Low Complexity Model Predictive Control of a Diesel Engine Airpath

by

Mike Huang

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor in Philosophy
(Aerospace Engineering)
in the University of Michigan
2016

Doctoral Committee:

Professor Ilya V. Kolmanovsky, Chair
Associate Professor Ella M. Atkins
Executive Engineer Kenneth R. Butts, Toyota Technical Center
Professor Jing Sun

© Mike Huang 2016

To Li Huanyuan.

ACKNOWLEDGMENTS

I would like to give my sincere thanks to my advisor, Professor Ilya Kolmanovsky, for his encouragement, support, and guidance throughout my Ph.D. experience, with which this dissertation has been made possible. It has been a great honor to have learned from and worked with Professor Ilya Kolmanovsky and I look forward to our future collaborations.

Many thanks go to my dissertation committee member, Dr. Ken Butts at Toyota Technical Center in Ann Arbor, for the valuable mentorship he has provided me throughout this whole process and for opening the door and encouraging me to start this journey in the first place. I would also like to thank my dissertation committee members, Professor Ella Atkins and Professor Jing Sun, for their valuable insights and suggestions which have contributed to this dissertation.

I wish to thank the many talented engineers at Toyota Technical Center in Ann Arbor and at Toyota Motor Corporation in Japan who have supported this work through modeling, tool development, experiment setup, discussions, and etc. I wish to acknowledge Toyota Motor Corporation which has provided the opportunity and funding to perform interesting and valuable research. Specifically, I would like to thank Dr. Hayato Nakada at Toyota Motor Corporation for providing a sharp perspective and direction for this research.

I would also like to thank Professor Asen Dontchev for valuable discussions that have inspired some of the work of this dissertation. Additionally, thank you Richard Choroszucha for serving as an indispensable sounding board for my, often times eccentric, ideas.

Finally, I would like to thank my family and friends, without whom I would not be where I am today.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

χ_{EGR} exhaust gas recirculation rate
γ specific heat ratio
ω_t turbocharger speed
c_p specific heat at constant pressure
c_v specific heat at constant volume
d_c compressor diameter
N_e engine speed
p_a ambient pressure
p_{ex} absolute exhaust pressure
p_{in} absolute intake pressure
p_{pre} prethrottle pressure
R_{air} specific gas constant for air
T_a ambient temperature
T_{eco} exhaust gas recirculation cooler out temperature
T_{ico} intercooler out temperature
T_{in} intake temperature
u_{EGR} exhaust gas recirculation valve position
u_{th} exhaust gas recirculation throttle position
u_{VGT} variable geometry turbine position
W_c compressor flow
W_f fuel rate
W_{cyl} cylinder flow
W_{EGR} exhaust gas recirculation valve flow
W_{th} throttle flow
C/GMRES Continuation and Generalized Minimum Residual Method
CAFE Corporate Average Fuel Economy
CI Compression Ignition
CLF Control Lyapunov Function
CMPC Contractive Model Predictive Control
CPU Central Processing Unit
DAP Diesel Air Path
ECU Engine Control Unit
EMPC Economic Model Predictive Control
EPA Environmental Protection Agency
GHG Greenhouse Gas
ICE Intermittent Constraint Enforcement

IMEP Indicated Mean Effective Pressure
KKT Karush-Kuhn-Tucker
LMPC Linear Model Predictive Control
LP Linear Program
LPV Linear Parameter Varying
MAF Mass Air Flow
MAP Manifold Absolute Pressure
MIMO Multiple-input Multiple-output
MIQP Mixed Integer Quadratic Program
MPC Model Predictive Control
MY Model Year
NEDC New European Drive Cycle
NHTSA National Highway Traffic Safety Administration
NMOG Non-methane Organic Gases
NMPC Nonlinear Model Predictive Control
NOx Nitrogen Oxides
PM Particulate Matter
PMP Pontrygin Maximum Principle
PWA Piecewise-Affine
PWM Pulse-Width-Modulation
QP Quadratic Program
RPI Robust Postively Invariant
SI Spark Ignition
SISO Single-input Single-output
SQP Sequential Quadratic Program
VGT Variable Geometry Turbocharger
WLTP World harmonized Light vehicles Test Procedure

ABSTRACT

Low Complexity Model Predictive Control of a Diesel Engine Airpath

by

Mike Huang

Chair: Ilya V. Kolmanovsky

The diesel air path (DAP) system has been traditionally challenging to control due to its highly coupled nonlinear behavior and the need for constraints to be considered for driveability and emissions. An advanced control technology, model predictive control (MPC), has been viewed as a way to handle these challenges, however, current MPC strategies for the DAP are still limited due to the very limited computational resources in engine control units (ECU). A low complexity MPC controller for the DAP system is developed in this dissertation where, by "low complexity," it is meant that the MPC controller achieves tracking and constraint enforcement objectives and can be executed on a modern ECU within 200 μsec, a computation budget set by Toyota Motor Corporation.

First, an explicit MPC design is developed for the DAP. Compared to previous explicit MPC examples for the DAP, a significant reduction in computational complexity is achieved. This complexity reduction is accomplished through, first, a novel strategy of intermittent constraint enforcement. Then, through a novel strategy of gain scheduling explicit MPC, the memory usage of the controller is further reduced and closed-loop tracking performance is improved. Finally, a robust version of the MPC design is developed which is able to enforce

constraints in the presence of disturbances without a significant increase in computational complexity compared to non-robust MPC. The ability of the controller to track set-points and enforce constraints is demonstrated in both simulations and experiments. A number of theoretical results pertaining to the gain scheduling strategy is also developed.

Second, a nonlinear MPC (NMPC) strategy for the DAP is developed. Through various innovations, a NMPC controller for the DAP is constructed that is not necessarily any more computationally complex than linear explicit MPC and is characterized by a very streamlined process for implementation and calibration. A significant reduction in computational complexity is achieved through the novel combination of Kantorovich's method and constrained NMPC. Zero-offset steady state tracking is achieved through a novel NMPC problem formulation, rate-based NMPC. A comparison of various NMPC strategies and developments is presented illustrating how a low complexity NMPC strategy can be achieved.

Chapter 1

Introduction

Emissions and fuel economy regulations for modern automobiles are becoming increasingly stringent. On April 1, 2010, the U.S. Environmental Protection Agency (EPA) and the National Highway Traffic Safety Administration (NHTSA) established a national program of standards for greenhouse gas (GHG) emissions and Corporate Average Fuel Economy (CAFE) for Model Year (MY) 2012 - 2016 light-duty vehicles [104]. These light-duty vehicles include passenger cars, light-duty trucks, and medium-duty passenger vehicles and are currently responsible for nearly 60% of U.S. transportation related petroleum use and GHG emissions [105]. These standards require light-duty vehicles to meet an estimated combined average of 250 grams per mile of carbon dioxide (CO_2) by MY 2016 which translates to 35.5 miles per gallon (mpg) if the CO_2 level is to be met purely through fuel economy improvements.

Following the establishment of GHG and CAFE regulations for MY 2012 - 2016, President Obama, on May 21, 2010, issued a continued request to the EPA and NHTSA to create a program for national fuel economy standards for vehicle MY 2017 - 2025:

> "I [President Barack Obama] request that the Administrators of the EPA and NHTSA develop, through notice and comment rulemaking, a coordinated national program under the CAA [Clean Air Act] and the EISA [Energy Independence and Security Act of 2007] to improve fuel efficiency and to reduce green-

Table 1.1: Fleet-wide emissions standards under the footprint-based CO_2 standards [g/mi] and corresponding fuel economy [mpg]. Source: [105].

	2016	2017	2018	2019	2020	2021	2022	2023	2024	2025
Passenger cars [g/mi]	225	212	202	191	183	172	164	157	150	143
Light trucks [g/mi]	298	295	285	277	269	249	237	225	214	203
Combined cars & light trucks [g/mi]	250	243	232	222	213	199	190	180	171	163
Combined cars & light trucks [mpg]	35.5	36.6	38.3	40.0	41.7	44.7	46.8	49.4	52.0	54.5

> house gas emissions of passenger cars and light-duty trucks of model years 2017 - 2025... The program should also seek to achieve substantial annual progress in reducing transportation sector greenhouse gas emissions and fossil fuel consumption, consistent with my Administration's overall energy and climate security goals, through the increased domestic production and use of existing, advanced, and emerging technologies [81]..."

In response to the Presidential Memorandum [81] the EPA and NHTSA have established the much publicized standard of 54.5 mpg for MY 2025 light-duty vehicle fleet average fuel economy [105]. Table 1.1 shows the GHG and CAFE standards progression from MY 2016 to MY 2025 for different vehicle classes based on vehicle footprint. Table 1.2 shows the MY 2025 GHG and fuel economy standards for representative MY 2012 vehicles of various sizes.

The reports [3, 4, 59] survey the outlook of meeting the MY 2025 fuel economy targets and give an overview of strategies that would be necessary, including regulatory actions, behavioral shifts of consumers and automotive manufacturers, and technological advancements. In [3], it was concluded that a 30%–50% improvement in vehicle fuel economy can be achieved by 2035. This improvement can be achieved in the short term through improvements in gasoline and diesel engines, transmissions, gasoline hybrid technologies, and reductions in vehicle weight and drag. In the long term, plug-in hybrid and hydrogen fuel cell vehicles would need to achieve enough market penetration to begin to make an impact

Table 1.2: MY 2025 CO_2 and fuel economy targets for representative MY 2012 vehicles. Source: [105].

Vehicle Type	Example Model	Example Model Footprint [sq. ft.]	EPA CO_2 Standard [g/mi]	NHTSA Fuel Economy Standard [mpg]
Example Passenger Cars				
Compact car	Honda Fit	40	131	61.1
Mid-size car	Ford Fusion	46	147	54.9
Full-size car	Chrysler 300	53	170	48.0
Example Light-duty Trucks				
Small SUV	4WD Ford Escape	43	170	47.5
Midsize crossover	Nissan Murano	49	188	43.4
Minivan	Toyota Sienna	56	209	39.2
Large pickup truck	Chevy Silverado	67	252	33.0

on fleet average fuel economy.

Currently, only about 18%–25% of the energy from fuel is actually transferred to the wheels in a conventional gasoline automobile during combined city and highway driving [107]. Of the energy delivered to the wheels, 9%–12% of the total fuel energy is dissipated as wind resistance, 5%–7% is dissipated as rolling resistance, and only 5%–7% is used to overcome the vehicle inertia, i.e., drive. The majority of the losses between the fuel tank and the wheels occur in the engine (68%–72%) due to thermal (58%–61%), combustion (3%), engine friction (3%), and pumping losses (3%). Thus there is a very large opportunity to improve vehicle fuel efficiency through improving engine efficiency. As it has been in the past, advanced control technologies in synergy with advanced hardware technologies will be a key component for meeting the strict and challenging regulatory standards of the future [15]. The work described in this dissertation focuses on the development of an advanced control technology that would enable more efficient use of engines by enabling them to operate close to their limits. Specifically, the control technology that is enhanced through the work described in

this dissertation is Model Predictive Control (MPC). In this work, MPC will be applied to diesel engines which are used in many different domains besides automotive transportation, including marine and locomotive, and accounts for more than 95 percent of all U.S. freight transportation [99].

The remainder of this chapter is organized as follows. Section 1.1 describes current trends in powertrain control: both engine hardware trends and the current state of powertrain control in industry. Section 1.2 gives a brief introduction to MPC and addresses the question "why should MPC be used for engine control?" Section 1.3 describes the main diesel air path (DAP) application example and control objective that will be used throughout this dissertation, the associated history, and the associated challenges. Section 1.4 gives an overview of the research contributions that overcome, in part, those aforementioned challenges. Finally, Section 1.5 gives an outline of the remaining chapters of this dissertation.

1.1 Trends in Powertrain Control

Engine downsizing has been a popular trend that can lead to higher fuel economy through reduced weight, friction losses, heat losses, and pumping losses. In fact, according to [59], aggressive vehicle-wide downsizing is required to meet current and future CAFE standards. The results in [59] suggest that if vehicle weight and horsepower were kept at constants levels between 1980 and 2006, fuel economy could have risen by nearly 60% due to progress in fuel economy related technologies. In contrast, the actual fuel economy increase was less than 6.5% from 1980 to 2004 as U.S. market preferences drove average horsepower up by 80% and 99% for cars and light-duty trucks, respectively, and average vehicle weight up by 12% and 26% for cars and light-duty trucks, respectively. Additionally, from 1980 to 2004, light-duty truck sales rose from 20% of total passenger vehicles to over 51%.

To maintain the horsepower that customers currently demand, downsized engines are now commonly coupled with turbochargers, [21, 29, 88, 100, 103, 117], which are able to con-

vert the energy that would typically be lost as exhaust heat, [29, 103], to air boosting and higher engine torque. In one extreme case, [103], a turbocharged and supercharged engine with a downsizing factor of 60% was able to achieve the same steady-state torque output as a naturally aspirated Jaguar Land Rover AJ133 5.0 liter V8 engine with 23% improved fuel consumption. As was noted in [103], the limit of extreme downsizing is not the combustion system but rather the air charge system. The challenge is to develop a system that can provide sufficient boost at low speed and low load without overboosting at high speed and high load [15, 116]. Actuators such as wastegates or variable geometry turbochargers (VGT) could be used to overcome this challenge, however the use of these actuators introduce increased complexity in control design and calibration. Achieving adequately fast transient response with turbochargers is another challenge, commonly referred to "turbo-lag." The turbo-lag is a delay caused by the turbocharger inertia and affects the time required to spin up the turbine and provide boost. Controllers aimed at reducing turbo-lag need to simultaneously meet boost pressure overshoot limits which would otherwise lead to undesirable torque disturbances [15, 57, 66, 91].

In addition to increasingly stringent CAFE standards, the EPA will also begin to phase in Tier 3 tailpipe emissions standards which, by 2025, will reduce the sum of non-methane organic gases (NMOG) and nitrogen oxides (NOx) by approximately 80% compared to today's fleet average standard and reduce particulate matter (PM) by 70% compared to today's standard [106]. Exhaust gas recirculation (EGR) is a popular strategy in reducing NOx in both spark ignition (SI) and compression ignition (CI, e.g., diesel) engines [91]. This strategy recirculates cooled exhaust gas back into the intake manifold and cylinders replacing O_2 with CO_2 which reduces the specific heat capacity of the in-cylinder gas mixture and peak in-cylinder temperature during combustion. This results in reduced NOx emissions. However, the NOx reduction achieved through EGR simultaneously increases PM [34, 91]. Furthermore, emissions reductions typically result in decreased fuel economy [3].

Diesel engines are also an attractive option to meet future fuel economy regulations be-

cause they are more fuel efficient compared to their gasoline counterparts by about 30% [3]. However, diesel engines typically utilize both turbocharger technologies and EGR technologies which lead to complex nonlinear interactions, e.g., DC gain reversal [38, 113], and performance and emissions trade-offs.

Regardless of the engine configuration, the industry trend has been to add new actuators and sensors to meet regulatory requirements. The interactions between the many components are difficult to optimize using current control methods. The current practice is based on wrapping individual feedback loops, e.g., PID, around subsystem actuators and sensors. The dynamic interactions are then reduced by detuning those system controllers. Furthermore many maps, e.g., for feed-forward commands and feedback gains, are required to handle the highly nonlinear nature of the engine. In addition, a large calibration effort from experienced calibrators is required. Thus there has been recent and growing interest in the utilization of MPC for its ability to simultaneously coordinate multiple actuators and objectives to improve performance, enforce reliability and emissions constraints, and operate near system boundaries without detuning, all in a systematic manner.

1.2 Why Model Predictive Control?

MPC is an optimization based control method that has been used in many applications in industry, primarily for chemical process control, see [12, 90] and references therein. The fundamental idea is that a model of the plant is used to predict future outputs with respect to a sequence of future inputs. Then, numerical optimization methods are used to compute the future inputs that minimize a cost function that captures the desired system behavior. The first element of the control sequence is then applied to to the plant. This procedure is then repeated at the next time step and all future time steps. Figure 1.1 illustrates the MPC process at time step k where the control objective is to drive the system output, y, to a desired set-point, y_{ref}. The process is then repeated at the next time step, $k+1$, see

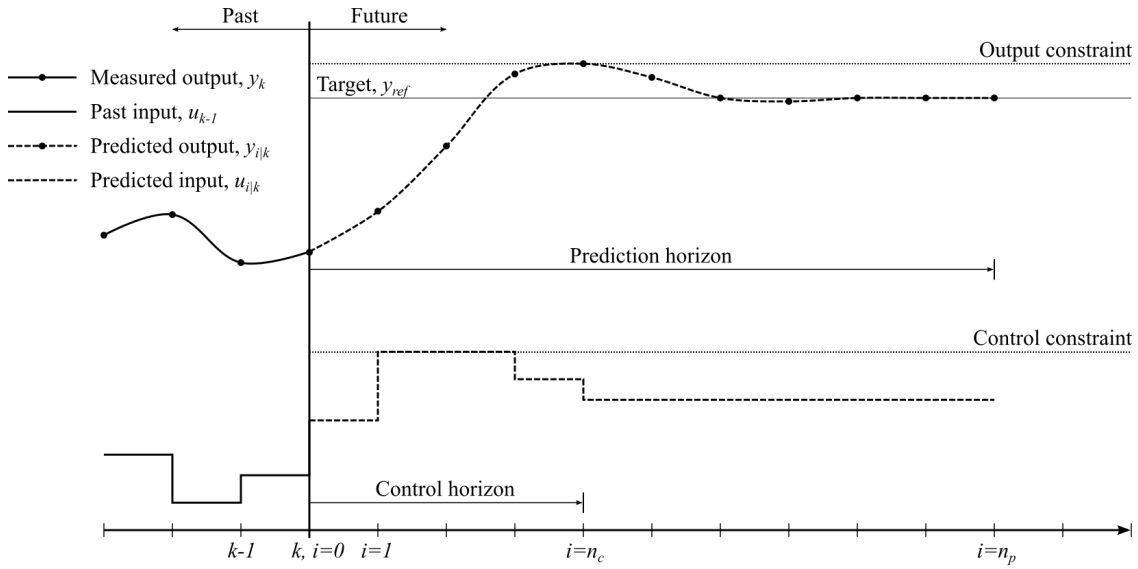

Figure 1.1: Illustration of the model predictive control process at time step k. The output is measured at time k. An optimization algorithm then chooses $n_c + 1$ control moves, $u_{i|k}$, at future steps in the control horizon $i \in \{0, 1, ...n_c\}$ which minimizes cumulative the error to the target, $y_{i|k} - y_{ref}$, over the prediction horizon $i \in \{0, 1, ...n_c\}$, $n_p \leq n_c$. This optimization is performed subject to control and output constraints. The optimal control $u_{0|k}$, i.e., the first element of the optimized control sequence, is then applied at time step k.

Figure 1.2. Throughout this dissertation, the subscript notation $i|k$ will be used where k denotes the current sample time and i denotes the time step in the prediction horizon where $i = 0$ corresponds to the current sample time k.

There are numerous advantages of using MPC for engine control versus traditional control methods, e.g., PID.

- Complex multiple-input multiple-output (MIMO) interactions, e.g., those occurring in modern engines, are inherently handled through model based optimization. This leads to better closed-loop performance compared to detuned coupled single-input single-output (SISO) control strategies. The required calibration time will also be reduced, e.g., the detuning process necessary for SISO strategies is not necessary for MIMO strategies.

- Closed-loop performance can be maximized while enforcing to constraints directly through the optimization process. Other control strategies often require detuning to

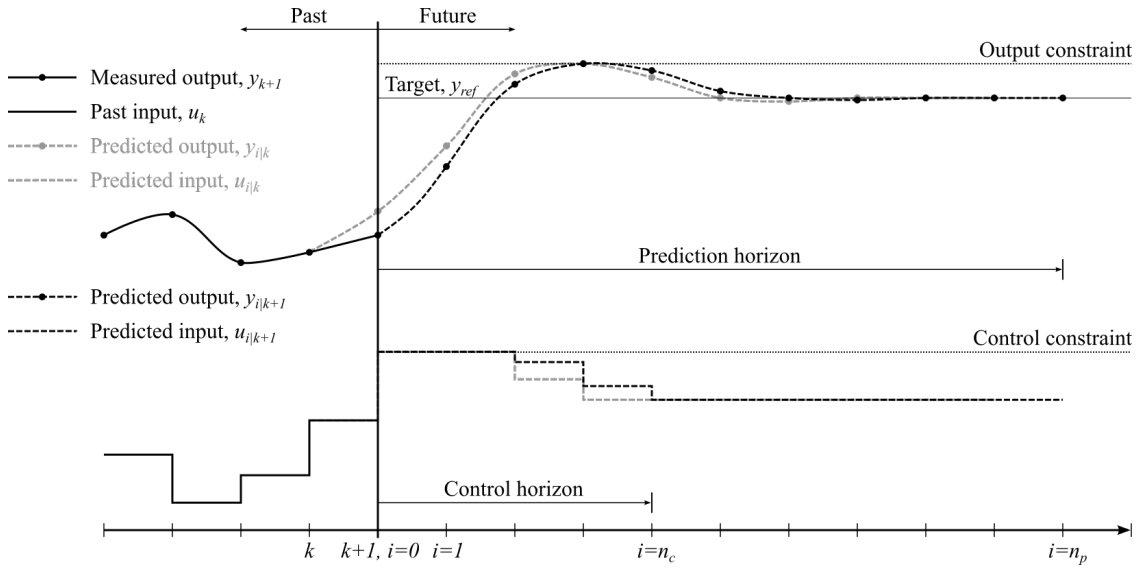

Figure 1.2: Illustration of the model predictive control process at time step $k+1$. The output is measured at time $k+1$. An optimization algorithm then chooses n_c+1 control moves, $u_{i|k+1}$, at future steps in the control horizon $i \in \{0, 1, ...n_c\}$ which minimizes cumulative the error to the target, $y_{i|k+1} - y_{ref}$, over the prediction horizon $i \in \{0, 1, ...n_c\}$, $n_p \leq n_c$. This optimization is performed subject to control and output constraints. The optimal control $u_{0|k+1}$, i.e., the first element of the optimized control sequence, is then applied at time step $k+1$.

avoid constraint violations. Note that reference governors have also been recently used for diesel engine control to handle constraints [79]. However, the ultimate goal is to simultaneously maximize performance and enforce constraints, while reference governors only enforce constraints.

- There is typically a direct correlation between terms in the cost function and desired closed-loop behavior which facilitates tuning/calibration.

There are also challenges in exploiting MPC for practical applications. First a model needs to be developed that can be used for prediction. Assuming that a model can be obtained, it is still non-trivial to implement the control law, particularly when constraints are considered, because of the potentially large amount of computations and memory that is required to solve the associated optimization problem online at every time step. Note that embedded micro-controllers are very limited compared to the computing power of modern

Table 1.3: Evaluation of methods for powertrain control: ◯ – good, Δ – intermediate, and × – bad.

Control Method	Objective Formulation	MIMO	Control of Nonlinear Systems	Constraint Enforcement	Computation Cost	Maturity
LQR	◯−	◯	Δ	×	◯	◯
H_∞	◯−	◯	Δ	×	◯	◯
Linear Reference Governor	×	◯	Δ	◯	◯-	◯
Nonlinear Reference Governor	×	◯	◯	◯	Δ+	Δ
Linear MPC	◯−	◯	Δ	◯	Δ+	◯−
Nonlinear MPC	◯	◯	◯	◯	Δ	Δ
Economic MPC	◯+	◯	◯	◯+	Δ-	×
Feedback Linearization	×	◯	◯	×	◯	◯
PID	×	Δ	Δ	×	◯	◯
Heuristic	×	Δ−	Δ	×	◯	◯

computers, e.g., a modern mid-range engine control unit (ECU) has on the order of 10 times slower clock speed compared to modern personal computers [26].

Table 1.3 shows an evaluation of various methods that can be used for powertrain control. Methods that are currently used in industry are quite basic, exploit primarily PID and heuristic-rule-based control, and require large calibration efforts to achieve the desired behavior for strongly coupled multivariable plants such as the DAP. Of the methods considered in Table 1.3, reference governors and MPC can be used to enforce constraints, however reference governors are used only to enforce constraints rather than improve closed-loop performance in general. Hereafter, linear MPC (LMPC) refers to MPC design based on a linear model, a quadratic or linear cost function, and linear constraints. Note that LMPC is actually a nonlinear controller, unless the system has no constraints. LMPC is a fairly mature control

technology, insofar as the implementation framework and supporting theoretical results are available. For example, offset-free, steady-state tracking versions of LMPC are available which typically utilize a disturbance observer or a linear-velocity form (referred to as rate-based form in this dissertation), [85, 114]. Robust versions, in the sense that constraints can be satisfied under model uncertainties, are available through tube-MPC, [64, 75]. A priori stability guarantees using terminal state penalties and terminal state constraints exist [72], and a posteriori stability analysis techniques are also available [89]. Explicit MPC can also be used to improve online computation times, [5]. LMPC typically reduces to solving a quadratic program (QP) or linear program (LP). Solving the QP or LP for applications that require fast sampling, e.g., powertrain control, is still non-trivial, particularly under many constraints. Naturally, nonlinear MPC (NMPC), see [30] and references therein, can offer better closed-loop performance when the underlying system is nonlinear compared to LMPC which typically handles nonlinearities through ad hoc methods. However, NMPC is less mature in contrast to LMPC. For example, it is not yet clear how offset-free, steady-state tracking can be accomplished with NMPC. Furthermore, the NMPC optimization problem is more difficult to solve within real-time limitations. Economic MPC (EMPC) is an emerging area of research which has been viewed as an enabling technology for the next industrial revolution, [2], or "Industry 4.0," a term originally used by the German government to promote the vision of the "Smart Factory." With EMPC the traditional objective functions, e.g., for set-point tracking, are replaced by ones that directly target some economical value. For powertrain control, this means that actuators will be coordinated to directly improve fuel economy and reduce emissions, rather than traditionally track some intermediate set-points which are separately calibrated to achieve economic goals.

The work presented in this dissertation focuses on overcoming the current challenges, particularly the computational challenges, of applying LMPC and NMPC to powertrain control. This work can be viewed as a stepping stone toward an ultimate goal of EMPC for the powertrain. The following Sections 1.2.1-1.2.3 give a brief introduction to MPC.

Throughout this dissertation, complexity analysis of various MPC strategies will be given using Big-O and little-o notation.

Definition 1.1: For a function $f : \mathbb{R} \to \mathbb{R}$ and $g : \mathbb{R} \to \mathbb{R}$, $f(x)$ is $\mathcal{O}(g(x))$ if and only if there exists $M \geq 0$ and $x_0 \geq 0$ such that $f(x) \leq Mg(x)$ for all $x \geq x_0$.

Typically Big-O notation, $\mathcal{O}$ in Definition 1.1, is used for complexity analysis of algorithms, however, Big-O notation is not well suited for functions of multiple variables and the existing definitions for Big-O notation of multiple variables are commonly inconsistent, [37]. Furthermore, one of the main strategies that will be used in this dissertation for managing computational complexity of MPC is to keep the problem size small, thus it is not particularly insightful to only examine the asymptotic complexity of algorithms. Thus we will also introduce little-o notation in this dissertation for the purpose of gaining insight into the computational complexity of algorithms for small problem sizes.

Definition 1.2: For a function $f : \mathbb{R}^n \to \mathbb{R}$ and $g : \mathbb{R}^n \to \mathbb{R}$, $f(x)$ is $o(g(x))$ if and only if there exists $M \geq 0$ such that $f(x) \leq Mg(x)$ for all $x \geq 0$.

1.2.1 Model Predictive Control

MPC is developed for a the discrete time plant model of the form,

$$x_{k+1} = f(x_k, u_k), \tag{1.1}$$

where k is the current sample instant and $x_k \in \mathbb{R}^{n_x}$, $u_k \in \mathbb{R}^{n_u}$, and $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$. MPC is based on minimizing a cost functional of the form,

$$J = \phi(x_{N|k}) + \sum_{i=0}^{N-1} l(x_{i|k}, u_{i|k}), \tag{1.2}$$

where $l : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}$ is the incremental cost function, $\phi : \mathbb{R}^{n_x} \to \mathbb{R}$ is the terminal cost function, and N is the prediction horizon. The minimization of (1.2) with respect to $x_{i|k}$ with $i \in \{1, ..., N\}$ and $u_{i|k}$ with $i \in \{0, ..., N-1\}$ is performed subject to equality and

inequality constraints,

$$
\begin{aligned}
x_{i+1|k} &= f(x_{i|k}, u_{i|k}),\ i \in \{0, ..., N-1\}, && (1.3)\\
x_{0|k} &= x_k, && (1.4)\\
g(x_{i|k}, u_{i|k}) &= 0,\ i \in \{0, ..., N-1\}, && (1.5)\\
h(x_{i|k}, u_{i|k}) &\leq 0,\ i \in \{0, ..., N-1\}, && (1.6)\\
\psi(x_{N|k}) &\leq 0, && (1.7)
\end{aligned}
$$

where i is the running time (discrete) over the prediction horizon N and $g : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_g}$, $h : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_h}$, and $\psi : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_\psi}$. The equality constraint (1.3) enforces the state dynamics over the prediction horizon. The equality constraint (1.4) embeds the initial condition. The equality constraint (1.5) can be made more general and, e.g., used to hold the control input at some given constant value at some instances of $i \in \{0, ..., N-1\}$, typically referred to as move-blocking. The constraint (1.6) enforces any desired inequality constraints, e.g, keeping the control and state sequences inside a "safe" set. The constraint (1.7) enforces a terminal state constraint which can, e.g., be used to enforce stability. Note that more general and/or special forms of the MPC optimization problem can be written. The optimization problem (1.2)-(1.7) is typically solved trough forming the associated Karush-Kuhn-Tucker (KKT) conditions for optimality. This converts (1.2)-(1.7) into a root finding problem which can be solved using a Newton method for $x_{i|k}$, $i \in \{1, ..., N\}$, and $u_{i|k}$, $i \in \{0, ..., N-1\}$. Once (1.2)-(1.7) is solved, $u_{0|k}$ is applied at the the sample instant k. This process is then repeated at the next time step.

1.2.2 Linear Model Predictive Control

One special case of the MPC formulation (1.2)-(1.7) is the case where f is linear, g, h, and ψ are affine, and l and ϕ are quadratic. This gives the typical formulation for LMPC

where the cost functional,

$$J = x_{N|k}^T P x_{N|k} + \sum_{i=0}^{N-1} x_{i|k}^T Q x_{i|k} + u_{i|k}^T R u_{i|k}, \tag{1.8}$$

is minimized subject to the affine constraints,

$$\begin{aligned} x_{i+1|k} &= Ax_{i|k} + Bu_{i|k},\ i \in \{0, ..., N-1\}, & (1.9)\\ G_x x_{i|k} + G_u u_{i|k} - G_0 &= 0,\ i \in \{0, ..., N-1\}, & (1.10)\\ H_x x_{i|k} + H_u u_{i|k} - H_0 &\leq 0,\ i \in \{0, ..., N-1\}, & (1.11)\\ \psi_x x_{N|k} + \psi_0 &\leq 0, & (1.12) \end{aligned}$$

and initial condition embedding (1.4). The cost functional (1.8) corresponds to a regulation objective with the matrices $P = P^T \geq 0$, $Q = Q^T \geq 0$, and $R = R^T > 0$. The matrices A and B in (1.9) are the linear system dynamic matrices. The matrices G_x and G_u and vector G_0 in (1.10) correspond to an affine equality constraint. The matrices H_x and H_u and vector H_0 in (1.11) correspond to an affine inequality constraint. The matrix ψ_x and vector ψ_0 in (1.12) correspond to an affine terminal state constraint. The LMPC formulation (1.8)-(1.12) and (1.4) is a QP. There are a variety of methods that can be used to solve the QP. An evaluation of various QP solution methods is summarized in Table 1.4. Table 1.4 evaluates QP solver characteristics such as central processing unit (CPU) computation times and memory usage, code simplicity, typical optimization problem size use cases, and etc.

1.2.3 Explicit Linear Model Predictive Control

Due to the highly limited computational resources available in the ECU for powertrain applications, MPC formulations with small problem sizes are typically used. Explicit MPC, which is especially fast in CPU computation time for the small problem sizes, i.e., about 6 to 8 control moves and 8 to 12 parameters [1, 7], is then used as the QP solution technique.

Table 1.4: Evaluation of QP solution methods: ◯ – good, Δ – intermediate, and × – bad. Source: [7].

	Explicit MPC	Implicit (on-line QP)		
		active set	interior point	gradient projection
CPU computation time	◯	Δ	Δ	Δ
CPU memory usage	×	◯	◯	◯
control code	◯	Δ	×	◯
ability to estimate tight bounds for worst case computation time	◯	×	×	Δ
amount of off-line computation	×	◯	◯	◯
generates feasible and optimal solutions	◯	◯	Δ	Δ
problem size	small	medium	large	medium

Explicit MPC is a solution method that can be used when the system dynamics, (A, B, C, D), are known a priori and do not change online. Then the solution to the QP (1.8)-(1.12) and (1.4) can be computed off-line for all initial conditions for prediction, x_k. Explicit MPC is essentially an active set QP method where the solution to the associated KKT conditions are precomputed for all possible combinations of active constraints [5, 23]. This results in a piecewise-affine (PWA) control law with the following form,

$$u^\star_{0|k} = G_j x_k + F_j \text{ if } K_j x_k \leq L_j,\ j \in \{1, .., N_r\}, \tag{1.13}$$

where $u^\star_{0|k}$ is the control to be applied and is the first element of the optimal control sequence associated with the QP (1.8)-(1.12), x_k is the current sampled state, and N_r is the number of regions of the PWA control law (1.13) resulting from the different combinations of active constraints. Because the regions are generated through a combinatorial process, N_r will be $o(\mathcal{C}(Nn_h + n_\psi, Nn_u))$, where Nn_u is the total number of optimization variables, $Nn_h + n_\psi$ is the total number of inequality constraints, and $\mathcal{C}(x, y)$ denotes the operation x choose y [23].

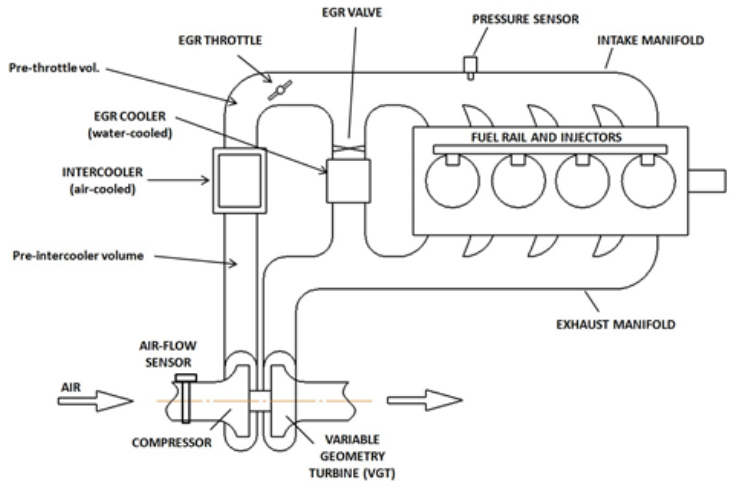

Figure 1.3: Diesel engine schematic.

1.3 Control Objective, History, and Challenges

The diesel air path (DAP) is pursued in this work as a representative system within automotive powertrains which has been traditionally challenging to control due to its highly coupled nonlinear behavior, the need for constraints to be considered for driveability and emissions, and the very limited computational resources in ECUs. The challenges that will be addressed are common in not only automotive engine control, but also control of gas turbine engines in aircraft, e.g., the challenges of improving transient response and reduce turbo-lag, coordinating multiple actuators to handle multiple objectives, handling constraints, and managing computational complexity.

The diesel engine schematic is shown in Figure 1.3. The engine consists of a cylinder block, intake and exhaust manifolds, exhaust gas recirculation (EGR) path, and a variable geometry turbocharger (VGT). The flows in the engine are controlled using a VGT, EGR valve, and EGR throttle. The control objective is to track set-points for intake manifold pressure, commonly referred to as manifold absolute pressure (MAP) or boost, and EGR rate (the ratio of EGR flow to cylinder flow). These set-points are provided by maps, i.e, look-up tables, as functions of the engine operating condition, i.e., the current engine speed and load (typically indicated by fuel rate). State and output constraints are to be imposed

on MAP, EGR rate, exhaust pressure, and turbine speed. Control constraints must also be enforced on the VGT, EGR valve, and EGR throttle. Furthermore, the computational complexity of the controller must meet stringent computation budget requirements.

Specifically, it is desired that the control can be computed within 200 μsec on a modern ECU, e.g., Freescale's MPC5644A microcontroller [26]. The 200 μsec execution time budget has been set by the sponsors of this research, Toyota Motor Corporation, and is based on a typical sampling period 32 msec for DAP control and the fact that the air path controller is only able to occupy a fraction of the ECU's computational resources which must be used for many other tasks. Throughout this dissertation, "low complexity MPC" refers to a MPC controller for the DAP that is able to track set-points and enforce input, state, and output constraints and whose control can be computed within 200 μsec on a modern ECU.

1.3.1 History of Linear MPC for the DAP

There has been growing interest in the use of MPC for the DAP. This has been driven by the need to be able to systematically handle MIMO control objectives and constraints derived from driveability and emissions requirements. Furthermore, the use of MPC for the DAP, and powertrain control in general, has been enabled by increases in the computational power of industry ECUs, albeit slower increases compared to CPUs in general. Use cases of LMPC for the DAP began to appear in 2007.

- Ortner and del Re in 2007, [84], first applied LMPC to the DAP. Their work utilized a two state linear model. The inputs were EGR position, VGT position, engine speed, and fuel rate. The tracked outputs were MAP and mass air flow (MAF, also referred to as compressor flow). Control was performed with EGR position and VGT position and the engine speed and fuel rate were treated as external measured disturbances. A two state linear model was generated using system identification [65]. To handle the DAP nonlinearities, the engine operating range, i.e, the engine speed and fuel rate range, was split into 12 zones, each with its own identified model and MPC controller. To achieve

zero-offset, steady-state tracking, an output disturbance estimator was used along with an input-velocity MPC formulation. Constraints were used to enforce control input limits and slew rate limits. Explicit MPC was used as the QP solution method. A control horizon of 1 was used with a 5 sec prediction horizon, i.e., only a single control move was allowed at time step $i = 0$ and held constant for the remainder of the 5 sec prediction horizon. The control horizon of 1 was chosen because $N_r = 25$ for the explicit MPC representation and became $N_r = 433$ with a control horizon of 2. The controller was implemented with dSPACE with a sampling rate of 50 msec. The controller was able to achieve 50% improvement in NOx and 10% improvement in PM emissions without a net increase in fuel consumption over the New European Drive Cycle (NEDC).

- Stewart and Borrelli in 2008, [101], applied MPC to the DAP. The system inputs and outputs are the same as in [84] with the exception that NOx was also considered as an output for the purpose of including a soft maximum NOx constraint. A PWA model approximation of a physics based nonlinear model of the DAP was obtained for prediction. With the PWA model, the linear model dynamics (A, B, C, D) were held constant at the PWA region at time step $i = 0$ for the remainder of the prediction horizon. An active set semi-explicit MPC[1] strategy was used to solve the associated QPs [10]. Handling the PWA model region switches during the prediction horizon would lead to a mixed integer quadratic program (MIQP) which is currently too computationally expensive to handle on an ECU. Offset-free, steady-state tracking is obtained through the combination of a disturbance estimator and input-velocity form MPC as was done in [84]. The resulting controller was able to demonstrate NOx constraint enforcement, albeit soft enforcement.

- Wang et al. in 2009, [115], also applied MPC to the DAP using a linear parameter

[1]The semi-explicit MPC solver precomputes and stores partial solutions to the associated KKT conditions rather than full solutions as in the explicit MPC case. This allows for a more efficient search through active set combinations compared to explicit MPC.

varying (LPV) model where the linear model dynamics (A, B, C, D) are scheduled continuously on engine speed and fuel rate rather than by zones as in [84]. Again, the controlled inputs are EGR position and VGT position and the tracked outputs are MAP and MAF. Offset-free, steady-state tracking is achieved through a disturbance estimator and input-velocity form MPC. An online QP solver rather than explicit MPC is used because (A, B, C, D) is continuously variable as a function of engine speed and fuel. A sampling time of 50 msec was used with a control horizon of 3 and prediction horizon of 120. Constraint enforcement was not demonstrated in the results of this work.

- Karlsson et al. in 2010, [56], applied MPC to the DAP utilizing more actuators and sensors relative to [84, 101, 115] and targets directly high-level objectives such as tracking engine torque and reducing NOx. The actuators that are used are crank angle degree of start of injection, fuel injection duration, EGR valve position, and VGT position. In-cylinder pressure measurements were used to compute indicated mean effective pressure (IMEP), combustion phasing, and maximum pressure derivative. NOx and soot sensors were also used in the control strategy. A sixth-order linear model was used for prediction around a single operating condition. Control moves were allowed in the prediction at steps $i \in \{0, 2, 4\}$ with a total prediction horizon of 100. High tracking weight was placed on IMEP to achieve fast engine torque response. A NOx set-point of 0 was used to reduce NOx as far as possible. A constant set point for combustion phasing was used. Soft constraints were placed on soot and maximum pressure derivative. The goal of this work was to illustrate the potential for MPC to meet high-level objectives when more degrees of freedom from the actuators are available and when more sensors are available. However, this proposed strategy relies on sensors that are not currently available in consumer engines.

- Recently, commercial software, Honeywell's OnRAMP Design Suite, [36], has become

available for systematic design of MPC controllers for powertrain applications. This software stems from [101].

Of the LMPC methods employed in the past, the strategy that is viable for implementation in current ECUs is characterized by, one, partitioning of the engine operating range into zones with a different LMPC controller per zone, and, two, utilizing explicit MPC as the solution method to the associated QPs. However there are still some drawbacks and challenges to using this strategy.

- This strategy leads to a larger than desired calibration effort since there is no well defined method for partitioning a plant operating range into zones and separate LMPC controller calibrations must be done in each zone. PWA models as in [101] could be used, however, this could lead to large memory usage in the ECU.
- The control can be discontinuous along the zone boundaries.
- Because explicit MPC is used, the partitioning strategy can lead to large memory usage, $o(N_z\mathcal{C}(Nn_h + n_\psi, Nn_u))$, where N_z refers to the number of zones or the number of regions in the PWA plant model, see Section 1.2.3.

1.3.2 History of Nonlinear Model Predictive Control for the Diesel Air Path

With the maturation of LMPC for the DAP, research focus has now shifted toward developing NMPC for the DAP. However, the feasibility of NMPC, in the sense of being able to achieve viable ECU computation times for the DAP, is questionable at best at the moment.

- Herceg et al. in 2006, [33], first demonstrated the use of NMPC to systematically achieve good transient performance while satisfying state and control constraints. The controlled inputs were EGR position and VGT position. The tracked outputs were

intake pressure, exhaust pressure, and compressor power. Minimum and maximum constraints were placed on intake pressure and exhaust pressure as well as on the controlled inputs. A three state nonlinear physics based model, [53], was used for prediction. Zero-offset, steady-state tracking in the presence of plant/model mismatch was not considered. While the tracking and constraint enforcement results were promising, "currently [in 2006] it is not possible to implement NMPC in real-time due to the limited computational power available on today's embedded control systems, which are not suitable for the required fast solution of the resulting finite-time optimal control problem."

- The opinion of [33] is bolstered in Wang et al. in 2009, [115], where the claim is made that using a nonlinear model "leads in general to a non convex problem which may be difficult to solve and usually needs a sequential approximation with quadratic programs at each time step, which is computationally expensive." Thus, LPV models were used in [115].

- More recently, Gaglieardi et al. in 2014, [27], demonstrated improved computation times (required around 50 msec for an update), contrary to [33], utilizing a discrete time polynomial model, [35], and the Continuation and Generalized Minimum Residual method (C/GMRES), [82]. The controlled inputs were EGR position and VGT position and the tracked outputs were MAP and MAF. Zero-offset, steady-state tracking was not considered.

- In 2014, Murilo et al., [78], demonstrated computation times of under 10 msec for the DAP application. Again, the controlled inputs were EGR position and VGT position and the tracked outputs were MAP and MAF. A 6 state nonlinear physics based model was used for prediction. A parametrization, utilizing exponential decay functions, of the discretized control sequence was used to reduce the number of optimization variables and a sequential quadratic program (SQP) was solved at each time step.

- Explicit NMPC has also been developed in the SI engine case, [31]. The control objective was to track MAP using a throttle and a turbocharger with a wastegate. A 10,000 times reduction in online computation time compared to online NMPC was achieved leading to sub-millisecond computation times on an i7 CPU. However, as the authors note, the complexity of explicit NMPC suffers from the "curse of dimensionality" with respect to the number of input parameters and does not scale well with even the addition of integral action. Furthermore, with explicit NMPC, controller reconfigurability to model changes and ability to incorporate adaptive models is largely lost.

While the 10 to 50 msec computation times demonstrated in [27, 78] for NMPC applied to the DAP may seem reasonable (the DAP control update is typically applied every 10 to 50 msec), in practice, the available ECU computation time allotment for DAP control is much smaller than the DAP control update rate because the ECU performs many other functions besides air path control and thus a sub-millisecond computation time for air path control is highly desirable. Furthermore it is not clear how zero-offset, steady-state tracking can be obtained with NMPC. Thus, there are still many challenges to overcome to apply NMPC to the DAP in practice.

1.4 Contributions

This dissertation focuses on overcoming the challenges of applying MPC to the DAP as outlined in Section 1.3. Namely, in the LMPC case, the need for partitioning the engine operating range is reduced. Furthermore, the number of constraints that are handled in this work are greater than in previously published cases [84, 101, 115] without a significant increase in the computational footprint of the associated explicit MPC controller. In the NMPC case, a number of strategies are developed that together achieve the desired sub-millisecond computation time required for current ECU implementation.

Most of the results outlined in the following have been published in or submitted to

journals, conferences proceedings, and patent applications, [38, 39, 41–45, 47–49].

1.4.1 Linear Model Predictive Control for the Diesel Air Path

The contributions made toward LMPC applied to the DAP are summarized as follows.

- A low complexity LMPC for the DAP, compared to [84, 101, 115], has been developed, [38, 39, 43, 49]. The controller is able to handle more constraints, a primary source of computational complexity, than in previous application of LMPC to the DAP. This is done with improved computation time and dramatically reduced memory usage. This complexity reduction is achieved through
  - partial nonlinear inversion through the choice of EGR flow as a control input,
  - rate-based MPC,
  - constraint remodeling,
  - and intermittent tightened constraint enforcement.
- Integral action is introduced to Contractive MPC (CMPC), [24, 32], a low complexity MPC formulation with stability guarantees, through the rate-based method [41, 44].
- A new, low complexity method for gain scheduling explicit MPC has been developed, [45, 48]. Traditionally, explicit MPC is thought of as non-reconfigurable, i.e., it is not able to accommodate changes in the model (A, B, C, D) matrices. The new gain scheduling method exploits an operating condition dependent gain on the controller output. The chosen gain improves the match between the nominal model and actual plant. In order to enforce control constraints, the method is implemented utilizing a switching structure similar to Pulse-Width-Modulation (PWM). Theoretical results show that as the switching frequency increases, the desired gain is recovered.
- Tube-MPC, a robust MPC method, is demonstrated in the DAP application [42]. It was found that a rate-based formulation reduces the conservativeness of tube-MPC.

The robust MPC controller was demonstrated in experiments using an ad hoc, approximate tube-MPC method to reduce computational complexity. Following the experiments, a new split tube-MPC method is introduced to formally reduce computational complexity.

Of these contributions, intermittent constraint enforcement, gain scheduling explicit MPC, and the split tube-MPC method are regarded as major innovations that also have broad applicability beyond the DAP example.

- The novel intermittent constraint enforcement strategy can be generally used to greatly reduce computational complexity of explicit MPC. The intermittent constraint enforcement strategy involves the application of constraints at only a subset of instances in the constraint horizon rather than, typically, over the full constraint horizon. This leads to an overall reduction in the number of constraints, a reduction in the number of regions of the associated explicit MPC controller, and a reduction in the worst case ECU computation time and memory usage. A procedure is also developed to choose the instances of constraint enforcement, i.e., the subset of the full constraint horizon.

- Gain scheduling, [28], can generally be used to improve the performance and reduce calibration effort of a linear based controller in loop with a nonlinear plant. This has previously been done for non-MPC DAP controllers, [108,109], and for turbofan engine controllers, [28]. However, this gain scheduling strategy has never been considered in coordination with explicit MPC with input constraints. To handle the input constraints within explicit MPC, a non-trivial, novel extension of the gain scheduling strategy is developed.

- A novel split tube-MPC strategy is developed that maintains the same constrained domain of attraction as standard tube-MPC with reduced computational complexity. Compared to non-tube-MPC, standard tube-MPC is much more computationally complex due to an increase in the number of optimization variables and constraints used

to enforce constraints robustly, i.e., under disturbances. The novel split tube-MPC strategy reduces the computational complexity of tube-MPC onto the same level as non-tube-MPC making the application of tube-MPC more computationally tractable.

1.4.2 Nonlinear Model Predictive Control for the Diesel Air Path

The contributions made toward NMPC applied to the DAP are summarized as follows.

- A NMPC design for the DAP with sub-millisecond update times is achieved, [47]. This is accomplished with a data driven modeling approach similar to [27, 35] to obtain a simple piecewise polynomial model that facilitates fast propagations of the state and co-state equations used in NMPC. C/GMRES is then applied with the polynomial model which represents the baseline design. With C/GMRES, it was previously unknown how inequality constraints, e.g., on states and controls, could be handled. Different inequality constraint handling techniques are explored and a solution that has considerable computational and performance advantages compared to C/GMRES is obtained. Furthermore, zero-offset set-point tracking is achieved with NMPC for the DAP. This is done through adaptation of the polynomial model.

- Kantorovich's method is used to freeze the Jacobian associated with the NMPC update, e.g., the Jacobians are precomputed and frozen for all time or computed only at the first iteration. By bypassing the Jacobian computation, the computational complexity is reduced. This is done in a way that can handle inequality constraints with little computational overhead.

- With linear MPC, it has been known that rate-based MPC achieves zero-offset, steady-state tracking while improving overall performance and mitigating the disadvantages of other strategies, see [18, 39, 85]. With NMPC, it is currently not well understood how to achieve zero-offset, steady-state tracking. Toward this end, the rate-based idea is used with NMPC. With rate-based NMPC, adaptation or disturbance estimation to

achieve zero-offset, steady-state tracking are no longer needed. This is advantageous for a number of reasons.

- The integral action does not need to be tuned separately from the controller, e.g., adaptation rates, and thus there are no concerns regarding the interactions between the control and estimation loops.
- Discontinuous control actions at zone boundaries are mitigated. This is because estimators, which would otherwise have to be reinitialized during zone switches, are not required in the proposed rate-based NMPC controller formulation for the DAP.

Of these contributions, Kantorovich's method applied to constrained NMPC and rate-based NMPC are regarded as major innovations that also have broad applicability beyond the DAP example.

1.5 Outline of Chapters

The remainder of this dissertation is organized as follows.

Chapter 2 describes the steps by which a low complexity LMPC for the DAP is achieved. A number of complexity reduction techniques are explained including partial nonlinear inversion, rate-based MPC, constraint remodeling, and intermittent tightened constraint enforcement. The computational complexity of the resulting LMPC controller is discussed and experimental results are shown. A tracking version of CMPC which guarantees local stability is also developed. An a posteriori stability analysis tool for MPC is also developed.

Chapter 3 describes the gain scheduling strategy for explicit MPC. This strategy achieves control constraint enforcement through a PWM-like strategy. Theoretical results prove that as the switching frequency increases, the originally desired behavior is recovered. A performance comparison to traditional gain scheduling strategies is given through simulations. The computational complexity of this gain scheduling strategy is also discussed.

Chapter 4 describes a robust LMPC, i.e., tube-MPC, design for the DAP. Furthermore, tube-MPC with zero-offset, steady-state tracking is achieved through a rate-based design which has not previously been done. An ad hoc approximation of the tube-MPC controller is then given to facilitate experimental implementation. After the experimental results, the chapter concludes with the introduction of a new reduced-complexity tube-MPC strategy.

Chapter 5 describes the steps by which a NMPC design for the DAP is achieved with sub-millisecond update times. This is achieved primarily through non-traditional handling of the inequality constraints. The performance and computational complexity for various inequality constraint handling techniques is compared. Kantorovich's method is also introduced to further reduce computational complexity. Finally, rate-based NMPC is introduced which achieves zero-offset, steady-state tracking.

Finally, Chapter 6 provides conclusions, acknowledges open questions, and discusses future research directions.

Chapter 2

Linear Model Predictive Control

This chapter presents the development and results of a LMPC controller for the diesel engine air path. The objective is to regulate the MAP measurement and EGR flow estimate to specified set-points by coordinated control of the VGT, EGR valve, and EGR throttle actuators. See Figure 1.3 for a schematic of the diesel engine. The steps by which a controller with low computational complexity, good tracking performance, and capability to enforce multiple constraints can be obtained is presented. Through the employed strategies, the need to cover the operating range with multiple linear models and to use a complicated switching controller structure is avoided. Experimental results are presented that demonstrate the ability of the LMPC controller to follow specified set-points while satisfying state and control constraints throughout the engine operating range.

2.1 Introduction

Prior research on the application of LMPC to the diesel engine air path (DAP) has been reported in [56,84,101], where the control strategy involves switching between multiple LMPC controller designs that cover the engine operating range. This work also focuses on applying LMPC to the DAP, however, the approach described in this chapter has several differences. Specifically, the strategies of partial inversion, [38, 113], and rate-based MPC (also refereed to as velocity form MPC), [18, 39, 43, 49, 85, 114], are combined. With this

approach, it is found that a single LMPC controller is sufficient to cover the engine operating range.

For the control implementation, explicit MPC, [5], is used to represent the solution to the underlying QP problem which is suitable given the small QP problem sizes that the strategy employs, i.e., a small number of constraints and optimization variables, [1, 7].

One possible choice of control inputs is to use VGT position (percent closed) and EGR valve position (percent open). In this work, however, the use of EGR valve flow as a control input rather than EGR valve position is considered. Partial nonlinear inversion is then used to backtrack both EGR valve position and EGR throttle position. Treating EGR flow as a control input has previously been found to alleviate the effect of DC gain reversal, [38, 113]. The approach of treating EGR flow as a control input has also been previously employed in [53] for diesel engines and in [54] for gasoline engines.

A rate-based framework will be used for the MPC design to achieve zero-offset, steady-state tracking. The rate-based strategy avoids windup issues evident in methods that rely on augmenting an integrator [41, 49]. Disturbance estimators are also commonly used in air path control [84, 95], however, require a disturbance model. Rate-based MPC avoids these issues which leads to overall better tracking performance and further extension of a single controller's operating range.

The contributions of the work to LMPC applied to the DAP discussed in this chapter are summarized as follows.

- A single zone LMPC strategy for the DAP is developed. This is achieved through
  - partial nonlinear inversion of the EGR valve orifice flow equation which makes the DAP plant appear more linear to the linear MPC controller,
  - and rate-based MPC.
- Further computational complexity reduction is achieved through constraint remapping and intermittent constraint enforcement.

- An offset-free version of MPC with stability guarantees, i.e., an offset-free version of CMPC, is developed.

The remainder of this chapter is organized as follows. Section 2.2 describes the partial inversion strategy that renders the plant more linear and the linear DAP model obtained through system identification. Section 2.3 describes the LMPC design for the DAP and complexity reduction techniques. Section 2.4 discusses stability of rate-based LMPC. Section 2.5 contains concluding remarks on LMPC.

Extensions of LMPC that target improved tracking performance through gain scheduling and robustness with respect to constraints through tube-MPC are described in Chapters 3 and 4, respectively.

2.2 Linear DAP Prediction Model

A linear DAP model is required for the subsequent MPC design. Similar to [36, 84], the engine operating range is split into zones centered at selected operating points and a linear model is identified to represent engine response in each zone. For example, a control designer may want to split the operating range into a low engine speed zone where dynamics are slower and a high engine speed zone where dynamics are faster. Based on initial closed loop nonlinear model simulations, [38, 39], it was found that only a single zone and a single MPC controller are sufficient. This has important advantages: computational overhead is reduced and the need to manage the controller switching is eliminated. One of the main strategies used to achieve a single zone design is the choice of EGR flow as a control input and the use of partial nonlinear inversion to obtain the corresponding EGR valve position which renders the plant more linear compared to choosing EGR valve position as a control input. Particularly, this choice has previously been found to alleviate the effect of DC gain reversal, [38, 113].

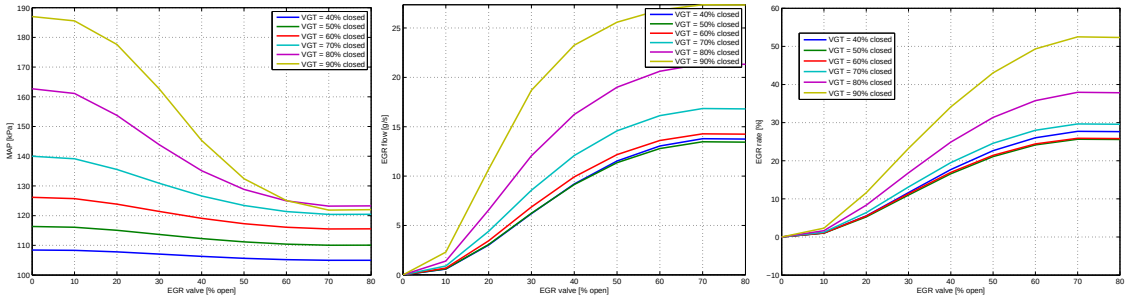

Figure 2.1: Gain reversal in MAP, EGR flow, and EGR rate when VGT and EGR valve position are used as control inputs. The data is obtained from a high fidelity physics based model.

2.2.1 Partial Nonlinear Inversion

The combination of feedback linearization and MPC has been explored in other applications, [19, 32, 80, 98]. To apply feedback linearization, a simple DAP model is required. A simple three state nonlinear model of the DAP, [53], has been considered, however, it was found that this model does not fit to experimental data adequately. Thus only a partial nonlinear inversion strategy will be used.

A steady-state map of the inputs to outputs of the diesel air path reveals a DC gain reversal. The DC gain reversal presents a serious impediment in multivariable diesel engine control design. This can be observed in Figure 2.1. When the EGR valve is past 60% open, MAP increases as the VGT closes toward 80% closed. However, further closing the VGT reduces MAP. Similarly, the EGR flow output also exhibits DC gain reversal. As the VGT closes from 40% to 50%, EGR flow first decreases. Continued closure of the VGT increases the EGR flow. Similar DC gain reversal behavior can also be seen on the EGR rate output. Since the region in which this gain reversal occurs is uncertain, controllers that incorporate integral control have to be carefully designed. By redefining the input as EGR valve flow rather than EGR position, i.e., by commanding EGR valve flow and determining the EGR valve position to replicate this commanded flow, the DC gain reversal disappears, see Figure 2.2.

Taking advantage of the reduced degree of nonlinearity, EGR flow is chosen as the control

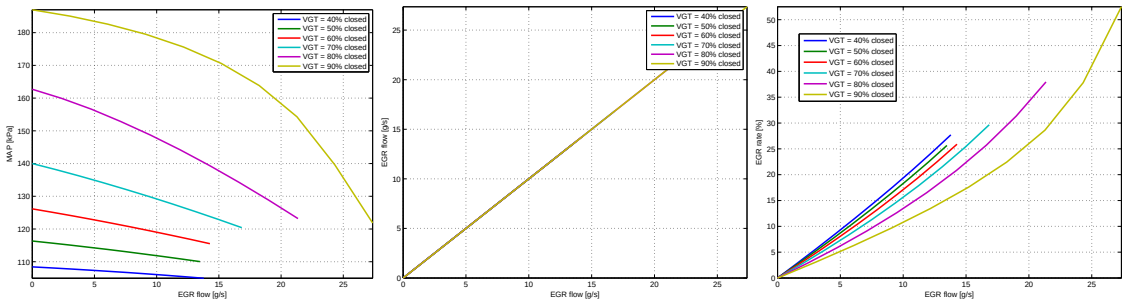

Figure 2.2: Monotonic behavior in MAP, EGR flow, and EGR rate when VGT position and EGR flow are used as control inputs. The data is obtained from a high fidelity physics based model.

input instead of EGR valve position. To actually achieve control using EGR flow, however, the EGR flow command, denoted as v_{EGR}, must be mapped to an EGR valve position command, u_{EGR}. This can be done by inversion of the EGR valve orifice flow equation. The EGR valve orifice flow model follows,

$$W_{EGR} = \frac{p_{ex}}{\sqrt{R_{air}T_{eco}}} C_d(u_{EGR}, N_e)\phi\left(\frac{p_{in}}{p_{ex}}\right), \tag{2.1}$$

$$\phi(\tau) = \begin{cases} \sqrt{\left(\frac{(\gamma-1)(1-\tau)}{2\gamma} + \tau\right)(1-\tau)} & \tau > \frac{1}{\gamma+1} \\ \sqrt{\frac{\gamma}{2(\gamma+1)}} & \tau \leq \frac{1}{\gamma+1} \end{cases}, \tag{2.2}$$

where C_d is a nonlinear function that encompasses the discharge coefficient and orifice area, see [22], and the nonlinear function ϕ handles choked flow. Then, based on current measurements or estimates of exhaust pressure (p_{ex}), intake pressure (p_{in}), EGR cooler out temperature (T_{eco}), engine speed (N_e), and commanded EGR flow (v_{EGR}), the EGR valve position (u_{EGR}) command can be obtained,

$$u_{EGR} = C_d^{-1}\left(N_e, \frac{v_{EGR}\sqrt{R_{air}T_{eco}}}{p_{ex}\phi\left(\frac{p_{in}}{p_{ex}}\right)}\right), \tag{2.3}$$

assuming that C_d is invertible.

The strategy of obtaining u_{EGR} from v_{EGR} and the inversion of the orifice flow equation (2.3) had previously been employed in [38, 39] in simulation assuming that the signals p_{ex} and T_{eco} were available for direct use in the strategy. However, in practice, these values are not measured, thus estimators are needed. A reliable measurement and estimate for W_{EGR} is also unavailable which eliminates the possibility of using an inner-loop controller, e.g., PID. While estimators for these values can be developed, see Section 2.3.4, it was found that using an estimator based strategy with (2.3) is unreliable, i.e., it is difficult to tune the complex interactions between the estimators and controller.

Thus, for implementation in practice, a strategy similar to [110] is used. The strategy combines the EGR valve and EGR throttle into a single actuator. This is done in a manner such that the EGR throttle only closes after the EGR valve is is fully open. A combined EGR valve and EGR throttle strategy for the DAP has a number of advantages: a square system is obtained from inputs to tracked outputs without having to define a separate strategy for the EGR throttle controller, and cylinder backpressure is reduced because the throttle is only closed when more EGR is needed (to meet set-points stemming from emissions regulations) than the EGR valve can provide.

The combined EGR valve command, u_{EGR}, and EGR throttle, u_{th}, command is denoted as θ_{EGR}. From θ_{EGR}, the EGR valve and EGR throttle position command can be obtained as

$$u_{EGR} = \begin{cases} 0, & \text{if } \theta_{EGR} \leq 0, \\ \theta_{EGR,} & \text{if } 0 < \theta_{EGR} < \bar{u}_{EGR}, \\ \bar{u}_{EGR}, & \text{if } \bar{u}_{EGR} \leq \theta_{EGR}, \end{cases} \tag{2.4}$$

$$u_{th} = \begin{cases} 0, & \text{if } \theta_{EGR} \leq \bar{u}_{EGR}, \\ \theta_{EGR} - \bar{u}_{EGR}, & \text{if } \bar{u}_{EGR} < \theta_{EGR} < \bar{u}_{EGR} + \bar{u}_{th}, \\ \bar{u}_{th}, & \text{if } \bar{u}_{EGR} + \bar{u}_{th} < \theta_{EGR}, \end{cases} \tag{2.5}$$

where $\bar{u}_{EGR}$ and $\bar{u}_{th}$ are the maximum EGR valve and EGR throttle positions, respectively.

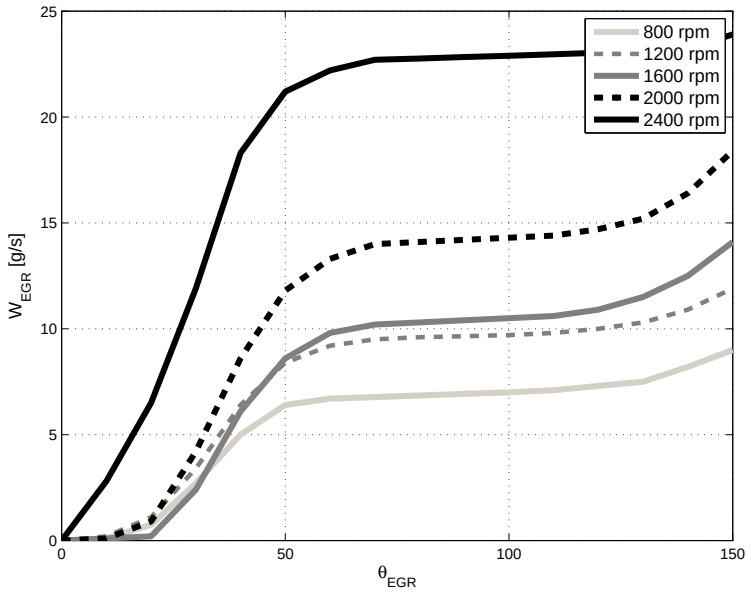

Figure 2.3: Steady-state EGR flow vs. combined EGR valve and EGR throttle position command, θ_{EGR}.

Figure 2.3 shows θ_{EGR} versus experimentally estimated W_{EGR} at different engine speeds in steady-state. The EGR valve first opens as θ_{EGR} changes from 0 to $\bar{u}_{EGR} = 70\%$. Then the EGR throttle closes from 0 to $\bar{u}_{th} = 80\%$. The maximum limits, $\bar{u}_{EGR}$ and $\bar{u}_{th}$, are chosen as 70% and 80%, respectively, since, for the experimental engine, the output responses are not sensitive to any EGR valve and EGR throttle variations past these values. Based on steady-state experimental data, a table-lookup scheduled on engine speed and fueling rate is constructed to invert the EGR flow command, v_{EGR}, to θ_{EGR}, i.e., the nonlinear map shown in Figure 2.3 is inverted. The maximum EGR flow that can be commanded, $\bar{v}_{EGR}$, which will be used by the MPC controller, is evaluated by looking up the maximum flow at a given engine speed and fueling rate.

2.2.2 Linear DAP Model Identification

For the controller presented in this chapter, system identification, [71], of the engine experimental hardware (plant) is performed to obtain a local seventh order linear model. The nominal operating point is chosen as 1600 rpm engine speed and 30 mm^3/st. fueling

rate. This point is at the center of the New European Drive Cycle (NEDC) range (from 800 rpm to 2400 rpm engine speed, and 0 mm^3/st. to 60 mm^3/st. fuel rate). Balanced truncation, [14, 67], is then applied to reduce the model order while preserving the system's DC gain.[1] Lowering the order of the linear prediction model is advantageous: this reduces the input parameter size of explicit MPC and its ROM and chronometric footprint. In fact, the computational cost of evaluating the explicit MPC controller in each region (1.13) is $o(n_x(Nn_h + n_\psi))$, where n_x is the number of states and $Nn_h + n_\psi$ is the total number of inequality constraints. This complexity arises from the size of the matrix K_j in (1.13) which is determined by the number of states and the number of inequality constraints, [5, 23].

The input sequence for the system identification process is composed of steps in commanded VGT position (u_{vgt}) and commanded EGR flow (v_{EGR}). The considered outputs are the intake manifold pressure (p_{in}), estimated EGR flow ($\hat{W}_{EGR}$), and estimated exhaust manifold pressure ($\hat{p}_{ex}$). Note that the input, v_{EGR}, is not the same as the output, $\hat{W}_{EGR}$. There are dynamics introduced between the two through the inversion of v_{EGR} to u_{EGR} and u_{th}, which is inexact and is based on steady-state maps only. Note that the EGR flow is chosen as an output rather than EGR rate. This is done to reduce the degree of nonlinearlity of the plant through bypassing the nonlinear EGR rate relationship, $\chi_{EGR} = (W_{cyl} - W_c)/W_{cyl}$. Set-points for $\hat{W}_{EGR}$ can easily be obtained through steady-state maps of χ_{EGR}, W_{cyl}, and W_c.

The estimators for EGR flow and exhaust pressure are described in Section 2.3.4. A constraint on turbocharger speed will also be imposed, however, will be remapped to an intake manifold pressure constraint. Thus an identified dynamic model for turbocharger speed is not required. Figure 2.4 shows a comparison between the system identification data, 7 state model, and 4 state reduced model. These responses show that a 4 state model is sufficient to match the dynamics of the 7 state model and experimental data.

[1] It has been observed that the modeling error is smaller when a higher order linear model is first identified and is then reduced compared to directly identifying a reduced order model. The analysis of this empirically observed property is left to future research.

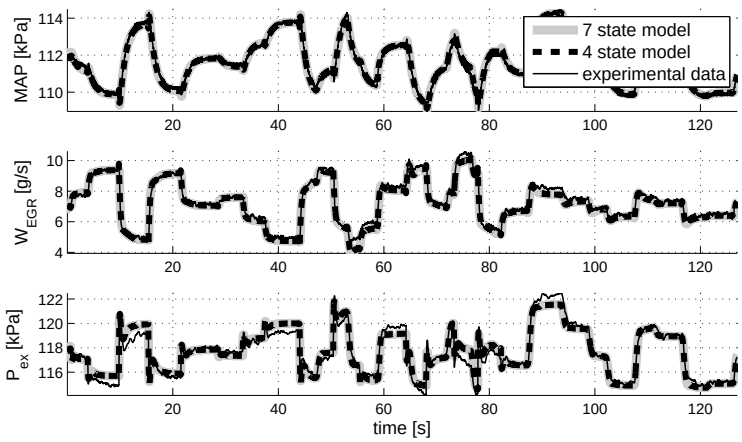

Figure 2.4: Comparison of MAP, EGR flow, and exhaust pressure response to changes in VGT position command and EGR flow command between a 7 state model, 4 state model, and experimental data.

The resulting linear diesel air path model has the following form,

$$x_{k+1} = Ax_k + Bu_k, \tag{2.6}$$

$$y_k = Cx_k + Du_k, \tag{2.7}$$

where $u_k = [\ u_{VGT,k} \quad v_{EGR,k}\ ]^T$ and $y_k = [\ p_{in,k} \quad \hat{W}_{EGR,k} \quad \hat{p}_{ex,k}\ ]^T$.

2.3 Linear MPC Design

The diesel air path control objective is to regulate the intake pressure and EGR flow to specified set-points through coordinated control of the VGT, EGR valve, and EGR throttle subject to actuator constraints on the VGT, EGR valve, and EGR throttle ranges. Output constraints on maximum intake pressure, EGR flow, exhaust pressure, and turbocharger speed must also be enforced. These output constraints are dictated by driveability, safety, reliability, noise, vibration, and harshness requirements.

Figure 2.5 shows the overall closed-loop diesel air path plant and controller structure. The engine speed (N_e) and fueling rate (W_f) are inputs to the diesel air path plant. The

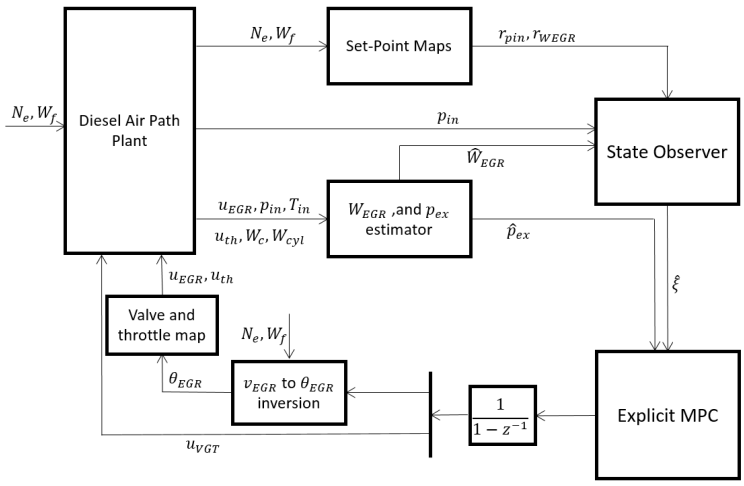

Figure 2.5: Controller structure.

set-points for MAP (r_{pin}) and EGR flow (r_{WEGR}) are functions of engine speed and fueling rate. Because sensors are not available for EGR flow (W_{EGR}) and exhaust pressure (p_{ex}), estimators for these signals are used. The measured intake pressure and estimated EGR flow are used by a state observer to estimate the states for MPC feedback. This is required because the process of system identification and model order reduction renders the states non-physical. The state estimate ($\hat{\xi}$) is used by the rate-based explicit MPC controller to determine incremental changes in the EGR flow command (v_{EGR}) and VGT position command (u_{VGT}). The incremental changes in the control signals are integrated to generate values for the EGR flow command and VGT position command. The partial nonlinear inversion block then inverts the virtual EGR flow command to a corresponding combined EGR valve and EGR throttle command (θ_{EGR}). The combined EGR valve and EGR throttle command is then mapped to individual commands for the EGR valve position and EGR throttle position.

In the remainder of this section, the controller design details are explained. First, rate-based MPC is described which will be used to obtain zero-offset, steady-state tracking. Two main pathways for computational complexity reduction are then discussed: intermittent constraint enforcement and constraint remapping. Estimators for various signals are described. The experimental diesel air path MPC formulation is then given and experimental results

are presented. This section closes with a discussion of the computational complexity.

2.3.1 Rate-based MPC

It is desirable for MPC applied to the diesel air path to incorporate integral type action to compensate for the mismatch between the model and true plant. In conventional MPC, the direct addition of an integrator may expose the controller to issues of integral windup. Note that constraints on integrator states could be added, however, it is unclear how to set the integrator limits for multiple-input multiple-output (MIMO) systems. This implies that an anti-windup scheme is required.

Another common strategy to achieve offset-free tracking with MPC is to use a disturbance estimator [84,95]. The disturbance estimator is typically coupled with an input-velocity form of the MPC controller so that the steady-state control values corresponding to the reference signal do not need to be known a priori. The disadvantage of using a disturbance estimator is that it requires a disturbance model, [85], and requires augmentation of both disturbance states and control states (from using input-velocity form MPC).

Alternate to integrator augmentation or disturbance estimators, a rate-based strategy can be used (also called velocity-form MPC, [85, 114]). A comparison between the augmented integrator and disturbance estimator strategies is presented in [95]. A comparison between augmented integrator and rate-based strategies is presented in [39]. The advantages and disadvantages of a rate-based strategy versus disturbance estimators are described in [85].

Of most importance to the DAP application, is that rate-based MPC achieves offset-free steady-state tracking without the need to deal with integrator windup. If instead, integrator augmentation were utilized in the presence of constraints and significant nonlinearities, the integrators would be consistently wound up, as was observed in [39], particularly when the engine is in transient operation. The windup can be mitigated by utilizing more local models and MPC controllers which leads to better plant/model matching and less "work" for the integrators. Furthermore, rate-based MPC achieves offset-free steady-state tracking without

disturbance models. The choice of disturbance models becomes more difficult in the presence of nonlinearities and multiple local linear disturbance models may be required. Thus, rate-based MPC can be viewed as not only a strategy to achieve offset-free steady-state tracking but also as a strategy to reduce computational complexity and calibration effort because fewer local models and controllers may be needed.

In rate-based MPC, the state and control increments and the augmented state vector are first defined,

$$\Delta u_k = u_k - u_{k-1}, \tag{2.8}$$

$$\Delta x_k = x_k - x_{k-1}, \tag{2.9}$$

$$e_k = y_{k-1} - r_{k-1}, \tag{2.10}$$

$$\xi_k = \begin{bmatrix} x_k \\ e_k \end{bmatrix}, \tag{2.11}$$

where e_k is the output tracking error. The rate-based model then has the form,

$$\xi_{k+1} = \bar{A}\xi_k + \bar{B}\Delta u_k, \tag{2.12}$$

$$e_k = \bar{C}\xi_k, \tag{2.13}$$

$$\bar{A} = \begin{bmatrix} A & 0 \\ C & I \end{bmatrix}, \bar{B} = \begin{bmatrix} B \\ D \end{bmatrix}, \bar{C} = \begin{bmatrix} 0 & I \end{bmatrix}. \tag{2.14}$$

A rate-based MPC quadratic programming problem, assuming the prediction, control, and

constraint horizons are equal to N, can be expressed as follows,

$$\begin{aligned}
\min_{\Delta u_{i|k},\, i\in\{0,\dots N-1\}} \quad & \xi_{N|k}^T P \xi_{N|k} + \sum_{i=0}^{N-1} e_{i|k}^T Q e_{i|k} + \Delta u_{i|k}^T R \Delta u_{i|k}, \\
\text{subject to}: \quad & \xi_{i+1|k} = \bar{A}\xi_{i|k} + \bar{B}\Delta u_{i|k}, \\
& e_{i|k} = \bar{C}\xi_{i|k}, \\
& e_{i|k} + r_{k-1} \in \mathbb{Y}, u_{k-1} + \sum_{j=0}^{i} \Delta u_{j|k} \in \mathbb{U}, \\
& \xi_{0|k} = \xi_k,
\end{aligned} \tag{2.15}$$

where k is the current time step and $i \in \{0, ..., N-1\}$. In the rate-based MPC problem, (2.15), the objective function penalizes the tracking error through the term $e_{i|k}^T Q e_{i|k}$ with $Q = Q^T \geq 0$, and control increments through the term $\Delta u_{i|k}^T R \Delta u_{i|k}$ with $R = R^T > 0$. A terminal state cost can be incorporated through the term $\xi_{N|k}^T P \xi_{N|k}$ with $P = P^T \geq 0$. Output and control constraints are enforced through $e_{i|k} + r_{k-1} \in \mathbb{Y}$ and $u_{k-1} + \sum_{j=0}^{i} \Delta u_{j|k} \in \mathbb{U}$, where u_{k-1} is the control applied at the previous time step, $k-1$. State constraints, $x_{i|k} \in \mathbb{X}$, though not explicitly shown in (2.15), can be added through additional outputs which would be integrators of the corresponding predicted state increments $\Delta x_{i|k}$. In rate-based MPC, the tracking objective naturally becomes a regulation objective. Once the optimization problem (2.15) is solved, the optimized first control increment, $\Delta u_{0|k}^\star$ is applied to the previously applied control, u_{k-1}, to obtain the control to be applied at the current time instant, $u_k = u_{k-1} + \Delta u_{0|k}^\star$. If zone switching is required with rate-based MPC, discontinuous control changes will be less apparent because the strategy determines the control value by integrating control increments which smooths discontinuous changes.

2.3.2 Intermittent Constraint Enforcement

The number of regions of explicit MPC is directly related to the number of constraints and optimization variables [23]. With intermittent constraint enforcement (ICE), the number of constraints is reduced by choosing to only enforce the output constraints intermittently,

i.e., at certain, but not all, time instants over the prediction horizon. In the ICE strategy, the intermittent constraints are tightened. Then during the intermediate steps over the prediction horizon, the originally imposed constraints are approximately satisfied. Note that the number of optimization variables can also be reduced by analyzing the impact of each optimization variable on the predicted state trajectories [83,94]. However, for the DAP MPC strategy, the number of optimization variables will be managed through using a short control horizon.

To gain insight into ICE, a simple second order system example illustrates the effect on the maximal output admissible set (MOAS), [60], as the intermittent output constraint is tightened. Consider the case where soft output constraints are used and observe the effect on the MOAS as the intermittent output constraints are tightened or relaxed. The MOAS, O_∞ , for the closed loop system with a MPC designed with intermittent constraints, $x_{k+1} = f_{cl}(x_k)$ and $y_k = g_{cl}(x_k)$, and output constraint $y \in \mathbb{Y}$ is defined by,

$$O_\infty = \left\{ x_0 \in \mathbb{R}^{n_x} : x_{k+1} = f_{cl}(x_k),\, y_k = g_{cl}(x_k) \in \mathbb{Y},\, \forall k \in \mathbb{Z}^+ \right\}. \tag{2.16}$$

The set O_∞ corresponds to the constrained domain of attraction of the closed loop system. Thus, the goal is to grow O_∞ by tightening the soft output constraints to achieve approximately the same constrained domain of attraction as when a full output constraint horizon is used. In the second order system setup, $A = \begin{bmatrix} 1 & 1 \\ & \\ 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 1 \\ \\ 1 \end{bmatrix}, C = \begin{bmatrix} 1 & 1 \end{bmatrix}$ and $D = 0$. The control constraint is $-1 \leq u_k \leq 1$ and the output constraint is $-1 \leq y_k \leq 1$. A single step prediction is used with the terminal state penalty P corresponding to state weight $Q = I_{2\times 2}$ and control weight $R = 100$. The output constraint is enforced only during the time steps in the ICE constraint set, $N_{ICE} \subset \mathbb{Z}^+$, and treated as soft with a slack variable, ϵ, and weight M. The amount of output constraint tightening is set through the term

α. The example MPC problem has the following form,

$$\begin{aligned} \min_{u_{0|k},\epsilon} \quad & \left(Ax_0 + Bu_{0|k}\right)^T P \left(Ax_0 + Bu_{0|k}\right) + u_{0|k}^T R u_{0|k} + M\epsilon^2 \\ \text{subject to :} \quad & -1 \leq u_{0|k} \leq 1,\ u_{i|k} = u_{0|k}\, \forall i \geq 1, \\ & x_{i+1|k} = Ax_{i|k} + Bu_{i|k}, y_{i|k} = Cx_{i|k}, \\ & -\alpha - \epsilon \leq y_{i|k} \leq \alpha + \epsilon,\ \forall i \in N_{ICE}. \end{aligned} \tag{2.17}$$

Once the explicit solution to (2.17) is computed, a closed loop PWA system can be formed,

$$x_{k+1} = (A + BG_j)\, x_k + BF_j \text{ if } H_j x_k \leq K_j, \text{ for } j \in \{1, ..., N_r\}\,. \tag{2.18}$$

For different values of the constraint set, N_{ICE}, and slack weight, M, the MOAS, O_∞, is computed for the PWA system (2.18) using Multi Parametric Toolbox (MPT) [63, 92]. The maximal control invariant set, C_∞, for the original (open loop) second order system with $x_{k+1} = f_{ol}(x_k)$ and $y_k = g_{ol}(x_k)$ can also be computed,

$$C_\infty = \left\{x_0 \in \mathbb{R}^{n_x} : \exists u_k \in \mathbb{U},\ x_{k+1} = f_{ol}(x_k, u_k),\ y_k = g_{ol}(x_k, u_k),\ y_k \in \mathbb{Y}\, \forall k \in \mathbb{Z}^+\right\}. \tag{2.19}$$

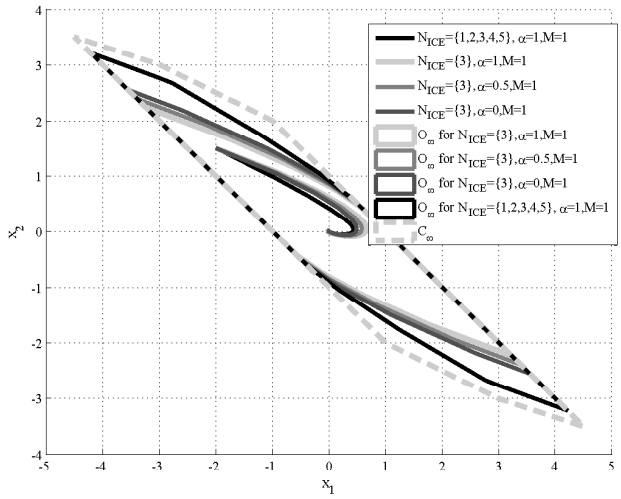

Figure 2.6: Maximal output admissible sets for $N_{ICE} = \{1, 2, 3, 4, 5\}$, $N_{ICE} = \{3\}$ and $M = 1$.

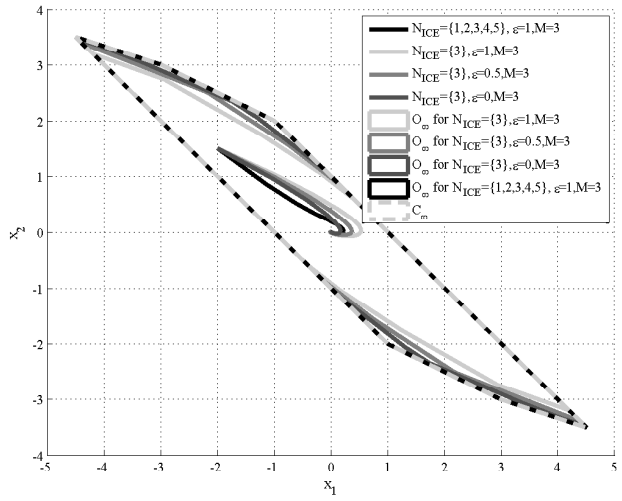

Figure 2.7: Maximal output admissible sets for $N_{ICE} = \{1, 2, 3, 4, 5\}$, $N_{ICE} = \{3\}$ and $M = 3$.

Figure 2.6 shows O_∞ computed for a full horizon of output constraints, $N_{ICE} = \{1, 2, 3, 4, 5\}$, and for various levels of constraint tightening with intermittent constraint enforcement, $N_{ICE} = \{3\}$. Observe that as the intermittent constraints are tightened, O_∞ grows. While this is somewhat counter-intuitive, it is a result of treating constraints as soft and more aggressive control action taken to drive the state into the tightened constraint set. Furthermore, as the slack weight is increased in Figure 2.7, observe that O_∞ actually approaches C_∞ for the second order system example. Note that in this example, the explicit controller with $N_{ICE} = \{1, 2, 3, 4, 5\}$ has 13 regions and 5 regions with $N_{ICE} = \{3\}$.

For simple systems, e.g., a second order system, the following procedure can be used to choose the set N_{ICE}.

1. Determine the maximum number of instances in the constraint horizon where the constraints will be enforced given the computational budget by looking at the number of regions of the explicit MPC controller and the corresponding worst case execution time and memory usage.

2. Given the maximum number of constraint enforcement instances in the constraint horizon from Step 1, compute the MOAS for each possible combination of constraint

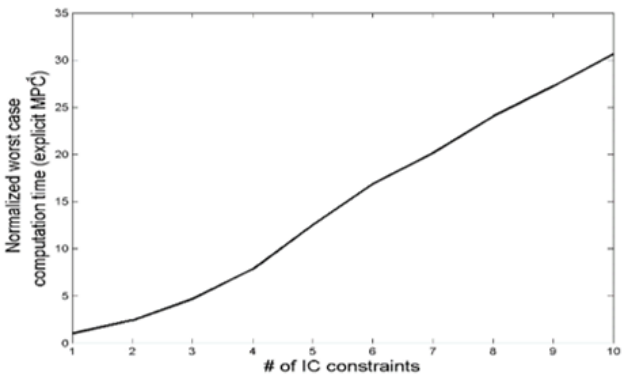

Figure 2.8: Normalized worst case execution time of the DAP explicit MPC controller as a function of the number of elements in N_{ICE}.

enforcement instances.

3. Choose N_{ICE} corresponding to the largest MOAS.

For systems where the MOAS cannot be explicitly computed and/or where there exists plant/model missmatch, as in the DAP application, N_{ICE} should be chosen through extensive simulations. The following procedure can be used to choose the set N_{ICE}.

1. Determine the maximum number of instances in the constraint horizon where the constraints will be enforced given the computational budget by looking at the number of regions of the explicit MPC controller and the corresponding worst case execution time and memory usage. For example, Figure 2.8 illustrates the growth in worst case execution time as a function of the number of elements in N_{ICE} for the DAP application. In the DAP application, a single instance of intermittent constraint enforcement was chosen to reduce the worst case execution time as much as possible.

2. Through simulations, analyze the constraint enforcement behavior for each possible combination of constraint enforcement instances and choose the "best" N_{ICE}. Figures 2.9 and 2.10 show the MAP overshoot constraint enforcement behavior for different cases of N_{ICE} for the DAP application where simulations were performed on a large fuel step from 10 mm^3/st. to 50 mm^3/st. at 2400 rpm. For the DAP application,

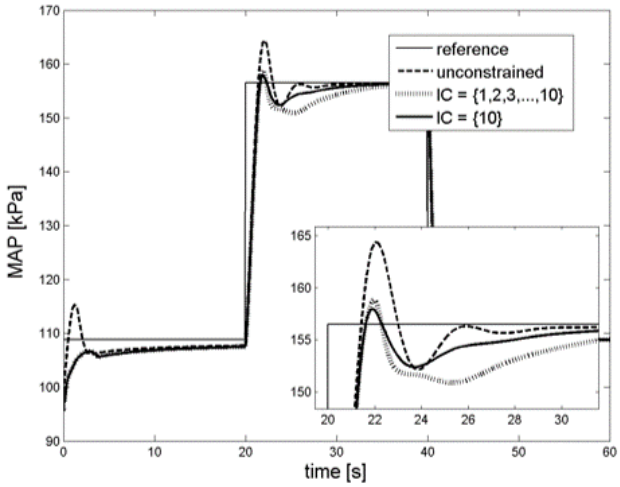

Figure 2.9: MAP overshoot comparison of difference choices of N_{ICE} during a fuel step from 10 mm^3/st. to 50 mm^3/st. at 2400 rpm.

where there is plant/model missmatch and a significant disturbance in fuel, the constraint enforcement behavior is counter-intuitive. For example, Figure 2.9 shows that using $N_{ICE} = \{10\}$ reduces the MAP overshoot more than using $N_{ICE} = \{1, ..., 10\}$. Based on a number of simulations using a single instance of intermittent constraint enforcement, it was observed that enforcing the single constraint near steady state, i.e., far out in the horizon, reduces the MAP overshoot the most.

2.3.3 Constraint Remapping

In addition to ICE, constraint remapping can be used to reduce the number of constraints and thus computational complexity. Consider the following, possibly nonlinear constraints,

$$\begin{aligned} h_1(x_k, u_k) &\leq \bar{h}_1, \\ h_2(x_k, u_k) &\leq \bar{h}_2. \end{aligned} \tag{2.20}$$

The goal of constraint remapping is to change the constraint h_2 into a constraint for h_1, thus reducing the total number of constraints. That is, the strategy of constraint remapping looks for a function $f_r(\bar{h}_2, x_k, u_k)$ such that

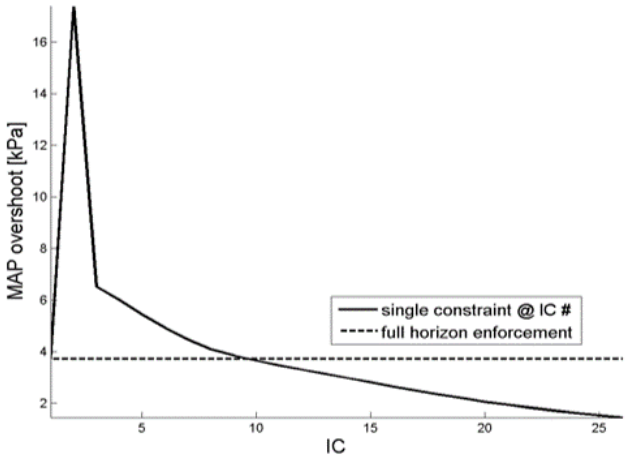

Figure 2.10: MAP overshoot comparison of difference choices of N_{ICE} with the maximum number of intermittent constraint enforcement equal to one.

$$h_1(x_k, u_k) \leq f_r(\bar{h}_2, x_k, u_k) \Rightarrow h_2(x_k, u_k) \leq \bar{h}_2. \tag{2.21}$$

Then (2.20) is satisfied through a reduced constraint of the form

$$h_1(x_k, u_k) \leq \min\{f_r(\bar{h}_2, x_k, u_k), \bar{h}_1\}. \tag{2.22}$$

For the DAP, constraint remapping can be accomplished a number of ways. Consider first a constraint on the maximum EGR flow output,

$$W_{EGR,k} \leq \bar{W}_{EGR}. \tag{2.23}$$

If the inversion from $v_{EGR,k}$ to $\theta_{EGR,k}$ is exact, i.e., when $\theta_{EGR,k}$ is applied, $W_{EGR,k} = v_{EGR,k}$, then (2.23) is equivalent to a constraint on the EGR flow command,

$$v_{EGR,k} \leq \bar{W}_{EGR}. \tag{2.24}$$

If the inversion is inexact, $\bar{W}_{EGR}$ can be calibrated such that the EGR flow near steady-state remains below the originally desired constraint, e.g., a look-up table can be made

for $\bar{W}_{EGR}$ as a function of the operating condition, $\bar{W}_{EGR}(N_{e,k}, W_{f,k})$. Thus the EGR flow output constraint and virtual EGR flow command constraint can be combined into a single constraint,

$$0 \leq v_{EGR,k} \leq \min\left\{\bar{W}_{EGR}(N_{e,k}, W_{f,k}), \bar{v}_{EGR,k}\right\}, \tag{2.25}$$

where the maximum EGR flow command is taken as the minimum between the EGR flow output constraint, $\bar{W}_{EGR}$, and the maximum EGR flow command allowed at the current operating condition, $\bar{v}_{EGR,k}$ (see Section 2.2.1).

Secondly, a constraint on the maximum turbocharger speed ($\omega_{t,k}$),

$$\omega_{t,k} \leq \bar{\omega}_t, \tag{2.26}$$

can be remapped to a constraint on the maximum intake pressure, ($\bar{p}_{in,k}^{remap}$). Since a constraint on the intake pressure, $p_{in,k} \leq \bar{p}_{in}$, will also be enforced, it can serve the dual purpose of also enforcing the turbocharger speed constraint,

$$p_{in,k} \leq \min\{\bar{p}, \bar{p}_{in,k}^{remap}\}. \tag{2.27}$$

The turbocharger constraint remapping is performed through the inversion of the compressor flow [77] and throttle flow models [22]. Consider first the compressor flow model,

$$W_{c,k} = \frac{\pi}{4}\frac{k_3\Gamma_k - k_1}{k_2 + \Gamma_k}\rho_a d_c^2 U_{c,k}, \tag{2.28}$$

$$\Gamma_k = \frac{2c_p T_a\left(\left(\frac{p_{pre,k}}{p_a}\right)^{\frac{\gamma-1}{\gamma}} - 1\right)}{U_{c,k}}, \tag{2.29}$$

$$U_{c,k} = \frac{\pi}{60} d_c \omega_{t,k}. \tag{2.30}$$

where $W_{c,k}$ is the compressor flow, T_a is the ambient temperature which is assumed to be constant, $p_{pre,k}$ is the prethrottle pressure, p_a is the ambient pressure which is assumed to

be constant, Γ_k is the head parameter, and $U_{c,k}$ is the blade tip speed. The maximum turbocharger speed, $\bar{\omega}_t$, is first mapped to a maximum prethrottle pressure, $\bar{p}_{pre,k}$, based on current measured compressor flow, $W_{c,k}$, maximum blade tip speed, $\bar{U}_c = \frac{\pi}{60} d_c \bar{\omega}_t$, and (2.28)-(2.30),

$$\bar{p}_{pre,k} = \left(\left(\frac{\left(\bar{U}_c\right)^2}{2c_p T_a} \right) \left(\frac{\frac{\pi}{4} k_1 p_a d_c^2 \bar{U}_c + W_{c,k} k_2}{W_{c,k} - \frac{\pi}{4} \rho_a d_c^2 \bar{U}_c k_3} \right) + 1 \right)^{\frac{\gamma}{\gamma-1}} p_a. \tag{2.31}$$

From the maximum prethrottle pressure, $\bar{p}_{pre,k}$, the throttle flow model can be inverted to obtain the maximum intake pressure. The throttle flow model follows [22],

$$W_{th,k} = \frac{p_{pre,k}}{\sqrt{R_{air} T_{ico}}} C_d \left(u_{th,k}, N_{e,k}\right) \phi \left(\frac{p_{in,k}}{p_{pre,k}} \right), \tag{2.32}$$

where ϕ is the same as in (2.2). The intercooler out temperature, T_{ico}, is assumed to be constant ($30^o C$ is used for the simulations and experiments that will be shown), and the throttle flow is modeled as a first order lag of compressor flow to approximate the prethrottle filling dynamics,

$$W_{th,k} = \alpha_{pre} W_{th,k-1} + (1 - \alpha_{pre}) W_{c,k}. \tag{2.33}$$

Let

$$\bar{\phi}_k = \frac{W_{th,k} \sqrt{R_{air} T_{ico}}}{C_d(u_{th,k}, N_{e,k}) p_{pre,k}}. \tag{2.34}$$

Then to enforce the turbocharger speed constraint, the intake pressure should remain below

$$\bar{p}_{in,k}^{remap} = \begin{cases} \bar{p}_{pre,k} \frac{-\frac{1}{\gamma} - \sqrt{\frac{1}{\gamma^2} - 4\left(\frac{\gamma-1}{2\gamma} - \bar{\phi}_k^2\right)\left(\frac{-\gamma-1}{2\gamma}\right)}}{\frac{-\gamma-1}{\gamma}} & \text{if } \frac{\bar{p}_{in,k}^{remap}}{p_{pre,k}} > \frac{1}{\gamma+1}, \\ \frac{1}{\gamma+1} \bar{p}_{pre,k} & \text{if } \frac{\bar{p}_{in,k}^{remap}}{p_{pre,k}} \leq \frac{1}{\gamma+1}. \end{cases} \tag{2.35}$$

2.3.4 Estimators

Several estimators are required to implement MPC on the DAP. The only measurements that are available on the target engine are compressor flow (W_c), intake pressure (p_{in}), and intake temperature (T_{in}). The production ECU estimate of cylinder flow (W_{cyl}) will also be

used directly. The signals that must be estimated are (i) the non-physical rate-based model state, Δx_k in (2.9), (ii) the EGR flow, W_{EGR}, and (iii) the exhaust pressure, p_{ex}.

The non-physical rate-based state, Δx_k is estimated using a Kalman filter,

$$\Delta\hat{x}_k = A\Delta\hat{x}_{k-1} + B\Delta u_{k-1} + L\left(\Delta\tilde{y}_{k-1}^{meas} - C\Delta\hat{x}_{k-1}\right), \tag{2.36}$$

where the gain, L, is designed for the incremental state and incremental output system and accounting for direct feed-through, i.e., based on the model,

$$\begin{aligned} \Delta x_{k+1} &= A\Delta x_k + B\Delta u_k, & (2.37)\\ \Delta\tilde{y}_k &= \Delta y_k - D\Delta u_k = C\Delta x_k. & (2.38) \end{aligned}$$

The EGR flow estimate is based on steady-state flow balance,

$$\hat{W}_{EGR,k} = W_{cyl,k} - W_{c,k}. \tag{2.39}$$

An input observer based estimator for EGR flow [102] was also investigated to improve transient estimation of EGR flow, however, it did not provide reliable results due to inaccuracies in the ECU estimated cylinder flow, sensor dynamics, and ECU signal conditioning.

The exhaust pressure is estimated by inverting the EGR flow model (2.1)-(2.2) given the current EGR flow estimate, EGR valve position, engine speed, and intake pressure. Because the EGR cooler out temperature is not measured, it is modeled as an affine function of intake temperature,

$$\hat{T}_{eco,k} = \beta_1 T_{in,k} + \beta_2. \tag{2.40}$$

A damped version of the model inversion is used to avoid a singularity when the EGR valve position is near zero. This is done by solving the following optimization problem,

$$\min_{\Delta p_{ex\,cl,k}} \frac{1}{2}\left(\hat{W}_{EGR,k} - W_{EGR}\left(\hat{p}_{ex,k}(\Delta p_{ex\,cl,k})\right)\right)^2 + \frac{1}{2}\kappa\Delta p_{ex\,cl,k}^2, \tag{2.41}$$

in which $\hat{W}_{EGR,k}$ denotes the current EGR flow estimate (2.39) and $W_{EGR}(p_{ex})$ denotes, with slight abuse of notation, the EGR flow as a function of exhaust pressure (2.1)-(2.2). The final exhaust pressure estimate is $\hat{p}_{ex,k}(\Delta p_{ex\,cl,k}) = p_{ex\,ol,k} + p_{ex\,cl,k-1} + \Delta p_{ex\,cl,k}$ which is the summation of an open loop estimate ($p_{ex\,ol}$), the previous "closed loop" estimate ($p_{ex\,cl,k-1}$), and the "closed loop" estimate increment ($\Delta p_{ex\,cl,k}$). The term $\kappa \Delta p_{ex\,cl,k}^2$ in (2.41) provides a calibratable estimator "gain." Adding the open loop estimate is beneficial as it will allow the total estimate to update when the EGR valve is closed; also, its use lowers the necessary closed loop estimator gain. The open loop estimate of the exhaust pressure is based on a polynomial model fitted to steady-state experimental data,

$$p_{ex\,ol,k} = \beta_3 + \beta_4 p_{in,k} + \beta_5 p_{in,k}^2 + \beta_6 p_{in,k} N_{e,k} + \beta_7 N_{e,k}^2 + \beta_8 N_{e,k}. \tag{2.42}$$

The optimization problem (2.41) is solved using Newton's Method which gives the following update for the exhaust pressure estimate,

$$\Delta p_{ex\,cl,k} = \left(\frac{\partial W_{EGR}}{\partial p_{ex}} \left(\hat{p}_{ex,k-1} \right) \right) \frac{\hat{W}_{EGR} - W_{EGR}\left(\hat{p}_{ex,k-1} \right)}{\left(\frac{\partial W_{EGR}}{\partial p_{ex}} \left(\hat{p}_{ex,k-1} \right) \right)^2 + \kappa}, \tag{2.43}$$

$$p_{ex\,cl,k} = p_{ex\,cl,k-1} + \Delta p_{ex\,cl,k}, \tag{2.44}$$

$$\hat{p}_{ex,k} = p_{ex\,ol,k} + p_{ex\,cl,k}. \tag{2.45}$$

The exhaust pressure estimate updates (2.43)-(2.45) are applied once per time step.

2.3.5 Experimental Controller Setup

In the following, the experimental LMPC controller setup is described using a rate-based strategy with ICE, constraint remapping, and associated estimators. The control objective is to regulate the intake pressure and EGR flow to prescribed set-points, $r_k = \begin{bmatrix} r_{pin,k} & r_{WEGR,k} \end{bmatrix}^T$, through coordinated control of the VGT, EGR valve, and EGR throttle. The constraints considered are the actuator limits and output constraints on MAP, EGR

flow, exhaust pressure, and turbocharger speed. The four state model obtained after system identification and model order reduction (2.6)-(2.7) will be used, where the control inputs contain the VGT position command and EGR flow command, $u = \begin{bmatrix} u_{VGT} & v_{EGR} \end{bmatrix}^T$. The outputs contain the intake pressure, estimated EGR flow, and estimated exhaust pressure, $y = \begin{bmatrix} p_{in} & \hat{W}_{EGR} & \hat{p}_{ex} \end{bmatrix}^T$. Following (2.8)-(2.14), the linear rate-based model for the diesel air path can be formed with

$$\xi_k = \begin{bmatrix} \Delta x_k \\ p_{in,k-1} - r_{pin,k-1} \\ \hat{W}_{EGR,k-1} - r_{WEGR,k-1} \\ \hat{p}_{ex,k-1} \end{bmatrix} = \begin{bmatrix} \Delta x_k \\ e_{pin,k} \\ e_{WEGR,k} \\ \hat{p}_{ex,k-1} \end{bmatrix}, \Delta u_k = \begin{bmatrix} \Delta u_{VGT,k} \\ \Delta v_{EGR,k} \end{bmatrix}, e_k = \begin{bmatrix} e_{pin,k} \\ e_{WEGR,k} \end{bmatrix}. \tag{2.46}$$

For notational purposes, let $\tilde{A}$ and $\tilde{B}$ be sub-matrices of $\bar{A}$ and $\bar{B}$ in (2.14) with (2.46) corresponding to the states $\begin{bmatrix} \Delta x_k^T & e_{pin,k}^T & e_{WEGR,k}^T \end{bmatrix}^T$.

A single step control horizon is chosen to reduce the number of optimization variables and computational complexity as much as possible. Through an appropriate choice of the terminal state penalty weight matrix, P, the infinite horizon Linear Quadratic Regulator (LQR) gain is recovered when constraints are not active. A slack variable, ϵ, is used to treat the output constraints as soft. This ensures feasibility of the optimization problem which could otherwise be lost due to disturbances and plant/model mismatch. The rate-based MPC cost functional has the following form,

$$J = \left(\tilde{A}\xi_0 + \tilde{B}\Delta u_{0|k}\right)^T P \left(\tilde{A}\xi_0 + \tilde{B}\Delta u_{0|k}\right) + \Delta u_{0|k}^T R \Delta u_{0|k} + M\epsilon^2, \tag{2.47}$$

where the control increment, $\Delta u_{0|k}$, and slack variable, ϵ, are optimization variables with $M > 0$. The state penalty, P, is chosen to be the solution to the Discrete Algebraic Riccati Equation (DARE) corresponding to the the system (2.12)-(2.13) and (2.46) and unconstrained cost, $\sum_{i=0}^{\infty} e_{i|k}^T Q e_{i|k} + \Delta u_{i|k}^T R \Delta u_{i|k}$, with weights $R > 0$, and $Q \geq 0$. Given this

choice of P, the unconstrained minimization of (2.47) with respect to $\Delta u_{0|k}$ results in a control law,

$$\Delta u_{0|k} = -\left(\tilde{B}^T P \tilde{B} + R\right)^{-1} \tilde{B}^T P \tilde{A} \xi_k = K_{LQR} \xi_k, \tag{2.48}$$

which exactly recovers the unconstrained LQR gain, K_{LQR}. Note that unconstrained stability is obtained through this choice of cost (2.47) and the unconstrained controller inherits all properties of the associated LQR controller.

Constraints on the control increment, $\Delta u_{0|k}$, are added to ensure that the applied control with the optimized control increment, $u_k = \Delta u^{\star}_{0|k} + u_{k-1}$, satisfies the control constraints,

$$\begin{aligned} \underline{u}_{VGT} - u_{VGT,k-1} \leq \quad & \Delta u_{VGT,0|k} \quad \leq \bar{u}_{VGT} - u_{VGT,k-1}, \\ -v_{EGR,k-1} \leq \quad & \Delta v_{EGR,0|k} \quad \leq \min\left\{\bar{W}_{EGR}(N_{e,k}, W_{f.k}), \bar{v}_{EGR.k}\right\} - v_{EGR,k-1}, \end{aligned} \tag{2.49}$$

where $u_{VGT,k-1}$ and $v_{EGR,k-1}$ are the VGT command and EGR flow command applied at the previous sample instant. Next, output constraints are added as

$$\begin{aligned} e_{pin,i} \quad &\leq \quad \min\{\bar{p}_{in} - r_{pin,k-1}, \bar{p}^{remap}_{in,k} - r_{pin,k-1}\} + \alpha_1 \epsilon, \; \forall i \in N_{ICE,pin} \subset \mathbb{Z}^+, \\ \hat{p}_{ex,i} \quad &\leq \quad \bar{p}_{ex} + \alpha_2 \epsilon, \; \forall i \in N_{ICE,pex} \subset \mathbb{Z}^+, \end{aligned} \tag{2.50}$$

where $\mathbb{Z}^+$ denotes the set of positive integers and where α_1, $\alpha_2, > 0$ are slack weights which can be used to assign priority to the different output constraints while using a single slack variable, ϵ. Smaller α's correspond to higher priority. For prediction, it is assumed that the controls are constant after the initial predicted time step, i.e., $\Delta u_{i|k} = 0$, $\forall i \geq 1$. The constraint values, $\bar{W}_{EGR,k}$, $\bar{v}_{EGR,k}$, and $\bar{p}^{remap}_{in,k}$ are assumed to be constant over the prediction/constraint horizon. Linear approximations of these constraint values as functions of states could be made as in [54,98], however this adds unnecessary computational complexity from a practical standpoint.

In order to achieve drastic reduction in the computational complexity for the diesel air path application, the tightened output constraints on intake pressure and exhaust pressure

are enforced at only a single time step so that $N_{ICE,pin} = N_{ICE,pex} = \{78\}$ (2.5 sec ahead based on a 32 msec sampling period). This choice of intermittent constraint is placed near steady state part of the predicted response, which is a reasonable choice because no overshoot to open loop step responses is exhibited in the primary measured output of interest, MAP, and a control horizon of one is used.

The resulting MPC optimization for the diesel air path application follows,

$$\begin{aligned}
\min_{\Delta u_0, \epsilon} \quad & \text{cost function (2.47)}, \\
\text{subject to :} \quad & \text{state equations (2.12)} - \text{(2.13) and (2.46)}, \\
& \text{control constraints (2.49)}, \\
& \text{output constraints (2.50)}, \\
& \Delta u_{i|k} = 0, \, \forall i \geq 1, \\
& \xi_{0|k} = \begin{bmatrix} \Delta \hat{x}_k^T & p_{in,k-1} - r_{pin,k-1} & \hat{W}_{EGR,k-1} - r_{WEGR,k-1} & \hat{p}_{ex,k-1} \end{bmatrix}^T .
\end{aligned} \tag{2.51}$$

In the implementation of the explicit MPC solution to (2.51), the constraints as well as the states are treated as input parameters, ζ_k. This results in an input parameter vector of size 13,

$$\zeta_k = \begin{bmatrix} \Delta \hat{x}_k \\ p_{in,k-1} - r_{pin,k-1} \\ \hat{W}_{EGR,k-1} - r_{WEGR,k-1} \\ \hat{p}_{ex,k-1} \\ \bar{u}_{VGT} - u_{VGT,k-1} \\ u_{VGT,k-1} - \underline{u}_{VGT} \\ \min\left\{\bar{W}_{EGR}(N_{e,k}, W_{f,k}), \bar{v}_{EGR,k}\right\} - v_{EGR,k-1} \\ v_{EGR,k-1} \\ \min\{\bar{p}_{in} - r_{pin,k-1}, \bar{p}_{in}^{remap} - r_{pin,k-1}\} \\ \bar{p}_{ex} \end{bmatrix}. \tag{2.52}$$

The explicit piecewise affine (PWA) control law with parameters ζ_k has the form

$$\Delta u^{\star}_{0|k} = G_j \zeta_k + F_j \text{ if } K_j \zeta_k \leq L_j, \text{ for } j \in \{1, ..., N_r\}\,, \tag{2.53}$$

where N_r denotes the number of regions generated. The control to be applied to the plant is $u_k = \Delta u^{\star}_{0|k} + u_{k-1}$.

Remark: Stabilizing terminal set constraints can formally be added to the rate-based MPC optimization problem (2.51), see [8]. However, the addition of a terminal set constraint with a short control horizon greatly limits the size of the feasible region of the optimization problem. In the presence of large fuel disturbances along with model uncertainty, states can easily be pushed outside of the feasible region. Guaranteeing recursive feasibility under all possible uncertainties (and reference changes) would result in a very conservative controller. Robust design can also be considered, i.e., constraint enforcement under bounded disturbances [9, 75]. Such a robust design has been explored in our previous work for the diesel air path [42] and is further developed in Chapter 4 of this dissertation. The controller so far developed in this chapter relies on soft constraints to guarantee recursive feasibility of the optimization problem and a posteriori analysis to establish stability, see Section 2.4. Further developments on a single step rate-based MPC controller with stability guarantees based on a Control Lyapunov Function (CLF) is also described in Section 2.4.

2.3.6 Experimental Results

Experiments with rate-based MPC have been conducted on a 3L displacement 4 cylinder diesel engine with a dSPACE rapid prototyping unit at Toyota Motor Corporation. The test results indicate good tracking performance of requested intake manifold pressure (MAP) and estimated EGR flow set-points on the New European Drive Cycle (NEDC) and Worldwide harmonized Light vehicles Test Procedure (WLTP). The ability to handle constraints on maximum MAP, and exhaust pressure in response to fuel steps, which also correspond to

steps in intake pressure and EGR flow set-points, is demonstrated. The turbocharger speed constraint handling is demonstrated on the NEDC. The sampling period is 32 msec.

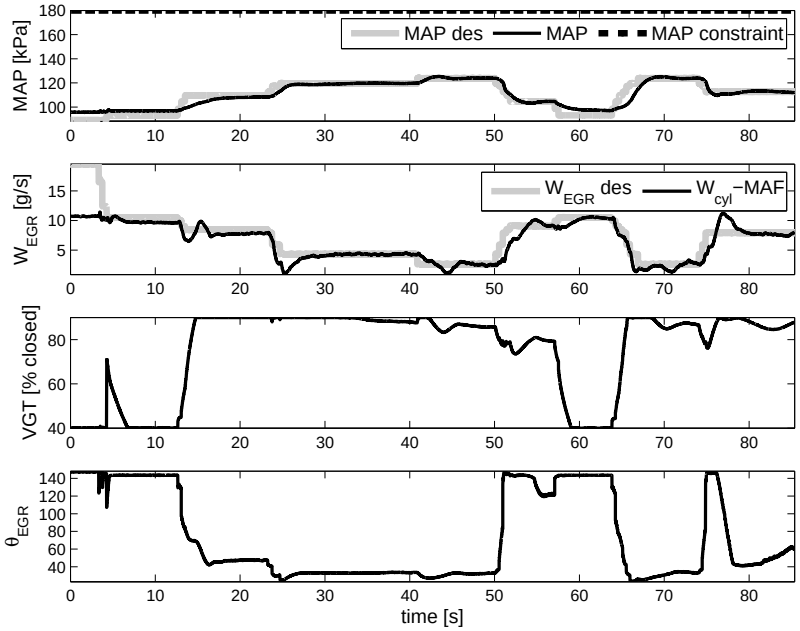

Figure 2.11: Fuel step experiments conducted at 1200 rpm.

All of the following results were conducted with a single rate-based MPC controller, with a control horizon of one, and whose nominal operating point and single linear design model are at 1600 rpm engine speed and 30 mm^3/st. fueling rate in the center of the NEDC range. The tuning matrices, Q and R, are tuned based on engine response to ± 5 mm^3/st. fuel steps and corresponding reference steps around the nominal operating point. The tuning was done emphasizing MAP tracking response over EGR flow because the EGR flow estimator, (2.39), is only valid at steady state. The Kalman filter (2.36) is not tuned assuming that the associated covariance matrices are identity. The exhaust pressure estimator gain, κ in (2.43), is tuned such that the estimator response to EGR valve and VGT steps around the nominal operating point is as fast as possible without exhibiting overshoot. The EGR flow estimator, (2.39), is based on measured MAF and ECU estimated cylinder flow. The latter is based on measured intake pressure and volumetric efficiency maps.

Remark: For the experiments that are presented here, a simplified version of the gain

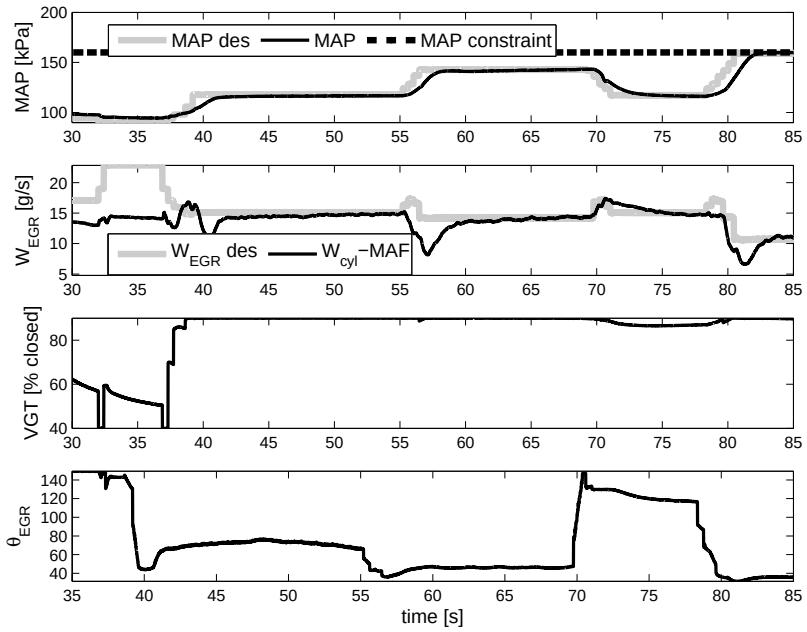

Figure 2.12: Fuel step experiments conducted at 1600 rpm.

scheduling strategy presented in Chapter 3 is used where only a diagonal gain is placed at the output of the explicit MPC controller. This was done in order to quickly calibrate the controller and improve tracking performance in limited experiment time. This does not diminish the claim that only a single rate-based MPC controller is sufficient to cover the entire engine operating range. Simulations without gain scheduling performed in loop with a high fidelity physics based nonlinear model will also be shown demonstrating that indeed a single zone is sufficient. This has also been shown in simulation in a previous publication, [39], and in preliminary experiments presented in [40].

Figure 2.11 shows the results of an experiment conducted at 1200 rpm with the fueling rate varying between 5 and 60 mm^3/st. The fuel profile is constructed through small 5 mm^3/st. steps. Between 0 and 25 sec and between 57 and 65 sec, tracking in both MAP and EGR flow channels is lost. This is because the combination of the two set-points is infeasible and at least one actuator (u_{VGT} and/or θ_{EGR}) is saturated. Otherwise, steady state tracking is demonstrated. Figure 2.12 shows the results of an experiment conducted at 1600 rpm with the fueling rate varying between 5 and 60 mm^3/st. In this case, tracking of

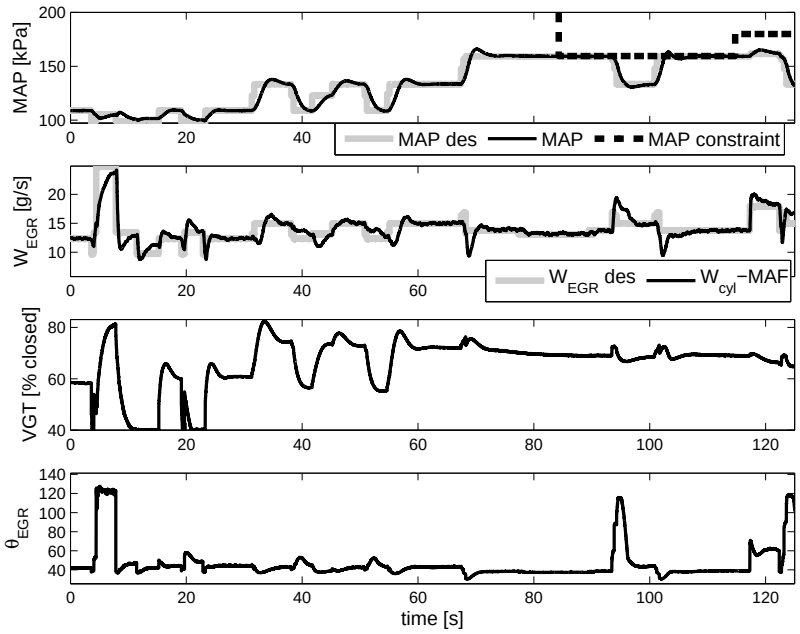

Figure 2.13: Fuel step experiment conducted at 2400 rpm.

EGR flow is slow because the VGT is saturated and the EGR valve is primarily being used to improve MAP tracking. This reflects the tuning choice which emphasizes MAP tracking and discounts transient EGR flow tracking because the EGR flow estimate (2.39) is only valid at steady state.

Figure 2.13 shows the results of an experiment conducted at 2400 rpm. Figure 2.14 shows a zoomed view of the MAP response at 70 sec overlaid with the MAP response at 100 sec from Figure 2.13 that compares an unconstrained response versus a response with a MAP constraint imposed. The overshoot is reduced in the constrained case as compared to the unconstrained case. In the constrained case, the constraint is momentarily violated because (i) the fuel step acts as a large disturbance that the air path MPC controller has no control over, (ii) the constraint is treated as soft, and (iii) there is a difference between the linear prediction model and true nonlinear plant. This behavior could be improved by including a disturbance model with respect to the measured fuel rate and through the gain scheduling strategy described in Chapter 3. Note that output constraints are always treated as soft to maintain feasibility. Figure 2.15 shows an experiment conducted at 1600 rpm demonstrating

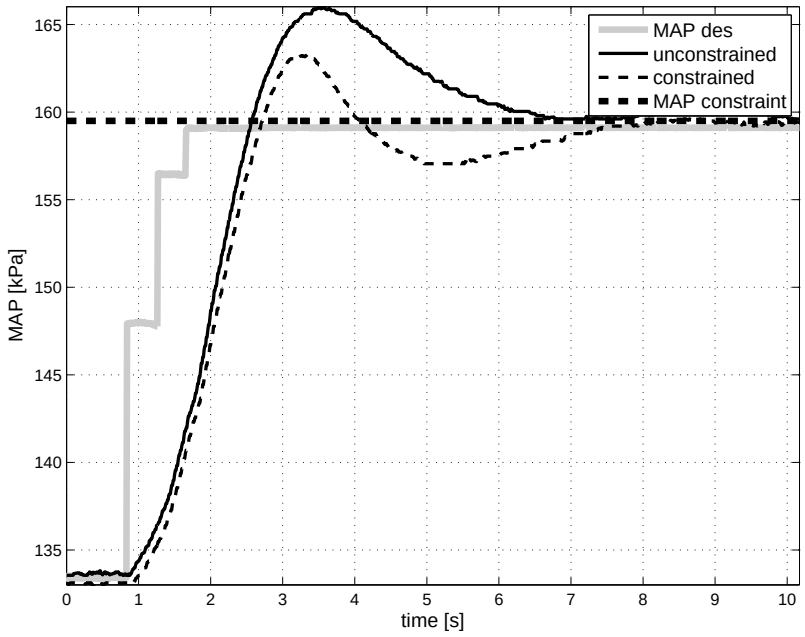

Figure 2.14: Comparison of constrained vs. unconstrained MAP response to a fuel step at 2400 rpm.

exhaust pressure constraint enforcement during a fueling rate step from 20 to 40 mm^3/st. Again, slight violation of the constraint is seen during the transient due to the fuel step, and a small violation is observed at steady state because the model predictive controller is trying to balance the cost of the soft constraint and tracking objectives. Figure 2.16 shows the results of an experiment conducted during the Extra Urban Drive Cycle portion of the NEDC demonstrating turbocharger speed constraint enforcement. This experiment demonstrates both intake pressure and turbocharger speed constraint enforcement. It can be seen that the strategy of remapping the turbocharger speed constraint to an intake pressure constraint is effective. Note that the VGT is already very oscillatory during the tip-in before the constraint activates. This is due to an aggressive controller tuning used during this experiment. These oscillations disappear in Figure 2.17, see between 500 and 600 sec, once the controller is re-tuned.

Finally, Figure 2.17 shows the tracking performance of the controller (with lower controller gain compared to Figure 2.16) during a portion of the NEDC containing the last

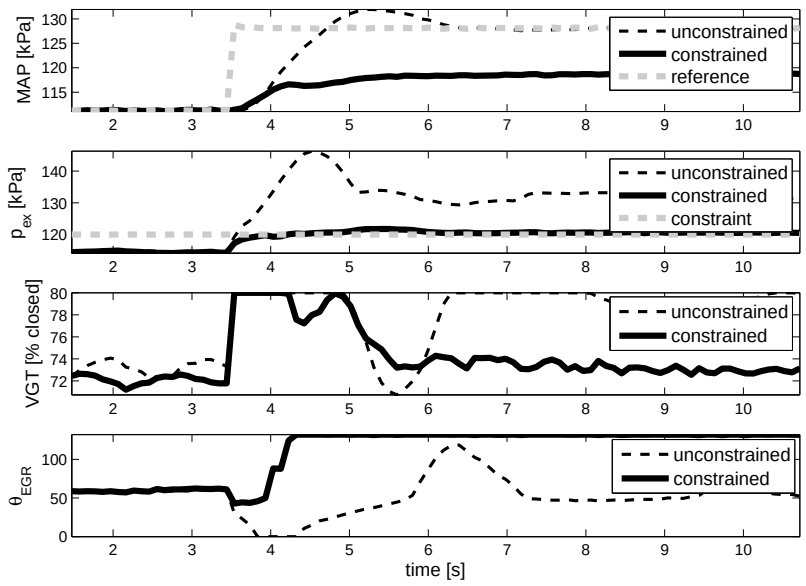

Figure 2.15: Demonstration of exhaust pressure constraint handling during a fuel step at 1600 rpm.

portion of the Urban Drive Cycle and full Extra Urban Drive Cycle and Figure 2.18 shows the last 500 sec of the WLTP. Note that feed-forward is not used in these experiments to fully test the ability of the feedback controller. The controller demonstrates very good tracking performance on intake pressure. There is a graceful loss of tracking at idle, e.g., at the beginning and end of the NEDC experiment, due to actuator saturation and loss of control authority. EGR flow tracking is good in slow transients, however performance suffers during large, faster transients. This is due to a tuning choice to de-emphasize EGR tracking cost due to the lack of a good transient EGR flow estimator. Highly oscillatory/noisy behavior is seen, for example, at 50 sec of the NEDC which is due to high sensitivity of the set-point maps to variations in engine speed. High activity in the VGT can be seen over the WLTP where the the VGT position swings between the minimum position and maximum position, e.g., between 1500 and 1550 sec. This high VGT activity may be undesirable and can be mitigated through changing the Q and R weights. This retuning is investigated in further simulations. The tuning that was used in the experiments in Figure 2.17 is $Q = diag([5,\ 1])$ and $R = diag([70,\ 20])$. When the controller is run with this tuning in loop with the high

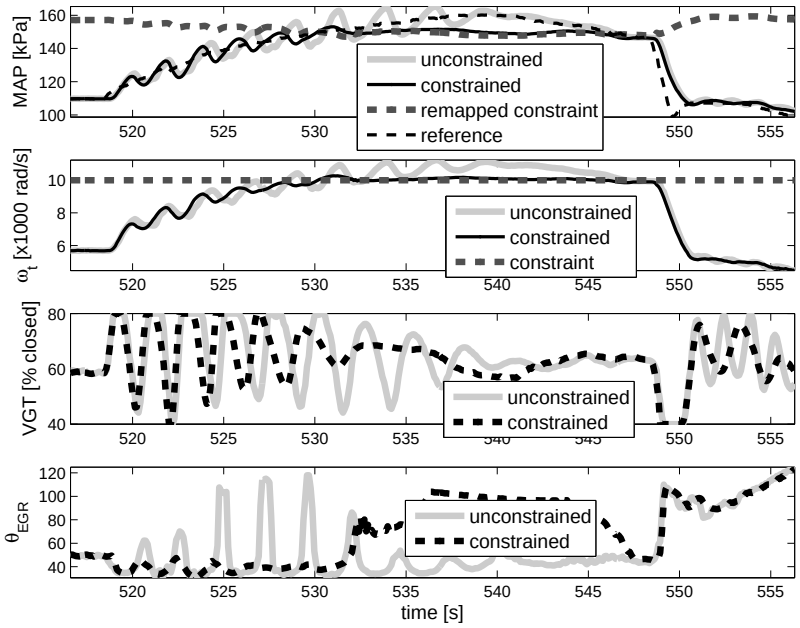

Figure 2.16: Demonstration of turbocharger speed constraint handling during the NEDC.

fidelity nonlinear model, similarly high VGT activity can be seen between 1550 sec and 1560 sec which also results in a oscillatory response in MAP, see Figures 2.19 and 2.20. When the controller is re-tuned with $Q = diag([1,\ 5])$ and $R = diag([300,\ 20])$, which now emphasizes W_{EGR} tracking and increases the penalty on VGT effort, the oscillations in VGT and MAP between 1550 sec and 1560 sec is smoothed out. Furthermore, the W_{EGR} tracking is much improved over the WLTP. Also note that no gain scheduling is used in these simulations, which indicates that a single controller is able to cover the entire engine operating range.

2.3.7 Computational Complexity

Table 2.1 compares the computational complexity of various MPC designs. MPC-A represents the direct application of integral augmented MPC to the diesel air path. MPC-B represents a design that utilizes partial nonlinear inversion of EGR flow to EGR valve position in the approach used by [38]. In [38], additional strategies such as elimination of rarely visited regions are employed for complexity reduction of MPC-A and MPC-B. These strategies are not reflected in Table 2.1 to provide a direct comparison to rate-based MPC (RB-MPC). N_z

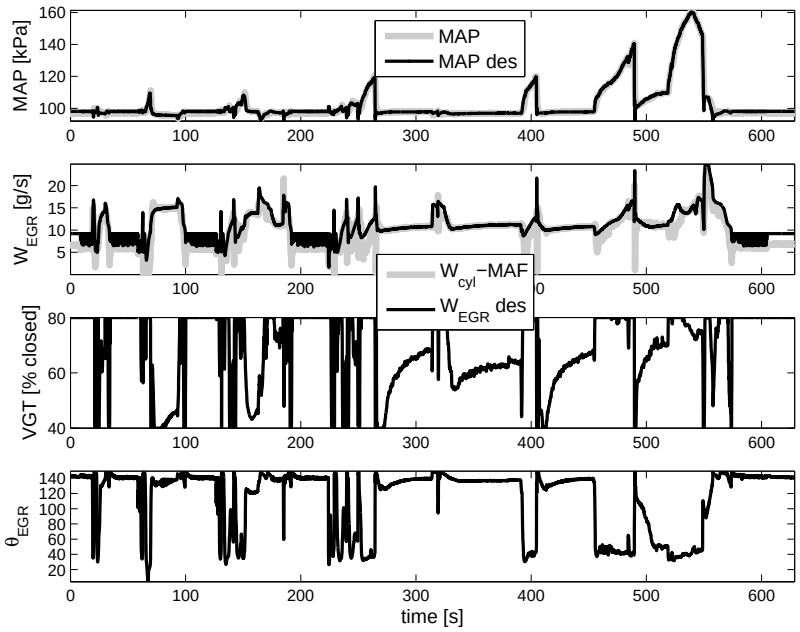

Figure 2.17: Experiments conducted on the NEDC demonstrate MAP and EGR flow tracking using a single zone LMPC controller.

denotes the number of zones used to split the engine operating range and number of local MPC controllers designed. N_y denotes the number of output constraints considered. In the cases of MPC-A and MPC-B, where $N_y = 1$, only the maximum intake pressure constraint is considered. In the RB-MPC case where $N_y = 4$, constraints are considered for the maximum intake pressure, exhaust pressure, remapped turbocharger speed, and remapped EGR flow constraint. The variable N_p denotes the number of input parameters to the explicit MPC controller, i.e., number of states, outputs, references, constraint settings, and etc. The N_c denotes the total number of constraints used per zone, i.e., the number of inequality constraints in (2.51). The control horizon for all of the compared MPC strategies is one. As previously noted in Section 1.2.3 the number of regions per zone, N_r, is $o(\mathcal{C}(N_c, N_o))$, see Definition 1.2, where N_o, the number of optimization variables, is three (two incremental control variables and one slack variable) for MPC-A, MPC-B, and RB-MPC. The complexity of storing the associated explicit MPC PWA control law over all zones is $o(N_z N_r N_c N_p)$, where $N_p = 15$ for MPC-A and MPC-B. The number of parameters, N_p, is composed of 4 states, 4 control constraint values, 1 output constraint value, 2 references, 2 integrator states, and

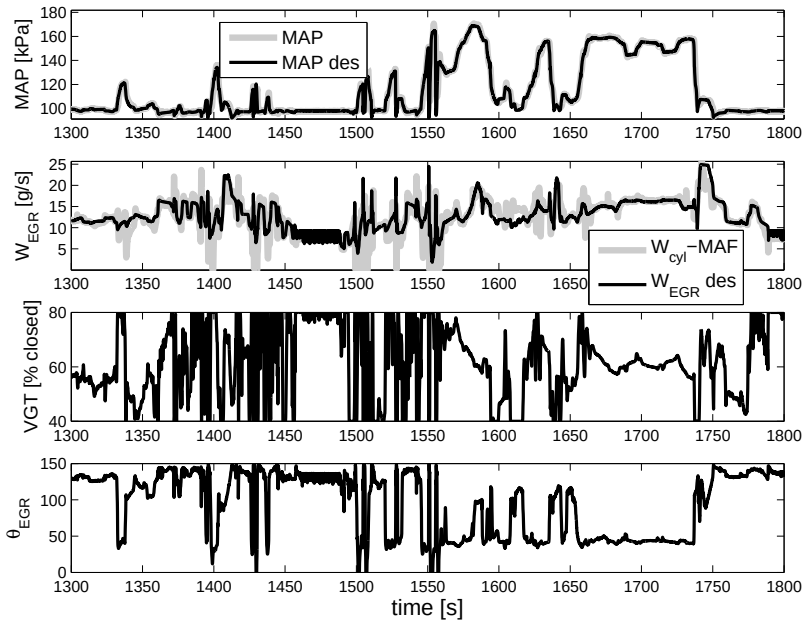

Figure 2.18: Experiments conducted on the WLTP demonstrate MAP and EGR flow tracking using a single zone LMPC controller.

2 control states. The number of parameters, N_p, is 13 for RB-MPC and is composed of 4 states, 4 control constraint values, 2 output constraint values, and 3 augmented output states. The complexity of evaluating the PWA control law is $o(N_r N_c N_p)$. Note that the computational cost of determining the zone is marginal compared to the rest of the cost, since zones are typically boxes rather than general polytopes as is the case for regions.

The computation time is estimated for a mid-range ECU, e.g., Freescale's MPC5644A microcontroller [26], with a 160MHz clock speed based on a worst case Floating Point Operations (FLOPS) count.[2] Note that ECU's typically perform many other functions beside air path control. With RB-MPC, less than 1% of the ECU's capability is used based on a 32 msec sampling period, see Table 2.1. Further computation time gains can be made by exploiting the simple structure of explicit MPC together with the Multiply and Accumulate (MAC) specific hardware inside the ECU which is not accounted for in Table 2.1.

[2]The worst case FLOPS count is a combined number of addition and multiply operations, which can be explicitly counted as the control is a PWA function (2.53) assuming that all regions must be checked before the control is applied. The computation time can then be obtained through ECU specifications, [26], e.g., clock speed, and number of floating point instructions per cycle.

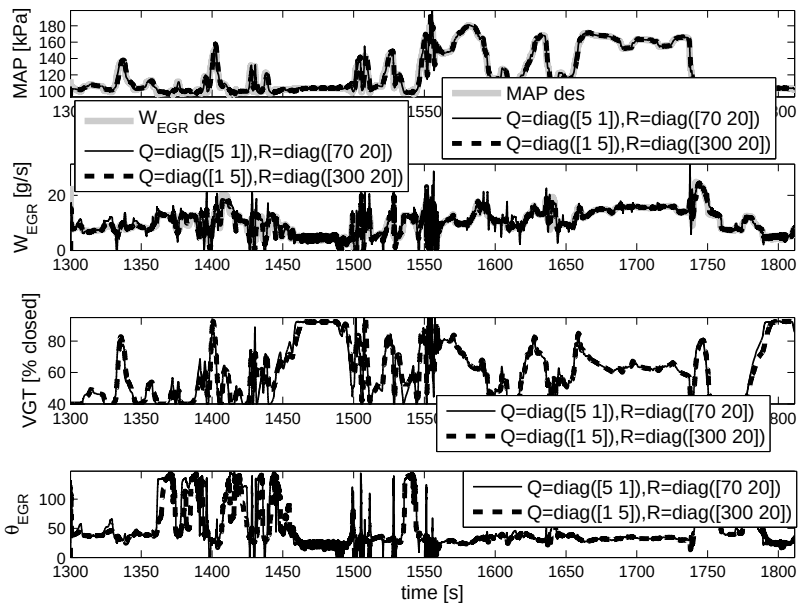

Figure 2.19: Simulations conducted on the WLTP comparing different Q and R tunings.

	N_z, N_y, N_c, N_p, ΣN_r	ROM[kB]	Time [μs]
MPC-A	15, 1, 34, 15, 945	548.1	725.6
MPC-B	2, 1, 6, 15, 65	37.7	374.1
RB-MPC	1, 4, 6, 13, 29	11.2	222.2

Table 2.1: Computational complexity of various MPC designs for the diesel air path.

Table 2.1 compares the computational complexity of the various methods without estimators, i.e., considering only the cost of evaluation the PWA control law (2.53). MPC-A enforces the intake pressure constraint through the entirety of a 30-step constraint horizon leading to a large computational complexity. Two zones are used with MPC-B since the diesel air path plant becomes "more linear" when treating EGR flow as a control input. The two zones separate a high engine speed zone from a low engine speed zone. Only two steps of intermittent constraint enforcement are used to enforce the intake pressure constraint which results in a dramatic computation time reduction compared to MPC-A. The final RB-MPC design presented in this chapter utilizes only a single zone and remaps the turbocharger and EGR flow constraints to existing constraints, leading to a lower computation time and ROM size, while enforcing more constraints. With only a single instance of intermittent constraint enforcement per output constraint, the total number of constraints, N_c, is the same

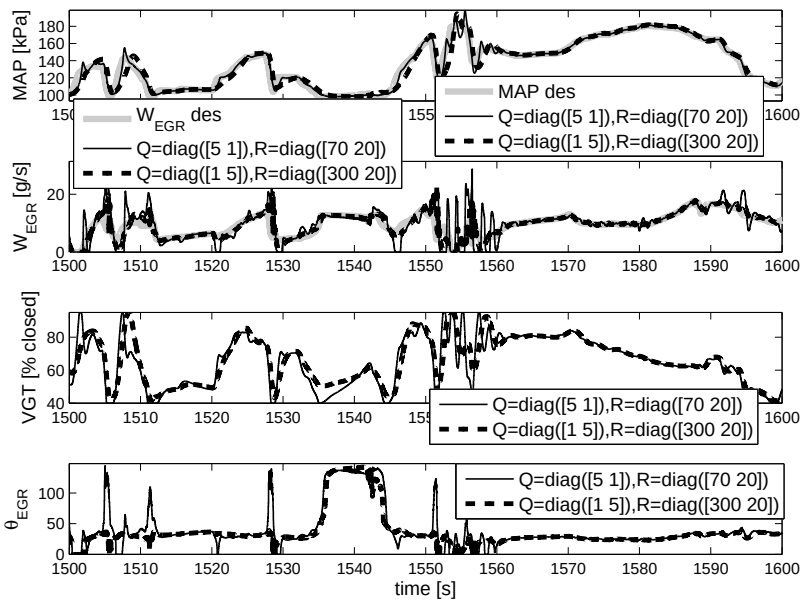

Figure 2.20: Zoomed view of simulations conducted on the WLTP comparing different Q and R tunings.

for RB-MPC and MPC-B. However, computation time for MPC-B is still larger because the references must be added as additional parameters to the optimization problem, leading to a larger PWA representation. This not the case for RB-MPC where the augmented state only needs to be driven to zero. Further, note that when all of the output constraints are considered, the estimators would be the same for MPC-A, MPC-B, and RB-MPC.

2.4 Stability of the DAP MPC Controller

For the diesel air path application, the optimization problem is kept small, with a small number of optimization variables and constraints, as an approach toward developing a computationally feasible model predictive controller. As remarked in Section 2.3.5, the designed controller does not incorporate a priori stability guarantees. Including a stability guarantee through incorporating a terminal set constraint, [72], may greatly increase the computational complexity of the explicit representation of the controller and limit the feasible region of the optimization problem. Furthermore, a terminal set constraint is typically used with

a long control horizon so that the feasible region of the optimization problem is not overly conservative, i.e, small. Instead, with the DAP MPC controller, local stability is guaranteed through a terminal penalty, P in (2.47), that is obtained from the corresponding solution to the infinite horizon unconstrained LQ problem.

This section presents another approach, contractive MPC (CMPC) [17, 24, 32], to guarantee local stability when a control and prediction horizon of 1 is to used. CMPC relies on explicitly enforcing a decay condition on a Lyapunov function. With CMPC, it has previously been unknown how to achieve zero-offset, steady-state tracking. An augmented integrator or disturbance estimator strategy will not work because the equilibrium state must be known a priori in order to use a Lyapunov function. A novel approach using rate-based CMPC avoids the issues of requiring a known equilibrium and is thus able to achieve zero-offset, steady-state tracking.

Even with CMPC, only local stability is achieved. Thus an a posteriori stability analysis tool is also developed to check the stability of an MPC controller and estimate the constrained domain of attraction.

2.4.1 Rate-based Contractive MPC

Contractive MPC (CMPC) utilizes the explicit enforcement of a Control Lyapunov Function (CLF) based constraint to achieve stability of the resulting MPC controller. Furthermore, a rate-based framework will be used to achieve zero-offset, steady-state tracking. Even though both CMPC and the MPC strategy described in Section 2.3 utilize a single step horizon and both guarantee local stability, CMPC has an advantage. With standard MPC, the cost functional must be chosen such that it also serves as an Lyapunov function to guarantee local stability, e.g., the terminal penalty, P, is chosen as the solution to the associated DARE, [72]. With CMPC, the cost functional can be chosen independently from the need to guarantee stability. This provides more freedom to tune the controller.

Let $\xi_k \in \mathbb{X} \subseteq \mathbb{R}^{n_x+n_y}$, where ξ_k is the rate-based state in (2.12) and $u_k \in \mathbb{U} \subseteq \mathbb{R}^{n_u}$ with

$u_k = \Delta u_k + u_{k-1}$ where $\mathbb{X}$ and $\mathbb{U}$ are specified sets. A function V is a local CLF in $\bar{\mathbb{X}} \subseteq \mathbb{X}$, for the system (2.12)-(2.14) if there exists a control law $\Delta u_k = \pi(\xi_k)$ such that

$$V(\xi_{k+1}) \leq V(\xi_k), \, \forall \xi \in \bar{\mathbb{X}}. \tag{2.54}$$

Flexibility in the Lyapunov stability condition (2.54) can be introduced by using a relaxation parameter τ. This allows the usage of the local CLF in a much larger subset of $\mathbb{X}$, see [32]. The enforcement of the CLF decay is done in the following manner. At each time step, k, a cost functional $J(\tau_k)$, a strictly increasing function of τ_k over Δu_k and τ_k, is minimized subject to the following constraints:

$$\begin{aligned} u_k \in \mathbb{U}, \, \xi_{k+1} \in \mathbb{X}, \, \tau_k \geq 0, \\ V(\xi_{k+1}) - \rho V(\xi_k) \leq \tau_k, \end{aligned} \tag{2.55}$$

where $\rho \in [0, 1)$ is the decay rate.

An infinity-norm CLF candidate for the construction of either an LP or QP implementation of the optimization problem subject to the constraints (2.55) is considered,

$$V(\xi) = ||P\xi||_\infty, \tag{2.56}$$

where $P \in \mathbb{R}^{p \times (n_x + n_y)}$ is a full column-rank matrix which can be determined by constructing a Lyapunov function for the pre-stabilized unconstrained system $\xi_{k+1} = (\bar{A} + \bar{B}K)\xi_k$, where the stabilizing gain K can be the result of an unconstrained LQR design. An infinity-norm Lyapunov function can then be generated using techniques in [63]. With an infinity-norm CLF candidate, the CLF decay condition in the constraint (2.55) becomes

$$||P(\bar{A}\xi_k + \bar{B}\Delta u_k)||_\infty - \rho ||P\xi_k||_\infty \leq \tau_k. \tag{2.57}$$

A constraint of the form $||P\xi||_\infty \leq c$ can be replaced by an equivalent set of linear

inequalities $\pm(P\xi)_m \leq c$, where m denotes the m-th row of $P\xi$. This results in constraints composed of $2p$ linear inequalities,

$$\pm(P(\bar{A}\xi_k + \bar{B}\Delta u_k))_m - \rho||P\xi_k||_\infty \leq \tau_k. \tag{2.58}$$

At each time instant k, the term $\rho||P\xi_k||_\infty$ in the equation (2.58) is computed outside the optimization problem and can be considered as an input parameter, e.g., to the PWA representation of the associated explicit MPC control law.

Simulations have been performed using CMPC in the loop with the high fidelity physics based model. The controlled inputs are VGT position and EGR flow. The tracked outputs are intake pressure and EGR rate. For the CMPC controller, the inversion of EGR flow to EGR valve position is done through inversion of the orifice for equation (2.3).[3] The maximum EGR flow, $\bar{v}_{EGR}(\bar{u}_{EGR}, N_e, p_{in}, \hat{p}_{ex}, \hat{T}_{eco})$, is obtained by evaluating the orifice equation (2.1)-(2.2) with the maximum valve position, $\bar{u}_{EGR}$, and current measurements or ECU estimates of engine speed, intake pressure, exhaust pressure, and EGR cooler out temperature. The constraints considered on the controls for the CMPC implementation are

$$\begin{aligned} \underline{u}_{VGT} - u_{VGT,k-1} \leq \quad & \Delta u_{VGT,0|k} \quad \leq \bar{u}_{VGT} - u_{VGT,k-1}, \\ -v_{EGR,k-1} \leq \quad & \Delta v_{EGR,0|k} \quad \leq \bar{v}_{EGR}(\bar{u}_{EGR}, N_{e,k}, p_{in,k}, \hat{p}_{ex,k}, \hat{T}_{eco,k}) - v_{EGR,k-1}. \end{aligned} \tag{2.59}$$

Note that the maximum EGR flow constraint in (2.59) is a nonlinear function of states, as opposed to $\bar{v}_{EGR,k}$ in (2.49) which is just based on a table-lookup as a function of the operating condition. The nonlinear EGR flow constraint (2.59) is easily handled in CMPC as, over a single time step, the constraint is affine with respect to the control.

A soft overshoot constraint is also imposed on intake pressure,

$$e_{pin,1|k} \leq \epsilon + \bar{p}_{in} - r_{pin,k-1}, \tag{2.60}$$

[3]Note that this implementation of CMPC pre-dates some of the observations and developments that led to the controller design described in Section 2.3. The target engine is also different. The CMPC implementation targets Toyota's AD engine rather than Toyota's KD engine. See [38, 41] for further details.

where ϵ is a slack variable and $\bar{p}_{in}$ is the intake pressure overshoot limit.

The cost functional in the CMPC problem is of the form,

$$J = (\bar{A}\hat{\xi}_k + \bar{B}\Delta u_{0|k})^T Q(\bar{A}\hat{\xi}_k + \bar{B}\Delta u_{0|k}) + \Delta u_{0|k}^T R \Delta u_{0|k} + M_1\tau_k^2 + M_2\epsilon^2, \tag{2.61}$$

and is minimized subject to the constraints,

$$\pm(P(\bar{A}\hat{\xi}_k + \bar{B}\Delta u_{0|k}))_m - \rho||P\xi_k||_\infty \leq \tau_k, \tag{2.62}$$

and (2.59)-(2.60). The cost functional penalizes the one step error through the term $(\bar{A}\hat{\xi}_k + \bar{B}\Delta u_{0|k})^T Q(\bar{A}\hat{\xi}_k + \bar{B}\Delta u_{0|k})$ with $Q = Q^T \geq 0$, the control effort through $\Delta u_{0|k}^T R \Delta u_{0|k}$ with $R = R^T > 0$, the Lyapunov decay constraint violation through $M_1\tau_k^2$ with $M_1 > 0$, and the overshoot constraint violation through $M_2\epsilon^2$ with $M_2 > 0$.

The final set of input parameters ζ_k for the explicit form of the CMPC controller is

$$\zeta_k = \begin{bmatrix} \hat{\xi}_k \\ \rho||P\xi_k||_\infty \\ \bar{u}_{VGT} - u_{VGT,k-1} \\ \underline{u}_{VGT} - u_{VGT,k-1} \\ \bar{v}_{EGR}(\bar{u}_{EGR}, N_{e,k}, p_{in,k}, \hat{p}_{ex,k}, \hat{T}_{eco,k}) - v_{EGR,k-1} \\ -v_{EGR,k-1} \\ \bar{p}_{in} - r_{pin} \end{bmatrix}. \tag{2.63}$$

In total for CMPC, there are 10 input parameters with $\hat{\xi}_k \in \mathbb{R}^4$, 4 optimization variables composed of the control increments and slacks, and 15 constraints composed of 8 from the CLF decay condition with $p = 4$ and 4 total min/max control constraints, and 1 overshoot constraint, and τ_k, $\epsilon_k \geq 0$. Note that a QP problem is formed in equation (2.61) rather than a LP as is done in [32]. This is because the QP formulation typically results in fewer regions

of the explicit control representation compared to a LP formulation. With a LP formulation, additional constraints and slacks are required to handle a step cost of infinity-norm type. The number of regions with the QP and LP formulations are 229 regions and 628 regions, respectively.

Simulation results of the CMPC controller in the loop with the high fidelity nonlinear DAP model are shown in Figures 2.21-2.24. The single linearization point for the CMPC prediction model is at 1750 rpm engine speed and 45mm^3/st. fuel rate. Figure 2.21 shows responses to fuel steps of 25 mm^3/st. $\pm$ 20mm^3/st. The decay rate, ρ, in (2.63) is set to 0.95, and the intake pressure overshoot constraint is set to +5 kPa. Feed-forward has not been added to demonstrate the capability of the feedback part of the controller. The time-constant for intake pressure response is fast, on the order of 1 sec and zero-offset, steady-state tracking error is achieved. The overshoot constraint can be seen to become active at 22 sec. Figure 2.22 shows the CLF decay after each fuel disturbance and reference change. Through NEDC simulations it can be seen that a single CMPC controller is able to stably control a large range of operating conditions (defined by fuel rate and engine speed). Figures 2.23-2.24 show a portion of a drive cycle simulation. The portion shown is characterized by engine speed ramps and fuel cuts. The fuel rate and engine speed vary between 0 mm^3/st. - 35 mm^3/st. and 1000 rpm - 2300 rpm, respectively. In Figure 2.24, the four spikes of EGR rate reference where there is a loss of EGR rate tracking corresponds to fuel cut events. EGR rate tracking is momentarily lost because the fuel flow instantaneously drops and exhaust pressure immediately drops to a point where intake pressure is greater than exhaust pressure so that no EGR flow is possible.

2.4.2 A Posteriori Stability Analysis

With the rate-based MPC controller described in Section 2.3 and with the CMPC controller described in Section 2.4.1, only local stability is guaranteed. Thus a tool is still desired to check for stability a posteriori, i.e., after the controller has been defined, and estimate

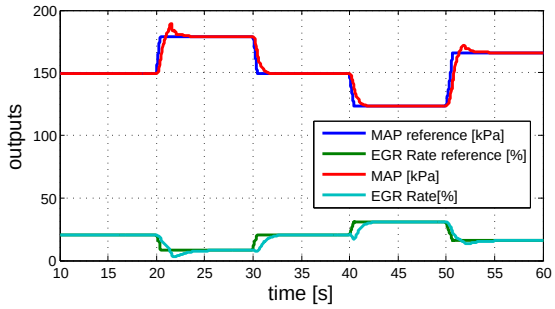

Figure 2.21: Responses of tracked outputs, MAP and EGR rate, to fuel steps.

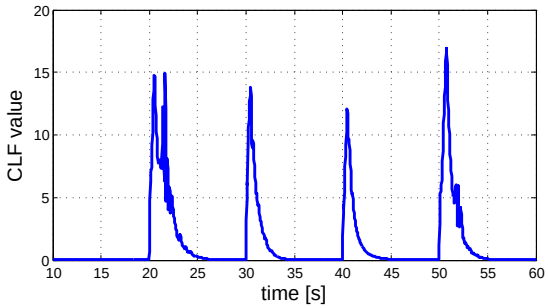

Figure 2.22: Response of CLF value to fuel steps.

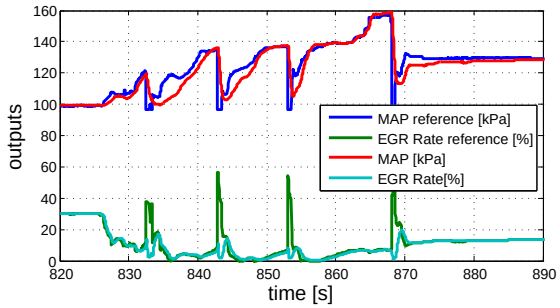

Figure 2.23: Responses of tracked outputs, MAP and EGR rate, on the NEDC.

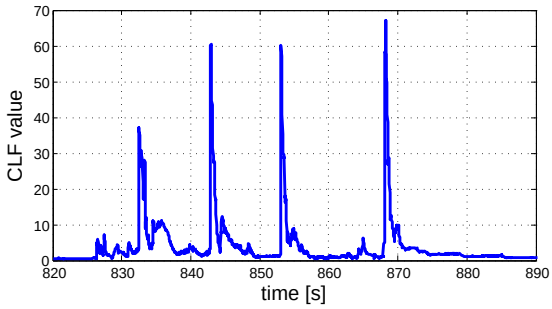

Figure 2.24: Response of CLF value on the NEDC.

procedure DECAY VIOLATION($x_{-1}, \hat{x}_{-1}, u_{-1}, u_{ss}$)

▷ Returns $c > 0$ if decay condition is violated and $c \leq 0$ if decay condition is met.

$r \leftarrow (C_i(I - A_i)^{-1}B_i + D_i)u_{ss}$

▷ Compute reference corresponding to steady state control.

$x_{ss} \leftarrow \left(I - \begin{bmatrix} A_i & 0 \\ LC_i & A - LC \end{bmatrix}\right)^{-1} \begin{bmatrix} B_i \\ B + LD_i - LD \end{bmatrix} u_{ss}$

▷ Compute steady state true and estimated states.

for $i \leftarrow 0, N$ **do** ▷ Simulate for N steps.

$y \leftarrow C_i x_{-1} + D_i u_{-1}$, $\hat{x}_0 \leftarrow A\hat{x}_{-1} + Bu_{-1} + L(y - C\hat{x}_{-1} - Du_{-1})$

$x_0 \leftarrow A_i x_{-1} + B_i u_{-1}$, $\Delta\hat{x}_0 \leftarrow \hat{x}_0 - \hat{x} - 1$, $e_0 \leftarrow y - r$, $\Delta\mathbb{U}_0 \leftarrow \mathbb{U} - u_{-1}$

$\Delta u^\star \leftarrow f^\star(\Delta\hat{x}_0, e_0, \Delta\mathbb{U}_0, \mathbb{Y})$ ▷ Compute delta observer state and optimal control.

if $i = 0$ **then** $b \leftarrow M||[([x_0^T \hat{x}_0^T] - x_{ss}^T)\ \ ([x_{-1}^T \hat{x}_{-1}^T] - x_{ss}^T)]^T||$

else $b \leftarrow \rho b$ ▷ Compute exponential decay bound.

end if

$c_i = \left|\left|\begin{bmatrix} ([x_0^T\ \hat{x}_0^T] - x_{ss}^T)^T \\ ([x_{-1}^T\ \hat{x}_{-1}^T] - x_{ss}^T)^T \end{bmatrix}\right|\right| - b$ ▷ Compute decay condition violation.

Note: we use both the current and delayed state to account for the delay introduced by a rate-based formulation.

$u_{-1} \leftarrow u_{-1} + \Delta u^\star\,, x_{-1} \leftarrow x_0\,, \hat{x}_{-1} \leftarrow \hat{x}_0$ ▷ Update states and controls.

end for

$c \leftarrow \max_{i \in \{0,\ldots,N\}} c_i$

return c

end procedure

Figure 2.25: Algorithm for computing the exponential decay condition violation.

the closed loop region of attraction. There are different approaches to a posteriori stability analysis. The Multi-Parametric Toolbox (MPT) [63] includes tools that attempt to find a quadratic or piece-wise quadratic Lyapunov function to certify stability. One could also try to find a quadratic Lyapunov function through solving a Linear Matrix Inequality (LMI) following the method in [89]. Both of these methods were not successful in finding a Lyapunov function for rate-based MPC controller described in Section 2.3 due to a large state-space size including both the states and state estimates, delayed states arising from a rate-based formulation, and the need to handle references as parameters. Thus, a different approach is taken that directly tries to establish exponential stability rather than trying to establish Lyapunov stability.

Definition 2.1: The origin is exponentially stable for a system, $x_{k+1} = f(x_k)$, if there exists $M > 0$ and $\rho \in (0, 1)$ such that $||x_k|| \leq M\rho^k||x_0||$ for all $k \geq 0$.

Rather than trying to construct a Lyapunov function, simulations can be performed to check for an exponential bound. Specifically, a nonlinear optimization problem can be solved that tries to find an initial condition that falsifies the exponential decay condition. If no such initial condition exists then the system is stable. In the following, let (A, B, C, D) correspond to the system matrices of the non-rate-based nominal model and let (A_i, B_i, C_i, D_i) correspond to system matrices of a non-rate-based off-nominal model. Let $\Delta u^\star = f^\star(\Delta\hat{x}_0, e_0, \mathbb{U}, \mathbb{Y}, u_{-1})$ be the optimal control law which is a function of the current state estimate increment, $\Delta\hat{x}_0$, error, e_0, control constraint set $\mathbb{U}_0$, output constraint set $\mathbb{Y}$, and previously applied control, u_{-1}. Also let L be the observer gain. Given an initial condition for the state, x_{-1}, state estimate $\hat{x}_{-1}$, control u_{-1}, and the expected steady state control u_{ss} corresponding to some reference, the procedure in Figure 2.25 can be used to determine if an exponential decay condition is violated.

When the nonlinear optimization problem is posed, linear constraints are required on the initial condition to ensure that the initial condition for the control u_{-1} is feasible and that the reference is reachable. Assume that the control constraints, $\mathbb{U} = \{u : \bar{u} \leq u \leq \underline{u}\}$, and output constraints, $\mathbb{Y} = \{y : \bar{y} \leq y \leq \underline{y}\}$, are of box type,

$$V \begin{bmatrix} x_{-1} \\ \hat{x}_{-1} \\ u_{-1} \\ u_{ss} \end{bmatrix} \leq \begin{bmatrix} \bar{u} \\ \bar{u} \\ -\underline{u} \\ -\underline{u} \\ \bar{y} \\ -\underline{y} \end{bmatrix}, V = \begin{bmatrix} 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \\ 0 & 0 & -I & 0 \\ 0 & 0 & 0 & -I \\ 0 & 0 & 0 & (C_i(I - A_i)^{-1}B_i + D_i) \\ 0 & 0 & 0 & -(C_i(I - A_i)^{-1}B_i + D_i) \end{bmatrix}. \tag{2.64}$$

Then for a given M and ρ, the following nonlinear optimization problem is solved,

$$\begin{aligned} \max_{x_{-1}, \hat{x}_{-1}, u_{-1}, u_{ss}} \quad & \text{DECAY VIOLATION}, \\ \text{subject to :} \quad & (2.64). \end{aligned} \tag{2.65}$$

If (2.65) is negative, then the system is stable. The optimization problem (2.65) is solved with multiple initial guesses and a nonlinear optimizer, e.g., MATLAB's fmincon. Using this method, it has been found that the controller is asymptotically stable for the set of initial conditions described by (2.64) at the nominal condition with $M = 10000$ and $\rho = 0.95$. Note that the set of initial conditions described by (2.64) is large and only requires that $u_{-1} \in \mathbb{U}$ and that there exists a steady state control $u_{ss} \in \mathbb{U}$ such that the corresponding $y_{ss} \in \mathbb{Y}$. Extensive nonlinear simulations and engine hardware experiments at various operating conditions have also confirmed closed-loop system stability properties.

2.5 Conclusions

A rate-based model predictive controller for diesel engine air path management by coordinating VGT, EGR valve, and EGR throttle actuators has been designed and experimentally validated. Using a rate-based MPC strategy, the engine operating range does not need to be partitioned into zones with a different controller designed and stored per zone (as in previous approaches). This reduces ECU memory requirements and calibration effort and avoids the need to address bumpless transfer as the controller can be discontinuous across zone boundaries. Compared to the augmented integral or disturbance estimator strategies, rate-based MPC does not require a disturbance model and does not require the previously applied control or reference to be treated as parameters to the optimization problem, thereby reducing the computation time and memory of the corresponding explicit MPC. Further computational complexity reduction can be achieved through the use of constraint remapping and the novel strategy of intermittent constraint enforcement. Furthermore, the control strategy enforces more constraints than has previously been demonstrated, e.g., a single constraint on soot in [56] or constraint on NOx in [101], with comparable computation time and lower ROM usage. Experimental tests have demonstrated that a controller based on a single zone can successfully accomplish both reference tracking and constraint enforcement. Stability of

rate-based MPC was also investigated. When a horizon of one is used, CMPC is an attractive way to obtain stability because performance tuning can be decoupled from providing a stability guarantee. A novel approach to CMPC using a rate-based form was used to achieve zero-offset steady-state tracking. With both standard rate-based MPC and CMPC controllers, only local stability is guaranteed. Thus an a posteriori stability analysis tool is also developed to check the stability of an MPC controller and estimate the closed loop region of attraction.

Chapter 3

Gain Scheduled Linear Model Predictive Control

This chapter describes a gain scheduling strategy that can be used in conjunction with explicit Model Predictive Control (MPC). Traditionally, explicit MPC is not reconfigurable to online model changes. To handle off-nominal plant conditions, a common practice is to design multiple explicit MPC's which are each valid locally around their respective operating points. This inevitably requires large amounts of memory to store the explicit MPC's and implementation of switching logic and observers. The gain scheduling strategy presented in this chapter bypasses the need to store multiple explicit MPC's. This is done by multiplying the control signal obtained from the nominal explicit MPC by a gain scheduling matrix such that the plant at off-nominal operating conditions is approximately matched to the nominal plant. This is further accomplished in a manner such that the original control constraints are satisfied. The gain scheduling strategy is demonstrated in simulations on a nonlinear diesel air path model over the New European Drive Cycle (NEDC).

3.1 Introduction

A common strategy when using linear model based explicit MPC for control of nonlinear plants is to define multiple operating conditions, design an explicit MPC [5] for each operating condition, and switch between the multiple MPC's as the operating condition changes [68]. This approach has been employed in previous diesel air path control applications [38,56,84].

It is also used by Honeywell's OnRAMP software [36] for systematic MPC design applied to powertrain control. This strategy comes at the cost of increased embedded processor memory usage for each additional off-nominal explicit MPC and calibration time for each controller. Furthermore, bumpless switching between controllers must be implemented.

The general design process of gain scheduled explicit MPC consists of the following main steps.

1. Define and tune a nominal explicit MPC for the nominal operating condition.

2. Through simulations or experiments, determine at which conditions the nominal explicit MPC no longer satisfies performance requirements.

3. Compute through linearization or system identification a linear model for the off-nominal condition.

4. Continue defining new operating conditions and linear models until the operating range is fully covered.

Performing this gain scheduling procedure can be cumbersome as there is no well-defined method for determining how to partition the plant operational space. The judgment of the engineer or calibrator with intuition of the nonlinear plant dynamics is typically required.

In addition to a reduction of memory requirements, as will be shown, the proposed gain scheduling method described in this chapter will reduce the need for defining operational zones of the plant.

The difference between the traditional strategy for gain scheduling explicit MPC and the strategy proposed in this chapter is that instead of modifying the controller to accommodate off-nominal operating conditions, the plant is pre-compensated at the input to resemble the nominal plant and to accommodate the nominal controller as is done in [28].

The gain scheduling strategy in [28] can be applied to unconstrained multi-input multi-output controllers, where an intermediate gain is placed between the nominal controller

output and the nonlinear plant. In this way, the combination of the intermediate gain and the off-nominal plant is made to approximate the nominal plant. See also [108, 109] where this strategy was employed with non-MPC controllers for diesel engines. While relatively simple, this technique has proven to be remarkably effective in engine control applications. When considering the use of explicit MPC as the nominal controller, the implementation of the approach of [28] is impeded by control constraints, as no guarantee exists that the control signal will satisfy them. This issue is successfully addressed and resolved in the new proposed strategy.

It will first be shown that, for 2 input, 2 output systems, diagonal or anti-diagonal 2×2 gain matrices on the output of the explicit MPC can be accommodated by treating control bounds as parameters in the explicit MPC formulation. A switching structure between the diagonal and anti-diagonal gains will be used to approximate the strategy of [28]. The switched explicit MPC (seMPC) converges to the strategy in [28] as the sampling frequency increases. The mechanism is similar to the well-known Pulse-Width-Modulation (PWM); the applied signal achieves the originally desired signal on average, or through natural filtering by plant dynamics. As will also be discussed, the method can be extended to systems with more inputs (and more outputs) at the cost of needing to either increase the sampling rate or reduce the control update frequency.

The seMPC strategy is simpler than conventional gain scheduled explicit MPC (gsMPC) in terms of microcontroller memory usage and calibration effort needed. While the performance of seMPC may, in general, be worse than gsMPC due to having fewer degrees of freedom to manipulate the model and closed-loop dynamics, it is found that for the highly nonlinear diesel air path application [61], the seMPC strategy with a single nominal linear model is sufficient for covering the entire engine operating range and performs comparably to gsMPC.

In the following, the seMPC strategy is developed for the 2 input, 2 output, diesel air path control application described in Chapter 2. The seMPC strategy is developed in a rate-based

framework to facilitate set-point tracking, however can easily be modified for non-rate-based MPC.

This chapter is organized as follows. Section 3.2 motivates the need for a switched gain scheduling structure for constraint handling in the context of explicit MPC and develops the seMPC strategy. Section 3.3 develops local stability results for seMPC. Finally Section 3.4 demonstrates the seMPC strategy in both linear and nonlinear diesel air path simulations. Section 3.5 contains concluding remarks on gain scheduling explicit MPC.

3.2 Switched Gain Scheduled Explicit MPC

This section begins by briefly describing rate-based explicit MPC [5,114] and, specifically, how extra input parameters can be added to the explicit MPC formulation to handle time-varying constraints. This section then develops the switched explicit MPC (seMPC) strategy.

3.2.1 Rate-Based Explicit MPC

In the following, a representative linear prediction model is considered for the nonlinear plant, assuming that the system is square,

$$x_{k+1} = Ax_k + Bu_k, \tag{3.1}$$

$$y_k = Cx_k + Du_k, \tag{3.2}$$

with box type control constraints,

$$\underline{u}_k \leq u_k \leq \overline{u}_k, \tag{3.3}$$

where $\underline{u}_k$ and $\overline{u}_k$ designate the lower and upper limits, respectively, which will be assumed to be constant over the prediction horizon.

Following the steps outlined in Chapter 2, a rate-based MPC optimization problem can be defined for the box control constrained system (3.1)-(3.3) that has the following form,

$$
\begin{aligned}
\min_{\Delta u_{i|k},\, i\in\{0,\dots N-1\}} \quad & \xi_{N|k}^T P \xi_{N|k} + \sum_{i=0}^{N-1} \xi_{i|k}^T Q \xi_{i|k} + \Delta u_{i|k}^T R \Delta u_{i|k}, \\
\text{subject to}: \quad & \xi_{i+1|k} = \bar{A}\xi_{i|k} + \bar{B}\Delta u_{i|k}, \\
& \tilde{u}_{i|k} = \sum_{j=0}^{i} \Delta u_{j|k}, \\
& \underline{u}_k - u_{k-1} \leq \tilde{u}_{i|k} \leq \overline{u}_k - u_{k-1}, \\
& \xi_{0|k} = \xi_k.
\end{aligned} \tag{3.4}
$$

Once the optimization problem (3.4) is solved at the current time step, the solution for $\Delta u_{0|k}$, denoted by $\Delta u_{0|k}^\star$, is applied to the plant.

Explicit MPC can be used to represent the solution to the optimization problem (3.4) with specified matrices, $\bar{A}, \bar{B}, P, Q$, and R, that is computed off-line for all possible states, ξ_k, and takes the form of a Piecewise-Affine (PWA) function [5],

$$
\Delta u_{0|k}^\star = K_j \xi_k + F_j \quad \text{if } H_j \xi_k \leq G_j \quad \text{for } j \in \{1, ..., N_r\}, \tag{3.5}
$$

where ξ_k is the PWA function input, $\Delta u_{0|k}^\star$ is the output, and j denotes the j-th polyhedral region of the PWA function. Then the control increment is integrated to generate the absolute control command, $u_k = u_{k-1} + \Delta u_{0|k}^\star$. Explicit MPC has been shown to be computationally faster compared to on-line Quadratic Programming (QP) solvers for small QP problems, i.e., small number of optimization variables and constraints [1, 7]. The PWA solution (3.5) to the MPC optimization problem (3.4) can be computed using Hybrid Toolbox [6] or Multi-Parametric Toolbox [63]. Note that u_{k-1} is an input parameter to the optimization problem (3.4) but is not treated as an input to (3.5). This will be handled in the following discussion.

Commonly, the state space model $(\bar{A}, \bar{B})$ is extended with additional states representing the control constraint values to accommodate nonlinear or time varying control constraints. This has been done previously in [32, 38, 39] with explicit MPC, where constraint values are nonlinear functions of the current state measurement or estimate and must be

recomputed at each time step and treated as constant over the prediction horizon. Let ζ_k be the extended state vector containing the original states and the control constraints, $\zeta_k = \left[\xi_k^T,\ \overline{u}_k^T - u_{k-1}^T,\ \underline{u}_k^T - u_{k-1}^T\right]^T$. The explicit MPC controller is then formed for the augmented system,

$$\zeta_{k+1} = \mathcal{A}\zeta_k + \mathcal{B}\Delta u_k, \tag{3.6}$$

$$\mathcal{A} = \begin{bmatrix} \bar{A} & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix},\ \mathcal{B} = \begin{bmatrix} \bar{B} \\ 0 \\ 0 \end{bmatrix}, \tag{3.7}$$

$$\xi_k = \mathcal{C}_\xi \zeta_k,\ \underline{u}_k - u_{k-1} = \mathcal{C}_{\underline{u}}\zeta_k,\ \overline{u}_k - u_{k-1} = \mathcal{C}_{\overline{u}}\zeta_k, \tag{3.8}$$

where the control bounds $\underline{u}_k - u_{k-1}$ and $\overline{u}_k - u_{k-1}$ are incorporated as non-dynamic states in (3.6)-(3.7) and the matrices $\mathcal{C}_\xi, \mathcal{C}_{\underline{u}}$, and $\mathcal{C}_{\overline{u}}$ in (3.8) extract the appropriate elements of ζ_k. Using the augmented system (3.6)-(3.8), the following MPC optimization problem is defined,

$$\begin{aligned} \min_{\Delta u_{i|k},\, i\in\{0,\ldots,N-1\}} \quad & \zeta_{N|k}^T \mathcal{C}_\xi^T P \mathcal{C}_\xi \zeta_{N|k} + \sum_{i=0}^{N-1} \zeta_{i|k}^T \mathcal{C}_\xi^T Q \mathcal{C}_\xi \zeta_{i|k} + \Delta u_{i|k}^T R \Delta u_{i|k}, \\ \text{subject to:} \quad & \zeta_{i+1|k} = \mathcal{A}\zeta_{i|k} + \mathcal{B}\Delta u_{i|k}, \\ & \tilde{u}_{i|k} = \sum_{j=0}^{i} \Delta u_{j|k}, \\ & \mathcal{C}_{\underline{u}}\zeta_k \le \tilde{u}_{i|k} \le \mathcal{C}_{\overline{u}}\zeta_k, \\ & \xi_{0|k} = \xi_k. \end{aligned} \tag{3.9}$$

The explicit MPC can then be computed where the control constraint values can be reconfigured through ζ_k at each time step,

$$\Delta u_{0|k}^\star = \mathcal{K}_j \zeta_k + \mathcal{F}_j \quad \text{if } \mathcal{H}_j \zeta_k \le \mathcal{G}_j \quad \text{for } j \in \{1, ..., N_r\}. \tag{3.10}$$

3.2.2 Switched Explicit MPC

Let $P_0(z)$ represent the plant at the nominal operating point and $P_\theta(z)$ represent the

plant at off-nominal operating points where θ is the scheduling parameter, e.g., engine speed and fuel rate in the diesel air path case. The goal of the switched explicit MPC (seMPC) strategy is to match off-nominal plants, $P_\theta(z)$ in Figure 3.2, to the nominal plant, $P_0(z)$ in Figure 3.1, by introducing a scheduled gain, S_θ, between the explicit MPC and plant. The gain S_θ is chosen such that the combination of the scheduled gain and the off-nominal plant, $S_\theta P_\theta(z)$, is made to approximate the nominal plant, $P_0(z)$, i.e., $S_\theta P_\theta(z) \approx P_0(z)$. In the simulation results that will be presented in Section 3.4, S_θ is chosen to match the DC gain of the nominal plant, $S_\theta = P_0(1)P_\theta^{-1}(1)$. This type of scheduling based on DC gain has previously been used for decentralized PI control of a EGR-VGT diesel engine [108,109]. The scheduled gain, S_θ, can also be chosen to minimize a closed loop transfer function difference as done in [28] or could be adapted in real-time.

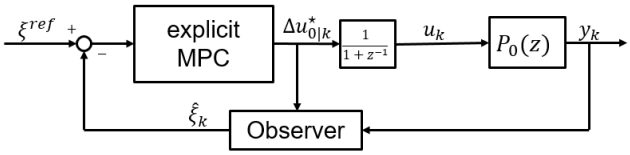

Figure 3.1: Nominal plant, $P_0(z)$, and control.

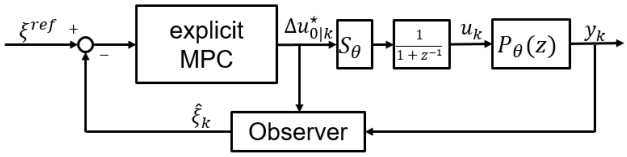

Figure 3.2: Off-nominal plant, $P_\theta(z)$, and control with scheduled gain S_θ.

When the scheduled gain, S_θ, is applied to explicit MPC, special considerations must be made to ensure control constraint enforcement. Explicit MPC will guarantee that the control, $u_k = \Delta u^\star_{0|k} + u_{k-1}$, satisfies control constraints. However, after the scheduled gain is applied, there is no guarantee that the resulting control signal, $u_k = S_\theta \Delta u^\star_{0|k} + u_{k-1}$, satisfies control constraints. Figure 3.3 illustrates how the control constraints are violated when the scheduled gain is directly applied to a 2 input system.

Consider the control constraints for a 2 input system, where $\tilde{u}_{1,i|k}$ and $\tilde{u}_{2,i|k}$ are the first and second inputs at the i-th time step in the horizon at sample time k, and $u_{1,k-1}$ and

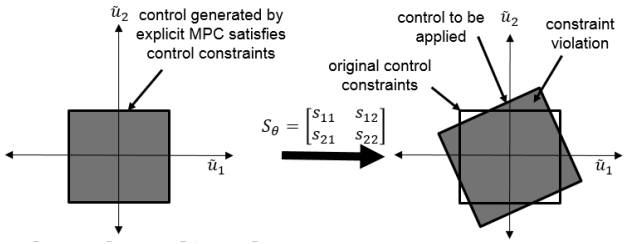

Figure 3.3: Direct application of the scheduled gain leads to constraint violation.

$u_{2,k-1}$ are the controls applied at the previous time step,

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} \tilde{u}_{1,i|k} \\ \tilde{u}_{2,i|k} \end{bmatrix} \leq \begin{bmatrix} \overline{u}_1 - u_{1,k-1} \\ \overline{u}_2 - u_{2,k-1} \\ -\underline{u}_1 + u_{1,k-1} \\ -\underline{u}_2 + u_{2,k-1} \end{bmatrix}. \tag{3.11}$$

In order to enforce the control constraints after the scheduled gain, the constraint (3.11) that must be enforced by MPC should become,

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \end{bmatrix} S_\theta \begin{bmatrix} \tilde{u}_{1,i|k} \\ \tilde{u}_{2,i|k} \end{bmatrix} \leq \begin{bmatrix} \overline{u}_1 - u_{1,k-1} \\ \overline{u}_2 - u_{2,k-1} \\ -\underline{u}_1 + u_{1,k-1} \\ -\underline{u}_2 + u_{2,k-1} \end{bmatrix}. \tag{3.12}$$

However, once the explicit MPC is computed for the nominal plant with nominal control constraint (3.11), only the right hand side of the control constraint inequality (3.11) can be accessed online (through the extended state vector, ζ_k). Thus the constraint (3.12) cannot be implemented unless S_θ is removed from the left hand side of (3.12). For a 2 input system, there are two cases of S_θ where it can be removed from the left hand side of (3.12). The first is if S_θ is diagonal which is illustrated in Figure 3.4.

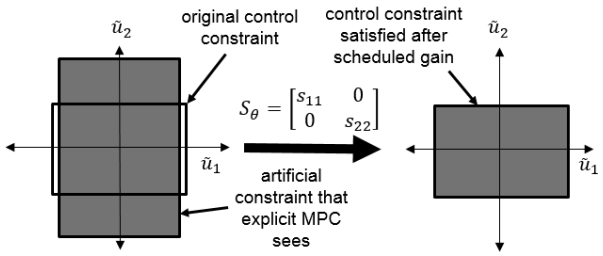

Figure 3.4: Diagonal only scheduled gain maintains constraint satisfaction.

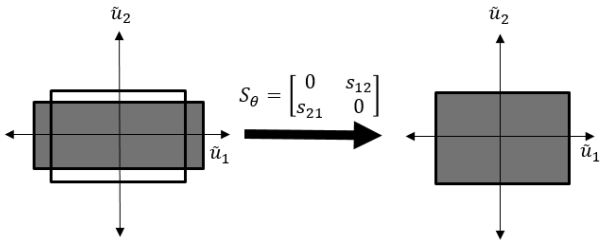

Figure 3.5: Anti-diagonal only scheduled gain maintains constraint satisfaction.

Let $S_\theta = diag(s_{11}, s_{22})$ and denote this as S_θ^d. With S_θ^d, the constraint (3.12) becomes,

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \end{bmatrix} S_\theta^d \begin{bmatrix} \tilde{u}_{1,i|k} \\ \tilde{u}_{2,i|k} \end{bmatrix} \leq \begin{bmatrix} \overline{u}_1 - u_{1,k-1} \\ \overline{u}_2 - u_{2,k-1} \\ -\underline{u}_1 + u_{1,k-1} \\ -\underline{u}_2 + u_{2,k-1} \end{bmatrix}, \tag{3.13}$$

which, assuming $s_{11}, s_{22} > 0$, can be rewritten as,

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} \tilde{u}_{1,i|k} \\ \tilde{u}_{2,i|k} \end{bmatrix} \leq \begin{bmatrix} s_{11}^{-1}(\overline{u}_1 - u_{1,k-1}) \\ s_{22}^{-1}(\overline{u}_2 - u_{2,k-1}) \\ -s_{11}^{-1}(\underline{u}_1 + u_{1,k-1}) \\ -s_{22}^{-1}(\underline{u}_2 + u_{2,k-1}) \end{bmatrix}, \tag{3.14}$$

where the extended state vector ζ_k should be set with the values of the right hand side of the inequality (3.14) rather than the original control constraint values in (3.11). Note that the inequality (3.14) is formed assuming that $s_{11}, s_{22} > 0$. If either s_{11} or s_{22} are negative,

then the corresponding minimum and maximum constraints should be switched, e.g.,

$$\begin{aligned}\begin{bmatrix} 1 \\ -1 \end{bmatrix} \tilde{u}_{1,i|k} &\leq \begin{bmatrix} s_{11}^{-1}(\overline{u}_1 - u_{1,k-1}) \\ -s_{11}^{-1}(\underline{u}_1 + u_{1,k-1}) \end{bmatrix} \quad if\ s_{11} > 0, \\ \begin{bmatrix} 1 \\ -1 \end{bmatrix} \tilde{u}_{1,i|k} &\leq \begin{bmatrix} s_{11}^{-1}(\underline{u}_1 - u_{1,k-1}) \\ -s_{11}^{-1}(\overline{u}_1 + u_{1,k-1}) \end{bmatrix} \quad if\ s_{11} < 0. \end{aligned} \tag{3.15}$$

If either s_{11} or s_{22} are zero, then the corresponding constraints can be set to infinity or an appropriately large value.

The second case where S_θ can be removed from the left hand side of (3.12) is if it is an anti-diagonal only matrix,

$$S_\theta = \begin{bmatrix} 0 & s_{12} \\ s_{21} & 0 \end{bmatrix}. \tag{3.16}$$

We denote the anti-diagonal scheduled gain (3.16) as S_θ^o. With S_θ^o, the constraint (3.12) in the anti-diagonal case, assuming $s_{12}, s_{21} > 0$, becomes,

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} \tilde{u}_{1,i|k} \\ \tilde{u}_{2,i|k} \end{bmatrix} \leq \begin{bmatrix} s_{12}^{-1}(\overline{u}_1 - u_{1,k-1}) \\ s_{12}^{-1}(\overline{u}_2 - u_{2,k-1}) \\ -s_{21}^{-1}(\underline{u}_1 + u_{1,k-1}) \\ -s_{21}^{-1}(\underline{u}_2 + u_{2,k-1}) \end{bmatrix}. \tag{3.17}$$

If either s_{12} or s_{21} are negative or zero, the change of sign or singularity needs to be accounted for similar to the diagonal case.

Let the desired scheduling gain be $S_\theta^\star$, which can be a fully populated matrix. This can be computed, for example, by matching the DC gain of the off-nominal plant to the nominal plant or by minimizing a closed-loop transfer function difference. The seMPC strategy aims to recover $S_\theta^\star$, particularly as the sampling frequency increases, by switching between the

diagonal scheduled gain S_θ^d and anti-diagonal scheduled gain S_θ^o every "half" time step, where

$$S_\theta^d + S_\theta^o = S_\theta^\star. \tag{3.18}$$

To implement this strategy, the discrete time nominal prediction model (3.1) should have the same sampling period, ΔT, as if no scheduling strategy is used. The explicit MPC should be evaluated every half time step, i.e., every $\Delta T/2$, with S_θ^d and constraints (3.14) on the even half-step, and with S_θ^o and constraints (3.17) on the odd half-step. Measurements or state estimates should only occur on the even half-step so that the states used in ζ_k on the odd half-step are the same as was used on the previous even half-step.

This strategy can be extended to systems with more inputs. For example, in the case of 3 inputs, the desired scheduled gain could be split into three,

$$S_\theta^1 = \begin{bmatrix} s_{11} & 0 & 0 \\ 0 & s_{22} & 0 \\ 0 & 0 & s_{33} \end{bmatrix}, S_\theta^2 = \begin{bmatrix} 0 & 0 & s_{13} \\ s_{21} & 0 & 0 \\ 0 & s_{32} & 0 \end{bmatrix}, S_\theta^3 = \begin{bmatrix} 0 & s_{12} & 0 \\ 0 & 0 & s_{23} \\ s_{31} & 0 & 0 \end{bmatrix}. \tag{3.19}$$

However, the explicit MPC needs to be evaluated an extra time per additional input per sampling period.

Note that the observer and constraints on tracked outputs may not require special treatment in the proposed switching strategy. Typically, output constraints are treated as soft to guarantee feasibility in the presence of plant/model mismatch and disturbances. Furthermore, it is reasonable to expect that estimation error and violations of constraints on tracked outputs will be smaller with seMPC than without any scheduling strategy because the off-nominal condition has been made to look like the nominal condition/prediction model. The handling of general state constraints requires further analysis because the system may not be square from inputs to constrained states. This analysis will be pursued in future work.

3.3 Analysis of Switched Explicit MPC

In this section, conditions for stability of seMPC are developed. The subscript θ has been dropped to simplify notation. Let "target system" refer to the closed-loop system where $S^\star$ is used and Δu_k is the result of a MPC optimization problem (MPC1) with the constraint (3.12) directly applied,

$$\begin{aligned} \xi_{k+1} &= \bar{A}\xi_k + \bar{B}S^\star \Delta u_k, \\ \xi_{k+2} &= \bar{A}^2\xi_k + \bar{A}\bar{B}S^\star \Delta u_k. \end{aligned} \tag{3.20}$$

Note the control, Δu_k in (3.20) is computed and applied at the "even" half-step, k, and is zero during the "odd" half-step, $k+1$. The system dynamic matrices, $\bar{A}$ and $\bar{B}$ in (3.20), correspond to the half-step sampling period, $\Delta T/2$.

The state update equations for the seMPC system with states, $\bar{\xi}_k$, can be written as

$$\begin{aligned} \bar{\xi}_{k+1} &= \bar{A}\bar{\xi}_k + \bar{B}S^d \Delta u_k^d, \\ \bar{\xi}_{k+2} &= \bar{A}^2\bar{\xi}_k + \bar{A}\bar{B}S^d \Delta u_k^d + \bar{B}S^o \Delta u_k^o, \end{aligned} \tag{3.21}$$

where Δu_k^d and Δu_k^o are the result of MPC optimization problems (MPC2 and MPC3) formulated with constraints (3.14) and (3.17), respectively. Note that locally, i.e., if all constraints are inactive, Δu_k, Δu_k^d, and Δu_k^o represent the same control policy with the same linear gain K because the cost function is the same for all three MPC controllers. The local closed loop state equations are

$$\xi_{k+2} = (\bar{A}^2 + \bar{A}\bar{B}S^\star K)\xi_k, \tag{3.22}$$

$$\bar{\xi}_{k+2} = \left(\bar{A}^2 + \left(\bar{A}\bar{B}S^d + \bar{B}S^o\right)K\right)\bar{\xi}_k. \tag{3.23}$$

The local error between the target system and the switched strategy system satisfies

$$\begin{aligned} e_k &= \bar{\xi}_k - \xi_k, \\ e_{k+2} &= \left(\bar{A}^2 + \left(\bar{A}\bar{B}S^d + \bar{B}S^o\right)K\right)e_k + \left(\bar{A}\bar{B}S^d + \bar{B}S^o - \bar{A}\bar{B}S^\star\right)K\xi_k. \end{aligned} \tag{3.24}$$

Proposition 3.1: If $\bar{A}^2 + \bar{A}\bar{B}S^\star K$ and $\bar{A}^2 + \left(\bar{A}\bar{B}S^d + \bar{B}S^o\right)K$ are asymptotically stable, the control constraint set, U, is closed and bounded and has non-empty interior, and MPC1, MPC2, and MPC3 have the same objective function, then there exists a set Ω such that if $\xi_0 \in \Omega$ and $\xi_0 = \bar{\xi}_0$, then $\lim_{k\to\infty} \xi_k = 0$ and $\lim_{k\to\infty} \bar{\xi}_k = 0$.

Proof: A maximal output admissible set, O_∞, exists and has non-empty interior for the ξ_k system with MPC1 and constraints $H_0\xi_k \le G_0$ (region zero of the explicit MPC where the unconstrained gain is obtained, $\Delta u_k = K\xi_k$). Similarly, maximal output admissible sets O_∞^d and O_∞^o exist for the $\bar{\xi}_k$ system with MPC2 and MPC3, respectively. Under our assumptions, a set Ω can be chosen to satisfy the following properties:

$$\left(\bar{A}^2 + \left(\bar{A}\bar{B}S^d + \bar{B}S^o\right)K\right)\Omega \subset \Omega, \tag{3.25}$$

$$\Omega \subset O_\infty, \tag{3.26}$$

$$\Omega \subset O_\infty^d, \tag{3.27}$$

$$\Omega \subset O_\infty^o. \tag{3.28}$$

Then $\Delta u_k = K\xi_k$ for all even k. $\Delta u_0^d = \Delta u_0^o = K\bar{\xi}_0$ and (3.25) implies $\Delta u_k^d = \Delta u_k^o = K\bar{\xi}_k$ for all even k. Consequently, $\lim_{k\to\infty} \xi_k = 0$ and $\lim_{k\to\infty} \bar{\xi}_k = 0$ because $\bar{A}^2 + \bar{A}\bar{B}S^\star K$ in (3.22) and $\bar{A}^2 + \left(\bar{A}\bar{B}S^d + \bar{B}S^o\right)K$ in (3.23) are asymptotically stable. ■

A local (unconstrained) error bound can also be established between the seMPC strategy and a continuous time target system, or "averaged" system, as the sampling period, ΔT, goes to zero. Let A_c and B_c be continuous time system matrices corresponding to the discrete time system matrices, A and B with a sampling period of ΔT. The continuous time rate-based

system [18] with seMPC is

$$\dot{\tilde{z}} = \tilde{A}\bar{z} + \tilde{B}\dot{\bar{u}} \tag{3.29}$$

$$\bar{z} = \begin{bmatrix} \ddot{\bar{x}} \\ \dot{\bar{y}} \end{bmatrix},\ \tilde{A} = \begin{bmatrix} A_c & 0 \\ C & 0 \end{bmatrix},\ \tilde{B} = \begin{bmatrix} B_c \\ D \end{bmatrix}. \tag{3.30}$$

The discrete time control update for seMPC can be expressed in continuous time as

$$\dot{\bar{u}} = \begin{cases} S^d K \begin{bmatrix} \bar{x}(t_k) - \bar{x}(t_{k-1}) \\ \bar{y}(t_{k-1}) \end{bmatrix} \delta(t_k), & t_k \le t < t_k + \frac{\Delta T}{2} \\ S^o K \begin{bmatrix} \bar{x}(t_k) - \bar{x}(t_{k-1}) \\ \bar{y}(t_{k-1}) \end{bmatrix} \delta(t_k + \frac{\Delta T}{2}), & t_k + \frac{\Delta T}{2} \le t < t_{k+1} \end{cases}, \tag{3.31}$$

where $\delta(t)$ is the dirac-delta function applied at time t.

The continuous time averaged system is

$$\dot{z} = \tilde{A}z + \tilde{B}\dot{u}, \tag{3.32}$$

where the control, $\dot{u} = S^\star K_c z$, is applied continuously rather than sampled. The discrete time update for z is

$$z(t_{k+1}) = e^{\Gamma \Delta T} z(t_k),\ \Gamma = \tilde{A} + \tilde{B} S^\star K_c. \tag{3.33}$$

Using the Taylor series expansion of the matrix exponential, (3.33) can be expressed as

$$z(t_{k+1}) = (I + \Gamma \Delta T + O_1(\Delta T^2)) z(t_k), \tag{3.34}$$

where $O_i(\Delta T^m)$ denotes terms of order ΔT^m and higher and i denotes an unique function O_i. Let K and K_c have the form

$$\begin{aligned} K &= [K_P\ K_I] = [K_{P,c} + O_2(\Delta T) \quad K_{I,c}\Delta T + O_3(\Delta T^2)], \\ K_c &= [K_{P,c}\ K_{I,c}], \end{aligned} \tag{3.35}$$

where K_P and K_I denote proportional and integral terms respectively. It is reasonable to assume that $K_P \to K_{P,c}$ as $\Delta T \to 0$ and that K_I is $O(\Delta T)$. If, instead, K_I is $O(1)$, then $u(t) \to \infty$ as $\Delta T \to 0$.

Returning to the seMPC system (3.29)-(3.30), the discrete time update is

$$\bar{z}(t_{k+1}) = e^{\tilde{A}\Delta T}\bar{z}(t_k) + e^{\tilde{A}\Delta T}\tilde{B}S^dK \begin{bmatrix} \bar{x}(t_k) - \bar{x}(t_{k-1}) \\ \bar{y}(t_{k-1}) \end{bmatrix} + e^{\tilde{A}\frac{\Delta T}{2}}\tilde{B}S^oK \begin{bmatrix} \bar{x}(t_k) - \bar{x}(t_{k-1}) \\ \bar{y}(t_{k-1}) \end{bmatrix}. \tag{3.36}$$

Using Euler integration,

$$\begin{bmatrix} \bar{x}(t_k) - \bar{x}(t_{k-1}) \\ \bar{y}(t_{k-1}) \end{bmatrix} = \begin{bmatrix} \Delta T\dot{\bar{x}}(t_k) + O_{4,k}(\Delta T^2) \\ \bar{y}(t_k) + O_{5,k}(\Delta T) \end{bmatrix}, \tag{3.37}$$

where $O_{i,k}(\Delta T^m)$ are the Euler integration truncation errors from time step t_k to t_{k-1}. Using (3.18), (3.35) and (3.37), the discrete time update for seMPC (3.36) can be written as

$$\bar{z}(t_{k+1}) = (I + \Gamma\Delta T + O_6(\Delta T^2))\bar{z}(t_k) + O_{7,k}(\Delta T^2), \tag{3.38}$$

where $O_{7,k}(\Delta T^2)$ in (3.38) is a function of $O_{4,k}(\Delta T^2)$ and $O_{5,k}(\Delta T)$ in (3.37).

The error system, $\varepsilon = \bar{z} - z$, can then be formed,

$$\varepsilon(t_{k+1}) = (I + \Gamma\Delta T + O_6(\Delta T^2))\varepsilon(t_k) + O_8(T^2)z(t_k) + O_{7,k}(\Delta T^2). \tag{3.39}$$

Proposition 3.2: Let $H(\Delta T) = I + \Gamma\Delta T + O_1(\Delta T^2)$ and $\gamma \neq 0$ such that $||H(\Delta T)|| \leq q(\Delta T) = 1 + \gamma\Delta T + O_8(\Delta T^2)$. Then $\lim_{\Delta T \to 0^+} |\varepsilon(t)| \leq e^{\gamma t}|\varepsilon(0)|$.

Proof: Let $n(t) = \lfloor t/\Delta T \rfloor$, $t_n = n(t)\Delta T$. The error system (3.39) can be bounded by $|\varepsilon(t_{k+1})| \leq q|\varepsilon(t_k)| + r$, where r is $O(\Delta T^2)$ and

$$r = ||O_8(\Delta T^2)|| \left(\max_{0 \leq \tau \leq t} ||e^{\Gamma\tau}|| \right) |z(0)| + \max_{0 \leq k \leq n} |O_{7,k}(\Delta T^2)|. \tag{3.40}$$

Then

$$|\varepsilon(t_n)| \leq q^n|\varepsilon(0)| + \frac{q^n - 1}{q - 1}r. \tag{3.41}$$

Note that $\lim_{\Delta T \to 0^+} q^n = e^{\gamma t}$, $\lim_{\Delta T \to 0^+} r/(q-1) = 0$, and $\lim_{\Delta T \to 0^+} t_n = t$. Thus $\lim_{\Delta T \to 0^+} |\varepsilon(t)| = \lim_{\Delta T \to 0^+} |\varepsilon(t_n)| \leq e^{\gamma t}|\varepsilon(0)|$. ■

Remark 1: Proposition 3.2 does not require stability and furthermore, if $\varepsilon(0) = 0$, then $\lim_{\Delta T \to 0^+} \varepsilon(t) = 0$.

Remark 2: Suppose the assumptions of Proposition 3.2 hold with $\gamma < 0$ for a certain induced norm $||\cdot||$ corresponding to a vector norm $|\cdot|$. This is a reasonable expectation if Γ is Hurwitz. Then for any $\varepsilon(0)$, $\lim_{t \to \infty} \lim_{\Delta T \to 0^+} \varepsilon(t) = 0$.

3.4 Diesel Air Path Simulation Results

The diesel air path can have different dynamics depending on the operating condition, e.g., low engine speed vs. high engine speed. However, the choice has been made to only design a single linear MPC to cover the entire engine operating range due to stringent Engine Control Unit (ECU) memory usage constraints and, furthermore, to simplify the calibration process, see Chapter 2. The controller in Chapter 2 has been tuned conservatively to be robust to different engine dynamics.

The control objective is to track set-points for intake manifold pressure (MAP) and Exhaust Gas Recirculation (EGR) rate. The set-points are provided by maps as functions of the engine operating conditions, i.e., the current engine speed and fueling rate. The controlled inputs are the Variable Geometry Turbocharger (VGT) position and commanded EGR flow. The EGR flow command is subsequently converted to EGR valve and EGR throttle position commands through partial nonlinear inversion. The explicit MPC is designed exploiting a rate-based model.

In the simulation results presented in this section, the scheduled gain $S_\theta^\star$ is chosen to

match the off-nominal plant DC gain to the nominal plant DC gain,

$$S_\theta^\star = P_0(1)P_\theta^{-1}(1). \tag{3.42}$$

The nominal model operating condition is at 1600 rpm engine speed and 30 mm^3/st. fuel rate. Off-nominal model linearizations are taken every 400 rpm between 800 rpm and 4000 rpm and every 10 mm^3/st. between 10 mm^3/st. and 50 mm^3/st. to populate the $S_\theta^\star$ elements stored in a linear interpolation based look-up table. The base sampling rate before the switched gain scheduling strategy is applied is $\Delta T = 32$ msec.

The rate-based MPC cost functional utilizes a control and prediction horizon of 1,

$$J = \left(\bar{A}\xi_0 + \bar{B}\Delta u_{0|k}\right)^T P \left(\bar{A}\xi_0 + \bar{B}\Delta u_{0|k}\right) + \Delta u_{0|k}^T R \Delta u_{0|k}. \tag{3.43}$$

There are control constraints on both the VGT position and EGR flow,

$$\underline{u} \leq u_k \leq \overline{u}. \tag{3.44}$$

Figure 3.6 shows a linear simulation of a MAP set-point step response where δ denotes the deviation from the nominal equilibrium. This figure illustrates that the seMPC strategy is able to successfully approximate a strategy where the full gain scheduling matrix is directly applied, i.e., $u_k = S_\theta^\star \Delta u_{0|k} + u_{k-1}$. Figure 3.7 shows a zoomed in view of Figure 3.6 highlighting that there is only a very small discrepancy between the full matrix gain scheduling and seMPC strategies.

Figures 3.8-3.9 show reference step responses around each operating condition of the nonlinear model without any gain scheduling strategy (left), with seMPC (middle), and with the traditional gsMPC strategy (right). In the gsMPC strategy, the tuning matrices, Q and R, are the same for each operating condition. In Figures 3.8-3.9, δ denotes the deviation from the set-point at that condition. Figure 3.8 shows that MAP responses are closer to the

nominal step response designed at 1600 rpm and 30 mm^3/st. when seMPC is used. The thicker gold line highlights the nominal MAP step response. The dotted cyan and green lines highlight the extreme conditions at 3200 rpm and 30 mm^3/st. and 800 rpm and 5 mm^3/st. With seMPC, the highly underdamped response seen at 3200 rpm and 30 mm^3/st. without seMPC disappears, and the slow response seen at 800 rpm and 5 mm^3/st. without seMPC is sped up. The response without any gain scheduling strategy at 3200 rpm and 50 mm^3/st. is not shown because it is unstable. The instability is stabilized with seMPC, and likewise gsMPC. Overall, the responses at the various conditions are drawn closer to the nominal step response. Figure 3.9 shows EGR rate reference steps at each condition. Again the highly underdamped response seen without seMPC at 3200 rpm and 30 mm^3/st. is removed with seMPC. There is some performance degradation in the 800 rpm and 5 mm^3/st. case which can be due to an inaccurate linear model at this condition or may suggest that DC gain matching is not the best choice for $S_\theta^\star$. However, overall, the responses are drawn closer to the nominal step response.

Table 3.1 shows a comparison of the cumulative absolute tracking error over the step responses shown in Figures 3.8-3.9 utilizing MPC (no scheduling), seMPC, and gsMPC, and their respective ROM usage. Note that ROM usage is $o(N_z N_r N_c N_p)$, where N_z is the number of zones and $N_z = 1$ in the seMPC case and $N_z = 9$ in the gsMPC case. N_p, N_r, and N_c, see Section 2.3.7, are the same for seMPC and gsMPC. Also note that the computation time of the scaled constraints (3.14) and (3.17) is marginal compared to the evaluation of the PWA control law (3.10), thus the computation time of seMPC and gsMPC are essentially the same.

As expected, both seMPC and gsMPC are able to track significantly better than MPC. Even with fewer degrees of freedom to modify the closed loop dynamics, the performance of seMPC is comparable to gsMPC. The ROM usage of seMPC is only slightly larger than MPC due to the storage of four 9×5 lookup tables (one table per element of $S^\star$). Nine explicit MPC's are used for this gsMPC implementation which is similar to previously published

Table 3.1: Comparison of tracking error and memory usage of MPC without gain scheduling, seMPC, and gsMPC.

Controller	MAP error	EGR rate error	ROM [kB]
MPC	2343	1115	73.6
seMPC	1172	716	74.3
gsMPC	1402	636	708.4

gsMPC strategies for diesel air path control, e.g., 12 explicit MPC's are used in [84]. The need to store multiple explicit MPC's dramatically increases the ROM usage for gsMPC.

Figure 3.10 shows the MAP response on a portion of the New European Drive Cycle with and without seMPC in closed loop with with nonlinear model. We observe that with seMPC, MAP is able to rise faster during the acceleration phases, e.g., from 844 sec to 849 sec and from 854 sec to 857 sec. The MAP response can be seen in Figure 3.12 to be approximately 100 msec faster during the acceleration phases with seMPC than without.

Figure 3.11 shows the EGR rate response on the same portion of the New European Drive Cycle with and without seMPC. We observe that with seMPC, the EGR rate response is significantly better able to track the EGR rate set-point at 854 sec. Figure 3.13 shows a zoomed view of the EGR rate response around 843 sec. With the seMPC design, the controller is able to recover EGR rate set-point tracking to the set-point at 843.5 sec while without seMPC the controller is not able to recover tracking.

Figure 3.14 shows the values of the elements of $S_\theta^\star$ that are used during the NEDC. These values are obtained from 2D, linear interpolation based, lookup tables with the engine speed and fueling rate as inputs. The table values are computed offline systematically by matching the DC gain of the off-nominal operating conditions to the nominal condition.

3.5 Conclusions

A gain scheduling strategy that can be used in conjunction with explicit MPC has been developed and validated in simulation for the diesel engine airpath. In a traditional strat-

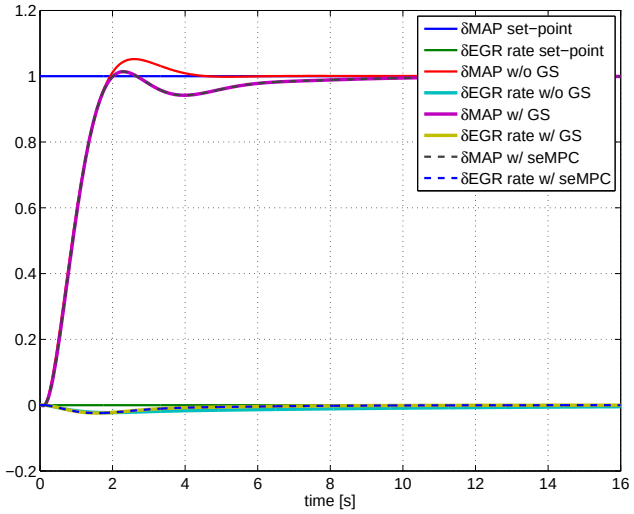

Figure 3.6: Linear step response simulation comparing MPC without gain scheduling, full gain scheduling, and seMPC.

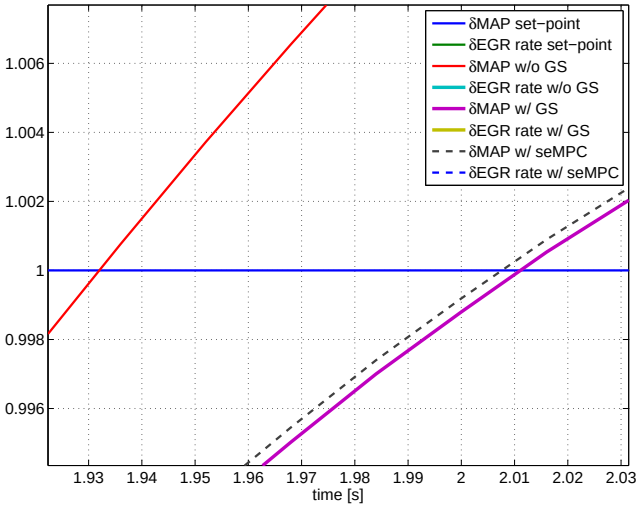

Figure 3.7: Zoomed view of linear step response.

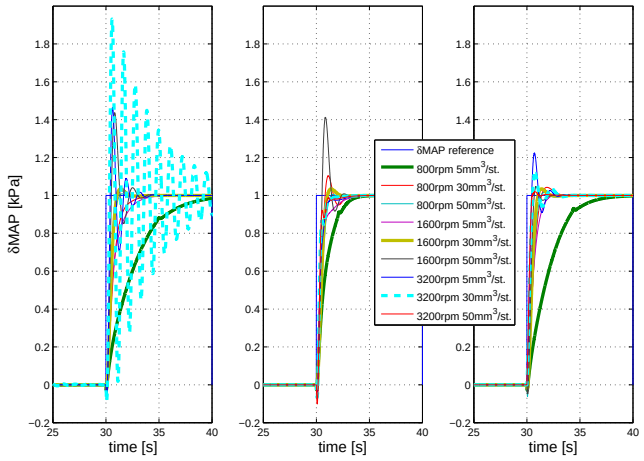

Figure 3.8: MAP set-point steps using MPC without gain scheduling (left), seMPC, (middle), and gsMPC (right) at different operating conditions of the high fidelity nonlinear model.

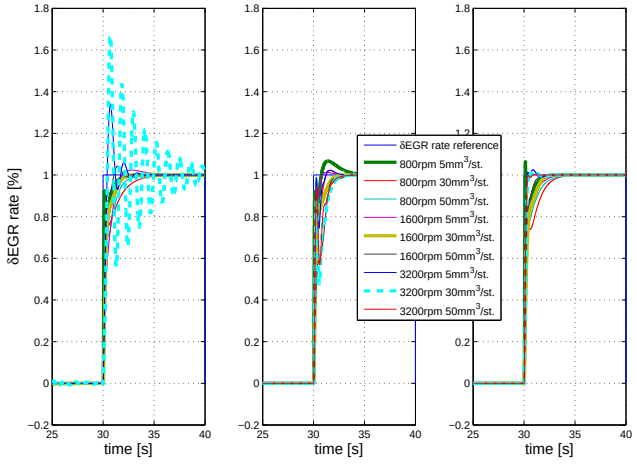

Figure 3.9: EGR rate set-point steps using MPC without gain scheduling (left), seMPC, (middle), and gsMPC (right) at different operating conditions of the high fidelity nonlinear model.

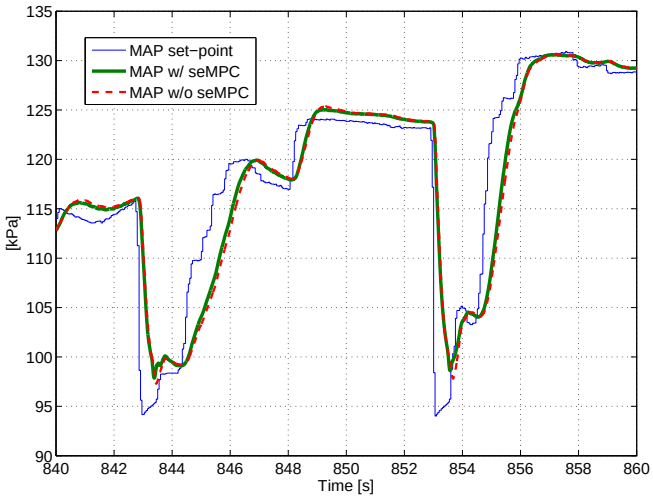

Figure 3.10: MAP response on the NEDC comparing no gain scheduling and seMPC.

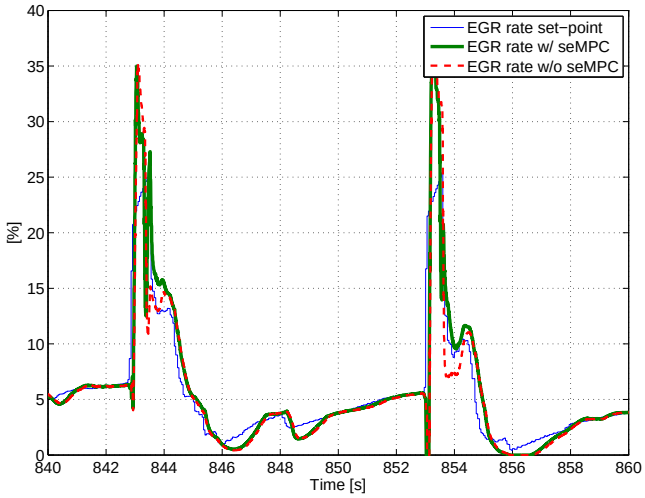

Figure 3.11: EGR rate response on the NEDC comparing no gain scheduling and seMPC.

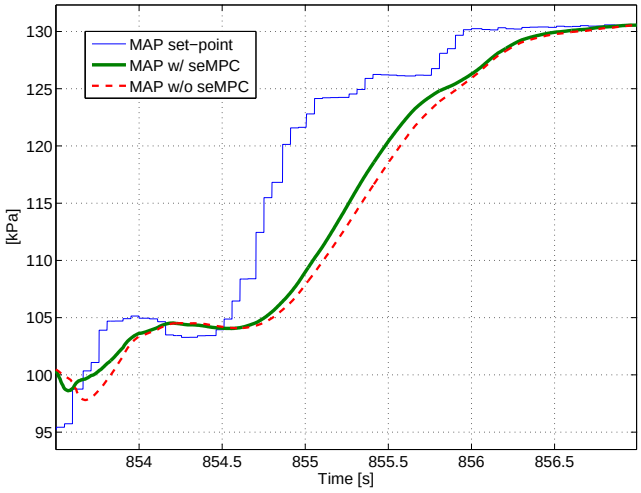

Figure 3.12: Zoomed view of MAP response on the NEDC comparing no gain scheduling and seMPC.

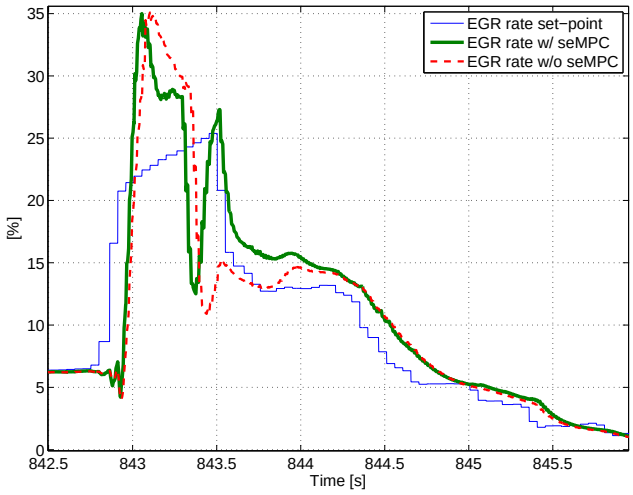

Figure 3.13: Zoomed view of EGR rate response on the NEDC comparing no gain scheduling and seMPC.

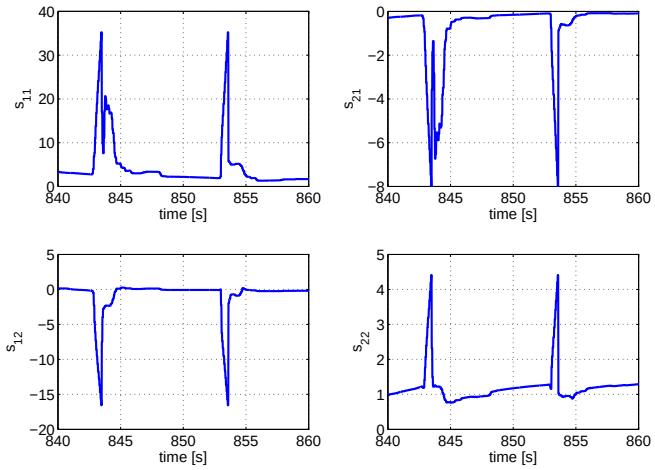

Figure 3.14: Switched gain scheduling elements on the NEDC.

egy for gain scheduled explicit MPC, multiple complex piecewise-affine functions need to be stored in memory and bumpless transfer across possibly discontinuous controller zone boundaries needs to be handled. In the novel scheduling strategy proposed in this chapter, simple lookup tables are used to store all of the scheduling parameters. The calibration of the scheduling parameters is performed in a systematic manner whereas in a traditional strategy, one has to individually tune the MPC cost function at each operating condition. A switching implementation has been proposed to satisfy original control constraints with the gain scheduled MPC. From a broader perspective, the scheduling strategy proposed in this chapter gives a degree of flexibility (reconfigurability to model changes) to explicit MPC which is traditionally rigid. In future work, the handling of general state constraints will be investigated. The limitations of the seMPC strategy, which offers fewer degrees of freedom to modify the closed-loop dynamics versus gsMPC, will also be explored.

Chapter 4

Robust Linear Model Predictive Control

This chapter describes an extension of model predictive control (MPC) for the diesel air path that is able to robustly enforce constraints in the presence of disturbances, commonly known as tube MPC. A rate-based tube MPC formulation is used to accomplish offset-free steady-state tracking. As a consequence, it is found that a rate-based formulation reduces the conservativeness of tube MPC, i.e., the amount of constraint tightening that is typically required with tube MPC is reduced. Approximations are then made to the rate-based tube MPC strategy to achieve a design that is viable for a diesel engine air path control application which has very limited computational resources. Simulation and experimental results are then presented using the approximate rate-based tube MPC strategy. A low-complexity tube MPC strategy is also developed that achieves the same maximal output admissible set as tube MPC, which is motivated by the need to formally reduce computational complexity, rather than in an approximate manner.

4.1 Introduction

The goal of this chapter is to develop a robust model predictive control (MPC) strategy that can can be used for diesel engine air path (DAP) control, [27,33,38,39,49,56,84,101]. In addition, a strategy is desired that does not significantly increase computational complexity compared to non-robust MPC. The first goal is achieved through the combination of rate-

based MPC (also referred to as velocity-form MPC), [8, 18, 39, 43, 49, 85, 114], and tube MPC, [9, 64, 74, 75]. The second goal is achieved through a novel strategy of splitting the standard tube MPC optimization problem into two smaller sub-problems while maintaining the same maximal output admissible set (MOAS), [60], as when a standard tube MPC controller is used.

Tube MPC is a variant of MPC that is able to handle state and output constraints in systems with bounded additive state disturbances. The strategy is to first consider a nominal controller with state feedback gain, K. Under this nominal controller, a robust positively invariant (RPI) set can be constructed for the closed loop system which is used to form the "tube." The MPC optimization problem is then solved to compute a center for the tube which will contain the true state. The center of the tube is deemed the nominal initial state from which prediction begins. The state constraints are tightened such that all possible trajectories inside the tube satisfy the desired constraints. Furthermore, the control constraints are tightened to account for the nominal controller gain that is applied on top of the MPC solution.

In many applications, it is desirable to incorporate integral type action into controllers to compensate for the mismatch between the model and true plant and achieve disturbance rejection. With MPC, three common strategies exist to achieve integral type action, hereafter referred to zero-offset steady-state tracking. The first approach is to directly incorporate an integrator on the tracking error. The second approach is to augment the nominal model with a disturbance model and estimate the disturbance online [69, 70, 86]. The third approach is to use a rate-based form [18, 85, 114].

As noted in [76], when a disturbance state is augmented to the original system, the resulting augmented system is not stabilizable. This renders the computation of an RPI set for the augmented system with disturbance states impossible. This is not the case for the augmented system when a rate-based form is used. Furthermore, achieving stability of tube MPC typically relies on a stabilizing terminal set constraint, [64, 72, 75]. The use of such a

terminal set constraint is difficult for the case of tracking MPC when integrators are directly added or disturbance estimators are used because the steady-state states corresponding to the references must be known a priori. Again, this is not the case with rate-based MPC. As a result of the rate-based tube MPC strategy described in this chapter, which builds from the rate-based tube MPC work of [9] and utilizes observations from [42], the conservativeness of the robust MPC controller is reduced because the effective disturbances are typically smaller with rate-based tube MPC than with conventional tube MPC.

The rate-based tube MPC strategy is applied to the diesel air path (DAP) control problem. See Figure 1.3 for a schematic of the diesel engine. One motivation for the application of MPC to diesel engine air path control is the ability of MPC to explicitly enforce constraints. For example, constraints which manifest from drivability requirements, e.g., a constraint on boost pressure overshoot to limit undesired engine torque fluctuations [15], must be considered. Furthermore, these constraints must be enforced in a robust manner when disturbances are present. The disadvantage of tube MPC is a potentially dramatic increase in computation time associated with the addition of extra constraints associated with the tube. In order to implement rate-based tube MPC, the computational complexity of the controller must be considered due to the limited computational resources available in engine control units (ECU). To manage the increase in computational complexity of tube MPC compared to standard MPC, approximations to the tube MPC strategy are made. Simulation of the controller in closed-loop with a nonlinear, physics based, DAP model are presented along with experimental results using the approximate rate-based tube MPC strategy.

While the approximate tube MPC strategy is effective in practice, the constraint satisfaction guarantees of tube MPC are lost. Thus, a low-complexity tube MPC strategy is also developed in this work which maintains the same MOAS as standard tube MPC. This is achieved through splitting the tube MPC optimization problem into more easily solvable sub-problems. The resulting computational complexity, in both computation time and memory usage, is on the same order as standard MPC.

The remainder of this chapter is organized as follows. Section 4.2 describes rate-based tube MPC. Section 4.3 describes the application to the DAP example and the associated approximations made to achieve an implementable design with low computational complexity. Section 4.4 gives a novel reduced complexity formulation of tube MPC. Finally, Section 4.5 contains concluding remarks on tube MPC.

Nomenclature: In the following, given two sets $\mathbb{U} \subset \mathbb{R}^n$ and $\mathbb{V} \subset \mathbb{R}^n$, the Minkowski set addition is defined by $\mathbb{U} \oplus \mathbb{V} = \{u + v | u \in \mathbb{U}, v \in \mathbb{V}\}$ and the Minkowski (Pontryagin) set difference is defined by $\mathbb{U} \ominus \mathbb{V} = \{x | \{x\} \oplus \mathbb{V} \subseteq \mathbb{U}\}$. Let $\mathbb{Z}^+ = \{1, 2, 3, ...\}$ be the set of strictly positive integers. The distance of a point $z \in \mathbb{R}^n$ to a set $Z \subseteq \mathbb{R}^n$ is denoted by $d(z, Z) = \inf_{x \in Z} ||x - z||$.

Definition 4.1: A set $\Omega \subset \mathbb{R}^n$ is robust positively invariant (RPI) for the system $x_{k+1} = f(x, w)$ and the constraint set $(\mathbb{X}, \mathbb{W})$ if $\Omega \subseteq \mathbb{X}$ and $f(x, w) \in \Omega, \forall w \in \mathbb{W}, \forall x \in \Omega$.

Definition 4.2: A set $O_\infty \subset \mathbb{R}^n$ is a maximal output admissible set (MOAS) for the closed loop system, $x_{k+1} = f_{cl}(x_k)$ and $y_k = g_{cl}(x_k)$, and output constraint $y \in \mathbb{Y}$ if for all $x_0 \in O_\infty$ and $k \in \mathbb{Z}^+$, $x_{k+1} = f_{cl}(x_k)$ and $y_k = g_{cl}(x_k) \in \mathbb{Y}$.

Definition 4.3: The origin is exponentially stable for a system, $x_{k+1} = f(x_k)$, with a region of attraction of $\mathbb{X}_N$ if there exists $M > 0$ and $\rho \in (0, 1)$ such that if $x_0 \in \mathbb{X}_N$, then $||x_k|| \leq M\rho^k ||x_0||$ for all $k \geq 0$.

Definition 4.4: A set Z is robustly exponentially stable, [74], for a system, $x_{k+1} = f(x_k, w_k)$, $w_k \in \mathbb{W}$, with a region of attraction of $\mathbb{X}_N$ if there exists $M > 0$ and $\rho \in (0, 1)$ such that any solution of $x_{k+1} = f(x_k, w_k)$ with an initial state, $x_0 \in \mathbb{X}_N$, and any admissible disturbance sequence, $w_k \in \mathbb{W} \, \forall k \geq 0$, satisfies $d(x_k, \mathbb{Z}) \leq M\rho^k d(x_0, \mathbb{Z})$.

4.2 Rate-Based Tube MPC

A rate-based formulation will be used in order to achieve zero-offset steady-state tracking

with tube MPC. The plant dynamics are assumed to have the form,

$$x_{k+1} = Ax_k + B_1u_k + B_2w_k, \tag{4.1}$$

$$y_k = Cx_k + D_1u_k + D_2w_k, \tag{4.2}$$

where it is desired to enforce state and control constraints of the form $x_k \in \mathbb{X} \subset \mathbb{R}^{n_x}$ and $u_k \in \mathbb{U} \subset \mathbb{R}^{n_u}$. Additionally, it is assumed that $y \in \mathbb{R}^{n_y}$ and $w \in \mathbb{W} \subset \mathbb{R}^{n_w}$ where $n_u = n_y$ and $\mathbb{X}$, $\mathbb{U}$, and $\mathbb{W}$ are compact, convex polytopes which contain the origin. With a rate-based formulation, control, state, and disturbance increments are defined as $\Delta u_k = u_k - u_{k-1}$, $\Delta x_k = x_k - x_{k-1}$, and $\Delta w_k = w_k - w_{k-1}$ respectively. The error to an output reference r, assumed to be constant, is defined as $e_k = y_k - r$. Let the augmented state vector be

$$\xi_k = \begin{bmatrix} \Delta x_k \\ e_{k-1} \end{bmatrix}. \tag{4.3}$$

Note that, unlike [8, 9, 85, 114], direct feedthrough is included in the plant model (4.1)-(4.2) through the terms D_1u_k and D_2w_k. For this reason e_{k-1} instead of e_k is used in ξ_k. The rate-based plant has the following form

$$\xi_{k+1} = \bar{A}\xi_k + \bar{B}_1\Delta u_k + \bar{B}_2\Delta w_k, \tag{4.4}$$

$$e_k = \bar{C}\xi_k, \tag{4.5}$$

$$\bar{A} = \begin{bmatrix} A & 0 \\ C & I \end{bmatrix}, \bar{B}_1 = \begin{bmatrix} B_1 \\ D_1 \end{bmatrix}, \bar{B}_2 = \begin{bmatrix} B_2 \\ D_2 \end{bmatrix}, \bar{C} = \begin{bmatrix} 0 & I \end{bmatrix}. \tag{4.6}$$

With tube MPC, a nominal model, i.e., without disturbances w_k, is defined with nominal

states, $\bar{x}_k$, nominal controls, $\bar{u}_k$, and nominal outputs, $\bar{y}_k$. The nominal model has the form,

$$\bar{x}_{k+1} = A\bar{x}_k + B_1\bar{u}_k, \tag{4.7}$$

$$\bar{y}_k = C\bar{x}_k + D_1\bar{u}_k, \tag{4.8}$$

and the nominal model has the associated rate-based form,

$$\bar{\xi}_{k+1} = \bar{A}\bar{\xi}_k + \bar{B}_1\Delta\bar{u}_k, \tag{4.9}$$

$$\bar{e}_k = \bar{C}\bar{\xi}_k. \tag{4.10}$$

With tube MPC, the control policy is composed of a nominal control term, to be determined by an optimization problem, and a state feedback term performed on the difference between the nominal state trajectory and true state trajectory,

$$\Delta u_k = \Delta\bar{u}_k + K\eta_k, \tag{4.11}$$

where the error, η_k, is defined as $\eta_k = \xi_k - \bar{\xi}_k$, and K is a stabilizing gain for the pair $(\bar{A}, \bar{B}_1)$ which can be chosen as the unconstrained Linear Quadratic Regulator (LQR) gain with weighting matrices $Q = Q^T \geq 0$ and $R = R^T > 0$. Let $A_K = \bar{A} + \bar{B}_1 K$The error, η_k, satisfies the difference equation,

$$\eta_{k+1} = A_K\eta_k + \bar{B}_2\Delta w_k. \tag{4.12}$$

The next step is to establish a bound on the error between the nominal state and true state, η_k, i.e., a robust positively invariant (RPI) set for the system (4.12). Towards this end, in [9], it was assumed that $w_k \in \mathbb{W}$. However, when a rate-based form is used, additional information/assumptions can be used. Specifically, as was done in [42], it will also be assumed that, in addition to magnitude, rate bounds exist, i.e., $\Delta w_k \in \mathbb{D}$, where $\mathbb{D}$ is a compact,

convex polytope containing the origin. This is advantageous because a rate bounded disturbance set can be much smaller than a magnitude bounded disturbance set in many physical systems. For example, for the DAP application in [42], the considered disturbances for the DAP system are the engine speed and fuel rate, which cannot change arbitrarily (within $\mathbb{W}$ which is "large") between sample times due to physical limitations. This leads to a set $\mathbb{D}$ that is much "smaller" than $\mathbb{W}$. With $\Delta w_k \in \mathbb{D}$, a RPI set, F_∞, can be constructed such that $\eta_k \in F_\infty \subset \mathbb{R}^{n_x+n_y}\, \forall k$, where

$$F_\infty = \bigoplus_{i=0}^{\infty} A_K^i \bar{B}_2 \mathbb{D}. \tag{4.13}$$

Typically a polytopic RPI outer approximation of F_∞ is used which can be computed in finite time through [93] assuming that A_K has eigenvalues inside the unit circle.

When disturbances are considered, the constraint sets, $x_k \in \mathbb{X}$ and $u_k \in \mathbb{U}$, for the nominal system must be tightened to ensure that that constraints are satisfied for the true system. With a rate-based design, $x_k \in \mathbb{X}$ and $u_k \in \mathbb{U}$ must be converted into constraints on ξ_k. This can be done, as shown in [9] (without direct feedthrough), through manipulating (4.2), i.e., $\Delta x_k = x_k - x_{k-1} = Ax_{k-1} + B_1 u_{k-1} + B_2 w_{k-1} - x_{k-1} = (A-I)x_{k-1} + B_1 u_{k-1} + B_2 w_{k-1}$. This leads to the connection between the rate-based states, $\begin{bmatrix} \Delta x_k^T & y_{k-1}^T \end{bmatrix}^T$, and the constrained states and controls, $\begin{bmatrix} x_{k-1}^T & u_{k-1}^T \end{bmatrix}^T$,

$$\begin{bmatrix} x_{k-1} \\ u_{k-1} \end{bmatrix} = \begin{bmatrix} A-I & B_1 \\ C & D_1 \end{bmatrix}^{-1} \begin{bmatrix} \Delta x_k \\ y_{k-1} \end{bmatrix} + \begin{bmatrix} A-I & B_1 \\ C & D_1 \end{bmatrix}^{-1} \begin{bmatrix} B_2 \\ D_2 \end{bmatrix} w_{k-1}, \tag{4.14}$$

where it is assumed that $\begin{bmatrix} A-I & B_1 \\ C & D_1 \end{bmatrix}$ is invertible. From x_{k-1}, u_{k-1}, and w_{k-1}, the

equation,

$$\begin{bmatrix} x_k \\ u_{k-1} \end{bmatrix} = \Pi \begin{bmatrix} \Delta x_k \\ y_{k-1} \end{bmatrix} + \left(\Pi \begin{bmatrix} B_2 \\ D_2 \end{bmatrix} + \begin{bmatrix} B_2 \\ 0 \end{bmatrix} \right) w_{k-1}, \tag{4.15}$$

$$\Pi = \begin{bmatrix} A & B_1 \\ 0 & I \end{bmatrix} \begin{bmatrix} A-I & B_1 \\ C & D_1 \end{bmatrix}^{-1}, \tag{4.16}$$

is obtained. Similarly, the relationship between $\begin{bmatrix} \Delta\bar{x}_k^T & \bar{y}_{k-1}^T \end{bmatrix}^T$ and $\begin{bmatrix} \bar{x}_k^T & \bar{u}_{k-1}^T \end{bmatrix}^T$ is

$$\begin{bmatrix} \bar{x}_k \\ \bar{u}_{k-1} \end{bmatrix} = \Pi \begin{bmatrix} \Delta\bar{x}_k \\ \bar{y}_{k-1} \end{bmatrix}. \tag{4.17}$$

With (4.15) and (4.17), the error between $\begin{bmatrix} x_k^T & u_{k-1}^T \end{bmatrix}^T$ and $\begin{bmatrix} \bar{x}_k^T & \bar{u}_{k-1}^T \end{bmatrix}^T$ can be established,

$$\begin{bmatrix} x_k \\ u_{k-1} \end{bmatrix} - \begin{bmatrix} \bar{x}_k \\ \bar{u}_{k-1} \end{bmatrix} = \Pi\eta_k + \left(\Pi \begin{bmatrix} B_2 \\ D_2 \end{bmatrix} + \begin{bmatrix} B_2 \\ 0 \end{bmatrix} \right) w_{k-1}, \tag{4.18}$$

and

$$\begin{bmatrix} x_k \\ u_{k-1} \end{bmatrix} - \begin{bmatrix} \bar{x}_k \\ \bar{u}_{k-1} \end{bmatrix} \in \Pi F_\infty \oplus \left(\Pi \begin{bmatrix} B_2 \\ D_2 \end{bmatrix} + \begin{bmatrix} B_2 \\ 0 \end{bmatrix} \right) \mathbb{W} \, \forall k. \tag{4.19}$$

So long as the constraint,

$$\begin{bmatrix} \bar{x}_k \\ \bar{u}_{k-1} \end{bmatrix} = \Pi \left(\bar{\xi}_k + \begin{bmatrix} 0 \\ r \end{bmatrix} \right) \in \bar{\mathbb{X}}, \tag{4.20}$$

where

$$\bar{\mathbb{X}} = (\mathbb{X} \times \mathbb{U}) \ominus \left(\Pi F_\infty \oplus \left(\Pi \begin{bmatrix} B_2 \\ D_2 \end{bmatrix} + \begin{bmatrix} B_2 \\ 0 \end{bmatrix} \right) \mathbb{W} \right), \tag{4.21}$$

is satisfied, then $\begin{bmatrix} x_k^T & u_{k-1}^T \end{bmatrix}^T \in \mathbb{X} \times \mathbb{U}$. Note that $\bar{\mathbb{X}}$, in (4.21), will generally be "larger" with $\Delta w \in \mathbb{D}$ than with only $w_k \in \mathbb{W}$ because F_∞ with $\Delta w \in \mathbb{D}$ is generally "smaller" than with only $w \in \mathbb{W}$.

A MOAS is typically used to impose a terminal state constraint in MPC for the purposes of establishing recursive feasibility and stability. This can be done as well with rate-based tube MPC. Let $\bar{K}$ be a stabilizing gain for the nominal system (4.9) and (4.10). A MOAS, O_∞, can then be computed for the asymptotically stable closed loop system

$$\bar{\xi}_{k+1} = \left(\bar{A} + \bar{B}_1 \bar{K}\right) \bar{\xi}_k \tag{4.22}$$

subject to the constraint,

$$\Pi \bar{\xi}_k \in \bar{\mathbb{X}} \ominus \left\{ \Pi \begin{bmatrix} 0 \\ r \end{bmatrix} \right\}, \tag{4.23}$$

assuming $\bar{\mathbb{X}} \ominus \left\{ \Pi \begin{bmatrix} 0 \\ r \end{bmatrix} \right\}$ contains the origin.

Assumption 4.1: The sets O_∞ and $\bar{\mathbb{X}}$ are non-empty, the assumptions required to compute O_∞ and $\bar{\mathbb{X}}$ are satisfied, $P = P^T \geq 0$, $Q = Q^T \geq 0$, and $R = R^T > 0$.

Given Assumption 4.1, the rate-based tube MPC optimization problem can now be formed as

$$\begin{aligned} \min_{\bar{\xi}_{0|k},\, \Delta\bar{u}_{i|k},\, i\in\{0,\ldots N-1\}} \quad & \bar{\xi}_{N|k}^T P \bar{\xi}_{N|k} + \sum_{i=0}^{N-1} \bar{\xi}_{i|k}^T Q \bar{\xi}_{i|k} + \Delta\bar{u}_{i|k}^T R \Delta\bar{u}_{i|k}, \\ \text{subject to}: \quad & \bar{\xi}_{i+1|k} = \bar{A}\bar{\xi}_{i|k} + \bar{B}\Delta\bar{u}_{i|k}, \\ & \Pi\left(\bar{\xi}_{i|k} + \begin{bmatrix} 0 \\ r \end{bmatrix}\right) \in \bar{\mathbb{X}}, \\ & \bar{\xi}_{N|k} \in O_\infty, \\ & \xi_k - \bar{\xi}_{0|k} \in F_\infty, \end{aligned} \tag{4.24}$$

where the true rate-based state, ξ_k, and reference, r, are inputs to (4.24). Let $\bar{\xi}^\star_{0|k}$ and $\Delta\bar{u}^\star_{|k}$

denote the optimal solution for $\bar{\xi}_{0|k}$ and $\Delta\bar{u}_{i|k}$ in (4.24). The control increment to be applied to the true system is $\Delta u_k = \Delta\bar{u}^\star_{0|k} + K\left(\xi_k - \bar{\xi}^\star_{0|k}\right)$.

Let $\Delta\bar{u}_{l:m|k}$, with $0 \leq l \leq m$, denote the sequence $\left\{\Delta\bar{u}_{l|k}, ..., \Delta\bar{u}_{m|k}\right\}$ and let $\Delta\bar{u}^\star_{l:m|k}$ denote the optimal sequence, $\left\{\Delta\bar{u}^\star_{l|k}, ..., \Delta\bar{u}^\star_{m|k}\right\}$, for (4.24) at time k. Let $\bar{\xi}^\star_{0|k}$ denote the optimal initial nominal state for (4.24) at time k.

Let $\bar{\mathbb{X}}_N$ be the set of nominal states, $\bar{\xi}_{0|k}$, for which there exists an admissible nominal control sequence for (4.24),

$$\bar{\mathbb{X}}_N = \left\{ \begin{array}{r} \bar{\xi}_0 \in \mathbb{R}^{n_x+n_y} \,|\, \exists \Delta\bar{u}_{0:N-1} : \bar{\xi}_{k+1} = \bar{A}\bar{\xi}_k + \bar{B}\Delta\bar{u}_k,\ \Pi\left(\bar{\xi}_k + \begin{bmatrix} 0 \\ r \end{bmatrix}\right) \in \bar{\mathbb{X}} \\ \forall k \in \{0, ..., N-1\},\ \bar{\xi}_N \in O_\infty \end{array} \right\}. \quad (4.25)$$

Proposition 4.1: Given Assumption 4.1 and assuming that $\xi_0 \in F_\infty \oplus \bar{\mathbb{X}}_N$, then (4.24) is feasible and $\begin{bmatrix} x_k^T & u_{k-1}^T \end{bmatrix}^T \in \mathbb{X} \times \mathbb{U}$ for all future time steps, $k \in \mathbb{Z}^+$. The set $F_\infty \oplus \bar{\mathbb{X}}_N$ is the MOAS for the plant, (4.4) and (4.5), under the nominal control defined by (4.24) and $\Delta u_k = \Delta\bar{u}^\star_{0|k} + K\left(\xi_k - \bar{\xi}^\star_{0|k}\right)$.

Proof: Because $\xi_0 \in F_\infty \oplus \bar{\mathbb{X}}_N$, there exists a $\bar{\xi}_{0|0} \in \bar{\mathbb{X}}_N$ such that (4.24) is feasible. By (4.24), $\Pi\left(\bar{\xi}_{1|0} + \begin{bmatrix} 0 \\ r \end{bmatrix}\right) \in \bar{\mathbb{X}}$ implies that $\begin{bmatrix} x_1^T & u_0^T \end{bmatrix} \in \mathbb{X} \times \mathbb{U}$. At time $k = 1$, $\bar{\xi}_{0|1} = \bar{\xi}^\star_{1|0}$ and $\Delta\bar{u}_{0:N-1|1} = \left\{\Delta\bar{u}^\star_{1:N-1|0}, \bar{K}\bar{\xi}^\star_{N|0}\right\}$ is an admissible initial nominal state and control sequence, respectively, for (4.24), where $\bar{\xi}^\star_{j|0}$ denotes the solution to (4.9) at step j given the initial condition $\bar{\xi}^\star_{0|0}$ and control sequence $\Delta\bar{u}^\star_{0:j-1|0}$. Then by induction, (4.24) is feasible and $\begin{bmatrix} x_k^T & u_{k-1}^T \end{bmatrix} \in \mathbb{X} \times \mathbb{U}\ \forall k \in \mathbb{Z}^+$ and $F_\infty \oplus \bar{\mathbb{X}}_N$ is the MOAS for the plant, (4.4) and (4.5), under the control defined by (4.24) and $\Delta u_k = \Delta\bar{u}^\star_{0|k} + K\left(\xi_k - \bar{\xi}^\star_{0|k}\right)$. ■

Assumption 4.2: The matrix $Q = Q^T > 0$ in (4.24). The matrix P in (4.24) is the solution to the Riccati equation corresponding to the system (4.9) and (4.10) and unconstrained infinite horizon cost, $\sum_{i=0}^{\infty} \bar{\xi}^T_{i|k} Q \bar{\xi}_{i|k} + \Delta\bar{u}^T_{i|k} R \Delta\bar{u}_{i|k}$ with $Q = Q^T > 0$ and $R = R^T > 0$. The matrix, $\bar{K}$ in (4.22), is the associated feedback gain, and O_∞ in (4.24) is constructed for the

system (4.22) subject to the constraint (4.23).

Theorem 4.1: Given the assumptions of Proposition 4.1 and Assumption 4.2, the set F_∞ is robustly exponentially stable (see Definition 4.4) for the true system (4.4) with a domain of attraction of $F_\infty \oplus \bar{\mathbb{X}}_N$.

Proof: Let

$$V_{j|k}\left(\bar{\xi}_{0|k}, \left\{\Delta\bar{u}_{0|k}, ..., \Delta\bar{u}_{j-1|k}\right\}\right) = \bar{\xi}_{j|k}^T P \bar{\xi}_{j|k} + \sum_{i=0}^{j-1} \bar{\xi}_{i|k}^T Q \bar{\xi}_{i|k} + \Delta\bar{u}_{i|k}^T Q \Delta\bar{u}_{i|k}, \tag{4.26}$$

where $\bar{\xi}_{i+1|k} = \bar{A}\bar{\xi}_{i|k} + \bar{B}\Delta\bar{u}_{i|k}$. Let $\bar{\xi}_{0|k}^\star$ and $\Delta\bar{u}_{0:N-1|k}^\star$ be the solution to (4.24) with a horizon length of N. Using the standard arguments of [72], the monotonicity property,

$$V_{N|k+1}\left(\bar{\xi}_{0|k+1}^\star, \Delta\bar{u}_{0:N-1|k+1}^\star\right) - V_{N|k}\left(\bar{\xi}_{0|k}^\star, \Delta\bar{u}_{0:N-1|k}^\star\right) \leq -\bar{\xi}_{0|k}^T Q \bar{\xi}_{0|k} - \Delta\bar{u}_{0|k}^T R \Delta\bar{u}_{0|k}, \tag{4.27}$$

is established. Then, using (4.27), Proposition 4.1, and the standard arguments of [74], the set F_∞ is robustly exponentially stable for the true system (4.4) with a domain of attraction of $F_\infty \oplus \bar{\mathbb{X}}_N$. ■

Remark: The state trajectories of the true system (4.4) approach the set F_∞ around the origin, which is generally smaller when $\Delta w \in \mathbb{D}$ than when it is only guaranteed that $w_k \in \mathbb{W}$.

4.3 Approximate Rate-Based Tube MPC

As an example, rate-based tube MPC is now applied to the diesel air path (DAP). A schematic of the DAP is shown in Figure 1.3, where the flows in the engine are controlled using a VGT, EGR valve, and EGR throttle. The control objective is to track set-points for intake manifold pressure, commonly referred to as manifold absolute pressure (MAP), p_{in}, and EGR rate (the ratio of EGR flow to cylinder flow), χ_{EGR}. The set-points are provided by maps (look-up tables) as functions of the engine operating condition, i.e., the current

engine speed and load (indicated by fuel rate). An output constraint must be enforced on maximum MAP (the EGR rate output constraint will be remapped to an EGR flow input constraint, see Section 2.3.3). The choice of inputs, to be determined by the MPC controller, are the VGT position (specifically, the VGT duty cycle command), u_{VGT}, and EGR flow, W_{EGR}. The desired EGR flow, determined by the MPC controller, is inverted to the EGR valve position based on the EGR valve orifice flow equation, [38,39]. The choice of EGR flow instead of EGR valve position is motivated by the observation that the DAP plant becomes "more linear" when EGR flow is used, [38,113]. Control constraints must also be enforced on the VGT position and EGR flow. The EGR throttle is controlled through only feed-forward values obtained from a lookup-table.

Due to computational limitations of production engine control units, [26], the general tube MPC described in Section 4.2 cannot be applied directly. An increase in computational complexity of tube MPC compared to standard MPC, particularly when a short control horizon is used, results from the increased number of optimization variables, i.e., for $\bar{\xi}_{0|k}$ in (4.24), and additional constraints, i.e., $\xi_k - \bar{\xi}_{0|k} \in F_\infty$ in (4.24). For the DAP example, a control horizon of 1-3 steps is typically used, [49, 56, 84, 101, 115]. Thus, the addition of $\bar{\xi}_{0|k}$ as an optimization variable significantly increases the total number of optimization variables and, as a result, may significantly increase computation time and memory, particularly when explicit MPC, [5,23], is used. Thus an approximate rate-based tube MPC formulation is now introduced with lower computational overhead compared to (4.24) and is less conservative than (4.24). However, with this approach, there will be no a priori guarantee of recursive feasibility and stability. Having a guarantee of this kind is only partly valuable since (i) the ultimate implementation is for the nonlinear diesel air path system rather than for a linear system, (ii) the ultimate implementation uses soft output constraints so feasibility is guaranteed as the result of the problem reformulation, and (iii) the assumption that the reference, r, is constant is not valid, i.e., r changes as a function of the operating condition. The constrained domain of attraction with the approximate rate-based tube

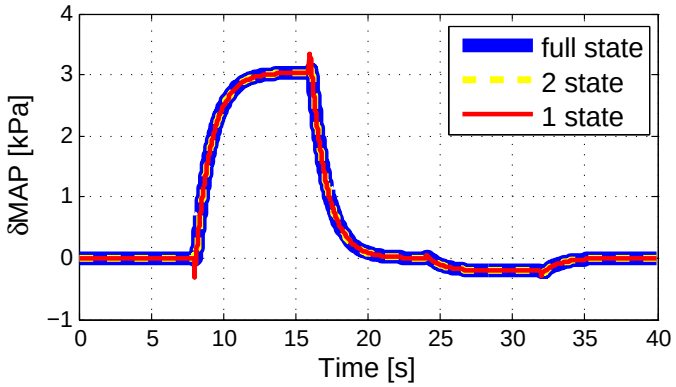

Figure 4.1: Comparison of MAP response to VGT position and EGR flow between the full state, 2 state, and 1 state linear model.

MPC formulation can be established by a posteriori analysis, e.g., based on linear matrix inequalities, [89], or through simulation, [49, 55].

For prediction, a single state model will be used because it is important to reduce the number of additional optimizations variables, i.e., $\bar{\xi}_{0|k}$, to manage computational complexity. The single state model is obtained through linearization of a high fidelity nonlinear DAP model and model order reduction, [14, 67]. The resulting model has the form of (4.1)-(4.2) where x_k is a non-physical state, y_k is composed of MAP and EGR rate, u_k is composed of VGT position and EGR flow, and w_k is composed of engine speed and fuel rate which are unknown, but bounded, in prediction. The EGR flow input is delivered by inversion of the EGR valve orifice flow equation (2.3). Figures 4.1 and 4.2 shows a comparison between the full state model model, 2 state reduced model, and 1 state reduced model. The outputs, MAP and EGR rate, respond to steps in VGT position and EGR flow where δ denotes the deviation from the equilibrium of the chosen linearization point. These responses show that a 1 state model is sufficient to approximately match the input-output response of the full state model.

The proposed approximate MPC formulation for the DAP has the following form,

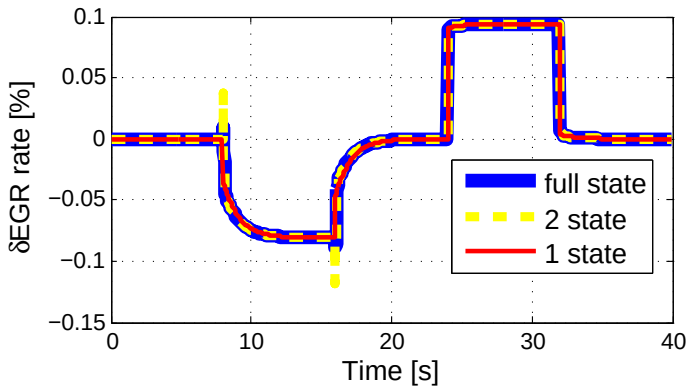

Figure 4.2: Comparison of EGR rate response to VGT position and EGR flow between the full state, 2 state, and 1 state linear model.

$$
\begin{aligned}
\min_{\bar{\xi}_{0|k},\, \Delta\bar{u}_{i|k},\, i\in\{0,\dots N-1\}} \quad & \bar{\xi}_{N|k}^T P \bar{\xi}_{N|k} + \sum_{i=0}^{N-1} \bar{\xi}_{i|k}^T Q \bar{\xi}_{i|k} + \Delta\bar{u}_{i|k}^T R \Delta\bar{u}_{i|k}, \\
\text{subject to :} \quad & \bar{\xi}_{i+1|k} = \bar{A}\bar{\xi}_{i|k} + \bar{B}\Delta\bar{u}_{i|k}, \\
& \bar{C}\left(\bar{\xi}_{i|k} + \begin{bmatrix} 0 \\ \\ r \end{bmatrix}\right) \in \mathbb{Y} \ominus \bar{C}F_\infty, \\
& u_{k-1} + \sum_{j=0}^{i} \Delta\bar{u}_{j|k} \in \mathbb{U} \ominus (i+1)KF_\infty, \\
& \xi_k - \bar{\xi}_{0|k} \in F_\infty,
\end{aligned}
\tag{4.28}
$$

where only output constraints are considered because the states are non-physical as a result of model order reduction and/or system identification. The constraint $\bar{C}\bar{\xi}_{i|k} \in \mathbb{Y} \ominus \bar{C}F_\infty$ in (4.28) guarantees that output constraints are satisfied over the prediction horizon for the true rate-based system (4.4), i.e., $\bar{C}\xi_{i|k} \in \mathbb{Y}$. The constraint $u_{k-1} + \sum_{j=0}^{i} \Delta\bar{u}_{j|k} \in \mathbb{U} \ominus (i+1)KF_\infty$ in (4.28) guarantees that $u_{i|k} \in \mathbb{U}$ over the prediction horizon, where the term $(i+1)$ is needed because the additional control increment, $K\eta_{i|k}$, must be applied at each predicted step to keep $\eta_{i|k} \in F_\infty$. Additionally, the prediction horizon must be limited, or K chosen appropriately, such that $NKF_\infty \subseteq \mathbb{U}$ for the MPC optimization problem to be feasible. Note that the terminal set constraint is no longer used, thus recursive feasibility of (4.28) and stability of the resulting closed loop system are no longer a priori guaranteed. However, assuming that P

is the solution to the Riccati equation associated with the corresponding LQR controller and K is the associated feedback, F_∞ is robustly exponentially stable with a domain of attraction that can be computed a posteriori. This is because the optimized policy for the nominal control will be $\Delta\bar{u}^\star_{0|k} = K\bar{\xi}^\star_{0|k}$ and the true control will be $\Delta u_k = \Delta\bar{u}^\star_{0|k} + K(\xi_k - \bar{\xi}^\star_{0|k}) = K\xi_k$.

The key difference between the approximate formulation (4.28) and the original formulation (4.24), is that the step, (4.14)-(4.21), of converting the constraints from the true states and controls, $\begin{bmatrix} x_k^T & u_{k-1}^T \end{bmatrix}^T$, to constraints on the nominal rate-based state, $\bar{\xi}_k$, is completely bypassed. The advantage of the approximate formulation is that only bounds for disturbances increments, $\mathbb{D}$, are required. The bounds for the absolute disturbances, $\mathbb{W}$, are completely unused, leading to a less conservative robust controller with (4.28) compared to (4.24). Additionally, the terminal set constraint is not used in (4.28) which reduces the computational complexity, see Chapter 2. Currently, it is unknown how O_∞ can be computed when there are absolute control constraints and uncertain disturbances without explicitly using the fact that $w_k \in \mathbb{W}$, i.e., only using $\Delta w_k \in \mathbb{D}$. Not using the terminal set constraint to guarantee recursive feasibility and stability can be justified as previously discussed. Furthermore using a terminal set constraint in addition to a short horizon, e.g., $N = 1$, would greatly limit the set of states for which (4.28) and (4.24) is feasible, which is undesirable. For example, the use of a terminal set constraint is investigated in Chapter 5 and it is demonstrated through simulations that a MPC controller with a stabilizing terminal constraint can be infeasible even if the same MPC controller without a stabilizing terminal constraint is already stabilizing.

The main disadvantage of the approximate formulation (4.28) compared to the original formulation (4.24) is that only constraints on the tracked outputs can be used, rather than general state and output constraints. Note that for rate-based non-tube MPC, state constraints can be added through adding constraints on additional outputs which are integrators of state increments [49]. However, if this were done in rate-based tube MPC, A_K in (4.12) would not be asymptotically stable, and F_∞ cannot be computed.

4.3.1 Approximate Tube MPC Applied to the DAP

For the DAP, the optimization problem is set up with a control horizon of 1 and $\Delta\bar{u}_{i|k} = 0$ for $i \geq 1$ in (4.28). The output constraint is enforced not at every time step, but rather at only at selected steps, $i = 12$ and $i = 40$ in (4.28). The details of the DAP optimization problem setup follows.

First, F_∞ is approximated. The process described in [93] can be used to compute a polytopic RPI outer approximation of F_∞. However, it was found in [42] that when A_K has eigenvalues close to the unit circle, the approximation of [93] becomes numerically difficult to compute due to the need to construct the convex hull of progressively increasing number of points. Furthermore the number of facets, i.e., number of linear constraints which describe the polytope, for the approximation of [93] would be prohibitively large and cannot be used for the DAP application, [42]. Thus, a simple box shaped approximation of F_∞ is constructed through simulations. The box approximation is not RPI. However, the approximate tube MPC formulation (4.28) already does not guarantee stability or recursive feasibility and the ultimate application will use soft constraints to guarantee feasibility. Then the fact that the box approximation is not RPI does not affect the properties of (4.28), e.g., F_∞ is still robustly exponentially stable and the domain of attraction can be computed a posteriori.

The box approximation of F_∞, hereby denoted F_{box}, is constructed by computing interior points of F_∞ based on randomly generated trajectories of for the error system (4.12). The bounds for $\Delta w_k \in \mathbb{D}$ are chosen based on the 95-th percentile of the sample to sample engine speed and fueling rate changes during the New European Drive Cycle (NEDC). An enclosing ellipse, $\xi^T E^{-1} \xi \leq 1$, is then computed using the Ellipsoidal Toolbox [62]. Let $E = U\Sigma V^T$ be the singular value decomposition of E. The ellipse can be enclosed by a box, $F_{box} = \{\xi | H_{box}\xi \leq G_{box}\}$, where

$$H_{box} = \begin{bmatrix} V^T \\ -V^T \end{bmatrix}, G_{box} = \begin{bmatrix} \sqrt{\Sigma} 1_{n_x+n_y \times 1} \\ \sqrt{\Sigma} 1_{n_x+n_y \times 1} \end{bmatrix}, \tag{4.29}$$

and $1_{n_x+n_y\times 1}$ denotes a column vector of ones with $n_x + n_y$ elements. The cost function is reformulated with $N = 1$,

$$J = \bar{\xi}_{1|k}^T P \bar{\xi}_{1|k} + \bar{\xi}_{0|k}^T Q \bar{\xi}_{0|k} + \Delta \bar{u}_{0|k}^T R \Delta \bar{u}_{0|k} + M\epsilon^2, \tag{4.30}$$

where ϵ is a slack variable used to treat the output constraint as soft.

The output constraints that are considered for this DAP application are on the maximum MAP overshoot and maximum EGR rate. The EGR rate is defined as the ratio of EGR flow, W_{EGR}, to cylinder flow, W_{cyl}, $\chi_{EGR} = W_{EGR}/W_{cyl}$. Because the EGR flow is treated as a control input, the EGR rate constraint is remapped to a maximum EGR flow control constraint,

$$u_{EGR,k} \leq \chi_{EGR}^{max} W_{cyl,k}. \tag{4.31}$$

This way, only a single output constraint needs to be considered. As in [49], intermittent constraint enforcement is used to reduce computational complexity. The output constraint is implemented as

$$\bar{C}_{MAP}\bar{\xi}_{i|k} + \epsilon \leq p_{in}^{max} - \max_{\xi \in F_{box}} \bar{C}_{MAP}\xi, \; \forall i \in N_{ICE} \subset N^+, \tag{4.32}$$

where p_{in}^{max} in (4.32) is the MAP overshoot limit, N_{ICE} denotes the set of indices of intermittently enforced constraints, and $\bar{C}_{MAP}$ corresponds to the row of $\bar{C}$ that extracts the MAP output. For this DAP application, $N_{ICE} = \{12, 40\}$, see Chapter 2.

The nominal VGT position, $\bar{u}_{VGT,k}$, constraint is implemented as,

$$\begin{aligned} u_{VGT,k-1} + \Delta \bar{u}_{VGT,0|k} &\leq VGT^{max} - \max_{\xi \in F_{box}} K_{VGT}\xi, \\ u_{VGT,k-1} + \Delta \bar{u}_{VGT,0|k} &\geq VGT^{min} - \min_{\xi \in F_{box}} K_{VGT}\xi, \end{aligned} \tag{4.33}$$

where K_{VGT} corresponds to the row of K that contributes to VGT control. Similarly, the

EGR flow control constraint is implemented as,

$$\begin{aligned} u_{EGR,k-1} + \Delta\bar{u}_{EGR,0|k} &\leq \min\{W_{EGR}^{max}, \chi_{EGR}^{max} W_{cyl}\} - \max_{\xi \in F_{box}} K_{VGT}\xi, \\ u_{EGR,k-1} + \Delta\bar{u}_{EGR,0|k} &\geq W_{EGR}^{min} - \min_{\xi \in F_{box}} K_{EGR}\xi, \end{aligned} \tag{4.34}$$

where K_{EGR} corresponds to the row of K that contributes to EGR control.

The resulting optimization problem minimizes the cost (4.30) subject to the constraints (4.32)-(4.34), the tube constraint, $\xi_k - \bar{\xi}_{0|k} \in F_{box}$, the nominal dynamics, $\bar{\xi}_{i+1|k} = \bar{A}\bar{\xi}_{i|k} + \bar{B}\Delta\bar{u}_{i|k}$, and a constraint that holds $\Delta\bar{u}_{i|k} = 0$ for $i \geq 1$.

Previous applications of MPC to the DAP, [49,56,84,101], utilize explicit MPC, [5], which precomputes and stores the solution to the MPC QP problem as a piecewise affine (PWA) function. However, the PWA map for approximate rate-based tube MPC has on the order of 2,000 regions compared to 20-30 regions for standard rate-based MPC. Thus an online QP solver will be used.

The approximate rate-based tube MPC problem can be compactly expressed as a QP problem,

$$\begin{aligned} \min_{z} \quad & \tfrac{1}{2} z^T \mathcal{Q} z \\ \text{subject to}: \quad & Hz \leq G, \end{aligned} \tag{4.35}$$

where $z = \begin{bmatrix} \Delta\bar{u}_{0|k}^T & \bar{\xi}_{0|k}^T & \epsilon \end{bmatrix}^T$. The dual problem of (4.35) has the form,

$$\begin{aligned} \min_{\lambda} \quad & \tfrac{1}{2}\lambda^T H \mathcal{Q}^{-1} H^T \lambda + \lambda^T G \\ \text{subject to}: \quad & \lambda \geq 0. \end{aligned} \tag{4.36}$$

The algorithm of [97] is used to solve the dual problem (4.36). The primal variables, z, are recovered after the completion of a predefined number of iterations as $z = -\mathcal{Q}^{-1}H^T\lambda$. The simulations that will be presented in Section 4.3.2 were conducted using both 200 and 12 solver iterations. The maximum difference in the MAP and EGR rate trajectories when different solver iterations are used is 0.4kPa and 0.6%, respectively, while 12 solver iterations

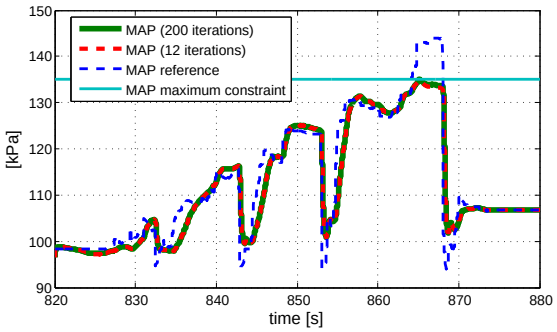

Figure 4.3: MAP response during NEDC simulation.

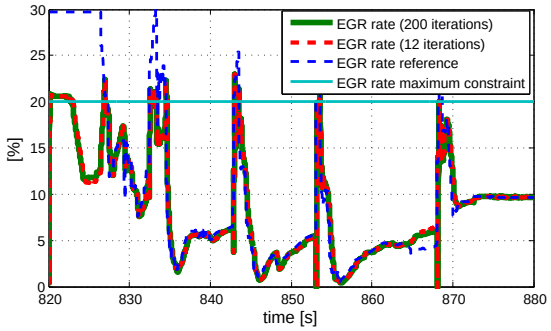

Figure 4.4: EGR rate response during NEDC simulation.

results in lower chronometric load. The experimental results, also shown in Section 4.3.2, use 12 solver iterations.

4.3.2 Simulation and Experimental Results Using Approximate Tube MPC

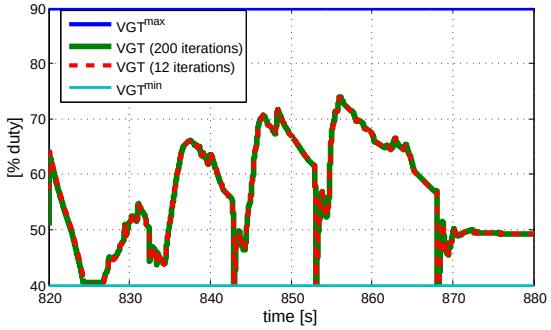

Figure 4.5: VGT response during NEDC simulation.

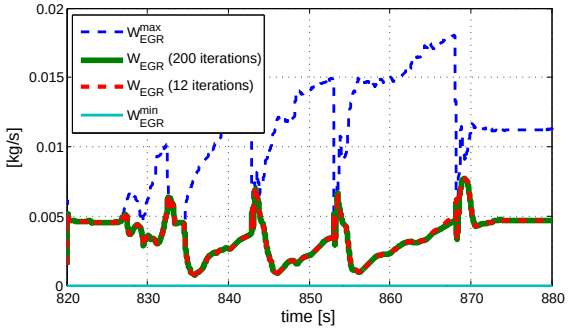

Figure 4.6: EGR flow response during NEDC simulation.

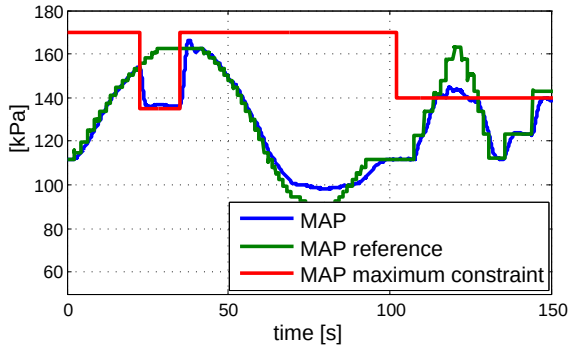

Figure 4.7: MAP response to fuel rate changes at 2,000 rpm.

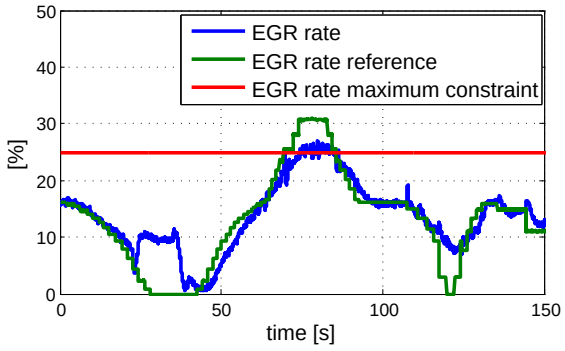

Figure 4.8: EGR rate response to fuel rate changes at 2,000 rpm.

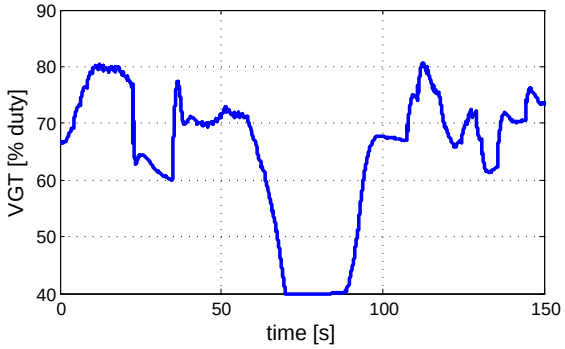

Figure 4.9: VGT response to fuel rate changes at 2,000 rpm.

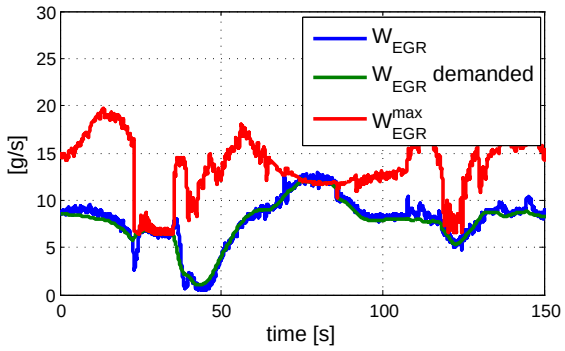

Figure 4.10: EGR flow response to fuel rate changes at 2,000 rpm.

Simulations using the approximate rate-based tube MPC controller have been performed in the loop with the high fidelity nonlinear DAP model over the NEDC. A portion of the NEDC closed loop response is shown in Figures 4.3-4.6. This portion of the NEDC is characterized by acceleration phases followed by fuel cuts and shift events. In general, the controller exhibits good tracking performance. Figure 4.3 shows the MAP constraint becoming active at 865 sec. Because the entire tube, $\{\bar{\xi}_k\} \oplus F_{box}$, must lie under the constraint, some offset can be seen between the achieved trajectory and the constraint upper bound. During this period, the engine speed and fueling rate continue to increase which the nominal controller is able to counteract. The EGR rate trajectory loses tracking at this time. This is because the controller will command a higher EGR flow to reduce MAP to satisfy the constraint. The EGR rate constraint is also satisfied with small violation. This is because the EGR flow command must be inverted to recover the EGR valve command, which is not exact. There is a loss of tracking in both MAP and EGR rate at 823 sec. This is due to an artificial DC gain reversal at near idle speed conditions inside the model. It has been verified that this DC gain reversal does not occur in the experimental engine.

Experimental results have been obtained using the approximate rate-based tube MPC controller on the diesel engine. These experiments were conducted on a steady state dynamometer at different engine speeds at Toyota Motor Corporation. The results presented in this chapter were conducted at 2,000 rpm engine speed. Figures 4.7-4.10 show the response to different fueling rate changes. The controller exhibits good tracking. As shown in Figure 4.7, the controller is able to enforce the MAP constraint with small violation at 125 sec. The small violation occurs because the MAP constraint is treated as soft and as such, the controller must weigh the opposing objectives of tracking and constraint satisfaction. Additionally, the discrepancy between the linear model and nonlinear engine behavior can contribute to constraint violation. At this time, EGR rate looses tracking to drive down MAP. At 25 sec, there is an instantaneous change in the MAP constraint and the controller is able to quickly drive down MAP, though with small violation due to similar reasons as at

125 sec. At 75 sec MAP tracking is lost because the EGR rate constraint becomes active and the VGT has no more authority to open further.

4.4 Reduced Complexity Tube MPC

The approximate tube MPC strategy presented in Section 4.3 was able to reduce the computational complexity of tube MPC by eliminating the terminal set constraint. Furthermore, for the application to the DAP, F_∞, was approximated by a non-RPI box set to obtain a simple polytopic representation for the tube. Despite the limitations of the approximate formulation (4.28), i.e., the MOAS for the closed loop system is not known explicitly, it indeed works quite well for the DAP application, insofar as it is able to enforce the desired MAP constraint in the presence of disturbances throughout simulations and experiments. However, one may wish for a technique that maintains the same MOAS as tube MPC with reduced complexity compared to tube MPC. Toward this end, a reduced complexity tube MPC strategy is developed in the following. The idea is to split the conventional tube optimization problem into two parts where the sum of the computation cost is less than the original. While this procedure will be formulated for rate-based tube MPC, the developed technique can easily be employed with approximate rate-based tube MPC and with conventional tube MPC.

To achieve complexity reduction of tube MPC, consider the situation when the nominal state, $\bar{\xi}_{0|k}$, is not chosen by the tube MPC optimization problem but rather by some external supplementary function. The optimization problem where $\bar{\xi}_{0|k}$ is determined externally will

be referred to as the "open loop" tube MPC problem, which has the following form,

$$\begin{aligned} \min_{\Delta\bar{u}_{i|k},\, i\in\{0,\dots N-1\}} \quad & \bar{\xi}_{N|k}^T P \bar{\xi}_{N|k} + \sum_{i=0}^{N-1} \bar{\xi}_{i|k}^T Q \bar{\xi}_{i|k} + \Delta\bar{u}_{i|k}^T R \Delta\bar{u}_{i|k}, \\ \text{subject to}: \quad & \bar{\xi}_{i+1|k} = \bar{A}\bar{\xi}_{i|k} + \bar{B}\Delta\bar{u}_{i|k}, \\ & \Pi\left(\bar{\xi}_{i|k} + \begin{bmatrix} 0 \\ \\ r \end{bmatrix}\right) \in \bar{\mathbb{X}}, \\ & \bar{\xi}_{N|k} \in O_\infty, \end{aligned} \tag{4.37}$$

where, compared to the standard tube problem (4.24), $\bar{\xi}_{0|k}$ is no longer an optimization variable but rather treated as an input parameter to (4.37), and the tube constraint, $\xi_k - \bar{\xi}_{0|k} \in F_\infty$, has also been removed. As in (4.24), the tightened constraints, $\bar{\mathbb{X}}$ in (4.37), is defined through (4.21) and O_∞ in (4.37) is the MOAS for the system (4.22) and constraint (4.23).

Open loop tube MPC has the same computational complexity as non-tube MPC. This is because the Minkowski set difference operation of polytopic sets, in (4.21), does not increase the number of facets from the original set, $\mathbb{X} \times \mathbb{U}$. Thus the number of total inequality constraints will remain the same or be less than in non-tube MPC. The number of optimization variables is the same for open loop tube MPC as with non-tube MPC. Particularly with active-set methods, the computational complexity of solving a QP is purely a function of the number of optimization variables and number of inequality constraints, see Chapter 2.

One way to choose $\bar{\xi}_{0|k}$ for the open loop tube MPC problem is to set $\bar{\xi}_{0|k} = \bar{\xi}_{1|k-1}$, and set $\bar{\xi}_{0|0} = \xi_0$. Then the open loop tube MPC strategy just evolves the nominal trajectory with no knowledge of the true state. Then the nominal controller, $K\eta_k$ in (4.11), keeps the true state close (within the RPI tube, F_∞) to the nominal trajectory. This strategy is exactly the strategy of [73]. Figure 4.11 shows a simulation of using this open loop strategy with the approximate rate-based tube MPC for the DAP. The MAP constraint is enforced at 865 sec in a manner similar to the situation shown in Figure 4.3. There is a momentary violation of the nominal MAP trajectory at 865 sec. This is because the constraints are only enforced at steps 12 and 40; $N_{ICE} = \{12, 40\}$ in (4.32). Even though the nominal MAP trajectory,

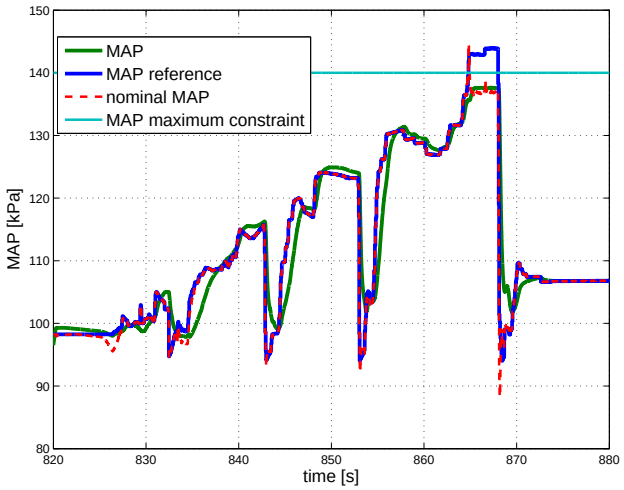

Figure 4.11: MAP response during NEDC simulation using open loop tube MPC.

$\bar{\xi}_{0|k}$, does not satisfy the constraint, $\bar{\xi}_{12|k}$ and $\bar{\xi}_{40|k}$ does.

Note that the set of $\bar{\xi}_0$ for which (4.37) is recursively feasible is $\bar{\mathbb{X}}_N$, (4.25). Unfortunately, the set $\bar{\mathbb{X}}_N$ is smaller than the MOAS for the initial true state under conventional tube MPC control which is $\bar{\mathbb{X}}_N \oplus F_\infty$, see Proposition 4.1.

Rather than just choosing, $\bar{\xi}_{0|k} = \bar{\xi}_{1|k-1}$, a supplemental optimization problem can be defined to choose $\bar{\xi}_{0|k}$, which, in combination with (4.37), will recover the MOAS under conventional tube MPC control. This supplemental optimization problem follows,

$$\begin{aligned} \min_{\bar{\xi}_{0|k}} \quad & \tilde{J}(\bar{\xi}_{0|k}, \xi_k), \\ \text{subject to :} \quad & \bar{\xi}_{0|k} \in \bar{\mathbb{X}}_N, \\ & \xi_k - \bar{\xi}_{0|k} \in F_\infty, \end{aligned} \tag{4.38}$$

where the constraint $\bar{\xi}_{0|k} \in \bar{\mathbb{X}}_N$ ensures feasibility of (4.37) and $\xi_k - \bar{\xi}_{0|k} \in F_\infty$ is the conventional tube constraint. Then, by construction, the MOAS under conventional tube MPC control, $\bar{\mathbb{X}}_N \oplus F_\infty$, will be recovered.

4.4.1 Example

Here, two choices for $\tilde{J}(\bar{\xi}_{0|k}, \xi_k)$ are examined. The first, denoted as configuration 1 with

$\tilde{J}_1(\bar{\xi}_{0|k}, \xi_k)$, is

$$\tilde{J}_1(\bar{\xi}_{0|k}, \xi_k) = \left(\bar{\xi}_{0|k} - \xi_k\right)^T \left(\bar{\xi}_{0|k} - \xi_k\right), \tag{4.39}$$

and the second, denoted as configuration 2 with $\tilde{J}_2(\bar{\xi}_{0|k}, \xi_k)$, is

$$\tilde{J}_2(\bar{\xi}_{0|k}, \xi_k) = \bar{\xi}_{0|k}^T P \bar{\xi}_{0|k}. \tag{4.40}$$

A simple second order system example will be used to illustrate the effect of using the reduced complexity tube strategies, e.g., open loop, config. 1, and config. 2, versus conventional tube MPC. The baseline conventional tube MPC controller utilizes a non-rate-based formulation, [64, 74]. The model, tuning, and constraint parameters follow:

$$\begin{gathered} x_{k+1} = Ax_k + Bu_k + w_k, \\ A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, R = 0.1, \\ \begin{bmatrix} -50 \\ -50 \end{bmatrix} \leq x_k \leq \begin{bmatrix} 3 \\ 3 \end{bmatrix}, \\ 3 \leq u_k \leq 3, \\ \begin{bmatrix} -0.5 \\ -0.5 \end{bmatrix} \leq w_k \leq \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}. \end{gathered} \tag{4.41}$$

Figures 4.12-4.15 show closed loop responses for the second order system (4.41) with different tube MPC controllers with respective illustrations of their explicit representations. Figure 4.12 shows this for conventional tube MPC. Figure 4.13 shows the closed loop response with "open loop" tube MPC and the associated explicit representation. Because the nominal state is no longer treated as an optimization variable and the tube constraint does not need to be explicitly considered, the computational complexity, i.e., the number of regions of the explicit controller, for "open loop" tube MPC is clearly less than for conventional tube MPC. However the feasible region for "open loop" tube MPC, which is the union of all of the regions

in Figure 4.13-right, is smaller than for conventional tube MPC. Figure 4.14 shows the closed loop response with tube MPC with config. 1 and the explicit representation of the associated auxiliary function, (4.38) and (4.39). The total number of regions for the controller is the sum of the regions for the open loop tube controller 4.13-right and the number of regions for the auxiliary function used to choose $\bar{\xi}_{0|k}$, (4.38) and (4.39). Notice that, as long as $\xi_k \in \bar{\mathbb{X}}_N$, then $\bar{\xi}_{0|k} = \xi_k$. This leads to a very simple explicit representation for (4.38) and (4.39). Furthermore, as long as $\xi_k \in \bar{\mathbb{X}}_N \oplus F_\infty$, which is also the MOAS for conventional tube MPC, then (4.38) and (4.39) is recursively feasible. The closed loop trajectory of tube MPC with config. 1 is very similar to "open loop" tube MPC. Figure 4.15 shows the closed loop response with tube MPC with config. 2 and the explicit representation of the associated auxiliary function, (4.38) and (4.40). In this case, the regulation performance more closely resembles conventional tube MPC, however the explicit representation of the auxiliary function with config. 2 has more regions than with config. 1. This is because it is no longer guaranteed that $\bar{\xi}_{0|k} \neq \xi_k$ if $\xi_k \in \bar{\mathbb{X}}_N$. Finally, Figure 4.16 shows the total number of regions for the explicit representation of different types of tube MPC controllers. The number of regions grow at the same rate (slope in Figure 4.16) for "open loop" tube MPC, config. 1, and config 2. This is because the number of regions representing of the auxiliary function does not change as a function of the horizon length. Because conventional tube MPC includes the nominal state as an optimization variable and extra inequality constraints associated with the tube constraint, the rate of growth in the number of regions as a function of the horizon length is significantly faster.

4.5 Conclusions

This chapter describes the development of a robust model predictive controller for the diesel engine air path. The approach utilizes a rate-based formulation for tube MPC, with the novel observation that if rate bounds are known for the disturbances, then the conserva-

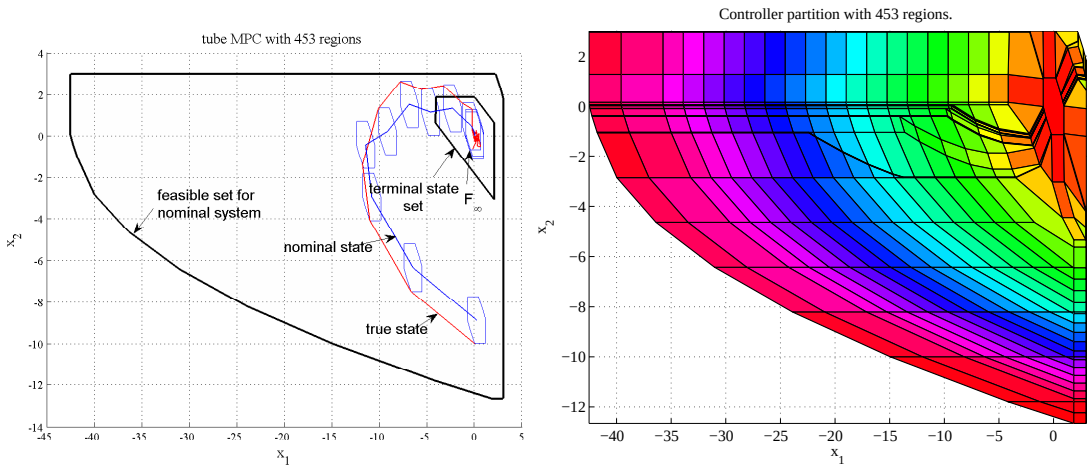

Figure 4.12: Conventional tube MPC example for the second order system (4.41): closed loop simulation (left) and explicit controller representation (right).

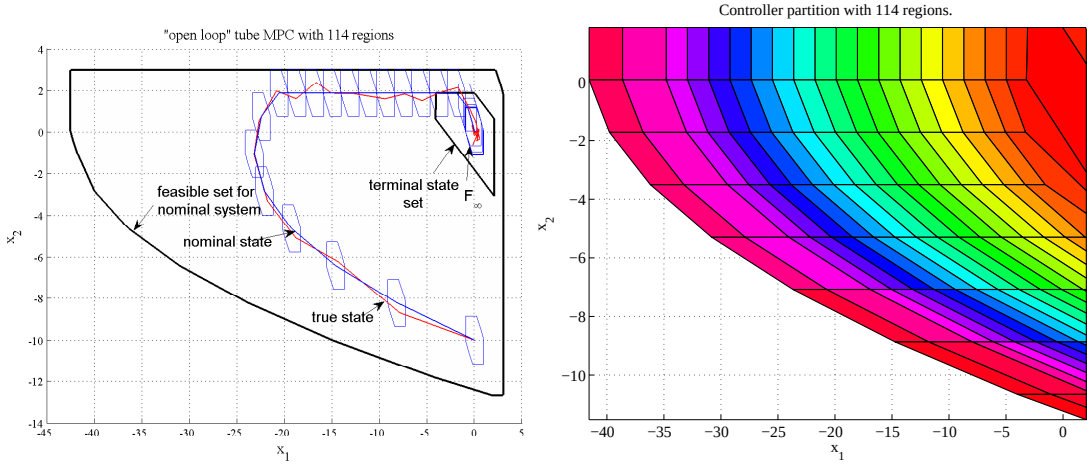

Figure 4.13: Open loop tube MPC example for the second order system (4.41): closed loop simulation (left) and explicit nominal controller representation (right).

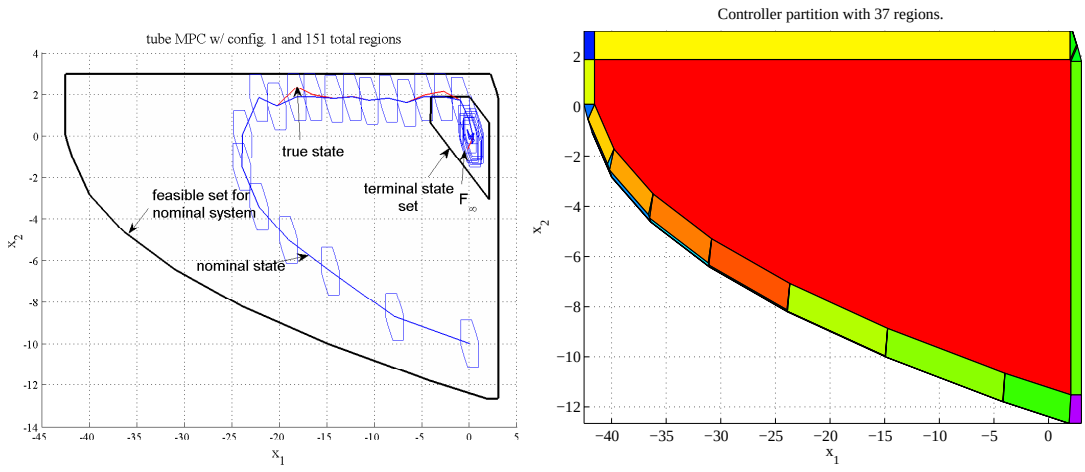

Figure 4.14: Tube MPC with config. 1 example for the second order system (4.41): closed loop simulation (left) and explicit representation of the auxiliary function (4.38) (right).

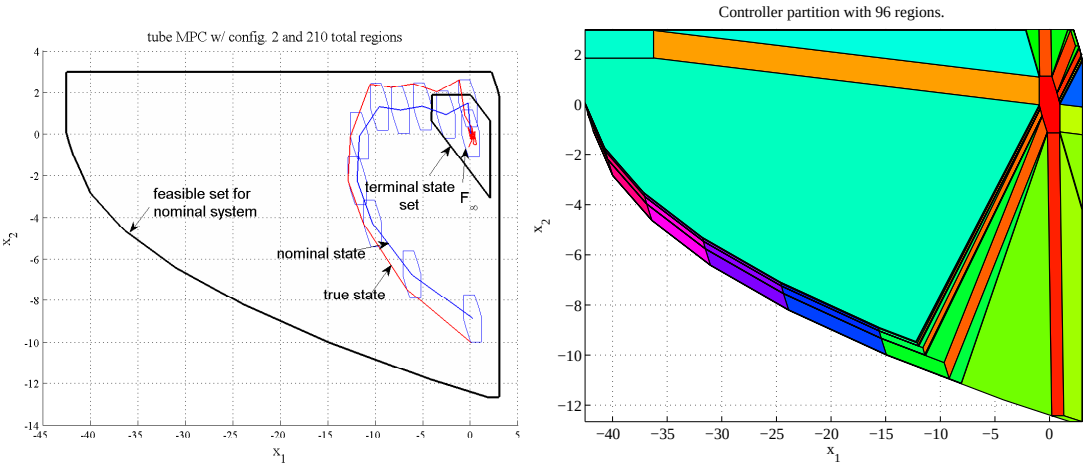

Figure 4.15: Tube MPC with config. 2 example for the second order system (4.41): closed loop simulation (left) and explicit representation of the auxiliary function (4.38) (right).

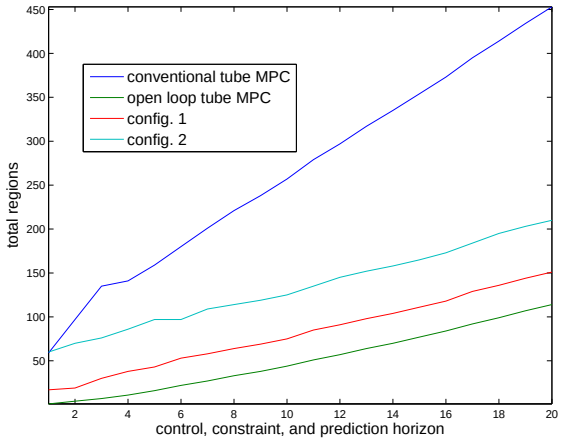

Figure 4.16: Number of total regions, including the number of regions of the auxiliary function (4.38), of different tube MPC strategies.

tiveness for tube MPC can be reduced. In an effort to reduce the computational complexity for standard tube MPC for the diesel air path application, various approximations were used which lead to a tube MPC design that does not explicitly guarantee constraint enforcement, however does in fact enforce constraints in practice as demonstrated in simulations and experiments. Additionally, a novel method to reduce the computational complexity of tube MPC, through splitting of the the MPC optimization problem, is developed which still maintains a guarantee of recursive constraint satisfaction and exactly the same maximal output admissible set as conventional tube MPC.

Chapter 5

Nonlinear Model Predictive Control

This chapter describes the development of a Nonlinear Model Predictive Controller (NMPC) for the diesel air path. The use of NMPC for the diesel engine air path is motivated by the ability of the controller to deal with nonlinear engine dynamics and handle input and output constraints. However, the Engine Control Unit (ECU) has limited computational resources and the addition of constraints can greatly increase the computational complexity of NMPC. A comparative assessment of the computation time and constraint violation for different NMPC problem formulations, constraint handling techniques, and solver techniques is presented. The ability of NMPC to regulate to specified set-points and to enforce constraints is demonstrated through nonlinear model simulations.

5.1 Introduction

The constrained multi-input multi-output nature of the diesel engine air path (DAP) control problem has motivated research into applications of Model Predictive Control (MPC), primarily linear MPC, to improve transient performance, reduce emissions, and ensure that constraints on actuators and on other engine variables are satisfied, see [36, 38, 41, 49, 56, 84, 101]. The linear MPC strategies typically rely on identification of multiple linear models valid locally around specified operating conditions and a separate MPC design for each operating condition. Explicit MPC, [5], is typically used for online implementation. Many of the

developments in this dissertation on linear MPC for the DAP, see Chapters 2-3, sought to reduce the degree of nonlinearity of the plant in order to facilitate the use of a single linear model and single explicit MPC design.

Recently, interest in nonlinear MPC (NMPC), see [30] and references therein, for DAP control has been increasing with the goal of providing better performance with lower calibration effort compared to linear MPC and conventional PID control. NMPC for diesel air path control has previously been thought to be computationally intractable, [33]. However, a recent study, [27], utilizing the Continuation and Generalized Minimum Residual method (C/GMRES) from [82] to solve the optimal control problem, has demonstrated improved computation times, compared to [33], requiring around 50 msec for an update. In [78], a sub-10 msec computation time for NMPC of the DAP was demonstrated, however, performance may be sacrificed due to its reliance on a simplifying parametrization of the predicted control sequence.

It must be recognized that, in practice, the available ECU computation time allotment is much smaller than the DAP control update period because the ECU performs many other functions besides air path control. Thus, considering that a standard DAP control update is applied every 10 msec to 50 msec, a sub-millisecond computation time for the DAP control is highly desirable. Explicit NMPC has been used in [31] for a turbocharged gasoline engine with a sub-millisecond computation time, however, as the authors note, the complexity of explicit NMPC suffers from the "curse of dimensionality" with respect to the number of input parameters and does not scale well with even the addition of integral action. Furthermore, with explicit NMPC, controller reconfigurability to model changes and ability to incorporate adaptive models is largely lost.

In the following, a C/GMRES strategy is considered, similar to [27], as the benchmark strategy because the utilization of C/GMRES has, so far, resulted in the fastest computation time for NMPC applied to the DAP without any of the apparent disadvantages of [31, 78]. Using the C/GMRES strategy as a launching point, the following main research contributions

have been made that together are able to achieve sub-millisecond computation times and demonstrate set-point tracking and constraint enforcement in the DAP.

- A data driven modeling approach similar to [27,35] is used to obtain a simple piecewise polynomial model that facilitates fast propagations of the state and co-state equations used in NMPC. C/GMRES is then applied with the polynomial model which represents the baseline design. Within C/GMRES, it was not previously known how inequality constraints, e.g., on states and controls, are handled. Different inequality constraint handling techniques are explored and a strategy that has considerable computational and performance advantages compared to C/GMRES is obtained. Furthermore, it is shown that zero-offset set-point tracking is achieved, which has not previously been demonstrated with NMPC for the DAP. This zero-offset set-point tracking is achieved through adaptation of the polynomial model. This work is also presented in [47]. Additional observations are also reported in this chapter that are not contained in [47].

- Kantorovich's method, [51], will be used where the Jacobian used in the solver is frozen, e.g., precomputed and frozen for all time or computed only at the first iteration. By bypassing the Jacobian computation, the computational complexity is reduced. Furthermore, this is done in a way that can handle inequality constraints with little computational overhead.

- With linear MPC, it has been known that rate-based MPC (also referred to as velocity-form MPC, [114]) achieves zero-offset set-point tracking while improving overall performance and mitigating the disadvantages of other strategies, see [18, 49, 85]. With NMPC, it is currently not well understood how to achieve zero-offset set-point tracking. Toward this end, a rate-based formulation is proposed for NMPC. With rate-based NMPC, adaptation or disturbance estimation is no longer needed to achieve zero-offset set-point tracking. This is advantageous for a number of reasons.

  – The integral action does not need to be tuned separately from the controller,

e.g., tuning adaptation parameters, and thus the control and estimation loop interactions are reduced.

– Discontinuous control actions at zone boundaries are mitigated. This is because estimators, which would otherwise have to be reinitialized during zone switches, are not required in the rate-based NMPC formulation for the DAP.

Simulation results for the various NMPC strategies in loop with engine mean-value-models are presented. In all of the simulations, only NMPC feedback is used, i.e., there is no feedforward so as to fully demonstrate the capability of the NMPC feedback controller.[1] Computational complexities of the various strategies are also discussed.

The remainder of this chapter is organized as follows. Section 5.2 describes the baseline NMPC design using C/GMRES, various constraint handling techniques within the C/GMRES framework, and the NMPC application to the DAP. Sections 5.3 and 5.4 describe variations of NMPC strategies and highlight their similarities and differences. Section 5.5 describes Kantorovich's method and its application to NMPC problems. Section 5.6 describes rate-based NMPC. Section 5.7 discusses the use of terminal set constraints to guarantee stability. Section 5.8 contains concluding remarks regarding NMPC for the DAP.

Note that the engine (Toyota KD engine) and model used in Section 5.2 are different than the engine (Toyota GD engine) and model used in Section 5.6 and the considered outputs are different. The prediction model used in Section 5.2 is a preliminary model used as a proof of concept and is made for the KD engine. The prediction model used in Section 5.6 is a refinement of the preliminary model in Section 5.2: it covers a larger area of the engine operating range and is updated for the GD engine and associated control objectives. The change to the GD engine has also been dictated by the fact that this engine represents a more recent technology that has been of interest on the application end. Modeling results are presented for both KD and GD engine in their respective sections.

[1]The potential issues of feedforward-feedback interactions are not addressed in this dissertation and require further research.

5.2 NMPC Using C/GMRES with Inequality Constraints

In the following, the application of NMPC is investigated for the DAP. One of the general approaches to NMPC is to transform the underlying optimization problem into an equality constrained root finding problem through the associated necessary conditions for optimality. In the context of nonlinear MPC, an interior point reformulation of the objective function is frequently used to handle inequality constraints, and a Newton type method is used to solve the root finding problem, [20]. Several methods for handling inequality constraints are investigated in this work. These methods include an exterior penalty method, the auxiliary variable method from [82], and semi-smooth transformations, [25, 51].

To work with the baseline strategy of C/GMRES, a continuous time nonlinear system is considered with the state equations,

$$\dot{x}(t) = f(x(t), u(t)), \tag{5.1}$$

where $x(t) \in \mathbb{R}^{n_x}$ and $u(t) \in \mathbb{R}^{n_u}$. Additionally, consider a NMPC problem based on minimizing a cost functional of the form,

$$J = \phi(x(T,t)) + \int_0^T l(x(\tau,t), u(\tau,t))d\tau, \tag{5.2}$$

where t is the current time, T is the prediction horizon, ϕ is the terminal penalty, l is the instantaneous state and control penalty, and τ, $0 \leq \tau \leq T$, is the running time over the prediction horizon. The minimization of (5.2) is performed subject to equality and inequality

constraints,

$$\begin{aligned} x_\tau(\tau,t) &= f(x(\tau,t),u(\tau,t)), && (5.3)\\ x(0,t) &= x(t), && (5.4)\\ g(x(\tau,t),u(\tau,t)) &= 0 \in \mathbb{R}^{n_g}, && (5.5)\\ h(x(\tau,t),u(\tau,t)) &\leq 0 \in \mathbb{R}^{n_h}, && (5.6)\\ \psi(x(T,t)) &\leq 0 \in \mathbb{R}^{n_\psi}, && (5.7) \end{aligned}$$

where $x_\tau = \partial x/\partial\tau$, $g = 0$ is an equality constraint, $h \leq 0$ is an inequality constraint, and $\psi \leq 0$ is a terminal state constraint. Let H denote the Hamiltonian,

$$H = l + p^T f + \mu^T g + \lambda^T h, \tag{5.8}$$

where p are Lagrange multipliers for the dynamic constraints (5.3), commonly referred to as co-states, μ are Lagrange multipliers for the equality constraint (5.5), and λ are Lagrange multipliers for the inequality constraint (5.6). The necessary conditions for optimality from

Pontrygin Maximum Principle (PMP), [13], are

$$
\begin{aligned}
x_\tau &= H_p = f,\ 0 \le \tau \le T, && (5.9)\\
p_\tau &= -H_x = -\left(l_x + f_x^T p + g_x^T \mu + h_x^T \lambda\right),\ 0 \le \tau \le T, && (5.10)\\
H_u &= l_u + f_u^T p + g_u^T \mu + h_u^T \lambda = 0,\ 0 \le \tau \le T, && (5.11)\\
g &= 0,\ 0 \le \tau \le T, && (5.12)\\
h &\le 0,\ 0 \le \tau \le T, && (5.13)\\
\lambda &\ge 0,\ 0 \le \tau \le T, && (5.14)\\
\lambda^T h &= 0,\ 0 \le \tau \le T, && (5.15)\\
x(0,t) &= x(t), && (5.16)\\
0 &= p(T,t) - \phi_x(x(T,t)) - \psi_x(x(T,t))^T v, && (5.17)\\
\psi(x(T,t)) &\le 0, && (5.18)\\
v &\ge 0, && (5.19)\\
v^T \psi(x(T,t)) &= 0, && (5.20)
\end{aligned}
$$

where subscripts x, p, and u denote partial derivatives with respect to x, p, and u, respectively, and v are Lagrange multipliers for the terminal inequality constraint (5.7).

Once the continuous time necessary conditions (5.9)-(5.20) are formed, they are discretized to facilitate the application of a numerical solver. The discretized conditions using

Euler discretization with a step size of $\Delta\tau$, assuming that $T/\Delta\tau = N \in \mathbb{Z}^+$, are

$$x_{i+1} = x_i + \Delta\tau f(x_i, u_i),\ i \in \{0, ..., N-1\}, \tag{5.21}$$

$$x_0 = x(t), \tag{5.22}$$

$$\begin{aligned} p_i = p_{i+1} + \Delta\tau(l_x(x_i, u_i) + f_x^T(x_i, u_i)p_{i+1} + g_x^T(x_i, u_i)\mu_i \\ + h_x^T(x_i, u_i)\lambda_i),\ i \in \{0, ..., N-1\}, \end{aligned} \tag{5.23}$$

$$0 = p_N - \phi_x(x_N) - \psi_x(x_N)^T v, \tag{5.24}$$

$$\begin{aligned} 0 = l_u(x_i, u_i) + f_u^T(x_i, u_i)p_{i+1} + g_u^T(x_i, u_i)\mu_i \\ + h_u^T(x_i, u_i)\lambda_i,\ i \in \{0, ..., N-1\}, \end{aligned} \tag{5.25}$$

$$0 = g(x_i, u_i),\ i \in \{0, ..., N-1\}, \tag{5.26}$$

$$0 \geq h(x_i, u_i),\ i \in \{0, ..., N-1\}, \tag{5.27}$$

$$\lambda_i \geq 0,\ i \in \{0, ..., N-1\}, \tag{5.28}$$

$$0 = \lambda_i^T h(x_i, u_i),\ i \in \{0, ..., N-1\}, \tag{5.29}$$

$$\psi(x_N) \leq 0, \tag{5.30}$$

$$v \geq 0, \tag{5.31}$$

$$v^T \psi(x_N) = 0. \tag{5.32}$$

In (5.21)-(5.32), $x_i = x(\tau_i)$, where τ_i is a time instant in the chosen time discretization mesh. The discretized control variables and Lagrange multipliers are denoted similarly with u_i, p_i, μ_i, and λ_i.

Now consider only the equality constrained NMPC problem without state, control, and terminal state inequality constraints. This will allow us to perform Newton type iterations to solve the necessary conditions and work in the framework of C/GMRES as a starting point. The necessary conditions without inequality constraints are (5.21), (5.22), (5.24), (5.26), and

the conditions,

$$p_i = p_{i+1} + \Delta\tau(l_x(x_i,u_i) + f_x^T(x_i,u_i)p_{i+1} + g_x^T(x_i,u_i)\mu_i), \quad i \in \{0,...,N-1\}, \tag{5.33}$$

$$0 = l_u(x_i,u_i) + f_u^T(x_i,u_i)p_{i+1} + g_u^T(x_i,u_i)\mu_i,\ i \in \{0,...,N-1\}. \tag{5.34}$$

The discretized inputs and Lagrange multipliers can be collected into a vector U,

$$U = \begin{bmatrix} u_0^T & \mu_0^T & ... & u_{N-1}^T & \mu_{N-1}^T \end{bmatrix}^T \in \mathbb{R}^{N(n_u+n_g)}. \tag{5.35}$$

The necessary conditions for the equality constrained NMPC problem can be written as an equation,

$$F(U,x(t)) = \begin{bmatrix} l_u(x_0,u_0) + f_u^T(x_0,u_0)p_1 + g_u^T(x_0,u_0)\mu_0 \\ g(x_0,u_0) \\ \vdots \\ l_u(x_{N-1},u_{N-1}) + f_u^T(x_{N-1},u_{N-1})p_N + g_u^T(x_{N-1},u_{N-1})\mu_{N-1} \\ g(x_{N-1},u_{N-1}) \end{bmatrix} = 0 \in \mathbb{R}^{N(n_u+n_g)}, \tag{5.36}$$

where $x_0,...,x_N$ and $p_N,...,p_1$ implicitly satisfy (5.21), (5.22), (5.24), and (5.33) given U and $x(t)$. Newton's method, [58], can then be used to solve the root finding problem, $F = 0$, and find a solution to the necessary conditions. The Newton iteration is

$$F_U(U_k,x(t))\Delta U_k = -F(U_k,x(t)), \tag{5.37}$$

$$U_{k+1} = U_k + \Delta U_k, \tag{5.38}$$

where U_{k+1} is the next iterate of U_k, and F_U is the Jacobian of F with respect to U. A Krylov space method, specifically the Forward Difference Generalized Minimal Residual

(FD-GMRES) method, can be used to efficiently solve (5.37) for ΔU_k without the need to explicit compute F_U, see [58].

Let s denote the cost of computing (5.21), (5.33), and (5.34) at stage i in the horizon. The complexity of computing $F(U_k, x(t))$ is $\mathcal{O}(N)$. FD-GMRES, see Algorithm 6.2.1 in [58], is an iterative solver which takes a maximum of $N(n_u + n_g)$ steps where, during each step, F is computed once and $\mathcal{O}(N)$ operations are used to construct the basis for the Krylov space. The final step of the algorithm attempts to find the minimum residual of a linear equation requiring $\mathcal{O}(N^3)$ computations. In total, the complexity of FD-GMRES is $\mathcal{O}(N^3)$ for sufficiently large N. With FD-GMRES, (5.37) does not need to be solved exactly, and if residuals are tolerable in the control strategy, i.e., controller performance is acceptable, then FD-GMRES can be terminated in k_{max} steps with $1 \leq k_{max} < N(n_u + n_g)$. Choosing $k_{max} = N(n_u + n_g)$ solves (5.37) exactly. With both k_{max} and N as design parameters, the complexity of FD-GMRES is $o(k_{max}^3 + k_{max}^2 + (k_{max} + 1)Ns)$. See [58] for convergence properties of FD-GMRES.

Note that with MPC of the DAP, parameters affecting the computational complexity are typically kept small, see Chapter 2. Thus, it is important to examine the computational complexity using both the Big-O and little-o notation, see Definitions 1.1 and 1.2.

A predictor-corrector strategy such as the Continuation/GMRES algorithm in [82] can also be used to solve the root finding problem. The goal of the continuation part of C/GMRES is to find a control U that drives $F(U, x(t))$ to zero in time, i.e., the following is satisfied,

$$\dot{F}(U, x(t)) = A_s F(U, x(t)), \tag{5.39}$$

where A_s is a stable matrix used for tuning. By differentiating F with respect to time, t, on the left hand side of (5.39), the expression,

$$F_U(U, x(t))\dot{U} = A_s F(U, x(t)) - F_x(U, x(t))\dot{x}, \tag{5.40}$$

is obtained. This is essentially a Newton step in continuous-time where $F_x(U, x(t))\dot{x}$ is a predictor term. FD-GMRES can also be used to solve (5.40) for $\dot{U}$. The complexity of solving (5.40) with FD-GMRES is $o(k_{max}^3 + k_{max}^2 + (k_{max} + 2)Ns)$ due to the additional computation of $F_x(U, x(t))\dot{x}$ and the complexity of solving (5.40) is $\mathcal{O}(N^3)$ for sufficiently large N.

5.2.1 Inequality Constraints

A major motivation for the use of MPC is the ability to handle inequality constraints. However, Newton type strategies such as FD-GMRES, C/GMRES, and etc. do not incorporate inequality conditions (5.27)-(5.32) directly. In the following, several methods for incorporating inequality constraints are considered. The methods investigated are the auxiliary variable method as in C/GMRES, [82], an exterior penalty method, and a semi-smooth transformation using Fischer-Burmeister functions, [25, 51]. In the following, let h_j denote the j-th row of the inequality constraint (5.6), $h(x(\tau, t), u(\tau, t)) \leq 0$.

Auxiliary Variable Method

In the auxiliary variable method used in C/GMRES, additional optimization variables are included to transform inequality constraints into equality constraints. The transformation is often referred to as Valentine's transformation, [52]. Let $\eta(\tau, t) \in \mathbb{R}^{n_h}$ be a vector of auxiliary variables that is used to transform the inequality constraint (5.6) into an equality constraint,

$$h(x(\tau, t), u(\tau, t)) + \eta(\tau, t)^2 = 0, \tag{5.41}$$

where $\eta(\tau, t)^2$ denotes a vector with components that are square of elements of $\eta(\tau, t)$. A penalty, $-r^T\eta(\tau, t)$, with $r \geq 0$, is added to keep the optimization problem well-conditioned.[2]

[2]As noted in [20], the auxiliary variable method can be viewed as an interior penalty method since the auxiliary variable can be written as $\eta(\tau, t) = \sqrt{-h(x(\tau, t), u(\tau, t))}$ and substituted into the cost function, with r acting as an interior penalty weight. Further, note that the Jacobian of the necessary conditions will be singular when $h(x_i, u_i) = 0$, hence the need for a penalty to keep $h(x_i, u_i) \neq 0$.

This leads to an approximate objective function,

$$J_{AV} = \phi(x(T,t)) + \int_0^T l(x(\tau,t), u(\tau,t)) - r^T \eta(\tau,t) d\tau. \tag{5.42}$$

The equality constraints (5.5) and (5.41) can be concatenated into

$$\tilde{g}(x(\tau,t), u(\tau,t)) = \begin{bmatrix} g(x(\tau,t), u(\tau,t)) \\ h(x(\tau,t), u(\tau,t)) + \eta(\tau,t)^2 \end{bmatrix} = 0 \in \mathbb{R}^{n_h+n_g}, \tag{5.43}$$

with corresponding Lagrange multipliers, $\tilde{\mu} \in \mathbb{R}^{n_h+n_g}$. The optimization problem is then to minimize (5.42) subject to equality constraints (5.3), (5.4), and (5.43). The vector of collected optimization variables becomes

$$\tilde{U} = \begin{bmatrix} u_0^T & \eta_0^T & \tilde{\mu}_0^T & \dots & u_{N-1}^T & \eta_{N-1}^T & \tilde{\mu}_{N-1}^T \end{bmatrix}^T \in \mathbb{R}^{N(n_u+n_g+2n_h)} \tag{5.44}$$

and the complexity of FD-GMRES becomes $o(k_{max}^3 + k_{max}^2 + (k_{max}+1)N\tilde{s})$, where $1 \leq k_{max} \leq N(n_u + n_g + 2n_h)$ and $\tilde{s} > s$ accounts for the additional stage complexity associated with the extra equality constraint (5.41). Again for sufficiently large N, the complexity of FD-GMRES remains $\mathcal{O}(N^3)$.

Exterior Penalty Method

The exterior penalty method adds a penalty on constraint violations to the cost function. Then the inequality constrained optimization problem is approximated by replacing the objective function (5.2) with

$$J_{EP} = \phi(x(T,t)) + \int_0^T l(x(\tau,t), u(\tau,t)) + \Sigma_{j=1}^{n_h} \gamma_j(x(\tau,t), u(\tau,t)) d\tau, \tag{5.45}$$

where

$$\gamma_j(x,u) = \begin{cases} 0, & h_j(x,u) \leq 0, \\ \frac{1}{2} r_j h_j(x,u)^2 & h_j(x,u) > 0, \end{cases} \tag{5.46}$$

with $r \geq 0$. The objective function (5.45) is then minimized subject to equality constraints (5.3)-(5.5). Unlike the auxiliary variable method, no extra optimization variables are added to U in (5.35) and the worst case complexity of FD-GMRES is $o(k_{max}^3 + k_{max}^2 + (k_{max}+1)Ns_\gamma)$, where $1 \leq k_{max} \leq N(n_u + n_g)$ and $s_\gamma > s$ accounts for the additional stage complexity associated with the penalty (5.46), and is $\mathcal{O}(N^3)$ for sufficiently large N.

Semi-Smooth Transformation Method

Another strategy for handling inequality constraints is to transform the complementarity conditions (5.13)-(5.15) into an equality condition using the Fischer-Burmeister function, [25],

$$\Phi_{FB}(a,b) = a + b - \sqrt{a^2 + b^2}. \tag{5.47}$$

The complementarity conditions are satisfied if and only if the following condition using the Fischer-Burmeister function is satisfied,

$$\Phi_{FB}(\lambda, -h) = 0, \tag{5.48}$$

where Φ_{FB} is applied element-wise. The discretized necessary conditions become

$$F(\hat{U}, x(t)) = \begin{bmatrix} l_u(x_0,u_0) + f_u^T(x_0,u_0)p_1 + g_u^T(x_0,u_0)\mu_0 + h_u^T(x_0,u_0)\lambda_0 \\ g(x_0,u_0) \\ \Phi_{FB}(\lambda_0, -h(x_0,u_0)) \\ \vdots \\ l_u(x_{N-1},u_{N-1}) + f_u^T(x_{N-1},u_{N-1})p_N + g_u^T(x_{N-1},u_{N-1})\mu_{N-1} \\ \qquad + h_u^T(x_{N-1},u_{N-1})\lambda_{N-1} \\ g(x_{N-1},u_{N-1}) \\ \Phi_{FB}(\lambda_{N-1}, -h(x_{N-1},u_{N-1})) \end{bmatrix} = 0 \in \mathbb{R}^{N(n_u+n_g+n_h)}, \tag{5.49}$$

where

$$\tilde{U} = \begin{bmatrix} u_0^T & \mu_0^T & \lambda_0^T & \dots & u_{N-1}^T & \mu_{N-1}^T & \lambda_{N-1}^T \end{bmatrix}^T \in \mathbb{R}^{N(n_u+n_g+n_h)}. \tag{5.50}$$

Like the auxiliary variable method, additional Lagrange multipliers must be added to U, however, there are no auxiliary variables, η. This leads to a smaller optimization problem compared to the auxiliary variable method. Note that the semi-smooth transformation provides an equivalent optimization problem to the original inequality constrained optimization problem while the exterior and auxiliary variable methods do not. The complexity of FD-GMRES using the semi-smooth transformation method is $o(k_{max}^3 + k_{max}^2 + (k_{max}+1)N\hat{s})$, where $1 \leq k_{max} \leq N(n_u + n_g + n_h)$ and $\hat{s} > s$ accounts for the additional stage complexity associated with the additional constraint (5.48), and is $\mathcal{O}(N^3)$ for sufficiently large N.

Remark: The Fischer-Burmeister function is Lipschitz continuous though non-differentiable at $(0,0)$ and formally a semi-smooth Newton method should be used, [51]. This requires the B-differential of a function.

Let $\Phi : \mathbb{R}^n \to \mathbb{R}^m$ be locally Lipschitz continuous at $z \in \mathbb{R}^n$ and S_Φ be the set of all points where Φ is differentiable.

Definition 5.1: The set

$$\partial_B \Phi(z) = \left\{ J \in \mathbb{R}^{m \times n} | \exists \{z^k\} \subset S_\Phi : \{z^k\} \to z, \Phi_z(z^k) \to J \right\} \tag{5.51}$$

is the B-differential of Φ at z, where Φ_z denotes the Jacobian of Φ, see [51].

The generalization of Newton's method for semi-smooth functions results in the following iteration,

$$J_k \Delta z_k = -\Phi(z_k), \tag{5.52}$$

$$z_{k+1} = z_k + \Delta z_k, \tag{5.53}$$

for some $J_k \in \partial_B \Phi(z_k)$. For the Fischer-Burmeister function, $\Phi_{FB}([a\, b]^T)$, the matrix $J \in \partial_B \Phi_{FB}([a\, b]^T)$ satisfies

$$J = \begin{cases} [\alpha\, \beta], & \text{if } a = b = 0, \\ \left[\begin{array}{cc} 1 - \frac{a}{\sqrt{a^2+b^2}} & 1 - \frac{b}{\sqrt{a^2+b^2}} \end{array} \right], & \text{else,} \end{cases} \tag{5.54}$$

with some α and β such that $(\alpha - 1)^2 + (\beta - 1)^2 = 1$, [51].

Also note that when the exterior penalty method is used, the penalty function

$$\gamma_j(z) = \begin{cases} 0, & h_j(z) \leq 0, \\ \frac{1}{2} r_j h_j(z)^2 & h_j(z) > 0, \end{cases} \quad j \in \{1, ..., n_h\}, \tag{5.55}$$

is only once continuously differentiable. It is assumed that $h(z)$ is twice continuously differentiable. Since $\gamma_{j,z}(z)$, denoting the partial differential of $\gamma_j(z)$ with respect to z, will appear

in the necessary conditions, the B-differential of $\gamma_{j,z}(z)$ is needed,

$$\partial_B \gamma_{j,z}(z) = \begin{cases} 0, & \text{if } h_j(z) < 0, \\ \left\{0, r_j \frac{\partial^2 h_j^2(z)}{\partial z^2}\right\}, & \text{if } h_j(z) = 0, \\ r_j \frac{\partial^2 h_j^2(z)}{\partial z^2}, & \text{if } h_j(z) > 0. \end{cases} \cdot \tag{5.56}$$

While formally, attention should be paid to the non-differentiability of the Fischer-Burmeister function at $(0, 0)$ or the non-differentiability of exterior penalty functions at $h = 0$, it is found that, in practice, when gradients and Jacobians are computed numerically, e.g., through forward differences, no special handling of these cases needs to be done. Specifically, the B-differential is not used with FD-GMRES but will be used when other algorithms are considered where the Jacobian is computed analytically, see Section 5.3.

5.2.2 Nonlinear DAP Model

To reduce NMPC computational complexity, a simple model of engine response is necessary. Towards this end, a data driven modeling approach was pursued in this work, similar to [27,35]. A reduced order physics based model, [53], was also investigated, however resulted in a poor fit to data. This could be due to a variety of reasons: a lack of sensors for obtaining modeling data, e.g., the EGR flow and cylinder flow data are obtained from ECU estimates rather than measurements, and the throttle pressure ratio stays consistently near 1 where the orifice flow equations are very sensitive to measurement errors. In this section, a continuous time data driven model is identified in order to directly use the C/GMRES NMPC formulation of [82]. The data driven model utilized in this section relies purely on the measured inputs (throttle, EGR valve, and VGT positions, engine speed, and fuel flow) and represents the response of measured outputs (MAP and MAF). Furthermore, the data driven model is simple to obtain, and has a simple structure conducive to NMPC implementation.

The data driven model takes a polynomial form,

$$\dot{x} = \theta^T f^l(x, u), \tag{5.57}$$

where x is a vector of measured outputs, i.e., $x = y$, u is a vector of inputs, θ is a matrix of coefficients to be estimated, and $f^l(x, u)$ is a vector of polynomial terms up to order l. For example, with $x \in \mathbb{R}^1$, $u \in \mathbb{R}^1$, and $l = 2$,

$$f^2(x, u) = [\ 1 \quad x \quad u \quad xu \quad x^2 \quad u^2\ ]^T. \tag{5.58}$$

For the diesel air path model in this section, the outputs/states are MAP and MAF, $x = [p_{in}\, W_c]^T$, and the inputs are EGR throttle [% closed], EGR valve [% open], VGT [% closed], engine speed [rpm], and fuel flow [mm3/st.], $u = [u_{th}\, u_{EGR}\, u_{VGT}\, N_e\, W_f]^T$. A polynomial order of $l = 2$ is used in the following. This leads to a total of 72 coefficients with 36 per state. For modeling, it is assumed that the states, x, and inputs, u, are measured. If the state derivatives, $\dot{x}$, are directly measured, then all information is available to estimate the coefficients θ, e.g., through linear least squares. In the case where $\dot{x}$ is not measured, it can be approximated by a filtered derivative. Specifically, applying $1/(s + \tau)$ to both sides of the polynomial model (5.57), the equation,

$$\left\{\frac{s}{s+\tau}\right\} x = \theta^T \left\{\frac{1}{s+\tau}\right\} f^l(x, u), \tag{5.59}$$

is obtained. Then the left had side of (5.59) can be generated through measurements of x and performing the operation $z = \{1/(s + \tau)\}\, x$. This yields a parametric model to which linear least squares can be applied to estimate θ. For the diesel air path application, the parametric model is,

$$z_1 = \theta_1^T \phi^r,\ z_1 = \left\{\frac{s}{s+\tau}\right\} p_{in}, \tag{5.60}$$

$$z_2 = \theta_2^T \phi^r,\ z_2 = \left\{ \frac{s}{s+\tau} \right\} W_c, \tag{5.61}$$

$$\phi^r = \left\{ \frac{1}{s+\tau} \right\} f^2 \left(\begin{bmatrix} p_{in} & W_c \end{bmatrix}^T, \begin{bmatrix} u_{th} & u_{EGR} & u_{VGT} & N_e & W_f \end{bmatrix}^T \right), \tag{5.62}$$

where $\theta \in \mathbb{R}^{36\times 2}$ and θ_i denotes the i-th column of θ. As a first pass plausibility check, the coefficients, θ, are estimated using least squares with measurements, z, and regressors, ϕ^r, generated from NEDC data obtained from a high fidelity physics based model. Note that this black box polynomial model is used for preliminary NMPC design applied to the DAP and evaluation of the various constraint handling techniques described in Section 5.2.1. This model will be further refined in Section 5.6 to target more specific control objectives.

The training data set for model identification is generated using the full NEDC and, afterward, repeating the high speed, high fuel region multiple times (final repetition of the Urban Drive Cycle (UDC) and Extra Urban Drive Cycle (EUDC)). Additional sinusoidal excitations were further superimposed onto the repeated cycle. Note that, in the future, optimal design of the input sequence, [35], should be considered. Figure 5.1 shows a simulation of the black box model on the extended NEDC versus the high fidelity model. The training and validation data sets are the same extended NEDC. The black box model is stable throughout the extended NEDC and there is good transient matching on the whole sequence.

Note that a polynomial model of order 1, i.e., a linear model, was also evaluated and found to be stable but provides a poor match to data. Polynomial orders greater than 2 have not yet been investigated, however, a polynomial order of 2 seems to be sufficient for control design.

Because the model is of black box type, one can question what the model actually learned in the extended NEDC. To check that the black box model behaves reasonably, step responses to the various actuators were simulated at various operating conditions. Figure 5.2 shows step responses to VGT opening between 0 and 5 sec, throttle opening between 10 and 15 sec, valve closing between 20 and 25 sec, fuel rate increase between 30 and 35 sec, and

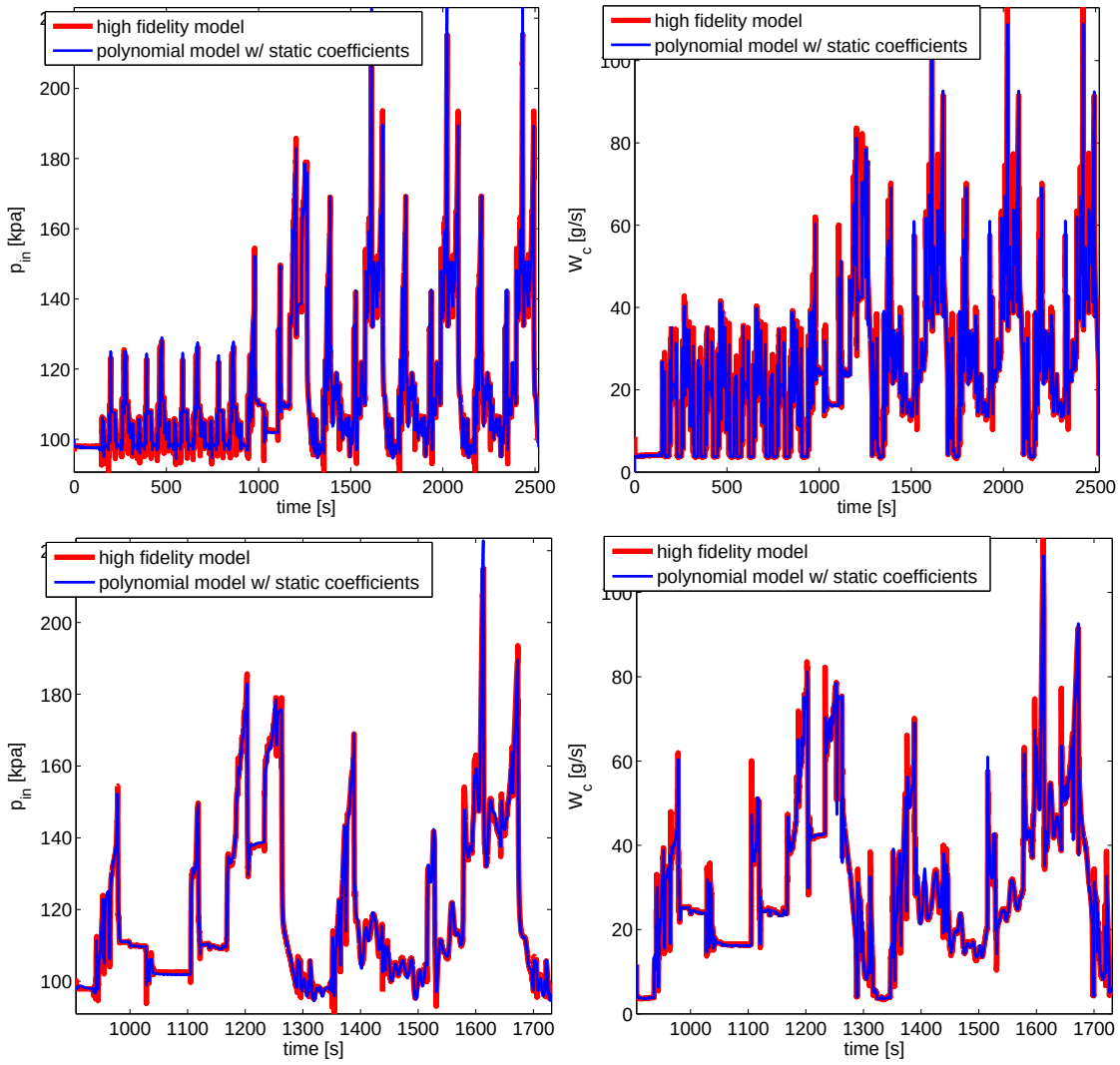

Figure 5.1: Simulation of the black box model versus high fidelity physics based model on the NEDC extended with additional excitations. Intake pressure (left), compressor flow (right), and zoomed views (bottom).

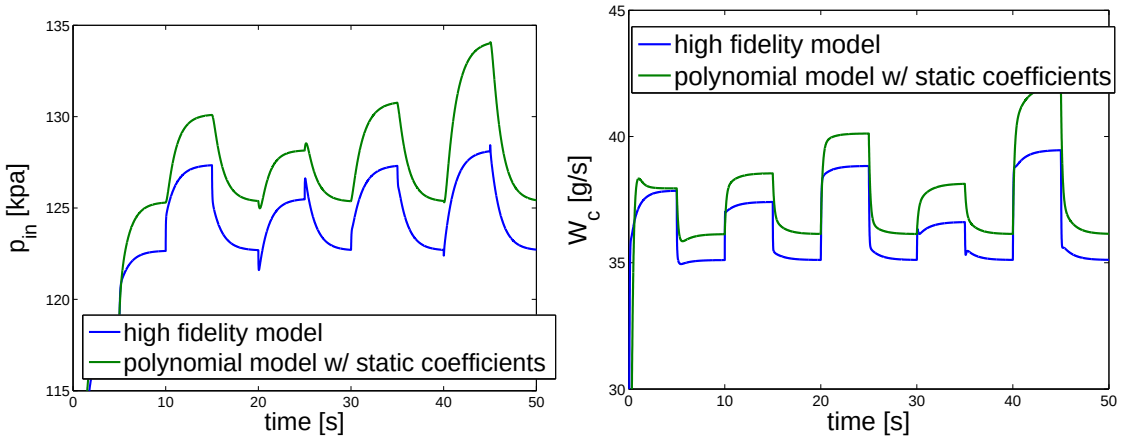

Figure 5.2: Step responses of intake pressure (left) and compressor flow (right) to VGT opening, valve closing, fuel rate increase, and engine speed increase comparing the high fidelity model and polynomial model.

engine speed increase between 40 and 45 sec. The steps shown were conducted around an operating condition of 2500 rpm. The figure shows the intake pressure and compressor flow responses using the coefficients learned during the extended NEDC. For both intake pressure and compressor flow, the sign of the DC gain is correct compared to the high fidelity model and the response size looks reasonable. The black box model was also able to learn that there is a non-minimum phase behavior from the EGR valve to intake pressure. This is physically correct. When the valve closes, flow into the intake manifold immediately decreases. Intake pressure then rises because closing the EGR valve increases exhaust pressure to spin up the turbocharger and, in the long term, provides greater boost, see [38, 113].

To achieve DC gain matching, which will be important for achieving zero-offset set-point tracking, the affine term in the polynomial model can be estimated online. This can be viewed as an additive disturbance estimator and is done through rearranging the parametric model,

$$\tilde{z}_1 = \theta_{1,1}^T \phi_1^r, \; \tilde{z}_1 = \left\{ \frac{s}{s+\tau} \right\} p_{in} - \theta_{1,2:36}^T \phi_{2:36}^r, \tag{5.63}$$

$$\tilde{z}_2 = \theta_{2,1}^T \phi_1^r, \; \tilde{z}_2 = \left\{ \frac{s}{s+\tau} \right\} W_c - \theta_{2,2:36}^T \phi_{2:36}^r, \tag{5.64}$$

where ϕ^r in (5.63) and (5.64) is the same as in (5.62), $\theta_{i,1}$ denotes the first element in the

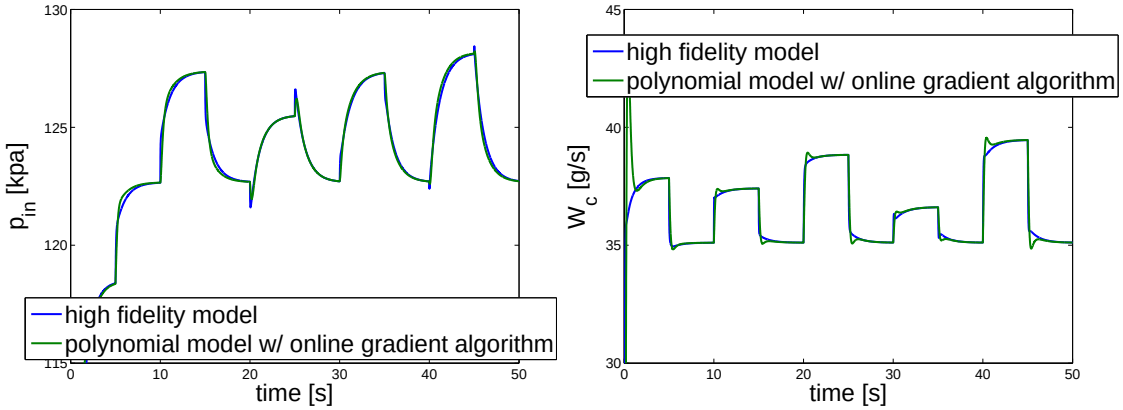

Figure 5.3: Step responses with gradient algorithm of intake pressure (left) and compressor flow (right) to VGT opening, valve closing, fuel rate increase, and engine speed increase comparing the high fidelity model and polynomial model.

i-th column of θ, and $\theta_{i,2:36}$ denotes the remaining elements of the i-th column of θ. The affine terms are then estimated online through a gradient algorithm, [50],

$$\dot{\tilde{\theta}}_{i,1} = \Gamma_i \epsilon_i \phi_1^r, \; \epsilon_i = \tilde{z}_i - \tilde{\theta}_{i,1}^T \phi_1^r, \tag{5.65}$$

where $\Gamma_i \geq 0$ is the adaptation gain and $\tilde{\theta}_{i,1}$ is the current estimate of the i-th affine term, $i \in \{1, 2\}$.

Figure 5.3 shows the same step responses as shown in Figure 5.2 but with online estimation of the affine terms with a gradient algorithm. With online estimation of the affine terms both the DC gains for intake pressure and compressor flow are matched.

5.2.3 DAP Simulation Results and Computational Comparison

The cost functional for the diesel air path application is as follows,

$$\begin{aligned} \mathcal{J} = \int_0^T & r_1(u_{VGT}(\tau,t) - u_{VGT}(t))^2 + r_2(u_{EGR}(\tau,t) - u_{EGR}(t))^2 \\ & + q_1(p_{in}(\tau,t) - r_{pin}(t))^2 + q_2(W_c(\tau,t) - r_{Wc}(t))^2 \\ & + \gamma_{pin}(p_{in}(\tau,t)) \\ & d\tau, \end{aligned} \tag{5.66}$$

where

$$\gamma_{pin}(p_{in}(\tau,t)) = \begin{cases} 0, & \text{if } p_{in}(\tau,t) \leq \bar{p}_{in}, \\ r_3(p_{in}(\tau,t) - \bar{p}_{in})^2, & \text{else.} \end{cases} \tag{5.67}$$

The cost functional (5.66) is minimized with respect to u_{VGT} and u_{EGR} subject to constraints,

$$x_\tau(\tau,t) = \theta^T f^2(x(\tau,t), u(\tau,t)), \tag{5.68}$$

$$h(u_{VGT}(\tau,t), u_{EGR}(\tau,t)) \leq 0, \tag{5.69}$$

where the inequality constraint, defined by the function h in (5.69) are specific hard constraints corresponding to the more general form in (5.6),

$$h(u_{VGT}(\tau,t), u_{EGR}(\tau,t)) = \begin{bmatrix} u_{VGT}(\tau,t) - \bar{u}_{VGT} \\ \underline{u}_{VGT} - u_{VGT}(\tau,t) \\ u_{EGR}(\tau,t) - \bar{u}_{EGR} \\ \underline{u}_{EGR} - u_{EGR}(\tau,t) \end{bmatrix}, \tag{5.70}$$

and where $x = [p_{in}\, W_c]^T$ and $u = [u_{th}\, u_{EGR}\, u_{VGT}\, N_e\, W_f]^T$. The measured disturbances, u_{th}, N_e, and W_f are treated as constant over the prediction horizon, as are the references, r_{pin} and r_{Wc}. The terms $u_{VGT}(t)$ and $u_{EGR}(t)$ in (5.66) are the values of the controls applied at the current sample time before the NMPC feedback is calculated and applied. A soft constraint is placed on MAP to ensure feasibility. Note that an exterior penalty is used for the MAP constraint but different strategies are considered for handling the control inequality constraints (5.69). This is done to prevent infeasibility in the presence of plant/model mismatch and disturbances.

Using $u_{VGT}(t)$ and $u_{EGR}(t)$ rather than the typical steady states, $u_{VGT,ss}$ and $u_{EGR,ss}$, associated with the references gives an NMPC problem formulation is similar to the input-velocity form used in linear MPC, [6]. This gives zero-offset steady-state tracking if the model

Table 5.1: Comparison of tracking performance, constraint handling, and worst-case computation time for various methods over the NEDC.

Method	e_{pin}	e_{Wc}	v_{pin}	v_{VGT}	v_{EGR}	N_f	ECU [ms]
fsolve	335.9	347.7	61.27	3.3e-6	4.46e-6	N/A	N/A
SS-GMRES	333.6	340.2	62.03	1.55	0.46	21	6.1
AV-GMRES, $r = 0.1$	3224	1863	24.21	0,47	0	31	9.0
AV-GMRES, $r = 0.01$	425.7	454.3	47.27	1.10	0.02	31	9.0
AV-C/GMRES, $r = 0.01$	426.5	453.7	48.41	1.19	0.01	32	9.3
EP-GMRES	333.5	340.1	61.62	0.40	0.12	11	3.2
EP-GMRES 2	334.0	341.1	61.87	0.39	0.11	3	0.9
EP-GMRES 1	427.4	422.5	66.48	0.27	0.10	2	0.6
EP-C/GMRES	374.6	394.2	82.0	0.38	0.16	12	3.8

matches the plant exactly at steady state, which is obtained through parameter estimation (5.65).

The resulting controller is run in the loop with the high fidelity nonlinear model on the NEDC. The VGT position constraint set is [40 90] % closed. The EGR valve position constraint set is [0 55] % open. The control constraints are tighter than typically used to demonstrate constraint activation. The maximum MAP constraint is applied with the upper bound of $\bar{p}_{in} = 170$kPa. The integration step size for the state and co-state equations is 32 msec. A control update is made every 64 msec. The prediction horizon is 320 msec (i.e., 5 control steps). The sampling period is 32 msec. The prediction horizon is chosen as the shortest horizon that results in a stable and not under-damped response over the NEDC. The gradient algorithm (5.65) for estimation of the affine terms in the polynomial model uses an adaptation gain, Γ, of 4 corresponding to a time constant of 0.25 sec for the adaptation rate. In all of the following simulations presented in this chapter, the VGT and EGR valve operates purely through feedback provided by the NMPC and no feed-forward is used.

Figure 5.4 shows a comparison between using Newton's method directly to solve $F = 0$ and when continuation is used, i.e., the predictor part using F_x is added. In the continuation strategy, the stabilization matrix, A_s, has been set such that without the predictor part, Newton's method is exactly recovered. In both cases, the exterior penalty method is used

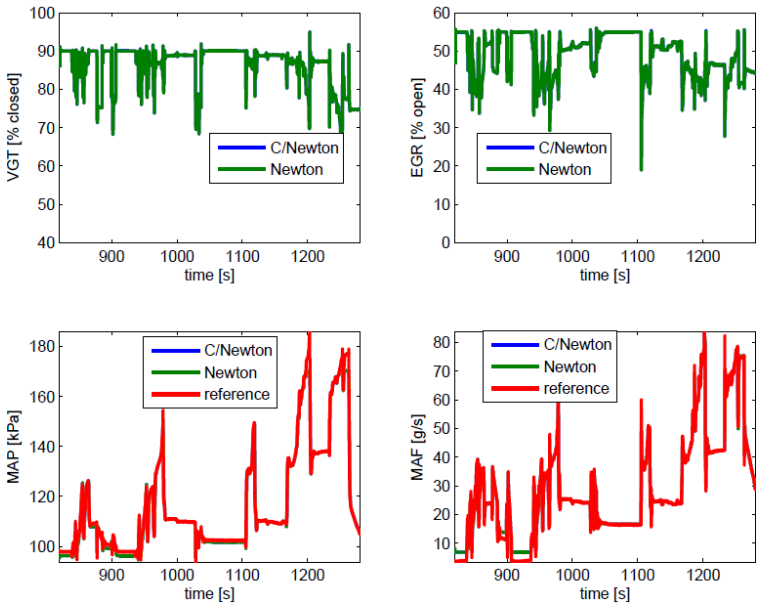

Figure 5.4: Closed loop response comparison for NMPC between Newton's method and Newton's method with continuation (exterior penalty).

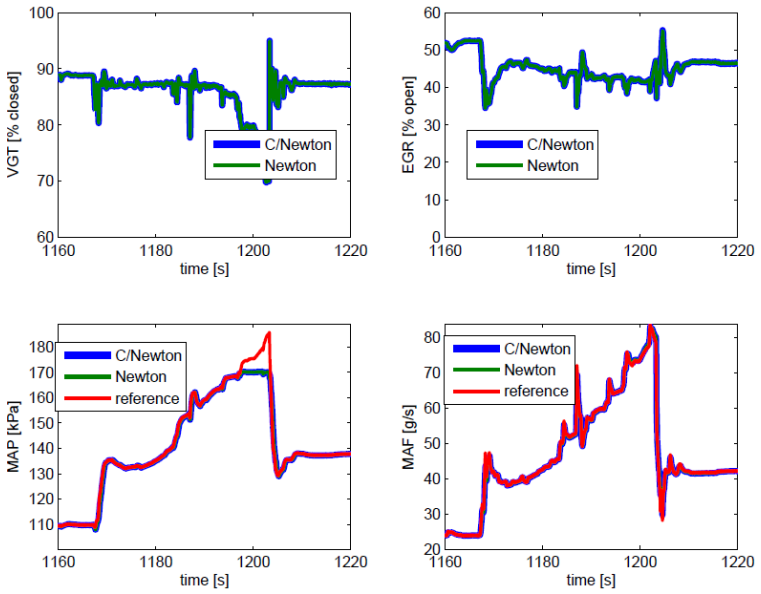

Figure 5.5: Zoomed view of closed loop response comparison for NMPC between Newton's method and Newton's method with continuation (exterior penalty).

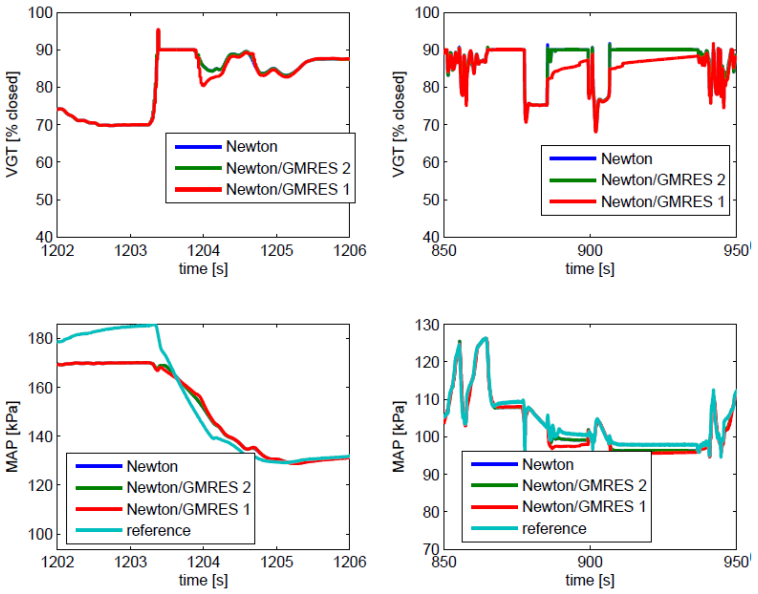

Figure 5.6: Closed loop response comparison for NMPC between Newton's method and inexact Newton's method using FD-GMRES with 2 and 1 inner iterations (exterior penalty).

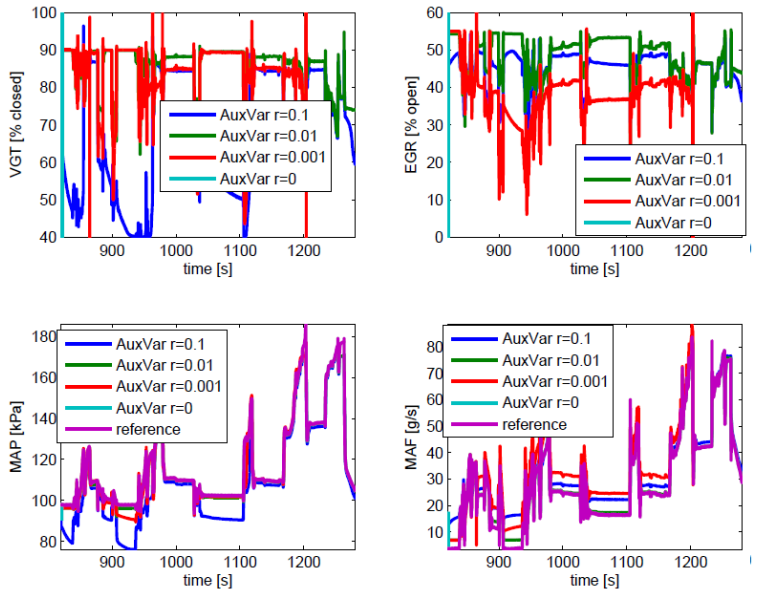

Figure 5.7: Closed loop response comparison for NMPC using the auxiliary variable method for inequality constraint handling with different interior weights, r.

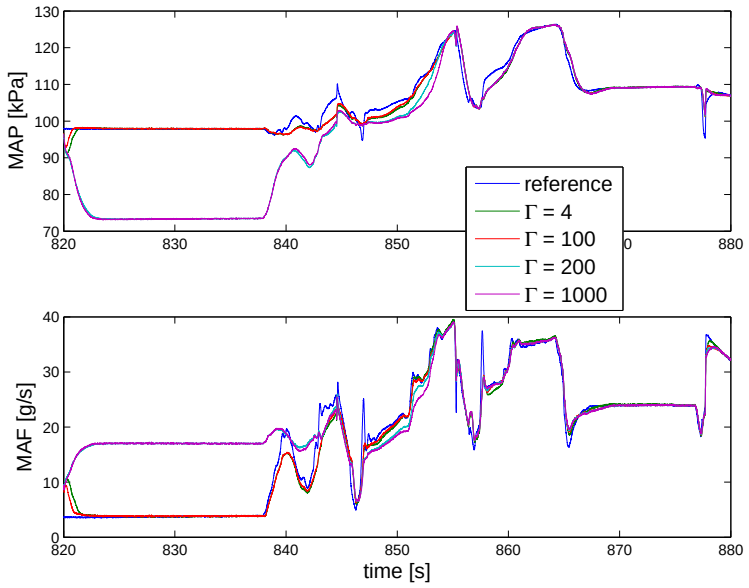

Figure 5.8: Comparison of NMPC controllers on the NEDC with different adaptation rates, Γ.

with only a single Newton step per time step. The solution from the previous time step is used as the initial condition for the Newton step at the current time. Tracking performance is very good except when the actuators are saturated, e.g., between 900 sec and 930 sec. Figure 5.5 shows a zoomed view of Figure 5.4. MAP does not increase past 170kPa due to the maximum MAP constraint. There is little difference between Newton's method and the continuation strategy, i.e., the predictor portion contributes little. In both cases there is significant violation of the maximum VGT constraint for a single step at 1203.2 sec. This is due to a rapid change in the parameters due to both the fuel cut and change in references at that time. This violation is tolerable since recovery to the constraint only takes a few time steps and a simple saturation can be used in the interim. Because a continuation strategy does not appear to offer much performance improvement compared to the direct application of Newton's method, subsequent investigations shown in this section will not use continuation. Note that continuation can be added to any strategy discussed.

Figure 5.6 compares the results when using inexact Newton with different numbers of FD-GMRES inner iterations, i.e., different k_{max}. The difference between using $k_{max} = 2$ and

a full Newton step, $k_{max} = 10$, is minimal. With $k_{max} = 1$, FD-GMRES 1, performance dramatically suffers, particularly near idle conditions. See Table 5.1 for a quantitative comparison of performance and computation time.

Figure 5.7 shows the controller performance using the auxiliary variable method with different interior weights. The performance is highly sensitive to the interior penalty weight, where using a penalty weight of $r = 0.01$ most closely resembles the exterior penalty method. Using $r = 0.1$ dramatically changes the steady state tracking performance. Using $r = 0.001$ or $r = 0$ leads to ill-conditioning of the Jacobian. With interior point methods (and exterior penalty methods) the penalty weight is typically decreased (increased with exterior penalty) over multiple Newton iterations to ensure convergence. However, due to the highly limited DAP NMPC computational allotment, it is not desirable to perform multiple Newton steps per sample time. Due to issues related to choice of the interior penalty, the use of the auxiliary variable method does not appear to be straight-forward for this application.

The semi-smooth method performs qualitatively similar to the exterior penalty method. Thus simulation figures are not shown. The results in Table 5.1 include a quantitative assessment of the semi-smooth method.

Table 5.1 summarizes the tracking performance, constraint handling, and computation time of the various methods. MATLAB's fsolve is representative of the nominal performance. SS refers to semi-smooth, AV refers to auxiliary variable with interior weight r, and AV-C/GMRES refers to continuation with FD-GMRES and auxiliary variables as in [82] and with FD-GMRES run to completion. EP refers to exterior penalty, FD-GMRES 2 refers to the use of $k_{max} = 2$ with FD-GMRES. The integrated normed tracking errors are denoted by e_{pin} and e_{Wc}. The integrated normed constraint violation for MAP, VGT position, and EGR position are denoted by v_{pin}, v_{VGT}, and v_{EGR}, respectively. The number of times F needs to be computed for forward difference calculations is denoted by N_F. The ECU computation time is estimated through explicitly counting the number of operations inside the generated code, [112], e.g., additions, multiplications, and divides, and mid-range, e.g.,

160MHz, ECU specifications.[3] Note that the computation of F is dominated by the need to propagate forward and backward the state and co-state equation. Thus the model structure and integration step size are important with respect to computations.

Generally, all methods perform similarly except FD-GMRES 1 and the AV methods. As expected from Figure 5.6, FD-GMRES 1 has larger tracking error than other methods because FD-GMRES does not utilize enough iterations to accurately solve for the Newton step in (5.37). Also, as expected from the AV simulations, tracking performance suffers greatly due to the interior penalty. Since the control and state trajectories with the AV method are far off from nominal, the constraint violation statistics carry little value. Despite the poor performance of the AV method, it is still an important case for comparison as it is the strategy employed in recent publications, [27], which is considered the benchmark strategy. From the simulation, EP-GMRES is currently the best option with both a low computation time and good ability to track references and enforce constraints over the NEDC.

Figure 5.8 shows a comparison of NMPC controllers on the NEDC with different adaptation rates, Γ. The time constant associated with the adaptation rate is $1/\Gamma$ sec. Note that the controller update rate is 32 msec. Figure 5.8 shows that if the adaptation rate is too large, e.g., with $\Gamma = 200$ and $\Gamma = 1000$, the controller is not able to track the reference at the beginning of the simulation (820 sec through 845 sec). However, reference tracking is recovered in these cases after 845 sec. This indicates that the adaptation, specifically the adaptation of an additive disturbance term, can be made arbitrarily fast if the adaptive model is sufficiently close to the nominal model. However, if the states, e.g., at the initial condition, are inside a poor extrapolation region of the nominal model, then the the adaptive model can enter the poor extrapolation region if the adaptation rate is too fast, e.g., before the initial dynamics are able to settle out. This can lead to instability and a loss of tracking. Note that with linear MPC, deadbeat additive disturbance estimation/adaptation can be used and achieves the same control law as rate-based linear MPC, [87]. However, the

[3] With the NMPC strategies considered in this chapter there is only one fixed path for code execution.

conditions for which deadbeat additive disturbance estimation/adaptation can be used with NMPC requires further investigation.

5.3 Multiple-Shooting versus Single-shooting

The NMPC formulation that is presented in the beginning of Section 5.2 is considered a "single-shooting" method, [20]. By single-shooting, it is meant that the states and co-states, $x_0, \ldots, x_N$, and $p_N, \ldots, p_1$, in the discretized necessary conditions (5.36) implicitly satisfy (5.21), (5.22), (5.24), and (5.33), i.e., the state and co-state equations are propagated in one "shot."

Another approach is to utilize a "multiple-shooting" method, [20]. With multiple-shooting $x_1, \ldots, x_N$ (and x_0 depending on the choice of problem setup) and $p_N, \ldots, p_1$ are treated as independent variables, and the conditions (5.21), (5.24), and (5.33) are included explicitly in the discretized necessary conditions. With multiple-shooting and only equality constraints, the discretized necessary conditions become,

$$\bar{F}(\bar{U}, x(t)) = \begin{bmatrix} \Delta\tau(l_u(x_0,u_0) + f_u^T(x_0,u_0)p_1 + g_u^T(x_0,u_0)\mu_0) \\ \Delta\tau g(x_0,u_0) \\ p_2 - p_1 + \Delta\tau(l_x(x_0,u_0) + f_x^T(x_0,u_0)p_1 + g_x^T(x_0,u_0)\mu_0) \\ x_0 - x_1 + \Delta\tau f(x_0,u_0) \\ \vdots \\ \Delta\tau(l_u(x_{N-1},u_{N-1}) + f_u^T(x_{N-1},u_{N-1})p_N + g_u^T(x_{N-1},u_{N-1})\mu_{N-1}) \\ \Delta\tau g(x_{N-1},u_{N-1}) \\ p_N - \phi_x(x_N) \\ x_{N-1} - x_N + \Delta\tau f(x_{N-1},u_{N-1}) \end{bmatrix} = 0 \in \mathbb{R}^{N(n_u+n_g+2n_x)}, \tag{5.71}$$

where

$$\bar{U} = \begin{bmatrix} u_0^T & \mu_0^T & x_1^T & p_1^T & ... & u_{N-1}^T & \mu_{N-1}^T & x_N^T & p_N^T \end{bmatrix}^T \in \mathbb{R}^{N(n_u+n_g+2n_x)} \tag{5.72}$$

and x_0 implicitly satisfies $x_0 = x(t)$. Note that $x_0 = x(t)$ can also be added to the conditions (5.71) and x_0 is added as an optimization variable to $\bar{U}$ in (5.72). The multiple shooting method leads to a larger optimization problem, however, in this form, the Jacobian of $\bar{F}$, $F_{\bar{U}}$, can be computed analytically and easily. Note that the conditions $H_u = 0$ and $g = 0$ in (5.36) have been changed to the equivalent conditions $\Delta\tau H_u = 0$ and $\Delta\tau g = 0$ in (5.71). This will lead to a symmetric Jacobian and facilitate the use of algorithms such as the well-known Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm if so desired.

Let $H_{uu,i} = H_{uu}(x_i, u_i, p_{i+1}, \mu_i)$, $H_{xx,i} = H_{xx}(x_i, u_i, p_{i+1}, \mu_i)$, $H_{ux,i} = H_{ux}(x_i, u_i, p_{i+1}, \mu_i)$, $g_{u,i} = g_u(x_i, u_i)$, $g_{x,i} = g_x(x_i, u_i)$ and $f_{u,i} = f_u(x_i, u_i)$. The Jacobain, $F_{\bar{U}}$, has the following form,

$$A_i = \begin{bmatrix} \Delta\tau H_{uu,i} & \Delta\tau g_{u,i}^T & 0 & \Delta\tau f_{u,i}^T \\ \Delta\tau g_{u,i} & 0 & 0 & 0 \\ 0 & 0 & \Delta\tau H_{xx,i+1} & -I \\ \Delta\tau f_{u,i} & 0 & -I & 0 \end{bmatrix}, \; i \in \{0, ..., N-2\}, \tag{5.73}$$

$$A_{N-1} = \begin{bmatrix} \Delta\tau H_{uu,N-1} & \Delta\tau g_{u,N-1}^T & 0 & \Delta\tau f_{u,N-1}^T \\ \Delta\tau g_{u,N-1} & 0 & 0 & 0 \\ 0 & 0 & \phi_{xx}(x_N) & -I \\ \Delta\tau f_{u,N-1} & 0 & -I & 0 \end{bmatrix}, \tag{5.74}$$

$$B_i = \begin{bmatrix} 0 & 0 & \Delta\tau H_{ux,i} & 0 \\ 0 & 0 & \Delta\tau g_{x,i} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 \end{bmatrix}, i \in \{1, ..., N-1\}, \tag{5.75}$$

$$\bar{F}_{\bar{U}} = \begin{bmatrix} A_0 & B_1^T & & & & \\ B_1 & A_1 & B_{2^T} & & & \\ & B_2 & \ddots & & & \\ & & & \ddots & B_{N-2}^T & \\ & & & B_{N-2} & A_{N-2} & B_{N-1}^T \\ & & & & B_{N-1} & A_{N-1} \end{bmatrix}, \tag{5.76}$$

and the Newton step is obtained from

$$\bar{F}_{\bar{U}}(\bar{U}_k, x(t))\Delta\bar{U}_k = -\bar{F}(\bar{U}_k, x(t)). \tag{5.77}$$

The computational complexity of computing $\bar{F}$ in (5.71) is $\mathcal{O}(N)$ which is the same as computing F in (5.36). Let s_J denote the computational cost of computing A_i and B_i at stage i using (5.73)-(5.75). The total computational complexity of computing the Newton step, e.g., through GMRES or LU decomposition, is $\mathcal{O}(N^3)$ for sufficiently large N. For small problem sizes, e.g., with small N, the computational complexity of computing the Newton step is $o(N(s + s_J) + N^2(n_u + n_g + 2n_x)^2 + N^3(n_u + n_g + 2n_x)^3)$ where the term $N(s + s_J)$ is the complexity contribution from the computation of $\bar{F}$ and $\bar{F}_{\bar{U}}$ and the term $N^2(n_u + n_g + 2n_x)^2 + N^3(n_u + n_g + 2n_x)^3$ is from the linear solve. When N is small, the complexity contribution from computing $\bar{F}$ and $\bar{F}_{\bar{U}}$ can dominate the complexity of the linear solve. Figure 5.9 shows a comparison of the computational cost of evaluating $\bar{F}$ and $\bar{F}_{\bar{U}}$ and solving the linear equation (5.77) for different horizon lengths. In fact, solving the linear equation (5.77) is cheaper than evaluating the nonlinear equations composing $\bar{F}$ and $\bar{F}_{\bar{U}}$ due to the sparse structure of $\bar{F}_{\bar{U}}$. The estimated computation time for a mid-range, 160MHz

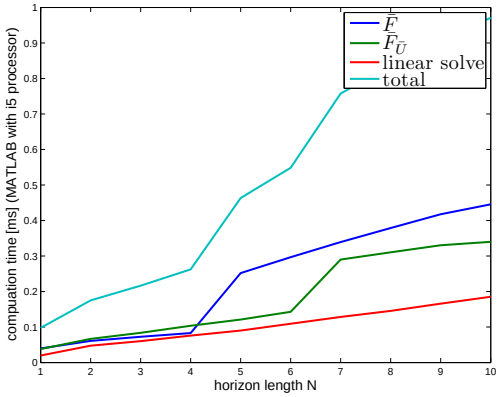

Figure 5.9: Comparison of computation times of $\bar{F}$, $\bar{F}_{\bar{U}}$, and the linear solve in (5.77) for different horizon lengths using MATLAB on an i5 processor. Note: the implementation of multiple-shooting uses MATLAB's Symbolic Toolbox to compute $\bar{F}$ and $\bar{F}_{\bar{U}}$, the uses MATLAB's *matlabfunction* command to autogenerate executable MATLAB code. The sharp increase in computation between $N = 4$ and $N = 5$ for the computation of $\bar{F}$ and similarly for $\bar{F}_{\bar{U}}$ is likely due to bloating in the code generation process.

ECU with the same horizon as in Table 5.1, $N = 5$, is 0.55 msec (compared to 0.9 msec with EP-GMRES 2 in Table 5.1).

5.4 Direct NMPC versus Indirect NMPC

The NMPC formulation that is presented in the beginning of Section 5.2 is considered "indirect NMPC," [20]. By indirect NMPC, it is meant that the necessary conditions, (5.9)-(5.20), are first formed for the continuous time system and continuous time optimization problem. The necessary conditions are then discretized leading to (5.21)-(5.32), which can then be solved using some form of Newton's method.

An alternative approach, "direct NMPC," [20], to NMPC is to first discretize the continuous time system dynamics, then form the root finding problem through deriving the Karush-Kuhn-Tucker (KKT) necessary conditions.

This section shows that the root finding problem is the same for indirect and direct NMPC when Euler discretization is used, as is the case in this work.

Consider the Euler discretized system dynamics, where $f(x, u)$ are the continuous time

dynamics,

$$x_{k+1} = x_k + \Delta\tau f(x_k, u_k). \tag{5.78}$$

The discrete time NMPC problem is based on minimizing a cost functional of the form,

$$J = \phi(x_{N|k}) + \sum_{i=0}^{N-1} \Delta\tau l(x_{i|k}, u_{i|k}), \tag{5.79}$$

The minimization of (5.79) is performed subject to equality and inequality constraints,

$$x_{i+1|k} = x_{i|k} + \Delta\tau f(x_{ik}, u_{i|k}), \tag{5.80}$$

$$x_{0|k} = x_k, \tag{5.81}$$

$$g(x_{i|k}, u_{i|k}) = 0, \tag{5.82}$$

$$h(x_{i|k}, u_{i|k}) \leq 0, \tag{5.83}$$

$$\psi(x_{N|k}) \leq 0, \tag{5.84}$$

where k is the current discrete sampling time, and i is the running time over the prediction horizon. To simplify notation for this discussion, let $k = 0$ be the current time step and drop the subscripts $|k$. Let $\mathcal{L}$ denote the Lagrangian,

$$\begin{aligned} \mathcal{L} = \phi(x_N) + \textstyle\sum_{i=0}^{N-1} \Delta\tau l(x_i, u_i) + p_{i+1}^T(x_i - x_{i+1} + \Delta\tau f(x_i, u_i)) \\ + \mu_i^T g(x_i, u_i) + \lambda_i^T h(x_i, u_i), \end{aligned} \tag{5.85}$$

The necessary conditions for optimality through KKT are

$$x_{i+1} = x_i + \Delta\tau f(x_i, u_i),\ i \in \{0, ..., N-1\}, \tag{5.86}$$

$$\mathcal{L}_{x_i} = \Delta\tau l_x(x_i, u_i) + \Delta\tau f_x^T(x_i, u_i)p_{i+1} - p_i + g_x^T(x_i, u_i)\mu_i + h_x^T(x_i, u_i)\lambda_i = 0, \quad i \in \{0, ...N-1\}, \tag{5.87}$$

$$p_N - \phi_x(x_N) - \psi_x(x_N)^T v = 0, \tag{5.88}$$

$$\mathcal{L}_{u_i} = \Delta\tau l_u(x_i, u_i) + \Delta\tau f_u^T(x_i, u_i)p_{i+1} + g_u^T(x_{i|k}, u_{i|k})\mu_{i|k} + h_u^T(x_i, u_i)\lambda_i = 0, \quad i \in \{0, ...N-1\}, \tag{5.89}$$

$$h(x_i, u_i) \leq 0,\ \lambda_i \geq 0,\ \lambda_i^T h(x_i, u_i) = 0,\ i \in \{0, ..., N-1\}, \tag{5.90}$$

$$\psi(x_N) \leq 0,\ v \geq 0,\ v^T\psi(x_N) = 0. \tag{5.91}$$

Note that $x_k = x(t)$ at sample times k and the equality and inequality constraints g and h can be scaled by $\Delta\tau$. Then the necessary conditions (5.86)-(5.91) for the direct approach are exactly the same as the discretized necessary conditions (5.9)-(5.20) for the indirect approach.

Remark: There is a difference between indirect and direct NMPC during the modeling stage. When direct NMPC is used, the system can be modeled in discrete time rather than relying on some discretization strategy. For example when data-driven modeling is used, as is the case in this work, a discrete time model rather than a continuous time model can be directly identified. This can be advantageous since move blocking is typically used to reduce the number of optimization variables, e.g., control decisions are only made every b steps and held constant between those steps. It may be possible to directly fit a discrete model, without changing the model structure, with a sampling period of $\Delta\tau b$. Then, the costly computations of propagating Euler discretized continuous time dynamics for b steps are forgone (and partial derivatives when an analytical Jacobian is used).

5.5 Kantorovich's Method

This section considers the application of Kantorovich's method, [51], to NMPC. The goal is to reduce the need to compute the Jacobian (of the discretized necessary conditions), thus reducing dramatically the computational burden of computing the Newton step. Recall that the Newton iteration solves

$$0 = F(z_k) + F_z(z_k)(z_{k+1} - z_k) \tag{5.92}$$

for x_{k+1}, where F is a continuous function from $\mathbb{R}^n$ to $\mathbb{R}^n$ and F_z is the Jacobian of F with respect to z. Commonly, an approximate Jacobian is used. One choice for an approximate Jacobian is to use the Jacobian at the first iteration, $F_x(z_0)$, and use this same Jacobian for all subsequent iterations. This is called Kantorovich's method. The Kantorovich iteration solves

$$0 = F(z_k) + F_z(z_0)(z_{k+1} - z_k) \tag{5.93}$$

for z_{k+1}. One unclear aspect of Kantorovich's method is how it can be applied to optimal control problems, specifically inequality constrained optimal control problems. With inequality constraints, the Jacobian can change dramatically when constraints become active. For example, consider any type of penalty method used for constraint handling. Typically, the gradient of the cost function approaches infinity and the Jacobian becomes near singular at the constraint boundaries. In such a case, using the Jacobian $F_z(z_0)$ may not be suitable.

The novel idea here is to freeze only a portion of the Jacobian, $F_z(z_k)$, at z_0, preferentially a portion that is expensive to calculate, and only update the remaining, preferentially cheap to compute, portion at every iteration k.

Here, the case when exterior penalty functions are used to enforce the inequality constraints is examined. Note that such an approach can used with other constraint enforcement methods, e.g., interior penalty, active set strategies, mixed complementarity solvers, and etc. The exterior penalty method is used here because it has already demonstrated good per-

formance in the DAP example with a very limited computational footprint, see Section 5.2, and provides a good benchmarking point.

A multiple-shooting framework will be used, see Section 5.3, for organizing the discretized necessary conditions. This will allow for easy handling of not only control constraints, but also state constraints and/or mixed constraints. The discretized necessary conditions with exterior penalty for the constraints $h(x,u) \leq 0$ can be written in the following form,

$$\bar{F}(\bar{U}, x(t)) + F^C(\bar{U}, x(t)) = 0, \tag{5.94}$$

$$F^C(\bar{U}, x(t)) = \begin{bmatrix} \Delta\tau\Gamma_u(x_0, u_0) \\ 0 \\ \Delta\tau\Gamma_x(x_1, u_1) \\ 0 \\ \vdots \\ \Delta\tau\Gamma_u(x_{N-1}, u_{N-1}) \\ 0 \\ 0 \\ 0 \end{bmatrix}, \tag{5.95}$$

where $\bar{F}(\bar{U}, x(t))$ in (5.94) is the same as in (5.71) and Γ in (5.95) is the exterior penalty function,

$$\begin{aligned} &\Gamma = \textstyle\sum_{j=1}^{n_h} \gamma_j(x,u), \\ &\gamma_j(x,u) = \begin{cases} 0, & h_j(x,u) \leq 0, \\ \frac{1}{2} r_j h_j(x,u)^2 & h_j(x,u) > 0. \end{cases} \end{aligned} \tag{5.96}$$

Note that the computation of $\bar{F}(\bar{U}, x(t))$ in (5.94) and the associated Jacobian, $\bar{F}_{\bar{U}}(\bar{U}, x(t))$, see (5.73)-(5.76), can be expensive since it requires propagating the nonlinear state and co-state equations. Instead of computing the Jacobian, $\bar{F}_{\bar{U}}(\bar{U}_k, x(t))$, at every Newton iteration, it can be frozen at $\bar{F}_{\bar{U}}(\bar{U}_0, x(0))$ as would be done in an unconstrained Kantorovich method.

Then, $F^C(\bar{U}, x(t))$ and the associated Jacobian, $F^C_{\bar{U}}(\bar{U}, x(t))$, is computed at every iteration of $\bar{U}$ since the local geometry of Γ can change very rapidly at during set changes. Assuming that the inequality constraints, $h(x,u) \leq 0$, have a simple form, e.g., a linear form, $F^C(\bar{U}, x(t))$ and $F^C_{\bar{U}}(\bar{U}, x(t))$ should be computationally simple to compute where

$$F^C_{\bar{U}}(\bar{U}, x(t)) = diag\left(\begin{bmatrix} \Delta\tau\Gamma_{uu}(x_0, u_0) \\ 0 \\ \Delta\tau\Gamma_{xx}(x_1, u_1) \\ 0 \\ \vdots \\ \Delta\tau\Gamma_{uu}(x_{N-1}, u_{N-1}) \\ 0 \\ 0 \\ 0 \end{bmatrix}\right), \tag{5.97}$$

and where $\Gamma_{uu}(x,u) \in \partial_{B,u}\left(\sum_{j=1}^{n_h} \gamma_{j,u}(x,u)\right)$, $\Gamma_{xx}(x,u) \in \partial_{B,x}\left(\sum_{j=1}^{n_h} \gamma_{j,x}(x,u)\right)$, and $\partial_{B,u}$ and $\partial_{B,x}$ refers to the B-differential of a function to u and x respectively and has the form of (5.56).

The Newton iteration associated with applying Kantorovich's method to the equation (5.94) is

$$\left(\bar{F}_{\bar{U}}(\bar{U}_0, x(0)) + F^C_{\bar{U}}(\bar{U}_k, x(t))\right)(\bar{U}_{k+1} - \bar{U}_k) = -\left(\bar{F}(\bar{U}_k, x(t)) + F^C(\bar{U}_k, x(t))\right). \tag{5.98}$$

Figures 5.10-5.12 show simulation results using Kantorovich's method (5.98) to solve the discretized necessary conditions (5.94). Only a single step of (5.98) is performed at each sampling time. The NMPC follows the model setup of Section 5.2.2, optimization problem setup of Section 5.2.3, and utilizes exterior penalty/soft constraints for all of the control and state constraints.

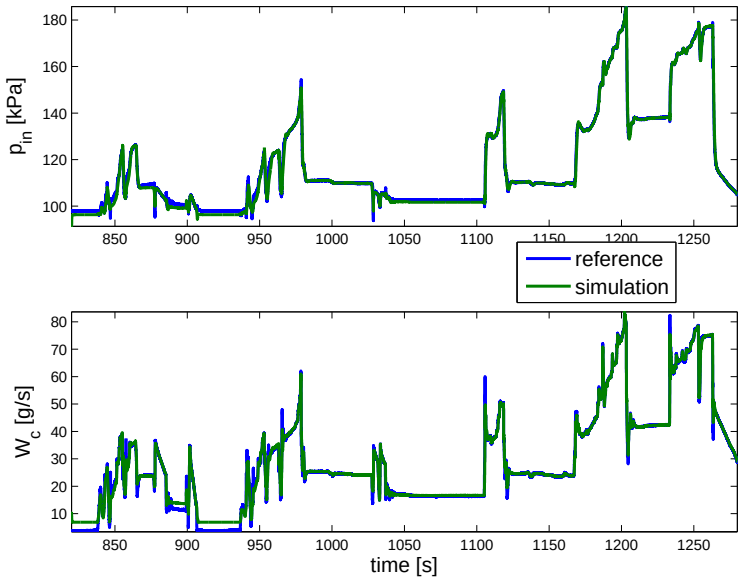

Figure 5.10: Simulation over the NEDC using Kantorovich's method where the Jacobian, $\bar{F}_{\bar{U}}(\bar{U}_0, x(0))$, corresponds to the Jacobian at simulation time 1210 sec.

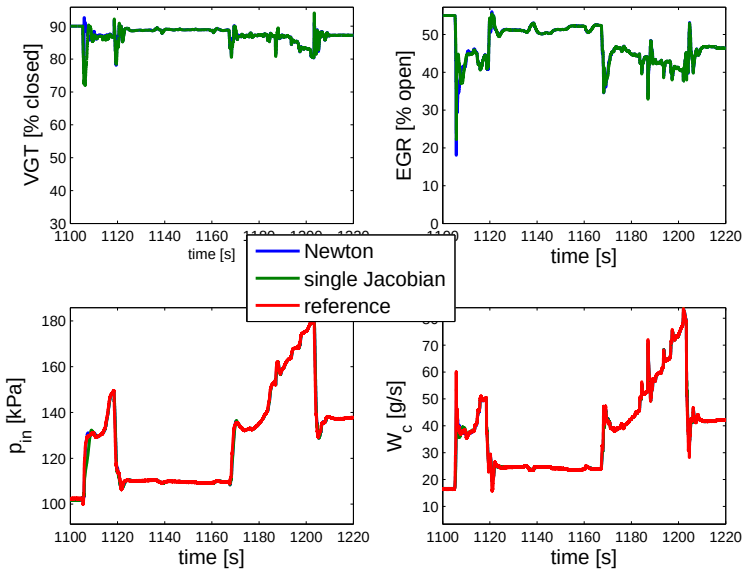

Figure 5.11: Simulations over the NEDC comparing Newton's method and Kantorovich's method.

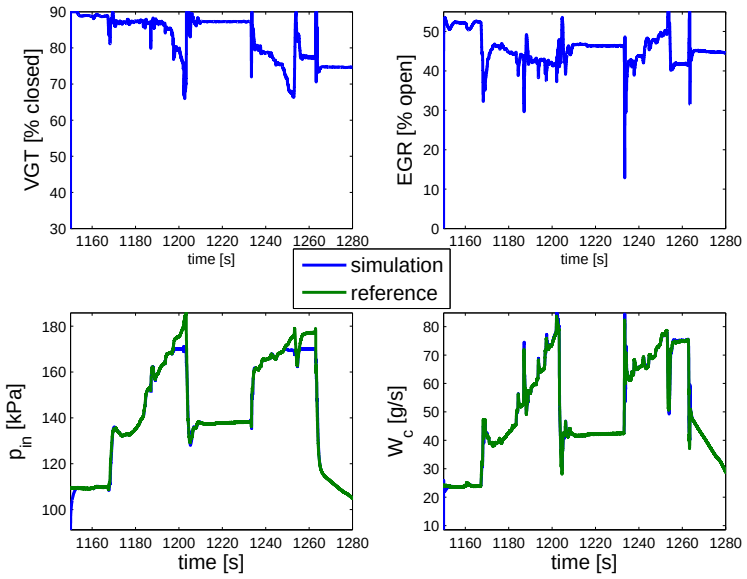

Figure 5.12: Simulation over the NEDC using Kantorovich's method demonstrating intake pressure maximum constraint satisfaction, $\bar{p}_{in} = 170$kPa.

In the simulations shown in Figures 5.10-5.12, $\bar{F}_{\bar{U}}(\bar{U}_0, x(0))$ in (5.98) is taken as the exact Jacobian at simulation time 1210 sec and used for the entirety of the drive cycle. As can be seen in Figure 5.10, the performance of the implementation based on Kantorovich's method is similar to Newton's method. The response of modified Newton iterations is slower than exact Newton at 1105 sec which can be seen in Figure 5.11. This sluggish response can be improved by precomputing a few Jacobians at different engine operating conditions and looking them up as the condition changes. Figure 5.12 shows that Kantorovich's method is also effective when the state constraint, $\bar{p}_{in} = 170$kPa, is active.

The complexity of solving (5.98) for $\bar{U}_{k+1}$ is $o(N(s+s_C) + N^2(n_u+n_g+2n_x)^2 + N^3(n_u+n_g+2n_x)^3)$ where s_C accounts for the complexity of computing $F^C(\bar{U}_k, x(t))$ and $F^C_{\bar{U}}(\bar{U}_k, x(t))$ for the i-th stage. Furthermore, when $h(x,u) \leq 0$ has a simple form, e.g., linear, it is a reasonable assumption that $s_C < s_J$ where s_J is the i stage cost of computing $\bar{F}_{\bar{U}}(\bar{U}_k, x(t))$. The estimated computation time using Kantorovich's method with a mid-range, 160MHz ECU with a the same horizon as in Table 5.1, $N = 5$, is 0.41 msec (compared to 0.55 msec with Newton's method and a multiple-shooting framework, see Section 5.3, and 0.9 msec with EP-GMRES 2 in Section 5.2, Table 5.1).

5.6 Rate-Based NMPC

In Section 5.2, zero-offset steady-state tracking was introduced through the use of parameter estimation (5.65) and an input-velocity-like form of the cost functional (5.66). As is known in the linear MPC case, the strategy of input-velocity form MPC and parameter/disturbance estimation has a number of disadvantages, [85]. These disadvantages include the need to design a disturbance model and estimator which also requires additional tuning parameters. Dynamic interactions between the controller and estimator may also need to be considered. Furthermore, if multiple zones, models, controllers, and estimators are required to cover a plant operating range, then initializations of the estimators must be considered to mitigate discontinuous control actions during zone switches. The NMPC design presented in Section 5.2 suffers from similar issues, e.g., the adapted parameters must be switched/reset somehow during zone switches.

In the linear MPC case, rate-based MPC has been shown to mitigate the above issues, [18,39,85,114]. What is unknown is how similar ideas can be applied to NMPC. The following discussion describes how rate-based NMPC can be developed. A few variants of rate-based NMPC are possible, including continuous time rate-based NMPC and discrete time rate-based NMPC. Simulation results on the GD engine high fidelity nonlinear model are then presented.

5.6.1 Continuous Time Rate-Based NMPC

For continuous time rate-based NMPC, the same idea as in [18] is used. Essentially, what is required is to take the time derivative of the model, $\dot{x} = f(x,u)$ and $y = f^y(x,u)$, which gives the following model with an augmented state vector $\xi = \begin{bmatrix} \dot{x}^T & x^T & y^T & u^T \end{bmatrix}^T \in \mathbb{R}^{2n_x+n_y+n_u}$,

$$\dot{\xi} = \bar{A}(\xi)\xi + \bar{B}(\xi)\dot{u}, \tag{5.99}$$

$$\bar{A}(\xi) = \begin{bmatrix} f_x(x,u) & 0 & 0 & 0 \\ I & 0 & 0 & 0 \\ f_x^y(x,u) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \bar{B}(\xi) = \begin{bmatrix} f_u(x,u) \\ 0 \\ f_u^y(x,u) \\ I \end{bmatrix}. \tag{5.100}$$

The continuous time rate-based NMPC optimization problem can then be formed, where the cost function,

$$J_{CRB} = \phi(\dot{x}(T), y(T) - r) + \int_0^T (y(\tau) - r)^T Q(y(\tau) - r) + \dot{u}(\tau)^T R\dot{u}(\tau), \tag{5.101}$$

$$\phi(\dot{x}(T), y(T) - r) = \begin{bmatrix} \dot{x}(T) \\ y(T) - r \end{bmatrix}^T P \begin{bmatrix} \dot{x}(T) \\ y(T) - r \end{bmatrix}, \tag{5.102}$$

is minimized subject to the constraints,

$$\xi_\tau(\tau) = \bar{A}(\xi(\tau))\xi(\tau) + \bar{B}(\xi(\tau))\dot{u}(\tau) \in \mathbb{R}^{2n_x+n_y+n_u}, \tag{5.103}$$

$$g(x(\tau), u(\tau)) = 0 \in \mathbb{R}^{n_g}, \tag{5.104}$$

$$h(x(\tau), u(\tau)) \leq 0 \in \mathbb{R}^{n_h}, \tag{5.105}$$

$$\psi(\dot{x}(T), y(T) - r) \leq 0 \in \mathbb{R}^{n_\psi}, \tag{5.106}$$

$$\dot{x}(0) = x(t),\ x(0) = x(t),\ y(0) = y(t),\ u(0) = u(t), \tag{5.107}$$

where $Q = Q^T > 0$, $R = R^T > 0$, $P = P^T \geq 0$, the reference, r, is treated as constant over the horizon, and it is assumed that the system is square, i.e., y, $u \in \mathbb{R}^{n_u}$. Now consider the simplified case where $x = y$, as is the case for the DAP prediction model framework (Section

5.2). Then, ξ becomes $\xi = \begin{bmatrix} \dot{x}^T & x^T & u^T \end{bmatrix}^T \in \mathbb{R}^{2n_x+n_u}$ and

$$\bar{A}(\xi) = \begin{bmatrix} f_x(x,u) & 0 & 0 \\ I & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \bar{B}(\xi) = \begin{bmatrix} f_u(x,u) \\ 0 \\ I \end{bmatrix}. \tag{5.108}$$

Furthermore, consider the case where multiple-shooting is used, as this allows for simple calculation of the Jacobian of the discretized necessary conditions, see Section 5.3, and allows for simple handling of state constraints while using Kantorovich's method, see Section 5.5. While the NMPC formulation (5.101)-(5.107) will give offset-free set-point tracking, the optimization problem, specifically the associated Newton iteration complexity will become larger compared to non-rate-based NMPC because many states and co-states need to be added as optimization variables. Again, let $N = T/\Delta\tau \in \mathbb{Z}^+$, where $\Delta\tau$ is the discretization period. The computational complexity of a Newton step (not Kantorovich) for the equality constrained discretized necessary conditions associated with (5.101)-(5.104) and (5.107) is $o(N(\tilde{s} + \tilde{s}_J) + N^2(n_u + n_g + 2n_x)^2 + N^3(n_u + n_g + 2n_x)^3)$ and $\mathcal{O}(N^3)$ for sufficiently large N, where $\tilde{s} > s$ and $\tilde{s}_J > s_J$ account for the additional stage cost associated with the extra states.

Here, continuous time rate-based NMPC has been introduced. However, it suffers from a number of issues inasmuch as the implementation is concerned due to (i) the bloating in the number of optimization variables, (ii) considerations for the limited computation time allotted for the DAP control problem, and (iii) advantages of data-driven discrete time modeling as discussed in the remark in Section 5.4. Instead, a discrete time rate-based NMPC formulation will be used to reduce the amount of bloating in the number of optimization variables.

5.6.2 Discrete Time Rate-Based NMPC

For discrete time rate-based NMPC, ideas similar to linear discrete time rate-based MPC will be used, see [85, 114]. What will be needed are state update equations for $\Delta x_k = x_k - x_{k-1}$. Let the discrete time model have the form,

$$x_{k+1} = f^d(x_k, u_k), \tag{5.109}$$

$$y_k = f^y(x_k, u_k). \tag{5.110}$$

Recall that the state and output equations (5.109) and (5.110) are equality constraints from the standpoint of the optimal control problem. An equivalent constraint to (5.109) and (5.110) is

$$\Delta x_{k+1} = x_{k+1} - x_k = f^d(x_k, u_k) - f^d(x_{k-1}, u_{k-1}), \tag{5.111}$$

$$\Delta y_k = y_k - y_{k-1} = f^y(x_k, u_k) - f^y(x_{k-1}, u_{k-1}). \tag{5.112}$$

Using the constraint (5.111) and (5.112), the following discrete time rate-based NMPC cost functional,

$$J_{DRB} = \phi(x_{N|k} - x_{N-1|k}, y_{N|k} - r) + \sum_{i=0}^{N-1} (y_{i|k} - r)^T Q (y_{i|k} - r) + (u_{i|k} - u_{i-1|k})^T R (u_{i|k} - u_{i-1|k}), \tag{5.113}$$

$$\phi(x_{N|k} - x_{N-1|k}, y_{N|k} - r) = \begin{bmatrix} x_{N|k} - x_{N-1|k} \\ y_{N|k} - r \end{bmatrix}^T P \begin{bmatrix} x_{N|k} - x_{N-1|k} \\ y_{N|k} - r \end{bmatrix}, \tag{5.114}$$

is minimized subject to the constraints,

$$x_{i+1|k} - x_{i|k} = f^d(x_{i|k}, u_{i|k}) - f^d(x_{i-1|k}, u_{i-1|k}), \tag{5.115}$$

$$y_{i|k} - y_{i-1|k} = f^y(x_{i|k}, u_{i|k}) - f^y(x_{i-1|k}, u_{i-1|k}), \tag{5.116}$$

$$g(x_{i|k}, u_{i|k}) = 0, \tag{5.117}$$

$$h(x_{i|k}, u_{i|k}) \leq 0, \tag{5.118}$$

$$\psi(x_{N|k} - x_{N-1|k}, y_{N|k} - r) \leq 0. \tag{5.119}$$

$$x_{0|k} = x_k, x_{-1|k} = x_{k-1}, y_{0|k} = y_k, y_{-1|k} = y_{k-1}, u_{-1|k} = u_{k-1}. \tag{5.120}$$

Again, consider the case where $y_k = x_k$, which is true for the DAP prediction model. Then (5.112) and (5.116), are no longer required. The computational complexity of a Newton step for the necessary conditions associated with the optimization problem (5.113), (5.115), (5.117), and (5.120) is $o(N(s+s_J)+N^2(n_u+n_g+2n_x)^2+N^3(n_u+n_g+2n_x)^3)$ and $\mathcal{O}(N^3)$ for sufficiently large N, which is the same as for the non-rate-based NMPC implementation in a multiple-shooting framework because the number of optimization variables does not change, i.e., because $u_{i|k}$'s and $x_{i|k}$'s are directly solved for rather than going through an intermediate step with $\Delta u_{i|k}$'s and $\Delta x_{i|k}$'s as would be the case in the linear MPC. Furthermore, the stage costs of computing the necessary conditions and associated Jacobian is the same as in the non-rate based case. This is because once $f^d(x_{i|k}, u_{i|k})$ is computed at stage i, $f^d(x_{i-1|k}, u_{i-1|k})$ does not need to be computed at stage $i+1$ and similarly with partial derivatives. However, the bandwidth of the sparse Jacobian will be about twice as large with rate-based NMPC compared with non-rate-based NMPC because $x_{i+1|k}$ in (5.111) is a function of $x_{i|k}$, $u_{i|k}$, $x_{i-1|k}$, and $u_{i-1|k}$ rather than just $x_{i|k}$ and $u_{i|k}$.

Remark: In the continuous time rate-based case, the number of optimization variables can be condensed by using the substitution $\Delta\tau\dot{u}_i = u_{i+1} - u_i$ and $\Delta\tau\dot{x}_i = x_{i+1} - x_i$ after the necessary conditions are discretized. However this will generally not result in the same necessary conditions as the second order derivatives, f_{xx}, f_{ux}, and f_{uu}, will appear in the

necessary conditions for the continuous time case which do not appear in the discrete time case. For the case where f is linear, the necessary conditions will be the same because the second order derivatives disappear. Note that this does not contradict the discussion in Section 5.4 because the models are fundamentally different, i.e., the discretization of the continuous time rate-based dynamics (5.99) does not result in the discrete time rate-based dynamics (5.111), unless f is linear.

Let $\bar{f}^d(x,u)$ and $\bar{f}^y(x,u)$, not necessarily equal to $f^d(x,u)$ and $f^y(x,u)$, describe the true plant. Let $\zeta_{k+1} = \begin{bmatrix} x_{k+1} \\ x_k \end{bmatrix} = \begin{bmatrix} \bar{f}^d(x_k,u_k) \\ x_k \end{bmatrix}$.

Proposition 5.1: Let $Q = Q^T > 0$, $R = R^T > 0$, and $P \geq 0$ in (5.113) and (5.114), and there exists $x_{ss} \in \mathbb{X}$ and $u_{ss} \in \mathbb{U}$ such that $x_{ss} = \bar{f}^d(x_{ss},u_{ss})$ and $r = \bar{f}^y(x_{ss},u_{ss})$, then $\begin{bmatrix} x_{ss}^T & x_{ss}^T \end{bmatrix}^T$ is an equilibrium for the closed loop system with the states ζ_k under under the closed loop unconstrained NMPC control defined by (5.113), (5.115), and (5.120).

Proof: Let $\zeta_k = \begin{bmatrix} x_k^T & x_{k-1}^T \end{bmatrix} = \begin{bmatrix} x_{ss}^T & x_{ss}^T \end{bmatrix}^T$. Then it is always possible to choose $u_{i|k} = u_{k-1} = u_{ss}$ for $i \in \{0,...,N-1\}$ which results in $x_{i|k} = x_k = x_{ss}$ for $i \in \{0,...,N\}$ in the optimization problem (5.113), (5.115), and (5.120). This results in the cost, $J_{DRB} = \phi(x_{N|k} - x_{N-1|k}, y_{N|k} - r) + \sum_{i=0}^{N-1}(y_{i|k} - r)^T Q(y_{i|k} - r) + (u_{i|k} - u_{i-1|k})^T R(u_{i|k} - u_{i-1|k}) = 0$. Because $J_{DRB} > 0$, $u_{i|k} = u_{k-1} = u_{ss}$ for $i \in \{0,...,N-1\}$ and $x_{i|k} = x_k = x_{ss}$ for $i \in \{0,...,N\}$ is a minimizer of (5.113), (5.115), and (5.120). With NMPC, the first control element, $u_{0|k}$, is applied which yields $x_{k+1} = x_k = x_{ss}$. Thus $\begin{bmatrix} x_{ss}^T & x_{ss}^T \end{bmatrix}$ is an equilibrium of the closed loop system with the states ζ_k under the closed loop unconstrained NMPC control defined by (5.113), (5.115), and (5.120). ■

Remark: Proposition 5.1 does not require or conclude anything regarding the stability of $x = x_{ss}$. In contrast, if non-rate-based NMPC is used, then it may not be the case that $u_{0|k} = u_{k-1}$ can be chosen and simultaneously satisfy $x_{i+1|k} = f^d(x_{i|k},u_{i|k}) = x_k = x_{ss}$ for $i \in \{0,...,N\}$. Then $u_{0|k} \neq u_{k-1}$ may be a minimizer and, once applied, x_{k+1} may not equal $x_k = x_{ss}$. Appropriate terminal penalties and terminal set constraints can be used to

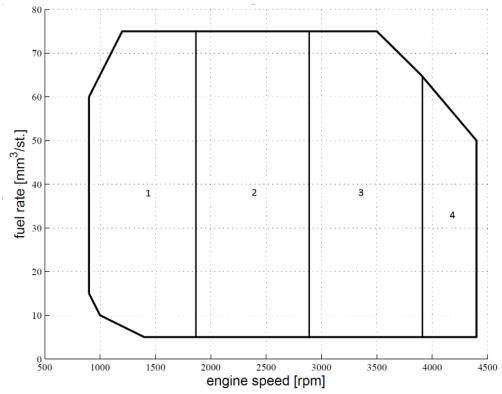

Figure 5.13: 4 zones of the piecewise polynomial model of the GD engine.

guarantee local stability, see Section 5.7.

5.6.3 DAP Simulation Results Using Rate-based NMPC

Simulations using discrete time rate-based NMPC have been conducted on the Toyota GD engine high fidelity nonlinear mean-value-model. The prediction model for the GD engine is identified directly in discrete time. The model uses a second order polynomial model similar to the continuous time case. The modeled outputs/states for the GD engine are MAP [kPa] and EGR rate [%], $x = [p_{in}\, \chi_{EGR}]^T$, and the inputs are EGR throttle [% closed], EGR valve [% open], VGT [% closed], engine speed [rpm], and fuel flow [mm3/st.], $u = [u_{th}\, u_{EGR}\, u_{VGT}\, N_e\, W_f]^T$. The discrete time model has the following form,

$$x_{k+1} = \theta f^2(x_k, u_k). \tag{5.121}$$

For the GD engine, the engine operating range is split into four zones, see Figure 5.13. As with rate-based linear MPC, zone switches, when a rate-based formulation is used, are not a concern. With the diesel air path rate-based NMPC formulation there are no integrators/estimators/adaptive parameters that need to be reset during zone switches. In each of the four zones, the polynomial model (5.121) is trained on the mean-value-model using a randomly generated step input trajectory that remains within $\pm 40\%$ of the full actuator

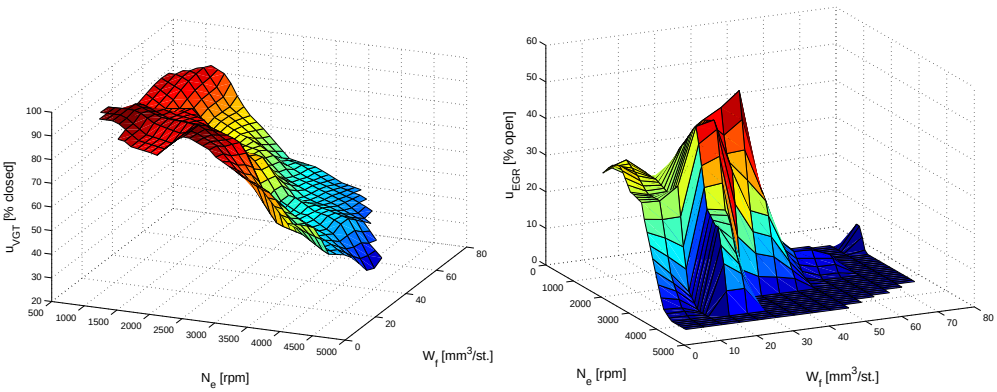

Figure 5.14: Nominal settings for VGT position and EGR valve position at different operating conditions.

range of the nominal production control settings. This is to ensure that the trajectory does not venture into poor extrapolation zones of the mean-value-model. The nominal settings obtained from an existing controller for the VGT position and EGR valve position are shown in Figure 5.14.

Figure 5.15 shows a transient comparison between the mean-value-model and polynomial model in zone 2. Indeed, the polynomial model is able to accurately represent the engine dynamics. The overall transient matching behavior between the mean-value-model and polynomial model is consistent over all four zones. Figure 5.16 shows good matching for the steady state values for p_{in} and χ_{EGR} using the nominal production control settings between the mean-value-model and polynomial model. Figure 5.17 shows good matching of DC gains around the nominal production control settings between the mean-value-model and polynomial model which indicates that the data-driven polynomial model learned something reasonable.

Figure 5.18 shows the closed loop simulation using discrete time rate-based NMPC (with polynomial prediction model) in loop with the GD engine mean-value-model. The control objective is to track p_{in} and χ_{EGR} through coordinated control of u_{VGT} and u_{EGR}. The prediction horizon is $N = 5$ with a sampling period and discretization period of $\Delta T = \Delta\tau = 32$ msec. The controller uses the rate-based formulation (5.113)-(5.118) and (5.120) with

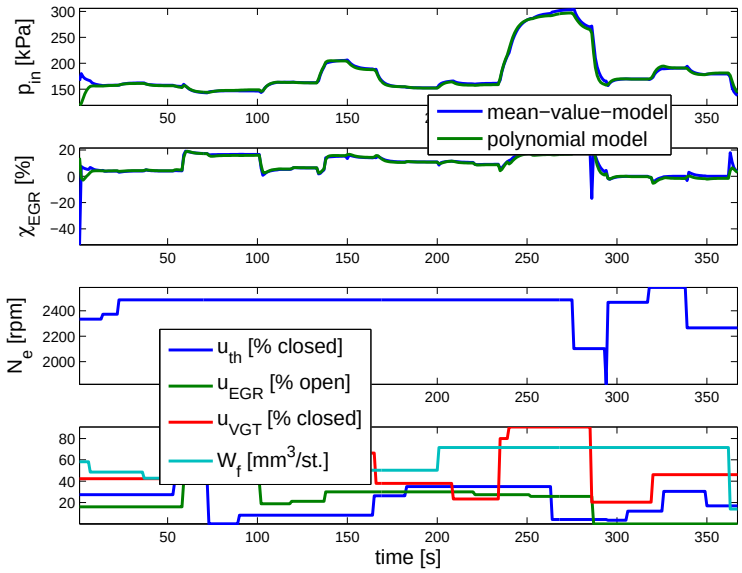

Figure 5.15: Transient comparison between the mean-value-model and polynomial model in zone 2.

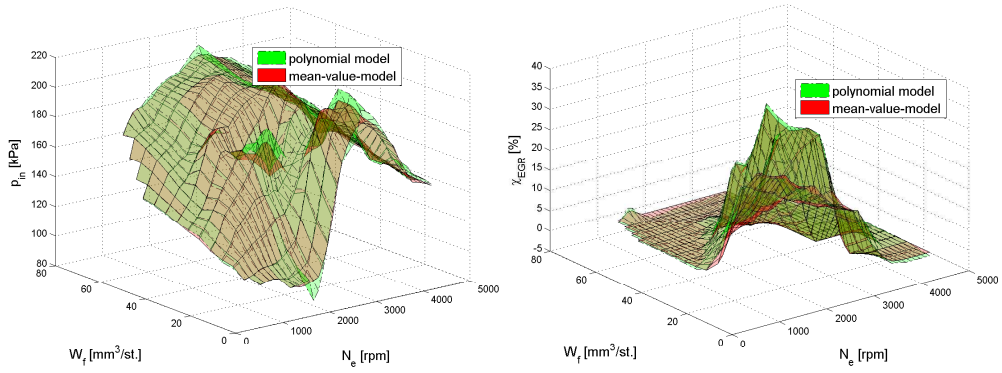

Figure 5.16: Steady state comparison between the mean-value-model and polynomial model at the nominal control settings.

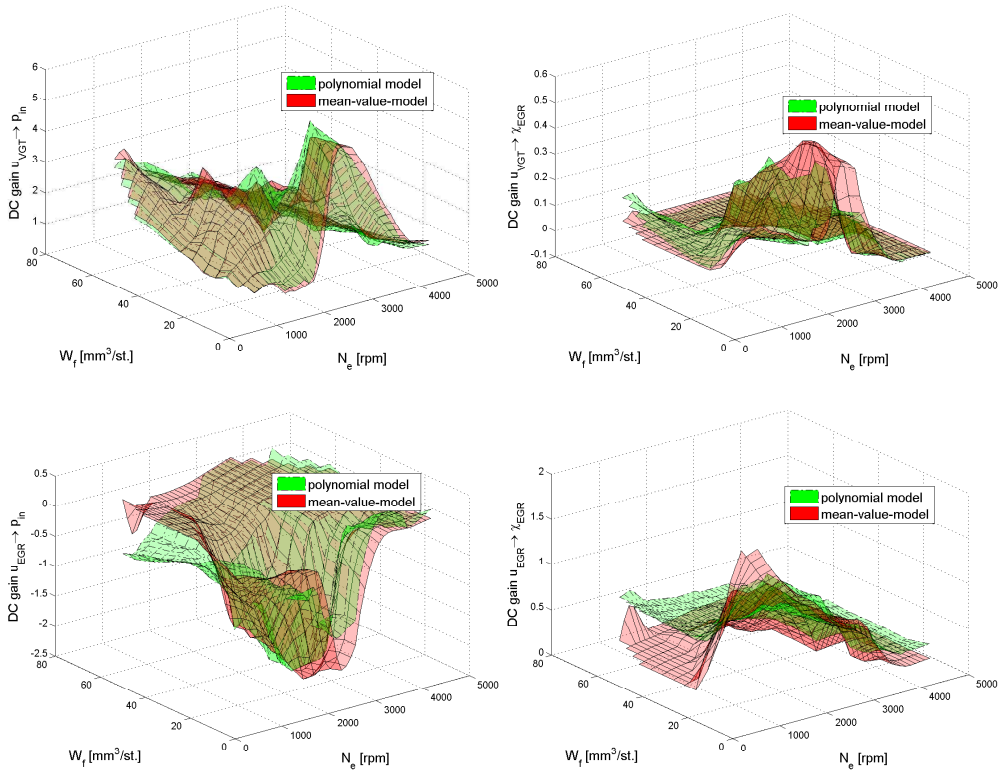

Figure 5.17: DC gain comparison between the mean-value-model and polynomial model around the nominal control settings.

the exterior penalties for control constraint enforcement, multiple-shooting to facilitate easy generation the Jacobian of the necessary conditions, and a single Newton (not Kantorovich) step per sample instant. As can be seen, rate-based NMPC is able to successfully achieve zero-offset steady-state tracking. Again this is done without any estimators, adaptation, or feed-forward.

Figure 5.19 tests the robustness of the rate-based NMPC controller in loop with the GD mean-value-model with measurement offsets on either $\pm 5\%$ in χ_{EGR} or ± 10 kPa in p_{in}. The test pattern corresponds to 15 mm^3/st. fuel steps from 0 to 90 mm^3/st. at each engine speed between 800 rpm and 4400 rpm in 200 rpm intervals. In all of these cases, the closed loop system is stable. In some cases, there are non-zero steady state offsets between the measurement and reference. In these cases, the actuators are saturating in the correct direction, e.g., the u_{VGT} is saturated closed when p_{in} is under the reference and the u_{EGR} is saturated closed when χ_{EGR} is above the reference. Figure 5.19 shows the controller performance when there are measurement offsets in both χ_{EGR} and p_{in}. The figure only shows the fuel steps at 800 rpm and 1000 rpm as the controller behaves stably and as expected for higher engine speeds. The controller begins to experience numerical issues, e.g., the Jacobian of the necessary conditions become singular, in the cases where there is a $+10$ kPa offset in p_{in}. For example, this can be seen as discontinuous control actions at 2.5 sec when there is a $+10$ kPa offset in p_{in} and $+5$ % offset in χ_{EGR}. Figure 5.21 shows the controller performance when there are lags, specifically first order filters, placed on the measurements of χ_{EGR} and p_{in}. The figure only shows the fuel steps at 800 rpm, 1000 rpm, and 1200 rpm as the controller behaves stably and as expected for higher engine speeds. The controller is robust to significant lags in χ_{EGR} with a time constant, $\tau_{\chi EGR}$, up to 1 sec. The controller is robust to small lags in p_{in} with a time constant, τ_{pin}, up to 0.1 sec. Figure 5.21 shows the controller is unstable at low engine speeds (800 rpm to 1200 rpm) when τ_{pin} is 0.15 sec. In future work, the polynomial model at low engine speeds will be improved, e.g., to improve extrapolation, to improve tracking performance and robustness

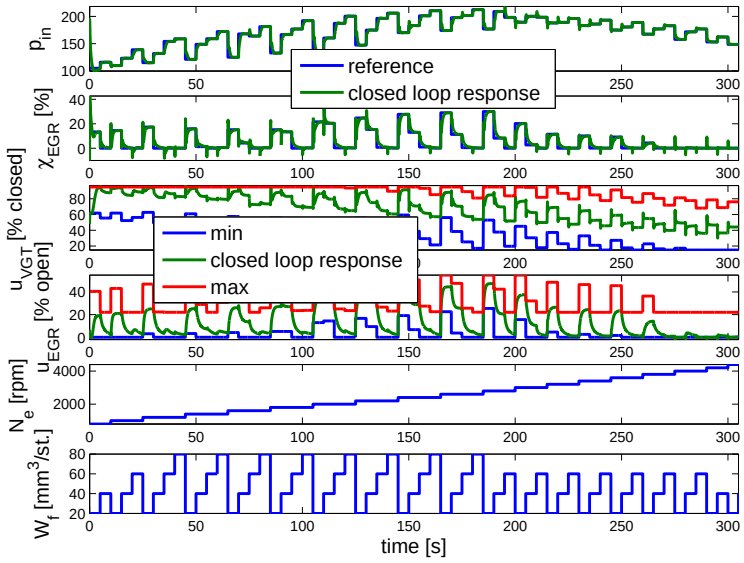

Figure 5.18: Closed-loop simulation using discrete-time rate-based NMPC on the GD engine mean-value-model.

at low engine speeds.

Figure 5.22 shows the sparsity pattern of the Jacobians of the necessary conditions with rate-based NMPC and non-rate-based NMPC using a multiple-shooting framework and horizon length $N = 5$. The estimated time computation time for a mid-range 160MHz ECU using discrete time rate-based NMPC is 0.87 msec (compared to 0.55 msec with non-rate-based multiple-shooting, see Section 5.3). This estimate is based on relative execution times of rate-based versus non-rate-based using MATLAB and an i5 processor. With a horizon of $N = 5$, the time required to solve the linear equation for the Newton step for rate-based NMPC is 0.115 msec and 0.109 msec with non-rate-based NMPC. This implies that the discrepancy between the 0.87 msec with rate-based and 0.55 msec with non-rate-based is primarily due to bloating in the code generation process, since stage costs of computing the necessary conditions and Jacobian of the necessary conditions are ideally the same.

5.7 Terminal Constraints for Stability with NMPC

Stability guarantees for NMPC, similar to linear MPC, are typically obtained through

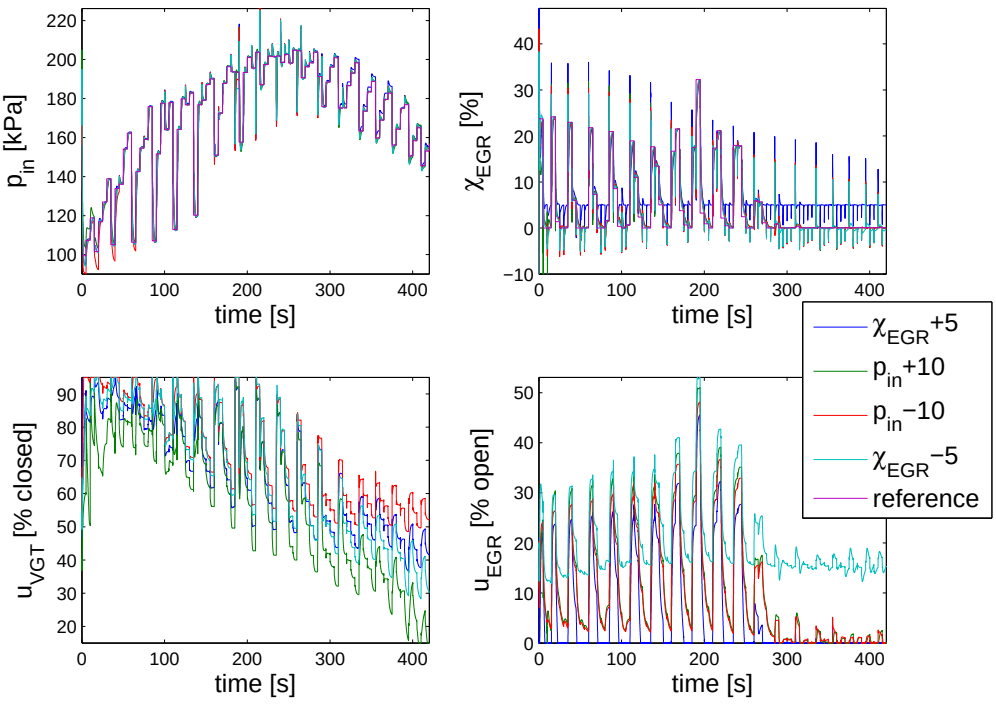

Figure 5.19: Closed-loop simulation using discrete-time rate-based NMPC on the GD engine mean-value-model with either χ_{EGR} or p_{in} measurement offsets.

the use of a terminal constraint and an associated terminal cost, see [16] and references therein. In this section, these types of constraints are applied to the DAP NMPC problem. Simulations using these stabilizing terminal set constraints are conducted in the loop with the high-fidelity mean-value-model of the GD engine at different operating conditions.

In the following, a discrete time system, $x_{k+1} = f^d(x_k, u_k)$ with $0 = f^d(0, 0)$, is considered. The NMPC control law will be obtained from minimizing a cost functional of the form,

$$J(x_k, u_{0:N-1|k}, N) = \phi(x_{N|k}) + \sum_{i=0}^{N-1} x_{i|k}^T Q x_{i|k} + u_{i|k}^T R u_{i|k}, \tag{5.122}$$

subject to the constraints,

$$x_{i|k+1} = f^d(x_{i|k}, u_{i|k}),\ x_{0|k} = x_k, \tag{5.123}$$

$$\psi(x_{N|k}) \leq 0, \tag{5.124}$$

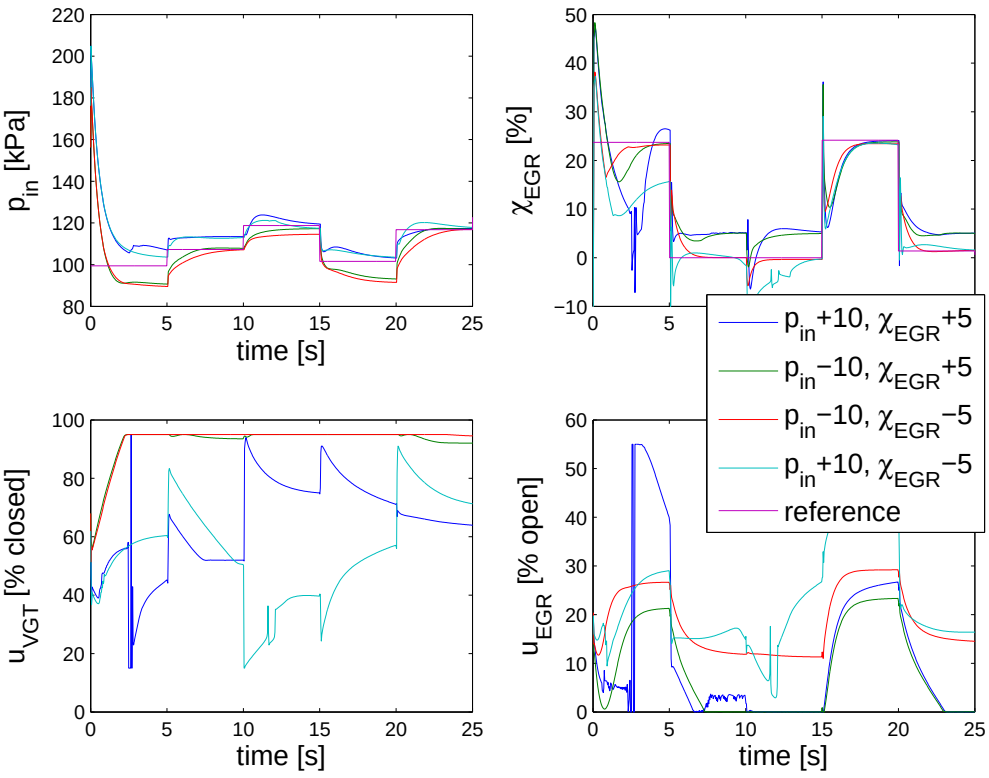

Figure 5.20: Closed-loop simulation using discrete-time rate-based NMPC on the GD engine mean-value-model with both χ_{EGR} and p_{in} measurement offsets.

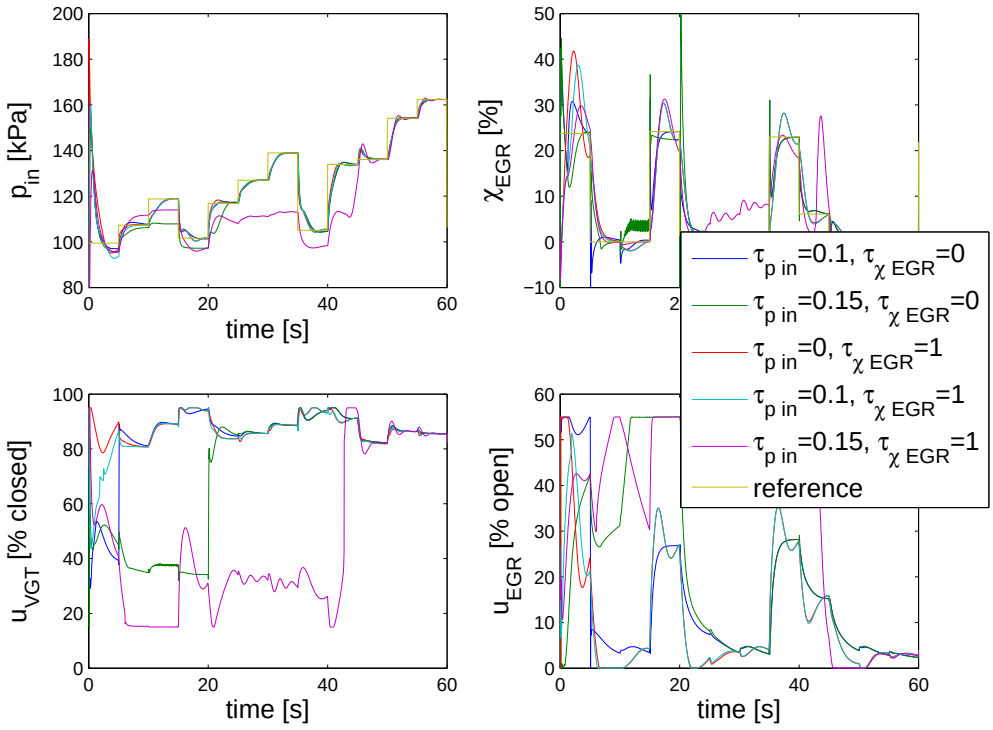

Figure 5.21: Closed-loop simulation using discrete-time rate-based NMPC on the GD engine mean-value-model with first order filters placed on the χ_{EGR} and p_{in} measurements.

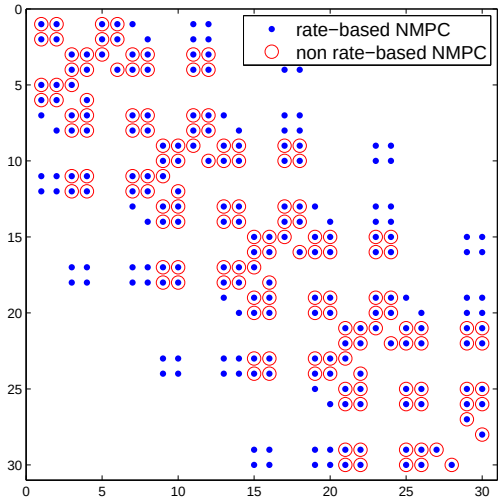

Figure 5.22: Sparsity pattern of the Jacobian of the necessary conditions using discrete time rate-based NMPC and non-rate-based NMPC, both with multiple-shooting and horizon $N = 5$. The number of non-zero entries are 232 and 160 with rate-based NMPC and non-rate-based NMPC respectively.

where $Q = Q^T \geq 0$ and $R = R^T > 0$. One way to obtain local stability is to use a contractive type of constraint combined with a single step horizon, i.e., $N = 1$. The contractive constraint has the following form,

$$\psi(x_{1|k}) = x_{1|k}^T P x_{1|k} - \alpha x_k^T P x_k,\ \alpha \in [0, 1),\ P = P^T > 0. \tag{5.125}$$

Let Contractive NMPC (CNMPC) denote a controller obtained from the optimization problem (5.122)-(5.125) with $N = 1$.

Assumption 5.1: There exists, $\tilde{\mathbb{X}}_1$, a nonempty neighborhood of the origin such that the optimization problem, (5.122)-(5.125), is recursively feasible.

Proposition 5.2: Under Assumption 5.1 and assuming that $x_0 \in \tilde{\mathbb{X}}_1$, the origin is asymptotically stable for the system, $x_{k+1} = f^d(x_k, u_k)$, under the CNMPC controller.

The proof of Proposition 5.2 follows directly from the fact that $x_k P x_k$ is a Lyapunov function for the closed loop by virtue of the contractive constraint (5.124)-(5.125) and it is explicitly assumed that $x_0 \in \tilde{\mathbb{X}}_1$, see [16].

Another choice for the terminal constraint and terminal cost when $N \geq 1$ is to choose,

$$\begin{aligned} \psi(x_{N|k}) = f^d(x_{N|k}, Kx_{N|k})^T P f^d(x_{N|k}, Kx_{N|k}) - x_{N|k}^T P x_{N|k} + x_{N|k}^T (Q + K^T R K) x_{N|k}, \\ \phi(x_{N|k}) = x_{N|k}^T P x_{N|k}, \\ P = P^T > 0, Q = Q^T > 0, R = R^T > 0, \end{aligned} \tag{5.126}$$

where P and K are chosen as the solution to the associated Discrete Algebraic Riccati Equation (DARE) for the linearized system and associated feedback gain, respectively.

Let Horizon N NMPC (HNNMPC) denote a controller obtained from the optimization problem (5.122)-(5.124) and (5.126) with $N > 1$.

Assumption 5.2: There exists, $\tilde{\mathbb{X}}_{FH}$, a nonempty neighborhood of the origin such that the optimization problem, (5.122)-(5.124) and (5.126), is recursively feasible.

Proposition 5.3: Under Assumption 5.2 and assuming that $x_0 \in \tilde{\mathbb{X}}_{FH}$, the origin is

asymptotically stable for the system, $x_{k+1} = f^d(x_k, u_k)$, under the HNNMPC controller.

The proof of Proposition 5.3 is the same as for the Quasi-Infinite controller in [16] where the cost functional (5.122) serves as a Lyapunov function for the closed loop system.

Remark: Note that no maximal output admissible set (MOAS) was used as the terminal set as is typically done, [16]. Rather, it is explicitly assumed that x_0 belongs to a set such that the optimization problem is recursively feasible. The constrained domain of attraction can always be computed a posteriori. Furthermore, in practice, soft state and soft terminal state constraints are always used to guarantee feasibility in the presence of plant/model mismatch and disturbances. The following simulations demonstrate that the constrained domain of attraction with soft stabilizing constraints may be larger than with hard stabilizing constraints.

Both CNMPC and HNNMPC have been implemented with rate-based NMPC, see Section 5.7, in loop with the mean-value GD engine model. Let $x_{k+1} = Ax_k + Bu_k$ and $y_k = Ix_k$ be the result of a linearization of (5.123). In these simulations with rate-based NMPC, P, is chosen as the solution to the Riccati equation based on the linear rate-based system, $\xi_k^T = [\Delta x_k \, e_k] = [x_k - x_{k-1} \, y_{k-1} - r]$, $\Delta u_k = u_k - u_{k-1}$,$\xi_{k+1} = \begin{bmatrix} A & 0 \\ I & I \end{bmatrix} \xi_k + \begin{bmatrix} B \\ 0 \end{bmatrix} \Delta u_k$, and $e_k = [0 \, I]\xi_k$, see [85, 114].

Simulations have been performed using CNMPC and HNNMPC and are shown in Figures 5.23 and 5.24 with the same optimization problem setup as in Section 5.7 with the addition of a soft CNMPC or HNNMPC terminal set constraint. Different simulations have been conducted showing how the closed loop behavior changes as the slack weights associated with the soft terminal set constraints increase (slack weight = 0 corresponds to completely inactive terminal set constraint). As can be seen in Figures 5.23 and 5.24, as the slack weights become large, i.e., when the soft constraints better approximate hard constraints, performance dramatically suffers, and, actually, stability is lost. This is because the use of terminal set constraints greatly limits the feasible region of the optimization problem. When

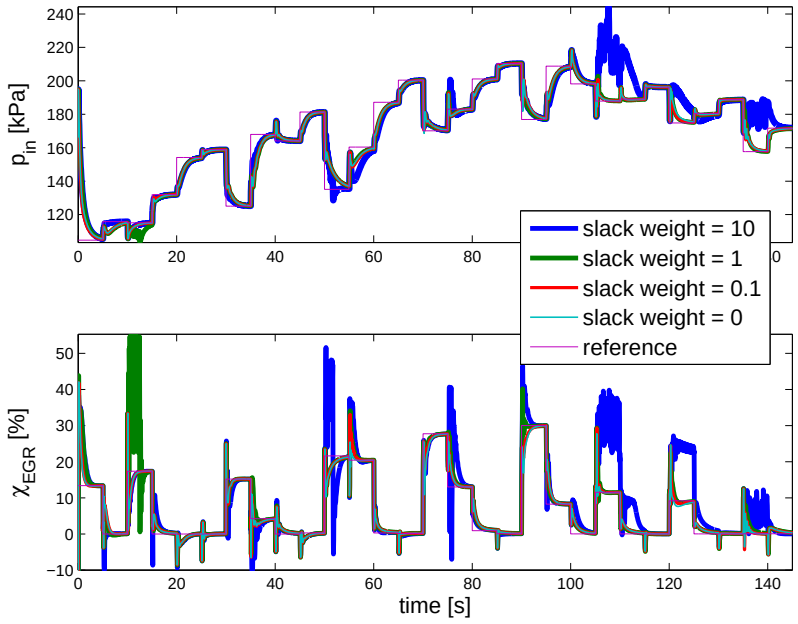

Figure 5.23: Simulations using CNMPC with soft contractive constraint enforcement in loop with the mean-value GD engine model at different operating conditions.

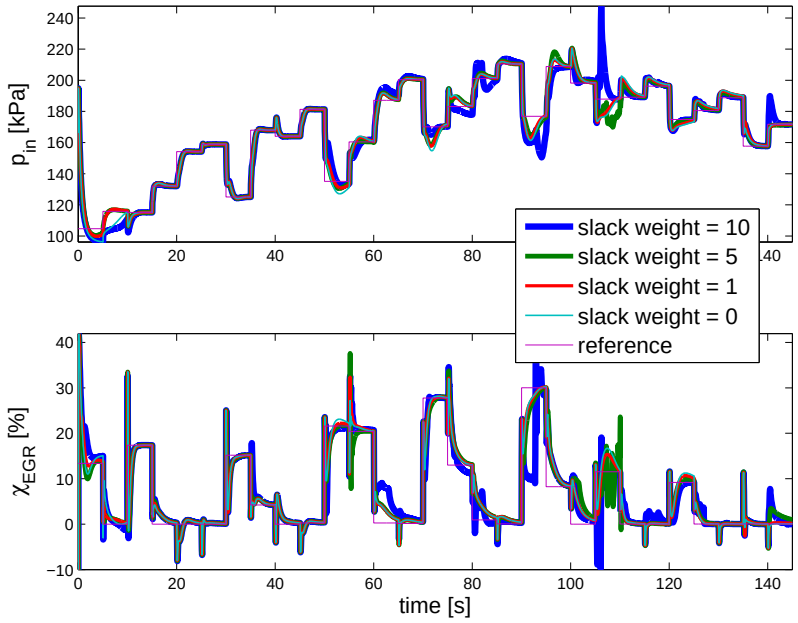

Figure 5.24: Simulations using HNNMPC with soft terminal set constraint enforcement in loop with the mean-value GD engine model at different operating conditions.

the optimization based controller is not feasible, the control is not defined. This infeasibility numerically leads to a singular Jacobian for the Newton step and zig-zagging behavior in the control and response. This can be seen at 10 sec in the CNMPC case and at 105 sec in both the CNMPC and HNNMPC case.

5.8 Conclusion

This chapter described the development of a Nonlinear Model Predictive Controller (NMPC) for the diesel air path. In total, through the various strategies described in this chapter, it has been shown that NMPC for engine control is indeed computationally possible, contrary to [33], and performs well throughout the engine operating range with minimal tuning effort. Furthermore, estimates of the ECU computation times for the NMPC strategies developed in this chapter (< 1 msec) are at least an order of magnitude faster than recently published work on the subject (on the order of 10's of msec), [27,78], and are not significantly slower than linear explicit MPC, see Chapter 2. Table 5.2 summarizes the computational progression of NMPC for the diesel air path with respect to the various strategies investigated and developed. Note that for all of the NMPC strategies in Table 5.2, the computational complexity is $\mathcal{O}(N^3)$ for sufficiently large N, however, it is more beneficial to analyze the computational complexity for small N, which is the case for the DAP application. The combination of Kantorovich's method with constrained NMPC is a major innovation that is able to reduce the computational complexity of NMPC. Furthermore, zero-offset set-point tracking is achieved through a novel rate-based NMPC strategy. Additional benefits of rate-based NMPC of the DAP are that (i) adaptation, which can lead to a complex dynamic interaction with the NMPC controller, is not required to achieve zero-offset set-point tracking and (ii) discontinuities at zone switches are mitigated because there are no adapted parameters to be reset/switched at the zone boundary, meaning that we can use more zones leading to more accurate models.

Table 5.2: Comparison of computation time for various NMPC methods.

Method	ECU [msec]	Complexity	Notes
AV-C/GMRES, $r = 0.01$	9.3	$o(k_{max}^3 + k_{max}^2 + (k_{max}+1)N\tilde{s})$, $1 \leq k_{max} \leq N(n_u + n_g + 2n_h)$	Benchmark strategy of [27], see Section 5.2.
EP-GMRES, $k_{max} = 2$	0.9	$o(k_{max}^3 + k_{max}^2 + (k_{max}+1)Ns_\gamma)$, $1 \leq k_{max} \leq N(n_u + n_g)$	With penalty function, no extra optimization variables need to be added, see Section 5.2
Multiple-shooting with analytical Jacobain	0.55	$o(N(s+s_J) + N^2(n_u + n_g + 2n_x)^2 + N^3(n_u + n_g + 2n_x)^3)$	Note that complexity analysis does not account for the fact that the Jacobian is sparse, see Section 5.3.
Kantorovich's method with multiple-shooting	0.41	$o(N(s+s_C) + N^2(n_u + n_g + 2n_x)^2 + N^3(n_u + n_g + 2n_x)^3)$	Forgo computing the Jacobian and assume that $s_C \leq s_J$, see Section 5.5.
Rate-based with multiple-shooting (w/o Kantorovich)	0.87	$o(N(s+s_J) + N^2(n_u + n_g + 2n_x)^2 + N^3(n_u + n_g + 2n_x)^3)$	Complexity is the same as multiple-shooting. The increase in ECU time is due to code generation bloating, see Section 5.6.
Explicit linear MPC	0.22	$o(N_r N_c N_p)$	See Chapter 2.

Chapter 6

Conclusions and Future Work

An advanced control technology, model predictive control (MPC), has been developed for the diesel air path (DAP) in this dissertation. Development of advanced control strategies, in the powertrain arena, has been motivated by the need to meet increasingly stringent fuel economy and emissions regulations, both current and on the horizon. The DAP is used as an exemplary powertrain system for which advanced control technologies must be developed.

The DAP application has been traditionally challenging to control due to its highly coupled nonlinear behavior, the need for constraints to be considered for driveability and emissions, and the very limited computational resources in engine control units (ECU). MPC has been viewed as a way to handle these challenges, however, current MPC solutions for the DAP are still limited. In the linear MPC (LMPC) case, the strategy of zoning, i.e., dividing a nonlinear operating range into multiple locally linear zones, is typically used. However, there are drawbacks to this strategy, as it can lead to a large calibration effort and memory usage in the ECU. In the nonlinear MPC (NMPC) case, it has previously been unknown if NMPC would be feasible for the DAP given the limited computational resources of ECUs and the perceived computational burden of NMPC.

This dissertation has focused on overcoming the challenges of applying MPC to the DAP, both in the linear and nonlinear case, and primarily addresses the need to reduce the computational complexity of MPC for DAP control. Even though the DAP is used as

the application in the work described in this dissertation, many of the developments have wider applicability. For example, the challenges of DAP control are analogous to those in gas turbine engine control in aircraft. Control of the DAP, in itself, has wide applicability as diesel engines power 95% of all U.S. freight transportation.

The main developments and results of this dissertation are summarized as follows.

- Chapter 2 described a rate-based MPC controller for the DAP. The use of rate-based MPC reduced the need to partition the nonlinear DAP operating range into zones, where, typically, a different model, controller, and calibration are needed for each zone. The use of partial inversion reduced the degree of nonlinearity of the nonlinear DAP plant. Combined with rate-based MPC, it was found that a single zone MPC controller was sufficient to cover the entire DAP operating range. Constraint remapping and the novel strategy of intermittent constraint enforcement were used to reduce the memory usage and chronometric load of the single zone controller. The resulting controller was used in simulations and in experiments throughout the engine operating range and has demonstrated good tracking and constraint enforcement performance.

- Chapter 3 described a novel gain scheduling strategy for explicit MPC. Typically, when gain scheduling explicit MPC, a different prediction model and resulting explicit MPC control law are stored per operating zone, which requires a large amount of ECU memory usage. To reduce the ECU memory usage, a simple gain block was placed between the output of the controller and input to the plant. Thus, only a simple fixed linear gain needs to be stored per zone, rather than a full explicit MPC controller. This reduced complexity gain scheduling strategy has been developed in a manner that is able to non-trivially enforce control constraints. This gain scheduling strategy was used successfully in closed loop simulations with a high fidelity nonlinear DAP model. It was demonstrated that the novel gain scheduling strategy resulted in similar tracking performance compared to the standard gain scheduling strategy with lower ECU memory usage. From a broader perspective, the scheduling strategy proposed in

this chapter gives a degree of flexibility (reconfigurability to model changes) to explicit MPC which is traditionally rigid.

- Chapter 4 described a robust linear MPC strategy for the DAP, i.e., a tube MPC strategy. Enhancements to the standard tube MPC strategy were made which takes into consideration that disturbances may not only be bounded in magnitude, but also that their rates-of-change may also be bounded. For example, the engine speed and fuel rate in the the DAP problem are treated as disturbances. However, these disturbances cannot change arbitrarily, and are in fact rate limited as a result of physical limitations. Taking these rate limits into account, the conservativeness of tube MPC is reduced, i.e., the amount of constraint tightening is reduced. This strategy demonstrated robust constraint enforcement for the DAP application in both simulations and experiments. The drawback of using tube MPC is that the computational complexity of the controller grows, and, in the case of the DAP, grows significantly. Motivated by the need to reduce computational complexity for tube MPC in general, i.e., not just for the DAP, a novel strategy of splitting the tube MPC optimization problem has been developed. As a result, the computational complexity for tube MPC with the split optimization problem is on the same order as for non-tube MPC.

- Chapter 5 described a nonlinear MPC strategy for the DAP. The conclusion is that NMPC is indeed viable for the DAP despite previous thoughts on the subject. This computationally simple NMPC strategy for the DAP was obtained through a number of strategies: a simple polynomial prediction model, exterior penalties for inequality constraint handling, multiple-shooting which simplifies the computation of the Jacobian, and the novel combination of Kantorovich's method with constrained NMPC which dictates that the Jacobian does not need to computed at every Newton iteration/time step. In fact, it was demonstrated in simulation that the Jacobian only needs to be computed at a single operating condition and good tracking performance

can still be achieved. Additionally, rate-based ideas, for which results are well known in the linear MPC case, have been extended to nonlinear MPC. Utilizing this novel rate-based NMPC strategy has a number advantages similar to the linear case: the integral action does not need to be tuned separately from the feedback controller, e.g., versus using disturbance estimation, and discontinuous zone switches are mitigated which means more zones can be used which can lead to a more accurate prediction model.

Of the developments on lMPC, intermittent constraint enforcement, gain scheduling explicit MPC, and the split tube-MPC method are regarded as major innovations that also have broad applicability beyond the DAP example. Of the developments on NMPC, Kantorovich's method applied to constrained NMPC and rate-based NMPC are regarded as major innovations that also have broad applicability beyond the DAP example.

Through the developments described in this dissertation, many of the computational challenges of applying MPC to the DAP have been overcome to the extent that MPC can be implemented on today's ECUs. While the rate-based MPC strategy for the DAP described in Chapter 2 was able to achieve low computational complexity, some performance was sacrificed. In order to improve the performance, e.g., for tracking, the gain scheduling strategy of Chapter 3 was developed and it was found that the performance of the gain scheduling strategy of Chapter 3 is comparable to a zoning strategy. Robust constraint enforcement, i.e., tube MPC, for the DAP was investigated in Chapter 4 and various methods for managing the computational complexity of tube MPC were developed.

While many performance and computational complexity gains were achieved through the linear MPC work described in this dissertation compared to existing methods, many methods specific to the DAP were used to render the DAP plant "more linear." Thus, NMPC was investigated in an effort to streamline the development of MPC for general nonlinear systems. Towards this end, a NMPC strategy has been developed for the DAP that, in addition to

the observations previously described, is very simple to implement. The rate-based NMPC has been run successfully "out-of-the-box" in loop with Toyota's GT-POWER based model of the GD engine and evaluations of real world impact are currently underway, [46], e.g., on fuel economy and emissions. Efforts to optimize the code generation process for the various NMPC strategies described in Chapter 5 are also underway, [112].

Throughout the work comprising this dissertation, a number of observations were made that require further research. The future research directions, some specific to the methods used in this dissertation and some more general, are summarized as follows.

- When intermittent constraint enforcement is used with hard constraints, how much do the constraints need to be tightened to guarantee that the original constraints are satisfied? Can recursive feasibility be guaranteed? Furthermore, is there a systematic way to choose the instances in the prediction horizon to enforce the constraints?

- Stability results for the switched explicit gain scheduling strategy were developed for the unconstrained case. Further work needs to be done to consider the case where constraints are active. For example, can an error bound between the nominal system and switched system be established when a constraint is active? If an error bound can be established, can the state and output constraints be appropriately tightened to guarantee enforcement of the original constraints. Furthermore, what are the limitations of the gain scheduling strategies considering that there fewer degrees of freedom to modify the closed loop dynamics compared to changing the prediction model online?

- With the reduced complexity tube MPC strategy, the same maximal output admissible set as with standard tube MPC can be established. However, stability of the reduced complexity of tube MPC strategy has yet to be established.

- Stability properties of NMPC when the optimality conditions are not solved to completion need to be established, e.g., when a limited number of Newton iterations is used.

- For the DAP NMPC formulation, exterior penalties were found to be quite successful, e.g., only a few Newton iterations are required to handle active set changes. However, fast, strictly feasible optimization methods for NMPC should also be investigated.

- More generally, the following question should be investigated; how accurate does the prediction model actually need to be to satisfy performance specifications?

- It has been demonstrated in this dissertation that NMPC for the DAP is feasible. Can the same be said for economic MPC where the goals of maximizing fuel economy, satisfying emissions regulations, and maintaining drivability are directly optimized? In light of current events regarding Volkswagon's use of defeat devices during EPA testing, [111], it will become increasingly important that those aforementioned goals are met in real world driving scenarios and economic MPC may be one of the key tools for achieving those goals.

Bibliography

[1] Alessio, A., and Bemporad, A., "A survey on explicit model predictive control," in *Nonlinear Model predictive Control: Towards New Challenging Applications*, Magni, L., Raimondo, D.M., and Allgöwer, F, Eds., Vol. 384 of *Lecture Notes in Control and Information Sciences*, pp. 345-369, 2009.

[2] Allgöwer, F., "Industry 4.0: challenges and opportunities for optimization-based control," *2014 14th International Conference on Control, Automation and Systems*, KINTEX, Korea, 2014

[3] Banivadekar, A., Bodek, K., Cheah, L., Evans, C., Groode, T., Heywood, J., Kasseris, E., Kromer, M., and Weiss, M., "One the road in 2035: reducing transportation's petroleum consumption and GHG emissions," *MIT Laboratory for Energy and Environment,* Cambridge, Massachusetts, LFEE 2008-05 RP, July 2008.

[4] Bastani, P., Heywood, J., and Hope, C., "U.S. CAFE standards: potential for meeting light-duty vehicle fuel economy targets, 2016–2025," *MIT Energy Initiative*, Cambridge, Massachusetts, 2012.

[5] Bemporad, A., Morari, M., Dua, V., and Pistikopoulos, E., "The explicit linear quadratic regulator for constrained systems," *Automatica*, Vol. 38, No. 1, pp 3-20, 2002.

[6] Bemporad, A., "Hybrid toolbox - user's guide," 2004, Available: http://cse.lab.imtlucca.it/ bemporad/hybrid/toolbox.

[7] Bemporad, A., "Explicit MPC: basics, fast implementation, advantages and limitations," *European Control Conference*, Zurich, July 17, 2013.

[8] Betti, G., Farina, M., and Scattolini, R., "An MPC algorithm for offset-free tracking of constant reference signals," *2012 IEEE 51st Annual Conference on Decision and Control,* Maui, HI., 2012.

[9] Betti, G., Farina, M., and Scattolini, R., "A robust MPC algorithm for offset-free tracking of constant reference signals" *IEEE Transactions in Automatic Control,* Vol. 58, No. 9, 2013.

[10] Borrelli, F., Baotić, M., Pekar, J., and Stewart, G., "On the computation of linear model predictive control laws," *Automatica*, Vol. 46, No. 6, pp. 1035-1041, June 2010.

[11] Boyd, S., and Vandenberghe, L., "Convex Optimization," *Cambridge University Press*, 2004.

[12] Camacho, E.F., and Alba, C.B., "Model Predictive Control," *Springer Science & Business Media*, 2013.

[13] Chachuat, B., "Optimal control lectures 25-27: maximum principles," *McMaster University*, 2009.

[14] Choroszucha, R., Jing, S., and Butts, K., "Closed-loop model order reduction and MPC for diesel engine airpath control," *Proc. of the 2015 American Control Conference*, July 1-3, Chicago, IL, 2015.

[15] Cook, J., Sun, J., Buckland, J., Kolmanovsky, I.V., Peng, H., and Grizzle, J.W., "Automotive powertrain control – a survey," *Asian Journal of Control*, Vol. 8, No. 3, pp. 237-260, 2006.

[16] De Nicolao, G., Magni, L., and Scattolini, R., "Stability and robustness of nonlinear receding horizon control," in *Nonlinear Model Predictive Control*, Allgöwer, F, and Zheng, A., Eds., Vol. 26 of *Progress in Systems and Control Theory*, pp. 3-22, 2000.

[17] de Oliveria Kothare, S.L., and Morari, M., "Contractive model predictive control for constrained nonlinear systems," *IEEE Transactions on Automatic Control*, Vol. 45, No. 6, pp 1053-1071, 2000.

[18] DeCastro, J., "Rate-based predictive control of turbofan engine clearance," *Journal of Propulsion and Power*, Vol. 23, No. 4, 2007.

[19] del Re, L., "Hybrid MPC for minimum phase nonlinear plants," *Proc. of the 3rd European Control Conference*, Rome, Italy, 1995.

[20] Diehl, M., "Numerical Optimal Control," *Optimization in Engineering Center (OPTEC)*, 2011.

[21] The Economist, "The little engine that could," March 7, 2015.

[22] Faust, J., Sun, J., Butts, K., Lu, Z., and Tanaka, S., "Parameterization and adaptation of gasoline engine air system model via linear programming support vector regression," *Proc. of the 2012 American Control Conference*, Montreal, 2012.

[23] Feller, C., "Explicit MPC of higher-order linear processes via combinatorial multiparametric quadratic programming," *Proc. of the 2013 European Control Conference*, Zurich, July 17-19, 2013.

[24] Feru, E., Lazar, M., Gielen, R.H., Cairano, S.D., and Kolmanovsky, I.V., "Lyapunov-based constrained engine torque control using electronic throttle and variable cam timing," *Proc. of the 2012 American Control Conference*, Montreal, Canada, June 27-29, 2012.

[25] Fischer, A. "A special Newton-type optimization method," *Optimization*, Vol. 24, pp. 269-284, 1992.

[26] Freescale Semiconductor, Inc., "MPC5644A microcontroller datasheet," 2014.

[27] Gaglieardi, D., Ohtsuka, T., and del Re, L., "Direct C/GMRES control of the air path of a diesel engine," *Proc. of the IFAC 19th World Congress*, Cape Town, South Africa, August 24-29, 2014.

[28] Garg, S., "A simplified scheme for scheduling multivariable controllers," *Control Systems, IEEE*, Vol. 17, No. 4, pp. 24-30, 1997.

[29] Grisson, T., "Turbocharging for improved engine performance & reduced CO_2 emissions," *BorgWarner*, 2013.

[30] Grüne, L., and Pannek, J., "Nonlinear Model Predictive Control, Theory, and Algorithms," *Springer*, 2011.

[31] Hadef, J.E., Olaru, S., Rodriguez-Averbe, P., Colin, G., Chamaillard, Y., and Talon, V., "Explicit nonlinear model predictive control of the air path of a turbocharged spark-ignited engine," *Proc. of the 2013 International Conference on Control Applications*, 2013.

[32] Hermans, R.M., Lazar, M., Gielen, R.H., Cairano, S.D., and Kolmanovsky, I.V., "Low-complexity model predictive control of electromagnetic actuators with a stability guarantee," *Proc. of the 2009 American Control Conference*, St. Louis, MO, June 10-12, 2009.

[33] Herceg, M., Raff, T., and Allgöwer, F., "Nonlinear model predictive control of a turbocharged diesel engine," *Proc. of the 2006 IEEE International Conference on Control Applications*, Munich, Germany, 2006.

[34] Heywood, J., "Internal Combustion Engine Fundamentals," *McGraw-Hill, Inc.*, 1988.

[35] Hirch, M., and del Re, L., "Iterative identification of polynomial NARX models for complex multi-input systems," *Proc. of the 8th IFAC Symposium on Nonlinear Control Systems*, University of Bologna, Italy, September 1-3, 2010.

[36] Honeywell, Inc., "Honeywell OnRAMP," 2014, Available: http://www.honeywellonramp.com.

[37] Howell, R., "On asymptotic notation with multiple variables," *Technical Report 2007-4*, 2008.

[38] Huang, M., Nakada, H., Polavarapu, S., Choroszucha, R., Butts, K., and Kolmanovsky, I.V., "Towards combining nonlinear and predictive control of diesel engines," *Proc. of the 2013 American Control Conference*, Washington, DC, 2013.

[39] Huang, M., Nakada, H., Polavarapu, S., Butts, K., and Kolmanovsky, I.V., "Rate-based model predictive control of diesel engines," *Proc. of the 7th IFAC Symposium on Advances in Automotive Control*, Tokyo, Japan, 2013.

[40] Huang, M., Nakada, H., Polavarapu, S., Butts, K., and Kolmanovsky, I.V., "Rate-based model predictive control of diesel engines," *7th IFAC Symposium on Advances in Automotive Control*, Tokyo, Japan, 2013, Presentation.

[41] Huang, M., Nakada, H., Polavarapu, S., Butts, K., and Kolmanovsky, I.V., "Rate-based contractive model predictive control of diesel air path," *Proc. of the ASME 2013 Dynamic Systems and Control Conference*, Palo Alto, CA, USA, 2013.

[42] Huang, M., Nakada, H., Butts, K., and Kolmanovsky, I.V., "Robust rate-based model predictive control of diesel engine air path," *Proc. of the 2014 American Control Conference*, Chicago, IL, USA, 2014.

[43] Huang, M., and Kolmanovsky, I.V., "Rate-based model predictive control method for internal combustion engine air path control," US Patent App. US20140174413 A1, June 26, 2014.

[44] Huang, M., Kolmanovsky, I., "Rate-based contractive model predictive control method for internal combustion engine air path control," US Patent App. US2014174414 A1, June 26, 2014.

[45] Huang, M., Kolmanovsky, I., "Switch gain scheduled explicit model predictive control of diesel engines," US Patent App. filed Oct. 21, 2014.

[46] Huang, M., Kolmanovsky, I., Kirolkar, S., and Butts, K., "Diesel engine air path control: August 2015 - quarterly report," *Toyota Internal Document*, 2015.

[47] Huang, M., Nakada, H., Butts, K., and Kolmanovsky, I.V., "Nonlinear Model Predictive control of diesel engine air path: a comparison of constraint handling and computational strategies," *Proc. of the 5th IFAC Conference on Nonlinear Model Predictive Control*, Seville, Spain, September 17-20, 2015.

[48] Huang, M., Butts, K., and Kolmanovsky, I.V., "A low complexity gain scheduling strategy for explicit model predictive control of a diesel air path," *Proc. of the ASME 2015 Dynamic Systems and Control Conference*, Columbus, OH, 2015.

[49] Huang, M., Zaseck, K., Butts, K., and Kolmanovsky, I.V., "Rate-based model predictive control for diesel engine air path: design and experimental evaluation," *IEEE Transactions on Control Systems Technology*, Under review.

[50] Ioannou, P., and Sun, J., "Robust Adaptive Control," *Prentice Hall*, 1996.

[51] Izmailov, A., and Solodov, M., "Newton-Type Methods for Optimization and Variational Problems," *Springer Series in Operations Research and Financial Engineering*, Springer, 2014.

[52] Jacobson, D.H., and Lele, M.M., "A transformation technique for optimal control problems with a state variable inequality constraint," *IEEE Transactions on Automatic Control*, Vol. 14, pp. 457-464, 1969.

[53] Jankovic, M., Jankovic, M., and Kolmanovsky, I.V., "Constructive Lyapunov control design for turbocharged diesel engines," *IEEE Transactions on Control Systems Technology*, Vol. 8, No. 2, pp. 288-299, 2000.

[54] Jimbo, T., Tanaka, S., Sata, K., and Hibino, R., "Predictive control for high-EGR SI engines without misfire via flow-based design," *Proc. of the 2012 IEEE Conference on Decision and Control*, Maui, HI, December, 2012.

[55] Kapinski, J., Deshmukh, J., Sankaranarayanan, S., and Arechiga, N., "Simulation-guided Lyapunov analysis for hybrid dynamical systems," *Proc. of the 17th International Conference on Hybrid Systems: Computation and Control*, Berlin, Germany, April 15-17, 2014.

[56] Karlsson, M., Ekholm, K., Strandh, P., Johansson, R., and Tunestål, P., "Multiple-input multiple-output model predictive control of a diesel engine," *Proc. of the 6th IFAC Symposium Advances in Automotive Control*, Munich, Germany, July 12-14, 2010.

[57] Karnik, A., Buckland, J., and Freudenberg, J., "Electronic throttle and wastegate control for turbocharged gasoline engines," *Proc. of the 2005 American Control Conference*, Portland, OR, 2005.

[58] Kelley, C.T., "Iterative Methods for Linear and Nonlinear Equations," *Society for Industrial and Applied Mathematics*, 1995.

[59] Knittel, C., "Automobiles on steroids: product attribute trade-offs and technological progress in the automobile sector, " *American Economic Review*, Vol. 101, No. 7, pp. 3368-3399, 2011.

[60] Kolmanovsky, I.V., and Gilbert, E.G., "Maximal output admissible sets for discrete-time systems with disturbances inputs," *Proc. of the American Control Conference*, Seattle, WA, 1995.

[61] Kolmanovsky, I.V., Moral, P., van Nieuwstadt, M., and Stefanopoulou, A., "Issues in modeling and control of intake flow in variable geometry turbocharged engines," in: *Systems Modelling and Optimization*, Edited by M.P. Polis et al. published by Chapman and Hall/CRC, Chapman Hall/CRC Research Notes in Mathematics, Vol. 396, pp. 436-445, 1999.

[62] Kurzhanskiy, A.A., and Varaiya, P., "Ellipsoidal Toolbox," 2006, Available: http://code.google.com/p/ellipsoids.

[63] Kvasnica, M., Grieder, P., and Baotić, M., "Multi Parametric Toolbox (MPT)," 2004, Available: http://control.ee.ethz.ch/mpt.

[64] Langson, W., Chryssochoos, I., Rakovic, S.V., and Mayne, D.Q., "Robust model predictive control using tubes," *Automatica*, Vol. 40, No. 1, pp. 125-133, 2004.

[65] Ljung, L., "System Identification – Theory for the User," *Prentice Hall*, 1999.

[66] Lezhnev, L., Kolmanovsky, I.V., and Buckland, J., "Boosted gasoline direct injection engines: comparison of throttle and VGT controllers for homogeneous charge operation," *SAE International*, 2002-01-0709, 2002.

[67] Mackenroth, U., "Robust control systems: theory and case studies," *Springer*, 2003.

[68] The Mathworks, Inc., "Model predictive control toolbox," 2014, Available: http://www.mathworks.com/help/mpc/index.html.

[69] Maeder, U., Borrelli, F., and Morari, M., "Linear offset-free model predictive control," *Automatica*, Vol. 45, No. 10, pp. 2214–2222, 2009.

[70] Maeder, U., Borrelli, F., and Morari, M., "Nonlinear offset-free model predictive control," *Automatica*, Vol. 48, No. 9, pp. 2059–2067, 2012.

[71] The Mathworks, Inc., "System Identification Toolbox," 2015, Available: http://www.mathworks.com/products/sysid.

[72] Mayne, D.Q., Rawlings, J.B., Rao, C.V., and Scokaert, P.O.M, "Constrained model predictive control: stability and optimality," *Automatica*, Vol. 36, No. 6, pp 789-814, 2000.

[73] Mayne, D.Q., and Langson, W., "Robustifying model predictive control of constrained linear systems," *Electronics Letters*, Vol. 37, No. 23, pp. 1422-1423, 2001.

[74] Mayne, D.Q., Seron, M.M., and Rakovic, S.V., "Robust model predictive control of constrained linear systems with bounded disturbances," *Automatica*, Vol. 41, No. 2, pp. 219-224, 2005.

[75] Mayne, D.Q., Rakovic, S.V., Findelsen, R., and Allgower, F., "Robust output feedback model predictive control of constrained linear systems," *Automatica*, Vol. 42, No. 7, pp. 1217-1222, 2006.

[76] Mayne, D.Q., Kerrigan, E.C., and Falugi, P., "Robust model predictive control: advantages and disadvantages of tube-based methods," *Proc. of the 18th IFAC World Congress*, Milano, Italy, 2011.

[77] Moraal, P., and Kolmanovsky, I.V., "Turbocharger modeling for automotive control applications," *SAE Technical Paper*, 1999-01-0908, 1999.

[78] Murilo, A., Alamir, M., and Alberer, D., "A general NMPC framework for a diesel engine air path," *International Journal of Control*, Vol. 87, No. 10, pp. 2194-2207, 2014.

[79] Nakada, H., Martin, P., Milton, G., Iemura, A., and Ohata, A., "An application study of online reference governor to boost pressure control for automotive diesel engines," *Proc. of the 2014 American Control Conference*, Portland, OR, 2014.

[80] Nevistic, V., and del Re, L., "Feasible suboptimal model predictive control for linear plants with state dependent constraints," *Proc. of the 1994 American Control Conference*, 1994.

[81] Obama, B., "Presidential memorandum regarding fuel efficiency standards," *Office of the Press Secretary, The White House*, May 2010.

[82] Ohtsuka, T., "A continuation/GMRES method for fast computation of nonlinear receding horizon control," *Automatica*, Vol. 40, No.4, pp. 563-574, 2004.

[83] Ong, C., "Model predictive control with reduced number of variables for linear systems with bounded disturbances," *51st IEEE Conference on Decision and Control*, Hawaii, USA, 2012.

[84] Ortner, P., and del Re, L., "Predictive control of a diesel engine air path," *IEEE Transactions on Control Systems Technology*, Vol. 15, No. 3, May 2007.

[85] Pannocchia, G., and Rawlings, J., "The velocity algorithm LQR: a survey," *Texas – Wisconsin Modeling and Control Consortium*, 2001.

[86] Pannocchia, G., and Rawlings, J. B. , "Disturbance models for offset-free model predictive control," *AIChE Journal*, Vol. 49, No. 2, pp. 426–437, 2003.

[87] Pannocchia, G., Gabiccini, M., and Artoni, A., "Offset-free MPC explained: novelties, subtleties, and applications," *Proc. of the 5th IFAC Conference on Nonlinear Model Predictive Control*, Seville, Spain, September 17-20, 2015.

[88] Police, G., and Motori, I., "Downsizing of SI engines by turbo-charging," *Proc. of the 8th Biennial ASME Conference on Engineering Systems Design and Analysis*, Torino, Italy, July 4-7, 2006.

[89] Primbs, J., "The analysis of optimization based controllers," *Proc. of the 1999 American Control Conference*, San Diego, CA, 1999.

[90] Qin, S.J., and Badgwell, T.A., "A survey of industrial model predictive control technology," *Control Engineering Practice*, Vol. 11, pp. 733-764, 2003.

[91] Rakopoulos, C.D., and Giakoumis, E.G, "Diesel Engine Transient Operation," *Springer*, 2009.

[92] Rakovic, S., Grieder, P., Kvasnica, M., Mayne, D.Q., and Morari, M., "Computation of invariant sets for piecewise affine discrete system subject to bounded disturbances," *Proc. of the 43rd IEEE Conference on Decision and Control*, 2004.

[93] Rakovic, S.V. Kerrigan, E.C., Kouramas, K.I., and Mayne, D.Q., "Invariant approximations of the minimal robust positively invariant set," *IEEE Transactions on Automatic Control*, Vol. 50, No. 3, pp. 406-410, 2005.

[94] Rojas, O., Goodwin, G., Seron, M., and Feuer, A., "An SVD based strategy for receding horizon control of input constrained linear systems," *International Journal of Robust and Nonlinear Control*, Vol. 14, No. 13-14, pp. 1207-1226, 2004.

[95] Santillo, M., and Karnik, A., "Model predictive controller design for throttle and wastegate control of a turbocharged engine," *Proc. of the 2013 American Control Conference*, Washington, DC, USA, 2013.

[96] Schmid, C., and Biegler, L.T., "Quadratic programming methods for reduced hessian SQP," *International Journal of Computer Applications in Chemical Engineering*, Vol. 18, No. 9, pp. 817-832, 1994.

[97] Sha, F., Lin, Y., Saul, L.K., and Lee, D.D., "Multiplicative updates for nonnegative quadratic programming," *Neural Computation*, Vol. 19, No. 8, pp. 2004-2031, 2007.

[98] Simon, D., Löfberg, J., and Glad, T., "Nonlinear model predictive control using feedback linearization and local inner convex constraint approximations," *Proc. of the 2013 European Control Conference*, Zürich, Switzerland, 2013.

[99] Smil, V., "Prime Movers of Globalization: The History and Impact of Diesel Engines and Gas Turbines," *MIT Press*, 2010.

[100] Squatriglia, C., "Three is the four as engines downsize," *Wired*, September 16, 2011.

[101] Stewart, G., and Borrelli, F., "A model predictive control framework for industrial turbodiesel engine control," *Proc. of the 47th IEEE Conference on Decision and Control*, Cancun, 2008.

[102] Stotsky, A., and Kolmanovsky, I.V., "Application of input estimation techniques to charge estimation and control in automotive engines," *Control Engineering Practice,* Vol. 10, No, 12, pp. 1371-1383, 2002.

[103] Turner, J.W.G., Popplewell, A., Patel, R. Johnson, T.R., Darnton, N.J., Richardson, S., Bredda, S.W., Tudor, R.J., Bithell, C.I., Jackson, R., Remmert, S.M., Cracknell, R.F., Fernandes, J.X., Lewis, A.G.J., Akehurst, S., Brace, C.J., Copeland, C., Martinez-Botas, R., Romagnoli, A., and Burlaka, A.A., "Ultra boost for economy: extending the limits of extreme engine downsizing," *SAE International,* 2014-01-1185, 2014.

[104] United States Environmental Protection Agency, "EPA and NHTSA finalize historic national program to reduce greenhouse gases and improve fuel economy for cars and trucks," *Office of Transportation and Air Quality, United States Environmental Protection Agency,* EPA-420-F-10-014, April, 2010.

[105] United States Environmental Protection Agency, "EPA and NHTSA set standards to reduce greenhouse gases and improve fuel economy for model years 2017 - 2025 cars and light trucks," *Office of Transportation and Air Quality, United States Environmental Protection Agency,* EPA-420-F-12-051, August, 2012.

[106] United States Environmental Protection Agency, "EPA sets Tier 3 tailpipe and evaporative emission and vehicle fuel standards," *Office of Transportation and Air Quality, United States Environmental Protection Agency,* EPA-420-F-14-008, March, 2014.

[107] United States Environmental Protection Agency and United States Department of Energy, "Where the energy goes: gasoline vehicles," www.fueleconomy.gov/feg/atv.shtml, Retrieved: September 8, 2015.

[108] Van Nieuwstadt, M.J., Kolmanovsky, I.V., and Moraal, P.E., "Coordinated EGR-VGT control for diesel engines: an experimental comparison," *SAE Technical Paper*, SAE 2000-01-0266, 2000.

[109] Van Nieuwstadt, M.J, Kolmanovsky, I.V., Moraal, P.E., Stefanopoulou, A., and Jankovic, M., "EGR-VGT control schemes: experimental comparison for a high-speed diesel engine," *Control Systems, IEEE*, Vol. 20, No. 3, pp. 63-79, 2000.

[110] Van Nieuwstadt, M.J., "Coordinated control of EGR valve and intake throttle for better fuel economy in diesel engines," *SAE Technical Paper*, 2003-01-0362, 2003.

[111] Volkswagen of America, Inc., "Message to our customers from Michael Horn, President and CEO, Volkswagen Group of America," www.vwdieselinfo.com, Retrieved: October 22, 2015.

[112] Walker, K., Samadi, B., Huang, M., Gerhard, J., Butts, K., and Kolmanovsky, I., "Design environment for nonlinear model predictive control," *SAE 2016 World Congress & Exhibition*, 2016, Submitted.

[113] Wahlström, J., and Eriksson, L., "Performance gains with EGR flow inversion for handling non-linear dynamic effects in EGR VGT CI engines," *Proc. of the 5th IFAC Symposium on Advances in Automotive Control*, Pajaro Dunes, CA, 2007.

[114] Wang, L., "A tutorial on model predictive control: using a linear velocity-form model," *Developments in Chemical Engineering and Mineral Processing*, Vol. 12., No. 5-6, pp. 573-614, 2004.

[115] Wang, X., Ortner, P., and Alberer, D., "A design framework for nonlinear predictive engine control," *Proc. of the 2009 IFAC Workshop on Engine and Powertrain Control, Simulation and Modeling*, IFP, Rueil-Malmaison, France, November 30 - December 2, 2009.

[116] Watson, N., and Janota, M.S., "Turbocharging the Internal Combustion Engine," *John Wiley & Sons Inc.*, September, 1982.

[117] Zsiga, N., Voser, C., Onder, C., and Guzzella, L., "Intake manifold boosting of turbocharged spark-ignited engines," *Energies*, Vol. 6, pp. 1746-1763, 2013.