Turbocharged Engine Control for Fuel Efficiency and Torque Responsiveness

by

Raechel Chu-Hui Tan

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering – Mechanical Engineering

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Masayoshi Tomizuka, Chair
Professor J. Karl Hedrick
Professor Laurent El Ghaoui

Summer 2016

Turbocharged Engine Control for Fuel Efficiency and Torque Responsiveness

Abstract

Turbocharged Engine Control for Fuel Efficiency and Torque Responsiveness

by

Raechel Chu-Hui Tan

Doctor of Philosophy in Engineering – Mechanical Engineering

University of California, Berkeley

Professor Masayoshi Tomizuka, Chair

Fuel economy standards for cars and other vehicles are growing increasingly stringent, thus motivating automakers to find ways to improve fuel efficiency. One popular strategy is to turbocharge a downsized (smaller displacement) engine, which can be more fuel efficient than a naturally aspirated engine delivering the same power output. However, turbocharged engines can be sluggish to respond to torque requests, which drivers often find undesirable. Unfortunately, improving torque responsiveness results in reduced fuel efficiency, and vice versa.

This dissertation explores two model-based control strategies to manage this tradeoff. The first strategy is a decentralized controller, in which the throttle and wastegate are controlled in separate loops. The throttle loop uses feedback linearization with supplemental PI control to obtain good torque tracking. The wastegate is opened or closed, based on a preview of the reference torque, to switch between fuel-optimal and torque-optimal modes. The second strategy is a multi-objective optimization scheme to obtain good fuel efficiency and fast torque response by controlling the throttle and wastegate simultaneously. Simulation results show promising performance from both strategies.

Additionally, the models used in these control methods are described in detail. A high-fidelity engine simulator in Simscape is used for controller validation. This simulator is too complex for controller design, so a simpler 4-state model is constructed. This model works well in continuous time, but the optimization-based control method requires a discrete-time model. Unfortunately, discretizing the 4-state model results in chattering due to numerical stiffness. This numerical stiffness is analyzed, and a solution is proposed to represent the throttle pressure ratio as a static map. This results in a 3-state model that is easily discretized.

Contents

List of Figures

List of Tables

Acknowledgments

This dissertation marks the end of six years in graduate school, and the end of ten years as a student at UC Berkeley. I can hardly believe how much how much I've changed since arriving as a freshman in 2006. I feel extraordinarily lucky to have spent the past decade in the beautiful city of Berkeley. And as this chapter of my life comes to a close, there are many people I want to thank.

First and foremost, thank you to my advisor, Professor Masayoshi Tomizuka. Your mentorship in both research and non-research matters helped me greatly on this journey through graduate school. In my undergraduate days, you encouraged me to go for a Ph.D. In my times of uncertainty, you helped to re-inspire me. I am truly grateful to have had the opportunity to work with you.

Thank you to my committee members, Professor J. Karl Hedrick and Professor Laurent El Ghaoui, for your guidance in the writing of this dissertation and the work therein. Your lessons on nonlinear control and optimization theory will serve me well for the rest of my career.

Thank you to the Toyota Motor Corporation for generously providing the funding for this research. Needless to say, this work would not have been possible without your support. Special thanks go to Ken Butts, Koichi Ueda, and Hisahiro Ito, for your valuable feedback and advice.

Thank you to Erin McGarrity and Kevin Oshiro at Mathworks for your assistance with Simscape. With your help, I learned a lot about the technical details of physical modeling.

Thank you to the Mechanical Engineering technical staff in Hesse Hall: Mike Neufer, Alex Jordan, and Pete Graham. You were always there to provide great technical and moral support. Thank you also to the Student Machine Shop staff: Scott McCormick, Mick Franssen, Gordon Long, Dennis Lee, and Jesse Lopez. I really appreciate all the help you've given me, and I am lucky to have learned from your expertise.

Thank you to my labmates in the Mechanical Systems Control Lab: KC, Joonbum, Hoday, Sanggyum, Wenjie, Evan, Pedro, Max, Mike, Oak, Chi-Shen, Wang Cong, Wenlong, Yizhou, Kengo, Chung-Yen, Junkai, Chen-Yu, Minghui, Hiroshi, Omar, Robert, Kevin, Hsien-Chung, Te, Yongxiang, Changliu, Dennis, Shiying, Alice, Yu, Peng Cheng, Daisuke, Jianyu, Shuyang, Kiwoo, Zining, and Liting. I am honored to be your academic sister.

Thank you to the Party Playlist Club for collectively creating over 70 playlists for all of our party needs. Thank you to all of the gym buddies who provided motivation to swat airborne objects and/or lift heavy objects. Thank you to the SMS softball team for providing me with the bragging rights for my first major sporting injury, and for putting up with my utter lack of ability to catch fly balls (at least I've gotten better at batting).

Thank you to all the other amazing friends I met in grad school (or around the periphery): Selina, Shih-Yuan, Andy, Jared, JPort, Chris Meissen, Alyssa, Paul, Chris and Alex Daily-Diamond, Elena, Theresa, Sarah, Ashwin, Claire, Jason, Frank, Jon Beard, Katee, Farzana, Dana, Maritza, Dave, Jorge, and Vince. Grad school would not have been the same without you.

Thank you to my parents, Anthea and Daniel, and my brother, Geoff. I could not have made it this far without your sacrifices, love, and support.

Finally, thank you to Nihal Murthy. I honestly don't think I would have made it to this point if you hadn't been there for me in my darkest days. I am forever indebted to you for your love, laughter, and delicious food that got me to the finish line. As they say, if you like piña coladas...

Chapter 1

Introduction

1.1 Background

Turbocharged engines

Due to concerns about climate change and global oil depletion, governmental requirements for fuel efficiency are growing increasingly stringent. To keep up with these requirements, automotive manufacturers must continually find ways to improve the fuel efficiency of their vehicles. One promising method, called "downsizing," is to reduce the displacement volume of the engine. Downsizing is effective because, for a given torque output, a smaller engine will operate at a higher BMEP* where the fuel consumption is lower [1]. Another way to understand this effect is by noting that a smaller engine must open the throttle valve wider to produce the same torque, which reduces pumping losses and improves overall efficiency.

Of course, a smaller engine also has a lower torque capacity, which means the vehicle will be slower to accelerate – not a desirable effect for drivers. To increase the torque capacity of a downsized engine, some type of forced induction must be used to increase the air and fuel density in the combustion chambers. One option is to add a turbocharger. In addition to increasing the air and fuel density, a turbocharger also recycles some of the energy from the hot and high pressure exhaust gases, which would be unused otherwise. For this reason, downsized and turbocharged engines are quickly becoming a popular fuel-efficient solution with minimal sacrifices to torque capacity.

Compared to a naturally aspirated engine, a turbocharged engine has several additional components (Fig. 1.1). The turbocharger itself consists of a compressor and a turbine, which are linked by a mechanical shaft. As exhaust gases spin the turbine, the turbocharger shaft transmits power to the compressor, which then drives fresh air into the intake path. After passing through the compressor, the fresh air goes through a heat exchanger called an intercooler. The intercooler reduces the temperature of the air to further increase its density. The flow of air through the intake path is regulated by the throttle valve. After passing through the throttle, the air collects in the intake

*BMEP: brake mean effective pressure

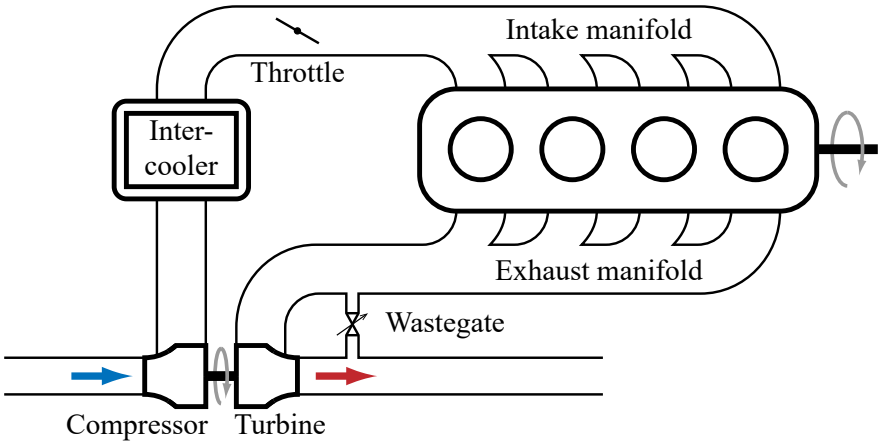


Figure 1.1: Schematic of a typical turbocharged engine airpath

manifold, and then is aspirated into the combustion chambers during the engine's intake stroke. In a direct injected engine, the fuel is injected directly into the combustion chamber and mixes with the fresh air during the engine's compression stroke. After combustion, the burned gas mixture exits into the exhaust manifold, and then passes through the turbine. Many turbochargers are equipped with a wastegate valve, which allows some of the exhaust gases to bypass the turbine. Opening the wastegate means that less exhaust gases pass through the turbine, which results in a slower turbocharger shaft speed and less energy recirculated back into the system.

In exchange for its benefits, a turbocharged engine presents the challenge of balancing fuel efficiency with torque responsiveness. Turbocharged engines can suffer from turbo lag, which is a slow transient response to torque demands. To minimize turbo lag, the wastegate should be kept closed as much as possible, thus keeping the turbocharger speed high. This strategy is often referred to as time-optimal, since it produces the fastest possible torque response. Unfortunately, the time-optimal strategy is detrimental to fuel efficiency because it requires the throttle to be closed more, resulting in a low intake manifold pressure compared to the exhaust manifold pressure. This means that the engine must do more work to "pump" the gases from the low-pressure intake manifold to the high-pressure exhaust manifold. This pumping loss represents fuel energy that is not converted to useful torque at the crankshaft. As shown in [2], the fuel-optimal strategy is to keep the wastegate as open as much as possible, but this is bad for torque responsiveness since the turbocharger speed will be relatively low. Managing this tradeoff in a systematic way is still an open challenge.

Model-based control

The control objectives for an automotive engine include obtaining good fuel efficiency, fast torque responsiveness, maximum torque output, low emissions, and low incidence of knock and misfire. To achieve these goals, there are numerous actuators that may be controlled, including the throttle,

wastegate, spark timing, valve timing, fuel injector timing, and exhaust gas recirculation. If the controllers are designed well, these actuators affect the airpath and combustion properties so that the control objectives are satisfied.

For the throttle and wastegate, a simple control technique is to map the driver's throttle-pedal position to a corresponding throttle angle, while using the wastegate to regulate the intake manifold pressure. This follows the logic of older mechanical systems, where the throttle pedal was mechanically linked to the throttle valve. However, many newer vehicles use drive-by-wire systems, where the throttle pedal simply sends a reference signal to the controller, which in turn sends the appropriate signals to the actuators. Thus, a more modern control technique is to interpret the driver's throttle-pedal position as a desired torque. This approach allows for more sophisticated control over the throttle and wastegate to achieve the desired torque.

From a control perspective, the engine is a multi-input/multi-output plant, with complex and nonlinear relationships between the inputs and outputs. Simple PID[†] control is not adequate, given the strict requirements for the control objectives. A traditional practice is map-driven control, in which characteristic maps (i.e., look-up tables) for the engine outputs are generated by testing the engine on a dynamometer. Based on these maps, the engine controllers can then be constructed as more look-up tables. However, this process is labor-intensive and time-consuming.

As more complexity is added to engine systems, the automotive industry is looking more toward model-based techniques to streamline the control design process. Model-based control can significantly reduce calibration effort and time, since it can account for interdependencies between different variables. This also means that model-based control is more easily adapted to changes in hardware, thus allowing the control design to be performed in parallel with the hardware design. Additionally, if a validation model is available, the controller can be extensively tested in software before being tested on the physical system. Not only are implementation efforts reduced, but the safety and performance of the controller can be more confidently predicted before running on a physical system.

Of course, designing a model-based controller is only possible with a model of the system. Mean-value engine models, such as those developed in [3–5], have become very popular. These models estimate the average values of the model variables and neglect fast variations within one engine cycle. For example, a mean-value model would include the average behavior of the torque output, but would neglect high-frequency torque fluctuations caused by discrete combustion events. This means that the engine dynamics can be captured with relatively few states, which is beneficial for controller design.

1.2 Overview of dissertation work

The goal of this work is to design a control strategy that can reap the benefits of both fuel-optimal and time-optimal strategies. Intuitively, this can be done by operating in fuel-optimal mode (i.e., wastegate open) when the reference torque is constant or decreased, and operating in time-optimal mode (i.e., wastegate closed) when the reference torque is increased. However, this strategy is only

[†]PID: proportional-integral-derivative

possible if a preview of the reference torque is available. This may be a reasonable assumption as certain advanced technologies, such as onboard navigation and autonomous driving systems, are integrated into passenger vehicles. This dissertation provides some evidence that the performance benefits may justify the effort of making a preview available.

One major assumption is that the reference torque signal is composed of a series of setpoints. This means that, over a period of time, the reference torque signal appears as a stepwise function. This is reasonable for the vast majority of driving scenarios, especially highway driving. Nevertheless, there are certain scenarios (e.g., smooth gear shifting [6]) that may require tracking of more complex reference torque trajectories. This type of scenario has not been explored in this work.

Mean-value modeling

Before a control strategy can be designed, a control-oriented model must be constructed. A good control-oriented model should capture the system dynamics with reasonable accuracy, while keeping the number of states relatively small. To this end, a mean-value methodology is used.

In this work, a high-fidelity engine simulator in Simscape is used to evaluate controller performance. However, it is too complex to use for controller design, so a simpler 7-state model is constructed. The 7-state model captures the pressure and turbocharger speed dynamics well, but does not capture the temperature dynamics accurately. The 7-state model is further reduced to a 4-state model, which actually performs better than the 7-state model.

The 4-state model is constructed in continuous time and works well for certain control methods, like feedback linearization. For other methods, like optimization, it is easier to use a discrete-time model. Unfortunately, discretizing the 4-state model results in chattering due to numerical stiffness. To fix this problem, a fast mode is eliminated by representing the throttle pressure ratio as a static map. This results in a 3-state model that is easily discretized.

Control strategies

The main actuators considered in this work are the throttle and wastegate. The engine models include the effect of spark timing, but an advanced control scheme for the spark timing is not developed in this work. This is justified because the spark timing affects the torque output directly, and does not have a large effect on the airpath dynamics. Therefore, a controller for the spark timing may be developed independently and will not significantly interfere with the throttle and wastegate control loops. Exhaust gas recirculation and variable valve timing are also excluded from this study to avoid an overly broad scope.

This work considers two different control strategies:

1. *Decentralized throttle and wastegate control.* The throttle and wastegate are controlled in separate loops. The throttle control loop regulates the torque to follow the reference signal, while the wastegate control loop switches between fuel-optimal and time-optimal modes.

2. *Multi-objective optimization.* The throttle and wastegate are controlled simultaneously in an optimization-based strategy.

In the first control strategy, the separation of the control loops can be justified by two observations. First, the throttle has more control authority than the wastegate over the engine torque. This is because the engine torque is more strongly determined by the intake manifold pressure than any other state, and the intake manifold pressure is directly affected by the throttle position. The wastegate has a much weaker influence on the intake manifold pressure, as the wastegate action must propagate through the state dynamics. Second, the wastegate position has a direct effect on fuel efficiency. This is because the fuel efficiency is closely related to the pumping loss, which is a function of both intake manifold pressure and exhaust manifold pressure, and the latter is directly affected by the wastegate position. By separating the two loops, the wastegate control loop acts as a high-level controller to determine when to switch between fuel-optimal and time-optimal modes, while the throttle control loop acts as a low-level controller for the engine torque.

The throttle controller must handle the nonlinear airpath dynamics to obtain good torque performance. Therefore, the throttle controller is constructed as a feedback linearization algorithm, which compensates for the nonlinear dynamics. The feedback linearization is supplemented with proportional-integral (PI) control to add robustness. This throttle controller produces good torque tracking results, with the engine torque transitioning smoothly and quickly between setpoints.

On the other hand, the wastegate controller does not need to be a complex algorithm, since it only has to switch between open and closed. However, it does require a preview of the reference torque signal so that it can decide the appropriate strategy in advance of a setpoint change. Specifically, the wastegate should be opened until an increase in torque setpoint is detected, closed during the torque step response, and reopened once the torque has settled to the new setpoint. However, it is not obvious how to choose the best timing for closing and reopening the wastegate. Therefore, a wide variety of timing combinations are tested for several setpoint changes, and the resulting torque and fuel performances are evaluated. This is a heuristic approach, but it is very easy to implement since there is not much computation required.

In the second control strategy, a multi-objective optimization scheme is designed to minimize torque tracking error and fuel consumption. Since the nonlinear state constraints make the problem non-convex, the problem is "convexified" by linearizing the dynamics over the system trajectory. This results in a quadratic program, which is solved iteratively. The two objectives are combined in a weighted-sum cost function, which is formulated in three different ways. The results from the three different cost functions are then compared with each other. The optimization scheme is more theoretically rigorous than the decentralized controller, but also requires much more computation.

1.3 Literature review

Control-oriented modeling of turbocharged engine systems has been actively studied [5, 7–15]. These models have been developed specifically to limit the number of states, which is crucial for facilitating the controller design. Our work draws extensively upon the existing literature to develop an appropriate control-oriented model.

There have been several studies on throttle and wastegate control in turbocharged engines. For example, [16] presented gain-scheduled control schemes based on linearized engine models.

Another example is [17], in which the engine model was linearized at several operating points to construct a linear nominal model with uncertainty. A PID controller was then designed to ensure robustness to the model uncertainty. These linearized-model methods offer the advantage of using well-known linear control methods. An alternative to these methods is to design a nonlinear controller that can handle the nonlinear model directly. For example, [10] presented a feedback linearization strategy, although this study focused on control of the wastegate only. In all of these cases, the control objective was to regulate one or more pressures in the engine airpath, and therby the engine torque. In our work, we attempt to control the torque while also minimizing fuel usage.

There has also been some work on optimization-based engine control. For example, model predictive controllers for turbocharged diesel engines were developed in [18, 19]. Model predictive control using preview has also been developed for autonomous driving [20, 21]. Our work draws upon these studies to design an optimization-based strategy.

1.4 Contributions

The contributions of this dissertation can be summarized as follows:

1. Creation of a high-fidelity turbocharged engine simulator in Simscape, and subsequent reduction to 7-state and 4-state control-oriented models.

2. A decentralized throttle and wastegate control strategy to obtain good fuel efficiency and fast torque response. The decentralized controller consists of two parts:

   - The throttle loop uses feedback linearization with supplemental PI control to obtain good torque tracking.
   - The wastegate is opened or closed, based on a preview of the reference torque, to switch between fuel-optimal and torque-optimal modes.

3. Analysis of numerical stiffness in the turbocharged engine model when converting to discrete time, and performance improvement by representing the throttle pressure ratio as a static map.

4. A multi-objective optimization scheme to obtain good fuel efficiency and fast torque response by controlling the throttle and wastegate simultaneously.

1.5 Dissertation outline

The remainder of this dissertation is organized as follows. Chapter 2 introduces the high-fidelity engine simulator and the simpler 7-state and 4-state models. The 4-state model is deemed to be more appropriate for controller design, and is used to design the feedback linearization algorithm in Chapter 3. Chapters 3 and 4 describe the throttle and wastegate loops, respectively, for the decentralized controller. The decentralized controller is validated on the high-fidelity engine simulator.

Chapter 5 describes and analyzes the numerical issues that result when the 4-state model is converted to discrete time, and presents a solution by representing the throttle pressure ratio as a static map. In this way, a fast mode is eliminated and the model is reduced to 3 states. Chapter 6 uses the discretized 3-state model in a multi-objective optimization scheme to determine the optimal throttle and wastegate trajectories. Finally, Chapter 7 presents the main results and conclusions of this work, as well as future work.

Experimental validation was not performed in this work, since a suitable engine test bench was not readily available. However, validation on the engine simulator provides some confidence that the controllers are robust to modeling error.

1.6 Notation

The following notation is used for common variables:

A	Valve open area
H	Enthalpy
J	Moment of inertia
m	Mass
$\dot{m}$	Mass flow rate
R	Specific gas constant
R_u	Universal gas constant
p	Pressure
Q	Heat
SA	Spark advance angle
T	Temperature
u_{th}	Throttle input (percentage)
u_{wg}	Wastegate input (position)
U	Internal energy
V	Volume
W	Work
γ	Heat capacity ratio
η	Efficiency
η_v	Volumetric efficiency
θ_{th}	Throttle angle
Λ_s	Stoichiometric air-fuel ratio
Π	Pressure ratio ($p_{\text{out}}/p_{\text{in}}$)
τ	Torque
ω	Angular velocity

The following subscripts are also used:

$()_a$	Fresh air
$()_{amb}$	Ambient condition
$()_b$	Burned gas
$()_c$	Compressor
$()_e$	Engine
$()_{ei}$	Flow into engine cylinders
$()_{eo}$	Flow out of engine cylinders
$()_f$	Fuel
$()_{em}$	Exhaust manifold
$()_{ic}$	Intercooler
$()_{im}$	Intake manifold
$()_t$	Turbine
$()_{tc}$	Turbocharger
$()_{th}$	Throttle
$()_{wg}$	Wastegate

Chapter 2

Turbocharged Engine Models*

2.1 Chapter overview

This chapter describes the construction of the engine models used in this dissertation. The engine under consideration is a four-cylinder, four-stroke, spark-ignition engine. We first describe a high-fidelity Simscape engine simulator, which is used to evaluate the performance of the controllers. We then construct a simpler 7-state model that considers both the pressures and temperatures in the manifolds, as well as the turbocharger speed, as states. This 7-state model captures the pressure and turbocharger speed dynamics well, but does not capture the temperature dynamics accurately. We then construct 4-state model, which neglects the temperature dynamics. The 4-state model is still able to capture the pressure and turbocharger speed dynamics well. Since it also has fewer states than the 7-state model, the 4-state model is deemed more appropriate for controller design. Finally, we describe the engine torque model that accompanies the 4-state model.

A complete list of model parameters, and the values used, can be found in Appendix A

2.2 Simscape model

A Simscape engine simulator is used to represent the "true" plant. This simulator is a physics-based simulation model that accounts for conservation of mass and energy in the many engine subcomponents. The complete model is more complex than the models used to design the controller. As a result, there is some modeling error which controller must be robust enough to withstand.

The engine simulator is built in Simscape, a toolbox in MATLAB/Simulink that facilitates a physical modeling approach. The Simscape foundation library provides several built-in pneumatic components, such as constant volume chambers, isentropic orifices, and tubes with flow resistance. For more details on the Simscape foundation library, refer to the documentation in [23].

*This chapter contains significant portions from [22], reprinted with permission from IEEE.

Base engine

The base engine simulator represents a naturally aspirated engine, and is used by Toyota's Model-Based Development group as an internal tool. The engine simulator uses custom pneumatic components, modified from the Simscape foundation library, to account for mixing of three gas types: fresh air, fuel vapor, and burned gas.

The air path is constructed from a series of constant volume chambers, including the air cleaner, intake and exhaust manifolds, and various intake and exhaust pipes. Each control volume component is modeled with a lumped parameter approach (i.e., mass and energy are uniformly distributed throughout the volume) and obeys conservation of mass and energy. The control volumes exchange heat with the environment through convection, with varying heat transfer coefficients depending on its location. Isentropic orifices and resistive tubes connect the control volumes together. Exhaust gas recirculation components are also included, but not considered in this study.

The engine block component is based on [14]. This component computes the mean-value torque, mass flow rates entering and exiting the engine cylinders, and conversion of air-fuel mixture to burned gas.

Additional turbocharger components

The base engine simulator was augmented with additional components to convert it to a turbocharged engine. These components are the compressor, turbine, intercooler, air bypass valve, and wastegate. The intercooler, air bypass valve, and wastegate were easily created using the existing Simscape components. However, the compressor and turbine required custom components.

A complete list of model parameters, and the values used, can be found in Appendix A

Compressor

The compressor component equations are based on [13]. These equations compute the molecular flow through the compressor, the enthalpy flows at the compressor inlet and outlet, and the torque absorbed by the compressor. The equations are briefly restated as follows.

The molecular flow rate and mass flow rate are related by:

$$R_u \dot{n}_c = R \dot{m}_c$$

The gas mixture at the compressor inlet is used to determine the specific gas constant R. The compressor molecular flow rate is modeled as:

$$\dot{n}_c = \frac{c_{r_1} p_{\text{in}} \omega_{tc}^{c_{r_2}} \eta_{rot}}{T_{\text{in}} R_u} + c_{g_1}(p_{\text{in}} - p_{\text{out}}) p_{\text{out}} \exp(-c_{g_2} \omega_{tc})$$

where η_{rot} is an efficiency modeled as:

$$\eta_{rot} = \left[1 - \frac{c_{r_4}(p_{\text{out}} - p_{\text{in}})}{\omega_{tc}^{c_{r_3}}}\right]^{c_{r_5}}$$

and the coefficients c_{r_1}–c_{r_5}, c_{g_1}, and c_{g_2} are tuning parameters. The enthalpy flows at the inlet and outlet are:

$$\dot{H}_{\text{in}} = c_p \dot{m}_c T_{\text{in}}$$
$$\dot{H}_{\text{out}} = c_p \dot{m}_c T_{\text{out}}$$

where c_p is the specific heat capacity of the inlet gas mixture. Energy balance gives the following relationship:

$$\dot{H}_{\text{out}} = \dot{H}_{\text{in}} + \tau_c \omega_{tc}$$

The torque absorbed by the compressor is modeled as:

$$\tau_c = c_{t_1} \dot{n}_c^{c_{t_2}} \omega_{tc}^{c_{t_3}}$$

where c_{t_1}, c_{t_2}, and c_{t_3} are tuning parameters. The ten tuning parameters are fit to a compressor map using the methodology outlined in [13].

Turbine

The turbine equations are based on [12, 15]. These equations compute the mass flow rate through the turbine, the enthalpy flows at the turbine inlet and outlet, and the torque produced by the turbine.

The mass flow rate through the turbine is modeled with a modified version of the isentropic orifice equation:

$$\dot{m}_t = A_{\text{eff}} \frac{p_{\text{in}}}{\sqrt{R T_{\text{in}}}} \Psi(\Pi_t)$$

where A_{eff} is the effective turbine area, $\Pi_t = p_{\text{out}}/p_{\text{in}}$ is the turbine pressure ratio, and $\Psi(\cdot)$ is the orifice flow function:

$$\Psi(\Pi) = \begin{cases} \sqrt{\gamma \left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{\gamma-1}}} & \text{for } \Pi < \left(\frac{2}{\gamma+1}\right)^{\frac{\gamma}{\gamma-1}} \\ \Pi^{\frac{1}{\gamma}} \sqrt{\frac{2\gamma}{\gamma-1} \cdot \left(1 - \Pi^{\frac{\gamma-1}{\gamma}}\right)} & \text{for } \Pi \geq \left(\frac{2}{\gamma+1}\right)^{\frac{\gamma}{\gamma-1}} \end{cases} \tag{2.1}$$

The gas mixture at the turbine inlet is used to determine the specific gas constant R and heat capacity ratio γ. The effective turbine area is modeled as:

$$A_{\text{eff}} = A_t \left[\frac{(t_{f1} \tilde{N}_{tc} + t_{f2})}{\Pi_t} + (t_{f3} \tilde{N}_{tc} + t_{f4}) \right]$$

where t_{f1}–t_{f4} and A_t are adjustable parameters, which can be fit to a turbine map. $\tilde{N}_{tc}$ is a scaled turbine parameter, defined as:

$$\tilde{N}_{tc} = \frac{60\, \omega_{tc}}{2\pi \sqrt{T_{\text{in}}}}$$

The enthalpy flows at the turbine inlet and outlet are defined similarly as for the compressor. The energy balance relationship is:

$$\dot{H}_{\text{out}} = \dot{H}_{\text{in}} - \tau_t \omega_{tc}$$

Most literature on turbine models do not model the turbine torque directly. Instead, a model for the turbine efficiency is usually presented (since efficiency maps are more readily available), and torque is then calculated from its physical relationship to efficiency:

$$\tau_t = \frac{c_p \dot{m}_t T_{\text{in}} \left(1 - \Pi_t^{\frac{\gamma-1}{\gamma}}\right)}{\omega_{tc}} \cdot \eta_t$$

However, the equation has a singularity when $\omega_{tc} = 0$, which Simscape cannot tolerate. Therefore, we seek to eliminate singularities in the turbine torque model.

We start with the efficiency model presented in [15]:

$$\begin{aligned}\eta_t &= \eta_{t,max} \cdot \left[1 - \left(\frac{BSR - BSR_{opt}}{BSR_{opt}}\right)^2\right] \\ &= \eta_{t,max} \cdot \left[\frac{2\,BSR}{BSR_{opt}} - \left(\frac{BSR}{BSR_{opt}}\right)^2\right]\end{aligned}$$

BSR is the blade-speed ratio, defined as:

$$BSR = \frac{r_t \omega_{tc}}{\sqrt{2 c_p T_{\text{in}} \left(1 - \Pi_t^{\frac{\gamma-1}{\gamma}}\right)}}$$

where r_t is the turbine blade radius. $\eta_{t,max}$ and BSR_{opt} are tuning parameters representing the maximum efficiency and the optimal blade-speed ratio, respectively.

We can construct a model for the turbine torque using this efficiency model. We substitute the full expression for BSR in the efficiency model and set it equal to the physical definition of η_t:

$$\frac{\tau_t \omega_{tc}}{\dot{m}_t \alpha} = \eta_{t,max} \cdot \left[\frac{2 r_t \omega_{tc}}{BSR_{opt}\sqrt{2\alpha}} - \frac{r_t^2 \omega_{tc}^2}{BSR_{opt}^2 \cdot 2\alpha}\right]$$

where $\alpha = c_p T_{\text{in}} \left(1 - \Pi_t^{(\gamma-1)/\gamma}\right)$. The turbine torque can then be simplified to:

$$\tau_t = \eta_{t,max} \dot{m}_t \left[\frac{r_t \sqrt{2\alpha}}{BSR_{opt}} - \frac{r_t^2 \omega_{tc}}{2\,BSR_{opt}}\right]$$

Note that this simplification step is valid when $\omega_{tc} \neq 0$ and $\Pi_t \neq 1$. These conditions are only violated when the turbocharger is completely at rest, and therefore should not be problematic for most scenarios. The final turbine torque model is free from singularities and implicitly matches the typical efficiency model when the turbocharger is spinning.

Turbocharger shaft dynamics

The turbocharger shaft is represented with a rotational inertia component from the Simscape foundation library. The shaft dynamics obey Newton's second law:

$$\dot{\omega}_{tc} = \frac{\tau_t - \tau_c}{J_{tc}}$$

2.3 7-state control-oriented model

Although the Simscape engine simulator has an underlying model, it is too complex to use for model-based control. Therefore, a simpler model must be created for the basis of control design.

The simpler model is constructed from alternating manifolds and restrictions, such as those presented in [9, 11, 15]. The manifolds are control volumes where mass can accumulate: the intercooler, intake manifold, and exhaust manifold. The restrictions are components that describe the flow of mass and energy: the throttle, wastegate, compressor, turbine, and engine. An air bypass valve was not considered in this study. We also do not consider the effect of exhaust gas recirculation, so we do not need to account for gas mixing.

In this model, both the pressure and temperature in each manifold are considered as dynamic states. Although many engine models in the literature do not consider manifold temperatures as states, we wish to determine if these additional dynamics can provide better accuracy.

Manifolds

The three manifolds considered are the intercooler, intake manifold, and exhaust manifold. Each manifold is assumed to have uniform pressure and temperature distributions. A lumped-parameter approach is taken, so that the temperature at the manifold outlet is assumed to be equal to the temperature in the manifold ($T = T_{\text{out}}$). Each manifold has a fixed volume V which can accumulate and lose gas molecules.

Internal energy and mass are conserved quantities, i.e.:

$$\frac{\mathrm{d}}{\mathrm{d}t}U(t) = \dot{U}_{\text{in}}(t) - \dot{U}_{\text{out}}(t) + \dot{Q}(t) + \dot{W}(t) \tag{2.2}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}m(t) = \dot{m}_{\text{in}}(t) - \dot{m}_{\text{out}}(t) \tag{2.3}$$

where $\dot{Q}$ is the rate of heat gain and $\dot{W}$ is the rate of work done on the gas. (Heat loss and work done by the gas have negative values.) The work done on or by the gas can be further separated as:

$$\dot{W}(t) = \frac{\mathrm{d}}{\mathrm{d}t}(p_{\text{in}}V_{\text{in}}) - \frac{\mathrm{d}}{\mathrm{d}t}(p_{\text{out}}V_{\text{out}})$$

Using the fundamental relationship between enthalpy and internal energy ($H = U + pV$), Eq. 2.2 can be restated as:

$$\frac{\mathrm{d}}{\mathrm{d}t}U(t) = \dot{H}_{\text{in}}(t) - \dot{H}_{\text{out}}(t) + \dot{Q}(t) \tag{2.4}$$

Eqs. 2.3 and 2.4 must now be reformulated to obtain the pressure and temperature dynamics. A reasonable assumption is that the manifold contains an ideal gas, so that the following relationship for internal energy can be used:

$$U(t) = c_v \cdot m(t) \cdot T(t) \tag{2.5}$$

where c_v is the specific heat of the gas at constant volume. Additionally, the enthalpy flows in and out of the manifold can be expressed as:

$$\dot{H}_{\text{in}}(t) = c_p \cdot \dot{m}_{\text{in}}(t) \cdot T_{\text{in}}(t) \tag{2.6a}$$
$$\dot{H}_{\text{out}}(t) = c_p \cdot \dot{m}_{\text{out}}(t) \cdot T_{\text{out}}(t) \tag{2.6b}$$

where c_p is the specific heat of the gas at constant pressure. Substituting Eqs. 2.5 and 2.6 into Eq. 2.4 yields:

$$c_v \dot{m} T + c_v m \dot{T} = c_p(\dot{m}_{\text{in}} T_{\text{in}} - \dot{m}_{\text{out}} T_{\text{out}}) + \dot{Q}$$

(The explicit dependencies on time have been omitted for clarity.) With some algebraic manipulation, this can be expressed as a dynamic equation for temperature:

$$\dot{T} = \frac{RT}{pV}\left[\gamma \dot{m}_{\text{in}} T_{\text{in}} - \gamma \dot{m}_{\text{out}} T_{\text{out}} - (\dot{m}_{\text{in}} - \dot{m}_{\text{out}}) T_{\text{out}} + \frac{\dot{Q}}{c_v}\right] \tag{2.7}$$

where $\gamma = c_p / c_v$.

The assumption of ideal gases also allows the use of the ideal gas law:

$$p(t) \cdot V = m(t) \cdot R \cdot T(t)$$

Differentiating both sides and some algebraic manipulation leads to the following relationship:

$$\dot{m} = \frac{\dot{p}V}{RT} - \frac{pV}{RT^2}\dot{T} \tag{2.8}$$

Substituting Eqs. 2.7 and 2.8 into Eq. 2.3 and rearranging yields a dynamic equation for pressure:

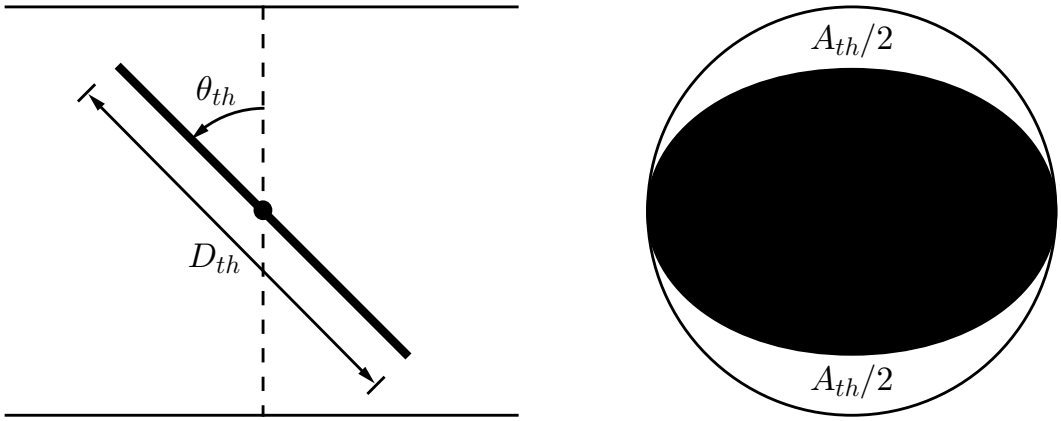
$$\dot{p} = \frac{\gamma R}{V}\left[\dot{m}_{\text{in}} T_{\text{in}} - \dot{m}_{\text{out}} T_{\text{out}} + \frac{\dot{Q}}{c_p}\right] \tag{2.9}$$

Eqs. 2.7 and 2.9 are the dynamic equations upon which each manifold model is built.

For each manifold, the heat exchange $\dot{Q}$ is modeled as convection:

$$\dot{Q}_{ic} = h_{ic}(T_{amb} - T_{ic}) \tag{2.10}$$
$$\dot{Q}_{im} = h_{im}(T_{amb} - T_{im}) \tag{2.11}$$
$$\dot{Q}_{em} = h_{em}(T_{ex} - T_{em}) \tag{2.12}$$

where h_{ic}, h_{im}, and h_{em} are the effective heat transfer coefficients (including the effect of surface area) for the intercooler, intake manifold, and exhaust manifold. T_{amb} is the ambient temperature on the intake side, and T_{ex} is the ambient temperature on the exhaust side.



(a) Side view (b) Front view

Figure 2.1: Throttle valve geometry

Throttle and wastegate

The valves considered in this study are the throttle and wastegate. These are modeled as isentropic orifices:

$$\dot{m} = A \cdot \frac{p_{\text{in}}}{\sqrt{RT_{\text{in}}}} \cdot \Psi(\Pi) \tag{2.13}$$

where A is the open area of the valve and $\Psi(\Pi)$ is the flow function given in Eq. 2.1.

The throttle is typically a butterfly valve. Its open area is given by:

$$A_{th} = \frac{\pi}{4} D_{th}^2 \cdot \frac{u_{th}}{100}$$

where D_{th} is the throttle bore diameter. The throttle input u_{th} is related to the throttle angle by simple geometry (Fig. 2.1):

$$u_{th} = 100 \left[1 - \cos\left(\theta_{th} \cdot \frac{\pi}{180} \right) \right]$$

The wastegate is typically a poppet valve, so the open area is approximated as:

$$A_{wg} = \frac{1}{8} D_{wg} S_{wg} \cdot \frac{u_{wg}}{100}$$

where D_{wg} and S_{wg} are the wastegate diameter and stroke.

Engine mass flows

We assume that the fuel is directly injected into the combustion chamber and neglect the time delays associated with the combustion cycle. Thus, the mass flow rate exiting the cylinders is approximately equal to the aspirated mass flow rate plus the fuel mass flow rate:

$$\dot{m}_{eo} = \dot{m}_{ei} + \dot{m}_f$$

The injected fuel rate is assumed to maintain stoichiometric conditions:

$$\dot{m}_f = \frac{\dot{m}_{ei}}{\Lambda_s}$$

The aspirated mass flow rate is modeled using the volumetric pump model given in [9]:

$$\dot{m}_{ei} = \rho_{im} \cdot \frac{V_d \omega_e}{4\pi} \cdot \eta_v = \frac{p_{im}}{RT_{im}} \cdot \frac{V_d \omega_e}{4\pi} \cdot \eta_v \tag{2.14}$$

where ρ_{im} is the density of the intake air and V_d is the engine displacement volume. The volumetric efficiency η_v is modeled as:

$$\eta_v(p_{im}, p_{em}, \omega_e) = \eta_{vp}(p_{im}, p_{em}) \cdot \eta_{v\omega}(\omega_e)$$

The first part η_{vp} is only a function of the intake and exhaust manifold pressures:

$$\eta_{vp}(p_{im}, p_{em}) = \frac{V_c + V_d}{V_d} - \left(\frac{p_{em}}{p_{im}}\right)^{\frac{1}{\gamma}} \cdot \frac{V_c}{V_d}$$

where V_c is the compressed volume of the engine. The second part $\eta_{v\omega}$ is fit to measurements of aspirated mass flow rate at various engine speeds.

Engine-out temperature

The temperature exiting the engine exhaust port is mainly a function of the spark timing and the fuel mass in the combustion chamber [15]. Increasing the spark advance results in a lower engine-out temperature, since advancing the spark timing allows more time for combustion before the exhaust valve is opened. This allows more energy to be converted into useful work, instead of waste heat. (In real engines, the longer time allowed for combustion also results in additional heat being absorbed by the cylinder walls, which further reduces the exit temperature. However, the engine simulation does not account for this effect.) Adding more fuel results in a higher engine-out temperature, since more combustion heat is produced. Since the injected fuel rate is assumed to maintain stoichiometric conditions, increasing the aspirated mass flow rate will cause a higher engine-out temperature.

The engine-out temperature is modeled in two parts:

$$T_{eo} = T_{e,m}(\dot{m}_{ei}) \cdot T_{e,s}(SA)$$

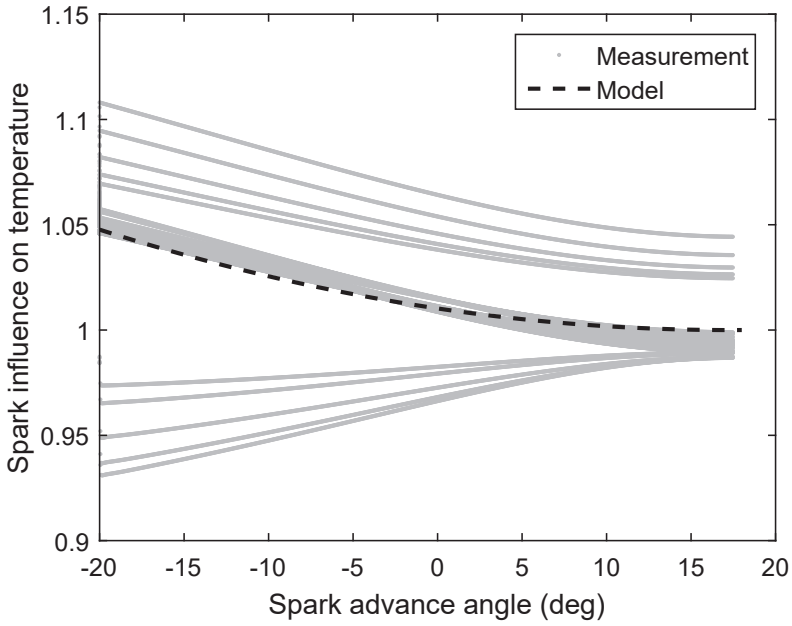
where $T_{e,m}$ depends only on the aspirated mass flow rate, and $T_{e,s}$ depends only on the spark advance angle. $T_{e,m}$ is modeled as a linear function of the aspirated mass flow rate:

$$T_{e,m}(\dot{m}_{ei}) = d_{m_1}\dot{m}_{ei} + d_{m_2}$$

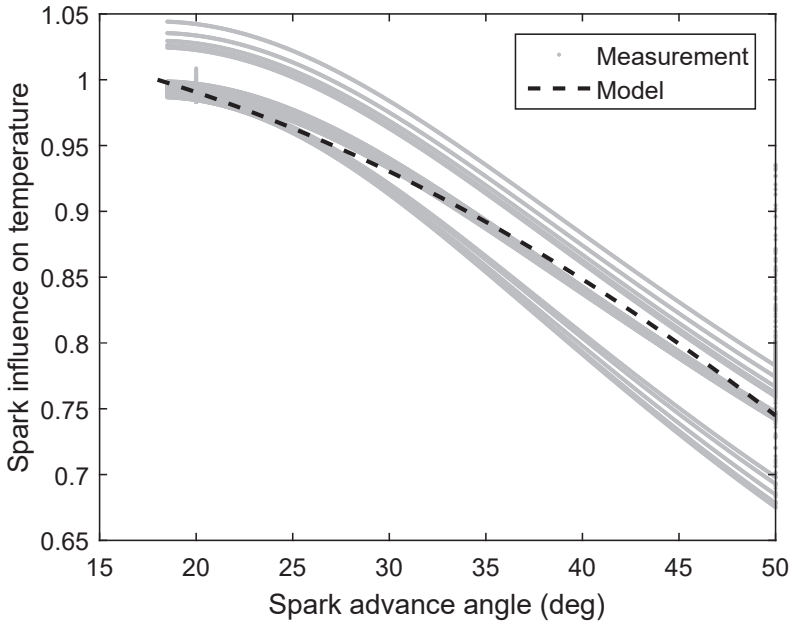
$T_{e,s}$ was split into two parts after observing that $SA = 18$ deg seemed to be a transition point. Each part is a quadratic function:

$$T_{e,s} = \begin{cases} d_{s_1,lo}(SA-18)^2 + d_{s_2,lo}(SA-18) + 1 & \text{if } SA \leq 18 \text{ deg} \\ d_{s_1,hi}(SA-18)^2 + d_{s_2,hi}(SA-18) + 1 & \text{if } SA > 18 \text{ deg} \end{cases}$$

The fits of both quadratic functions are shown in Fig. 2.2.



(a) $SA \leq 18^{\circ}$



(b) $SA > 18^{\circ}$

Figure 2.2: Spark influence on engine exit temperature

Turbocharger

The turbocharger components are modeled as described in Section 2.2. The temperature at the outlet of the compressor can be determined from the energy balance:

$$T_c = T_{amb} + \frac{\tau_c \omega_{tc}}{c_{p,a} \dot{m}_c}$$

where $c_{p,a}$ is the specific heat capacity of air.

Overall model accuracy

The preceding equations can be combined to form a nonlinear dynamic system:

$$\begin{aligned}
\dot{p}_{ic} &= \frac{\gamma_a R_a}{V_{ic}} \left\{ \dot{m}_c(p_{ic}, \omega_{tc}) T_c(p_{ic}, \omega_{tc}) - \dot{m}_{th}(p_{ic}, p_{im}, T_{ic}, u_{th}) T_{ic} + \frac{\dot{Q}_{ic}(T_{ic})}{c_{p,a}} \right\} \\
\dot{p}_{im} &= \frac{\gamma_a R_a}{V_{im}} \left\{ \dot{m}_{th}(p_{ic}, p_{im}, u_{th}) T_{ic} - \dot{m}_{ei}(p_{im}, p_{em}, T_{im}) T_{im} + \frac{\dot{Q}_{im}(T_{im})}{c_{p,a}} \right\} \\
\dot{p}_{em} &= \frac{\gamma_b R_b}{V_{em}} \left\{ \dot{m}_{eo}(p_{im}, p_{em}, T_{im}) T_{eo}(p_{im}, p_{em}, SA) - \left[\dot{m}_t(p_{em}, \omega_{tc}) + \dot{m}_{wg}(p_{em}, u_{wg}) \right] T_{em} \right. \\
&\qquad \left. + \frac{\dot{Q}_{em}(T_{em})}{c_{p,b}} \right\} \\
\dot{T}_{ic} &= \frac{R_a T_{ic}}{p_{ic} V_{ic}} \left\{ \gamma_a \dot{m}_c(p_{ic}, \omega_{tc}) T_c(p_{ic}, \omega_{tc}) - \gamma_a \dot{m}_{th}(p_{ic}, p_{im}, T_{ic}, u_{th}) T_{ic} \right. \\
&\qquad \left. - \left[\dot{m}_c(p_{ic}, \omega_{tc}) - \dot{m}_{th}(p_{ic}, p_{im}, T_{ic}, u_{th}) \right] T_{ic} + \frac{\dot{Q}_{ic}(T_{ic})}{c_{v,a}} \right\} \\
\dot{T}_{im} &= \frac{R_a T_{im}}{p_{im} V_{im}} \left\{ \gamma_a \dot{m}_{th}(p_{ic}, p_{im}, u_{th}) T_{ic} - \gamma_a \dot{m}_{ei}(p_{im}, p_{em}, T_{im}) T_{im} \right. \\
&\qquad \left. - \left[\dot{m}_{th}(p_{ic}, p_{im}, u_{th}) - \dot{m}_{ei}(p_{im}, p_{em}, T_{im}) \right] T_{im} + \frac{\dot{Q}_{im}(T_{im})}{c_{v,a}} \right\} \\
\dot{T}_{em} &= \frac{R_b T_{em}}{p_{em} V_{em}} \left\{ \gamma_b \dot{m}_{eo}(p_{im}, p_{em}, T_{im}) T_{eo}(p_{im}, p_{em}, SA) - \gamma_b \left[\dot{m}_t(p_{em}, \omega_{tc}) + \dot{m}_{wg}(p_{em}, u_{wg}) \right] T_{em} \right. \\
&\qquad \left. - \left[\dot{m}_{eo}(p_{im}, p_{em}, T_{im}) - \dot{m}_t(p_{em}, \omega_{tc}) - \dot{m}_{wg}(p_{em}, u_{wg}) \right] T_{em} + \frac{\dot{Q}_{em}(T_{em})}{c_{v,b}} \right\} \\
\dot{\omega}_{tc} &= \frac{1}{J_{tc}} \left\{ \tau_t(p_{em}, \omega_{tc}) - \tau_c(p_{ic}, \omega_{tc}) \right\}
\end{aligned}$$

where the subscripts $(\,)_a$ and $(\,)_b$ indicate the fresh air and burned gas values, respectively, for R, γ, c_p, and c_v. Note that this dynamic system has a standard nonlinear form:

$$\dot{x} = f(x,u) \qquad x \in \mathbb{R}^7, u \in \mathbb{R}^3$$

where $x = [\,p_{ic}\ \ p_{im}\ \ p_{em}\ \ T_{ic}\ \ T_{im}\ \ T_{em}\ \ \omega_{tc}\,]^{\mathrm{T}}$ is the state vector and $u = [\,u_{th}\ \ u_{wg}\ \ SA\,]^{\mathrm{T}}$ is the input vector.

The 7-state model is now compared to the true behavior. The 7-state model is simulated in Simulink, using the variable-step solver ode23s. The true behavior is obtained by running the Simscape engine simulator at a fixed step size of 1 ms. In both cases, the engine speed is set constant at 2000 RPM and the spark advance angle is set constant at 10 deg.

The results show that the model predicts the pressure and turbocharger speed states fairly well, but predicts the temperature states very poorly (Fig. 2.3). The fast transient behavior for T_{im} goes in the opposite direction as the true behavior, while T_{em} barely shows any transient behavior at all. The most likely explanation is that the simple heat exchange models (Eqs. 2.10–2.12) are not sufficiently accurate. Additionally, it should be noted that typical automotive temperature sensors have time constants of several seconds, and therefore would most likely be unable to measure the fast transient behavior observed from the true plant. For these reasons, we conclude that the modeling of the temperature dynamics may not provide better accuracy.

In the following section, we investigate a model that neglects the temperature dynamics.

2.4 4-state control-oriented model

It is shown in [11] that the loss of model accuracy due to neglecting the manifold temperature dynamics may be acceptable for control design purposes. Since the 7-state model did not predict the temperature dynamics well, we now investigate a simpler model that neglects these dynamics.

Manifolds

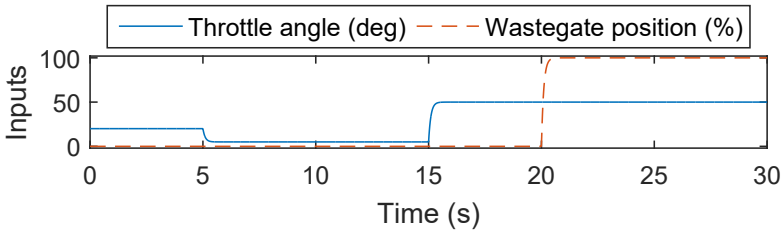
Each manifold is still assumed to have a uniform pressure distribution. However, the temperature is now assumed to be constant throughout each manifold. The pressure dynamics of each manifold can then be modeled using the ideal gas law:

$$\dot{p} = \frac{RT}{V}(\dot{m}_{\mathrm{in}} - \dot{m}_{\mathrm{out}}) \tag{2.15}$$

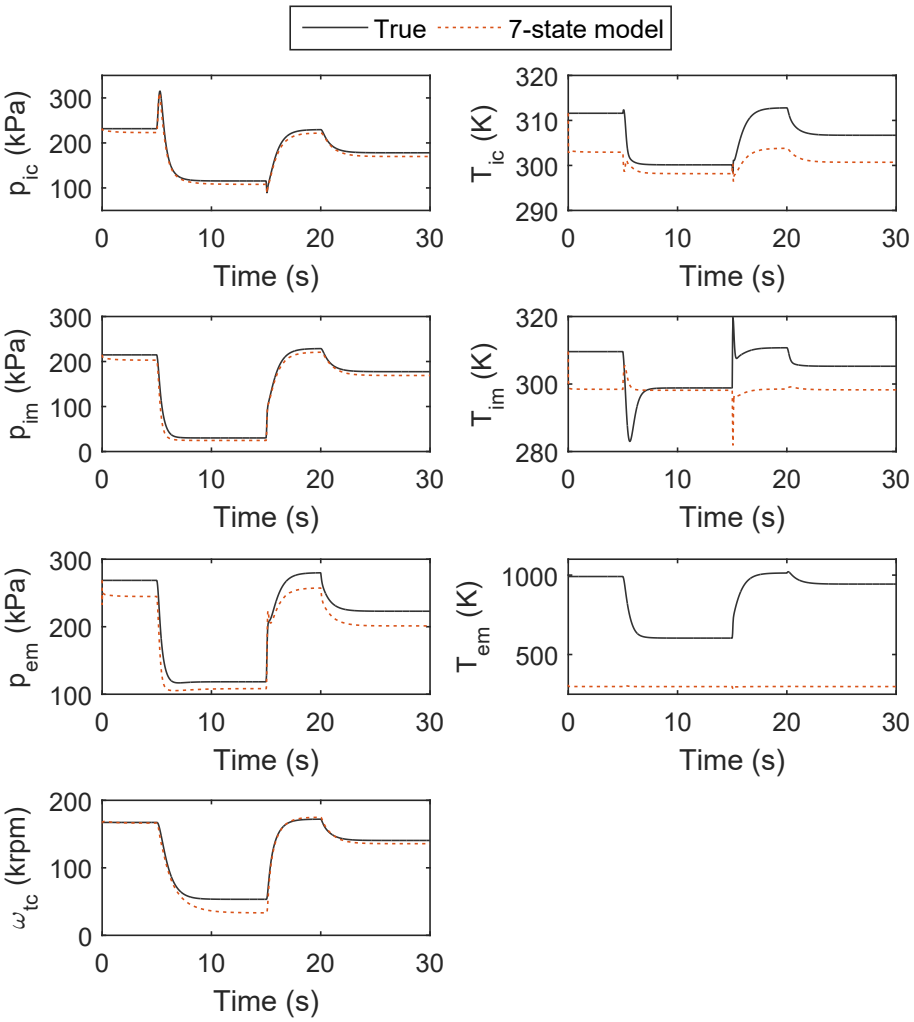
The temperature of each manifold is set to an average value, based on observations from the true behavior. It should be noted that this formulation neither obeys conservation of energy nor accounts for heat transfer effects. However, this is acceptable for the purposes of controller design. The authors of [11] suggest using static maps to define the manifold temperatures, but we observed that using constant temperatures was sufficiently accurate.

Throttle and wastegate

The throttle and wastegate mass flow rates are modeled as described in Section 2.3.



(a) Input signals



(b) States

Figure 2.3: 7-state model vs. true behavior

Engine mass flows

The mass flow rate exiting the cylinders is modeled as described in Section 2.3, with the injected fuel rate assumed to maintain stoichiometric conditions.

The aspirated mass flow rate model is initially modeled with the volumetric pump model, as described in Section 2.3. However, when the intake manifold temperature is assumed to be constant, we observed that the aspirated mass flow rate can be approximated by a linear function of the form:

$$\dot{m}_{ei} \approx a_1 p_{im} + a_2 p_{em} + a_3 \tag{2.16}$$

The linear approximation matches the nonlinear model very well, except for regions where p_{im} is very low and p_{em} is very high (Fig. 2.4). Fortunately, normal engine operating conditions will not enter this region very often, since p_{em} cannot rise too high when p_{im} is small. Therefore, we use the simpler, linear approximation in Eq. 2.16 for control design.

On an aside, note the magnitudes of the fit values for the a_i coefficients:

$$\begin{aligned} a_1 &= 2.26 \times 10^{-4} \ \frac{\text{kg/s}}{\text{kPa}} \\ a_2 &= -2.95 \times 10^{-5} \ \frac{\text{kg/s}}{\text{kPa}} \\ a_3 &= 2.3 \times 10^{-3} \ \text{kg/s} \end{aligned}$$

Since $|a_1|$ is an order of magnitude larger than $|a_2|$, we can conclude that the aspirated mass flow rate (and therefore the engine torque) depends more strongly on the intake manifold pressure than the exhaust manifold pressure.

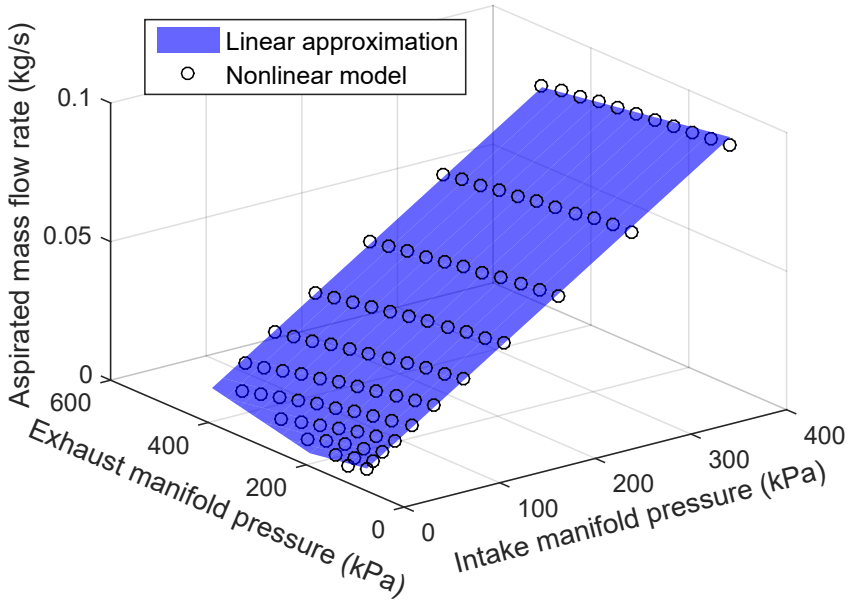
Turbocharger

The turbocharger components are modeled as described in Section 2.2, although enthalpy flows are not considered.

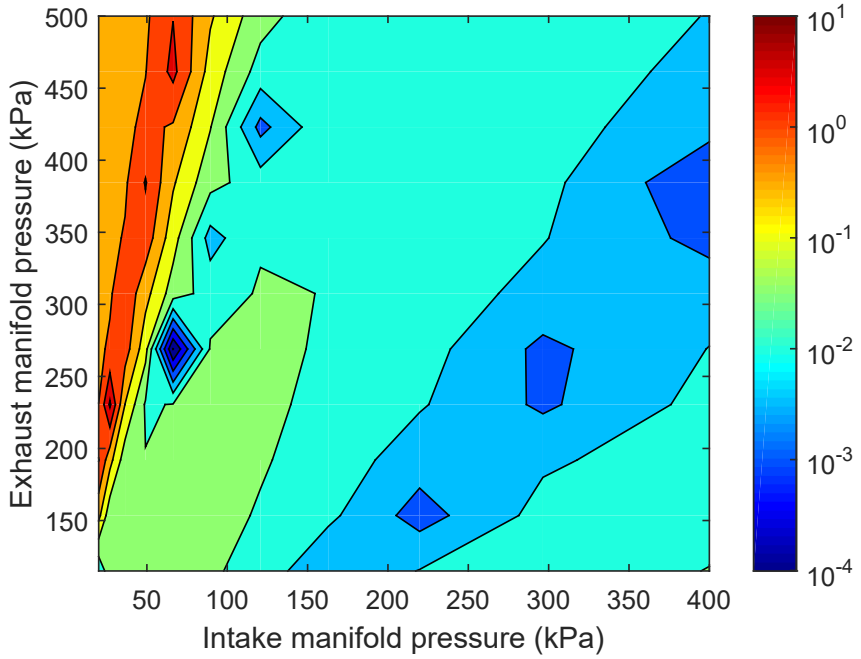
Overall model accuracy

The preceding equations can be combined to form a nonlinear dynamic system:

$$\begin{aligned} \dot{p}_{ic} &= \frac{R_a T_{ic}}{V_{ic}} \Big[\dot{m}_c(p_{ic}, \omega_{tc}) - \dot{m}_{th}(p_{ic}, p_{im}, u_{th}) \Big] \\ \dot{p}_{im} &= \frac{R_a T_{im}}{V_{im}} \Big[\dot{m}_{th}(p_{ic}, p_{im}, u_{th}) - \dot{m}_{ei}(p_{im}, p_{em}) \Big] \\ \dot{p}_{em} &= \frac{R_b T_{em}}{V_{em}} \Big[\dot{m}_{eo}(p_{im}, p_{em}) - \dot{m}_t(p_{em}, \omega_{tc}) - \dot{m}_{wg}(p_{em}, u_{wg}) \Big] \\ \dot{\omega}_{tc} &= \frac{1}{J_{tc}} \Big[\tau_t(p_{em}, \omega_{tc}) - \tau_c(p_{ic}, \omega_{tc}) \Big] \end{aligned}$$



(a) Linear fit



(b) Relative error between the linear and nonlinear models

Figure 2.4: Linear approximation of aspirated mass flow rate (Eq. 2.16) compared to nonlinear model (Eq. 2.14)

This dynamic system has a standard nonlinear form:

$$\dot{x} = f(x,u) \qquad x \in \mathbb{R}^4, u \in \mathbb{R}^2 \tag{2.17}$$

where $x = [\,p_{ic} \;\; p_{im} \;\; p_{em} \;\; \omega_{tc}\,]^{\mathrm{T}}$ is the state vector and $u = [\,u_{th} \;\; u_{wg}\,]^{\mathrm{T}}$ is the input vector. Note that in this model, the spark timing does not affect the state dynamics. The spark timing only affects the engine torque output, which will be described in Section 2.5.

The 4-state model is now compared to the true behavior. The 4-state model is simulated in Simulink, using the variable-step solver ode23s. The true behavior is obtained by running the Simscape engine simulator at a fixed step size of 1 ms. In both cases, the engine speed is set constant at 2000 RPM.

The results show that the 4-state model predicts the manifold pressures and turbocharger speed quite well (Fig. 2.5). In fact, the exhaust manifold pressure is predicted more accurately in the 4-state model than in the 7-state model. The 4-state model has the additional benefit of having fewer states, which is better for controller design. Therefore, we choose to use the 4-state model to design our controller.

2.5 Engine torque model

In this section, an engine torque model is developed to accompany the 4-state model. The engine torque is considered an output, and not a dynamic state.

As with the other variables, the engine torque is modeled using a mean-value methodology. The mean-value torque is calculated by taking the net work produced over one cycle, and averaging it over the rotation of one cycle (4π radians for a 4-stroke engine). The net work consists of the indicated work minus the pumping work.

Indicated work

The indicated work (the work produced before losses) is the work produced by the combustion of fuel. Therefore, the injected fuel mass is a key component in the indicated work model. Another major component is the spark timing. The indicated work is modeled as suggested in [15].

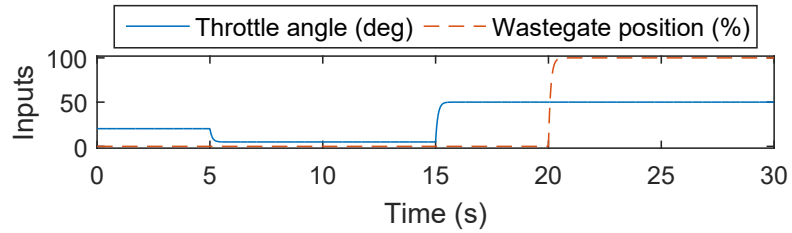
The fuel mass captured in the engine cylinders over one cycle can be approximated as:

$$m_f = \dot{m}_f \cdot \frac{4\pi}{\omega_e}$$

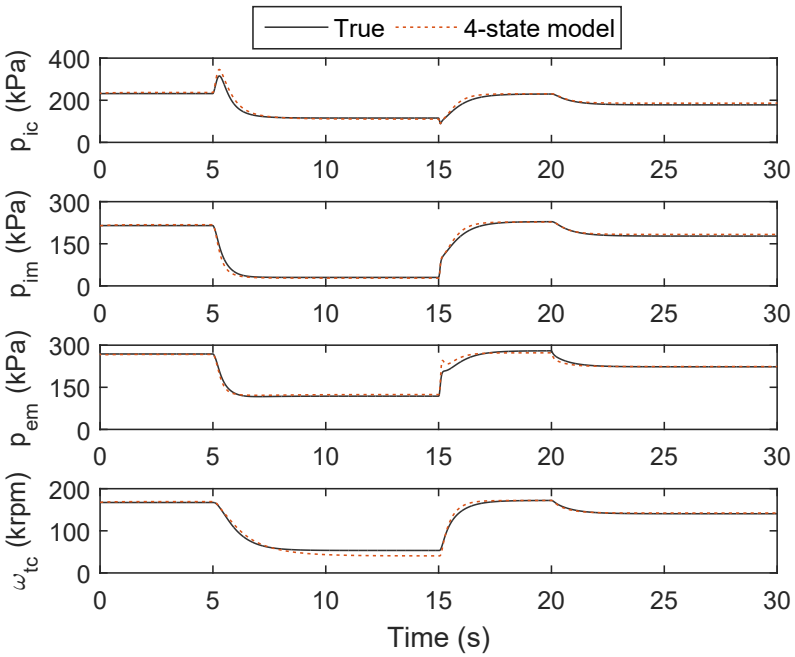
The indicated work over one cycle is then modeled as:

$$W_i = m_f \cdot q_{LHV} \cdot \eta_o \cdot \eta_{SA}$$

where q_{LHV} is the lower heating value of the fuel, η_o is the efficiency for an ideal Otto cycle, and η_{SA} is a spark influence function that models the effect of the spark timing.



(a) Input signals



(b) States

Figure 2.5: 4-state model vs. true behavior (© 2016 IEEE)

The efficiency for an ideal Otto cycle is modeled as:

$$\eta_o = 1 - \frac{1}{r_c^{\gamma-1}}$$

where r_c is the engine compression ratio.

The spark influence function η_{SA} is modeled as a third-order polynomial function of the spark advance angle, as suggested in [15]:

$$\eta_{SA} = b_1\left[1 - b_2(SA - SA_{opt})^2 - b_3(SA - SA_{opt})^3\right]$$

where b_1, b_2, b_3, and SA_{opt} are tuning parameters. These parameters were fit to data collected from the Simscape engine simulator (Fig. 2.6).

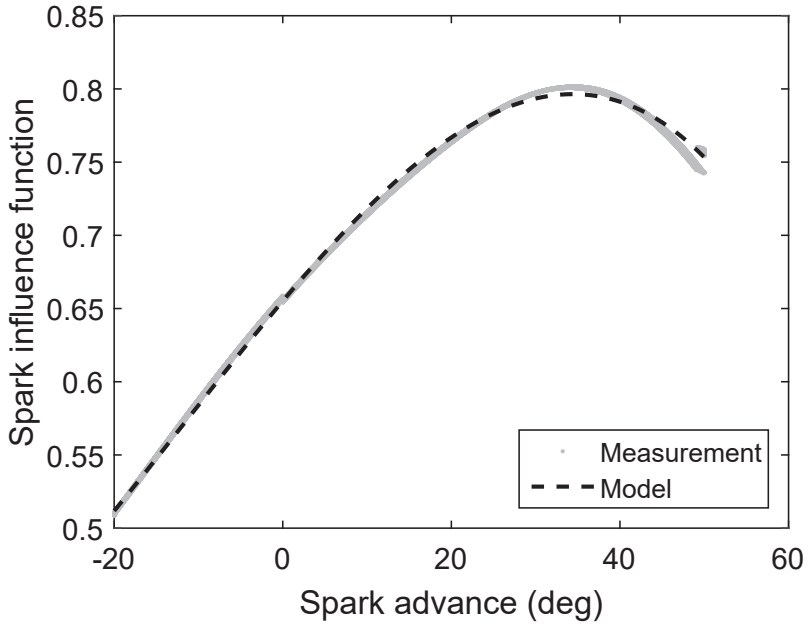


Figure 2.6: Spark influence on engine torque

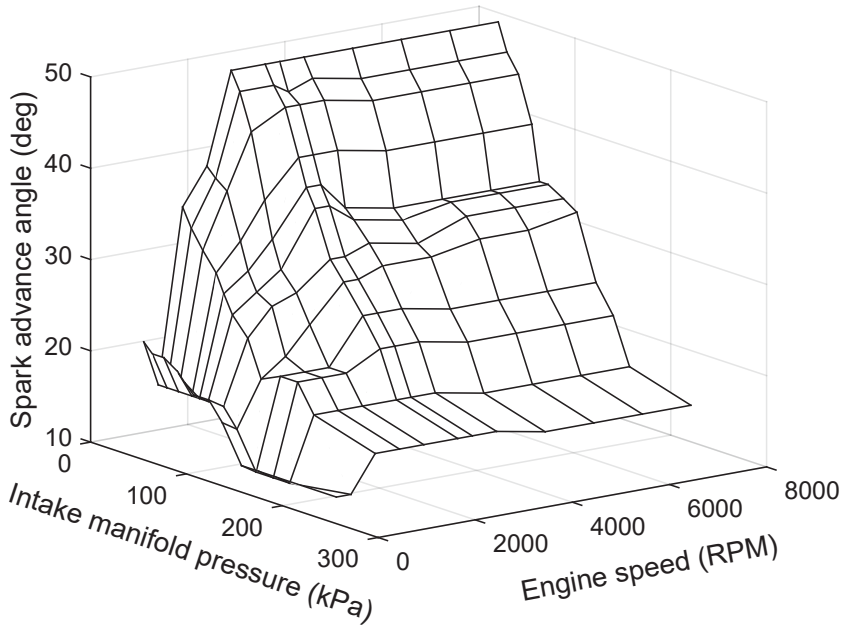


Figure 2.7: Look-up table for spark advance angle (degrees before top-dead-center)

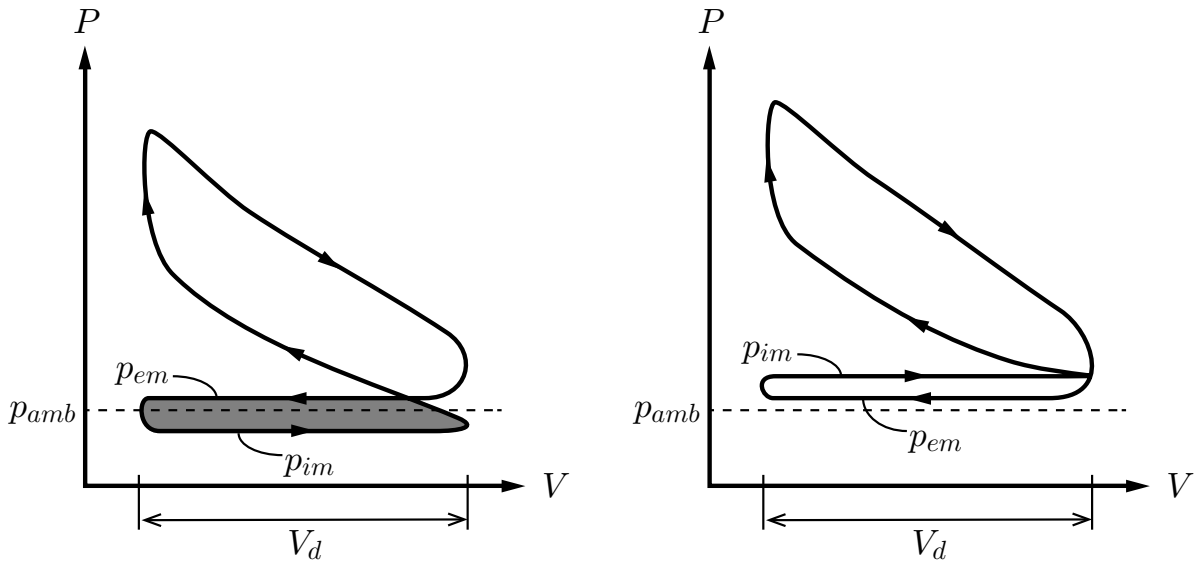
In this work, we do not consider advanced control of the spark advance angle. Instead, we use a conventional look-up table for the spark advance angle, based on the intake manifold pressure and engine speed (Fig. 2.7):

$$SA = SA(p_{im}, \omega_e)$$

Pumping work

One significant source of losses is the pumping effect of moving gases from the lower pressure intake manifold to the higher pressure exhaust manifold. When the intake pressure is lower than the exhaust pressure, the engine must do work to "pump" the gases through, and therefore the pumping work is positive. When the intake pressure is higher than the exhaust pressure, the engine is actually assisted by the pressure difference, and therefore the pumping work is negative. The pumping loop can be seen on the pressure-volume diagram of the combustion cycle (Fig. 2.8). The pumping work is modeled by approximating the pumping loop by a rectangle [24, p. 727]:

$$W_p = V_d(p_{em} - p_{im})$$



(a) Work lost to pumping, $W_p > 0$

(b) Work gained from pumping, $W_p < 0$

Figure 2.8: Pressure-volume (P-V) diagrams of combustion cycle. The shaded region indicates work lost to pumping.

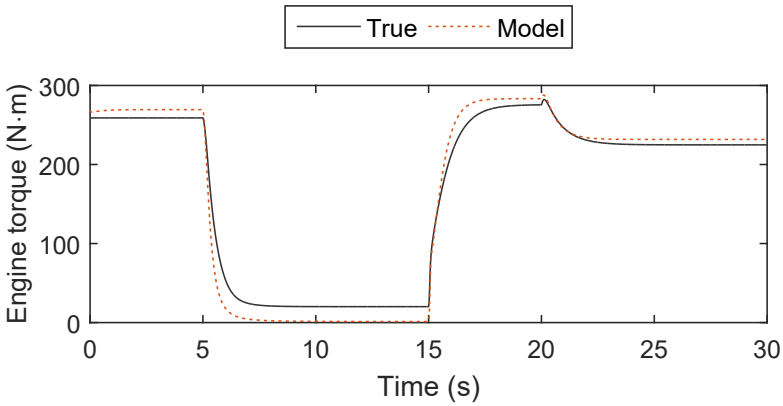


Figure 2.9: Torque model vs. true behavior

Net torque

The net work produced by the engine is the indicated work minus the pumping work. The engine torque is then calculated by averaging the engine work over one cycle:

$$W_e = W_i - W_p$$
$$\tau_e = \frac{W_e}{4\pi} \tag{2.18}$$

Other mechanical and frictional losses are not considered, since they are not captured in the engine simulation.

The torque model is combined with the 4-state model (from Section 2.4) and compared to the true behavior. The results show that the torque model has some steady-state error, but captures transient behavior well (Fig. 2.9). The steady-state error in the torque model is a result of steady-state error in the four states. However, this steady-state error is not a large concern since it can be compensated by using integrator action in the controller.

2.6 Chapter summary

This chapter described the construction of three engine models. The first was a high-fidelity Simscape engine simulator, which is considered to be the "true" plant. The second was a simpler 7-state model, which captured the pressure and turbocharger speed dynamics well, but did not capture the temperature dynamics accurately. The third was a simpler 4-state model, which captured the pressure and turbocharger speed dynamics better than the 7-state model, even though the temperature dynamics were neglected. A torque model was then created to accompany the 4-state model. This overall model was observed to be a good approximation of the true plant, particularly in the capturing of transient behavior. Some steady-state error was observed, but this can be compensated by using integrator action in the controller. The 4-state model will be used for controller design in the next chapter, while the engine simulator will be used for controller validation.

Chapter 3

Low-Level Torque Control via Throttle Actuation

3.1 Chapter overview

This chapter describes the first part of a strategy that separates the throttle and wastegate controls. In this chapter, we design a throttle control strategy to follow the reference torque signal. The wastegate position is assumed to be known, so that the throttle control loop acts as a low-level controller for the engine torque.

The throttle controller is constructed as a feedback linearization algorithm, which compensates for the nonlinear dynamics. The algorithm uses the 4-state model from Section 2.4. However, the model is slightly modified so that the throttle position is the only input and the engine torque is the only output. This results in a single-input/single-output system, which is easily managed by the feedback linearization algorithm.

Additionally, the engine speed, spark timing, and wastegate position are considered as measurable parameters. This means that they are treated as constants during the feedback linearization design, but they are updated with their measured values when the algorithm is executed. We can justify this treatment for each of the three variables. The engine speed tends to change much slower than the airpath states, since it evolves according to the overall powertrain and vehicle dynamics. Thus, it makes sense to treat the engine speed as nearly constant. The spark timing affects the torque output directly and does not propagate through the airpath states. Thus, we treat the spark timing as constant to avoid excessive complexity during the feedback linearization design. The wastegate position will be determined in the wastegate control loop, and will simply switch between open and closed.

The state variables are also scaled to provide better numerical conditioning. This scaling corresponds to using units of kPa for the pressure states and krad/s for the turbocharger speed state.

The state x, input u, output y, and parameter w are summarized as:

$$x = \begin{bmatrix} p_{ic} \cdot 10^{-3} \\ p_{im} \cdot 10^{-3} \\ p_{em} \cdot 10^{-3} \\ \omega_{tc} \cdot 10^{-3} \end{bmatrix} \qquad u = u_{th} \qquad y = \tau_e \qquad w = \begin{bmatrix} u_{wg} \\ \omega_e \\ SA \end{bmatrix}$$

The model can be expressed a form where the input appears affinely:

$$\begin{aligned} \dot{x} &= F(x,w) + G(x,w) \cdot u \\ y &= H(x,w) \end{aligned}$$

The nonlinear functions F and G are:

$$F(x,w) = \begin{bmatrix} \frac{R_a T_{ic}}{V_{ic}} \dot{m}_c(x,w) \\ -\frac{R_a T_{im}}{V_{im}} \dot{m}_{ei}(x,w) \\ \frac{R_b T_{em}}{V_{em}} \left(\dot{m}_{eo}(x,w) - \dot{m}_t(x,w) - \dot{m}_{wg}(x,w) \right) \\ \frac{1}{J_{tc}} \left(\tau_t(x,w) - \tau_c(x,w) \right) \end{bmatrix}$$

$$G(x,w) = \begin{bmatrix} -\frac{R_a T_{ic}}{V_{ic}} \left(\frac{10^3 \pi D_{th}^2 x_1}{100 \cdot 4 \sqrt{R_a T_{ic}}} \Psi\left(\frac{x_2}{x_1}\right) \right) \\ \frac{R_a T_{im}}{V_{im}} \left(\frac{10^3 \pi D_{th}^2 x_1}{100 \cdot 4 \sqrt{R_a T_{ic}}} \Psi\left(\frac{x_2}{x_1}\right) \right) \\ 0 \\ 0 \end{bmatrix}$$

The nonlinear function H is the engine torque equation (Eq. 2.18).

3.2 Feedback linearization

The output has relative degree 1, meaning that the input appears after taking a single derivative:

$$\begin{aligned} \dot{y} &= \frac{\partial H(x,w)}{\partial x} \cdot F(x,w) + \frac{\partial H(x,w)}{\partial x} \cdot G(x,w) \cdot u \\ &= L_F H(x,w) + L_G H(x,w) \cdot u \end{aligned}$$

where Lie derivative notation is used above. Note that $L_G H(x,w) \neq 0$.

We define the synthetic input $v := L_F H(x,w) + L_G H(x,w) \cdot u$. After designing v, we can apply the control input:

$$u = \frac{v - L_F H(x,w)}{L_G H(x,w)} \tag{3.1}$$

This yields a linear input-output relationship: $\dot{y} = v$.

We design the control law for the synthetic input as:

$$v = \dot{y}_d + c_0(y_d - y) \tag{3.2}$$

where y_d is the desired output and c_0 is a constant tuning parameter. This yields the following closed-loop error dynamics:

$$(\dot{y}_d - \dot{y}) + c_0(y_d - y) = 0 \tag{3.3}$$

As long as $c_0 > 0$, the closed-loop error dynamics will be exponentially stable.

Note that the control law in Eq. 3.2 requires the derivative of the desired output. This is not readily available, since the torque reference is usually a series of setpoints. Therefore, we use a reference model to generate y_d and $\dot{y}_d$:

$$\lambda \dot{y}_d + y_d = r$$

where λ is the time constant and r is the original reference signal. We choose $\lambda = 0.1$ seconds, which is sufficiently fast for a typical torque response. From this reference model, we easily obtain $\dot{y}_d$, as well as a smoothed version of the reference signal.

Stability analysis of zero dynamics

Since the relative degree of the system is smaller than the size of the state, we must analyze the stability of the unobservable internal dynamics. The standard procedure is to analyze the stability of the "zero" dynamics, i.e., the internal dynamics when y is invariant ($\dot{y} = 0$).

The first step is to define a change of coordinates between the original states x and the transformed states $[y \;\; \xi^{\mathrm{T}}]^{\mathrm{T}}$, where ξ are the states of the internal dynamics. In this case, the relative degree is 1 while the size of x is 4, so the size of ξ is $(4-1) = 3$. Thus, we must choose (ξ_1, ξ_2, ξ_3) to obtain:

$$\begin{bmatrix} y \\ \xi_1 \\ \xi_2 \\ \xi_3 \end{bmatrix} = \Phi(x)$$

where $\Phi(\cdot)$ is a diffeomorphism (smooth, 1-to-1 map). By the Inverse Function Theorem, a sufficient condition for Φ to be a diffeomorphism is if its Jacobian matrix is full rank. Choosing $\xi_1 = x_1$, $\xi_2 = x_3$, and $\xi_3 = x_4$ yields the following Jacobian for Φ:

$$\mathbf{D}\Phi = \begin{bmatrix} 0 & \frac{\partial H}{\partial x_2} & \frac{\partial H}{\partial x_3} & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

It is clear that $\mathbf{D}\Phi$ is full rank when $\frac{\partial H}{\partial x_2} \neq 0$. Looking closely at this term reveals:

$$\frac{\partial H}{\partial x_2} = \frac{10^3 V_d}{4\pi} + \frac{a_1 \eta_o \eta_{SA} q_{LHV}}{\Lambda_s \omega_e}$$

Note that a_1 is one of the constants from Eq. 2.16, and is strictly positive. All other parameters in the above equation are also strictly positive for all physically possible conditions. Therefore, we can conclude that $\mathbf{D}\Phi$ is full rank for all physically possible conditions, and our choice for Φ is valid. This means that we can invert Φ to express x in terms of (y, ξ):

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \Phi^{-1}(y,\xi) = \begin{bmatrix} \xi_1 \\ \frac{\pi\Lambda_s\omega_e y + (250\Lambda_s\omega_e V_d - a_2\pi q_{LHV}\eta_o\eta_{SA})\xi_2 - a_3\pi q_{LHV}\eta_o\eta_{SA}}{250\Lambda_s\omega_e V_d + a_1\pi q_{LHV}\eta_o\eta_{SA}} \\ \xi_2 \\ \xi_3 \end{bmatrix}$$

The next step is to obtain the ξ-dynamics in autonomous form, i.e., $\dot{\xi} = \Gamma(\xi)$. However, our choice of ξ does not immediately lead to this form, since we have:

$$\begin{bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \\ \dot{\xi}_3 \end{bmatrix} = \begin{bmatrix} F_1(x,w) \\ F_3(x,w) \\ F_4(x,w) \end{bmatrix} + \begin{bmatrix} G_1(x,w) \\ 0 \\ 0 \end{bmatrix} \cdot u = \Gamma'(x,u,w)$$

To remove u as a dependency, we substitute our control law from Eqs. 3.1 and 3.2:

$$u = \frac{\dot{y}_d + c_0(y_d - y) - L_F H(x,w)}{L_G H(x,w)}$$

We assume a constant desired torque, so $\dot{y}_d = 0$. Since the closed-loop error dynamics (Eq. 3.3) are exponentially stable, we can conclude that $y \to y_d$ and $\dot{y} \to 0$. Therefore:

$$u \to \frac{-L_F H(x,w)}{L_G H(x,w)}$$

Now, using the above control law and our change of coordinates, we obtain:

$$u = \frac{-L_F H\left(\Phi^{-1}(y_d,\xi),w\right)}{L_G H\left(\Phi^{-1}(y_d,\xi),w\right)} \quad \Longrightarrow \quad \begin{aligned} \dot{\xi} &= \Gamma'\left(\Phi^{-1}(y_d,\xi), \frac{-L_F H\left(\Phi^{-1}(y_d,\xi),w\right)}{L_G H\left(\Phi^{-1}(y_d,\xi),w\right)}, w\right) \\ &= \Gamma(y_d,\xi,w) \end{aligned}$$

After assigning constant values to y_d and w, we finally obtain the autonomous dynamics $\dot{\xi} = \Gamma(\xi)$.

The last step is to use Lyapunov's linearization method to analyze the local stability of the autonomous ξ-dynamics. More precisely, we linearize the ξ-dynamics around its equilibrium point, and analyze the local stability of the equilibrium point using linear methods. To linearize the ξ-dynamics, we calculate the Jacobian matrix of Γ:

$$\mathbf{D}\Gamma = \begin{bmatrix} \frac{\partial\Gamma_1}{\partial\xi_1} & \frac{\partial\Gamma_1}{\partial\xi_2} & \frac{\partial\Gamma_1}{\partial\xi_3} \\ \frac{\partial\Gamma_2}{\partial\xi_1} & \frac{\partial\Gamma_2}{\partial\xi_2} & \frac{\partial\Gamma_2}{\partial\xi_3} \\ \frac{\partial\Gamma_3}{\partial\xi_1} & \frac{\partial\Gamma_3}{\partial\xi_2} & \frac{\partial\Gamma_3}{\partial\xi_3} \end{bmatrix}$$

This would be fairly straightforward, except that several terms require differentiating the piecewise orifice flow function $\Psi(\Pi)$ (Eq. 2.1). Fortunately, $\frac{\partial \Psi}{\partial \Pi}$ is piecewise continuous:

$$\frac{\partial \Psi}{\partial \Pi} = \begin{cases} 0 & \text{for } \Pi < \left(\frac{2}{\gamma+1}\right)^{\frac{\gamma}{\gamma-1}} \\ \frac{-(\gamma+1)\Pi+2\Pi^{\frac{1}{\gamma}}}{(\gamma-1)\Pi\sqrt{\frac{-2\gamma}{\gamma+1}\left(\Pi^{\frac{\gamma-1}{\gamma}}-1\right)}} & \text{for } \Pi \geq \left(\frac{2}{\gamma+1}\right)^{\frac{\gamma}{\gamma-1}} \end{cases}$$

For the throttle flow function, the pressure ratio is the one across the throttle, so $\Pi = (x_2/x_1)$. For the wastegate flow function, the pressure ratio is the one across the wastegate, so $\Pi = (p_{ex} \times 10^{-3}/x_3)$. We can use our change of coordinates to express these in terms of ξ. Thus, we can compute the relevant $\mathbf{D}\Gamma_{ij}$ terms using the chain rule:

$$\mathbf{D}\Gamma_{ij} = \frac{\partial \Gamma_i}{\partial \xi_j} + \frac{\partial \Gamma_i}{\partial \Psi} \cdot \frac{\partial \Psi}{\partial \Pi} \cdot \frac{\partial \Pi}{\partial \xi_j}$$

After computing $\mathbf{D}\Gamma$, we can check its eigenvalues to evaluate local stability of the equilibrium point. Since the equilibrium point depends on our choice of y_d and w, we repeat the stability analysis at several different values of (y_d, w). We choose SA according to the spark advance look-up table (Fig. 2.7), but y_d, u_{wg}, and ω_e can be chosen independently. Several different choices all produce negative eigenvalues for $\mathbf{D}\Gamma$ (Table 3.1). Note that the exact values of the eigenvalues are not too important, as long as they all have negative real parts. Therefore, we conclude that the zero dynamics are stable.

Table 3.1: Eigenvalues of $\mathbf{D}\Gamma$ for various choices of y_d and u_{wg}, at $\omega_e = 2000$ RPM

	$y_d = 50$ N·m			$y_d = 200$ N·m		
$u_{wg} = 0\%$	-3133.5	-59.5	-15.1	-4191.8	-59.8	-6.4
$u_{wg} = 50\%$	-2302.1	-81.9	-23.0	-3767.8	-58.9	-7.7
$u_{wg} = 100\%$	-1693.6	-115.5	-34.5	-3360.5	-65.1	-9.4

3.3 Simulation results

We first test the feedback linearization algorithm on the 4-state model. We run the plant model in continuous time, but we run the controller in discrete time with a step size of 10 ms. The plant measurements are sampled at this fixed rate, and the control signals are passed through a zero-order-hold. Although the sampler and zero-order-hold introduce some modeling error, it is small enough that the feedback algorithm still performs very well (Figs. 3.1–3.3). Using a large value of c_0 in the control law (Eq. 3.2) speeds up the system response, but also increases the amount of overshoot for torque steps up, especially when the step is large (Figs. 3.2–3.3).

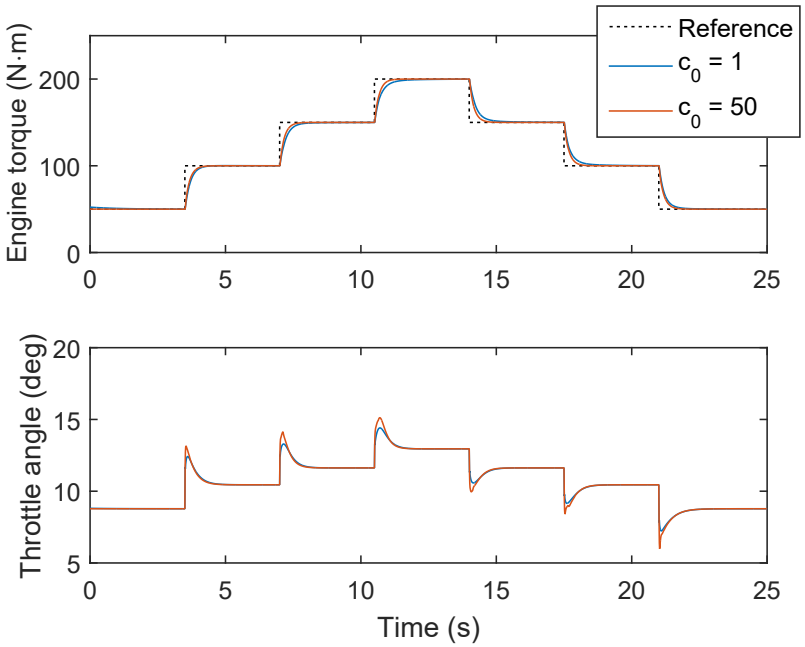


Figure 3.1: Simulation results of implementing the feedback linearization algorithm on the 4-state model. Torque setpoint changes are relatively small. Engine speed held constant at 2000 RPM. Wastegate held at 0% (fully closed).

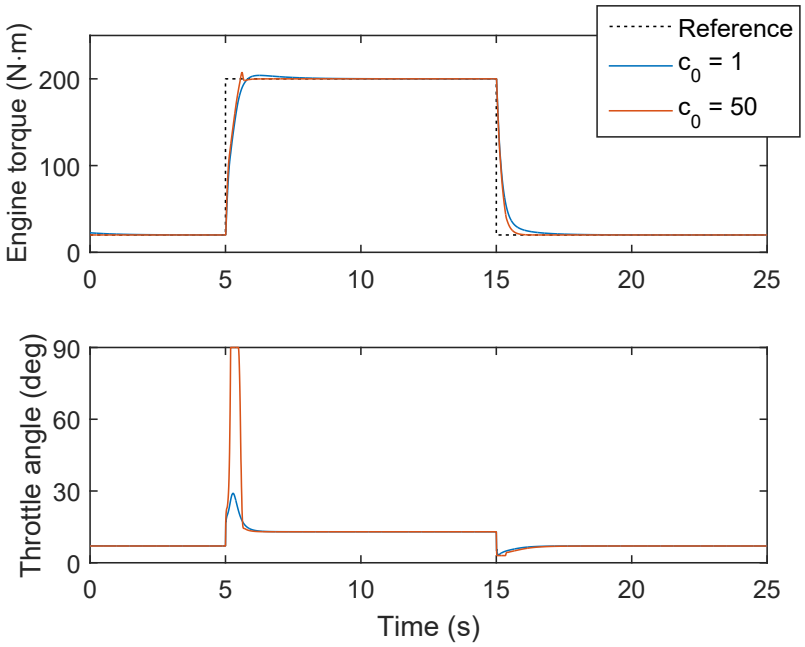


Figure 3.2: Simulation results of implementing the feedback linearization algorithm on the 4-state model. Torque setpoint changes are relatively large. Engine speed held constant at 2000 RPM. Wastegate held at 0% (fully closed).

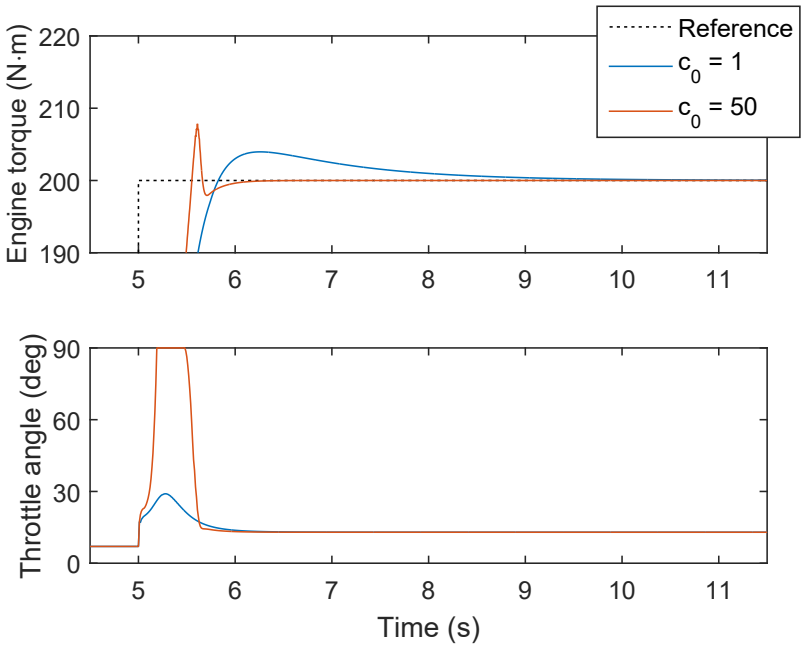


Figure 3.3: Closeup of overshoot from the plot in Fig. 3.2

3.4 Implementation on Simscape engine simulator

Next, we test the feedback linearization algorithm on the Simscape engine simulator. We still run the controller with a 10 ms step size, along with a sampler and zero-order-hold. In this case, there is substantially more modeling error. Because of this, feedback linearization alone cannot provide robust performance. Therefore, we supplement the feedback linearization with proportional-integral (PI) control (Fig. 3.4).

The PI block is placed outside of the feedback linearization block so that it is applied directly to the throttle angle instead of the synthetic input. This is done to facilitate the use of integrator clamping (to prevent integrator windup when the control signal saturates). If the integrator were applied to the synthetic input, the integrator clamp would require the saturation limits of the synthetic input, which are state-dependent. Instead, we can use the fixed saturation limits of the throttle angle.

Additionally, the output of the PI block is scaled according to the output of the feedback linearization block. This is done because the torque output is much more sensitive to throttle changes when the throttle is near closed than when it is near wide-open. Scaling down the output of the PI block protects against undesirable oscillation when the throttle is near closed.

The feedback linearization and PI blocks work together to obtain good torque tracking performance. The feedback linearization handles transient response, while the integral control compensates for steady-state error. The proportional control helps with both, and additionally helps reduce overshoot for large torque steps up. This combination provides better performance than feedback linearization alone (Figs. 3.5–3.6).

The feedback linearization parameter was set to $c_0 = 30$, while the PI gains were set to $K_p = 2$ and $K_i = 5$. All three values were manually tuned to achieve good performance.

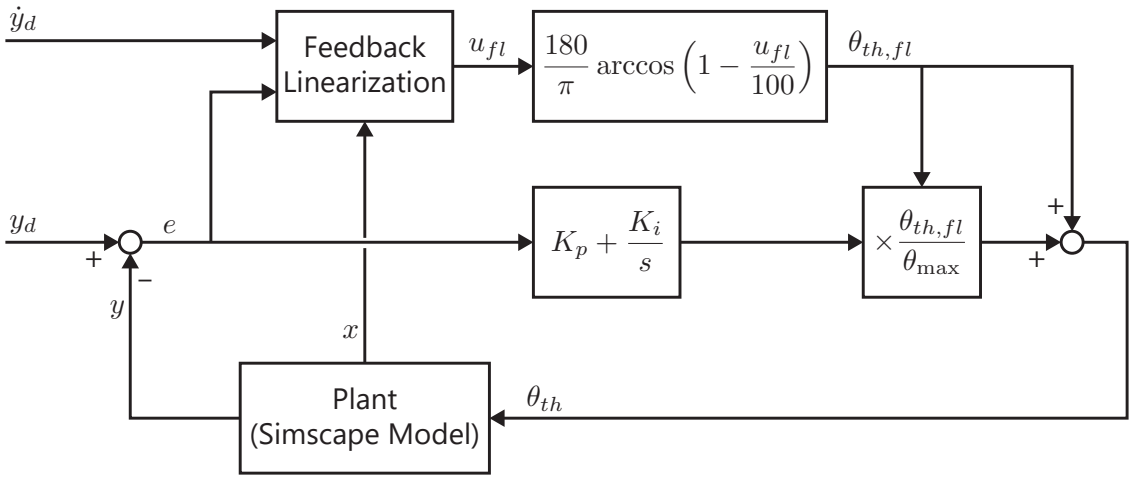


Figure 3.4: Block diagram for feedback linearization with supplemental PI control. Integrator clamping algorithm not shown.

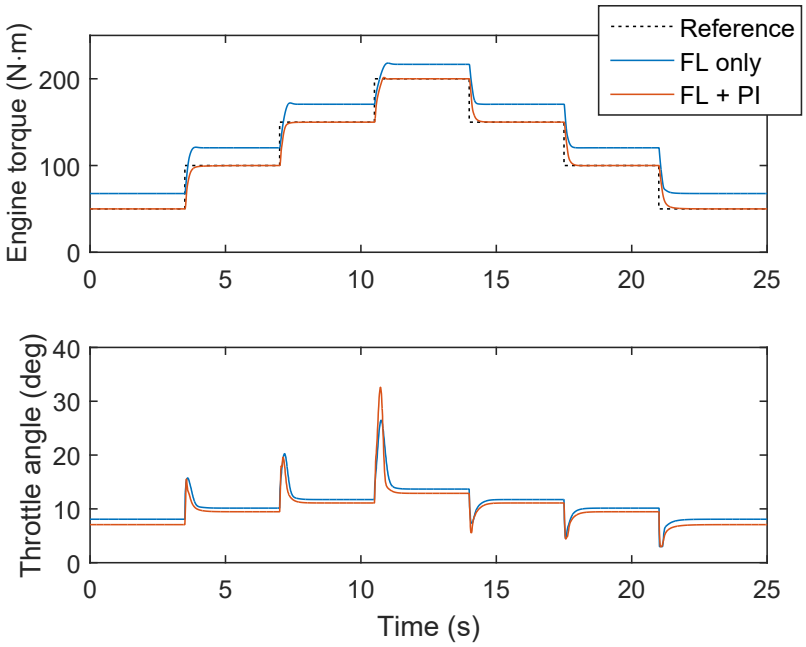


Figure 3.5: Simulation results of implementing the feedback linearization algorithm on the Simscape engine simulator. Torque setpoint changes are relatively small. Engine speed held constant at 2000 RPM. Wastegate held at 0% (fully closed). Blue: feedback linearization only. Red: feedback linearization with supplemental PI control.

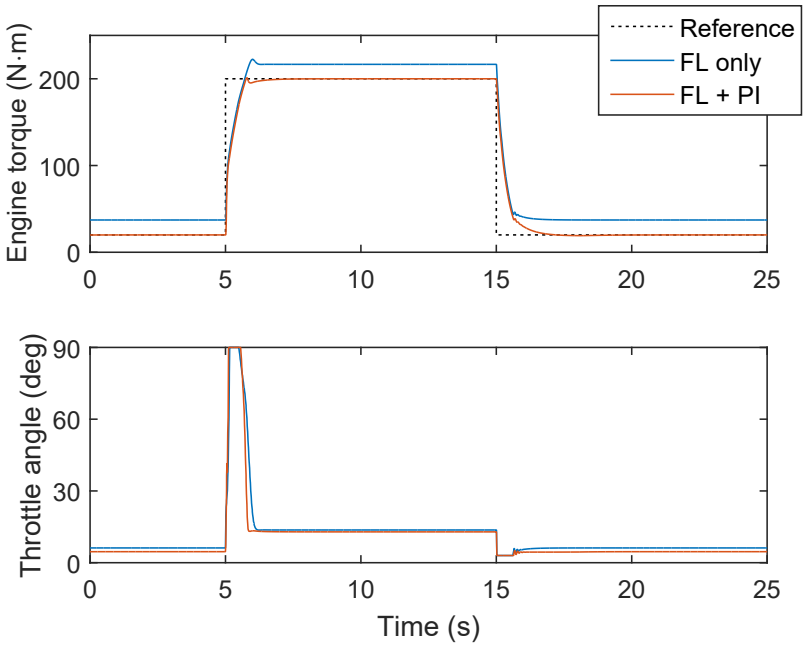


Figure 3.6: Simulation results of implementing the feedback linearization algorithm on the Simscape engine simulator. Torque setpoint changes are relatively large. Engine speed held constant at 2000 RPM. Wastegate held at 0% (fully closed). Blue: feedback linearization only. Red: feedback linearization with supplemental PI control.

3.5 Comparison of performance in time-optimal and fuel-optimal modes

All simulations thus far have been run while holding the wastegate closed. In this section, we compare the performance of the throttle controller in time-optimal and fuel-optimal modes. In the time-optimal mode, the wastegate is held fully closed, so the results are identical to those in the previous simulations. In the fuel-optimal mode, the wastegate is held fully open. The feedback linearization parameter c_0 and the PI gains are kept the same in both cases.

The simulation results show that the throttle controller can perform well in both modes, even though the controller parameters are not changed (Figs. 3.7–3.8). However, there are differences in the torque responses. For increases in the torque setpoint, holding the wastegate open can result in a much slower settling time*(Table 3.2). For decreases in the torque setpoint, holding the wastegate open can result in a slower settling time, but the differences are generally smaller than in the cases of setpoint increases (Table 3.3).

There are also clear differences in the fuel usage. Holding the wastegate open resulted in fuel savings of 1.9% and 2.6% compared with holding the wastegate closed. This is not surprising, since holding the wastegate open is already known to be the fuel-optimal strategy.

Table 3.2: Engine torque settling times for various setpoint increases

	Torque setpoint change (N·m)			
	50 to 100	100 to 150	150 to 200	20 to 200
Wastegate closed	0.55 s	0.39 s	0.39 s	1.07 s
Wastegate open	0.91 s	0.48 s	1.00 s	1.56 s
Difference	0.36 s	0.09 s	0.61 s	0.49 s

Table 3.3: Engine torque settling times for various setpoint decreases

	Torque setpoint change (N·m)			
	200 to 150	150 to 100	100 to 50	200 to 20
Wastegate closed	0.41 s	0.37 s	0.60 s	1.46 s
Wastegate open	0.53 s	0.50 s	0.97 s	1.39 s
Difference	0.12 s	0.13 s	0.37 s	−0.07 s

*The settling time is defined as the time to reach and stay within a 2% error band around the final value. The 2% error band is defined as ±2% of the difference between the initial and final values. For example, consider the case when the initial value is y_i, the final value is y_f, and $y_i < y_f$. Then, the signal is considered to have settled once it reaches and stays within the range of $[y_i + 0.98(y_f - y_i),\ y_i + 1.02(y_f - y_i)]$.

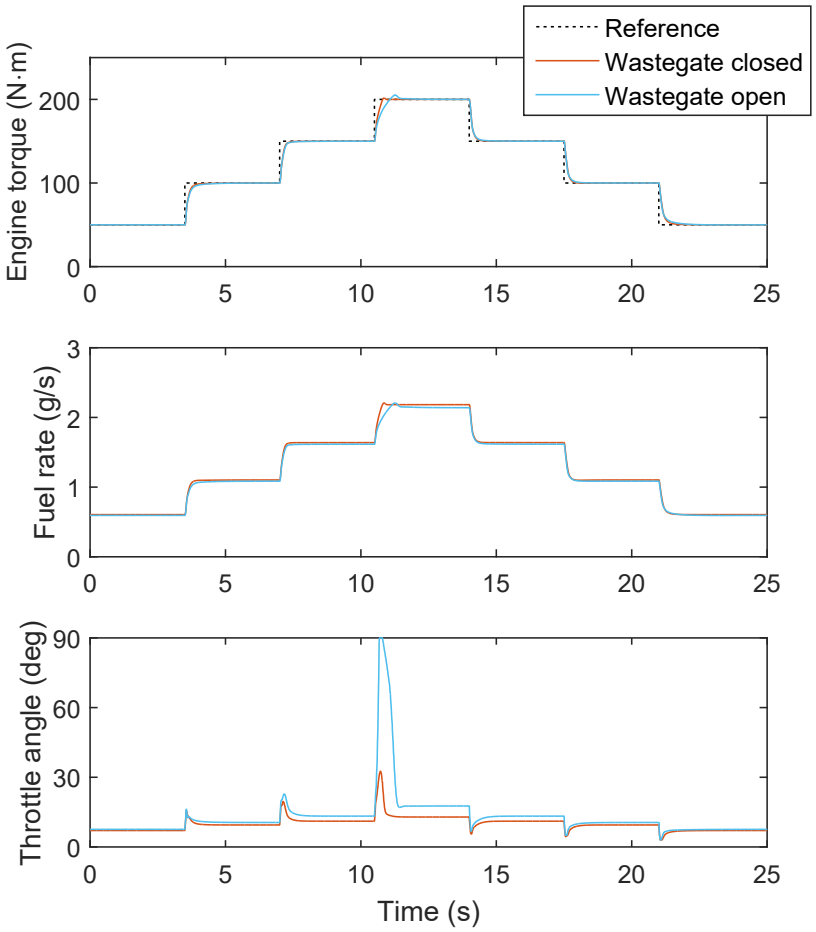


Figure 3.7: Comparison of performance in time-optimal (wastegate closed) and fuel-optimal (wastegate open) modes. Torque setpoint changes are relatively small. Engine speed held constant at 2000 RPM.

Table 3.4: Total fuel usage for simulations in Fig. 3.7

	Fuel mass used	Relative fuel savings with wastegate open
Wastegate closed	31.4 g	1.9%
Wastegate open	30.8 g	

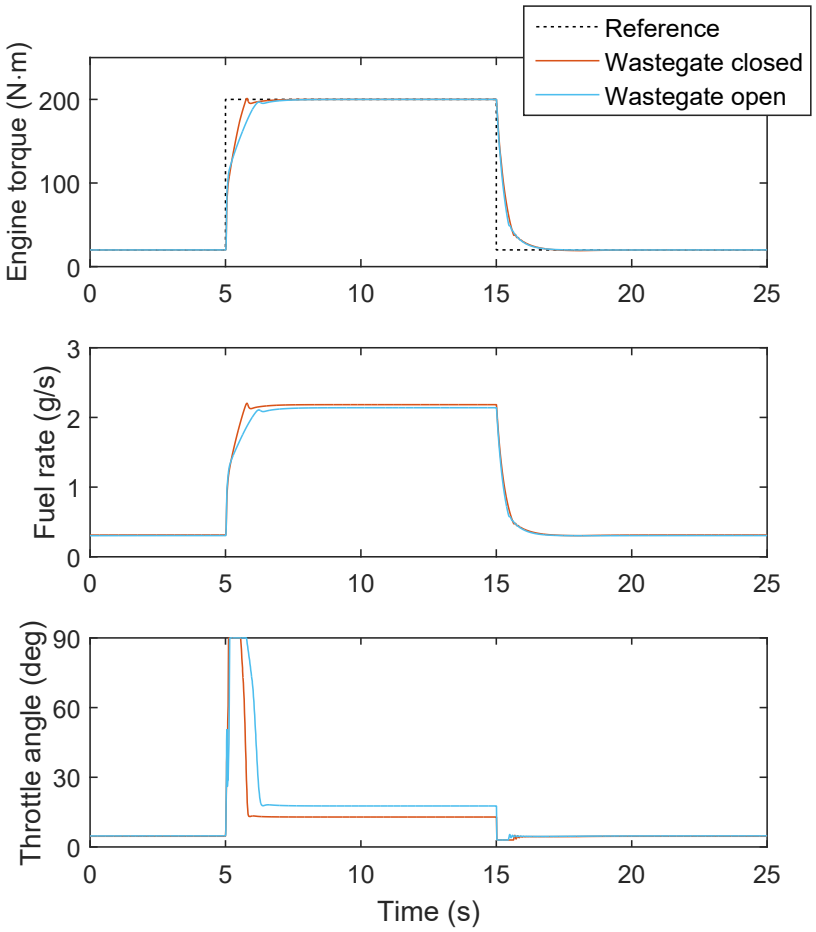


Figure 3.8: Comparison of performance in time-optimal (wastegate closed) and fuel-optimal (wastegate open) modes. Torque setpoint changes are relatively large. Engine speed held constant at 2000 RPM.

Table 3.5: Total fuel usage for simulations in Fig. 3.8

	Fuel mass used	Relative fuel savings with wastegate open
Wastegate closed	26.6 g	2.6%
Wastegate open	25.9 g	

3.6 Chapter summary

In this chapter, a feedback linearization algorithm was designed for the throttle control loop. When applied to the Simscape engine simulator, the feedback linearization was supplemented with PI control to compensate for modeling error. This controller produced good torque tracking for a variety of setpoint changes. The controller parameters were manually tuned to produce good performance in time-optimal (wastegate closed) mode. The controller also performed well in fuel-optimal (wastegate open) mode without any changes to the controller parameters. In the next chapter, the throttle controller will be used to maintain torque tracking while the wastegate switches between open and closed.

Chapter 4

High-Level Mode Switching via Wastegate Actuation

4.1 Chapter overview

This chapter describes the second part of a strategy that separates the throttle and wastegate controls. In this chapter, we investigate a high-level control strategy for the wastegate to switch between fuel-optimal and time-optimal modes, using a preview of the reference torque signal. More specifically, we conduct an extensive simulation study to determine how the timing of the wastegate close-and-open switching affects the overall performance.

The intuitive strategy is to operate in fuel-optimal mode (i.e., wastegate open) when the torque setpoint is constant or decreased, and operate in time-optimal mode (i.e., wastegate closed) when the torque setpoint is increased. Closing the wastegate may also speed up the response for torque decreases. However, as observed in Section 3.5, the difference in settling time between fuel-optimal and time-optimal modes is generally smaller for setpoint decreases than for setpoint increases. Additionally, there are other ways to reduce the engine torque quickly (e.g., shifting to a lower gear) if necessary. Therefore, we focus on determining the wastegate action for torque setpoint increases. We assume that the wastegate is open by default, and it is only closed when an increase in the torque setpoint is detected.

At this point, it is still not exactly clear how to best choose the wastegate timing. The wastegate timing is defined by the answers to the following questions:

1. At what time *before* the setpoint increase should the wastegate be *closed*?
2. At what time *after* the setpoint increase should the wastegate be *opened*?

The best wastegate timing is one that produces a fast torque response without causing excessive overshoot or deviations from the steady-state tracking. It also should not require too much extra fuel. Due to the system nonlinearities, the best wastegate timing may depend on the initial setpoint, the final setpoint, the size of the setpoint change, or some combination of these.

4.2 Design of wastegate timing tests

To investigate how the wastegate timing affects the overall performance, we conducted an extensive simulation study on the Simscape engine simulator. We chose 22 different scenarios in which the setpoint would be increased to a higher value. The setpoint changes range from 10 N·m to 200 N·m, and the initial setpoints were chosen across the engine's available torque range. Each simulation was 10 seconds long, with the setpoint change occuring at the 5-second mark. For each scenario, the following wastegate timing tests were conducted:

- Wastegate held fully closed
- Wastegate held fully open
- All 30 combinations of the following wastegate timings:
  - Wastegate closed at (1, 0.75, 0.5, 0.25, 0.1) seconds before the setpoint change
  - Wastegate opened at (0.1, 0.25, 0.5, 0.75, 1, 2) seconds after the setpoint change

In each simulation, the throttle was controlled using the feedback linearization algorithm with supplemental PI control, as described in Section 3.4. The feedback linearization parameter c_0 and the PI gains were kept the same in all simulations. Additionally, the engine speed was held constant at 2000 RPM for all simulations.

Although our main goal is to strike a balance between fuel usage and settling time, the wastegate timing can cause other side effects on the torque response. During otherwise steady-state tracking, the engine torque may deviate from the desired setpoint due to a change in wastegate position. During step responses, the peak overshoot may increase or decrease depending on the timing of the wastegate opening. These additional effects must be considered as part of the performance. Therefore, to evaluate the overall performance of each test, we constructed a multi-objective cost function that gives a weighted sum of the following performance indicators (see Fig. 4.1 for visual explanation):

- Fuel usage (grams) over the entire 10-second simulation.
- 2% settling time (seconds).
- Peak overshoot as a percentage of the final setpoint value, i.e., $c/r_f \times 100$.
- Pre-step deviation as a percentage of the setpoint change, i.e., $\max(a,b)/(r_f - r_i) \times 100$. This penalizes deviation from steady-state tracking of the initial setpoint due to the wastegate closing.
- Post-step deviation as a percentage of the setpoint change, i.e., $\max(d,e)/(r_f - r_i) \times 100$. This penalizes deviation from steady-state tracking of the final setpoint due to the wastegate opening. If the wastegate is opened before the signal finishes settling, then the post-step deviation is considered to be 0.

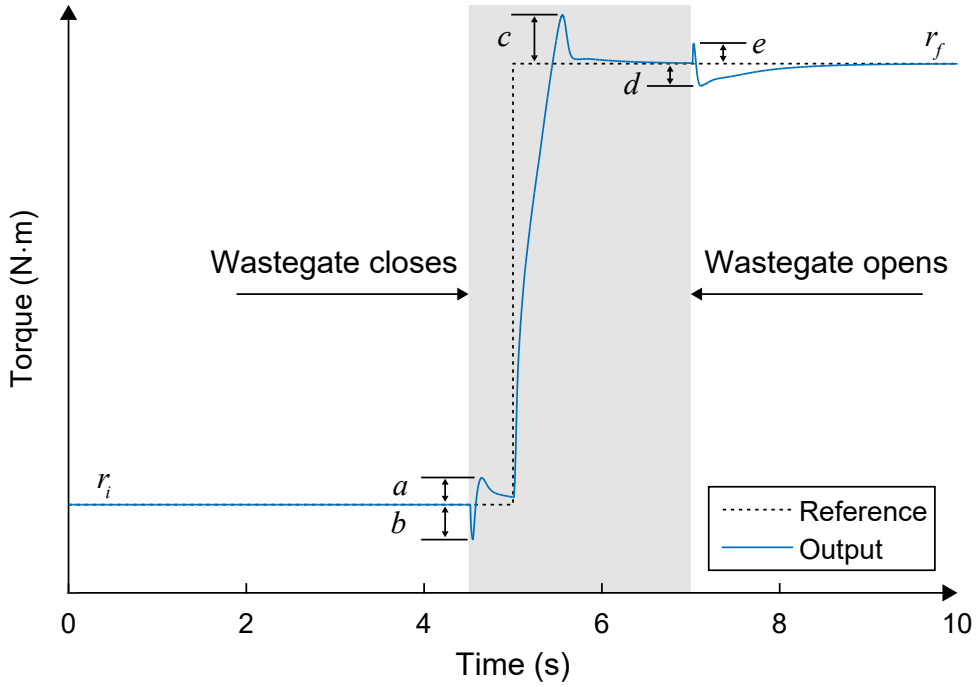


Figure 4.1: Quantities used for calculating performance indicators. The shaded region indicates the time when the wastegate is closed.

The cost function is constructed as:

$$\begin{aligned}\text{cost} = &\left[C_{\text{F}} \times (\text{fuel usage})\right] + \left[C_{\text{ST}} \times (\text{settling time})\right] + \left[C_{\text{PeakOS}} \times (\text{peak overshoot})\right] \\ &+ \left[C_{\text{PreDev}} \times (\text{pre-step deviation})\right] + \left[C_{\text{PostDev}} \times (\text{post-step deviation})\right]\end{aligned}$$

The weights were set to the following values:

$$C_{\text{F}} = 1 \qquad C_{\text{ST}} = 100 \qquad C_{\text{PeakOS}} = 2 \qquad C_{\text{PreDev}} = 2 \qquad C_{\text{PostDev}} = 2$$

To give an idea of the relative importance of each performance indicator in the cost function, the range of values for each indicator was:

$$\begin{aligned}\text{fuel usage} &\in [2, 25] \\ \text{settling time} &\in [0.2, 5] \\ \text{peak overshoot} &\in [0, 50] \\ \text{pre-step deviation} &\in [0, 90] \\ \text{post-step deviation} &\in [0, 110]\end{aligned}$$

4.3 Results of wastegate timing tests

For each setpoint change scenario, we computed the multi-objective cost for each of the 31 wastegate timings to determine which one produced the best performance. The tests with wastegate fully closed were excluded from this process, since the wastegate is assumed to be open by default. However, the results with the wastegate fully closed are still useful for performance comparisons. For some setpoint changes, holding the wastegate fully open produced the best performance. For others, it was more beneficial to close and reopen the wastegate. These results are summarized in Table 4.1. Overall, the best wastegate timings produced fuel savings of between 1% and 3% compared with holding the wastegate closed, and reduced settling times by up to 1 second compared with holding the wastegate open (Table 4.2). Additional figures are included in Appendix B.

Table 4.1: Summary of wastegate timings producing the best performances

Initial setpoint (N·m)	Final setpoint (N·m)	Best time to close wastegate before setpoint increase (s)	Best time to open wastegate after setpoint increase (s)
10	20	$*$	$*$
10	35	1	0.25
10	60	0.5	1
10	85	1	2
10	110	1	2
10	160	1	0.1
10	210	1	0.25
30	40	$*$	$*$
50	60	$*$	$*$
50	75	$*$	$*$
50	100	0.25	2
50	125	0.25	2
50	150	1	2
60	210	0.75	2
100	110	$*$	$*$
100	125	$*$	$*$
100	150	$*$	$*$
110	210	1	0.25
135	210	0.25	0.25
160	210	0.1	0.1
185	210	$*$	$*$
200	210	$*$	$*$

Entries marked $*$ indicate that the best performance was obtained by holding the wastegate open.

Table 4.2: Changes in fuel and settling time resulting from the best wastegate timing for each setpoint change

Initial setpoint (N·m)	Final setpoint (N·m)	Fuel savings relative to holding wastegate closed	Change in settling time relative to holding wastegate open
10	20	2.0%	$*$
10	35	1.8%	-0.87 s
10	60	2.1%	-0.42 s
10	85	1.8%	-0.25 s
10	110	1.5%	-0.26 s
10	160	1.4%	-0.26 s
10	210	2.0%	-0.20 s
30	40	2.9%	$*$
50	60	1.9%	$*$
50	75	2.1%	$*$
50	100	1.4%	-0.38 s
50	125	1.7%	-0.34 s
50	150	1.2%	-0.27 s
60	210	1.2%	-0.45 s
100	110	1.6%	$*$
100	125	2.1%	$*$
100	150	1.5%	$*$
110	210	1.0%	-0.83 s
135	210	1.6%	-0.98 s
160	210	1.2%	-0.66 s
185	210	1.3%	$*$
200	210	1.6%	$*$

Entries marked $*$ indicate that the best performance was obtained by holding the wastegate open.

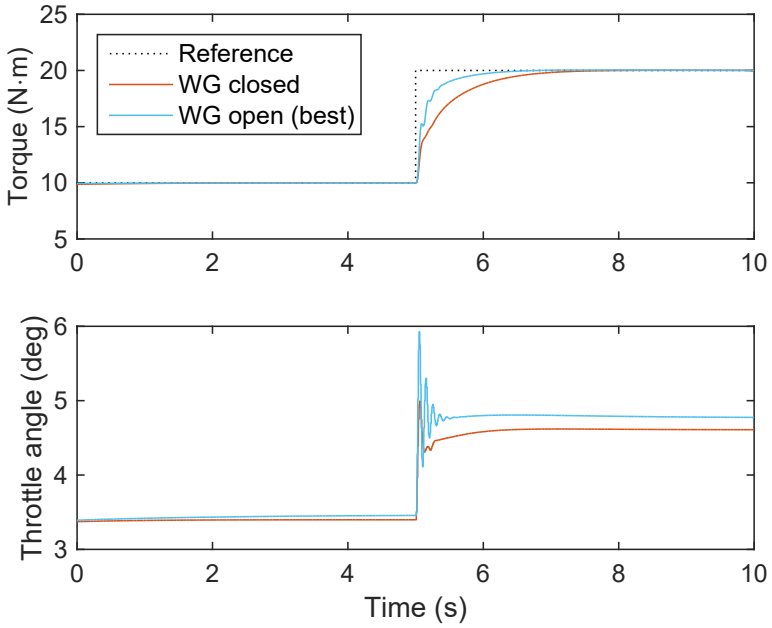


Figure 4.2: Time response for setpoint change from 10 N·m to 20 N·m. Holding the wastegate fully open produces a faster response than holding the wastegate fully closed.

There are a few specific results worth noting. First, for all scenarios with a setpoint change of 10 N·m, the best performance was obtained by holding the wastegate open. This suggests that for small setpoint changes, the improvement in settling time may not be worth the exchange for other negative performance effects.

There are also unusual results at very low torques. For the setpoint change from 10 N·m to 20 N·m, holding the wastegate open actually resulted in a faster time response than holding the wastegate closed (Fig. 4.2). This is because at low torque values, the pumping losses dominate the torque equation (Eq. 2.18). This suggests that at sufficiently low torque values, there is actually no benefit to closing the wastegate.

In several scenarios, the wastegate is only opened after the torque response has settled to the new setpoint. However, for certain setpoint changes, opening the wastegate relatively early can improve both the settling time and the overshoot. One example is the setpoint change from 135 N·m to 210 N·m (Fig. 4.3). When the wastegate is opened, the torque response gets a slight boost from the sudden reduction in pumping losses (due to a sudden drop in exhaust manifold pressure). However, this effect wears off just in time to cancel out the tendency of the throttle controller to overshoot. In this case, the best performance shows less overshoot than holding the wastegate fully closed or fully open.

Unfortunately, there are certain setpoint changes where excessive oscillations are observed in the torque response. One example is the setpoint change from 10 N·m to 35 N·m (Fig. 4.4). The oscillations may be caused by the throttle controller gains being too high for this region.

The performance may be improved by scheduling the throttle controller gains according to the operating conditions. Reducing the PI gains to $K_p = 1$ and $K_i = 3$ for the 10 N·m to 35 N·m setpoint change resulted in significantly reduced oscillations.

Another interesting point is that the absolute pre-step deviations appear to be almost exclusively a function of the initial setpoint, although this relationship is highly nonlinear (Fig. 4.5). Nevertheless, all of the pre-step deviations were under 10 N·m, which may be within acceptable torque tracking bounds.

There also appears to be a strong relationship between the absolute post-step deviations and the final setpoint, although there is slightly more scatter in the data (Fig. 4.6). The post-step deviation can also be recorded as 0 if the wastegate is opened before the torque response settles. Additionally, the post-step deviations appear to be much smaller at final setpoints that are greater than 50 N·m. This may be due to the fact that the engine torque is less sensitive to throttle changes when the throttle is more open, as it is at large torque values.

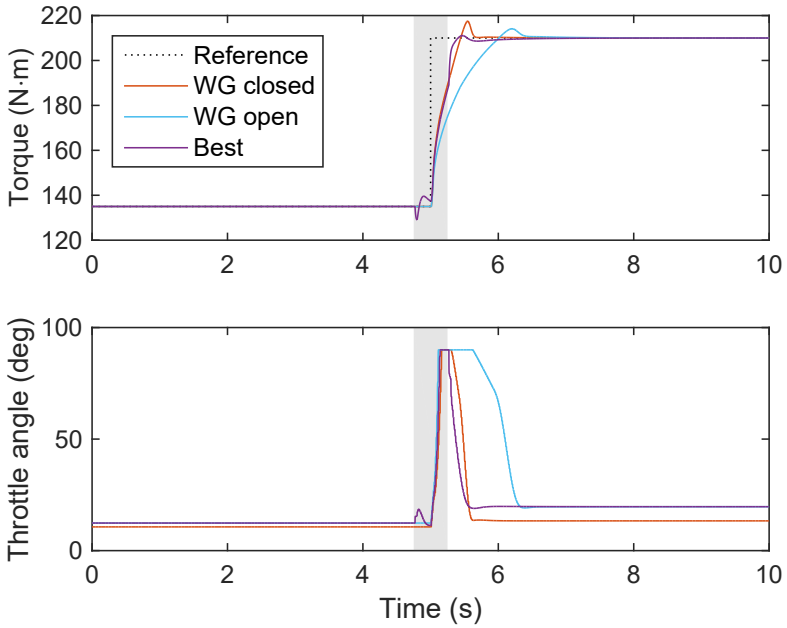


Figure 4.3: Time response for setpoint change from 135 N·m to 210 N·m. The shaded gray region indicates the time period that the wastegate was closed to give the best performance. The torque response gets a slight boost when the wastegate is reopened.

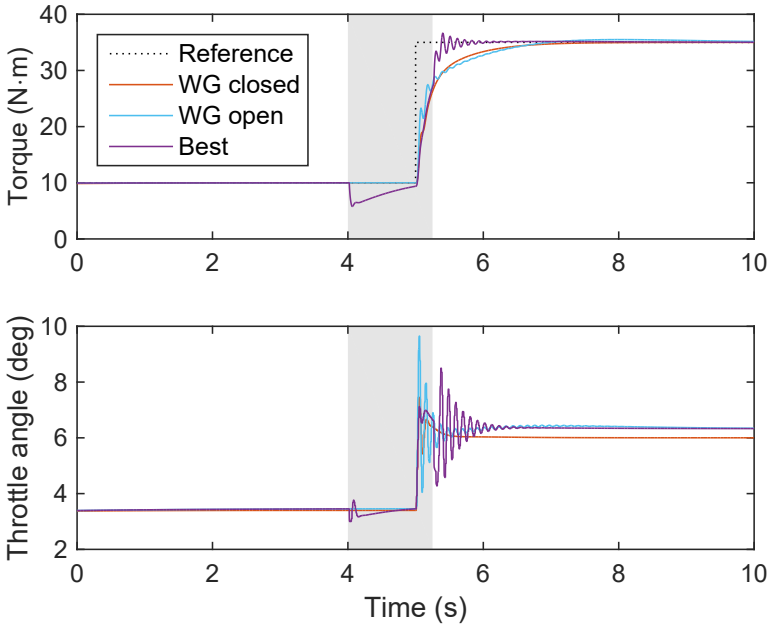


Figure 4.4: Time response for setpoint change from 10 N·m to 35 N·m. The shaded gray region indicates the time period that the wastegate was closed to give the best performance. Excessive oscillations were observed for this setpoint change.

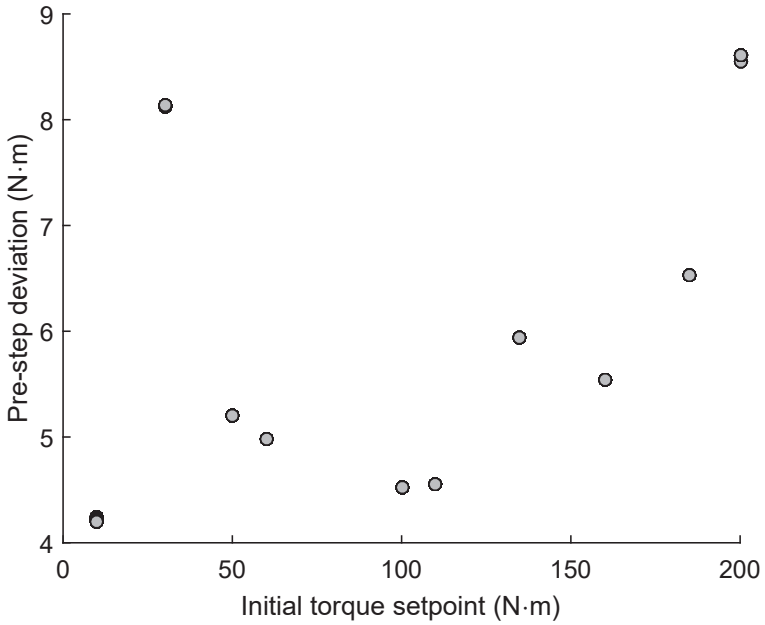


Figure 4.5: Absolute pre-step deviations vs. initial torque setpoints

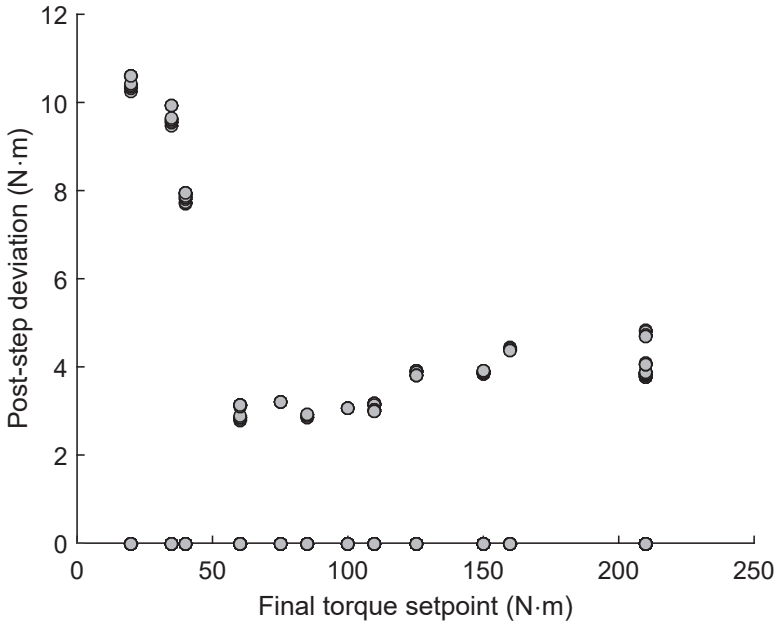


Figure 4.6: Absolute post-step deviations vs. final torque setpoints

4.4 Discussion

The mode-switching strategy for the wastegate potentially offers the best of both fuel-optimal and time-optimal modes. This relatively simple strategy relies on a preview of the reference torque, which may become more readily available as onboard navigation and autonomous driving systems are integrated into passenger vehicles. The results show that just 1 second of preview is sufficient to switch from fuel-optimal to time-optimal mode, and thereby gain a faster response to torque setpoint changes.

Switching the wastegate between open and closed may cause some undesired side effects in the torque response. Most notably, closing the wastegate causes a slight deviation in the torque setpoint tracking. This may be mitigated by adding closed-loop control on the wastegate position. However, the deviation observed in our simulation was relatively small, and may be within acceptable torque tracking bounds.

The throttle controller developed in Chapter 3 performed generally well during wastegate switching. However, there were a few scenarios where excessive oscillations were observed. This may be mitigated by reducing the throttle controller gains in the operating region that is sensitive to oscillations. This seems to be a relatively small region, as oscillations were only observed in a small number of scenarios.

Finally, additional work is still needed to extend the wastegate control strategy to the entire operating space.

4.5 Chapter summary

In this chapter, an extensive simulation study was conducted on the Simscape engine simulator to determine how the timing of the wastegate close-and-open switching affects the overall engine performance. The results can be used to construct a controller that closes and opens the wastegate at the appropriate times before and after a setpoint change. This strategy relies on a preview of the reference torque, so that the wastegate timing can be determined from the initial and final setpoints. The engine can thus run in a fuel efficient mode when the torque setpoint is constant, while still producing a fast torque response when a setpoint increase is required.

Chapter 5

Discrete-Time Turbocharged Engine Model*

5.1 Chapter overview

Thus far, the 4-state model from Section 2.4 has been used for controller design. This model is designed in continuous time, but it is more beneficial to use a discrete-time model in certain control strategies. For example, optimization methods are much easier to handle in discrete time. There is also some evidence that accounting for the discrete sample time during the controller design can reduce errors from digital implementation [25]. For these reasons, we seek to convert the 4-state model to discrete time.

In this chapter, we describe and analyze the numerical issues that result when the 4-state model is discretized. We then present a solution by representing the throttle pressure ratio as a static map, which also allows one state to be removed. The resulting 3-state model is easily converted to discrete time.

5.2 Issues in conversion to discrete time

The Euler method is a common technique for converting nonlinear dynamic models to discrete time, due to its ease of formulation and implementation. It is equivalent to using a first-order Taylor approximation of the continuous-time dynamic equation to estimate the state evolution:

$$x(t_0 + t_s) \approx x(t_0) + t_s \dot{x}(t_0)$$

where t_s is the step size. Given the differential equation in Eq. 2.17, the discretized dynamic system is then expressed as:

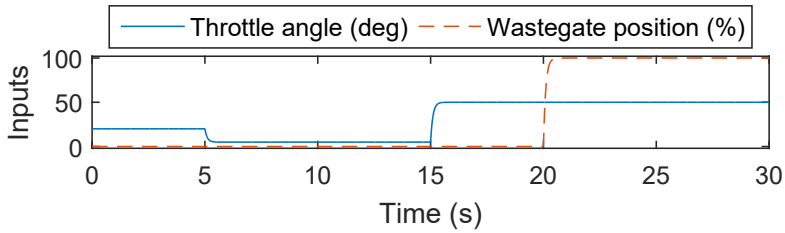
$$\begin{aligned} x_{k+1} &= x_k + t_s f(x_k, u_k) \\ &= f_d(x_k, u_k) \end{aligned}$$

where $x_k = x(kt_s)$ and $u_k = u(kt_s)$.

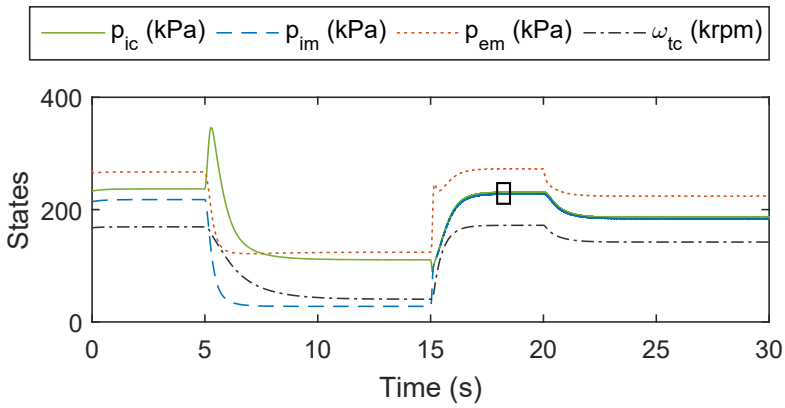
*This chapter contains significant portions from [22], reprinted with permission from IEEE.

It is well known that the Euler method can be numerically unstable for stiff models. Stiff models are usually characterized as having at least one dynamic mode with a substantially faster time constant than the other modes. This fast mode, when combined with a large step size, can cause inaccuracy or even instability. This problem can be mitigated by reducing the step size, but this is not always practical. A natural choice of step size is the expected controller sampling period. For automotive engine systems, the controller sampling period is typically around 10 ms [26]. If a multi-rate scheme is used, it may be permissible to use a step size no smaller than 1 ms.

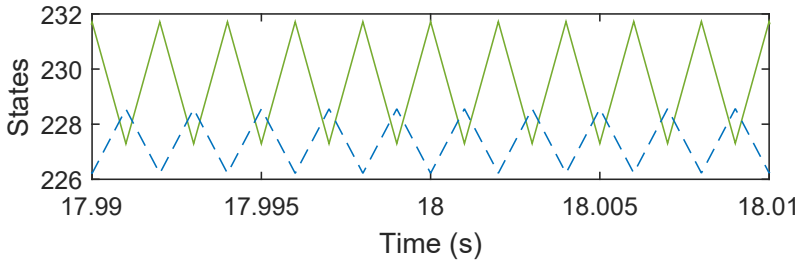
We use the Euler method to discretize the 4-state engine model described in Section 2.4. When the model is discretized with a 1 ms step size, significant chattering occurs at wide-open throttle



(a) Input signals

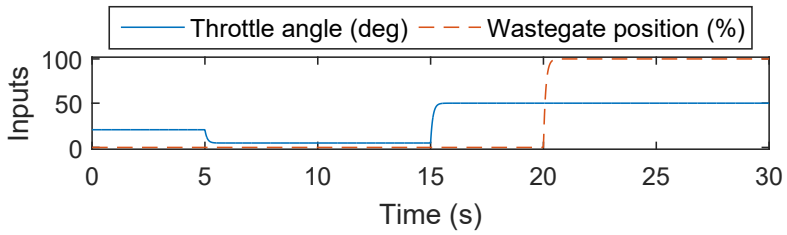


(b) Discretized 4-state model (see below for closeup of chattering in box)

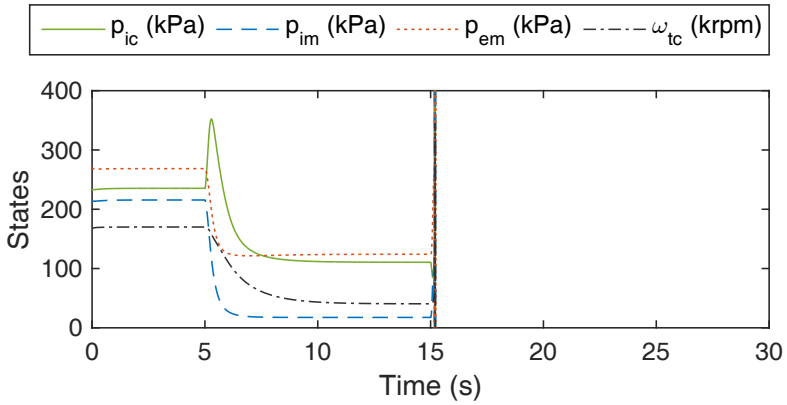


(c) Closeup of chattering

Figure 5.1: Visible chattering in simulation of discretized 4-state model, with 1 ms step size (© 2016 IEEE)



(a) Input signals



(b) Discretized 4-state model goes unstable when throttle is opened

Figure 5.2: Numerical instability in simulation of discretized 4-state model, with 10 ms step size

(Fig. 5.1). When a 10 ms step size is used, the discretized model becomes unstable soon after the throttle is opened around the 15 second mark (Fig. 5.2). It should be clarified that the chattering and instability are purely due to numerical error after discretization, and do not occur in continuous-time simulations.

To explain this behavior, we must examine a unique characteristic of turbocharged engines with throttle valves (typically gasoline engines). In these engines, the pressure upstream of the throttle is generally above ambient pressure, and can vary significantly with varying operating conditions. This behavior is not present in naturally aspirated engines, since the pressure upstream of the throttle is close to ambient and usually does not vary significantly. It is also not present in turbocharged engines equipped with variable-geometry turbochargers, since they usually are not equipped with throttle valves. It is the combination of turbocharger and throttle characteristics that results in chattering after discretization.

Under partial-throttle conditions in these engines, there is usually a substantial pressure difference across the throttle, meaning that the throttle pressure ratio $\Pi_{th} = p_{im}/p_{ic}$ is well below 1. From a physical intuition, this pressure difference provides the driving force for flow to move through the valve. If the pressures on both sides of the valve were equal ($\Pi_{th} = 1$), then there would be zero flow. This is consistent with the valve model in Eqs. 2.1 and 2.13.

Since the pumping action of the engine always draws some flow out of the intake manifold, Π_{th} will always be less than 1 in the real system. At most, it may get very close to 1 when the throttle

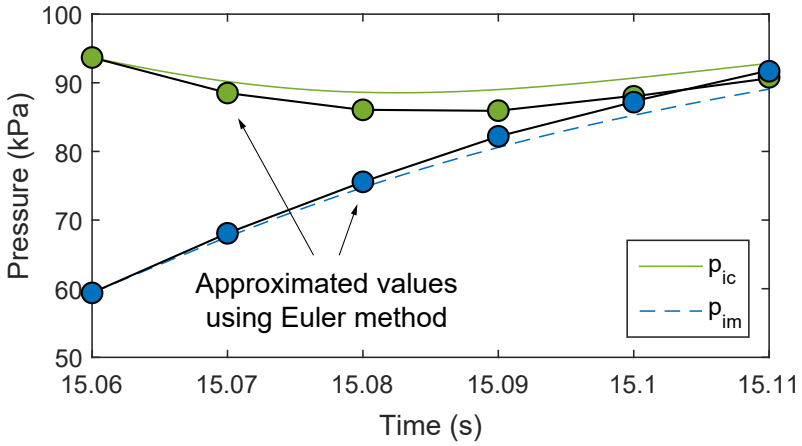


Figure 5.3: Illustration of numerical error in Euler method (© 2016 IEEE)

is wide open. In simulation, however, the numerical error may push Π_{th} above 1. Therefore, it is wise to include code that defines the throttle flow behavior for $\Pi_{th} > 1$. There are two obvious methods to do so. The first is to simply set the throttle mass flow rate $\dot{m}_{th}$ equal to 0 for $\Pi_{th} > 1$, effectively making the throttle a one-way valve. The second is to include a model of reverse flow, following a similar structure as Eq. 2.13.

Applying the Euler method to a dynamic model with this throttle behavior can lead to chattering, especially at wide-open-throttle conditions. When the throttle is wide open, Π_{th} approaches 1. Because of numerical error, p_{im} then crosses above p_{ic}, causing $\Pi_{th} > 1$ (Fig. 5.3). At the next time step, Π_{th} will fall back below 1. This is obvious in the reverse flow model, but it also occurs in the zero flow model since the engine's pumping action will decrease the intake manifold pressure. The resulting back-and-forth crossing of 1 causes chattering and sometimes even instability.

It is unclear if an alternate discretization method can mitigate this problem. Using the common "RK4" Runge-Kutta method to discretize the model also showed similar chattering behavior. If the continuous-time model is simulated using the ode45 solver, a variable-step method based on RK4, the step size is automatically chosen to be around 1 ms after the throttle is opened (Fig. 5.4). This suggests that even more accurate discretization methods may not be able to obtain good performance at step sizes larger than 1 ms.

Another possibility could be to use an implicit method, such as the backward Euler method:

$$x_{k+1} = x_k + t_s f(x_{k+1}, u_{k+1})$$

However, this may be impractical since the complexity of the engine model makes it difficult to solve for x_{k+1}.

It should also be noted that some alternative models may not suffer from chattering after discretization. For example, the model presented in [27] eliminates the dependence on pressure ratio at wide-open conditions. This option has not been investigated.

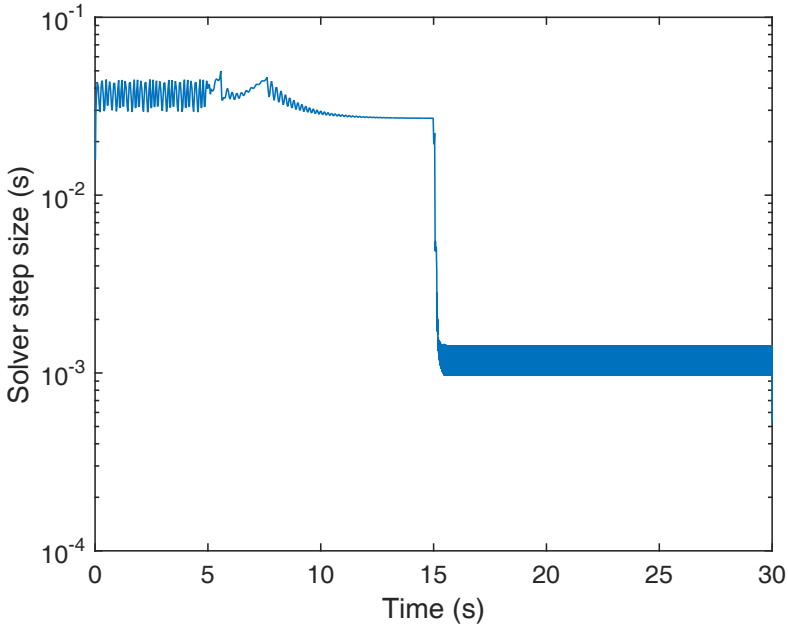


Figure 5.4: Step size chosen by ode45 solver

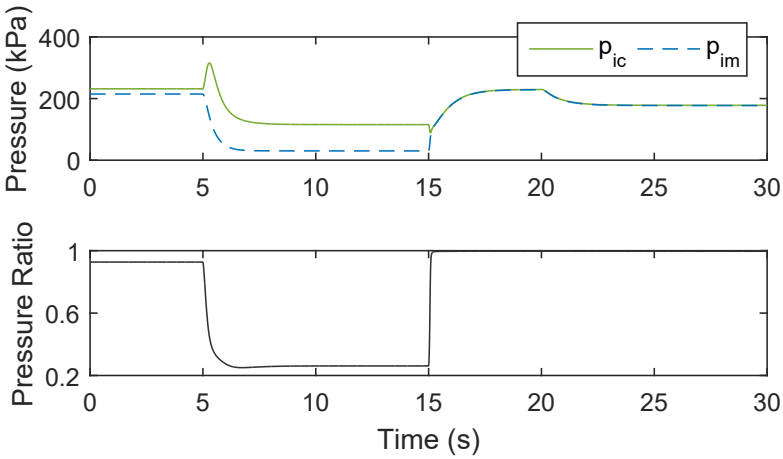


Figure 5.5: Time response of throttle pressure ratio compared with absolute pressures (© 2016 IEEE)

5.3 3-state reduced model for discretization

Since the chattering and instability in the discretized model is caused by the throttle pressure ratio crossing 1, we now analyze the pressure ratio dynamics further. Looking closely at the continuous-time simulation of the 4-state model, we can see that Π_{th} responds much faster to throttle step changes than p_{ic} and p_{im} (Fig. 5.5).

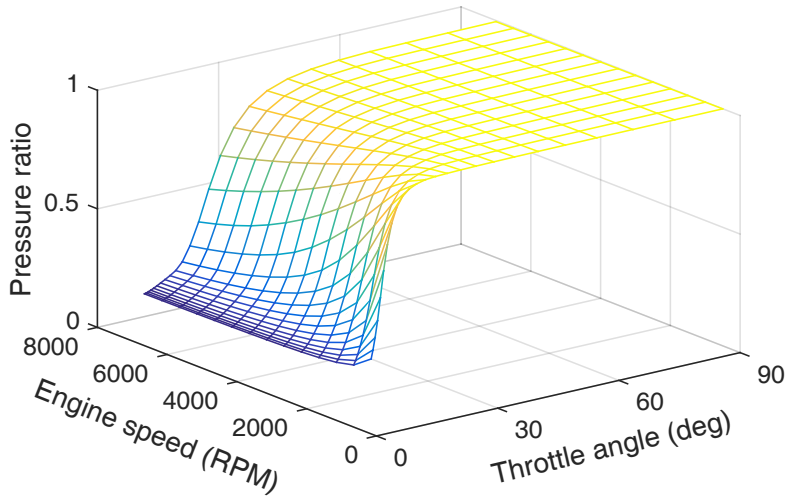


Figure 5.6: Steady-state throttle pressure ratio map (© 2016 IEEE)

According to conventional modeling wisdom, we can replace fast modes with static functions to reduce the stiffness of the model. Thus, we created a map of the throttle pressure ratio as a function of throttle angle and engine speed (Fig. 5.6). The throttle angle grid points were chosen as 25 points logarithmically spaced between 3 and 90 deg, while the engine speed grid points were chosen as 8 points evenly spaced between 500 and 7000 RPM. The pressure ratio value at each grid point was computed by simulating the continuous-time model using the ode23s solver in MATLAB, with 60 seconds of simulation time to let the system to reach steady state. We observed that the wastegate position had a negligible effect on the steady-state values (Fig. 5.7), regardless of the engine speed, so the final map values were computed with the wastegate held at 0% (fully closed). Linear interpolation can be used between grid points to give a static function of $\Pi_{th}(\theta_{th}, \omega_e)$.

We can now reduce the model by removing p_{im} as a state variable. This leaves $x = [\, p_{ic} \;\; p_{em} \;\; \omega_{tc} \,]^{\mathrm{T}}$ as the state vector, still evolving dynamically according to Eq. 2.15, while the intake manifold pressure is modeled as:

$$p_{im} = p_{ic} \cdot \Pi_{th}(\theta_{th}, \omega_e)$$

This 3-state model is now evaluated for accuracy and ease of discretization. To evaluate accuracy, we simulate the continuous-time versions of the 3-state and 4-state models using the variable-step solver ode23s. The simulations show that the 3-state model is roughly as accurate as the 4-state model, relative to the true behavior (Fig. 5.8). To evaluate ease of discretization, we apply the Euler method to the 3-state model. Since the 3-state model uses a static map of throttle pressure ratio, and all function values in the map are less than 1, chattering is eliminated for both 1 ms and 10 ms step sizes. In fact, simulating the 3-state model in continuous time and in discrete time (with a 10 ms step size) produces virtually indistinguishable results (Fig. 5.9). Thus, the 3-state model is both accurate and easily discretized, making it well-suited for discrete-time control design.

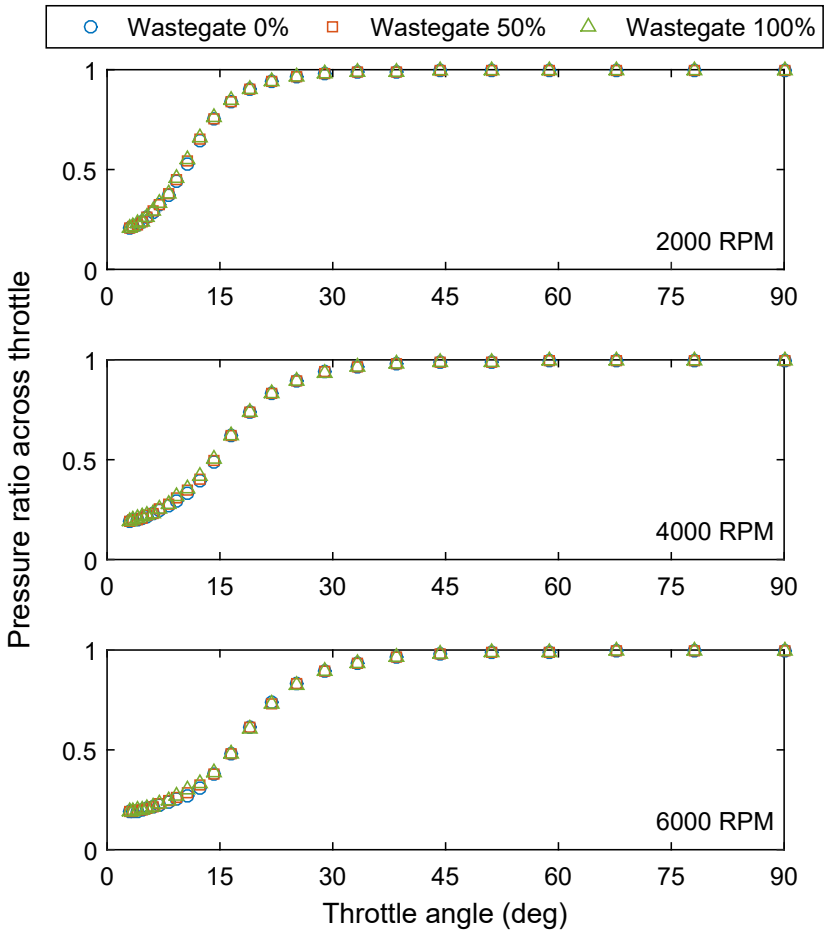


Figure 5.7: Wastegate position has a negligible effect on the steady-state throttle pressure ratio, regardless of engine speed.

5.4 Chapter summary

In this chapter, the model of the turbocharged engine was converted to discrete time. The original 4-state model suffered from undesirable chattering after discretization. To fix this problem, we eliminated a fast mode corresponding to the throttle pressure ratio. The pressure ratio is instead represented as a static function of the throttle position and engine speed, and is mapped from a grid of steady-state values. The resulting 3-state model has a comparable performance to the 4-state model, and is also easily discretized. This 3-state model will be used for the optimization scheme in the next chapter.

Replacing a fast dynamic mode with a static function is fairly common when dealing with stiff and singular perturbation. The methods in this chapter may be extended to other stiff systems that need to be modeled in discrete time. The key difficulty is identifying the fast dynamic, which may not be immediately obvious.

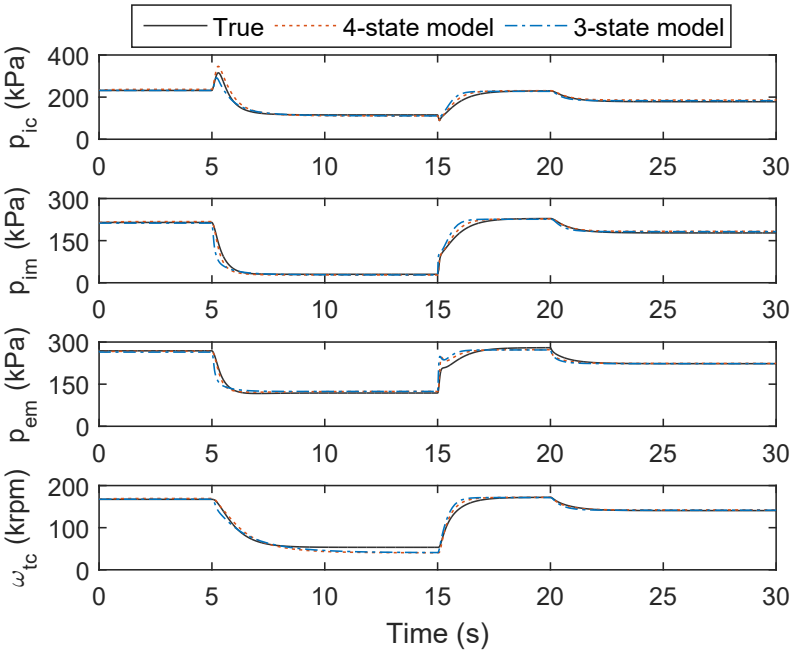


Figure 5.8: Comparison of continuous-time 3-state model, continuous-time 4-state model, and true behavior (© 2016 IEEE)

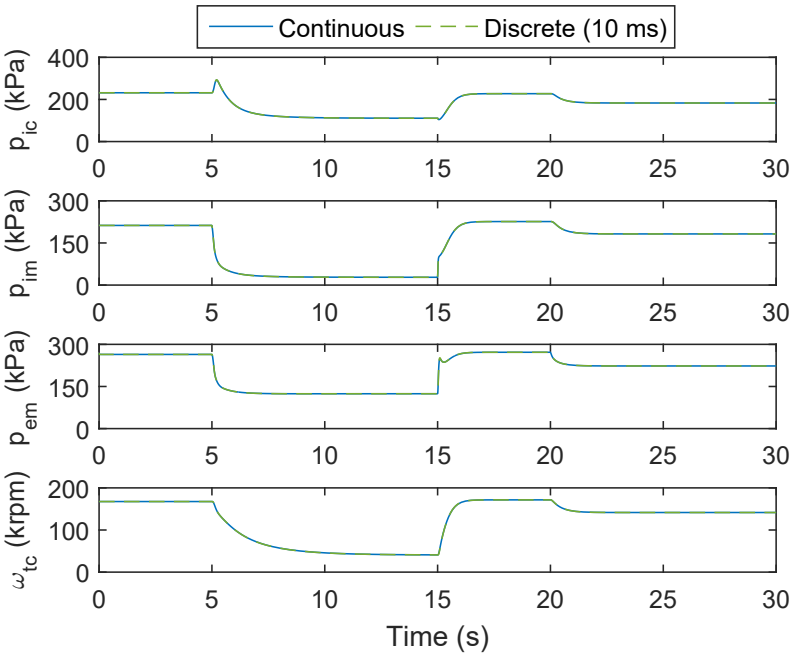


Figure 5.9: Continuous-time vs. discrete-time (10 ms step size) simulations of 3-state model (© 2016 IEEE)

Chapter 6

Multi-Objective Optimization

6.1 Chapter overview

This chapter describes a multi-objective optimization scheme to coordinate the throttle and wastegate simultaneously. The discrete-time 3-state model from Chapter 5 is used to formulate constraints on the state dynamics. Since the nonlinear state constraints make the problem non-convex, the problem is "convexified" by linearizing the dynamics over the system trajectory. This results in a quadratic program, which is solved iteratively. A multi-objective cost function is used to trade off between torque tracking and fuel usage. The cost function is formulated in three different ways, and the results are compared with each other.

The discrete-time model can be expressed as:

$$\begin{aligned} x_{k+1} &= f_d(x_k, u_k, w_k) \\ y_k &= h(x_k, u_k, w_k) \end{aligned} \tag{6.1}$$

where x is the state, u is the input, y is the output, and w is the vector of measurable parameters. In this chapter, we consider both the throttle and wastegate positions as the inputs. The outputs are the engine torque and the fuel mass flow rate. As in Chapter 3, the measurable parameters are the engine speed and spark timing, and the states are scaled to provide better numerical conditioning. The fuel mass flow rate is also scaled.

The state x, input u, output y, and parameter w are summarized as:

$$x = \begin{bmatrix} p_{ic} \cdot 10^{-3} \\ p_{em} \cdot 10^{-3} \\ \omega_{tc} \cdot 10^{-3} \end{bmatrix} \qquad u = \begin{bmatrix} \theta_{th} \\ u_{wg} \end{bmatrix} \qquad y = \begin{bmatrix} \tau_e \\ \dot{m}_f \cdot 10^3 \end{bmatrix} \qquad w = \begin{bmatrix} \omega_e \\ SA \end{bmatrix}$$

6.2 Non-convex optimization formulation

Recall that our goal is to track a reference trajectory with as little fuel usage as possible. Let $Y_{1d} = \{y_{1d,k}\}_{k=0}^{p}$ denote the sequence of desired torque values, with length p. Then, let $U = \{u_k\}_{k=0}^{p}$,

$X = \{x_k\}_{k=0}^{p}$, and $Y = \{y_k\}_{k=0}^{p}$ repectively denote the sequence of the model inputs, states, and outputs over this same length. Via choice of U, we would like to minimize the error between Y_1 and Y_{1d}, while also minimizing Y_2 in some way. A simple way to combine these objectives is with a weighted-sum cost function.

One obvious choice for the cost function is:

$$J_1(Y) = c_\tau \|Y_{1d} - Y_1\|_2^2 + c_f \|t_s Y_2\|_2^2 \tag{6.2}$$

where t_s is the discrete-time step size. Note that $\|t_s Y_2\|_2^2 = \sum_{k=0}^{p} (t_s y_{2,k})^2$, which is analogous to the total fuel usage. Using this cost function will directly attempt to minimize the torque tracking error and the total fuel usage.

Instead of directly minimizing the fuel usage, an alternative would be to maximize the wastegate position. This exploits the knowledge that opening the wastegate results in lower fuel usage. The corresponding cost function would look like:

$$J_2(U,Y) = c_\tau \|Y_{1d} - Y_1\|_2^2 + c_f \left\|u_{2,\max} - U_2\right\|_2^2 \tag{6.3}$$

In this case, we attempt to minimize the wastegate deviation from a wide-open position.

A third option is similar to (6.3), except that a 1-norm is used on the wastegate term:

$$J_3(U,Y) = c_\tau \|Y_{1d} - Y_1\|_2^2 + c_f \left\|u_{2,\max} - U_2\right\|_1 \tag{6.4}$$

Because using the 1-norm tends to produce sparse results, this formulation may result in the wastegate staying wide open for most of the time.

We can formulate our optimization problem as:

$$\begin{aligned} \min_{U,X,Y} \quad & J(U,Y) \\ \text{s.t.} \quad & x_{k+1} = f_d(x_k, u_k, w_k) && k = 0,\ldots,p-1 \\ & y_k = h(x_k, u_k, w_k) && k = 0,\ldots,p \\ & u_{\min} \le u_k \le u_{\max} && k = 0,\ldots,p \\ & 0 \le x_k \le x_{\max} && k = 0,\ldots,p \\ & y_k \ge 0 && k = 0,\ldots,p \\ & x_0 = x_i \end{aligned} \tag{6.5}$$

where $J(U,Y)$ is one of the cost functions from (6.2–6.4). The first two constraints ensure that X and Y obey the nonlinear dynamics. The next three constraints enforce box constraints on the input, state, and output. The states are given an upper bound to ensure they stay within reasonable limits. The last constraint enforces the initial state to be equal to the specified initial condition x_i.

Unfortunately, the optimization problem in (6.5) is not convex, due to the nonlinear equality constraints. While some methods exist for solving non-convex problems, their performance is highly dependent on the initial guess and the degree of non-convexity. Several attempts were made to solve (6.5) using the fmincon solver in MATLAB, with both the interior-point and sequential-quadratic-programming algorithms. However, even after several days of running, the solver could not converge on a solution for any of the three cost function formulations.

6.3 Convex optimization with iterative linearization

Instead of attempting to solve the non-convex problem directly, one alternative is to "convexify" the problem by relaxing the non-convex components. In our case, the non-convexity stems from the nonlinear state and output constraints. Inspired by the approach in [20, 21], we can convexify these constraints by linearizing around the system trajectory. The optimization problem can then be reformulated into a quadratic program (QP), which is easily solved with any standard QP solver. The solution of the QP gives an update to the input sequence. Using the updated input sequence, we generate a new trajectory and linearize again. This process is repeated until the input sequence converges.

Let $\bar{U} = \{\bar{u}_k\}_{k=0}^{p}$ denote a candidate solution to (6.5). Then, let $\bar{X} = \{\bar{x}_k\}_{k=0}^{p}$ and $\bar{Y} = \{\bar{y}_k\}_{k=0}^{p}$ be the sequences of states and outputs, respectively, generated by the nonlinear model (6.1) with the input sequence $\bar{U}$ and initial state $\bar{x}_0 = x_i$. The nonlinear model can then be linearized around $(\bar{X}, \bar{U})$ to produce a linear time-varying system:

$$\begin{aligned} \delta x_{k+1} &= A(\bar{x}_k, \bar{u}_k, \bar{w}_k)\delta x_k + B(\bar{x}_k, \bar{u}_k, \bar{w}_k)\delta u_k \\ \delta y_k &= C(\bar{x}_k, \bar{u}_k, \bar{w}_k)\delta x_k + D(\bar{x}_k, \bar{u}_k, \bar{w}_k)\delta u_k \end{aligned}$$

where δu_k, δx_k, and δy_k are the perturbations from $\bar{u}_k$, $\bar{x}_k$, and $\bar{y}_k$, respectively. A, B, C, and D are time-varying Jacobian matrices, defined by:

$$A(\bar{x}_k, \bar{u}_k, \bar{w}_k) = \left.\frac{\partial f_d}{\partial x}\right|_{(\bar{x}_k, \bar{u}_k, \bar{w}_k)} \qquad B(\bar{x}_k, \bar{u}_k, \bar{w}_k) = \left.\frac{\partial f_d}{\partial u}\right|_{(\bar{x}_k, \bar{u}_k, \bar{w}_k)}$$

$$C(\bar{x}_k, \bar{u}_k, \bar{w}_k) = \left.\frac{\partial h}{\partial x}\right|_{(\bar{x}_k, \bar{u}_k, \bar{w}_k)} \qquad D(\bar{x}_k, \bar{u}_k, \bar{w}_k) = \left.\frac{\partial h}{\partial u}\right|_{(\bar{x}_k, \bar{u}_k, \bar{w}_k)}$$

We should note that the discrete-time model contains the static map for the throttle pressure ratio, which is not easily differentiated. Therefore, we calculate estimates of A, B, C, and D using the numerical differentiation algorithm from [28].

The optimization problem from (6.5) can then be reformulated as a QP:

$$\begin{aligned} \underset{\delta U, \delta X, \delta Y}{\text{min.}} \quad & J(U,Y) + c_{u_1}\|\delta U_1\|_2^2 + c_{u_2}\|\delta U_2\|_2^2 && && (6.6) \\ \text{s.t.} \quad & \delta x_{k+1} = A(\bar{x}_k, \bar{u}_k, \bar{w}_k)\delta x_k + B(\bar{x}_k, \bar{u}_k, \bar{w}_k)\delta u_k && k = 0, \ldots, p-1 \\ & \delta y_k = C(\bar{x}_k, \bar{u}_k, \bar{w}_k)\delta x_k + D(\bar{x}_k, \bar{u}_k, \bar{w}_k)\delta u_k && k = 0, \ldots, p \\ & u_{\min} \le \bar{u}_k + \delta u_k \le u_{\max} && k = 0, \ldots, p \\ & 0 \le \bar{x}_k + \delta x_k \le x_{\max} && k = 0, \ldots, p \\ & \bar{y}_k + \delta y_k \ge 0 && k = 0, \ldots, p \\ & \delta x_0 = 0 && \\ & |\delta u_{1,k}| \le \varepsilon_{u_1} \cdot (\bar{u}_{1,k}/u_{1,\max}) && k = 0, \ldots, p \\ & |\delta u_{2,k}| \le \varepsilon_{u_2} && k = 0, \ldots, p \end{aligned}$$

where $\delta U = U - \bar{U}$, $\delta X = X - \bar{X}$, and $\delta Y = Y - \bar{Y}$. Note that two additional terms have been added to the cost function. These terms penalize the deviation δu_k from $\bar{u}_k$ to improve the convergence of the algorithm. The values of c_{u_1} and c_{u_2} were both set to 0.01.

Two additional trust-region constraints have also been added to ensure accuracy of the linearized model. The trust-region constraint on $\delta u_{1,k}$ is tightened at small values of $\bar{u}_{1,k}$, since the torque is much more sensitive to throttle changes when the throttle is near closed. The value of ε_{u_1} was set to 8, while the value of ε_{u_2} was set to 5.

The overall algorithm is as follows:

Initialize $\bar{U}$ ▷ Initialize candidate
while $\|\delta U_1\|_2 \geq \Delta_1$ **and** $\|\delta U_2\|_2 \geq \Delta_2$ **do**
  $\bar{X} = \text{SimSys}(x_i, \bar{U})$ ▷ Simulate nonlinear system (6.1)
  $(A, B, C, D) = \text{LinSys}(\bar{U}, \bar{X})$ ▷ Linearize system
  $\delta U = \text{SolveQP}(A, B, C, D, x_i, \bar{U})$ ▷ Solve QP (6.6)
  $\bar{U} = \bar{U} + \delta U$ ▷ Update candidate
end while

The function SimSys$(\cdot)$ generates the state sequence $\bar{X}$ from the nonlinear model (6.1). The function LinSys$(\cdot)$ computes the sequences of the A, B, C, and D Jacobian matrices by linearizing the nonlinear model around $(\bar{X}, \bar{U})$. The function SolveQP$(\cdot)$ solves (6.6) to produce $\delta U = \{\delta u_k\}_{k=0}^{p}$, the update to the input sequence. The algorithm terminates when $\|\delta U_1\|_2$ and $\|\delta U_2\|_2$ are sufficiently small, indicating convergence. The values of Δ_1 and Δ_1 were both set to 10.

6.4 Optimization results

In each iteration of the algorithm, we solve the corresponding QP with MOSEK. This software is used in a YALMIP framework [29], which runs in MATLAB.

Since it was not obvious which of the three cost function formulations (6.2–6.4) would produce the best results, all three were tested. The performance results for the three cost function formulations are summarized in Table 6.1, along with the values of the c_{tau} and c_f parameters. Each of the three produced reasonably good results. J_1 produced the best torque tracking, but also used the most fuel. J_2 produced the worst tracking, but used the least fuel. J_3 produced a compromise in between J_1 and J_2.

The three cost function formulations show clear differences in the resulting input sequences (Fig. 6.1). For J_3, the wastegate closes around the 0.5-second mark (when the setpoint increases), and remains closed until around the 0.6-second mark. This behavior closely resembles the wastegate timings tested in Chapter 4. J_2 produced a similar wastegate trajectory as J_1, but the wastegate does not close all the way and it stays partially open for a longer period of time. For J_3, the wastegate remains closed before the setpoint change, but opens up arounod the 0.6-second mark. This gives the torque response a quick boost due to the sudden reduction in pumping loss. This is an effect that was also observed in Chapter 4.

Table 6.1: Optimization parameters and results for three different cost function formulations

Cost function	c_τ	c_f	Torque error (N·m) $\|Y_{1d} - Y_1\|_2$	Fuel usage (g) $\sum_{k=0}^{p} t_s y_{2,k}$
$J_1(Y) = c_\tau \|Y_{1d} - Y_1\|_2^2 + c_f \|t_s Y_2\|_2^2$	1	500	91.5	2.10
$J_2(U,Y) = c_\tau \|Y_{1d} - Y_1\|_2^2 + c_f \|u_{2,\max} - U_2\|_2^2$	5	1	172.5	2.00
$J_3(U,Y) = c_\tau \|Y_{1d} - Y_1\|_2^2 + c_f \|u_{2,\max} - U_2\|_1$	1	8	137.3	2.03

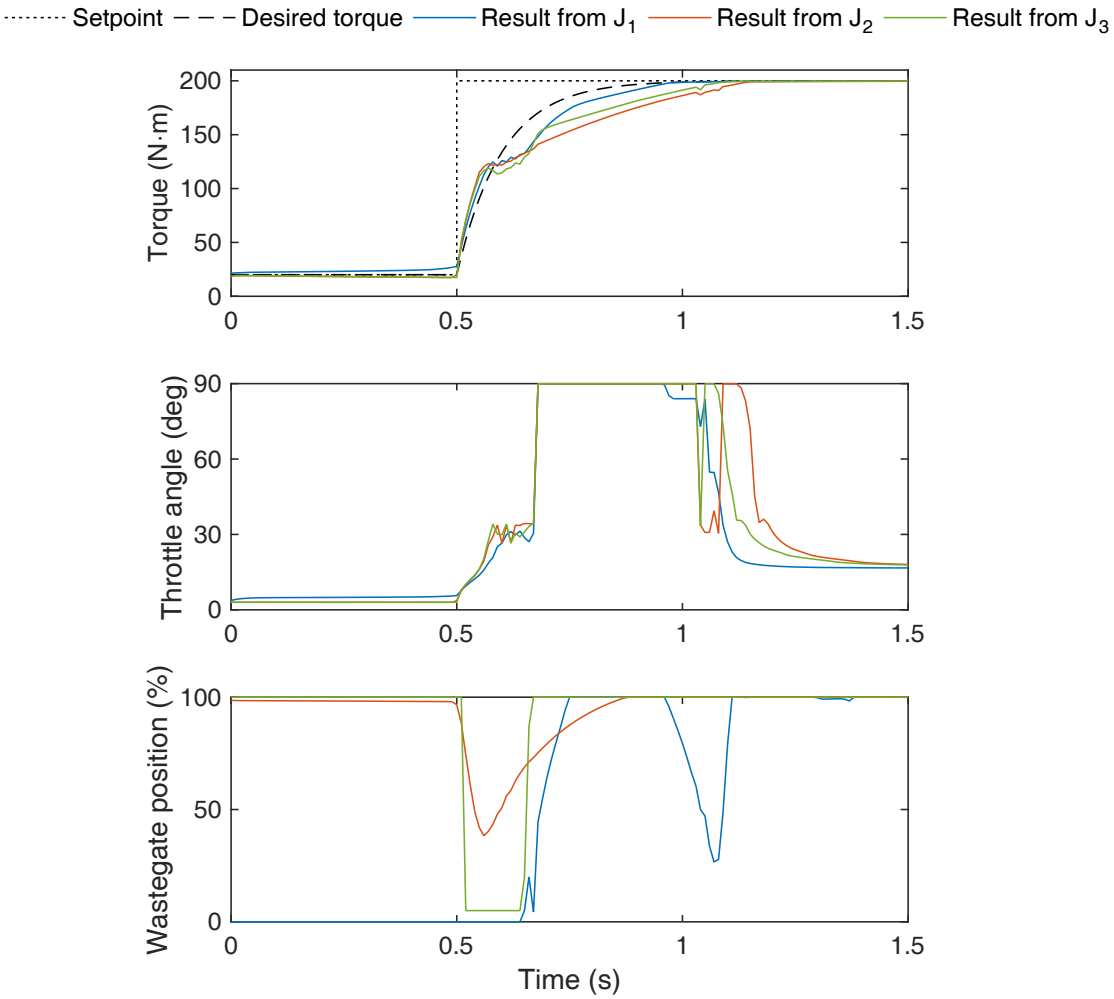


Figure 6.1: Optimal throttle and wastegate sequences for three different cost function formulations, and resulting torque outputs from the discrete-time 3-state model

The torque performances cannot be directly compared with the results from Chapter 4, since there is some modeling error between the 3-state model and the Simscape engine simulator. However, the results of the optimization provide confirmation for some of the beneficial control actions observed in Chapter 4. These actions include holding the wastegate open when the torque setpoint is constant, closing the wastegate for the setpoint increase, and opening the wastegate shortly after the setpoint increase to reduce pumping losses and provide a boost to the torque response.

The main limitation to this optimization scheme is the time required to converge to a solution. Each of the three cost function formulations required several hours to converge. As suggested in [21], a warm-start initialization may speed up convergence. Another major bottleneck seems to be the linearization step, where the Jacobian matrices are calculated. If analytical forms of the Jacobian matrices can be found, the linearization step would speed up considerably. However, this may be difficult since the discrete-time model uses a static map for the throttle pressure ratio, which is not easily differentiable.

Nonetheless, if the convergence time can be sufficiently reduced, it would be possible to use this optimization scheme in a model predictive controller. The desired torque trajectory would come from a preview of the reference torque, and the optimization problem would be solved at every time step, using new measurements to update the initial state.

6.5 Chapter summary

In this chapter, a multi-objective optimization scheme was designed to coordinate the throttle and wastegate simultaneously. Since the nonlinear state constraints made the problem non-convex, the problem was convexified by linearizing the dynamics over the system trajectory. This resulted in a quadratic program, which was solved iteratively. The cost function was formulated in three different ways. All three penalized the torque tracking error, but each formulation penalized fuel usage in different ways. The first directly penalized the fuel usage. The second penalized the deviation of the wastegate away from fully open, using a 2-norm. The third was similar to the second, except that a 1-norm was used. All three produced reasonably good results, with each producing a different tradeoff between torque tracking error and fuel usage. The resulting wastegate trajectories also displayed characteristics similar to those observed in Chapter 4, thus providing some confirmation that the heuristics behind the decentralized controller have the correct logic.

Chapter 7

Conclusion

7.1 Summary of dissertation

Control strategies

This dissertation presented two model-based control methods to balance fuel efficiency and torque responsiveness in a turbocharged spark-ignition engine.

The first method was a decentralized controller, in which the throttle and wastegate were controlled in separate loops. In Chapter 3, a throttle controller was designed for torque tracking. This controller consisted of a feedback linearization algorithm with supplemental proportional-integral (PI) control, and produced good torque tracking performance in spite of modeling error. Chapter 4 described an extensive simulation study to determine how the timing of the wastegate close-and-open switching affects the overall engine performance. The wastegate is left open when the torque setpoint is constant, and is closed during a setpoint increase. This strategy relies on a preview of the reference torque. The results showed improvements in both fuel usage and settling time, compared to holding the wastegate either fully closed or fully open.

The second method was a multi-objective optimization scheme, in which the throttle and wastegate were controlled simultaneously. In Chapter 6, a multi-objective cost function was used in an optimization scheme to trade off between torque tracking and fuel usage. However, the nonlinear state constraints made the problem non-convex, so the problem was convexified by linearizing the dynamics over the system trajectory. This resulted in a quadratic program, which was solved iteratively. The multi-objective cost function was formulated in three different ways, each producing a different tradeoff between torque tracking and fuel usage.

The decentralized controller takes a heuristic approach in the wastegate loop, but it is very easy to implement since there is not much computation required. On the other hand, the optimization scheme is more theoretically rigorous, but requires much more computation. In general, optimization-based methods can be challenging to use for real-time control because of the computation time required. When the problem is non-convex, there is no guarantee that a solution can be found in finite time. Even using the convexified method presented in Chapter 6, each of the three cost function formulations required several hours to converge. However, the optimization solutions

do provide some insight for what the best control actions should be. In particular, the optimization solutions produced wastegate trajectories that displayed similar characteristics as those observed in Chapter 4, thus providing some confirmation that the heuristics behind the decentralized controller have the correct logic.

Mean-value modeling

The models used in these control methods were also described in detail. A mean-value methodology was used, which entails estimating the average values of the model variables and neglecting fast variations within one engine cycle.

Chapter 2 introduced the high-fidelity engine simulator and the simpler 7-state and 4-state models. The Simscape engine simulator was used as the "true" plant for controller validation. The 7-state model captured the pressure and turbocharger speed dynamics well, but did not capture the temperature dynamics accurately. The 7-state model was further reduced to a 4-state model, which actually performed better than the 7-state model. The 4-state model was then used to design the throttle controller in Chapter 3.

Chapter 5 developed a discrete-time model for the turbocharged engine. First, an attempt was made to discretize the 4-state model, but it exhibited undesirable chattering and sometimes instability after discretization. These numerical issues were analyzed, and a solution was proposed by representing the throttle pressure ratio as a static map. In this way, a fast mode was eliminated and the model was reduced to 3 states. This 3-state model was easily converted to discrete-time. The discretized 3-state model was used in the optimization scheme in Chapter 6.

Replacing a fast dynamic mode with a static function is fairly common when dealing with stiff and singular perturbation. The methods in Chapter 5 may be extended to other stiff systems that need to be modeled in discrete time. The key difficulty is identifying the fast dynamic, which may not be immediately obvious.

7.2 Future work

Additional work can be done to improve upon the results presented in this dissertation.

The throttle controller in Chapter 3 was tuned to produce good performance when the wastegate is closed. While it also produced generally good performance when the wastegate was open, there were certain operating regions were undesirable oscillations occurred. This may be mitigated if the throttle controller gains are reduced in the regions that are sensitive to oscillations.

The wastegate controller in Chapter 4 requires additional effort before it can be fully implemented. More work must be done to determine the best wastegate timing for any arbitrary setpoint change. Adding closed-loop control may also help reduce undesired side effects from switching the wastegate open and closed.

In Chapter 5, it was observed that the wastegate position had only a negligible effect on the steady-state throttle pressure ratio. With further analysis, it may be possible to prove that this is

also true in theory. Additionally, it would be interesting if the throttle pressure ratio dynamics could be theoretically proven to be faster than the absolute pressure dynamics.

The optimization scheme in Chapter 6 may be used in a model predictive control strategy if the algorithm convergence time can be reduced. As suggested in [21], a warm-start initialization may speed up convergence. Additionally, some computation time may be reduced if analytical forms of the Jacobian matrices can be found. This would eliminate the need to calculate numerical estimates. The main difficulty here would be in differentiating the throttle pressure ratio map.

The controllers described in Chapters 3 and 6 rely on the availability of full-state feedback. This may not be practical, since sensors to measure the exhaust manifold pressure and turbocharger speed are typically too expensive for production engines. Therefore, a state observer should be developed to estimate these unmeasured states.

Finally, experimental testing on a physical engine can provide more validation of the control methods proposed in this dissertation. Experimental testing was not performed in this work, since a suitable engine test bench was not readily available. An attempt was made to set up a working engine test bench, but there were numerous hardware problems and the effort was eventually abandoned due to lack of time and resources.

Bibliography

[1] P. Leduc, B. Dubar, A. Ranini, G. Monnier, and G. Monnier, "Downsizing of gasoline engine: an efficient way to reduce CO_2 emissions," *Oil & Gas Science and Technology*, vol. 58, pp. 115–127, Jan. 2003.

[2] L. Eriksson, S. Frei, C. Onder, and L. Guzzella, "Control and optimization of turbocharged spark ignited engines," in *IFAC World Congress*, vol. 15, 2002.

[3] D. Cho and J. K. Hedrick, "Automotive powertrain modeling for control," *Journal of Dynamic Systems, Measurement, and Control*, vol. 111, no. 4, pp. 568–576, 1989.

[4] E. Hendricks and S. C. Sorenson, "Mean value modelling of spark ignition engines," *SAE Technical Paper*, 1990.

[5] J.-P. Jensen, A. Kristensen, S. C. Sorenson, N. Houbak, and E. Hendricks, "Mean value modeling of a small turbocharged diesel engine," *SAE Technical Paper*, 1991.

[6] T. Nagata, *Application and Enhancement of Model-Based Design for Automotive Engine and Powertrain Control*. Ph.D. dissertation, University of California, Berkeley, 2009.

[7] M. Kao and J. J. Moskwa, "Turbocharged diesel engine modeling for nonlinear engine control and state estimation," *Journal of Dynamic Systems, Measurement, and Control*, vol. 117, p. 20, Mar. 1995.

[8] L. Eriksson, L. Nielsen, J. Brugård, J. Bergström, F. Pettersson, and P. Andersson, "Modeling of a turbocharged SI engine," *Annual Reviews in Control*, vol. 26, pp. 129–137, 2002.

[9] L. Guzzella and C. Onder, *Introduction to Modeling and Control of Internal Combustion Engine Systems*. Springer Berlin Heidelberg, 2010.

[10] P. Moulin and J. Chauvin, "Modeling and control of the air system of a turbocharged gasoline engine," *Control Engineering Practice*, vol. 19, pp. 287–297, Mar. 2011.

[11] R. Sharma, D. Nesic, and C. Manzie, "Model reduction of turbocharged (TC) spark ignition (SI) engines," *IEEE Transactions on Control Systems Technology*, vol. 19, pp. 297–310, Mar. 2011.

[12] P. Moraal and I. Kolmanovsky, "Turbocharger modeling for automotive control applications," *SAE Technical Paper*, 1999.

[13] H. Ito, H. Yohata, J. Kako, and Y. Kuroda, "Development of high level modeling method for rapid modeling process," *SAE International Journal of Engines*, vol. 6, pp. 669–679, Apr. 2013.

[14] S. Yoshida, H. Ito, T. Morikawa, J. Kako, and Y. Kuroda, "Development of high level modeling method for rapid modeling process – application for combustion modeling in engine cylinder," *Japan-SAE Technical Paper 20145044*, 2014.

[15] L. Eriksson and L. Nielsen, *Modeling and Control of Engines and Drivelines*. John Wiley and Sons, Ltd, 2014.

[16] A. Y. Karnik, J. H. Buckland, and J. S. Freudenberg, "Electronic throttle and wastegate control for turbocharged gasoline engines," in *2005 American Control Conference (ACC)*, pp. 4434–4439, 2005.

[17] G. Colin, Y. Chamaillard, and B. Bellicaud, "Robust control for the air path of a downsized engine," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 225, pp. 930–943, June 2011.

[18] P. Ortner and L. del Re, "Predictive control of a diesel engine air path," *IEEE Transactions on Control Systems Technology*, vol. 15, pp. 449–456, May 2007.

[19] G. Stewart and F. Borrelli, "A model predictive control framework for industrial turbodiesel engine control," in *47th IEEE Conference on Decision and Control*, pp. 5704–5711, Dec. 2008.

[20] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat, "Predictive active steering control for autonomous vehicle systems," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 3, pp. 566–580, 2007.

[21] A. Carvalho, Y. Gao, A. Gray, H. E. Tseng, and F. Borrelli, "Predictive control of an autonomous ground vehicle using an iterative linearization approach," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pp. 2335–2340, Oct. 2013.

[22] R. Tan and M. Tomizuka, "Control-oriented model of a turbocharged engine airpath with discrete-time considerations," in *2016 American Control Conference (ACC)*, pp. 5075–5080, July 2016. © 2016 IEEE.

[23] Mathworks, *Simscape Documentation: Physical Modeling*, 2015. http://www.mathworks.com/help/physmod/simscape/index.html.

[24] J. Heywood, *Internal Combustion Engine Fundamentals*. New York: McGraw-Hill, 1988.

[25] K. Edelberg, S. Pan, and J. K. Hedrick, "A discrete-time sliding mode formulation for automotive cold-start emission control," in *52nd IEEE Conference on Decision and Control*, pp. 6818–6823, Dec. 2013.

[26] K. Butts and H. Ito. Private communication, July 7, 2015.

[27] L. Jiang, J. Vanier, H. Yilmaz, and A. Stefanopoulou, "Parameterization and simulation for a turbocharged spark ignition direct injection engine with variable valve timing," *SAE Technical Paper*, 2009.

[28] J. D'Errico, "Adaptive robust numerical differentiation." MATLAB File Exchange, Dec. 2014. https://www.mathworks.com/matlabcentral/fileexchange/13490-adaptive-robust-numerical-differentiation.

[29] J. Löfberg, "YALMIP: A toolbox for modeling and optimization in MATLAB," in *2004 IEEE International Symposium on Computer Aided Control Systems Design*, pp. 284–289, 2004.

Appendix A

List of Model Parameters

The following is a complete list of the model parameters, along with their values, that are used in Chapter 2.

Parameter	Value	Description
a_1	2.26×10^{-7} (kg/s)/Pa	Aspirated mass flow rate parameter
a_2	-2.95×10^{-8} (kg/s)/Pa	Aspirated mass flow rate parameter
a_3	2.3×10^{-3} kg/s	Aspirated mass flow rate parameter
b_1	0.7964	Spark influence function parameter
b_2	2.000×10^{-4}	Spark influence function parameter
b_3	1.456×10^{-6}	Spark influence function parameter
A_t	10^{-3} m^2	Turbine model parameter
c_{g_1}	2.817×10^{-3}	Compressor model parameter
c_{g_2}	0.3102	Compressor model parameter
$c_{p,a}$	1.0047×10^3 J/(kg·K)	Specific heat of air at constant pressure
$c_{p,b}$	1.0393×10^3 J/(kg·K)	Specific heat of burned gas at constant pressure
c_{r_1}	9.408	Compressor model parameter
c_{r_2}	0.5499	Compressor model parameter
c_{r_3}	8.598	Compressor model parameter
c_{r_4}	0.7078	Compressor model parameter
c_{r_5}	38.60	Compressor model parameter
c_{t_1}	1.251×10^{-3}	Compressor model parameter
c_{t_2}	0.3109	Compressor model parameter
c_{t_3}	1.855	Compressor model parameter
d_{m_1}	5.137×10^4 K/(kg/s)	Engine-out temperature parameter

(Continued on next page)

(Continued from previous page)

Parameter	Value	Description
d_{m_2}	1.280×10^3 K	Engine-out temperature parameter
$d_{s_1,hi}$	-1.082×10^{-4}	Engine-out temperature parameter
$d_{s_1,lo}$	3.425×10^{-5}	Engine-out temperature parameter
$d_{s_2,hi}$	-4.513×10^{-3}	Engine-out temperature parameter
$d_{s_2,lo}$	4.713×10^{-5}	Engine-out temperature parameter
D_{th}	0.055 m	Diameter of throttle valve
D_{wg}	0.0342 m	Diameter of wastegate valve
h_{em}	208.71 (J/s)/K	Exhaust manifold effective heat transfer coefficient
h_{ic}	354.24 (J/s)/K	Intercooler effective heat transfer coefficient
h_{im}	246.54 (J/s)/K	Intake manifold effective heat transfer coefficient
J_{tc}	10^{-5} kg·m^2	Turbocharger inertia
p_{amb}	101325 Pa	Ambient pressure (intake side)
p_{ex}	115000 Pa	Ambient pressure (exhaust side)
q_{LHV}	4.25×10^7 J/kg	Lower heating value of fuel
r_c	13	Compression ratio of engine
r_t	0.035 m	Turbine blade radius
R_a	287.058 J/(kg·K)	Specific gas constant of air
R_b	282.79 J/(kg·K)	Specific gas constant of burned gas
R_u	8.314462 J/(mol·K)	Universal gas constant
S_{wg}	0.007 m	Stroke of wastegate valve
SA_{opt}	34.41 deg	Spark influence function parameter
t_{e_1}	0.8	Turbine model parameter
t_{e_2}	0.7	Turbine model parameter
t_{f_1}	4.949×10^{-6}	Turbine model parameter
t_{f_2}	9.134×10^{-3}	Turbine model parameter
t_{f_3}	6.391×10^{-6}	Turbine model parameter
t_{f_4}	1.260×10^{-4}	Turbine model parameter
T_{amb}	298.15 K	Ambient air temperature (intake side)
T_{em}	850 K	Exhaust manifold temperature (for 4-state model)
T_{ex}	566.62 K	Ambient air temperature (exhaust side)
T_{ic}	312 K	Intercooler temperature (for 4-state model)
T_{im}	309 K	Intake manifold temperature (for 4-state model)

(Continued on next page)

(Continued from previous page)

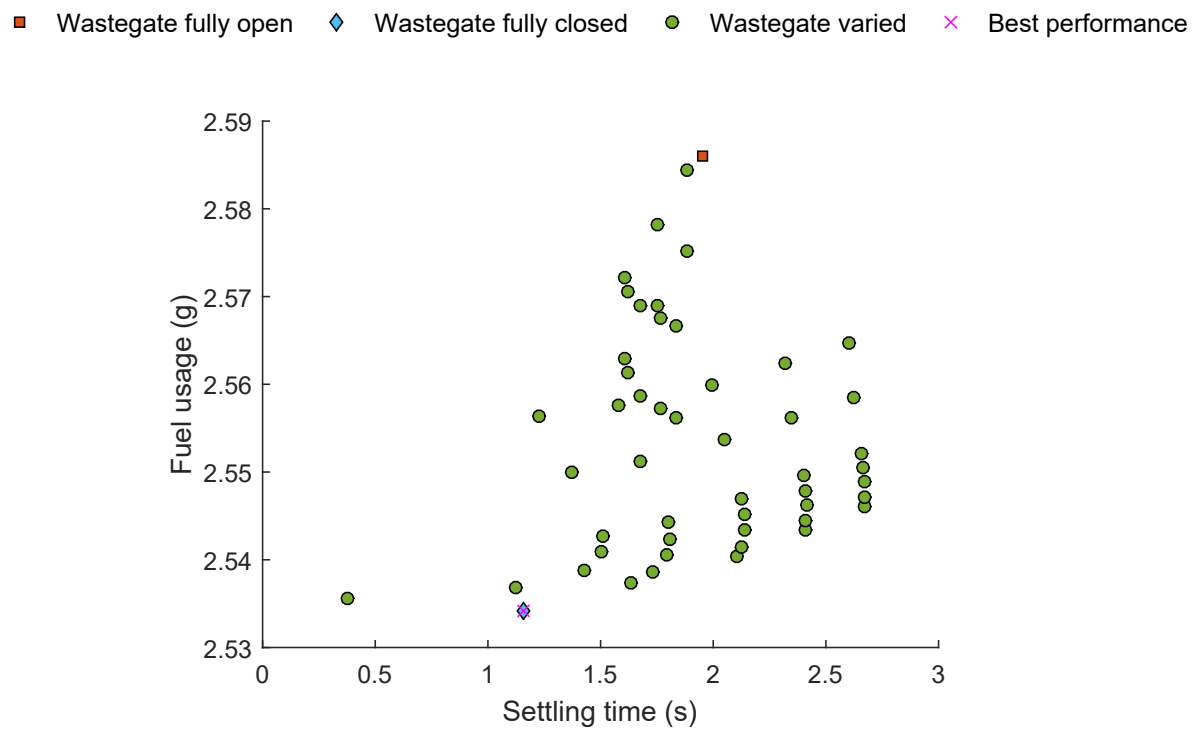
Parameter	Value	Description
V_c	V_d/r_c	Engine compression volume
V_d	1.798×10^{-3} m^3	Engine displacement volume
V_{em}	8.6489×10^{-4} m^3	Exhaust manifold volume
V_{ic}	1.9453×10^{-3} m^3	Intercooler volume
V_{im}	3.6455×10^{-3} m^3	Intake manifold volume
γ_a	1.4	Ratio of specific heats (c_p/c_v) for air
γ_b	1.373	Ratio of specific heats (c_p/c_v) for burned gas
Λ_s	14.6	Stoichiometric air-fuel ratio

Appendix B

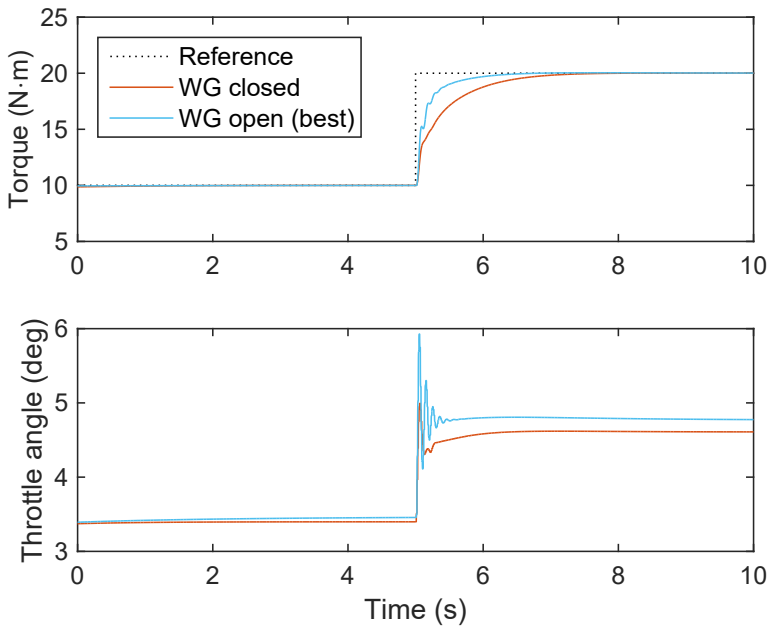
Additional Results of Wastegate Timing Tests

This appendix presents additional results of all wastegate timing tests, as described in Chapter 4. For each setpoint change, two plots are shown. The first is a scatter plot of the total fuel usage versus the 2% settling time for each test. The second is a time response plot showing the torque response for when the wastegate is fully closed, fully open, and switched between closed and open for the best performance. For some setpoint changes, holding the wastegate fully open produced the best performance. For all other setpoint changes, the time response plot shows a shaded gray region indicating the time period that the wastegate was closed to give the best performance.

In all scenarios, the test with wastegate fully closed was never chosen to have the best performance, since the wastegate is assumed to be open by default. However, the results with the wastegate fully closed are included in the plots for performance comparisons.

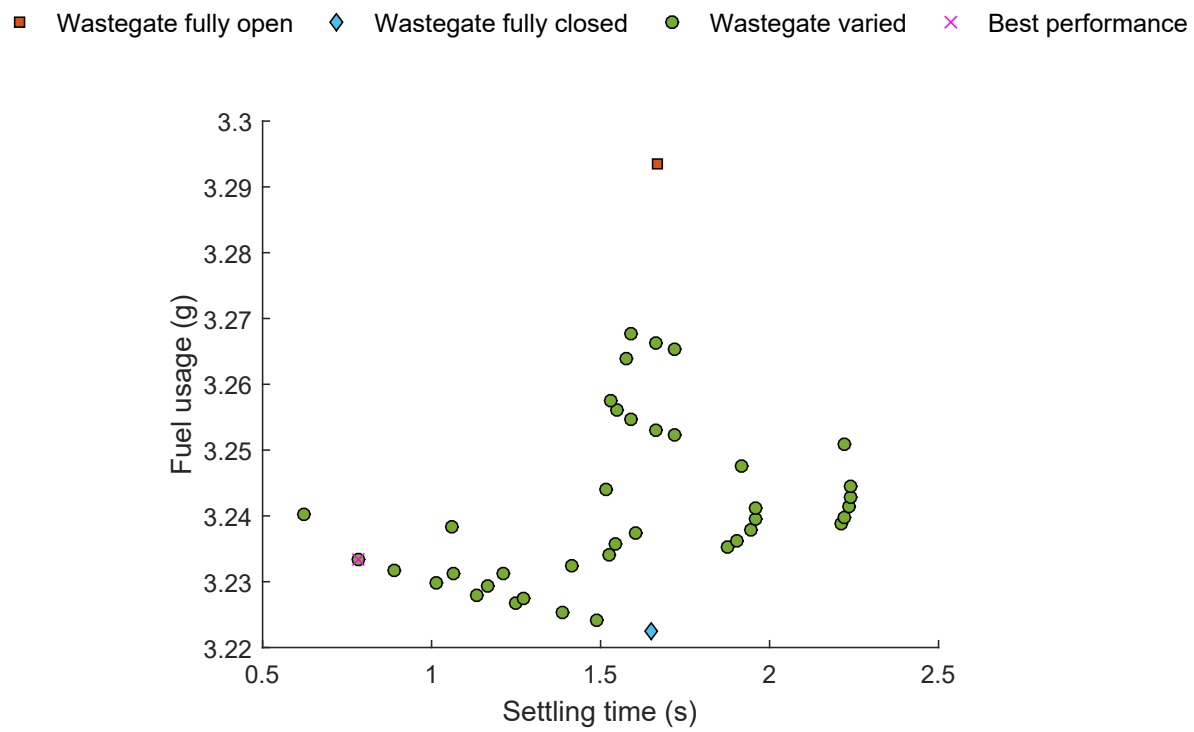


(a) Fuel usage vs. settling time for all wastegate timing tests

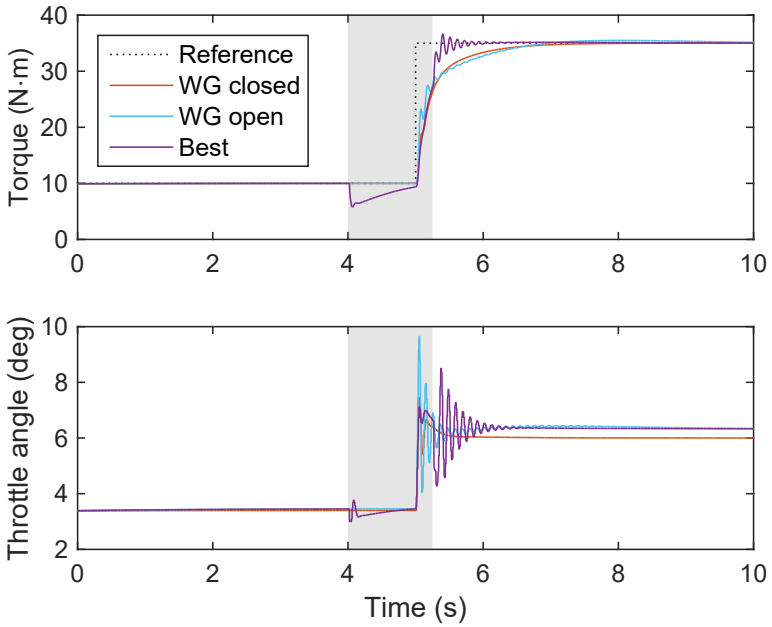


(b) Time response for selected wastegate timing tests

Figure B.1: Torque setpoint change from 10 N·m to 20 N·m

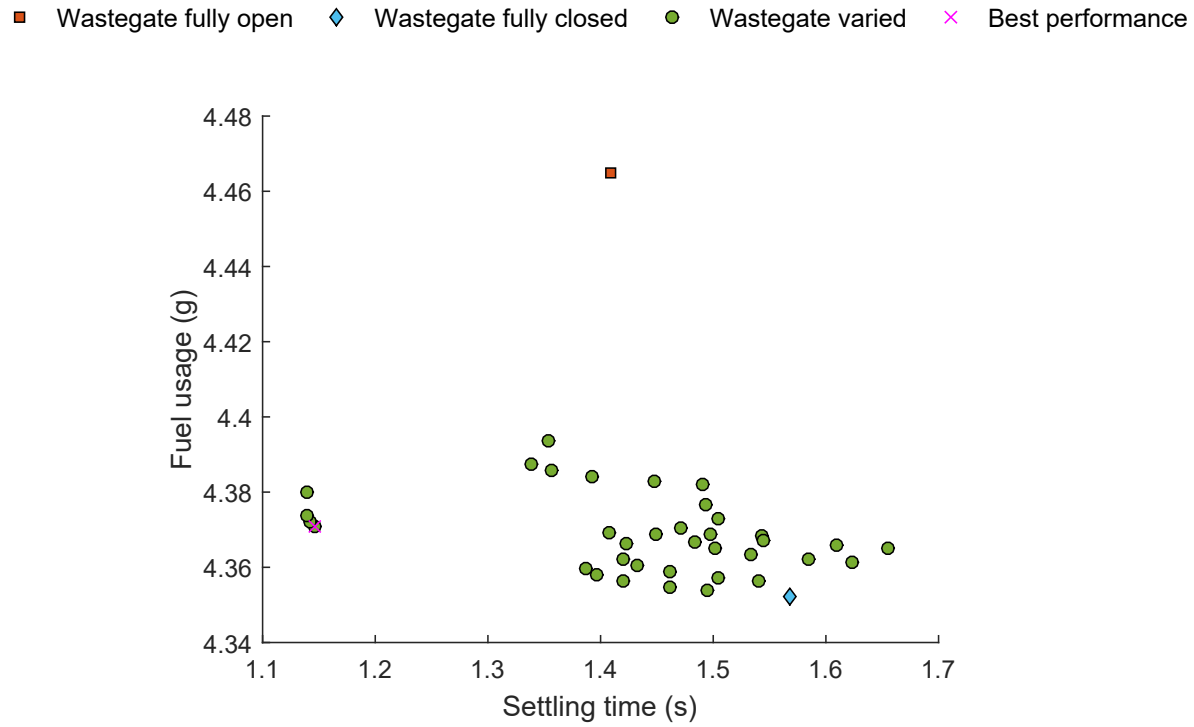


(a) Fuel usage vs. settling time for all wastegate timing tests

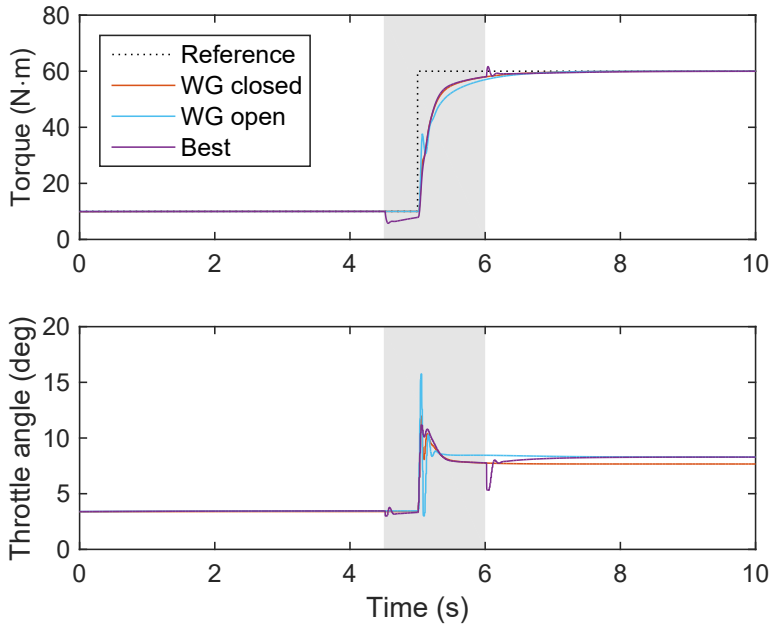


(b) Time response for selected wastegate timing tests

Figure B.2: Torque setpoint change from 10 N·m to 35 N·m

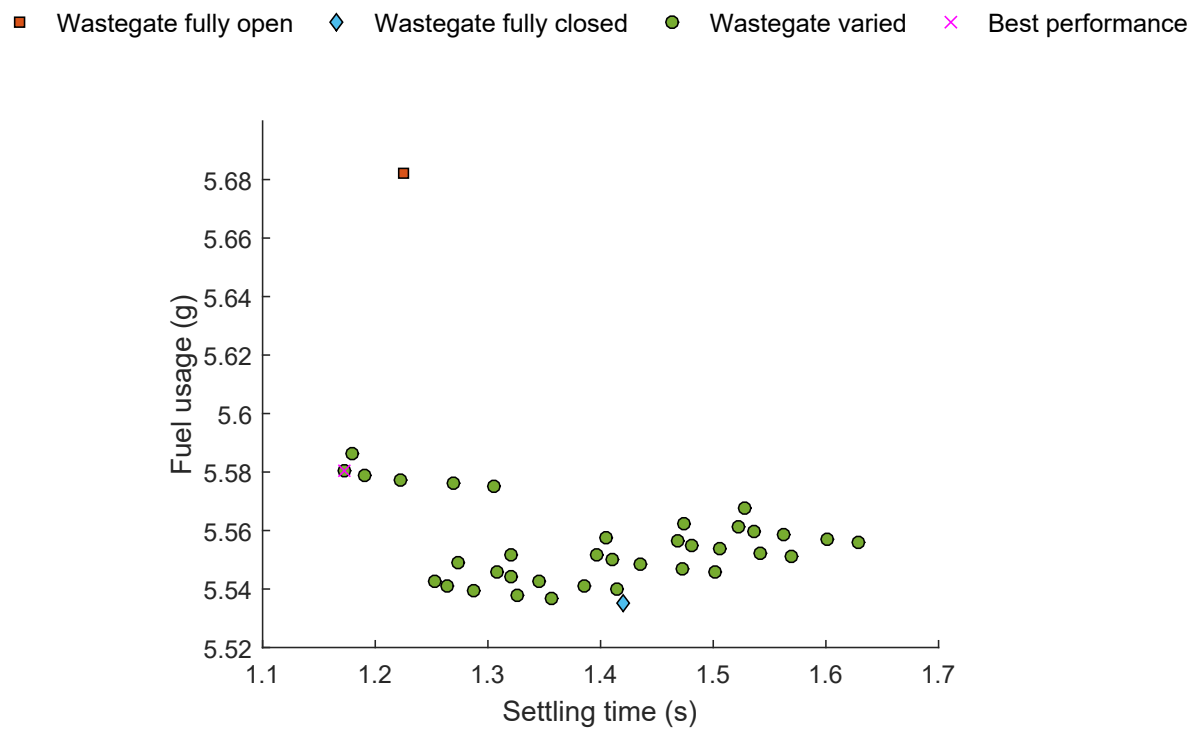


(a) Fuel usage vs. settling time for all wastegate timing tests

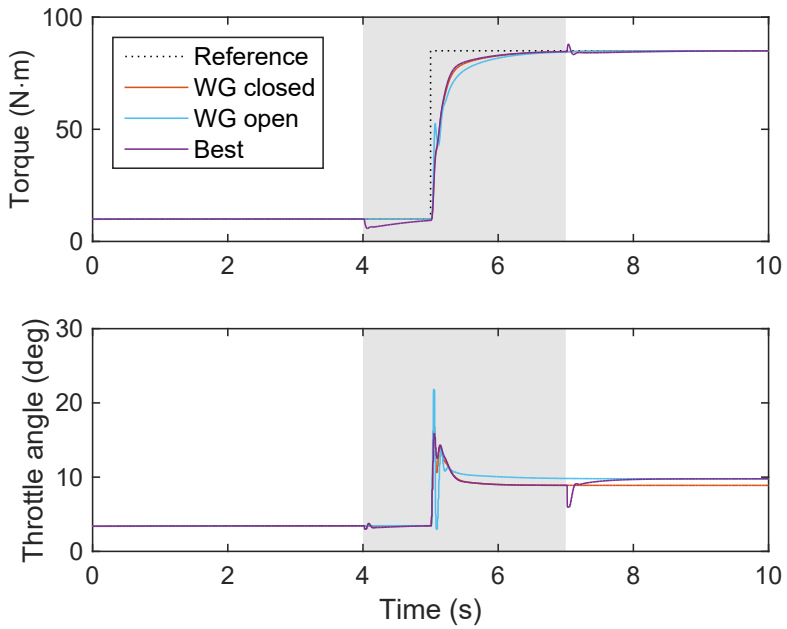


(b) Time response for selected wastegate timing tests

Figure B.3: Torque setpoint change from 10 N·m to 60 N·m

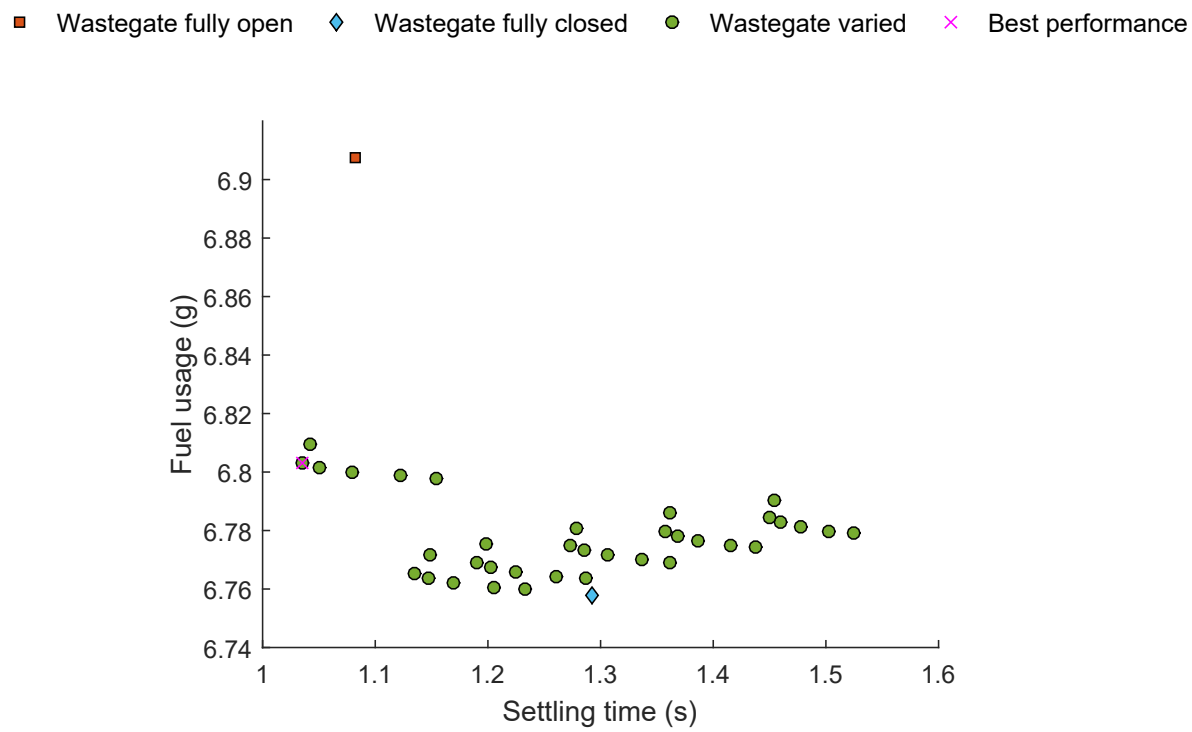


(a) Fuel usage vs. settling time for all wastegate timing tests

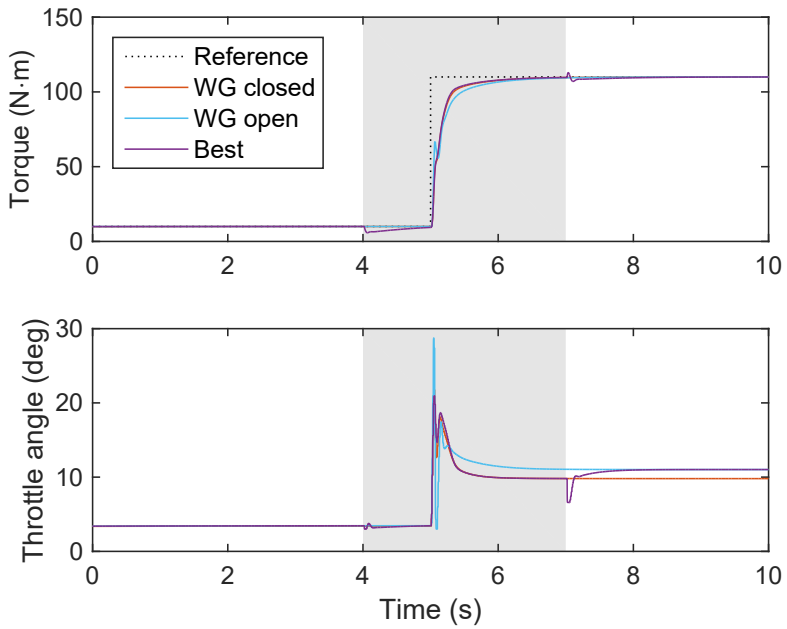


(b) Time response for selected wastegate timing tests

Figure B.4: Torque setpoint change from 10 N·m to 85 N·m

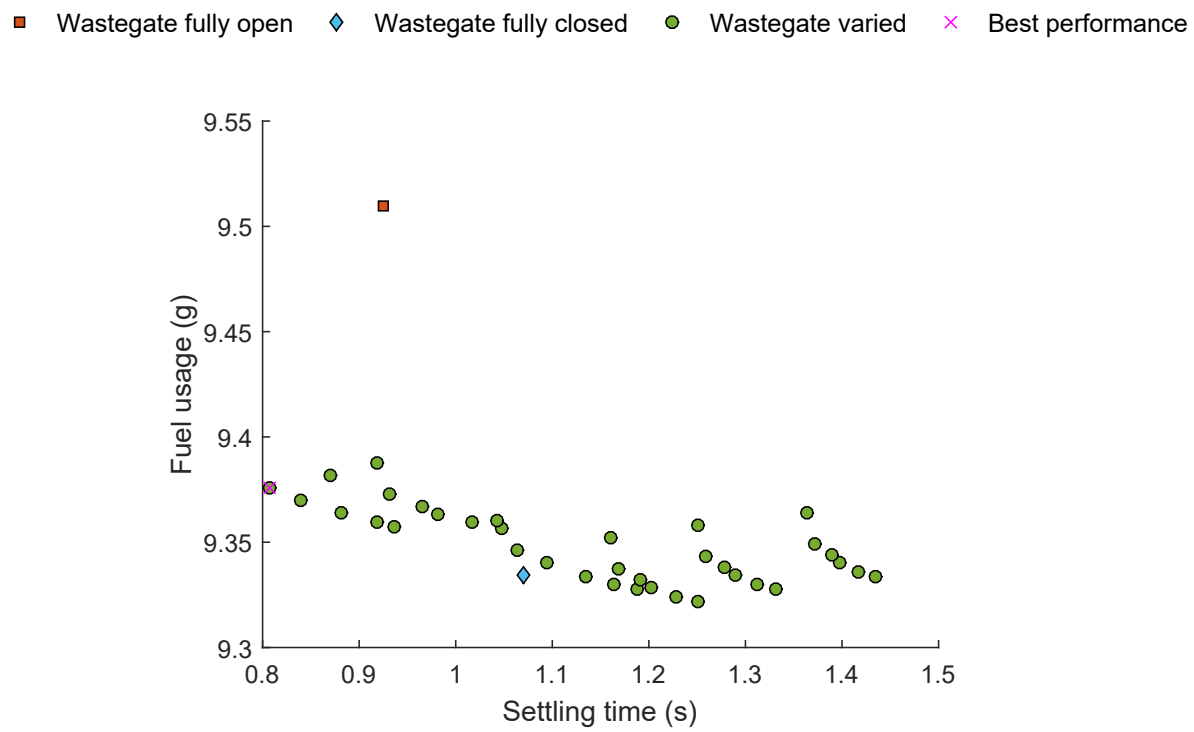


(a) Fuel usage vs. settling time for all wastegate timing tests

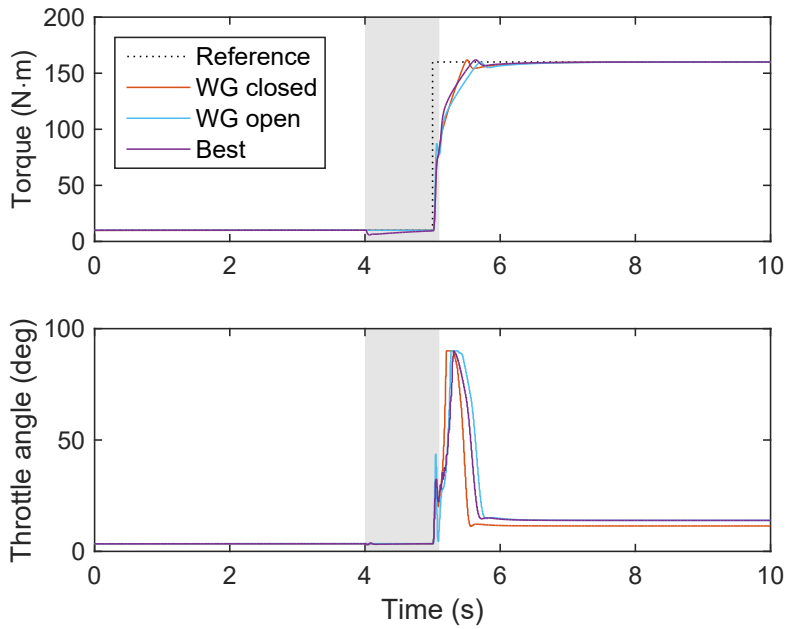


(b) Time response for selected wastegate timing tests

Figure B.5: Torque setpoint change from 10 N·m to 110 N·m

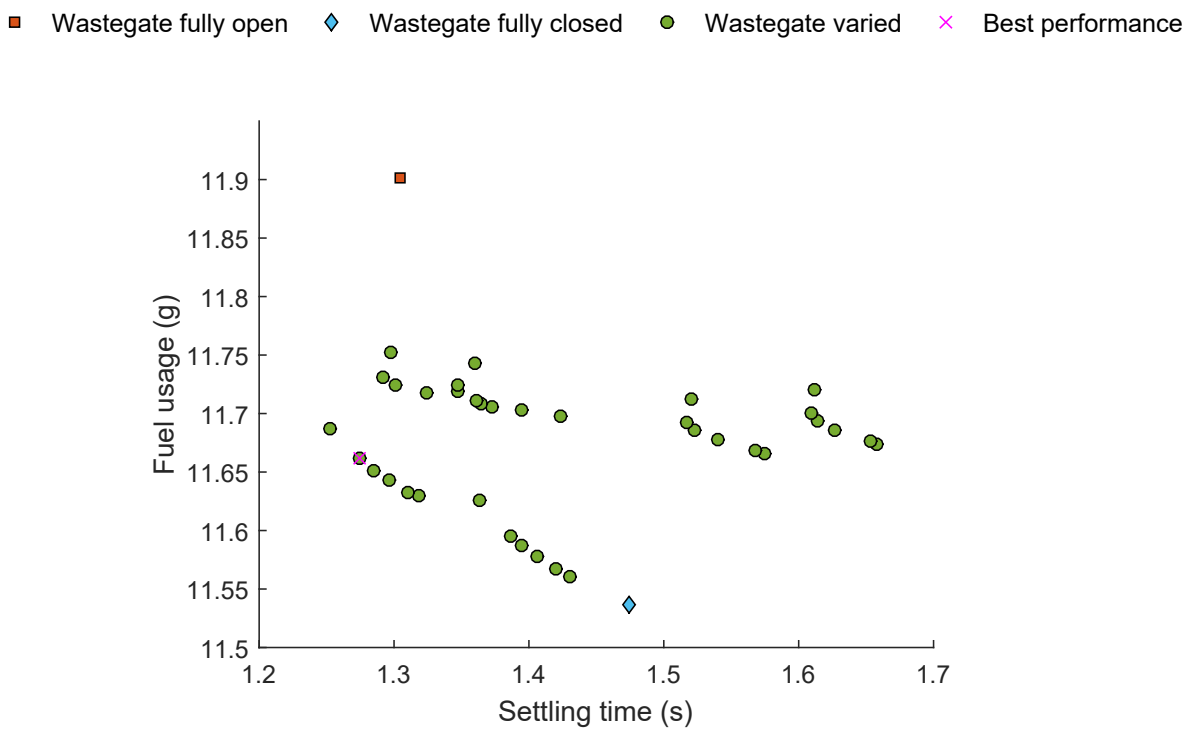


(a) Fuel usage vs. settling time for all wastegate timing tests

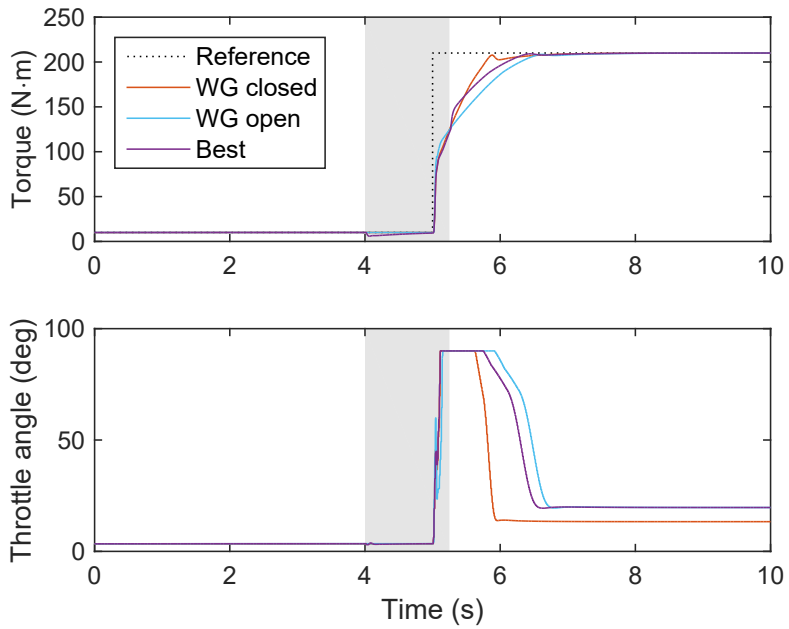


(b) Time response for selected wastegate timing tests

Figure B.6: Torque setpoint change from 10 N·m to 160 N·m

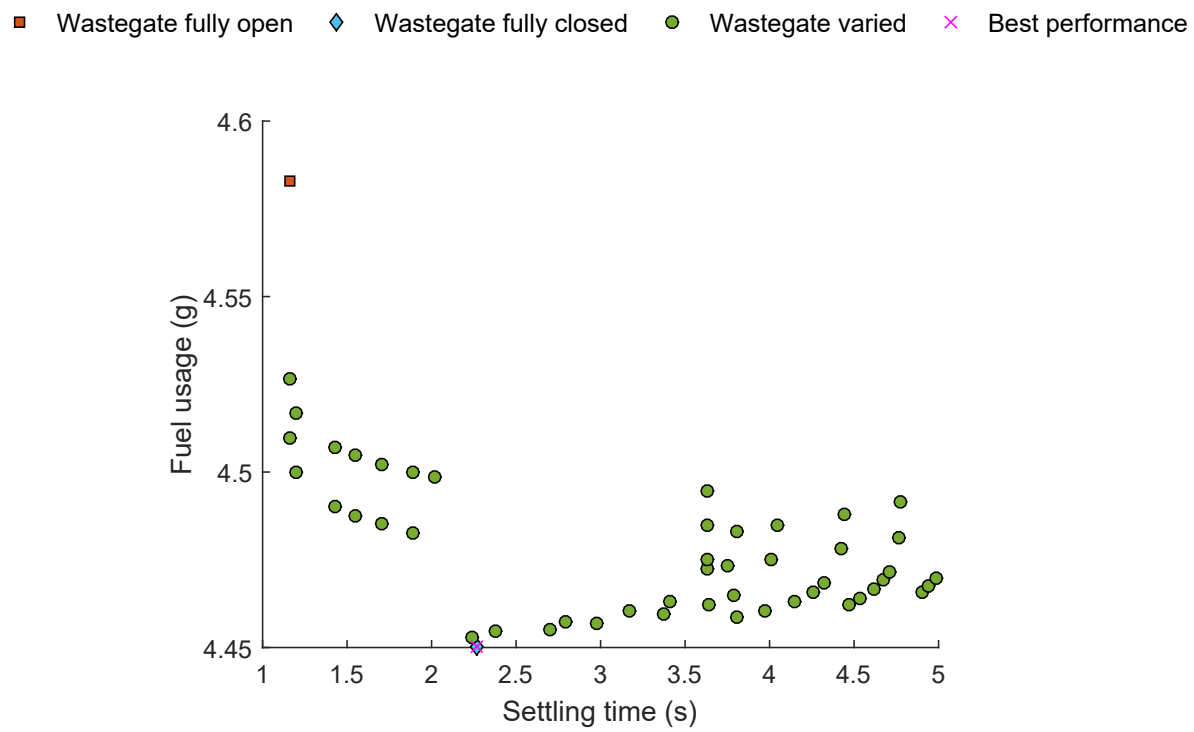


(a) Fuel usage vs. settling time for all wastegate timing tests

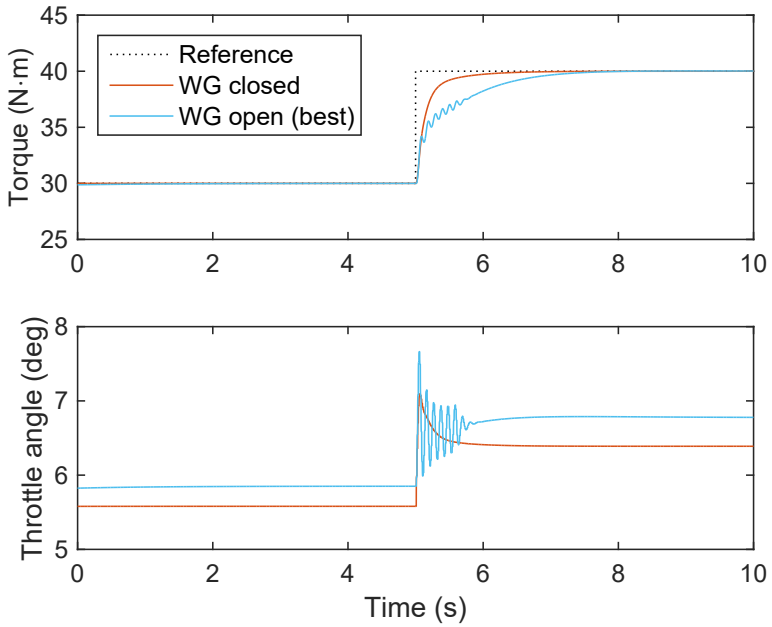


(b) Time response for selected wastegate timing tests

Figure B.7: Torque setpoint change from 10 N·m to 210 N·m

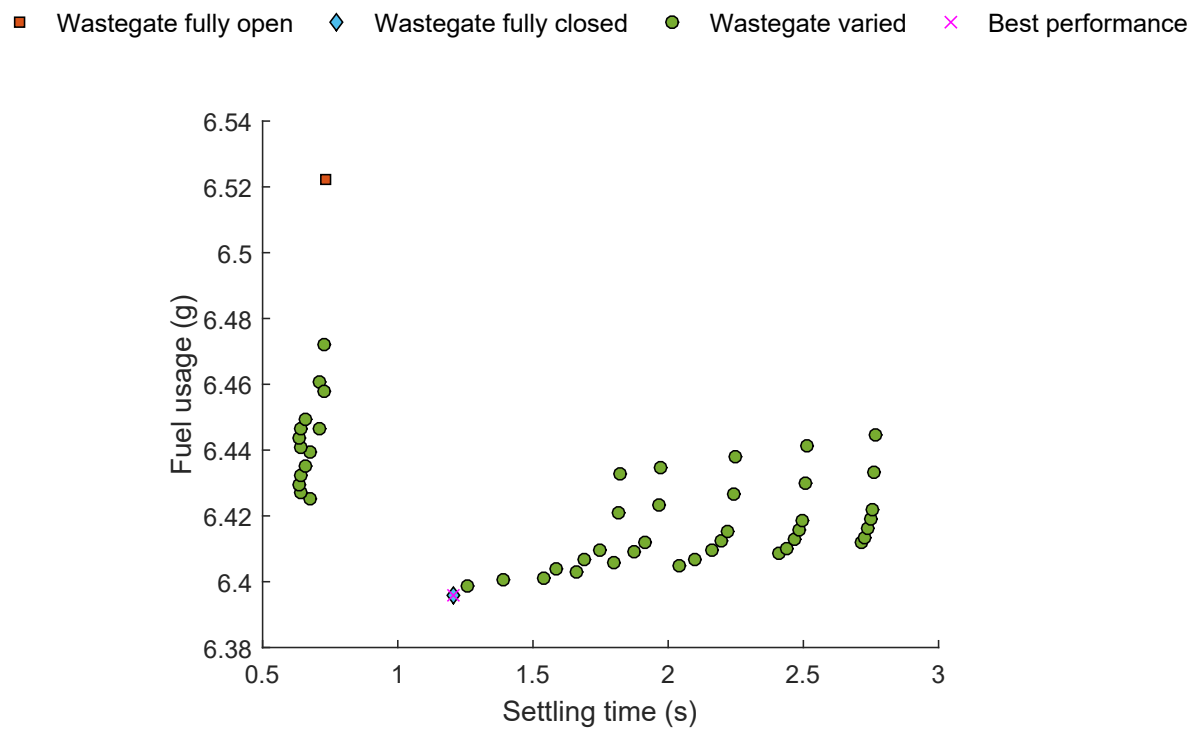


(a) Fuel usage vs. settling time for all wastegate timing tests

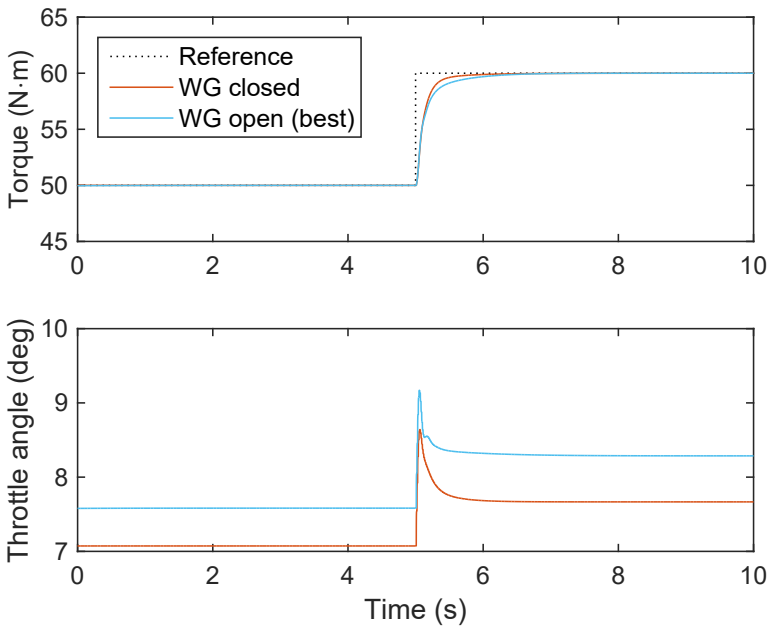


(b) Time response for selected wastegate timing tests

Figure B.8: Torque setpoint change from 30 N·m to 40 N·m

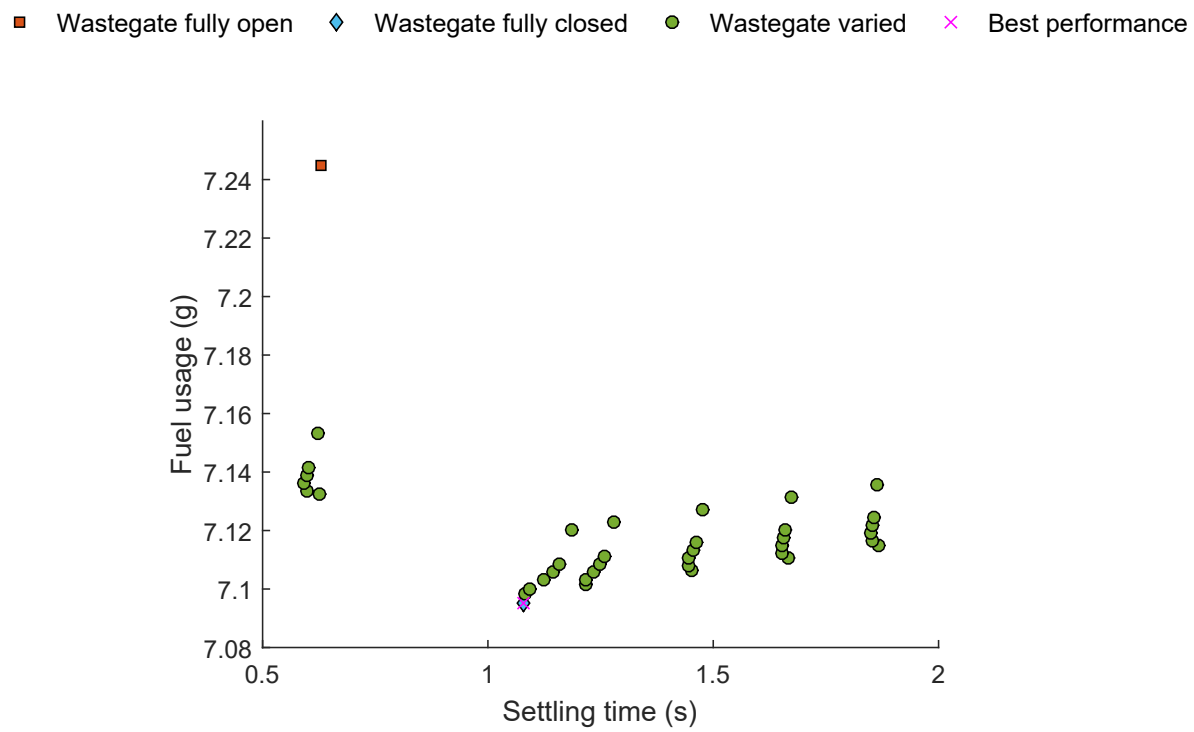


(a) Fuel usage vs. settling time for all wastegate timing tests

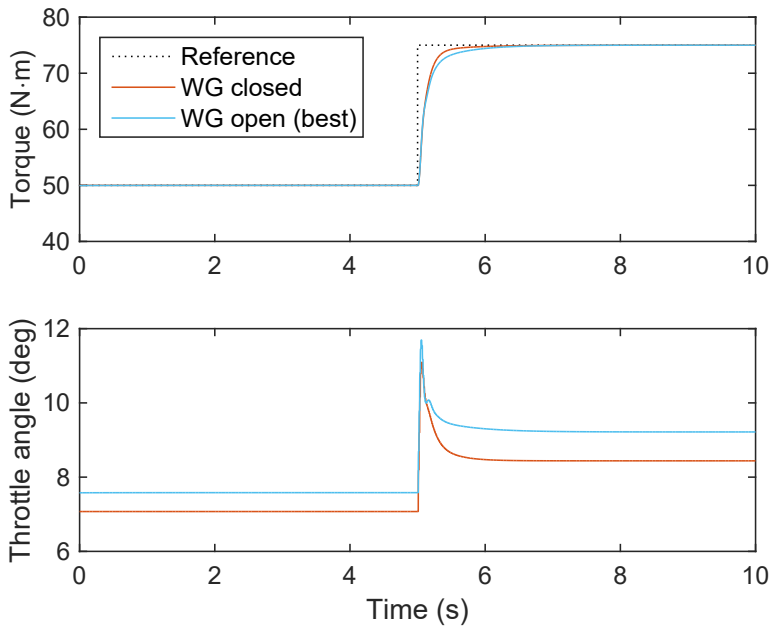


(b) Time response for selected wastegate timing tests

Figure B.9: Torque setpoint change from 50 N·m to 60 N·m

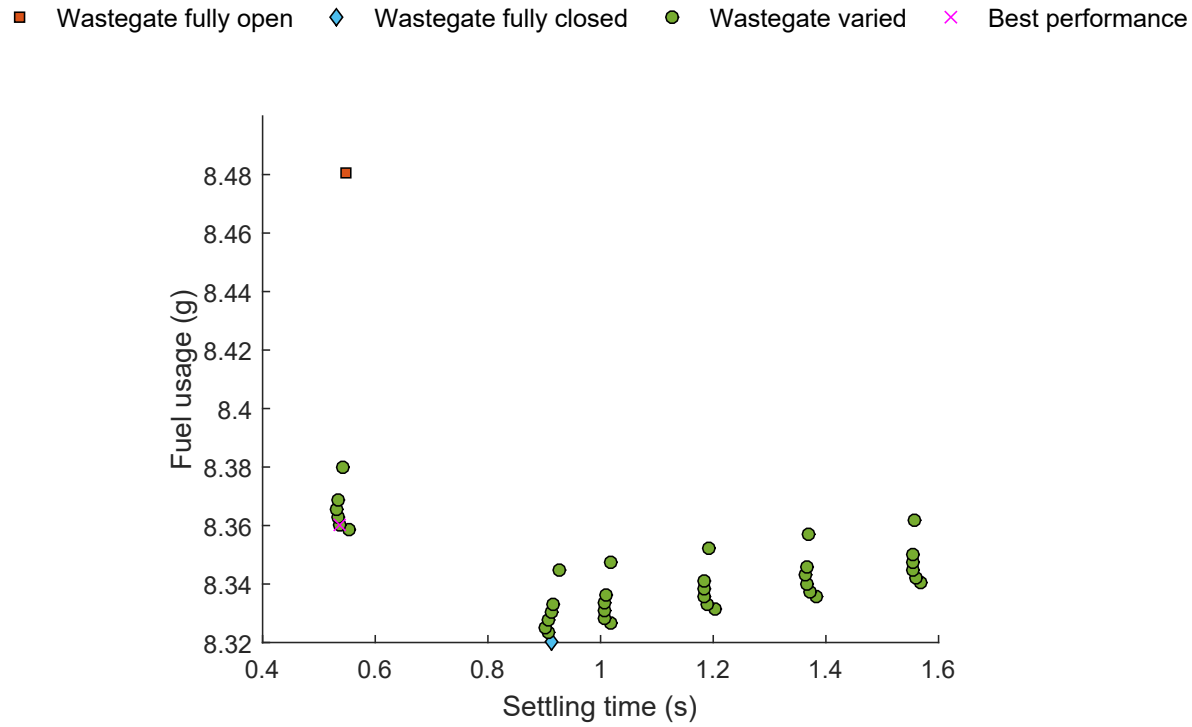


(a) Fuel usage vs. settling time for all wastegate timing tests

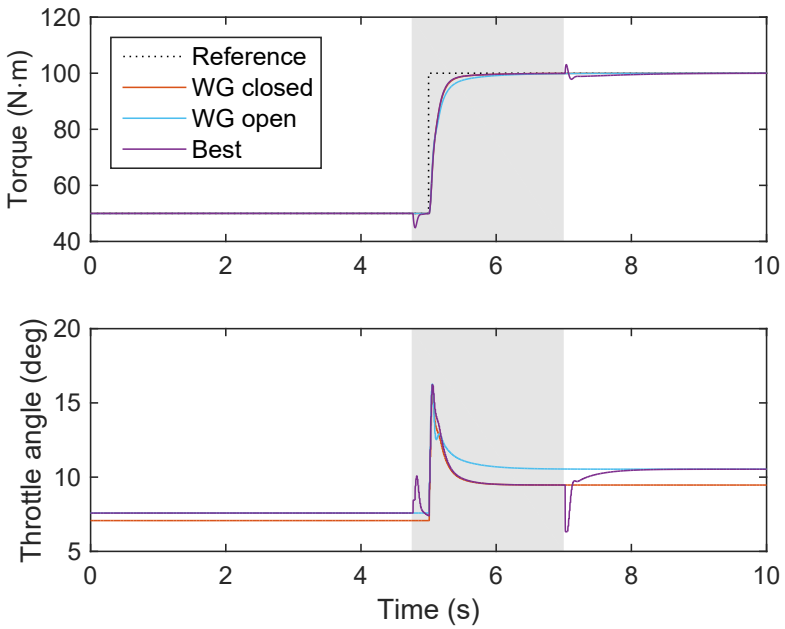


(b) Time response for selected wastegate timing tests

Figure B.10: Torque setpoint change from 50 N·m to 75 N·m

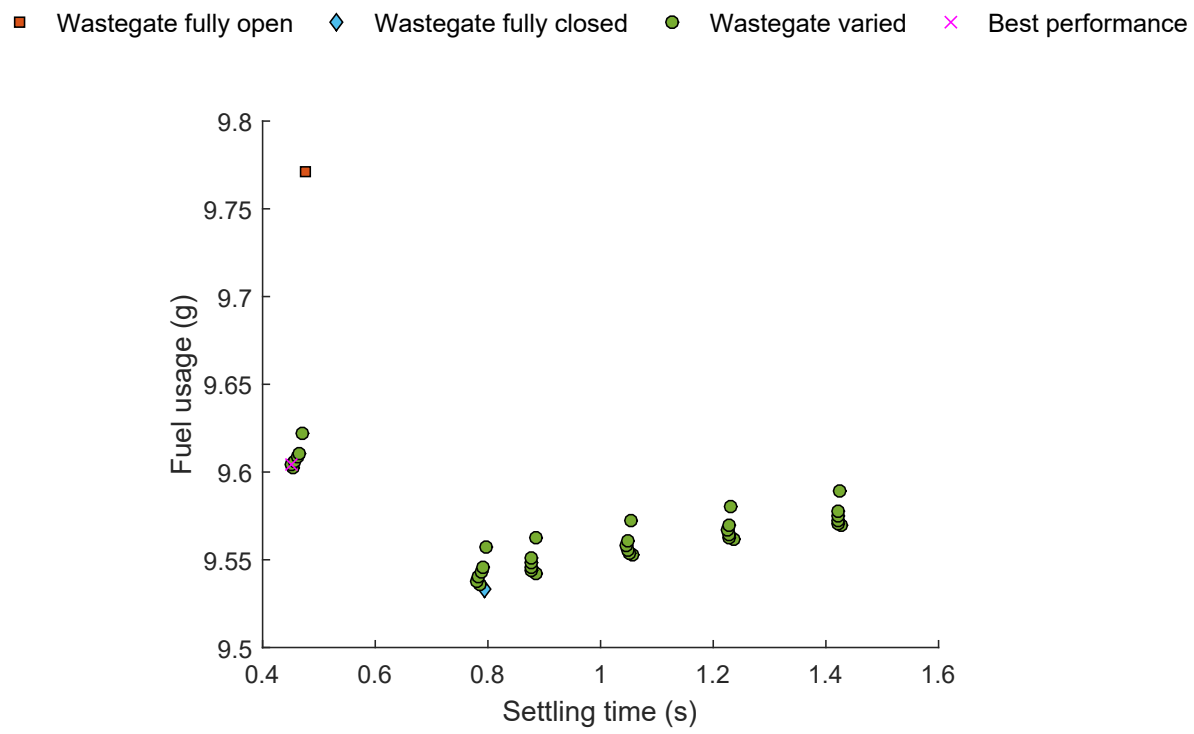


(a) Fuel usage vs. settling time for all wastegate timing tests

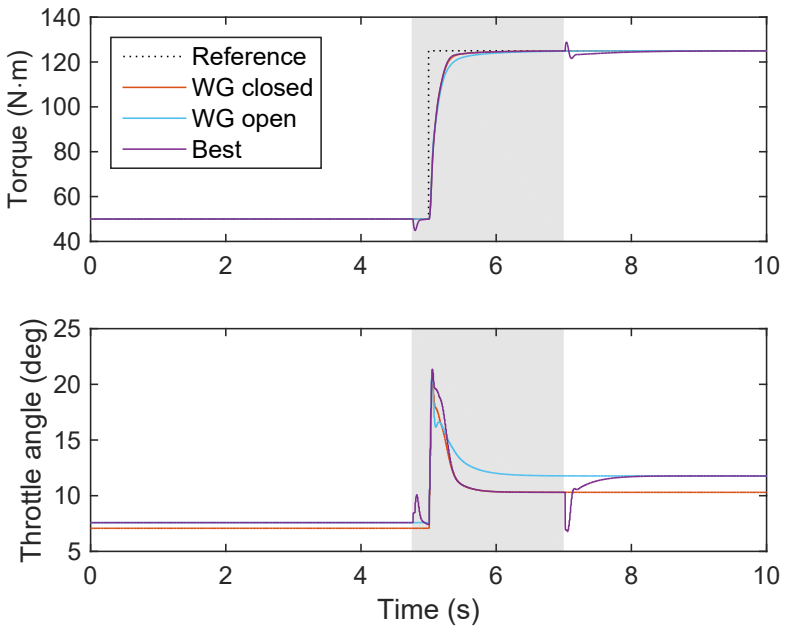


(b) Time response for selected wastegate timing tests

Figure B.11: Torque setpoint change from 50 N·m to 100 N·m

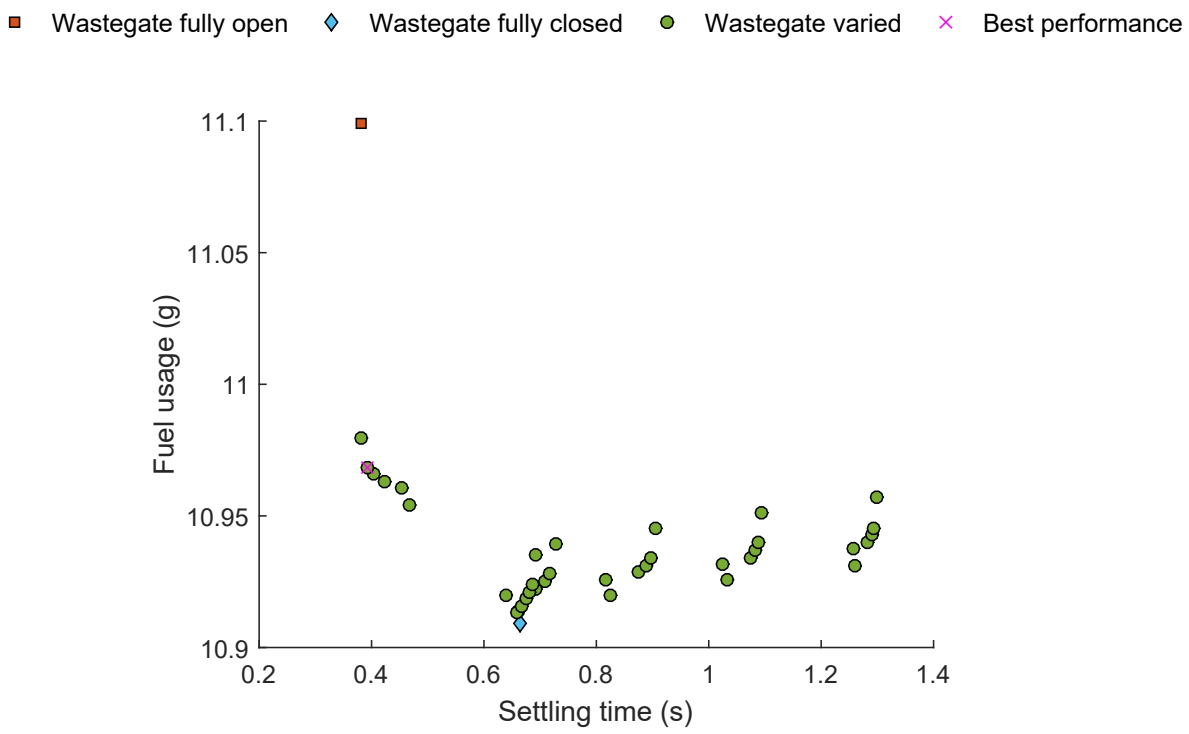


(a) Fuel usage vs. settling time for all wastegate timing tests

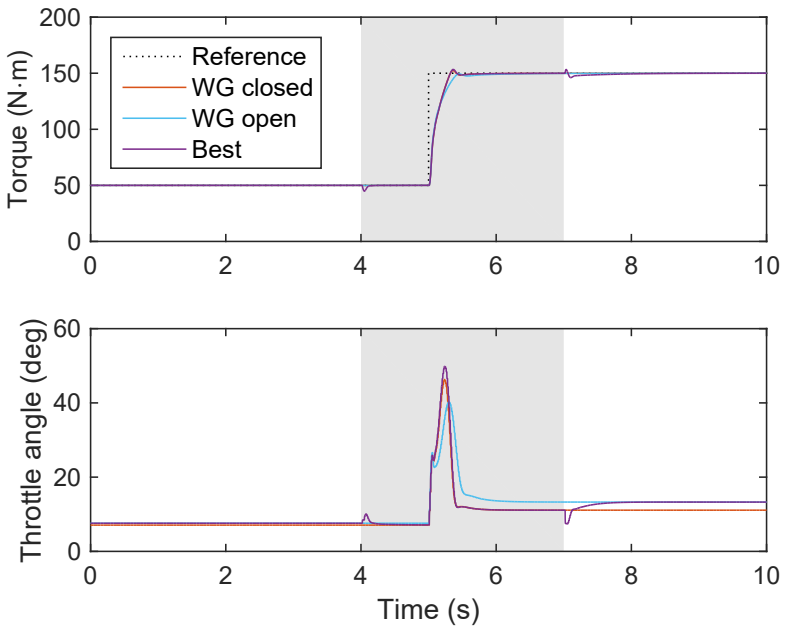


(b) Time response for selected wastegate timing tests

Figure B.12: Torque setpoint change from 50 N·m to 125 N·m

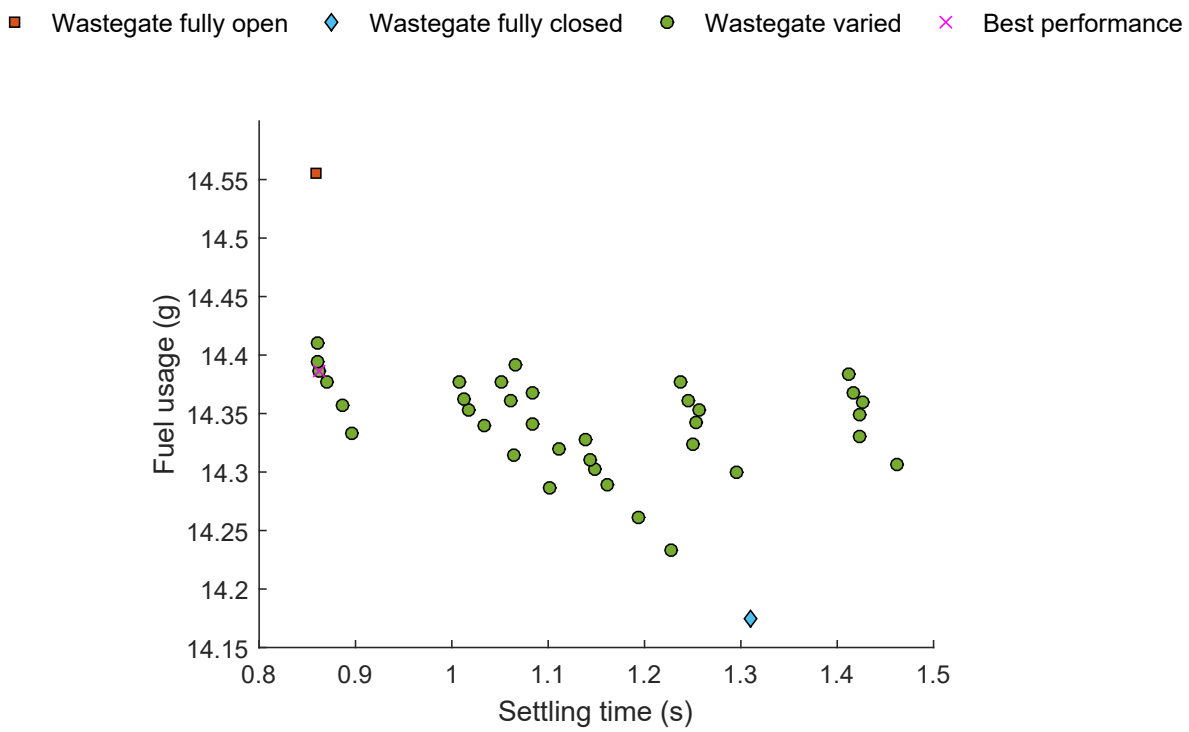


(a) Fuel usage vs. settling time for all wastegate timing tests

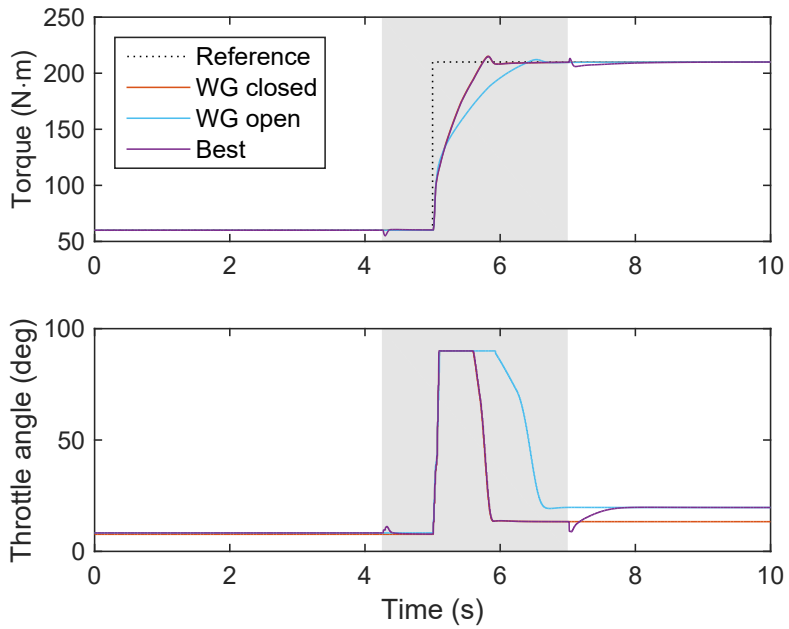


(b) Time response for selected wastegate timing tests

Figure B.13: Torque setpoint change from 50 N·m to 150 N·m

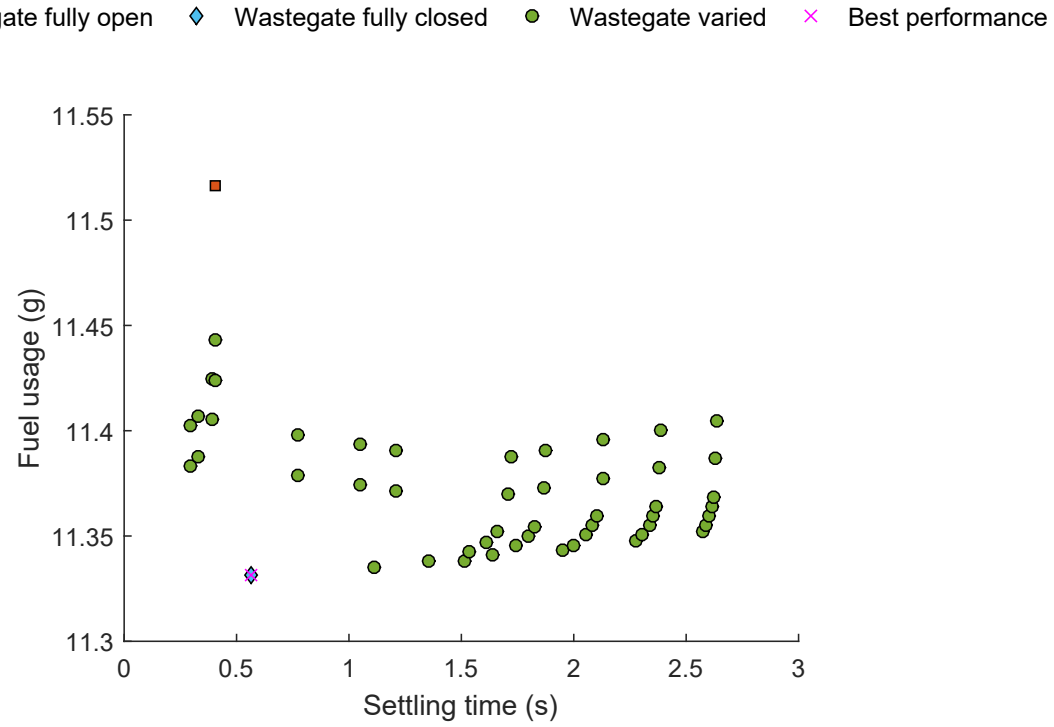


(a) Fuel usage vs. settling time for all wastegate timing tests

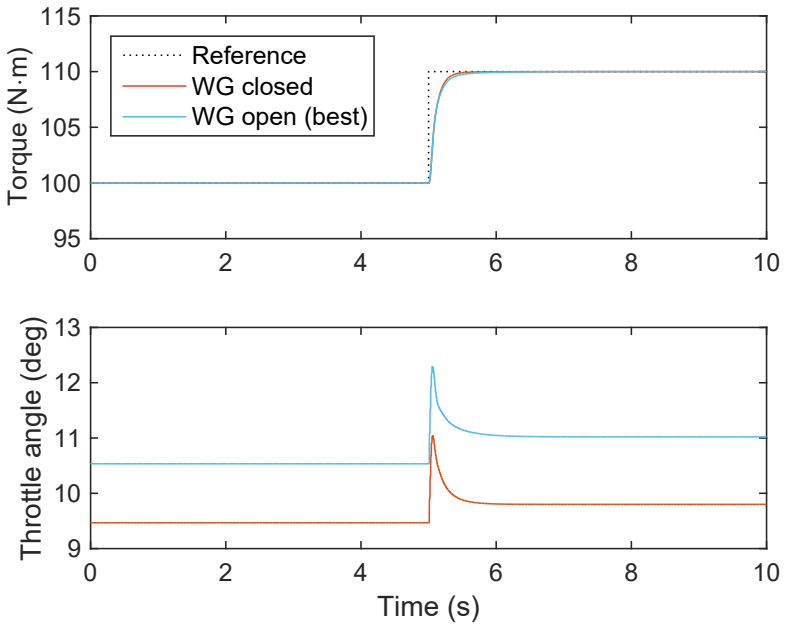


(b) Time response for selected wastegate timing tests

Figure B.14: Torque setpoint change from 60 N·m to 210 N·m

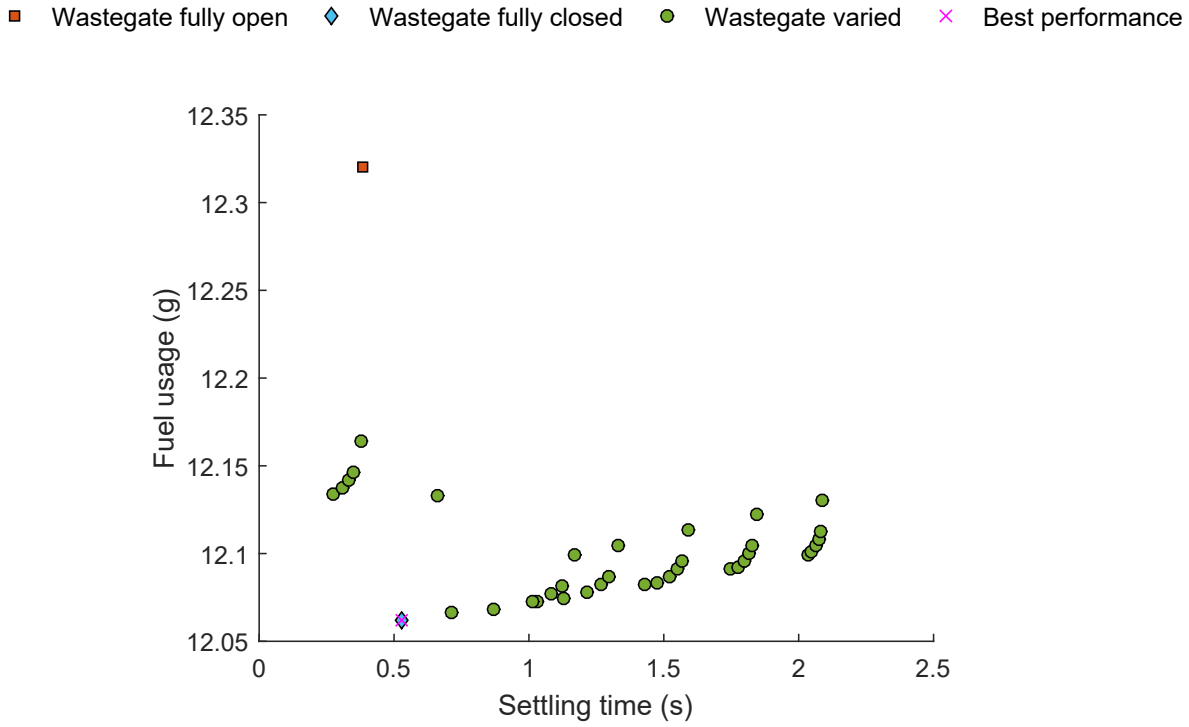


(a) Fuel usage vs. settling time for all wastegate timing tests

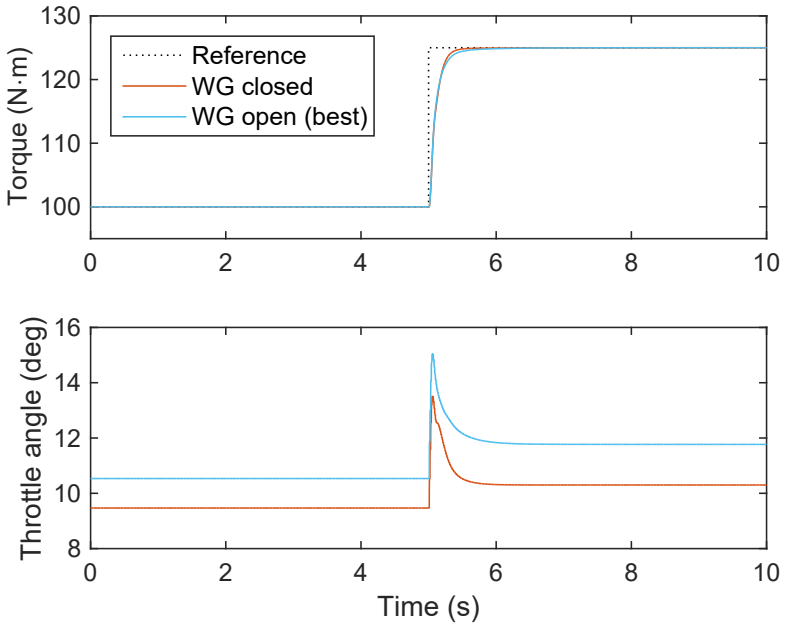


(b) Time response for selected wastegate timing tests

Figure B.15: Torque setpoint change from 100 N·m to 110 N·m

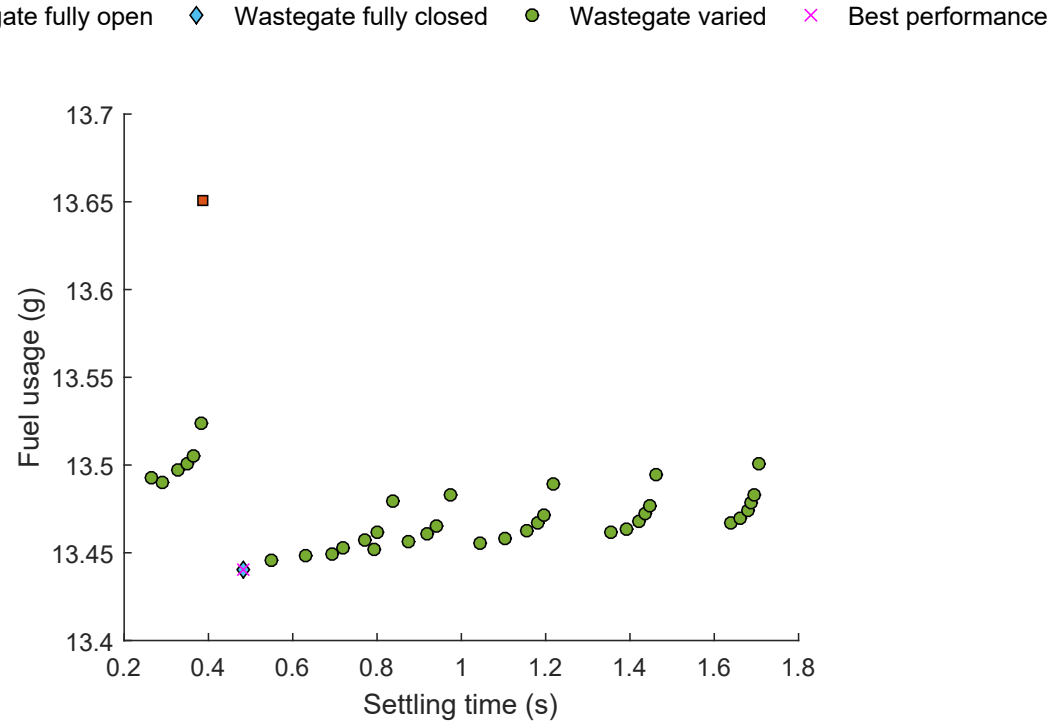


(a) Fuel usage vs. settling time for all wastegate timing tests

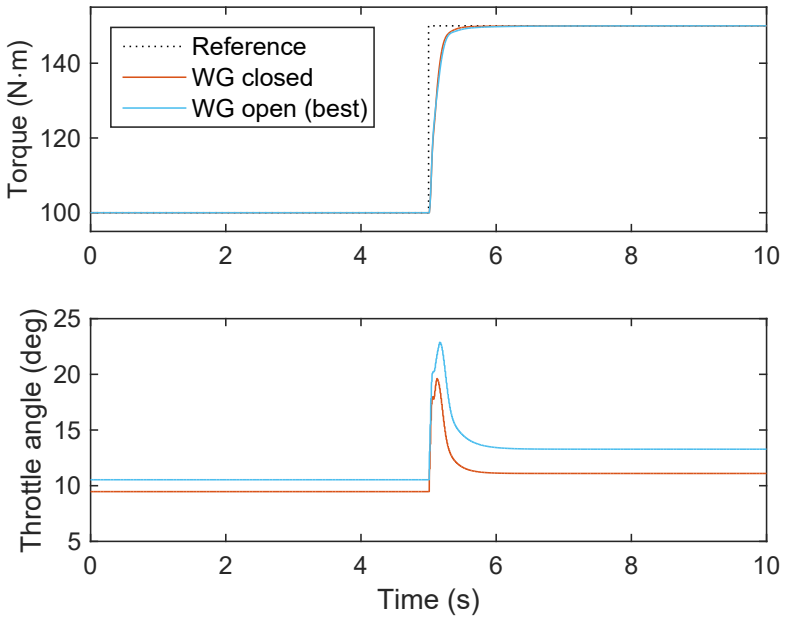


(b) Time response for selected wastegate timing tests

Figure B.16: Torque setpoint change from 100 N·m to 125 N·m

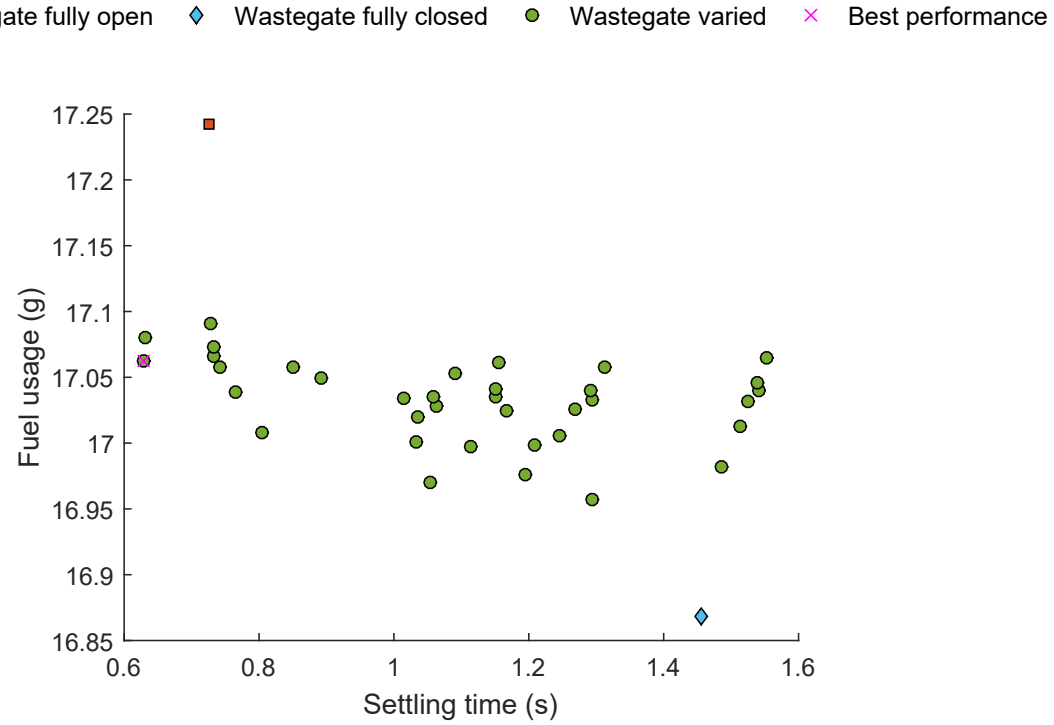


(a) Fuel usage vs. settling time for all wastegate timing tests

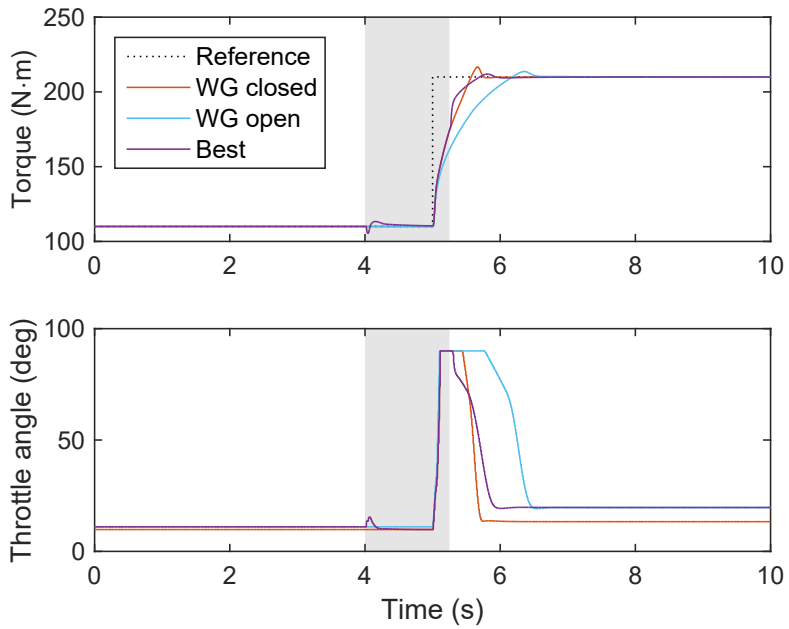


(b) Time response for selected wastegate timing tests

Figure B.17: Torque setpoint change from 100 N·m to 150 N·m

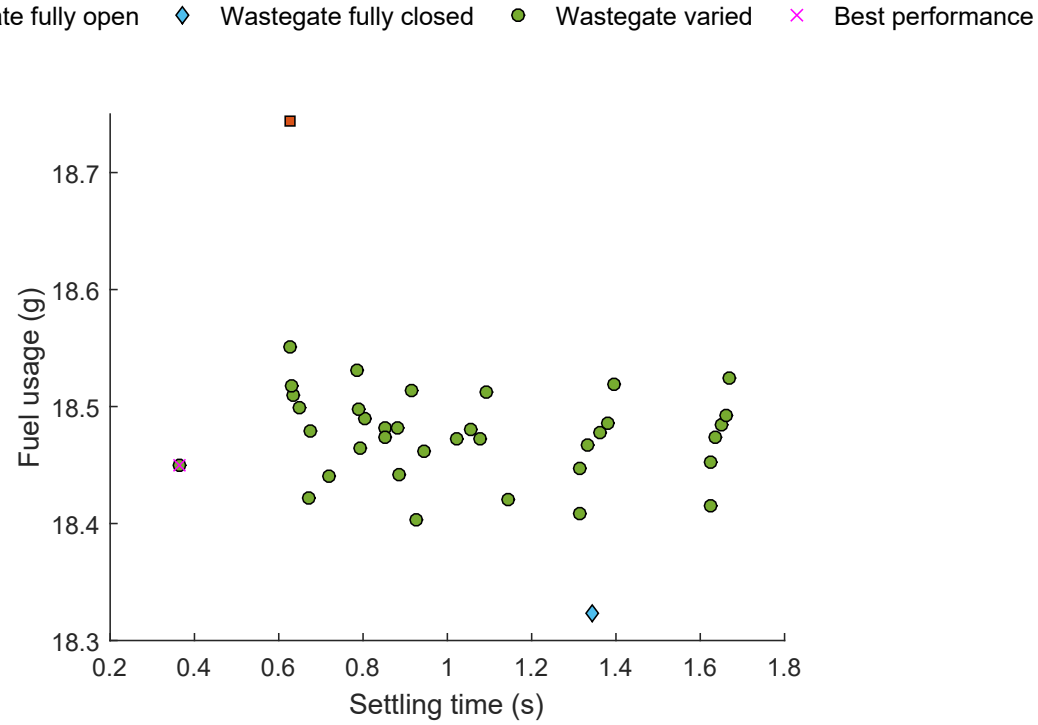


(a) Fuel usage vs. settling time for all wastegate timing tests

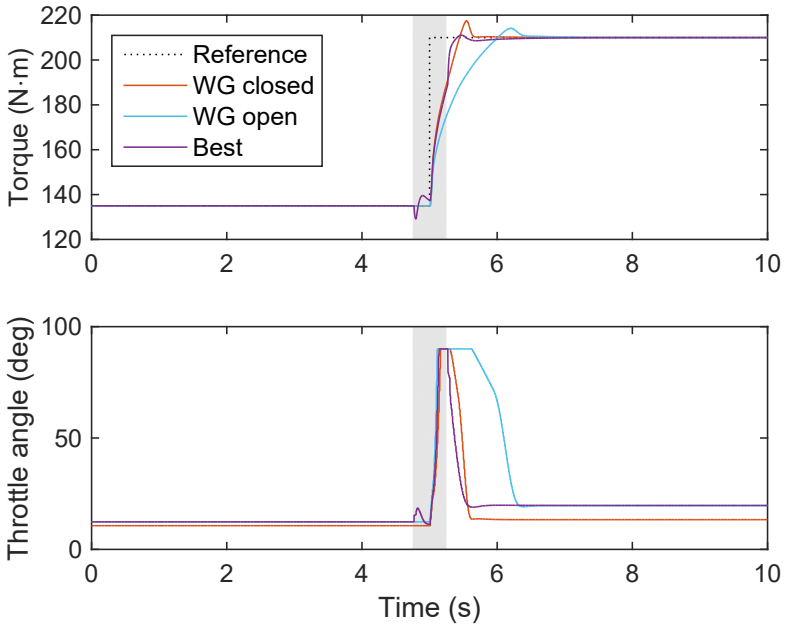


(b) Time response for selected wastegate timing tests

Figure B.18: Torque setpoint change from 110 N·m to 210 N·m

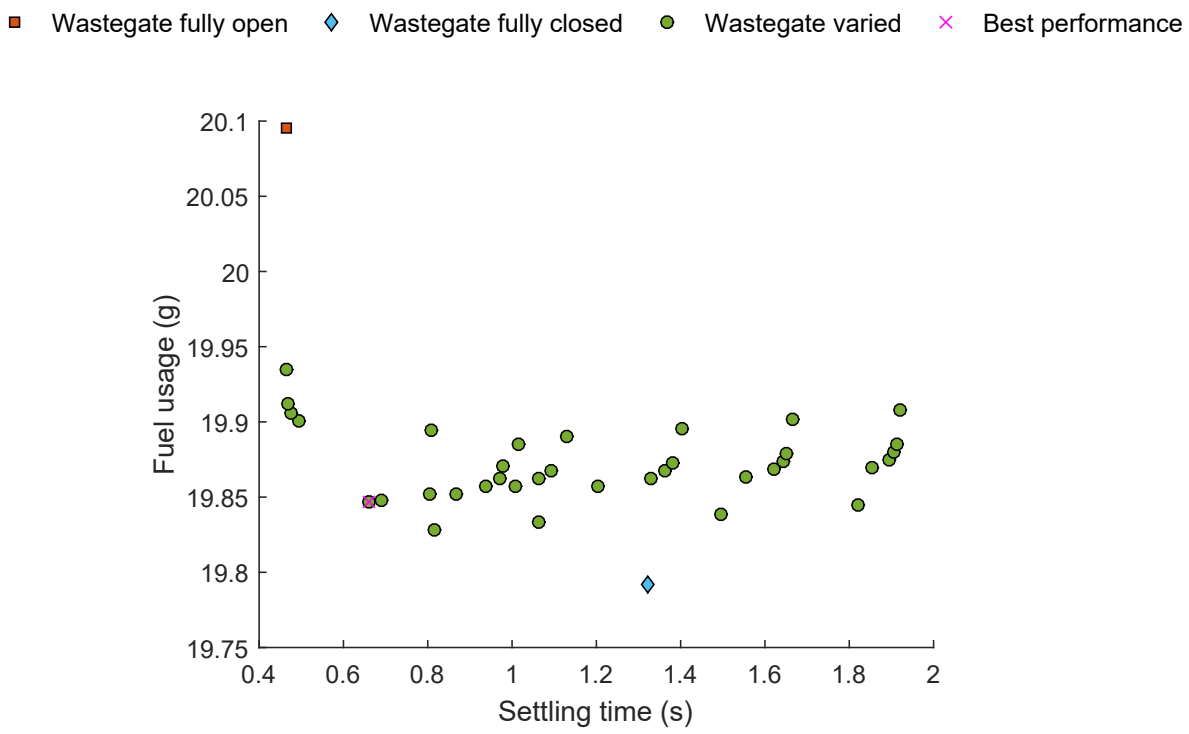


(a) Fuel usage vs. settling time for all wastegate timing tests

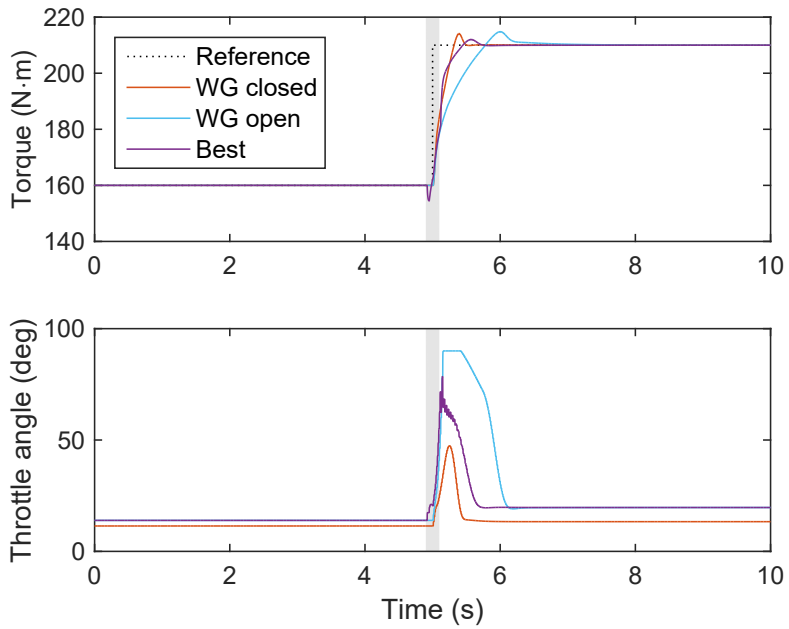


(b) Time response for selected wastegate timing tests

Figure B.19: Torque setpoint change from 135 N·m to 210 N·m

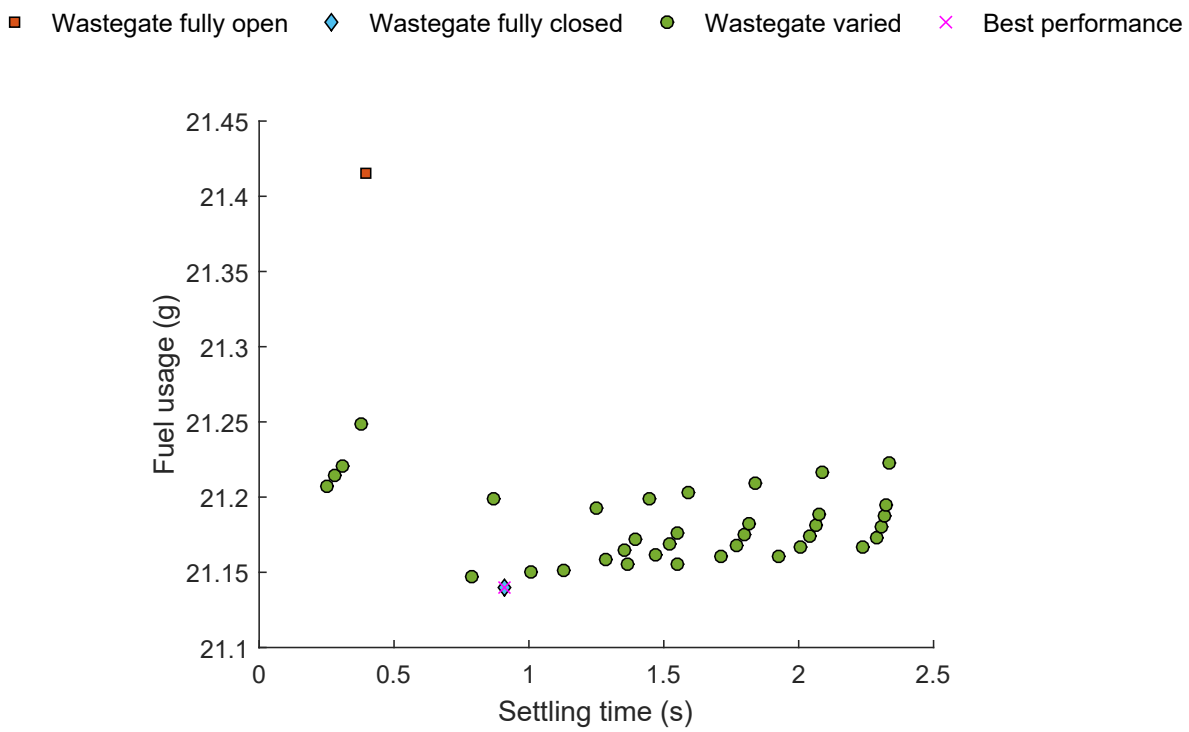


(a) Fuel usage vs. settling time for all wastegate timing tests

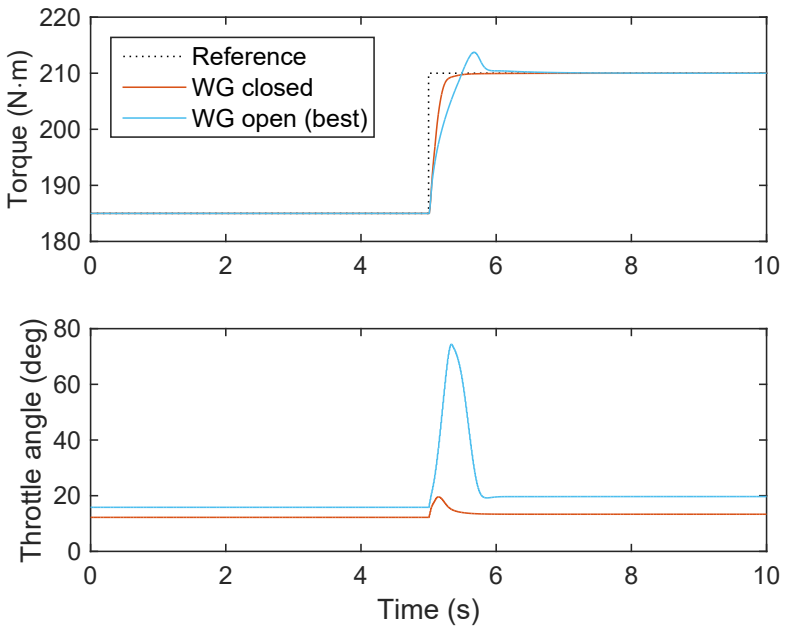


(b) Time response for selected wastegate timing tests

Figure B.20: Torque setpoint change from 160 N·m to 210 N·m

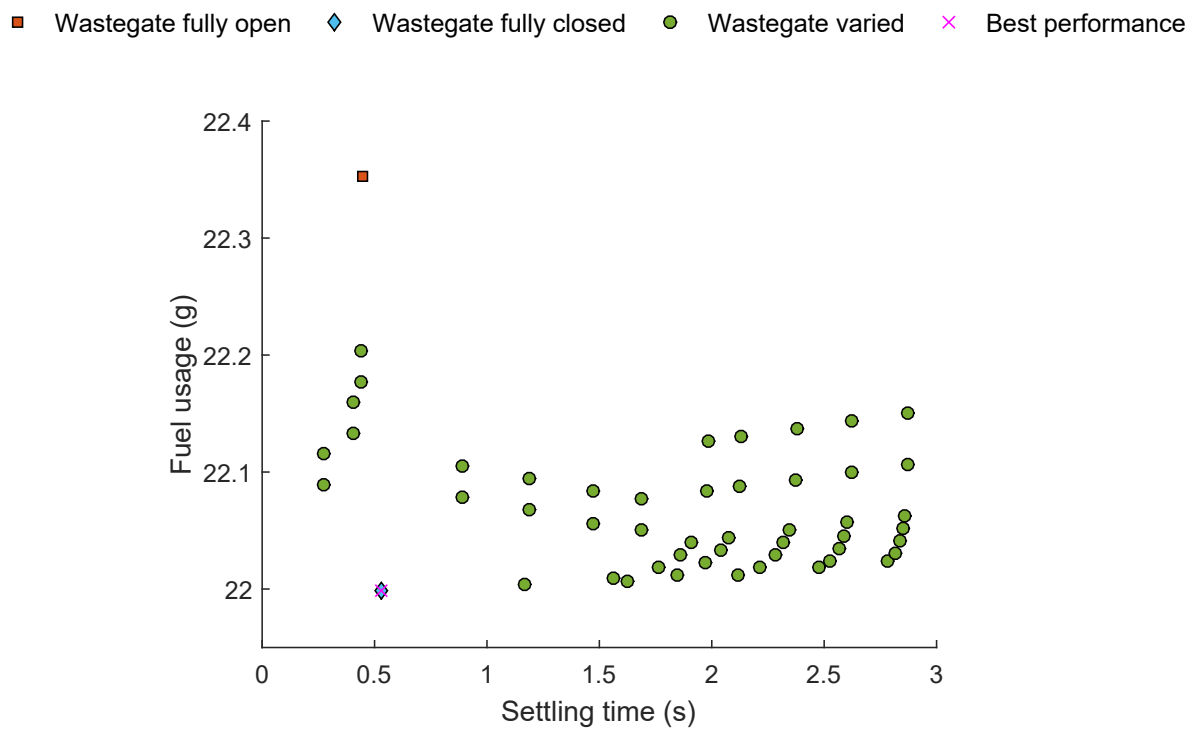


(a) Fuel usage vs. settling time for all wastegate timing tests

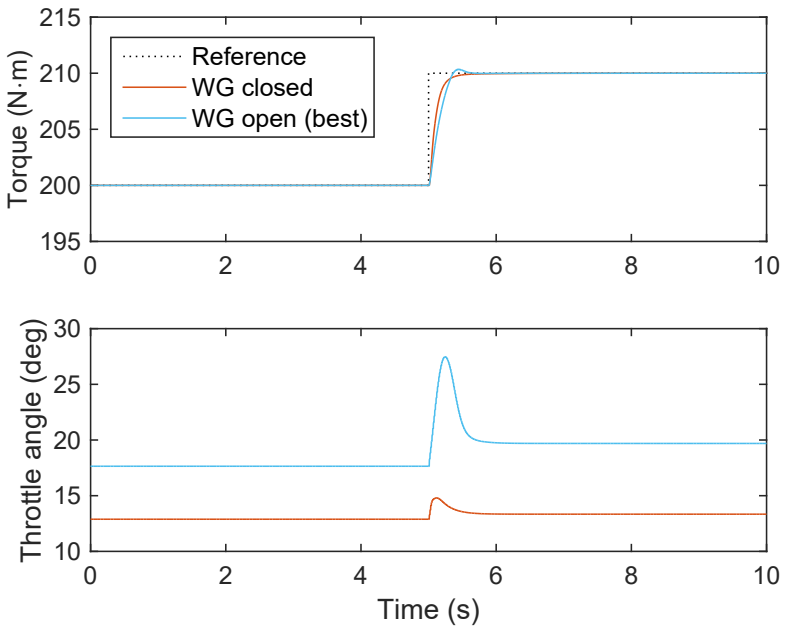


(b) Time response for selected wastegate timing tests

Figure B.21: Torque setpoint change from 185 N·m to 210 N·m



(a) Fuel usage vs. settling time for all wastegate timing tests



(b) Time response for selected wastegate timing tests

Figure B.22: Torque setpoint change from 200 N·m to 210 N·m