP has the special expression as

$$AP = \begin{bmatrix} -\hat{\theta} \\ I_{n-1} \quad 0 \end{bmatrix} P = \begin{bmatrix} -\hat{\theta}P \\ [I_{n-1} \quad 0]P \end{bmatrix}, \tag{4.18}$$

which will be exploited next to handle the technical difficulties in the constrained parameter identification.

The Lyapunov inequality has the structure of an LMI. Therefore some background on LMI and its application to system stability is included.

Linear Matrix Inequality (LMI) with Applications to System Stability
An LMI has the form

$$F(\chi) := F_0 + \sum_{i=1}^{m} \chi_i F_i \succ 0, \tag{4.19}$$

where $\chi = [\chi_1, \chi_2, ..., \chi_m]^T \in \mathcal{R}^m$ is the variable, $F_i \in \mathcal{S}^k, i = 0, ..., m$ are given, and $\mathcal{S}^k$ is the set of $k \times k$ symmetric matrices. $F(\chi) \succ 0$ means that $F(\chi)$ is positive-definite. LMIs are convex constraints, i.e., the subsets in the space of χ constrained by LMIs are convex [79].

LMIs have a wide application in system and control theory [79], because many control problems can be reformulated as LMIs [80]. For a continuous-time linear time-invariant (LTI) system $\dot{x} = Ax$, the necessary and sufficient condition for its stability is that $\exists P \succ 0$, such that $A^T P + PA \prec 0$ [81]. The Lyapunov inequality is an LMI if either A or P is known. In the literature, LMI have been applied to the stability analysis of a linear system with a known A with a specified class of uncertainty [82] and an unknown A that lies in a polytope with known vertices [83].

Parameter identification with an LMI constraint can be casted as a constrained optimization problem that minimizes a convex cost function of a variable $\chi \in \mathcal{R}^m$

subject to an LMI [85]:

$$\underset{\chi}{\text{minimize}} \ f(\chi), \ \text{subject to} \ F(\chi) \succeq 0, \tag{4.20}$$

where $f : \mathcal{R}^n \to \mathcal{R}$, the LMI $F(\chi)$ is defined in (4.19), except that $F(\chi)$ is positive semi-definite, i.e., the constraint is a non-strict LMI. If $f(\chi)$ is convex, the optimization problem (4.20) can be solved effectively with interior point methods for which many commercial or open source tools are available [85].

4.2.2 Convex Optimization Problem Formulation

Let $\theta \in \mathcal{R}^n$ be an estimate of the parameter vector of an n-th order transfer function identified using standard techniques, such as the gradient algorithm. In this section, we will formulate an optimization problem to find a stable $\hat{\theta} \in \mathcal{R}^n$ that best approximates the unconstrained parameter θ. Since the optimization problem is solved at kT_s, where T_s is the sampling time. Hybrid normalized gradient method (2.19) is adopted to calculate θ.

Standard projection algorithm projects θ onto the feasible region by minimizing the distance between θ and $\hat{\theta}$ in the feasible region. Similarly, a natural formulation of the optimization problem is to minimize the quadratic error between θ and $\hat{\theta}$, subject to that $\hat{\theta}$ satisfies the stability constraint:

$$\underset{\hat{\theta},P}{\text{minimize}} \ ||\theta - \hat{\theta}||_2^2, \tag{4.21}$$

$$\text{subject to} \ P \succ 0 \ \text{and} \ \begin{bmatrix} -\hat{\theta}P \\ [I_{n-1} \quad 0]P \end{bmatrix} + \begin{bmatrix} -\hat{\theta}P \\ [I_{n-1} \quad 0]P \end{bmatrix}^T \prec 0.$$

However, $\hat{\theta}P$ in (4.21) introduces a nonconvex bilinear matrix inequality (BMI). The bilinear optimization problem (4.21) can be solved with global approaches such

as branch and bound, but it is computationally expensive [85]. To reduce the computational complexity, the bilinear optimization problem is reformulated as a convex optimization problem. We define a new variable

$$H = \hat{\theta} P \in \mathcal{R}^n. \tag{4.22}$$

A weighting matrix P is added to the quadratic cost function $||\theta - \hat{\theta}||_2^2$, and (4.21) becomes

$$\underset{P,H}{\text{minimize}} \ ||\theta P - H||_2^2, \tag{4.23}$$

$$\text{subject to } P \succ 0 \text{ and } \begin{bmatrix} -H \\ [I_{n-1} \ \ 0]P \end{bmatrix} + \begin{bmatrix} -H \\ [I_{n-1} \ \ 0]P \end{bmatrix}^T \prec 0.$$

Note that the reformulated cost function and redefined parameters, (4.23) has an LMI constraint.While (4.23) is not equivalent to (4.21), the new optimization formulation eliminates the BMI and replaces it with an LMI, thereby leading to a simpler problem amendable to many effective solvers.

For uniqueness of the optimal solution P^* and H^*, a regularizing term $\gamma||[vec(P), H]||_2^2$ is added to the cost function in (4.23) to make the cost strictly convex, where $\gamma > 0$ is a small constant, $vec(P)$ is a vector with all the unknown parameters in P. The constraint in (4.23) are tightened by $P \succeq \epsilon_0 I$ and $PA^T + AP \preceq -\epsilon_0 I$ where $\epsilon_0 > 0$ is a small constant. These modifications transform (4.23) to:

$$\underset{P,H}{\text{minimize}} \ ||\theta P - H||_2^2 + \gamma||[vec(P), H]||_2^2, \tag{4.24}$$

$$\text{subject to } P \succeq \epsilon_0 I \text{ and } \begin{bmatrix} -H \\ [I_{n-1} \ \ 0]P \end{bmatrix} + \begin{bmatrix} -H \\ [I_{n-1} \ \ 0]P \end{bmatrix}^T \preceq -\epsilon_0 I.$$

Solving the optimization problem returns the optimal P^* and H^*, and $\hat{\theta}$ can be

calculated as $\hat{\theta} = H^*P^{*-1}$ according to (4.22). (4.24) can be solved efficiently with interior point methods, for which there are many mature tools available [79].

4.2.3 Constrained Parameter Identification Implementation

Given the parametric model (2.8) with the normalized estimation error ϵ defined in (2.11), and the convex optimization problem proposed in Section 4.2.2, the implementation of the constrained parameter identification is summarized as following:

At the k-th sample:

1. Use unconstrained hybrid adaptive law (2.19) to calculate $\theta(k)$.

2. Solve the convex optimization problem in (4.24) for the optimal solution P^* and H^*, with $\theta = \theta(k)$.

3. Set $\hat{\theta}(k) = H^*P^{*-1}$, and use $\hat{\theta}(k)$ for control signal computation.

4.2.4 Constrained Parameter Identification Analysis

The continuity and boundedness properties of the identified parameter $\hat{\theta}$ are crucial for establishing the closed-loop stability of the adaptive control system. In this section, we establish these properties by borrowing tools from the optimization field to analyze the optimization problem (4.24).

When $\gamma > 0$, (4.24) has a strictly convex cost. Since the feasible set is nonempty, closed, and convex, from Lemma A.1, there exists a unique optimal solution. Let $f(\chi, \theta, \gamma)$ represent the cost function of (4.24), where $\chi = [vec(P), H]$. Let $P^*(\theta, \gamma)$ and $H^*(\theta, \gamma)$ represent the optimal solution. $\hat{\theta}(\theta, \gamma) = H^*(\theta, \gamma)P^*(\theta, \gamma)^{-1}$.

Lemma IV.1. $\hat{\theta}(\theta, \gamma)$ *has the following properties:*

(i) $\hat{\theta}(\theta, \gamma)$ *is Lipschitz continuous w.r.t.* θ *and* γ *when* $\gamma > 0$.

(ii) When θ is stable, $\lim_{\gamma \to 0} \hat{\theta}(\theta, \gamma) = \theta$, i.e., $\hat{\theta}(\theta, \gamma)$ is Lipschitz continuous w.r.t. γ when $\gamma \geq 0$.

Proof: (i) The Lipschitz continuity of $P^*(\theta, \gamma)$ and $H^*(\theta, \gamma)$ are first proven using Lemma A.3 of Appendix A: The second-order growth condition holds because the cost function $f(\chi, \theta, \gamma)$ is a strictly convex quadratic function of χ when $\gamma > 0$. Consider the difference between $f(\chi, \theta_0, \gamma)$ and $f(\chi, \theta, \gamma)$, namely $||\theta P - H||_2^2 - ||\theta_0 P - H||_2^2$. It is Lipschitz continuous w.r.t. P and H, modulus $c||\theta - \theta_0||$ for some $c > 0$ for bounded P, H, and θ.

Applying Lemma A.3, $||\chi^*(\theta, \gamma) - \chi^*(\theta_0, \gamma)|| \leq c||\theta - \theta_0||$ for some $c > 0$. Therefore, the optimal solution $P^*(\theta, \gamma)$ and $H^*(\theta, \gamma)$ are Lipschitz continuous w.r.t. θ with $\gamma > 0$.

$\hat{\theta}(\theta, \gamma) = H^*(\theta, \gamma) P^*(\theta, \gamma)^{-1}$. Since $P^*(\theta, \gamma) \succeq \epsilon_0 I$, $\hat{\theta}(\theta, \gamma)$ is also Lipschitz continuous w.r.t. θ when $\gamma > 0$.

Similarly, $\hat{\theta}(\theta, \gamma)$ is Lipschitz continuous w.r.t. γ when $\gamma > 0$.

(ii) Let $v^*(\theta, \gamma)$ represent the optimal value of the cost function. When $\gamma = 0$, and θ is stable, the optimal cost value $v^*(\theta, 0) = 0$. Therefore, $\theta P^*(\theta, 0) - H^*(\theta, 0) = 0$, and $\hat{\theta}(\theta, 0) = \theta$.

From Lemma A.2 of Appendix A, $\lim_{\gamma \to 0} v^*(\theta, \gamma) \leq v^*(\theta, 0) = 0$. $v^*(\theta, \gamma)$ is a non-negative quadratic function. Thus,

$$\lim_{\gamma \to 0} v^*(\theta, \gamma) = \lim_{\gamma \to 0} (||\theta P^*(\theta, \gamma) - H^*(\theta, \gamma)||_2^2 + \gamma ||[vec(P^*(\theta, \gamma)), H^*(\theta, \gamma)]||_2^2) = 0,$$

which implies

$$\lim_{\gamma \to 0} (\theta P^*(\theta, \gamma) - H^*(\theta, \gamma)) = 0.$$

Thus,

$$\lim_{\gamma\to 0}(\theta - H^*(\theta,\gamma)P^*(\theta,\gamma)^{-1}) = \lim_{\gamma\to 0}(\theta - \hat{\theta}(\theta,\gamma)) = 0.$$

Therefore, when θ is stable, $\lim_{\gamma\to 0}\hat{\theta}(\theta,\gamma) = \theta$.

□

Theorem IV.2. *For a linear parametric model $z = \theta^{*T}\phi$, using the unconstrained hybrid gradient method to estimate $\theta(k+1) = \theta(k) + \Gamma \int_{kT_s}^{(k+1)T_s} \epsilon(t)\phi(t)dt$, the constrained adaptive law presented in Section 4.2.3 guarantees that*

(i) $\hat{\theta} \in l_\infty$.

(ii) $\Delta\hat{\theta} \in l_2$, where $\Delta\hat{\theta}(k) = \hat{\theta}(k+1) - \hat{\theta}(k)$.

(iii) $\hat{\epsilon}, \hat{\epsilon}m \in \mathcal{L}_\infty$, where $\hat{\epsilon}(t) = \frac{z(t) - \hat{\theta}(k)^T\phi(t)}{m(t)^2}$, $m(t)^2 = 1 + \phi(t)^T\phi(t)$, $\forall kT_s \leq t < (k+1)T_s$.

(iv) If $\theta(k)$ is stable $\forall k$, $\lim_{\gamma\to 0}\hat{\epsilon} = \epsilon$, $\lim_{\gamma\to 0}\hat{\epsilon}m = \epsilon m$.

Proof: (i) From Lemma II.4, $\theta \in l_\infty$. From Lemma A.1, $P^*, M^* \in l_\infty$. Since $\hat{\theta} = H^*P^{*-1}$, $P^* \succeq \epsilon_0 I$, we can conclude that $\hat{\theta} \in l_\infty$.

(ii) From Lemma II.4, $\Delta\theta_k \in l_2$. From Lemma IV.1 (i), $\hat{\theta}$ is Lipschitz continuous w.r.t. θ, i.e. $\exists c > 0$, $||\Delta\hat{\theta}_k|| \leq c||\Delta\theta_k||$. Therefore, $\Delta\hat{\theta} \in l_2$.

(iii) $\hat{\epsilon}(t) = \epsilon(t) + \frac{(\theta(k) - \hat{\theta}(k))^T\phi(t)}{m(t)^2}$, $\forall kT_s \leq t < (k+1)T_s$. From Lemma II.4, $\epsilon \in \mathcal{L}_\infty$, $\frac{\phi}{m^2} \in \mathcal{L}_\infty$, $\theta - \hat{\theta} \in l_\infty$. Therefore, $\hat{\epsilon} \in \mathcal{L}_\infty$. Similarly, $\hat{\epsilon}m \in \mathcal{L}_\infty$.

(iv) $\hat{\epsilon}(t) = \epsilon(t) + \frac{(\theta(k) - \hat{\theta}(k))^T\phi(t)}{m(t)^2}$, $\forall kT_s \leq t < (k+1)T_s$. From Lemma IV.1 (ii), when $\theta(k)$ is stable, $\lim_{\gamma\to 0}\hat{\theta}(k) = \theta(k)$. Since $\frac{\phi(t)}{m(t)^2} \in \mathcal{L}_\infty$, $\lim_{\gamma\to 0}\hat{\epsilon} = \epsilon$. Similarly, $\lim_{\gamma\to 0}\hat{\epsilon}m = \epsilon m$.

□

4.3 Stability Proof of n-th order CAIMC in the Ideal Case

In this section, the stability and asymptotic performance of the n-th order CAIMC for the ideal case is established. Before the analysis, the CAIMC scheme is summarized as following:

Plant:
Model: $y = \left\{ \frac{Z_M(s)}{R_M(s)} \right\} u$. Inverse: $u = \left\{ \frac{Z_Q(s)}{R_Q(s)} \right\} y$.
Parametric Model:
Model: $z_M = \theta_M^{*T} \phi_M$ with z_M and ϕ_M defined in (4.4). $\theta_M^* = [\theta_a^{*T}, \theta_b^{*T}]^T$, where θ_a^* is the coefficient vector of $\Lambda_M - R_M(s)$ and θ_b^* is the coefficient vector of $Z_M(s)$.   Inverse: $z_Q = \theta_Q^{*T} \phi_Q$ with z_Q and ϕ_Q defined in (4.14). $\theta_Q^* = [\theta_c^{*T}, \theta_d^{*T}]^T$, where θ_c^* is the coefficient vector of $Z_Q(s)$, and θ_d^* is the coefficient vector of $\Lambda_Q - R_Q(s)$.
Adaptive Law:
θ_M and θ_Q are identified by the unconstrained hybrid adaptive law (2.19). $\hat{\theta}_M$ and $\hat{\theta}_Q$ are identified by the constrained parameter identification approach in Section 4.2.3, where the stability constraints are imposed on $\hat{\theta}_a$, $\hat{\theta}_c$, and $\hat{\theta}_d$ and the bandwidth constraints are imposed on $\hat{\theta}_d$.
Control Law:
$u = \left\{ (\Lambda_Q - \hat{R}_Q(s)) \frac{1}{\Lambda_Q} \right\} u + \left\{ \hat{Z}_Q(s) \frac{1}{\Lambda_Q} \right\} l$ as in (4.16),   where $l = r - y + y_M$, $y_M = \hat{\theta}_M^T \phi_M$ as in (4.6). $\hat{R}_Q$ and $\hat{Z}_Q$ are the identified R_Q and Z_Q with $\hat{\theta}_Q$.

Remark IV.1. In general, the inverse (4.2) is an approximate inverse of (4.1). When the plant is strictly proper, some residual unmodeled dynamics are unavoidable when using a proper transfer function to represent the inverse dynamics. For the simplicity

of the presentation, the stability proof and asymptotic performance are established for the ideal case without unmodeled dynamics. The ideal case assumes that the plant is **stable, minimum phase, and has relative degree zero**. The effects of the unmodeled dynamics are discussed later.

Theorem IV.3. *Consider the plant (4.1), (4.2) subject to the CAIMC scheme without unmodeled dynamics. For any bounded reference r, all the signals in the closed-loop system are uniformly bounded. When θ_M and θ_Q identified from the unconstrained adaptive law are Hurwitz stable $\forall t > t_c$, where t_c is a finite number, the tracking error $e = r - y$ converges to zero as $\gamma \to 0$.*

Proof: Given that Λ_M in (4.4) and Λ_Q in (4.14) are Hurwitz polynomials that serve as the regressor filters, we choose $\Lambda_M = \Lambda_Q = \Lambda$ with the coefficients θ_λ throughout the proof. Note that the same analysis can be carried out with arbitrary choice of Hurwitz Λ_M and Λ_Q at the expense of some additional algebra. Define that

$$y_f = \left\{\frac{1}{\Lambda}\right\} y, u_f = \left\{\frac{1}{\Lambda}\right\} u, \epsilon_f = \left\{\frac{1}{\Lambda}\right\} (\hat{\epsilon}_M m_M^2), \qquad (4.25)$$

and establish signal boundedness in the following steps:

Step 1:

Correlate u and y to the estimation error: Defining the augmented states x as $[y_f, y_f^{(1)}, ..., y_f^{(n-1)}, u_f, u_f^{(1)}, ..., u_f^{(n-1)}, \epsilon_f, \epsilon_f^{(1)}, ..., \epsilon_f^{(n-1)}]^T$, we have

$$\dot{x} = A(t)x + b_1(t)\hat{\epsilon}_M m_M^2 + b_2 \bar{r},$$

$$\begin{bmatrix} y \\ u \end{bmatrix} = C(t)x + d_1(t)\hat{\epsilon}_M m_M^2 + d_2 \bar{r}. \qquad (4.26)$$

59

where

$$A(t) = \begin{bmatrix} 0_{(n-1)\times 1}|I_{n-1} & 0_{(n-1)\times n} & 0_{(n-1)\times n} \\ (\hat{\theta}_a - \theta_\lambda)^T & \hat{\theta}_b^T & 0_{1\times n} \\ \hline 0_{(n-1)\times n} & 0_{(n-1)\times 1}|I_{n-1} & 0_{(n-1)\times n} \\ 0_{1\times n} & (\hat{\theta}_d - \theta_\lambda)^T & -(\hat{\theta}_{cn} - \eta\theta_\lambda)^T \\ \hline 0_{(n-1)\times n} & 0_{(n-1)\times n} & 0_{(n-1)\times 1}|I_{n-1} \\ 0_{1\times n} & 0_{1\times n} & -\theta_\lambda^T \end{bmatrix},$$

$$b_1(t) = \begin{bmatrix} 0_{(n-1)\times 1} \\ 1 \\ \hline 0_{(n-1)\times 1} \\ \eta \\ \hline 0_{(n-1)\times 1} \\ 1 \end{bmatrix}, b_2(t) = \begin{bmatrix} 0_{n\times 1} \\ \hline 0_{(n-1)\times 1} \\ 1 \\ \hline 0_{n\times 1} \end{bmatrix},$$

$$C(t) = \begin{bmatrix} \hat{\theta}_a^T & \hat{\theta}_b^T & 0_{(1\times n)} \\ 0_{(1\times n)} & \hat{\theta}_d^T & -(\hat{\theta}_{cn} - \eta)^T \end{bmatrix},$$

$$d_1(t) = \begin{bmatrix} 1 \\ \eta \end{bmatrix}, d_2(t) = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

where $\begin{bmatrix} \hat{\theta}_{cn} \\ \eta \end{bmatrix} = \hat{\theta}_c$, $\hat{\theta}_{cn} \in \mathcal{R}^n$ and η is the $(n+1)$-th entry of $\hat{\theta}_c$, $\bar{r} = \{\hat{Z}_Q(s)\frac{1}{\Lambda}\}r$.

The derivation (4.26) are given in Appendix C.

Step 2:

Establish the exponential stability of the homogeneous part of (4.26): A(t) has a block upper-triangular structure, whose eigen-values are the same as the eigen-values of its diagonal matrices, i.e., for each fixed time t, $A(t)$ has the same

eigen-values as

$$A_1 = \begin{bmatrix} 0_{(n-1)\times 1}|I_{n-1} \\ (\hat{\theta}_a - \theta_\lambda)^T \end{bmatrix}, A_2 = \begin{bmatrix} 0_{(n-1)\times 1}|I_{n-1} \\ (\hat{\theta}_d - \theta_\lambda)^T \end{bmatrix}, \text{ and } A_3 = \begin{bmatrix} 0_{(n-1)\times 1}|I_{n-1} \\ -\theta_\lambda^T \end{bmatrix}.$$

Since $\hat{\theta}_a$ is the coefficient vector of $\Lambda - \hat{R}_M(s)$, and $\hat{\theta}_d$ is the coefficient vector of $\Lambda - \hat{R}_Q(s)$, the eigen-values of A_1 and A_2 are the solutions of $\hat{R}_M(s) = 0$ and $\hat{R}_Q(s) = 0$ respectively, which have negative real parts $\forall t \geq 0$ because of the constraint enforced in deriving the estimated parameters, as discussed in Section 4.2. The eigen-values A_3 are the solutions of $\Lambda = 0$ which also have negative real parts.

By Theorem IV.2, the constrained parameter identification guarantees that $\hat{\theta}_a, \hat{\theta}_b,$ $\hat{\theta}_c, \hat{\theta}_d \in l_\infty$, $\Delta\hat{\theta}_a, \Delta\hat{\theta}_b, \Delta\hat{\theta}_c, \Delta\hat{\theta}_d \in l_\infty \cap l_2$. Their zero-order hold (ZOH) signals are used in $A(t)$. Thus, $A(t)$ is piecewise differentiable w.r.t. t. $\|A(t)\| \in \mathcal{L}_\infty$. Applying Lemma B.1, let $k_0 = T_s$, the system is exponentially stable, and the state transition matrix $\Phi(t,\tau)$ associated with $A(t)$ satisfies $\|\Phi(t,\tau)\| \leq \kappa_1 e^{-\kappa_2(t-\tau)}, \forall t \geq \tau \geq 0$ for some constants $\kappa_1, \kappa_2 > 0$.

Step 3:

Establish signal boundedness: The $\mathcal{L}_{2\delta}$ norm $\|(\bullet)_t\|_{2\delta}$ for some $\delta > 0$ is the exponentially weighted $\mathcal{L}_2$ norm defined as $\|x_t\|_{2\delta} := (\int_0^t e^{-\delta(t-\tau)} x^T(\tau) x(\tau) d\tau)^{\frac{1}{2}}$. Applying Lemma B.2 to the state space equation (4.26), we can obtain

$$\|x_t\|_{2\delta} \leq c\|(\hat{\epsilon}_M m_M^2)_t\|_{2\delta} + c, \tag{4.27}$$

$$|x(t)| \leq c\|(\hat{\epsilon}_M m_M^2)_t\|_{2\delta} + c,$$

where $|\bullet|$ is a vector norm, for any $\delta \in [0, \delta_1)$ where $\delta_1 > 0$ is any constant less than $2\kappa_2$ and some finite constant $c \geq 0$. For simplicity of the representation, in this thesis, c is used to represent a generic constant.

61

We define the fictitious normalizing signal $m_f^2 := 1 + \|u_t\|_{2\delta}^2 + \|y_t\|_{2\delta}^2$. From the state space equation, we have $\|u_t\|_{2\delta} + \|y_t\|_{2\delta} \leq c\|x_t\|_{2\delta} + c\|(\hat{\epsilon}_M m_M^2)_t\|_{2\delta} + c$. With equation (4.27), we have $\|u_t\|_{2\delta} + \|y_t\|_{2\delta} \leq c\|(\hat{\epsilon}_M m_M^2)_t\|_{2\delta} + c$, implying

$$m_f^2 \leq c\|(\hat{\epsilon}_M m_M^2)_t\|_{2\delta}^2 + c.$$

From (4.4), applying Lemma B.3,

$$m_M = \sqrt{1 + \phi_M^T \phi_M} \leq cm_f, \tag{4.28}$$
$$m_f^2 \leq c\|(\tilde{g} m_f)_t\|_{2\delta}^2 + c,$$

where $\tilde{g} = \hat{\epsilon}_M m_M \in \mathcal{L}_{2e}$. Or

$$m_f^2 \leq c \int_0^t e^{-\delta(t-\tau)} \tilde{g}^2(\tau) m_f^2(\tau) d\tau + c,$$

where $0 < \delta \leq \delta^*$ and $\delta^* = min[2\lambda, \delta_1]$, $\delta_1 \in (0, 2\kappa_2)$.

Applying Lemma B.4 the B-G lemma, we can establish that $m_f \in \mathcal{L}_\infty$. Then with (4.28), we have $m_M \in \mathcal{L}_\infty$ and therefore $\phi_M, x, \dot{x}, u, y \in \mathcal{L}_\infty$, $\epsilon_Q m_Q^2 \in \mathcal{L}_\infty$, $m_Q = \sqrt{1 + \phi_Q^T \phi_Q} \in \mathcal{L}_\infty$. All the signals in the closed-loop system are uniformly bounded.

Step 4:

Establish that the tracking error e converges to 0:

Since the tracking error e can be expressed as the sum of $e_M = y - y_M$ and $e_Q = l - y$. We can establish the convergence of e by demonstrating the convergence of e_M and e_Q respectively.

First, we consider the forward model estimation error equation $\hat{\epsilon}_M m_M^2 = y - y_M$: With the assumption that θ_M is satisfy the constraints, according to Theorem IV.2,

$\hat{\epsilon}_M m_M^2 \to \epsilon_M m_M^2$ as $\gamma \to 0$.

There exists a continuous signal $\bar{\theta}_M$ such that $|\bar{\theta}_M - \theta_M| \in \mathcal{L}_2$ and $\dot{\bar{\theta}}_M \in \mathcal{L}_2 \cap \mathcal{L}_\infty$. Let $\bar{\epsilon}_M m_M^2 = z_M - \bar{\theta}_M^T \phi_M$. $\bar{\epsilon}_M m_M \in \mathcal{L}_2 \cap \mathcal{L}_\infty$, because $\epsilon_M m_M \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ based on Lemma II.2. $m_M \in \mathcal{L}_\infty$ from Step 3. Therefore, $\bar{\epsilon}_M m_M^2 \in \mathcal{L}_2 \cap \mathcal{L}_\infty$. Since $\dot{y}, x, \dot{x}, \bar{\theta}_M, \dot{\bar{\theta}}_M \in \mathcal{L}_\infty$, $\frac{d}{dt}(\bar{\epsilon}_M m_M^2) \in \mathcal{L}_\infty$. $\bar{\epsilon}_M m_M^2 \to 0$. It follows that $e_M = \hat{\epsilon}_M m_M^2 \to \epsilon_M m_M^2 \to \bar{\epsilon}_M m_M^2 \to 0$ as $t \to \infty$.

Following the same procedure, it can be shown that $\hat{\epsilon}_Q m_Q^2 \to 0$ as $t \to \infty$. From Lemma II.2, we have $\Delta\theta_Q \in l_2 \cap l_\infty$, $\Delta\theta_Q \to 0$. From (4.17), we have $e_Q = \epsilon_Q m_Q^2 - \epsilon_3$, and $\epsilon_3 \to 0$ since $\Delta\theta_Q \to 0$. Therefore, $e_Q \to 0$.

$e = e_M + e_Q$. Therefore, $e \to 0$ as $t \to \infty$ and $\gamma \to 0$, when θ_M and θ_Q satisfy the constraints.

<div align="right">□</div>

Remark IV.2. Note that ϵ_Q does not appear in the closed-loop representation (4.26), therefore the property of ϵ_Q is not required for establishing the stability of CAIMC. It is only needed for establishing the convergence of the tracking error.

Remark IV.3. Theorem IV.3 shows that the tracking error $e \to 0$ as $\gamma \to 0$. But γ has to be non-zero to assure that the optimization problem (4.24) has an unique optimal solution. Therefore, we will discuss here the implication when γ is a very small non-zero number.

According to Lemma IV.1, when θ satisfies the constraints, $\hat{\theta}(\theta, \gamma)$ is Lipschitz continuous w.r.t. $\gamma \geq 0$. Therefore, $\hat{\epsilon}_M m_M^2 = z_M - \hat{\theta}_M^T \Phi_M$ and $\hat{\epsilon}_Q m_Q^2 = z_Q - \hat{\theta}_Q^T \Phi_Q$ are Lipschitz continuous w.r.t. $\gamma \geq 0$, which implies that e is Lipschitz continuous w.r.t. $\gamma \geq 0$. Therefore, when θ satisfies the constraints and $\gamma << 1$, e is small.

Remark IV.4. For simplicity and clarity, the stability analysis is performed under the assumption that there are no unmodeled dynamics. As discussed in Remark IV.1, in general there are unmodeled dynamics in the presentation of the physical plant or its inverse dynamics. To handle the unmodeled dynamics, generally, a deadzone is

added to the estimation error for robust estimation [78]. The robust CAIMC stability proof follows a very similar procedure by expanding the proof here as shown in [78] for robust adaptive pole placement control, and its tracking error e is bounded.

4.4 Simulation Results on a Third-order LTI Plant

In this section, CAIMC scheme as summarized in Section 4.3 is applied to a third-order LTI plant. The constrained parameter identification approach proposed in Section 4.2 is used to identify the unknown parameters of the plant dynamics and its inverse simultaneously.

Given an LTI plant

$$y = \frac{\theta_{b0}^* s^3 + \theta_{b1}^* s^2 + \theta_{b2}^* s + \theta_{b3}^*}{s^3 + \theta_{a1}^* s^2 + \theta_{a2}^* s + \theta_{a3}^*} u, \tag{4.29}$$

whose inverse is

$$u = \frac{\theta_{c0}^* s^3 + \theta_{c1}^* s^2 + \theta_{c2}^* s + \theta_{c3}^*}{s^3 + \theta_{d1}^* s^2 + \theta_{d2}^* s + \theta_{d3}^*} y, \tag{4.30}$$

where all the θ^* are unknown, $\theta_a^* = [2, 2, 3]^T, \theta_b^* = [2, 3, 1, 1]^T, \theta_c^* = [0.5, 1, 1, 1.5]^T, \theta_d^* = [1.5, 0.5, 0.5]^T$. The plant is stable, minimum-phase, and has relative degree zero. There are no unmodeled dynamics. The plant is third-order, thus the feasible region that satisfies the stability constraint in the parameter space in nonconvex.

CAIMC scheme as summarized in Section 4.3 is applied to the plant. The plant (4.29) and its inverse (4.30) are identified simultaneously using the constrained parameter identification approach proposed in Section 4.2. The constraints are imposed by the stability of θ_a and θ_d. At each sample time, unconstrained adaptive law (2.17) is first used to identify θ, then the constrained optimization problem (4.24) is solved to find $\hat{\theta}$. The initial conditions of the parameters are $\theta_a(0) = [10, 0.5, 4]^T, \theta_b(0) =$
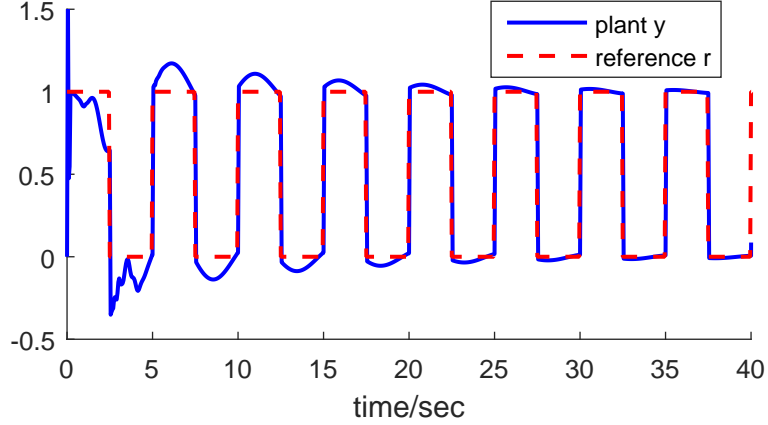
Figure 4.2: CAIMC simulation result.

$[1.5, 3.5, 2, 1.5]^T, \theta_c(0) = [0.8, 1.5, 1, 1.2]^T, \theta_d(0) = [5, 0.3, 0.7]^T.$

The closed-loop result is shown in Figure 4.2. The reference r is a square wave with period $5sec$ and amplitude 1. The plant response y tracks the reference, and the performance is improved with the online identification of the parameters.

Due to the space limitation, not all identified parameters are shown here. The identified parameters for θ_d^* at $0 - 6sec$ are shown in Figure 4.3. The initial condition $\theta_d(0)$ is stable, and the true θ_d^* is stable, however, θ_d from the unconstrained adaptive law is unstable in the shaded area around $0.2 - 3.8sec$. $\hat{\theta}_d$ from the constrained optimization problem is always stable, and when θ is stable, $\hat{\theta}_d \approx \theta_d$. The result is consistent with the properties established in Lemma IV.1.
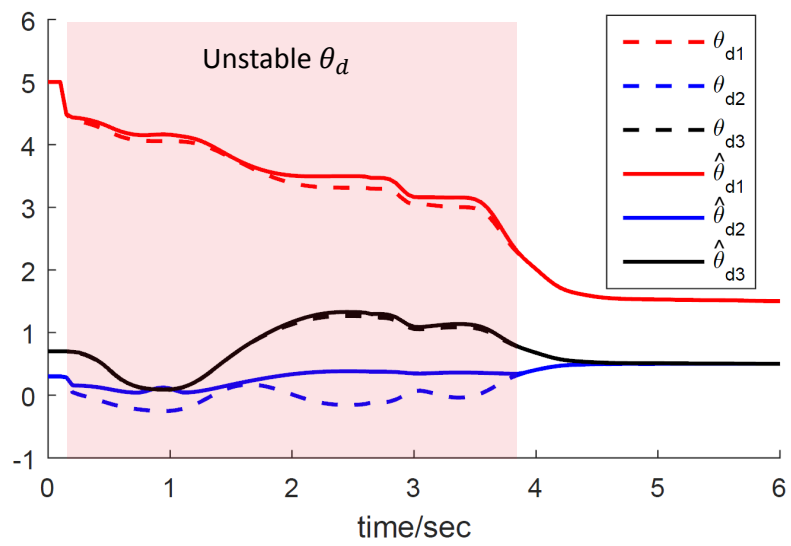
Figure 4.3: CAIMC simulation parameters.

CHAPTER V

Application of CAIMC to the Boost-Pressure Control Problem of a Turbocharged Gasoline Engine

In this chapter, CAIMC is applied to the boost-pressure control problem of a turbocharged gasoline engine. In Section 5.1, the schematic of a turbocharged gasoline engine is introduced, and the boost-pressure control problem is discussed. In Section 5.2.1, the model and inverse structures are proposed as first-order linear structures, and their parametrization and identification are discussed in Section 5.2.2. In Section 5.3, CAIMC-LI and CAIMC-RI are implemented on a high-fidelity turbocharged gasoline engine model. In Section 5.4, CAIMC-RI is applied on the Ford Explorer EcoBoost with a 2.0 L four-cylinder engine. Its performance is compared to the baseline controller, which is a PI controller with feedforward. CAIMC demonstrates similar performance to the baseline, and great advantages in terms of calibration.

5.1 Turbocharged Gasoline Engine Boost-Pressure Control Problem Overview

Gasoline engines have been aggressively downsized in an effort to reduce the fuel consumption and CO_2 emissions [74]. However, the torque provided by the engine

is proportional to the air delivered to the cylinders. To meet the consumer demands for performance on the downsized engines, i.e., to maintain the engine output torque, turbochargers are widely adopted due to their higher power density and better fuel economy. The working principle of a turbocharged gasoline engine is shown in Fig. 1.3. The wastegate is the main actuator to control boost-pressure by changing the rotational speed of the turbine/compressor. The air is compressed by the compressor, and passes through an intercooler and a throttle before entering the engine intake port. The engine exhaust port is connected to the turbine, which is mechanically connected to the compressor. A wastegate actuator controls the opening of the turbine bypass path in this application [17], affecting the compressor speed and therefore the boost-pressure. The boost-pressure, which refers to the pressure before the throttle and after the intercooler, is one of the main variables that affect the turbocharged engine performance [75].

The turbocharged gasoline engine is expected to produce the desired engine torque, with higher fuel efficiency and power density [86]. To achieve such goal, the desired engine torque is calculated from the driver pedal position. The desired engine torque is then mapped into desired intake manifold pressure and boost-pressure considering the fuel economy and emission. These two pressures are then tracked through throttle and wastegate. This two-input two-output control problem can often be tackled with a decentralized controller: using the throttle to track the desired intake manifold pressure, and using the wastegate to track the desired boost-pressure [74]. Here we will focus on using the wastegate to track the desired boost-pressure, and the throttle is considered as an exogenous input. Boost-pressure set-point tracking is a critical enabling technology for achieving improved fuel efficiency, power density, and reducing emissions [86].

There are many challenges in the boost-pressure control problem. First, the turbocharged gasoline engine is high-order and nonlinear [41]. Second, an engine has

to operate in different operating points and ambient environment. While providing adequate boost at low speed and load, the turbocharger system also has to avoid over-boost situation at higher speed and load [76]. The tracking error has to be kept small because under-delivering the desired boost is undesirable as it results in under-delivered torque, over-delivering the desired boost-pressure is undesirable as it causes NVH (Noise, Vibration, and Harshness) and compressor surge [74]. CAIMC simultaneously identifies the linear models M and Q and minimizes the tracking error at different operating condition through adaptation, which makes CAIMC very appealing to the boost-pressure control problem. First-order CAIMC as presented in Chapter III is applied to the boost-pressure control problem in the next section.

5.2 Applying CAIMC to the Boost-Pressure Control Problem

5.2.1 Plant and Inverse Dynamic Models

The first step of CAIMC design is to find a feasible model and inverse structure to capture the plant and inverse dynamics. To sufficiently describe the air dynamics of a turbocharged gasoline engine, a fourth-order nonlinear first principle model is often derived [42,43]. Karnik et al. simplified the fourth-order nonlinear model into a model as shown in Fig. 5.1 [17]. It includes a steady-state mapping from the wastegate u_w to the turbo speed N_t, a first-order linear model whose time constant is gain-scheduled, and a compressor map. This structure motivates us to further simplify and propose a model structure as in Fig. 5.2(a), and first-order linear inverse structure as in Fig. 5.2(b). u_w represents the fraction of wastegate closing (0 means open, and 1 means closed), P_b is the boost-pressure, and $P_{b,ss}$ is the boost-pressure steady-state value at the current operating condition. Map A and B are exact inverses of each other. They have $\pm 5\%$ accuracy.
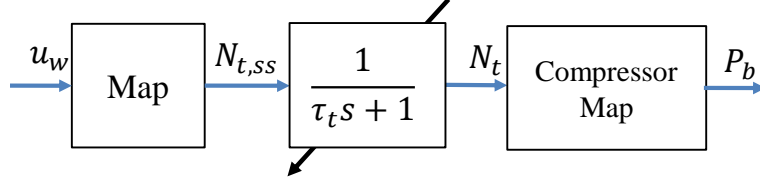
Figure 5.1: First-order gain-scheduled linear model for the turbocharged gasoline engine proposed in [17].
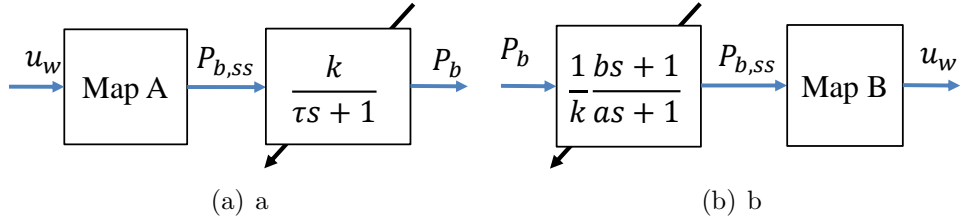

(a) a

(b) b

Figure 5.2: Plant and inverse dynamic models: (a) First-order plant model structure for identification. (b) First-order inverse model structure for identification.

5.2.2 Plant and Inverse Parametrization and Identification

The model and inverse to be identified are $P_b = \frac{k^*}{\tau^* s + 1} P_{b,ss}$ and $P_{b,ss} = \frac{1}{k^*} \frac{b^* s + 1}{a^* s + 1} P_b$. Let $y = P_b$ and $u = P_{b,ss}$, the parametrization in (3.12, 3.16) can be adopted for CAIMC-LI, and the parametrization in (3.12, 3.22) can be adopted for CAIMC-RI. Note that k in the forward model and the inverse are kept the same due to property 1) of IMC. Two ks are identified simultaneously in the model and the inverse, but k from model identification is adopted for controller implementation in this case. This change will affect the tracking error representation. However, k serves as a gain to compensate the mapping inaccuracy, and is always close to 1. Therefore using k identified from the model parametric model is a reasonable approximation.

CAIMC has the layout as shown in Fig. 5.3 after incorporating the identified forward and inverse models as M and Q. The two mappings do not affect the tracking error representation of CAIMC. Note that
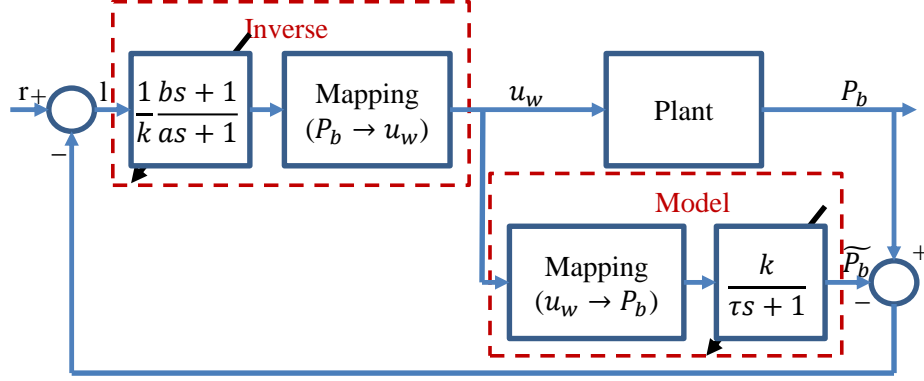
$$e = (P_b - \tilde{P}_b) + (l - P_b),$$

70

Figure 5.3: CAIMC applied to the boost-pressure control of a turbocharged gasoline engine.

where $P_b - \tilde{P}_b = e_M = \epsilon_M m_M^2$, $(l - P_b) = e_Q = \{X\}\epsilon_Q m_Q^2$ for CAIMC-LI, and $l - P_b = e_Q \approx \epsilon_Q m_Q^2$ for CAIMC-RI. The cost functions $J(\theta_1) = \frac{\epsilon_M^2 m_M^2}{2}$ and $J(\theta_2) = \frac{\epsilon_Q^2 m_Q^2}{2}$ are minimized through forward model and inverse model identification respectively.

5.3   Simulation Results

CAIMC-LI and CAIMC-RI are evaluated by applying them to a proprietary model of the turbocharged gasoline engine. Some details of the model can be found in [75]. For comparison, all the calibration parameters and references of CAIMC-LI and CAIMC-RI are kept the same. The calibration parameters are listed in Table. 5.1. For robustness purpose, a which is the filter time constant of the inverse Q is lower-bounded at 0.5 to limit the bandwidth of the IMC control system. When the airflow is low and the desired boost-pressure is low, the wastegate is set to be open by default.

Table 5.1: Parameters for CAIMC simulation.

Identification	$\tau_{M,Q}$	$\theta(0)$	Γ	deadzone
Model	0.1	$\begin{bmatrix} 0.25 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 2 & 0 \\ 0 & 0.5 \end{bmatrix}$	1
Inverse	0.1	$\begin{bmatrix} 0.6 \\ 1 \\ 0.55 \end{bmatrix}$	$\begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 2 \end{bmatrix}$	1

CAIMC-LI and CAIMC-RI are compared in two sets of simulations. In the first simulation, the engine speed is kept constant at 3000 rpm and the desired airflow is kept constant at 100 gps. The simulation results after the identified parameters converge are shown in Fig. 5.4. The performance are very close. CAIMC-RI has slightly less rise time and overshoot. As shown in TABLE 5.2, the RMS tracking errors for CAIMC-RI and CAIMC-LI are 1.855 and 1.894 respectively.

In the second simulation, the engine speed, the desired airflow, and the boost-pressure reference are from vehicle data. The vehicle data was obtained from driving a Ford Explorer EcoBoost with a $2.0L$ engine on the US06 drive cycle, where the engine is boosted for a good fraction of the time. The vehicle speed is shown in Fig. 5.5. It is favorable for testing the boost-pressure control performance. Fig. 5.6 is the simulation result of applying CAIMC-RI and CAIMC-LI to the model with the engine speed, the desired airflow, and the boost-pressure reference from the US06 vehicle data. Both CAIMC-RI and CAIMC-LI track the reference very well. The RMS tracking error for CAIMC-RI is 2.648, whereas for CAIMC-LI is 3.027. Some zoomed-in comparisons of CAIMC-LI and CAIMC-RI are shown in Fig. 5.7. CAIMC-RI has faster rise time and less overshoot in all the simulations. All the RMS error comparisons between CAIMC-RI and CAIMC-LI are summarized in TABLE 5.2; CAIMC-RI has less RMS tracking error in all simulations.

Table 5.2: RMS Errors for CAIMC simulations.

Vehicle speed	constant	US06	zoom 1	zoom 2
CAIMC-RI	1.855	2.648	5.095	3.732
CAIMC-LI	1.894	3.027	5.780	4.257

5.4 Vehicle Testing Experimental Results

CAIMC-RI was implemented and tested on a Ford Explorer EcoBoost with a $2.0L$ cylinder turbocharged gasoline engine with vacuum actuated wastegate. The
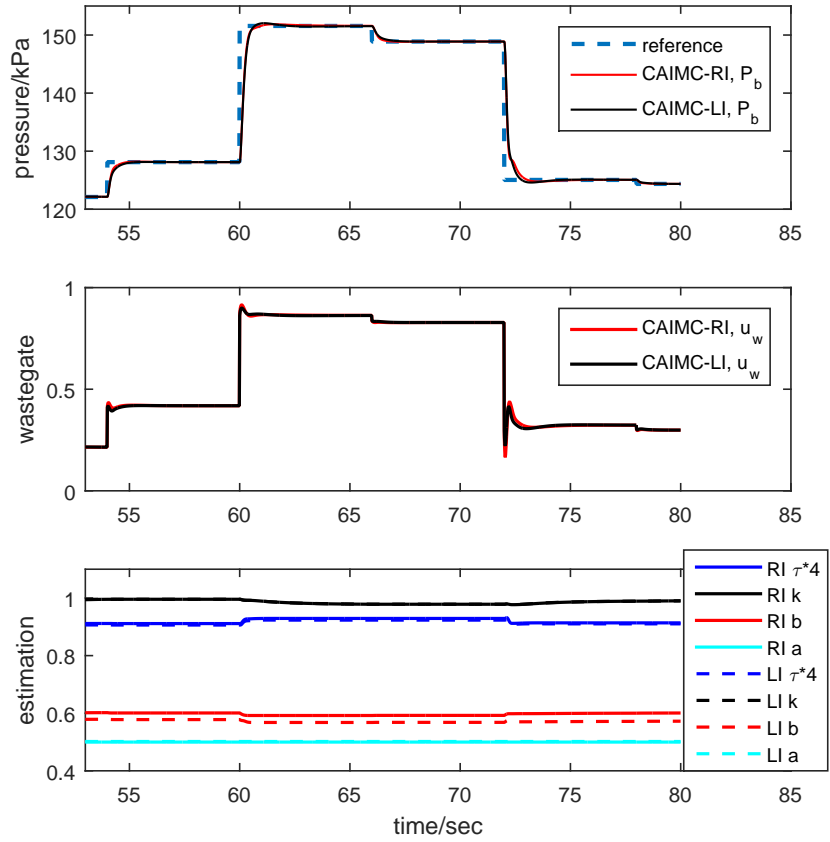
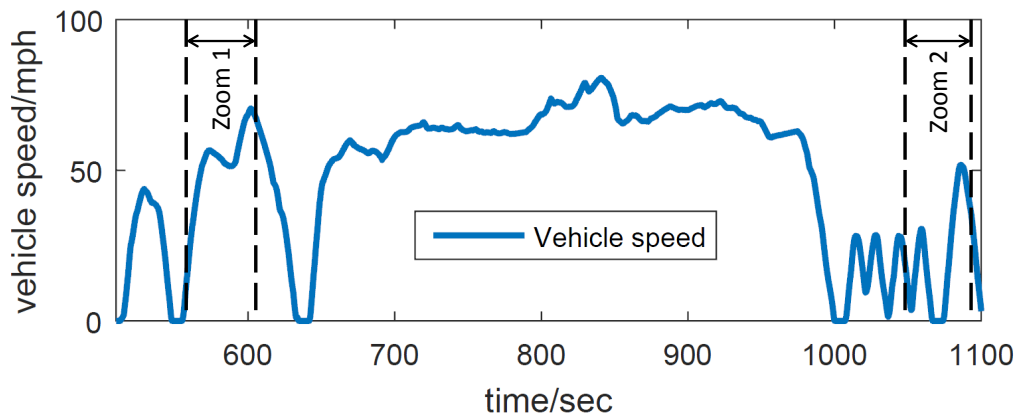Figure 5.4: Simulation result of CAIMC-RI and CAIMC-LI with constant engine speed and desired airflow.
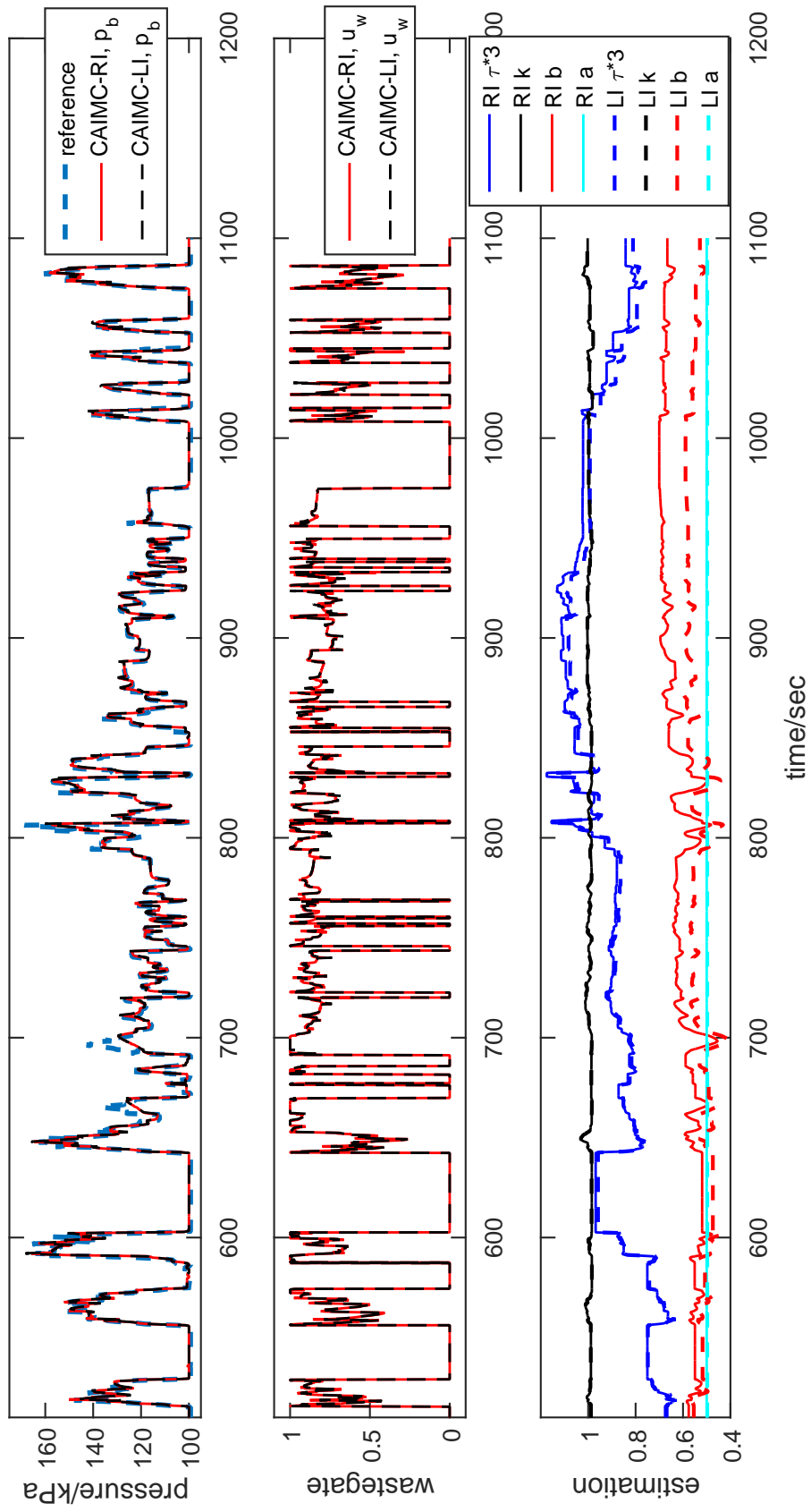

Figure 5.5: US06 speed profile.

73

Figure 5.6: Simulation of CAIMC-RI and CAIMC-LI with the US06 vehicle data.
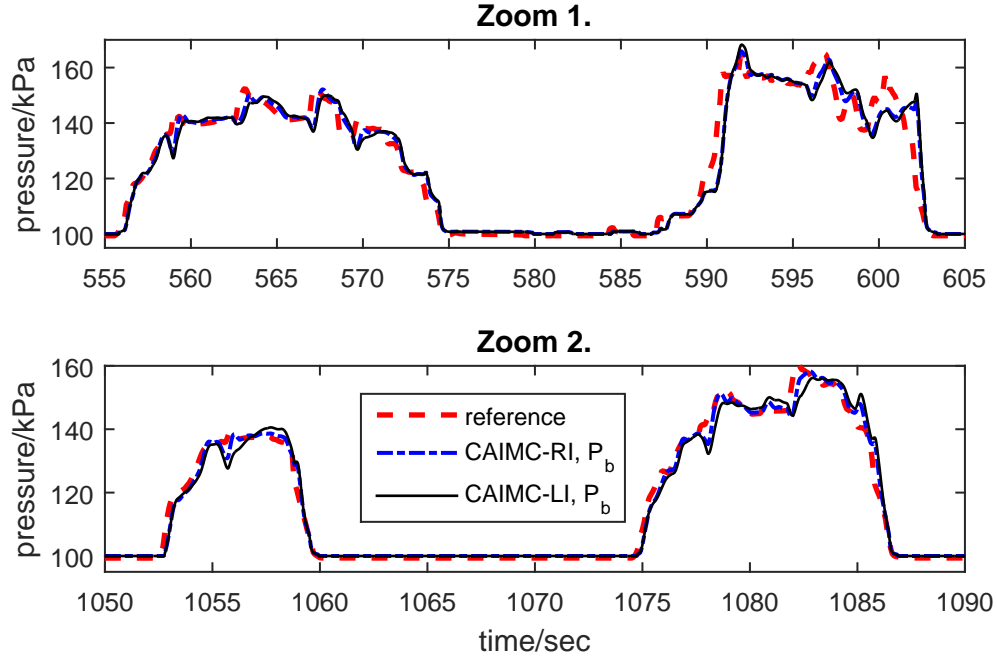
Figure 5.7: Comparison of simulation of CAIMC-LI and CAIMC-RI with the US06 vehicle data.

wastegate canister vacuum determines the position of the wastegate. On the specific vehicle, higher wastegate canister vacuum corresponds to higher boost-pressure, and vice versa. The application of CAIMC on the vehicle is the same as the proprietary model as shown in Fig. 5.3, except that on the vehicle, CAIMC commands the wastegate canister vacuum instead of the wastegate position.

5.4.1 Experimental Setup

For experiments the rapid control prototyping tools as shown in Fig. 5.8 were adopted: the boost control strategy was coded in MATLAB®/Simulink®[1], and then compiled and executed in real time on the hardware platform dSPACE®[2]. dSPACE® can communicate with powertrain control module (PCM) of the vehicle to access vehicle data and command wastegate canister vacuum. The boost control strategy was executing at the sample time of $15ms$. Both the boost-pressure and the wastegate

---

[1]MATLAB and Simulink are registered trademarks of The MathWorks, Inc.
[2]dSPACE is a registered trademark of dSPACE GmbH.

Figure 5.8: Rapid control prototyping process.

canister pressure were measured by pressure sensors. An accurate estimation of the engine airflow was accessible. The identifications of M and Q were performed online simultaneously using the normalized gradient algorithm, and the identified parameters τ, k, a, and b in M and Q were updated at each sample time.

The experiment was performed in a dynamometer by driving the vehicle through a warm-up cycle, and then the US06 cycle as shown in Fig. 5.5. The calibration parameters for vehicle testing are shown in TABLE 5.3.

5.4.2 Experimental Results

The vehicle testing result is shown in Fig. 5.9. The boost-pressure tracks the reference closely, and it has RMS tracking error of 7.076. Zoom-in results are shown in Fig. 5.10. For comparison, the baseline controller has six gain-scheduled look-up tables, and its RMS tracking error is 7.268 for the same test.

Table 5.3: Parameters for CAIMC vehicle testing.

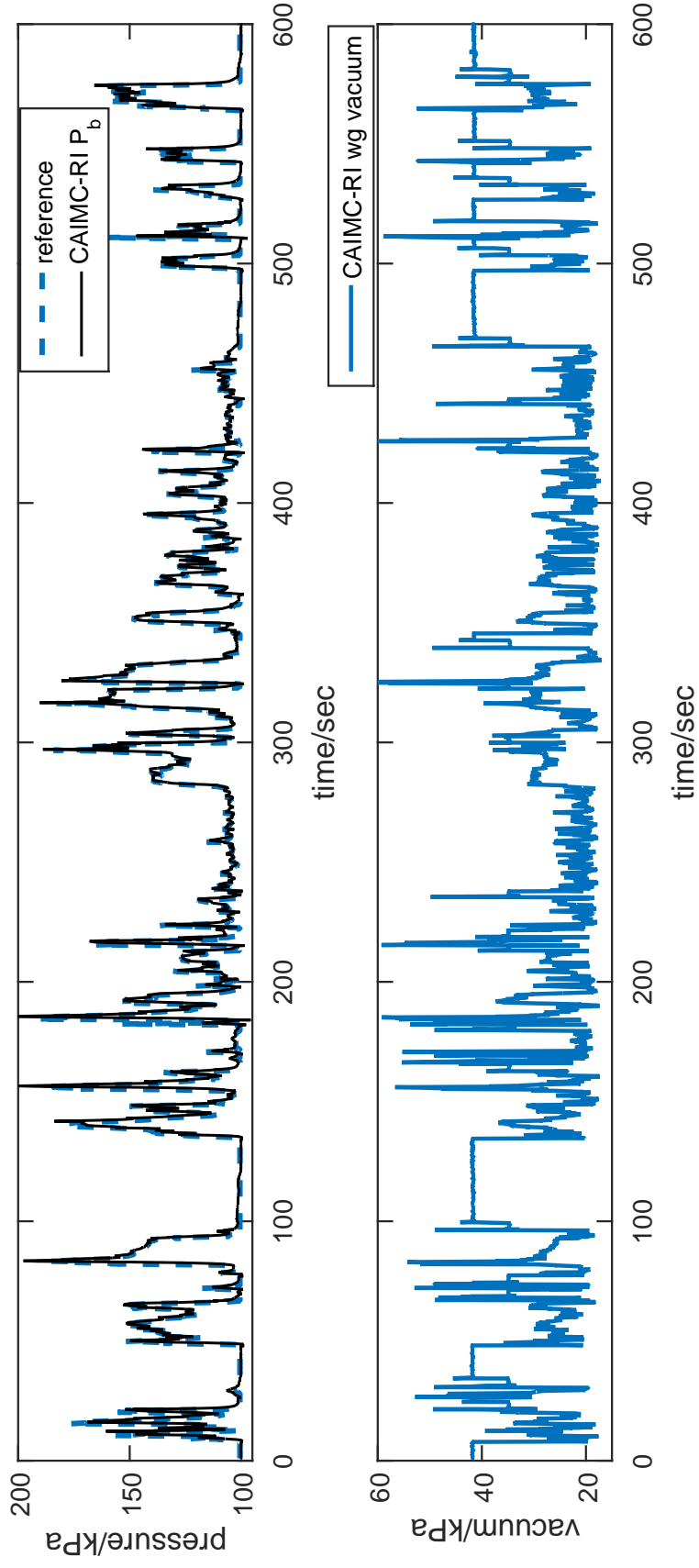| Identification	$\tau_{M,Q}$	$\theta(0)$	Γ	deadzone
Model	0.2	$\begin{bmatrix} 0.35 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$	1
Inverse	0.2	$\begin{bmatrix} 0.65 \\ 1 \\ 1.1 \end{bmatrix}$	$\begin{bmatrix} 2 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 2 \end{bmatrix}$	1

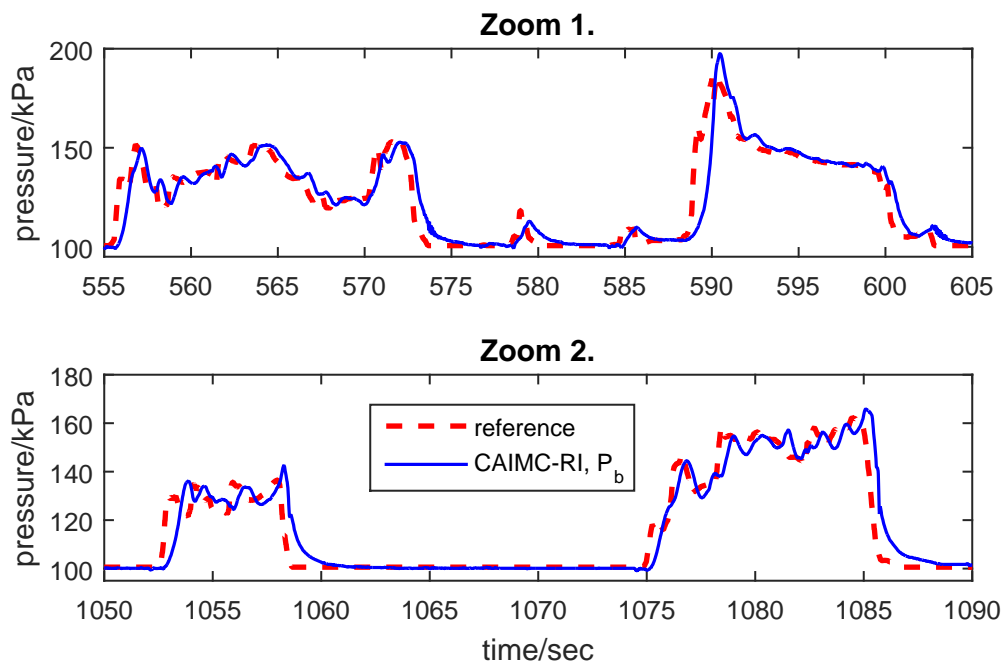Figure 5.9: Vehicle testing result of CAIMC–RI with US06 drive cycle.

Figure 5.10: Zoomed-in vehicle testing result of CAIMC-RI with US06 drive cycle.

CHAPTER VI

Nonlinear IMC Design with Quasi-Linear Parameter Varying Model for the Boost-Pressure Control Problem

This chapter investigates the feasibility, performance, advantages, and limitations of a nonlinear IMC for the boost-pressure control problem of a turbocharged gasoline engine. Inverting the nonlinear model for the IMC design represents the major challenge. To facilitate the IMC design, a quasi-LPV model [40] for the nonlinear model is developed. More importantly, the special quasi-LPV model structure is exploited, and a structured quasi-LPV model is proposed, which leads to a feasible nonlinear inverse, referred to as the structured quasi-LPV inverse. The IMC based on the structured quasi-LPV inverse is developed, and its performance is analyzed. Simulation results, using a validated "virtual" plant model, are presented to demonstrate the effectiveness of the proposed design. Nonlinear IMC is presented in the context of the boost-pressure control of a turbocharged gasoline engine. Its generalization remains to be an open problem.

Section 6.1 presents the main tools used: LPV model. Section 6.2 presents the nonlinear model for the turbocharged gasoline engine. Section 6.3 exploits quasi-LPV approach to derive the inverse of the nonlinear model. Section 6.4 analyzes the IMC

implementation results on a high-fidelity turbocharged gasoline engine model.

6.1 Background on Quasi-Linear Parameter Varying (LPV) Models

Quasi-LPV models are LPV models for nonlinear systems where nonlinearities are hidden through state-dependent parameters, so that a nonlinear model can be represented by an LPV model and treated by LPV design techniques [40].

In general, a nonlinear model in the form of

$$\dot{x} = f(x, u) \tag{6.1}$$

can be expressed as an LPV model in the form of

$$\dot{x} = A(p)x + B(p)u \tag{6.2}$$

if the model (6.1) is affine in u and the time varying parameter vector p in (6.2) is allowed to be state-dependent to disguise the nonlinearities [40]. For example, the nonlinear model

$$\dot{x}_1 = x_1^2 + x_1 x_2, \dot{x}_2 = \sin x_1 + u$$

can be expressed in quasi-LPV form as

$$\dot{x} = A(p)x + Bu = \begin{bmatrix} x_1 & x_1 \\ \frac{sin(x_1)}{x_1} & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u,$$
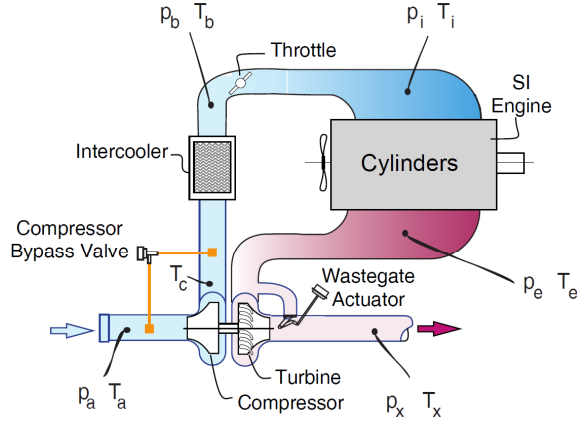
Figure 6.1: System schematic of a turbocharged gasoline engine [75].

with $p = [x_1, \frac{sin(x_1)}{x_1}]^T$, or

$$\dot{x} = A(p)x + Bu = \begin{bmatrix} x_1 + x_2 & 0 \\ \frac{sin(x_1)}{x_1} & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u,$$

with $p = [x_1 + x_2, \frac{sin(x_1)}{x_1}]^T$.

In the next section, a nonlinear model for the turbocharged gasoline engine is presented. The quasi-LPV approach is exploited for representing it in a linear structure to aid deriving an inverse for the nonlinear model in IMC implementation.

6.2 A Nonlinear Turbocharged Gasoline Engine Model For IMC Design

Control-oriented models serve the IMC design and implementation in two different ways: first, the IMC incorporates a system model directly in its implementation as shown in Fig. 1.3; second, the standard IMC design procedure takes an inverse of the process model and augments it with a proper filter to avoid non-causal implementation to form the inverse Q. The nonlinear model for the boost-pressure dynamics of a turbocharged engine presented is based on the work of [75]. The nonlinear model has

the following states and one input:

$$x = [P_b, P_i, P_e, N_t]^T, u = u_w,$$

where P_b is the boost-pressure, P_i is the intake pressure, P_e is the exhaust pressure, N_t is the turbocharger speed, and the input u_w is the wastegate, which is the fraction of the opening and takes values in the range of $[0, 1]$. The dynamics of the pressures P_b, P_i, and P_e are derived using mass conservation along with isothermal manifold assumptions, while the dynamics of the turbocharger speed N_t are derived by a power balance between the turbine and the compressor as described in [17,75]. The equations are summarized as follows:

$$
\begin{aligned}
\frac{dP_b}{dt} &= \frac{RT_b}{V_b}(W_c - W_{th}), \\
\frac{dP_i}{dt} &= \frac{RT_i}{V_i}(W_{th} - W_{en}), \\
\frac{dP_e}{dt} &= \frac{RT_e}{V_e}(W_{en}\frac{1 + A/F}{A/F} - W_t - W_w), \\
\frac{dN_t}{dt} &= \frac{1}{I_t N_t}(H_t - H_c),
\end{aligned}
\tag{6.3}
$$

where R is the ideal gas constant, A/F is the air to fuel ratio, T, V, W, I and H are temperature, volume, mass flow rate, inertia, and power respectively. The subscript indicates the physical location of the variable as in Fig. 6.1, and b, c, th, i, en, e, t, and w are boost, compressor, throttle, intake, engine, exhaust, turbine, and wastegate respectively. Modeling of W (mass flow rate) and H (power) are described in detail

in [17, 75], and the resulting functional expressions are summarized as follows:

$$W_c = f_c(\frac{P_b}{P_a}, N_t),$$

$$W_{th} = \frac{\text{sat}(0, u_{th}, 1)}{\sqrt{RT_b}} \gamma P_b \phi(\frac{P_i}{P_b}),$$

$$W_{en} = P_i \eta_{en} \frac{V_{en}}{RT_i} \frac{N_{en}}{2},$$

$$W_w = \frac{\text{sat}(0, u_w, 1)}{\sqrt{RT_e}} \gamma P_e \phi(\frac{P_x}{P_e}), \qquad (6.4)$$

$$W_t = f_t(\frac{P_e}{P_x}, \frac{N_t}{\sqrt{T_e}}) \frac{P_e}{\sqrt{T_e}},$$

$$H_t = c_{p,e} T_e W_t \eta_t \psi_t, \psi_t = 1 - (\frac{P_x}{P_e})^{\frac{\gamma_e - 1}{\gamma_e}},$$

$$H_c = c_{p,a} T_a W_c \frac{1}{\eta_c} \psi_c, \psi_c = (\frac{P_b}{P_a})^{\frac{\gamma_a - 1}{\gamma_a}} - 1,$$

where P_a is the ambient pressure, P_x is the turbine exit pressure, $\phi(\cdot)$ is a function of pressure ratio across the component, ψ is a mass flow parameter, γ is the specific heat ratio for air, $c_{p,(\cdot)}$ is the specific heat at constant pressure, η is the isentropic efficiency, u_{th} is the throttle opening, N_{en} is the engine speed, and $\text{sat}(0, u, 1)$ limits u to be in the range $[0, 1]$. The temperatures T_b, T_i, and T_e are assumed to be measured. Typically the temperature sensors have a slow response time and delay, and the measurements are lead filtered to improve the response time. Therefore, the measurement inaccuracy is not considered in this work. A/F is the stoichiometric ratio of gasoline. u_{th} and N_{en} are considered as exogenous inputs in this work and they are measurable. All the variables and subscripts for the nonlinear model of turbocharged gasoline engine are summarized in Table 6.1.

The nonlinear model is evaluated by comparing its responses with those of the virtual "plant", which is a high fidelity Ford proprietary model that has been validated extensively. It includes the intercooler, throttle, engine, wastegate, turbine, and compressor [75]. Responses to a step change in the wastegate setting from 0.25 to 0.75 at $t = 5sec$ for the nonlinear model and the virtual "plant" are shown in Fig.
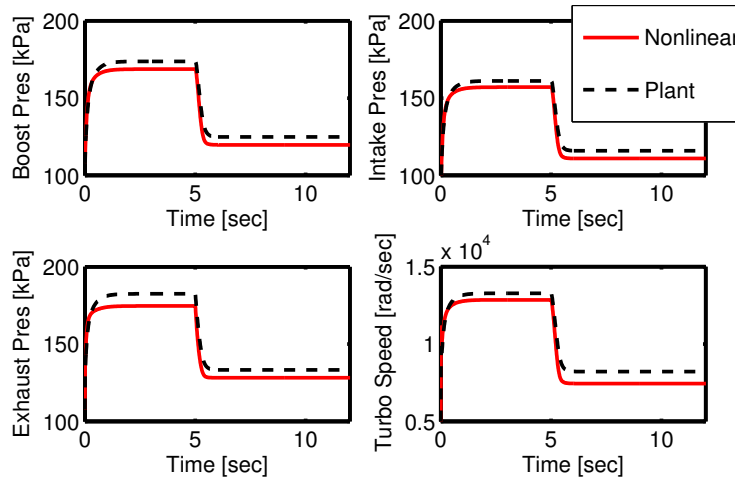
Figure 6.2: Comparison of responses of the nonlinear model and the "plant" for a step change in wastegate actuation.

6.2, confirming that the control-oriented nonlinear model and the "plant" have very similar dynamic responses.

6.3 Quasi-LPV Model and its Inverse

6.3.1 Quasi-LPV Turbocharged Gasoline Engine Model

IMC control can be achieved by designing the inverse Q in Fig. 1.3 as the inverse of the model M. For linear models, their inverses can be achieved by inverting their transfer functions and appending a proper filter to assure causality. To extend this approach to nonlinear models, the quasi-linear parameter varying model approach is explored to represent the nonlinear model in a linear structure.

Note that there are an infinite number of quasi-LPV models in the form of (6.2) that can match (6.1), depending on the choice of the varying parameter p. For the turbocharged gasoline engine system, the physical couplings of state variables are considered and the following structure that leads to the most sparse A, B matrices is

Table 6.1: Nomenclature for modeling of turbocharged gasoline engine.

Variables		Subscripts	
A/F	Air to fuel ratio	a	Ambient
$c_{p,(\cdot)}$	Specific heat at constant pressure	b	Boost
$f.(\cdot)$	Compressor/turbine map	c	Compressor
I	Moment of inertia	e	Exhaust
N	Rotational speed	en	Engine
P	Pressure	i	Intake
H	Power	t	Turbine
R	Ideal gas constant	th	Throttle
T	Temperature	w	Wastegate
u	Fraction/degree of opening	x	Turbine exit
V	Volume		
W	Mass flow rate		
$\phi(\cdot)$	Function of pressure ratio across the component		
η	Isentropic efficiency		
ψ	Mass flow parameter		
γ	Ratio of specific heats		

chosen:

$$A = \begin{bmatrix} a_{11} & 0 & 0 & a_{14} \\ a_{21} & a_{22} & 0 & 0 \\ 0 & a_{32} & a_{33} & 0 \\ 0 & 0 & a_{43} & a_{44} \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ b_3 \\ 0 \end{bmatrix}, \tag{6.5}$$

$$x = [P_b, P_i, P_e, N_t]^T, u = u_w, y = x_1.$$

The non-zero elements in (6.5) are defined as follows:

$$a_{11} = -\frac{RT_b}{V_b}\frac{W_{th}}{P_b} = -\frac{\sqrt{RT_b}}{V_b}sat(0, u_{th}, 1)\gamma\phi(\frac{P_i}{P_b}),$$

$$a_{14} = \frac{RT_b}{V_b}\frac{W_c}{N_t} = \frac{RT_b}{V_bN_t}f_c(\frac{P_b}{P_a}, N_t),$$

$$a_{21} = \frac{RT_i}{V_i}\frac{W_{th}}{P_b} = \frac{RT_i}{V_i}\frac{sat(0, u_{th}, 1)A_{max}}{\sqrt{RT_b}}\phi(\frac{P_i}{P_b})\gamma,$$

$$a_{22} = -\frac{RT_i}{V_i}\frac{W_{en}}{P_i} = -\frac{RT_i}{V_i}\frac{V_{en}N_{en}}{2RT_i},$$ (6.6)

$$a_{32} = \frac{1+A/F}{A/F}\frac{RT_e}{V_e}\frac{W_{en}}{P_i} = \frac{1+A/F}{A/F}\frac{T_e\eta_{en}V_{en}N_{en}}{2V_eT_i},$$

$$a_{33} = -\frac{RT_e}{V_e}\frac{W_t}{P_e} = -\frac{R\sqrt{T_e}}{V_e}f_t(\frac{P_e}{P_x}, \frac{N_t}{\sqrt{T_e}}),$$

$$a_{43} = \frac{H_t}{I_tN_tP_e} = \frac{1}{I_tN_t}c_{p,e}T_e\frac{W_t}{P_e}\eta_t\psi_t,$$

$$a_{44} = -\frac{H_c}{I_tN_t^2} = -\frac{1}{I_tN_t^2}c_{p,a}T_aW_c\frac{1}{\eta_c}\psi_c,$$

$$b_3 = -\frac{RT_e}{V_e}\frac{W_w}{sat(0, u_w, 1)} = -\frac{\sqrt{RT_e}}{V_e}\gamma P_e\phi(\frac{P_x}{P_e}).$$

6.3.2 Structured Quasi-LPV Inverse

Given that the parameters defined by (6.6) are varying fast during transients, treating the parameters as frozen and deriving the transfer function of (6.5) will not be effective for deriving the inverse. Indeed, by numerical simulations it is confirmed that the transfer function inverse does not represent the nonlinear model inverse. In this section, the special form of the quasi-LPV structure of (6.5) is explored to derive its inverse model in an effort to minimize the approximation error.

6.3.2.1 Quasi-LPV model inverse structure

Exploiting the sparsity of the A, B matrices of model (6.5), the quasi-LPV model is expressed as an integration of several first-order sub-models. With this very special structure of the nonlinear model, the inverse can be pursued by deriving the inverse of multiple first-order nonlinear models, which will involve limited approximation.

Given the sparse matrices A, B in the form of (6.5), the following first-order
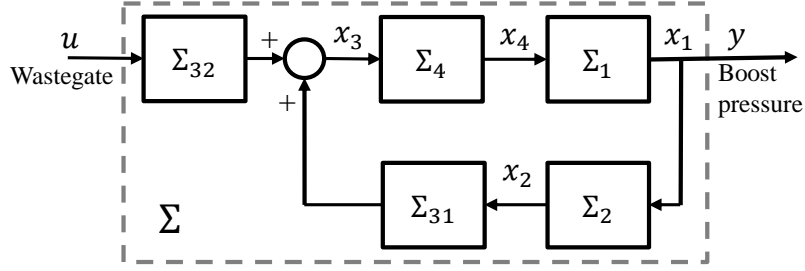
Figure 6.3: Interconnection of the first-order quasi-LPV sub-models for the fourth-order turbocharged gasoline engine LPV model.

sub-models $\Sigma_1, \Sigma_2, \Sigma_3, \Sigma_4$ are defined as

$$\Sigma_1 : \dot{x}_1 = a_{11}x_1 + a_{14}x_4 \implies x_1 = \Sigma_1(x_4),$$

$$\Sigma_2 : \dot{x}_2 = a_{22}x_2 + a_{21}x_1 \implies x_2 = \Sigma_2(x_1), \tag{6.7}$$

$$\Sigma_3 : \dot{x}_3 = a_{33}x_3 + a_{32}x_2 + b_3u \implies x_3 = \Sigma_3(x_2, u),$$

$$\Sigma_4 : \dot{x}_4 = a_{44}x_4 + a_{43}x_3 \implies x_4 = \Sigma_4(x_3).$$

Further expressing Σ_3 to be

$$\Sigma_3(x_2, u) = \Sigma_{31}(x_2) + \Sigma_{32}(u),$$

we can show that the input-output relation of the quasi-LPV model can be expressed as a composition of these sub-models:

$$y = x_1 = \Sigma_1(\Sigma_4(\Sigma_3(\Sigma_2(x_1), u)))$$

$$= \Sigma_1(\Sigma_4(\Sigma_{31}(\Sigma_2(x_1)) + \Sigma_{32}(u))), \tag{6.8}$$

whose block diagram representation is shown in Fig. 6.3.

Expressing the input u in terms of the output y based on (6.8),

$$u = \Sigma_{32}^{-1}(\Sigma_4^{-1}(\Sigma_1^{-1}(y)) - \Sigma_{31}(\Sigma_2(y))), \tag{6.9}$$

which can be viewed as an inverse model of (6.8). Fig. 6.4 shows the block diagram representation of the inverse of the quasi-LPV model through the integration of several inverse models of first-order blocks as expressed in (6.9), in which $\overline{\Sigma_1^{-1}}$, $\overline{\Sigma_4^{-1}}$, and $\overline{\Sigma_{32}^{-1}}$ are approximate inverses of Σ_1, Σ_4, and Σ_{32}. The derivations of the $\overline{\Sigma_1^{-1}}$, $\overline{\Sigma_4^{-1}}$, and $\overline{\Sigma_{32}^{-1}}$ are explained in the following section. $\overline{x_1}$-$\overline{x_4}$ in Fig. 6.4 are approximations of x_1-x_4 in Fig. 6.3.

6.3.2.2 First-order quasi-LPV inverse

We now derive the inverse of the first-order quasi-LPV model and define its property in order to derive the representation for the inverse model (6.9).

For a first-order quasi-LPV model:

$$\Sigma_i : \dot{x} = ax + bu, \tag{6.10}$$

define

$$\bar{u} = \left\{ \frac{1}{\tau s + 1} \right\} u, \tag{6.11}$$

where $\left\{ \frac{1}{\tau s+1} \right\}$ denotes the first-order filter with a transfer function $\frac{1}{\tau s+1}$. Then, the following Lemma gives the representation of $\bar{u}$.

Lemma VI.1. *Let a, b be time varying parameters with $a, b \in \mathcal{L}_\infty$, $b \neq 0$, and $b \in \mathcal{C}^1$.*
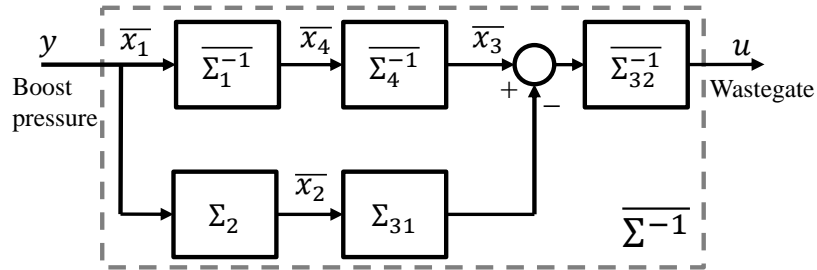

Figure 6.4: Interconnection of first-order quasi-LPV sub-models for inverse of the LPV model shown in Fig. 6.3.

Then, for any u, $\bar{u}$ given by (6.11) can be expressed in terms of the state x as:

$$\bar{u} = \frac{1}{\tau b} x - z, \tag{6.12}$$

$$\dot{z} = \frac{1}{\tau}\left(-z + \left(\frac{1}{b\tau} - \frac{\dot{b}}{b^2} + \frac{a}{b}\right)x\right),$$

where x is given by (6.10).

Proof: Note that

$$\bar{u} = \left\{\frac{1}{\tau s + 1}\right\} u = \left\{\frac{1}{\tau s + 1}\right\}\left(\frac{\dot{x}}{b} - \frac{ax}{b}\right). \tag{6.13}$$

Since

$$\frac{d}{dt}\left(\frac{x}{b}\right) = \frac{\dot{x}}{b} - \frac{\dot{b}x}{b^2}, \tag{6.14}$$

we have, from (6.13), that

$$\begin{aligned}
\bar{u} &= \left\{\frac{1}{\tau s + 1}\right\}\left(\frac{d}{dt}\left(\frac{x}{b}\right) + \frac{\dot{b}x}{b^2} - \frac{ax}{b}\right) \\
&= \left\{\frac{1}{\tau s + 1}\right\}\{\tau s + 1\}\left(\frac{x}{\tau b}\right) \tag{6.15} \\
&\quad - \left\{\frac{1}{\tau s + 1}\right\}\left[\left(\frac{1}{b\tau} - \frac{\dot{b}}{b^2} + \frac{a}{b}\right)x\right].
\end{aligned}$$

Note that the time invariant operator $\{\tau s + 1\}$ can now be cancelled with $\left\{\frac{1}{\tau s+1}\right\}$ since there is no time varying signal in between. Let

$$z = \left\{\frac{1}{\tau s + 1}\right\}\left[\left(\frac{1}{b\tau} - \frac{\dot{b}}{b^2} + \frac{a}{b}\right)x\right], \tag{6.16}$$

then the dynamics from output x to $\bar{u}$ can be represented by a first-order LPV model

$$\bar{u} = \frac{1}{\tau b}x - z, \tag{6.17}$$

$$\dot{z} = \frac{1}{\tau}\left(-z + \left(\frac{1}{b\tau} - \frac{\dot{b}}{b^2} + \frac{a}{b}\right)x\right).$$

Remark VI.1. Treat x as the input, $\bar{u}$ as the output, and a, b, and τ as the parameters, the BIBO stability of the first-order system (6.12) can be easily established given that the system has a single frozen-time pole at $-\frac{1}{\tau}$, and $\frac{1}{\tau b}$, $\frac{1}{\tau}(\frac{1}{b\tau} - \frac{\dot{b}}{b^2} + \frac{a}{b})$ are bounded.

Remark VI.2. Note that since $\bar{u} \approx u$ for small τ, one can treat (6.12) as an approximate inverse model of (6.10). Moreover, $|\bar{u} - u| \propto \mathcal{O}(\tau)$, namely, the inverse model error can be made arbitrarily small with a properly chosen τ.

Remark VI.3. Lemma VI.1 assumes a continuous-time implementation of the inverse of (6.10). When (6.12) is discretized for real engine implementation, its BIBO stability remains due to Remark VI.1. If the delay caused by discretization is small, its performance will not be substantially affected.

Now the approximate inverse model $\overline{\Sigma_i^{-1}}$ is given by (6.12). For simplicity, one can drop the $\dot{b}/b^2$ term if the parameter variation is substantially slower than the system dynamics. However, this is not the case in this application. $\dot{b}/b^2$ as in (6.6) includes the states. Therefore, $\dot{b}/b^2$ is not slower than the system dynamics. The following simulation also verified that $\dot{b}/b^2$ should not be omitted: To validate the first-order sub-model inverse, the two systems Σ_i and $\overline{\Sigma_i^{-1}}$ are connected in cascade as shown in Fig. 6.5. According to Remark VI.2, the output v in Fig. 6.5 should be close to the input $\bar{v}$. A numerical analysis of the inverse performance is shown in Fig. 6.6. It is obvious that the inverse incorporating the $\dot{b}/b^2$ term with smaller time constant τ has better accuracy. Therefore, the inverse incorporating $\dot{b}/b^2$ is adopted in the subsequent derivation. The time constant τ is the tuning parameter for the IMC design. Fig. 6.6 indicates that smaller time constants lead to a better inverse,
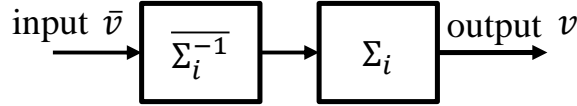
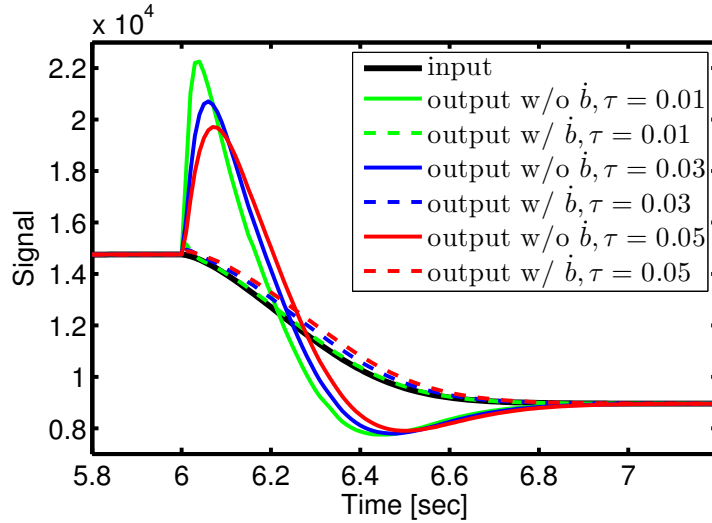Figure 6.5: Structure for validation of first-order inverse.

Figure 6.6: Analysis of first-order inverse $\overline{\Sigma_1^{-1}}$ with and without $\dot{b}$ ($\dot{b}$ is derived from numerically differentiating b).

as expected.

6.3.2.3 Structured quasi-LPV inverse

Representing each first-order model inverse in (6.9) with (6.12), an inverse model for the nonlinear model is derived, which will be referred to as the structured quasi-LPV inverse.

To incorporate the quasi-LPV inverse model in the IMC structure, two implementable configurations are possible as shown in Fig. 6.7(a) and Fig. 6.7(b). First, one can use the states in the nonlinear model Σ to schedule the parameters in the inverse model $\overline{\Sigma^{-1}}$ (as in Fig. 6.7(a)). Since the states used for parameter scheduling are external to $\overline{\Sigma^{-1}}$, Fig. 6.7(a) is referred to as the externally scheduled quasi-LPV inverse. Secondly, since all the states in the original quasi-LPV model are explicit in its inverse structure, one can derive parameters used in A and B (defined in (6.5))
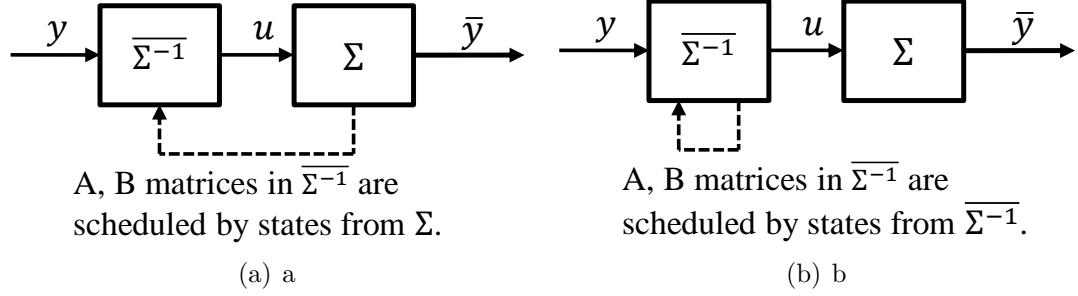
Figure 6.7: Validation structures: (a) Externally scheduled quasi-LPV inverse validation structure. (b) Internally scheduled quasi-LPV inverse validation structure.

for $\overline{\Sigma^{-1}}$ based on the internal states in $\overline{\Sigma^{-1}}$ (as in Fig. 6.7(b)), which is referred to as the internally scheduled quasi-LPV inverse.

While the externally scheduled quasi-LPV inverse looks appealing at first, its utility is ruled out after more in-depth analysis and simulation. The dashed line in Fig. 6.7(a) which represents the gain scheduling signal forms a feedback loop, which causes instability. In this work, the proposed IMC controller uses the internally scheduled quasi-LPV inverse model for its implementation. It should be noted that the internally scheduled quasi-LPV inverse is not possible for inverse LPV model derived by general system inverting methodologies, unless the states are preserved in the inverse model. With the structured quasi-LPV inverse model shown in Fig. 6.4, it is true that all states are explicitly preserved in the inverse model, thereby making the internally scheduled quasi-LPV inverse implementation possible.

6.3.2.4 Stability of the structured quasi-LPV inverse

Even though each subsystem in the structured quasi-LPV model is BIBO stable, the overall inverse $\overline{\Sigma^{-1}}$, with internally scheduled parameters, still has stability issues due to the feedback loop introduced by those state-dependent parameters.

Since the state-dependent parameters are the root cause for the instability phenomenon in the structured quasi-LPV inverse model, an in-depth analysis of the pa-
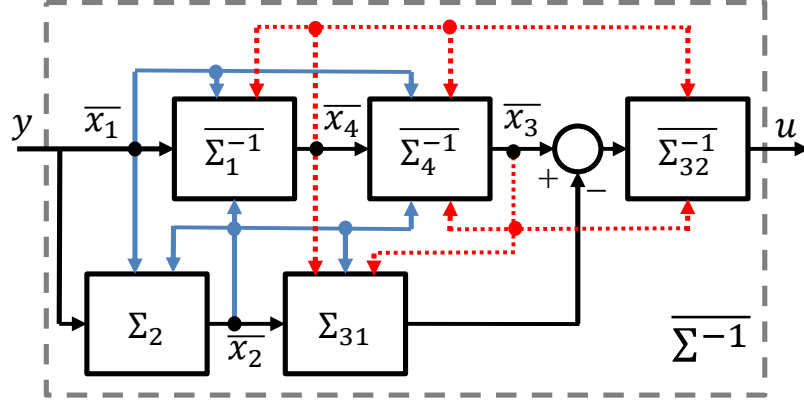
Figure 6.8: Parameter scheduling relationship in the internally scheduled quasi-LPV inverse (Blue solid lines indicate that the states are actually used for scheduling. Red dotted lines represent the use of the steady-state value generated from steady-state mapping in scheduling.).

rameter scheduling is carried out in the inverse model. Fig. 6.8 shows the scheduling of the varying parameters in each sub-model for the internally scheduled quasi-LPV inverse. One can see that some of the interconnections, shown by the blue solid lines in Fig. 6.8 from the states $\overline{x_1}$ and $\overline{x_2}$, do not introduce additional feedback loops except those within their own sub-models. Others, that are shown by red dotted line for the parameters scheduled based on $\overline{x_3}$ and $\overline{x_4}$, form additional feedback loops within the structured quasi-LPV inverse. Meanwhile, the errors from $\overline{\Sigma_1^{-1}}$, $\overline{\Sigma_4^{-1}}$, and $\overline{\Sigma_{32}^{-1}}$ propagate within the structure, leading to complicated dynamic responses. To construct a stable inverse, the scheduling signals $\overline{x_3}$ (approximate exhaust pressure) and $\overline{x_4}$ (approximate turbo speed) are replaced by their steady-state values. The steady-state maps are generated with respect to different engine speed and throttle opening.

Remark VI.4. Note that, based on the dual stability property of IMC [9], the stability of the closed-loop system can be assured if P and Q are stable and $M = P$. The stability of the quasi-LPV inverse Q can be established given that it is made up by stable sub-models Σ_is and $\overline{\Sigma_i^{-1}}$s through feedforward connections. All feedback loops

in Q are eliminated after replacing the gain-scheduling elements that forms feedback loops using their steady-state values. Therefore, the stability of the closed-loop system with IMC can be assured using $Q = \overline{\Sigma^{-1}}$ given in Fig. 6.8.

6.3.2.5 Structured quasi-LPV inverse model validation

The inverse model shown in Fig. 6.8, with some of the state-dependent parameters replaced by steady-state mapping values, is validated through simulation. The validation is performed by connecting the inverse to the original model as in Fig. 6.7(b) and the validated results are shown in Fig. 6.9.

Note that the quality of the inverse model depends on the tuning parameters, which are the time constants τ as in each first-order LPV sub-model inverse $\overline{\Sigma_1^{-1}}$, $\overline{\Sigma_4^{-1}}$, and $\overline{\Sigma_{32}^{-1}}$ as shown in Fig. 6.4. Theoretically if the time constants are small, the output has faster responses but also potential oscillations during transients. If the time constants are large, the transient response will be slow. Two results with different tuning parameters are shown in Fig. 6.9, which validates the inverse model. Inverse 1 has the time constants 0.1s, 0.04s, and 0.02s in $\overline{\Sigma_1^{-1}}$, $\overline{\Sigma_4^{-1}}$, and $\overline{\Sigma_{32}^{-1}}$, respectively. Inverse 2 has the time constants 0.05s, 0.04s, and 0.02s in $\overline{\Sigma_1^{-1}}$, $\overline{\Sigma_4^{-1}}$, and $\overline{\Sigma_{32}^{-1}}$, respectively. Inverse 1, which has the larger time constant, is more damped than inverse 2 and has little overshoot, which matches the intuition. The time constant for $\overline{\Sigma_1^{-1}}$ is chosen to be larger than the others, considering the error in its output $\overline{x_4}$ propagates to $\overline{x_3}$ and u, as shown in Fig. 6.8.

6.4 Application of the Nonlinear IMC on the Boost-Pressure Control Problem of a Turbocharged Gasoline Engine

With the fourth-order nonlinear model and the model inverse developed, IMC can be designed by applying the nonlinear model and its inverse into Fig. 1.3. The IMC is
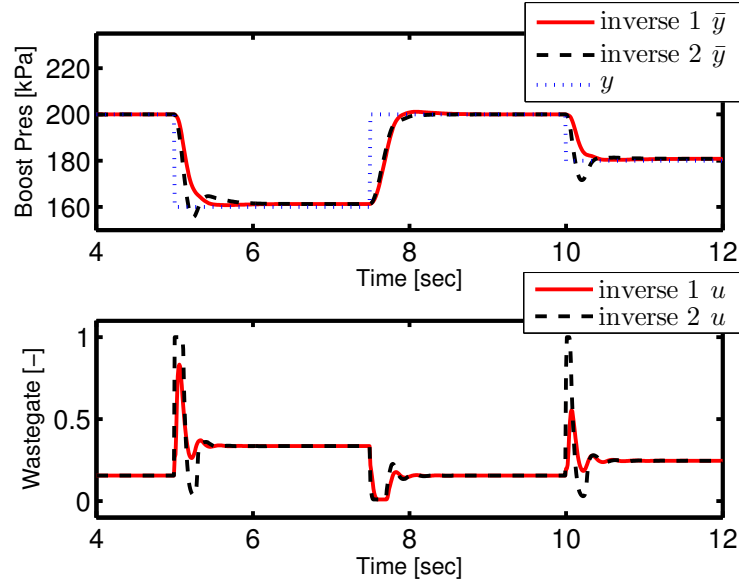
Figure 6.9: Validation of structured quasi-LPV inverse.

implemented in continuous-time domain, and is applied to the virtual "plant", which is a validated Ford proprietary model. The structure of the resulting nonlinear IMC system is shown in Fig. 6.10. The tuning parameters are the time constants in each first-order LPV sub-model inverse $\overline{\Sigma_1^{-1}}$, $\overline{\Sigma_4^{-1}}$, and $\overline{\Sigma_{32}^{-1}}$, as shown in Fig. 6.4. The time constants are chosen at 0.1s, 0.04s, and 0.02s, which are the same as in inverse 1. The pressure and temperature sensors are assumed to be accurate. Performance of the resulting control system is evaluated in this section together with the robustness analysis with respect to different operating conditions and measurement noises.

6.4.1 Performance Evaluation

To evaluate the performance of IMC, some features have to be considered [87]:

- The overshoot of P_b has to be minimized to avoid throttle re-closing.

- The pressure oscillation of P_b while tracking a step function is undesirable because they could generate torque oscillations that are noticeable to the driver.

IMC is compared with a well-tuned PI controller with extensive feedforward and
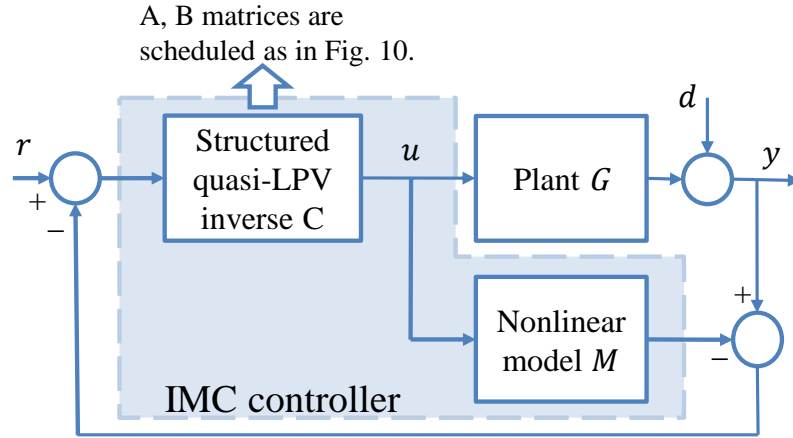
95

Figure 6.10: IMC structure with structured quasi-LPV inverse.

anti-windup built in, which is referred to as PI control in the context. The system response and control input are compared in two cases: constant engine speed of 3000 rpm (as in Fig. 6.11(a)); varying engine speed(as in Fig. 6.11(b)). Here varying engine speed rises from 1500 to 3000 rpm gradually. The throttle opening is 45 degrees in both simulations. In Fig. 6.11(a), it can be observed that IMC achieved a faster reference tracking than PI with less overshoot or oscillation. In Fig. 6.11(b), the IMC response does not overshoot, even without incorporating an explicit anti-windup strategy.

Overall, the nonlinear IMC for the wastegate control of a turbocharged gasoline engine shows promising performance. It shows good reference tracking, no steady-state error, no need for a separate anti-windup design, and intuitive tuning. Its performance matches, and in cases exceeds, that of a well-tuned PI control with extensive feedforward and anti-windup built in.

6.4.2 Performance in the Presence of Disturbances

In real applications of boost-pressure control of a turbocharged gasoline engine, the reference and operating points vary. More analysis is performed herein to evaluate the system performance sensitivity with respect to the variation of operating conditions
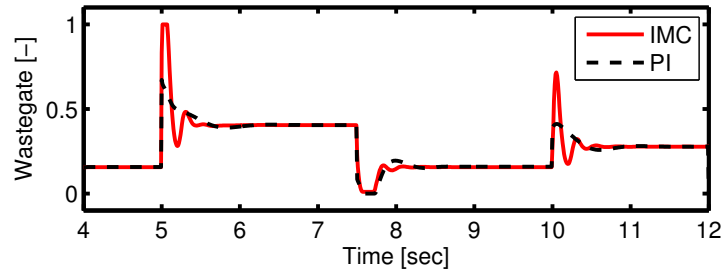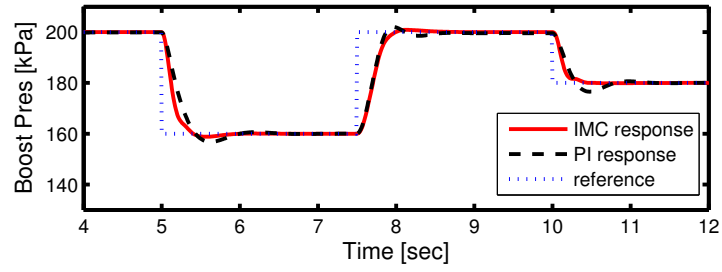
(a) a
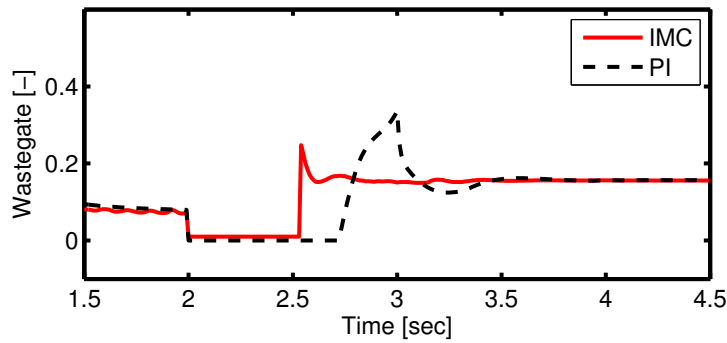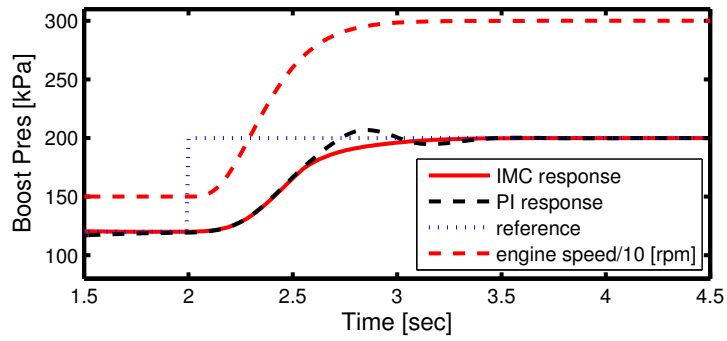
(b) b

Figure 6.11: Simulation results: (a) IMC performance: constant engine speed. (b) IMC performance: varying engine speed.

(engine speed and throttle opening). Note that only one set of tuning parameters is used for all the tests.

First, the impact of the engine speed N_{en} is considered. Two sets of tests are performed: the same P_b reference step at different engine speed N_{en} (as in Fig. 6.12(a)), constant P_b reference with a step change in N_{en} (as in Fig. 6.12(b)), in which case the variation in N_{en} can be viewed as a disturbance. The step change in N_{en} is from 2500 to 3000 rpm. The throttle opening is 45 degrees in all simulations. The results show that IMC performs well at all engine speeds, and it rejects the disturbance in N_{en}.

Next, the impact of the throttle opening u_{th} is considered. Two sets of tests are performed: the same P_b reference step at different throttle opening u_{th} (as in Fig. 6.13(a)), constant P_b reference with a step change in u_{th} (as in Fig. 6.13(b)), in which case the variation in u_{th} can be viewed as a disturbance. The step change in u_{th} is from 45 to 30 degrees. The engine speed is 3000rpm in all simulations. The results show that IMC performs well at all throttle openings, and it rejects the disturbance in u_{th}.
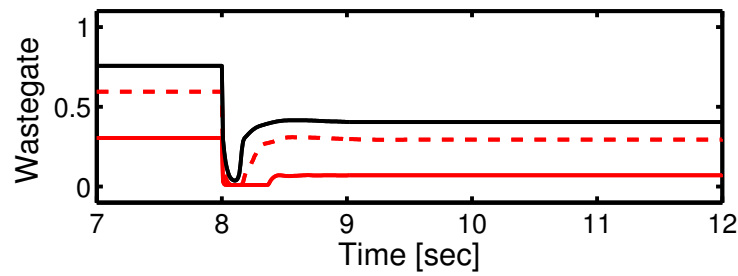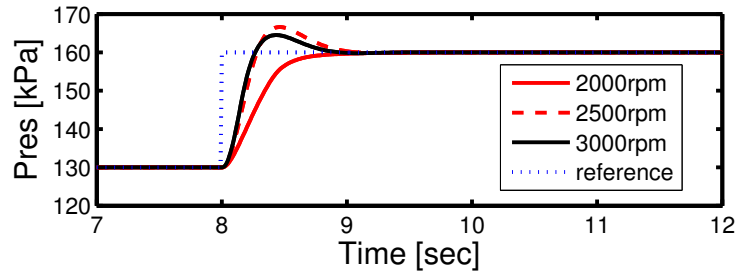
(a) a
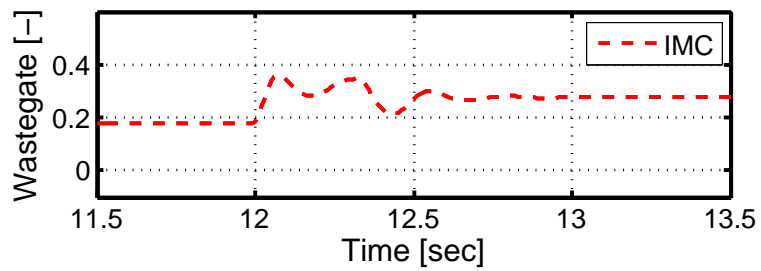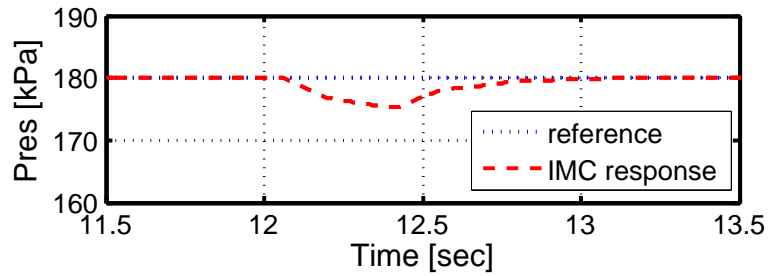
(b) b

Figure 6.12: IMC robustness evaluation with respect to engine speed: (a) IMC robustness: different engine speed. (b) IMC robustness: varying engine speed, constant reference.

(a) a

(b) b

Figure 6.13: IMC robustness evaluation with respect to throttle opening: (a) IMC robustness: different throttle opening. (b) IMC robustness: step change in throttle opening, constant reference.
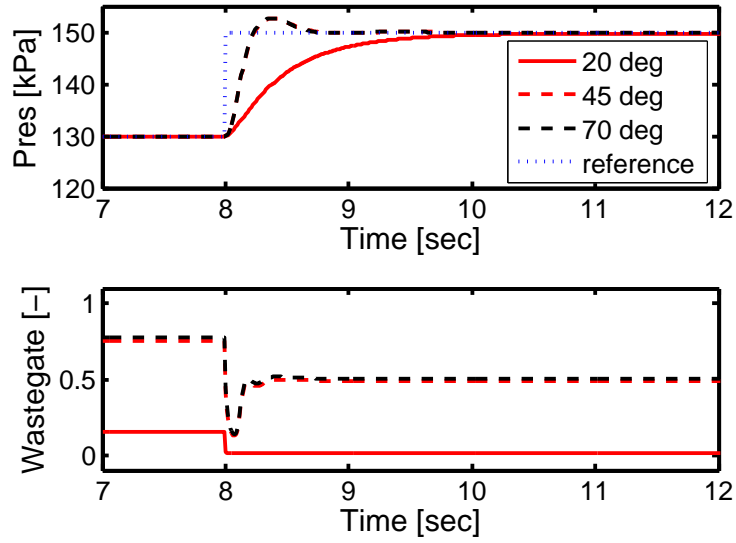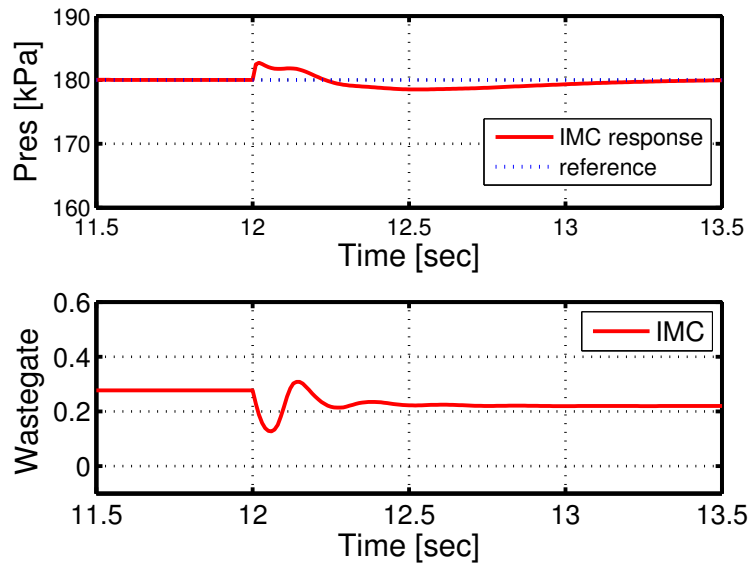
CHAPTER VII

Conclusions

7.1 Conclusions

In this thesis, motivated by the need for a control design approach to reduce the controller design and calibration effort for the automotive industry, we exploited the internal model control (IMC) framework, which offers an intuitive control structure and simple tuning philosophy. Two directions are pursued: adaptive IMC (AIMC) and nonlinear IMC.

By combining IMC and parameter identification techniques, we developed composite AIMC (CAIMC), which simultaneously identifies the plant model and the plant inverse online. Through the simultaneous online identification of the model and the inverse, CAIMC minimizes the forward and inverse modeling errors, and further reduces the tracking error. CAIMC is first developed for a first-order plant, and then extended to an n-th order plant. The design procedure, stability proof, and asymptotic performance are presented. The general CAIMC design procedure follows very closely to the first-order CAIMC, but the stability requirement poses non-trivial constraints for parameter identification. A convex programming problem with a linear matrix inequality constraint is formulated to handle the stability constraint.

For the problem of boost-pressure control of a turbocharged gasoline engine, CAIMC is first validated on a proprietary model, and then validated on a vehicle

with a $2.0L$ four-cylinder turbocharged gasoline engine. Both simulations and experiments show that CAIMC not only improves tracking performance, but also drastically reduces the calibration effort compared to the traditional PI controller with feedforward.

A nonlinear IMC design is presented in the context of the boost-pressure control. A nonlinear fourth-order dynamics model is adopted in the controller. The challenges for inverting the nonlinear model are addressed by: (1) representing the nonlinear dynamics with a quasi-LPV model, (2) exploring the special quasi-LPV model structure, (3) using inverse of the simple first-order quasi-LPV model, and (4) assuring the stability of the inverse model by eliminating the internal loops. It is shown to be effective and robust for the boost-pressure control, but it is hard to generalize due to its special structural requirement for the nonlinear model.

7.2 Future Research Directions

Two directions are presented in this thesis: adaptive IMC and nonlinear IMC. While contributions are made in both directions, they also lead to new open challenges:

7.2.1 Adaptive IMC

- **Generalization of CAIMC:** First, CAIMC for a SISO plant can be generalized to be applicable to a MIMO plant. Second, CAIMC for a linear plant can be generalized to CAIMC with nonlinear plant. As discussed in Chapter IV, there are almost always unmodeled dynamics in CAIMC from representing the inverse dynamics. When approximating a nonlinear plant with a linear plant, more unmodeled dynamics are introduced. Therefore, robust adaptive control techniques have to be adopted in CAIMC. Because transient performance and robustness are trade-offs in adaptive control framework, it may be desirable to
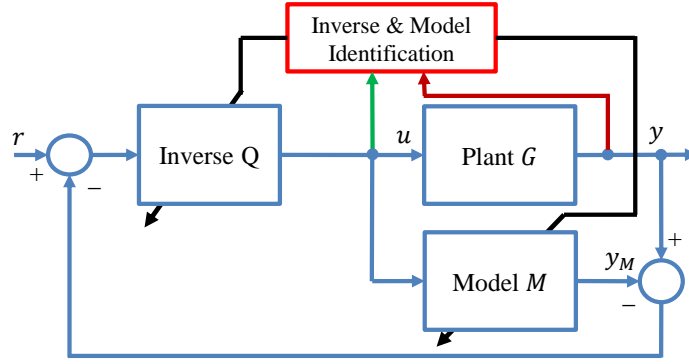
Figure 7.1: IMC with coordinative model and inverse identifications.

adopt nonlinear models to identify the plant and its inverse to achieve better transient performance.

- **IMC with coordinative model and inverse identifications** In terms of the inverse design philosophy, AIMC and CAIMC are the two extremes. As shown in Fig. 2.1, AIMC identifies the model, and the inverse is derived from inverting the model. As shown in Fig. 3.1, CAIMC identifies the model and inverse in parallel. At the expense of the extra identification of the inverse, CAIMC reduces the tracking error when compared with AIMC. However, the inverse identification is independent of the model identification, while their dynamics are closely related. Coordinating the simultaneous model and inverse identification as shown in Fig. 7.1 is the middle ground between AIMC and CAIMC, which has not been explored yet. It has a great potential in terms of further improving the control performance or reducing the computational complexity.

7.2.2 Nonlinear IMC

- **Generalization of the nonlinear IMC** In this thesis, nonlinear IMC is presented in the context of the boost-pressure control of a turbocharged gasoline engine. It is applicable when the nonlinear system has a special structural

property and has not been generalized yet. Future work in this direction includes expanding this specific nonlinear IMC design into a more general methodology, and developing a systematic approach for calibration.

APPENDICES

APPENDIX A

Preliminaries for Convex Programming Problem Analysis

Relevant results for a convex optimization problem are introduced as following. Consider the optimization problem

$$\underset{\chi}{\text{minimize}} \; f(\chi), \text{subject to } \chi \in \Phi. \tag{A.1}$$

$\chi \in \mathcal{R}^n$ is the optimization variable, and the function $f : \mathcal{R}^n \to \mathcal{R}$ is the objective or cost function. The optimal value of the cost function is defined as $v^* = \inf\{f(\chi)|\chi \in \Phi\}$. χ^* is an optimal solution, if $\chi^* \in \Phi$ and $f(\chi^*) = v^*$.

Lemma A.1. *[88] For (A.1), let Φ be a nonempty closed convex set and f be a strictly convex function over Φ, then the optimal solution χ^* is unique.*

Consider the parameterized optimization problem

$$\underset{\chi}{\text{minimize}} \; f(\chi, p), \text{subject to } \chi \in \Phi, \tag{A.2}$$

where the cost function $f(\chi, p)$ depends on the parameter vector $p \in \Pi \subset \mathcal{R}^m$, and

the feasible region Φ is independent of p. Let $v^*(p) : \mathcal{R}^m \to \mathcal{R}$ represent the optimal cost value function.

Lemma A.2. *[89] For (A.2), the optimal cost value function $v^*(p)$ is upper semi-continuous, i.e., $\lim_{p \to p_0} v^*(p) \leq v^*(p_0)$.*

Assume that the optimal solution is unique and let $\chi^*(p) : \mathcal{R}^m \to \mathcal{R}^n$ represent the optimal solution function.

Lemma A.3. *[89] For (A.2), suppose that*

(i) The second order growth condition holds for $f(\chi, p)$ at $\chi^*(p_0)$, i.e. there exists a neighbourhood N of $\chi^*(p_0)$ and a constant $c > 0$ such that $f(\chi) \geq f(\chi^*) + c(\chi - \chi^*)^2, \forall \chi \in \Phi \cap N$.

(ii) The difference function $f(\chi, p) - f(\chi, p_0)$ is Lipschitz continuous with respect to χ modulus κ on $\Phi \cap N$, i.e. $\exists \kappa < \infty, \|(f(\chi_1, p) - f(\chi_1, p_0)) - (f(\chi_2, p) - f(\chi_2, p_0))\| \leq \kappa\|\chi_1 - \chi_2\|, \forall \chi_1, \chi_2 \in \Phi \cap N$.

Then $\|\chi^*(p) - \chi^*(p_0)\| \leq c^{-1}\kappa$.

APPENDIX B

Mathematical Tools for Stability Proof

The stability proof of CAIMC is done by representing the closed-loop system as a linear time-varying (LTV) system. Relevant results are introduced here to establish the exponential stability and signal boundedness of linear systems.

Lemma B.1. *For a LTV system*

$$\dot{x} = A(t)x,$$

where $x \in \mathcal{R}^n$, and the elements of $A(t)$ are piecewise differentiable and bounded. Assume that $Re\{\lambda_i(A(t))\} \leq -\delta_s \ \forall t \geq 0$ and for $i = 1, 2, \ldots, n$ where $\delta_s > 0$ is some constant. $||A(t)|| \leq c$, for some constant $c > 0$, $\forall t \geq 0$, where $||A(t)||$ is the induced norm. If one of the following statements holds:

- $||\dot{A}(t)|| \leq c$, *for some $c > 0$, where $||\dot{A}(t)||$ is the induced norm.*

- $\exists k_0 > 0, \delta_0 > 0, \sup_{0 \leq \tau \leq k_0} ||A(t + \tau) - A(t)|| \leq \delta_0.$

Then the equilibrium state $x_e = 0$ is exponentially stable, i.e., the state transition matrix

$$||\Phi(t, \tau)|| \leq \lambda_0 e^{-\alpha_0(t-\tau)}, \forall t \geq \tau \geq 0$$

for some $\lambda_0, \alpha_0 > 0$ [90].

Lemma B.2. *Consider the LTV system given by*

$$\dot{x} = A(t)x + B(t)u,$$

where $x \in \mathcal{R}^n, y \in \mathcal{R}^r, u \in \mathcal{R}^m$, and the elements of the matrices $A(t), B(t)$ are bounded piecewise continuous functions of time. If the state transition matrix $||\Phi(t, \tau)|| \leq \lambda_0 e^{-\alpha_0(t-\tau)}$ for some $\lambda_0, \alpha_0 > 0$ and $u \in \mathcal{L}_{2e}$, i.e. $||u_t||_2 := (\int_0^t |u(\tau)|^2 d\tau)^{\frac{1}{2}}$ exists for any finite t. Then for any $\delta \in [0, \delta_1)$ where $0 < \delta_1 < 2\alpha_0$ is arbitrary, we have

- $|x(t)| \leq \frac{c\lambda_0}{\sqrt{2\alpha_0 - \delta}} ||u_t||_{2\delta} + \epsilon_t$

- $||x_t||_{2\delta} \leq \frac{c\lambda_0}{\sqrt{(\delta_1 - \delta)(2\alpha_0 - \delta_1)}} ||u_t||_{2\delta} + \epsilon_t$

where $c = \sup_t ||B(t)||$, and ϵ_t is an exponentially decaying to zero term due to the initial condition [78].

Lemma B.3. *Consider a LTI system given by*

$$y = \{H(s)\}u,$$

where $H(s)$ is strictly proper and analytic in $Re(s) \geq -\frac{\delta}{2}$ for some $\delta > 0$ and $u \in \mathcal{L}_{2e}$.

Then we have

$$|y(t)| \leq c||u_t||_{2\delta}$$

for some c [78].

Lemma B.4. *Bellman-Gronwall (B-G) Lemma: [78]*

Let $\lambda(t), g(t), k(t)$ be nonnegative piecewise continuous functions of time t. If a

function $f(t)$ satisfies the inequality

$$f(t) \le g(t) \int_{t_0}^{t} k(s)f(s)ds + \lambda(t), \forall t \ge t_0 \ge 0,$$

then

$$f(t) \le g(t) \int_{t_0}^{t} \lambda(s)k(s)[exp(\int_{s}^{t} k(\tau)g(\tau)d\tau)]ds + \lambda(t), \forall t \ge t_0 \ge 0.$$

In particular, if $\lambda(t) \equiv \lambda$ is a constant and $g(t) \equiv 1$, then

$$f(t) \le \lambda exp(\int_{t_0}^{t} k(s)ds), \forall t \ge t_0 \ge 0.$$

APPENDIX C

Derivation of the Closed-loop State-space Equation of CAIMC

Combining (4.4), (4.6), and (4.25), we have

$$\hat{\epsilon}_M m_M^2 = z_M - \hat{\theta}_M^T \phi_M$$

$$= \{s^n\} y_f + \theta_\lambda^T \left\{ \begin{bmatrix} 1 \\ s \\ \vdots \\ s^{n-1} \end{bmatrix} \right\} y_f - \hat{\theta}_a^T \left\{ \begin{bmatrix} 1 \\ s \\ \vdots \\ s^{n-1} \end{bmatrix} \right\} y_f - \hat{\theta}_b^T \left\{ \begin{bmatrix} 1 \\ s \\ \vdots \\ s^{n-1} \end{bmatrix} \right\} u_f.$$

Therefore,

$$y_f^{(n)} = (\hat{\theta}_a - \theta_\lambda)^T \left\{ \begin{bmatrix} 1 \\ s \\ \vdots \\ s^{n-1} \end{bmatrix} \right\} y_f + \hat{\theta}_b^T \left\{ \begin{bmatrix} 1 \\ s \\ \vdots \\ s^{n-1} \end{bmatrix} \right\} u_f + \hat{\epsilon}_M m_M^2. \tag{C.1}$$

From (4.16), let $\bar{r} = \{\hat{Z}_Q(s)\frac{1}{\Lambda}\}r$ and $\hat{\theta}_c = \begin{bmatrix} \hat{\theta}_{cn} \\ \eta \end{bmatrix}$, where $\hat{\theta}_{cn} \in \mathcal{R}^n$ and η is the $(n+1)$-th entry of $\hat{\theta}_c$. Then,

$$\{\hat{R}_Q(s)\}u_f = \bar{r} - \left\{\hat{Z}_Q(s)\frac{1}{\Lambda}\right\}\hat{\epsilon}_M m_M^2 = \bar{r} - \{\hat{Z}_Q(s) - \eta\Lambda\}\epsilon_f + \eta\hat{\epsilon}_M m_M^2,$$

and

$$\{s^n\}u_f + (\theta_\lambda - \hat{\theta}_d)^T \left\{\begin{bmatrix} 1 \\ s \\ \vdots \\ s^{n-1} \end{bmatrix}\right\} u_f = \bar{r} - (\hat{\theta}_{cn} - \eta\theta_\lambda)^T \left\{\begin{bmatrix} 1 \\ s \\ \vdots \\ s^{n-1} \end{bmatrix}\right\} \epsilon_f + \eta\hat{\epsilon}_M m_M^2,$$

$$u_f^{(n)} = (\hat{\theta}_d - \theta_\lambda)^T \left\{\begin{bmatrix} 1 \\ s \\ \vdots \\ s^{n-1} \end{bmatrix}\right\} u_f - (\hat{\theta}_{cn} - \eta\theta_\lambda)^T \left\{\begin{bmatrix} 1 \\ s \\ \vdots \\ s^{n-1} \end{bmatrix}\right\} \epsilon_f + \eta\hat{\epsilon}_M m_M^2 + \bar{r}.$$

$$\tag{C.2}$$

From (4.25),

$$\epsilon_f^{(n)} = -\theta_\lambda^T \left\{\begin{bmatrix} 1 \\ s \\ \vdots \\ s^{n-1} \end{bmatrix}\right\} \epsilon_f + \hat{\epsilon}_M m_M^2. \tag{C.3}$$

Combining (C.1), (C.2), and (C.3), we have

$$\dot{x} = A(t)x + b_1(t)\hat{\epsilon}_M m_M^2 + b_2\bar{r},$$

$$\begin{bmatrix} y \\ u \end{bmatrix} = C(t)x + d_1(t)\hat{\epsilon}_M m_M^2 + d_2\bar{r},$$

where

$$A(t) = \left[\begin{array}{c|c|c} 0_{(n-1)\times1}|I_{n-1} & 0_{(n-1)\times n} & 0_{(n-1)\times n} \\ (\hat{\theta}_a - \theta_\lambda)^T & \hat{\theta}_b^T & 0_{1\times n} \\ \hline 0_{(n-1)\times n} & 0_{(n-1)\times1}|I_{n-1} & 0_{(n-1)\times n} \\ 0_{1\times n} & (\hat{\theta}_d - \theta_\lambda)^T & -(\hat{\theta}_{cn} - \eta\theta_\lambda)^T \\ \hline 0_{(n-1)\times n} & 0_{(n-1)\times n} & 0_{(n-1)\times1}|I_{n-1} \\ 0_{1\times n} & 0_{1\times n} & -\theta_\lambda^T \end{array}\right],$$

$$b_1(t) = \begin{bmatrix} 0_{(n-1)\times1} \\ 1 \\ \hline 0_{(n-1)\times1} \\ \eta \\ \hline 0_{(n-1)\times1} \\ 1 \end{bmatrix}, b_2(t) = \begin{bmatrix} 0_{n\times1} \\ \hline 0_{(n-1)\times1} \\ 1 \\ \hline 0_{n\times1} \end{bmatrix},$$

$$C(t) = \begin{bmatrix} \hat{\theta}_a^T & \hat{\theta}_b^T & 0_{(1\times n)} \\ 0_{(1\times n)} & \hat{\theta}_d^T & -(\hat{\theta}_{cn} - \eta)^T \end{bmatrix},$$

$$d_1(t) = \begin{bmatrix} 1 \\ \eta \end{bmatrix}, d_2(t) = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] "Light-Duty Automotive Technology, Carbon Dioxide Emissions, and Fuel Economy Trends: 1975 Through 2016," *United States Environmental Protection Agency Report*, 2016.

[2] H. Xie, R. Stobart, P. Tunestal, L. Eriksson, Y. Huang, and P. Leteinturier, "Future Engine Control Enabling Environment Friendly Vehicle," *SAE Technical Paper*, 2011-01-0697, 2011.

[3] *Measurement, ECU Calibration, and Diagnostics - Development Solutions for Automotive Embedded Systems*, ETAS Group, 2010.

[4] A. Martyr and M. A. Plint, *Engine Testing, Fourth Edition: The Design, Building, Modification and Use of Powertrain Test Facilities*, Butterworth-Heinemann, 2012.

[5] T. Samad and A. M. Annaswamy, *The Impact of Control Technology, 2nd*, IEEE Control Systems Society, 2014.

[6] U. Christen and R. Busch, "The Art of Control Engineering: Science Meets Industrial Reality," *2012 IFAC Workshop on Engine and Powertrain Control, Simulation and Modeling*, 2012.

[7] A. Dauron, "Model-Based Powertrain Control: Many Uses, No Abuse," *Oil & Gas Science and Technology*, vol.62, no.4, pp.427-435, 2007.

[8] M. Morari and E. Zafiriou, *Robust Process Control*, Prentice Hall, 1989.

[9] C. E. Garcia and M. Morari, "Internal Model Control. A Unifying Review and Some New Results," *Ind. Eng. Chem. Process Des. Dev.*, vol.21, no.2, pp.308-323, 1982.

[10] I. G. Horn, J. R. Arulandu, C. J. Gombas, J. G. VanAntwerp, and R. D. Braatz, "Improved Filter Design in Internal Model Control," *Ind. Eng. Chem. Res*, vol.35, no.10, pp.3437-3441, 1996.

[11] M. Campi, W. S. Lee, and B. D. O. Anderson, "New Filters For Internal Model Control Design," *International Journal of Robust and Nonlinear Control*, vol.4, pp.757-775, 1994.

[12] S. Hong, I. Park, and M. Sunwoo, "Model-Based Gain Scheduling Strategy for an Internal Model Control-Based Boost-Pressure Controller in Variable Geometric Turbocharger System of Diesel Engines," *Journal of Dynamic Systems, Measurement, and Control*, vol.138, no.3, 2016.

[13] D. Schwarzmann, R. Nitsche, and J. Lunze, "Diesel Boost-Pressure Control Using Flatness-Based Internal Model Control," *SAE Technical Paper*, 2006-01-0855, 2006.

[14] R. Nitsche, D. Schwarzmann, and J. Hanschke, "Nonlinear Internal Model Control of Diesel Air Systems," *Oil & Gas Science and Technology - Rev. IFP*, vol.62, no.4, pp.501-512, 2007.

[15] D. Schwarzmann, R. Nitsche, J. Lunze, and A. Schanz, "Pressure Control of a Two-stage Turbocharged Diesel Engine Using a Novel Nonlinear IMC Approach," *2006 IEEE International Conference on Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control*, pp.2399-2404, 2006.

[16] A. Thomasson, L. Eriksson, O. Leufven, and P. Andersson, "Wastegate Actuator Modeling and Model-based Boost-Pressure Control," *In IFAC Workshop on Engine and Powertrain Control, Simulation and Modeling*, vol.41, 2009.

[17] A. Karnik and M. Jankovic, "IMC based Wastegate Control Using a First-order Model for Turbocharged Gasoline Engine," *American Control Conference*, pp.2872-2877, 2012.

[18] D. Rupp and L. Guzzella, "Iterative Tuning of Internal Model Controllers With Application to Air/Fuel Ratio Control," *IEEE Transactions on Control System Technology*, vol.18, no.1, pp.177-184, 2010.

[19] N. E. Kahveci, S. T. Impram, and A. Umut Genc, "Adaptive Internal Model Control for Air-Fuel Ratio Regulation," *2014 IEEE Intelligent Vehicles Symposium*, pp.1091-1096, 2014.

[20] E. Alfieri, A. Amstutz, and L. Guzzella, "Gain-scheduled Model-based Feedback Control of the Air/Fuel Ratio in Diesel Engines," *Control Engineering Practice*, vol.17, no.12, pp.1417-1425, 2009.

[21] E. Gruenbacher and L. Marconi, "Idle Mode Control on a Combustion Engine Test Bench via Internal Model Control," *American Control Conference*, pp.2045-2050, 2009.

[22] A. Thomasson and L. Eriksson, "Model-Based Throttle Control Using Static Compensators and IMC based PID-Design," *IFAC Workshop on Engine and Powertrain Control, Simulation, and Modeling*, 2009.

[23] M. Vaezi, H. E. Shoori J., and S. Anwar, "IMC-PID Traction Control System for an Automobile via Engine Torque Control," *2015 IEEE International Conference on Industrial Technology*, pp.296-302, 2015.

[24] M. Canale, L. Fagiano, A. Ferrara, and C. Vecchio, "Comparing Internal Model Control and Sliding-Mode Approaches for Vehicle Yaw Control," *IEEE Transactions on Intelligent Transportation Systems*, vol.10, no.1, pp.31-41, 2009.

[25] J. A. Richalet, A. Rault, J. L. Testud, and J. Papon, "Model Predictive Heuristic Control: Applications to an Industrial Process," *Automatica*, vol.14, no.5, pp.413-428, 1978.

[26] C. R. Cutler and B. L. Ramaker, "Dynamic Matrix Control - A Computer Control Algorithm," *AlChE National Mtg.*, 1979.

[27] Y. S. Lee and S. J. Elliott, "Active Position Control of a Flexible Smart Beam Using Internal Model Control," *Journal of Sound and Vibration*, vol.242, no.5, pp.767-791, 2001.

[28] S. Adivikolanu and E. Zafiriou, "Extensions and Performance/Robustness Trade-offs of the EWMA Run-to-run Controller by Using the Internal Model Control Structure," *IEEE Transactions on Electronics Packaging Manufacturing*, vol.23, no.1, pp.56-68, 2000.

[29] Y. Arkun, W. M. Canney, J. Hollett, and M. Morari, "Experimental Study of Internal Model Control," *Ind. Eng. Chem. Process Des. Dev.*, vol.25, no.1, pp.102-108, 1986.

[30] T. H. Lee, T. S. Low, A. Al-Mamun, and C. H. Tan, "Internal Model Control (IMC) Approach for Designing Disk Drive Servo-Controller," *IEEE Transactions on Industrial Electronics*, vol.42, no.3, pp.248-256, 1995.

[31] L. Colasanto, N. G. Tsagarakis, Z. Li, and D. G. Caldwell, "Internal Model Control for Improving the Gait Tracking of a Compliant Humanoid Robot," *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp.5347-5352, 2012.

[32] E. Jahanshahi and S. Skogestad, "Comparison Between Nonlinear Model-based Controllers and Gain-scheduling Internal Model Control based on Identified Model," *IEEE Conference on Decision and Control*, pp.853-860, 2013.

[33] D. E. Rivera, M. Morari, and S. Skogestad, "Internal Model Control: PID Controller Design," *Ind. Eng. Chem. Process Des. Dev.*, vol.25, no.1, pp.252-265, 1986.

[34] R. Vilanova, "IMC based Robust PID Design: Tuning Guidelines and Automatic Tuning," *Journal of Process Control,*, vol.18, no.1, pp.61-70, 2008.

[35] R. M. Hirschorn, "Invertibility of Nonlinear Control Systems," *SIAM Journal on Control and Optimization*, vol.17, no.2, pp.289-297, 1979.

[36] C. G. Economou, M. Morari, and B. O. Palsson, "Internal Model Control: Extension to Nonlinear Systems," *Ind. Eng. Chem. Process Des. Dev.*, vol.25, no.2, pp.403-411, 1986.

[37] M. A. Henson and D. E. Seborg, "An Internal Model Control Strategy for Nonlinear Systems," *AIChE Journal*, vol.37, no.7, pp.1065-1081, 1991.

[38] J. P. Calvet and Y. Arkun, "Feedforward and Feedback Linearization of Nonlinear System and Its Implementation Using Internal Model Control (IMC)," *Industrial & Engineering Chemistry Research*, vol.27, no.10, pp.1822-1831, 1988.

[39] H. T. Toivonen, K. V. Sandström, and R. H. Nyström, "Internal Model Control of Nonlinear Systems Described by Velocity-based Linearizations," *Journal of Process Control*, vol.13, no.3, pp.215-224, 2003.

[40] W. J. Rugh and J. S. Shamma, "Research on Gain Scheduling," *Automatica*, vol.36, no.10, pp.1401-1425, 2000.

[41] J. Mohammapour, J. Sun, A. Karnik, and M. Jankovic, "Internal Model Control Design for Linear Parameter Varying Systems," *American Control Conference*, pp.2409-2414, 2013.

[42] Z. Qiu, J. Sun, M. Jankovic, and M. Santillo, "Nonlinear Internal Model Controller Design for Wastegate Control of a Turbocharged Gasoline Engine," *American Control Conference*, pp.214-219, 2014.

[43] Z. Qiu, J. Sun, M. Jankovic, and M. Santillo, "Nonlinear Internal Model Controller Design for Wastegate Control of a Turbocharged Gasoline Engine," *Control Engineering Practice*, vol.46, pp.105-114, 2016.

[44] K. J. Hunt and D. Sbarbaro, "Neural Networks for Nonlinear Internal Model Control," *IEE Proceedings D - Control Theory and Applications*, vol.138, no.5, pp.431-438, 1991.

[45] E. P. Nahas, M. A. Henson, and D. E. Seborg, "Nonlinear Internal Model Control Strategy for Neural Network Models," *Computers Chem. Eng*, vol.16, no.12, pp.1039-1057, 1992.

[46] I. Rivals and L. Personnaz, "Internal Model Control Using Neural Networks," *Proceedings of the IEEE International Symposium on Industrial Electronics*, vol.1, pp.109-114, 1996.

[47] M. Mohammadzaheri, L. Chen, and S. Grainger, "A Critical Review of the Most Popular Types of Neuro Control," *Asian Journal of Control*, vol.14, no.1, pp.1-11. 2012.

[48] I. Rivals and L. Personnaz, "Nonlinear Internal Model Control Using Neural Networks: Application to Processes with Delay and Design Issues," *IEEE Transactions on Neural Networks*, vol.11, no.1, pp.80-90, 2000.

[49] D. C. Psichogios and L. H. Ungar, "Nonlinear Internal Model Control and Model Predictive Control Using Neural Networks," *5th IEEE International Symposium on Intelligent Control, 1990. Proceedings*, pp.1082-1087, 1990.

[50] A. Aoyama and V. Venkatasubramanian, "Internal Model Control Framework Using Neural Networks for the Modeling and Control of a Bioreactor," *Engineering Applications of Artificial Intelligence*, vol.8, no.6, pp.689-701, 1995.

[51] R. E. Haber and J. R. Alique, "Nonlinear Internal Model Control Using Neural Networks: an Application for Machining Processes," *Neural Computing and Applications*, vol.13, no.1, pp.47-55, 2004.

[52] R. Boukezzoula, S. Galichet, and L. Foulloy, "Fuzzy Nonlinear Adaptive Internal Model Control," *European Journal of Control*, pp.523-540, 2001.

[53] R. Boukezzoula, S. Galichet, and L. Foulloy, "Nonlinear Internal Model Control: Application of Inverse Model based Fuzzy Control," *IEEE Transactions on Fuzzy Systems,* vol.11, no.6, pp.814-829, 2003.

[54] W. F. Xie and A. B. Rad, "Fuzzy Adaptive Internal Model Control," *IEEE Transactions on Industrial Electronics*, vol.47, no.1, pp.193-202, 2000.

[55] K. K. Ahn and N. B. Kha, "Internal Model Control for Shape Memory Alloy Actuators Using Fuzzy Based Preisach Model," *Sensors and Actuators A: Physical*, vol.136, no.2, pp.730-741, 2007.

[56] S. Li and H. Gu, "Fuzzy Adaptive Internal Model Control Schemes for PMSM Speed-Regulation System," *IEEE Transactions on Industrial Informatics*, vol.8, no.4, pp.767-779, 2012.

[57] G. Gregorcic and G. Lightbody, "Gaussian Process Internal Model Control," *International Journal of Systems Science*, vol.43, no.11, pp.2079-2094, 2012.

[58] Z. Zhao, Z. Liu, and J. Zhang, "Internal Model Control Based on LS-SVM for a Class of Nonlinear Process," *2012 International Conference on Solid State Devices and Materials Science*, vol.25, pp.1900-1908, 2012.

[59] A. Bhattacharjee and A. Sutradhar, "Data Driven Nonparametric Identification and Model Based Control of Glucose-insulin Process in Type 1 Diabetics," *Journal of Process Control*, vol.41, pp.15-24, 2016.

[60] A. Bhattacharjee and A. Sutradhar, "Online Identification and Internal Model Control for Regulating Hemodynamic Variables in Congestive Heart Failure Patient," *International Journal of Pharma Medicine and Biological Sciences*, vol.4, no.2, pp.85-89, 2015.

[61] M. Shafiq, "Internal Model Control Structure Using Adaptive Inverse Control Strategy," *ISA Transactions*, vol.44, no.3, pp.353-362, 2005.

[62] M. Shafiq and S. H. Riyaz, "Internal Model Control Structure Using Adaptive Inverse Control Strategy," *4th International Conference on Control and Automation*, pp.148-152, 2003.

[63] B. Widrow and E. Walach, *Adaptive Inverse Control, Reissue Edition: A Signal Processing Approach*, John Wiley & Sons, 2008.

[64] A. Datta, *Adaptive Internal Model Control*, Springer, 1998.

[65] A. Datta and J. Ochoa, "Adaptive internal Model Control: Design and Stability Analysis," *Automatica*, vol.32, no.2, pp.261-266, 1996.

[66] A. Datta and J. Ochoa, "Adaptive Internal Model Control: H_2 Optimization for Stable Plants," *Automatica*, vol.34, no.1, pp.75-82, 1998.

[67] A. Datta and L. Xing, "Adaptive Internal Model Control: H_∞ Optimization for Stable Plants," *IEEE Transactions on Automatic Control*, vol.44, no.11, pp.2130-2134, 1999.

[68] D. Rupp and L. Guzzella, "Adaptive Internal Model Control with Application to Fueling Control," *Control Engineering Practice*, vol.18, no.8, pp.873-881, 2010.

[69] N. R. Lakshmi Narayanan, P. R. Krishnaswamy, and G. P. Rangaiah, "An Adaptive Internal Model Control Strategy for pH Neutralization," *Chemical Engineering Science*, vol.97, no.18, pp.3067-3074, 1997.

[70] X. Shao, J. Zhang, Z. Zhao, and X. Wen, "Adaptive Internal Model Control of Permanent Magnet Synchronous Motor Drive System," *Proceedings of the Eighth International Conference on Electrical Machines and Systems ICEMS*, vol.3, pp.1843-1846, 2005.

[71] Z. Qiu, M. Santillo, M. Jankovic, and J. Sun, "Composite Adaptive Internal Model Control and Its Application to Boost-Pressure Control of a Turbocharged Gasoline Engine," *IEEE Transactions on Control Systems Technology*, vol.23, no.6, pp.2306-2315, 2015.

[72] Z. Qiu, M. Santillo, J. Sun, and M. Jankovic, "Enhanced Composite Adaptive IMC for Boost-Pressure Control of a Turbocharged Gasoline Engine," *American Control Conference*, pp.3286-3291, 2016.

[73] Z. Qiu, J. Sun, M. Jankovic, and M. Santillo, "Generalized Composite Adaptive IMC: Design and Analysis," *IEEE Conference on Decision and Control*, pp.334-340, 2016.

[74] A. Y. Karnik, J. H. Buckland, and J. S. Freudenberg, "Electronic Throttle and Wastegate Control for Turbocharged Gasoline Engines," *American Control Conference*, vol.7, pp.4434-4439, 2005.

[75] J. Buckland, "Estimation Methods for Turbocharged Spark Ignition Engines," Ph.D Dissertation, Dept. Elect. Eng., Univ. Michigan, Ann Arbor, MI, USA, 2009.

[76] J. A. Cook, J. Sun, J. H. Buckland, I. V. Kolmanovsky, H. Peng, and J. W. Grizzle, "Automotive Powertrain Control - A Survey," *Asian Journal of Control*, vol.8, no.3, pp.237-260, 2006.

[77] J. Sun, I. V. Kolmanovsky, J. A. Cook, and J. H. Buckland, "Modeling and Control of Automotive Powertrain Systems: A Tutorial," *American Control Conference*, vol.5, no.13, pp.3271-3283, 2005.

[78] P. A. Ioannou and J. Sun, *Robust Adaptive Control*, Dover, 2012.

[79] S. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, Society for Industrial and Applied Mathematics, 1994.

[80] S. Boyd and L. Vandenberghe, *Convex Programming*, Cambridge University Press, 2004.

[81] H. K. Khalil, *Nonlinear Systems*, Prentice Hall, 2002.

[82] D. Bernstein and W. Haddad, "Robust Stability and Performance Analysis for Linear Dynamic Systems," *IEEE Transactions on Automatic Control*, vol.34, no.7, pp.751-758, 1989.

[83] D. Arzelier, J. Bernussou, and G. Garcia, "Pole Assignment of Linear Uncertain Systems in a Sector via a Lyapunov-type Approach," *IEEE Transactions on Automatic Control*, vol.38, no.7, pp.1128-1131, 1993.

[84] K. S. Tsakalis and P. A. Ioannou, *Linear Time-Varying Systems*, Prentice Hall, 1993.

[85] L. Vandenberghe and S. Boyd, "Semidefinite Programming," *1996 Society for Industrial and Applied Mathematics*, vol.38, no.1, pp.49-95, 1996.

[86] L. Guzzella and C. H. Onder, *Introduction to Modeling and Control of Internal Combustion Engine Systems, 2nd ed.*, Springer, 2010.

[87] P. Moulin and J. Chauvin, "Modeling and Control of the Air System of a Turbocharged Gasoline Engine," *Control Engineering Practice*, vol.19, no.3, pp.287-297, 2011.

[88] A. Nedic, *Network Mathematics Graduate Programme* lecture notes, Hamilton Institute, Maynooth, Ireland, 2008.

[89] J. F. Bonnans and A. Shapiro, *Perturbation Analysis of Optimization Problems*, Springer, 2000.

[90] A. Ilchmann, D. H. Owens, and D. Pratzel-Wolters, "Sufficient Conditions for Stability of Linear Time-Varying Systems," *Systems and Control Letters*, vol.9, no.2, pp.157-163, 1987.