Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

Model Predictive Emissions Control of a Diesel Engine Airpath: Design and Experimental Evaluation

Dominic Liao-McPherson*[1] | Mike Huang[2] | Shinhoon Kim[2] | Masanori Shimada[3] | Ken Butts[2] | Ilya Kolmanovsky[1]

[1]Department of Aerospace Engineering, University of Michigan, Michigan, USA
[2]Toyota Technical Center, Toyota Motor North America, Michigan, USA
[3]Higashi-Fuji Technical Centre, Toyota Motor Corporation, Shizuoka, Japan

Correspondence
*Dominic Liao-McPherson. 1221 Beal Avenue, Ann Arbor, MI 48109. Email: dliaomcp@umich.edu

Present Address
Department of Aerospace Engineering, University of Michigan, 1221 Beal Avenue, Ann Arbor, MI 48109

Summary

This paper presents the development and experimental validation of an emissions oriented model predictive controller for a diesel engine. The control objective is to minimize cumulative NOx and hydrocarbon emissions while limiting visible smoke production and without compromising fuel economy or torque response. This is accomplished by using a supervisory model predictive controller (SMPC) and nonlinear model predictive controller (NMPC) in tandem. The SMPC controller coordinates the exhaust gas recirculation (EGR) rate target and fuel supplied to the engine in real-time to satisfy combustion quality constraints while the NMPC controller tracks the EGR rate target by manipulating the EGR throttle, EGR valve, and variable geometry turbine (VGT). The NMPC controller uses MPC for feedforward and feedback in a novel configuration to simultaneously achieve fast tracking performance, disturbance rejection, and robustness. We demonstrate that the proposed diesel engine MPC controller is able to reduce cumulative emissions by 10-15% relative to a state of the art benchmark strategy when placed in closed-loop with an engine on a transient dynamometer.

KEYWORDS:
Nonlinear Model Predictive Control, Automotive Applications, Diesel, Constrained Control, Experimental Validation.

ACRONYMS

EGR	Exhaust Gas Recirculation
VGT	Variable Geometry Turbocharger
MAP	Manifold Air Pressure
MAF	Mass Air Flow
MPC	Model Predictive Control
SMPC	Supervisory Model Predictive Control
NMPC	Nonlinear Model Predictive Control
LPV	Linear Parameter Varying
OCP	Optimal Control Problem
DAP	Diesel Air Path
NOx	Oxides of Nitrogen
THC	Total Hydrocarbon
PM	Particulate Matter
QP	Quadratic Program
FF	Feedforward
FB	Feedback
WLTC	Worldwide Harmonized Light Vehicles Test Cycle
NEDC	New European Drive Cycle
BGF	Burnt Gas Fraction
LTI	Linear Time Invariant
LAPACK	Linear Algebra Package
BLAS	Basic Linear Algebra Subroutines
ECU	Engine Control Unit
CPU	Central Processing Unit

NOMENCLATURE

q, q^{trg}	Fueling rate and fueling rate target
N_e	Engine speed
$\rho = [q^{trg} \quad N_e]$	Engine operating condition: target fueling rate and engine speed
$p_{im}, p_{im}^{trg}, p_{ex}$	Intake manifold pressure, intake manifold pressure target, and exhaust manifold pressure
χ, χ^{trg}	EGR rate and EGR rate target
$w_c, w_f, w_{egr}, w_{cyl}$	Compressor, fuel, EGR, and cylinder mass flow rates
F_1, F_2	Intake and exhaust manifold burnt gas fractions
$u_{thr}, u_{val}, u_{vgt}$	EGR throttle, EGR valve, and variable geometry turbocharger (VGT) positions
τ_q, τ_w	Engine and wheel torque
$\psi_{THC}, \psi_{NO_x}, \psi_{OP}$	Exhaust gas THC concentration, NOx consecration, and opacity
ϕ	Fuel-air ratio
N_s, N_{fb}, N_{ff}	Supervisory, feedback, and feedforward prediction horizon lengths
$R_s, \alpha_s, \gamma_s, \beta_s$	Supervisory controller tuning parameters
$Q_{fb}, R_{fb}, Q_{ff}, R_{ff}$	Feedback and feedforward positive definite weighting matrices
$\bar{p}_{im}(\rho), \bar{\chi}_{im}(\rho)$	Steady state intake pressure and Exhaust Gas Recirculation (EGR) rate setpoints as a functions of engine operating condition
s_{veh}	Longitudinal speed of the vehicle
V_{im}, V_{ex}	Intake and exhaust manifold volumes
T_{im}, T_{ex}	Intake and exhaust manifold temperatures
R_{air}, R_{ex}	Gas constants of fresh air and exhaust gases
$(A/F)_s$	Stoichiometric air-fuel ratio

1 | INTRODUCTION

Increasingly stringent emissions and fuel economy regulations have created a need for more complex engine systems and advanced engine control strategies. Diesel engines offer superior fuel economy compared to their gasoline counterparts; however sophisticated strategies are needed to manage oxides of nitrogen (NO_x) and particulate matter (PM) emissions to meet regulatory standards. Many modern diesels incorporate a variable geometry turbocharger (VGT) to improve power output and fuel efficiency, and an external exhaust gas recirculation (EGR) system to reduce emissions. However, the addition of both the turbocharger and EGR systems into the engine design introduces strong nonlinearities and interactions, complicating control development.

Emissions control in automotive diesels is especially challenging. Unlike in e.g., marine or generator applications, load and engine speed are highly transient, leading to frequent emissions spikes which must be carefully managed. In particular, smoke control is strongly coupled to torque response (drivability) and there is a strong tradeoff between transient NO_x and smoke production. Model predictive control (MPC) provides a useful framework for managing these tradeoffs via real-time constraint enforcement. MPC [1,2] is a control technique that computes a control action at each sampling instant by solving an optimal control problem (OCP) over a finite receding horizon. MPC is of increasing interest in the automotive and engine control communities [3,4] because it can systematically handle constraints, nonlinearities, and achieve fast transient responses.

Existing literature on diesel engine MPC can be broadly divided into two classes, (i) those with airflow/pressure setpoint tracking formulations and (ii) those that consider high level objectives such as fuel economy or emissions.

Prior research on pressure and/or airflow setpoint tracking using linear MPC often uses switching or piecewise linear models to cover the engine operating range [5,6,7,8,9]. This approach may lead to chattering and poor performance; multiple estimator/controllers running in parallel are often needed to overcome these issues, increasing the computational footprint of the controller. To avoid the use of multiple models, the use of partial nonlinear inversion in combination with rate-based MPC [10] or disturbance observers [11] has been proposed to allow a single linear MPC controller to cover the entire operating range. However, the use of a single controller leads to reduced model accuracy and performance. Nonlinear MPC (NMPC) can incorporate more accurate models and improve performance while avoiding the need for nonlinear inversion and/or multiple linear models. Due to the high computational complexity of nonlinear optimization, most literature on the topic involves only simulation studies [12,13,14]. However, Murilo et al. [15] present experimental results which demonstrate good tracking performance with an NMPC controller over a drive cycle but does not offer comparisons with industrial benchmarks (e.g., existing mass production controllers) or insight into how the NMPC controller impacts emissions or fuel economy.

MPC also is capable of directly optimizing or constraining high level/economic objectives such as emissions or fuel economy, these outputs are nonlinear and thus are usually addressed with NMPC. Controllers that maximize fuel economy subject to emissions constraints have been proposed for heavy duty diesels [16,17], off-highway diesels [18] and marine/generator diesels [19]. The associated simulation and experimental results are promising, however, these varieties of diesel engines are not subject to the same transient operating condition (speed/load) variations as automotive diesel engines. Karlsson et al. [20] propose an MPC controller that directly considers high level objectives. However, they use cylinder pressure sensors which are not available in production vehicles, linear models resulting in poor closed-loop performance, and present results only for a single operating point rather than complete driving cycles.

In summary, Linear MPC controllers are not well suited to optimizing high level objectives in diesel engines, such as those related to emissions and fuel economy, due to the nonlinearity of these outputs. At the same time, implementing nonlinear MPC controllers for automotive diesels remains challenging due to fast sampling rates and the high computational complexity of nonlinear optimization. Moreover, most experimental results for diesel engine MPC controllers focus either on step responses, rather than realistic driving patterns, or do not consider how the NMPC controller impacts high level objectives. In particular, comprehensive experimental emission control results, i.e., considering realistic driving patterns, using only production sensors, and including comparisons with existing industrial benchmarks, have not been presented for automotive diesel engines.

In this paper, we develop a novel nonlinear model predictive control framework for emissions coordinated constrained control of a diesel engine. The framework involves an inner-loop nonlinear tracking-type MPC controller for coordinated control of engine airpath actuators (EGR valve, VGT, throttle) and a supervisory outer-loop MPC/reference governor hybrid that manipulates the EGR rate setpoint and fuel input to enforce emissions constraints. Our contributions are as follows:

1. We propose an inner-outer loop MPC based framework that enables systematic design of constrained controllers for the diesel engine system, rather than using heuristic or rule based design as is current industrial practice. For the outer-loop, we propose a novel supervisory MPC (SMPC) controller that is able to enforce pointwise-in-time emissions constraints and for

the inner-loop we develop a controller that uses a novel NMPC based feedforward/feedback architecture to simultaneously achieve fast transient responses and integral action. Theoretical closed-loop stability results supporting the controller are also included.

2. We experimentally verify that our integrated SMPC/NMPC controller is able to reduce engine-out NOx and hydrocarbon emissions by $10-15\%$ compared to a state-of-the-art industrial benchmark controller over a transient driving cycle. Further, we demonstrate visible smoke control during fast transients which is both challenging and specific to automotive diesel applications. This is accomplished using only production sensors which implies that our controller can be deployed on existing engines without hardware modifications.

3. We demonstrate that it is possible to implement real-time capable high performance NMPC controllers at greater than 100 Hz sampling rates by leveraging recent advances in computational algorithms for MPC[21] and symbolic tools for automatic code generation[22]. In particular, our inner-loop NMPC controller is able to execute more than 5 times faster than the most comparable automotive diesel NMPC controller in the literature[15].

While developed here for diesel engine applications, we believe that our hierarchical control framework is more broadly applicable e.g., for control of other internal combustion engines and other industrial applications. Demonstrations of this kind are a crucial step in the transition of NMPC for diesel engines from academic research to industrial practice.

This paper builds upon our previous work. The SMPC controller extends the economic MPC controller of Liao-McPherson et al.[23] and the NMPC controller combines the exterior penalty approach of Huang et al.[12] and a nonlinear extension of the rate-based formulation presented by Huang et al.[10] for linear MPC. This paper integrates the SMPC and NMPC modules and presents experimental results for the nonlinear case; our previous publications[23,12] only contain simulation results. An overview of the project, including a high level description of the NMPC controller, is also available[24].

A model predictive controller is fundamentally based on three constituents: (i) Prediction models used to estimate the response of the system to a control action (and any associated estimators), (ii) An optimal control problem (OCP) formulation, and (iii) a method for solving the aforementioned OCP in real-time. The rest of Section 1 provides some background on engine control, describes the engine system and outlines the control objectives. Section 2 describes the prediction models used in the SMPC and NMPC controllers. Section 3 describes the control architecture in detail, as well as the optimal control problems defining the SMPC and NMPC controllers. Section 4 describes how the SMPC and NMPC optimal control problems are solved in real-time. Section 5 contains experimental results and analysis demonstrating the performance and reliability of the proposed strategy. Finally, Sections 6 and 7 discuss future perspectives and concluding remarks.

1.1 | Diesel Powertrain Overview

We begin with a brief description of the diesel powertrain operation, see Figure 1. The driver inputs a pedal angle command that is mapped to a fuel request q^{trg}. In response to fuel input, the engine produces torque which is applied to the transmission input shaft and transmitted to the wheels through the transmission and differential. The engine also produces emissions as byproducts which are heavily regulated. In modern vehicles, engine exhaust gases pass through an aftertreatment system, see e.g., Recsitouglu et al.[25] for more details, before being emitted into the atmosphere. From the perspective of the engine there are two inputs, the fueling rate command (q^{trg}) and the engine speed (N_e). The fueling rate command represents a torque request from the driver and the engine speed is determined by the powertrain, i.e., it is a function of the vehicle speed and the transmission gear ratio. The outputs are the engine torque applied to the transmission input shaft (τ_q), the NO_x (ψ_{NO_x}), and total hydrocarbon (THC) (ψ_{THC}) concentrations, and the exhaust gas opacity (ψ_{OP}). More details on powertrain modelling and control can be found in the literature[26,27].

1.2 | System Description

A schematic of the diesel engine is shown in Figure 2. The engine consists of a cylinder block, intake and exhaust manifolds, an external EGR system, and a turbocharger. The Manifold Air Pressure (MAP) and Mass Air Flow (MAF) sensors are used to measure the intake manifold pressure and compressor flow, respectively. The exogenous inputs to the system are the fuel target and the engine speed. Fluid flows through the engine are controlled by the EGR throttle, EGR valve and variable geometry turbocharger (VGT). The EGR valve controls the amount of exhaust gas allowed to flow into the intake manifold; adding exhaust gas into the intake manifold reduces peak combustion temperature in the cylinders, lowering NO_x concentrations, but

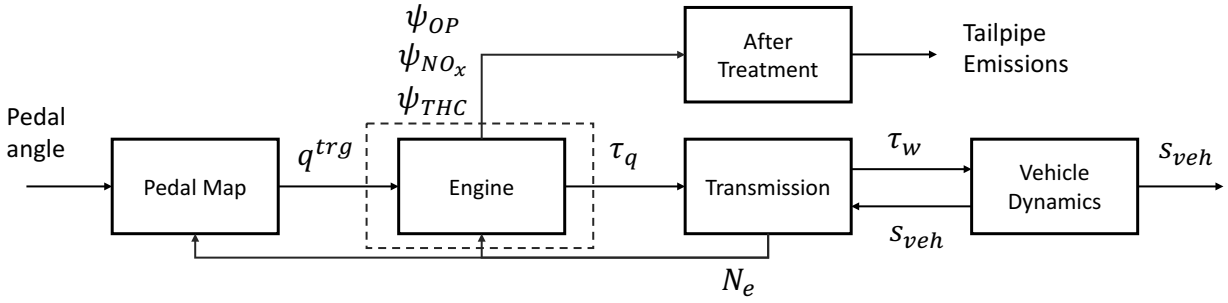

FIGURE 1 A high level diagram illustrating the architecture of a typical diesel powertrain.

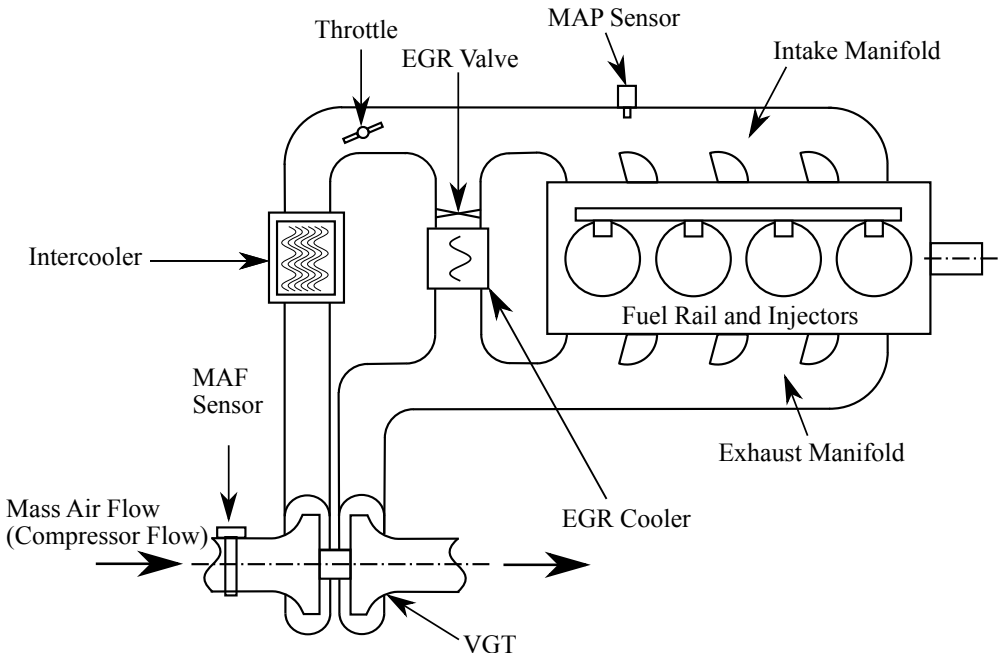

FIGURE 2 A diagram of the diesel engine airpath.

can cause poor quality combustion, leading to vibration, roughness, and smoke. The EGR throttle is used to reduce airflow into the intake manifold allowing for a higher proportion of exhaust gases when it is beneficial. The VGT influences the pressure in the intake manifold by controlling how much energy is extracted from the exhaust gases. Higher intake manifold pressure increases air flow into the cylinders, increasing engine power. In this paper, we consider the total fueling rate as a control input; an external injection strategy is used for the rest of the fuel path control.

The engine operating condition consists of the fueling rate[†] target and the engine speed, i.e.,

$$\rho = [q^{trg} \;\; N_e]^T. \tag{1}$$

The state vector for the supervisory controller consists of the intake pressure, exhaust pressure, compressor flow, intake manifold burnt gas fraction, and exhaust manifold burnt gas fraction and the control vector consists of the EGR rate target passed to the

[†]The fueling rate is the ratio of fuel mass flow and engine speed, is proportional to torque (when losses are neglected), and has units of volume per stroke.

inner-loop controller and the fueling rate command applied to the engine i.e.,

$$x = \left[p_{im}\ p_{ex}\ w_c\ F_1\ F_2\right]^T, \text{ and } u = \left[\chi^{trg}\ q\right]^T. \tag{2}$$

The EGR rate is defined as the mass flow through the EGR valve into the intake manifold over the total mass flow into the cyinders, i.e., $\chi = w_{egr}/w_{cyl}$.The fueling rate applied to the engine q may be different than the target q^{trg} in order to enforce constraints. The state vector of the supervisory controller x can be partitioned into airpath and EGR loop variables, denoted by

$$\xi = \left[p_{im}\ p_{ex}\ w_c\right]^T, \text{ and } \Upsilon = \left[F_1\ F_2\right]^T, \tag{3}$$

respectively so that $x = [\xi^T\ \Upsilon^T]^T$. The state vector for the airpath controller is the intake pressure and EGR rate, the control inputs are the throttle, EGR valve, and VGT positions, i.e.,

$$z = \left[p_{im}\ \chi\right]^T, \text{ and } v = \left[u_{thr}\ u_{val}\ u_{vgt}\right]^T. \tag{4}$$

The system outputs are split into measurements, and performance variables, defined respectively as,

$$y_m = \left[p_{im}\ w_c\ N_e\right]^T \text{ and } y_p = \left[\psi_{OP}\ \psi_{NO_x}\ \psi_{THC}\ \tau_q\right]^T. \tag{5}$$

The measurements, y_m, are used for feedback. The performance outputs: torque, exhaust opacity (smoke), NO_x concentration, and total hydrocarbon concentration (THC) (strongly correlated with particulate matter), are measured for evaluation purposes but are not available for feedback. The engine is also instrumented to measure the exhaust pressure p_{ex}, the intake and exhaust manifold temperatures T_{im}, T_{ex} and the exhaust manifold burnt gas fraction F_2. These measurements are used for model validation and calibration but not for performance evaluation or feedback.

Remark 1. The dimension of the supervisory controller state vector is larger than that of the airpath controller, the extra states are needed to model emissions.

1.3 | Objectives and Methodology

The objective of diesel airpath control is to supply the torque requested by the driver while maximizing fuel economy, respecting regulatory constraints on NO_x and particulate matter (PM), and limiting visible smoke. The focus of this paper is emissions reduction, our methodology is to use the constraint handling capabilities of MPC to reduce conservatism and operate closer to constraint boundaries during transients to reduce emissions. Specifically, there is a tradeoff between NOx and smoke emissions during transients which is controlled by the EGR valve. Opening the valve and increasing EGR flow decreases NO_x production (by reducing peak combustion temperatures[28]) but reduces available oxygen and increases smoke production. The conventional strategy for managing this tradeoff is to shut the EGR valve when fuel/torque is requested by the driver to suppress smoke, leading to a transient spike in NO_x emissions. Our controller uses dynamic models of the engine to permit more EGR flow while still preventing visible smoke production and thus reduce overall emissions.

Figure 3 provides a high level overview of the controller architecture, a more detailed diagram can be found in Figure 7. The motivation behind the design is to decouple nonlinearity compensation and integral action from constraint satisfaction. This reduces the complexity of the controller as well as making the design modular (an important consideration in industrial settings).

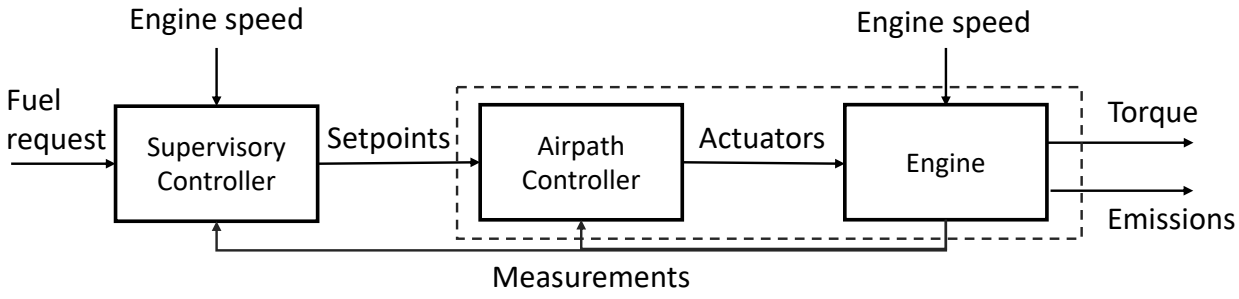

FIGURE 3 A high level control architecture diagram.

The controller consists of an outer-loop supervisory module and an inner-loop airpath module. The inner-loop airpath controller compensates for nonlinearities and coordinates multiple actuators. It also incorporates integral action to account for variability and noise in the engine. The supervisory controller enforces emissions constraints while keeping the system as close as possible to preselected operating points. It uses a dynamic model of the system to do so in a minimally conservative manner, allowing safe operation near the constraint boundaries and improving performance, i.e., reducing emissions. This approach is well established in the context of reference governors[29] and has been successfully applied to enforce compressor surge margin constraints in fuel cell systems[30], automotive gasoline engines[31], and aircraft gas turbines[32].

In what follows, we detail the controller development steps which encompass engine modelling, control design, and experimental validation.

2 | ENGINE MODELLING

This section describes the prediction models used by the various MPC controllers. First we describe the data driven open-loop engine models, the closed-loop data driven models of the engine in closed-loop with the NMPC controller, and finally the physics based burnt gas fraction models. The first is used by the inner-loop NMPC controller while the latter two are used by the outer-loop MPC controller.

Throughout the paper, we make use of a 2 dimensional grid of operating conditions,

$$\begin{bmatrix} \rho_1 & \rho_{15} & \cdots & \rho_{141} \\ \rho_2 & \rho_{16} & \cdots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ \rho_{14} & \cdots & \cdots & \rho_{154} \end{bmatrix} = \begin{bmatrix} (q_1^{trg}, N_{e1}) & (q_1^{trg}, N_{e2}) & \cdots & (q_1^{trg}, N_{e11}) \\ (q_2^{trg}, N_{e1}) & (q_2^{trg}, N_{e2}) & \cdots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ (q_{14}^{trg}, N_{e1}) & \cdots & \cdots & (q_{14}^{trg}, N_{e11}) \end{bmatrix} \tag{6}$$

which are contained within the box $[q_1^{trg}, q_{14}^{trg}] \times [N_{e,1}, N_{e,11}]$ and are used for scheduling gains, targets, and model parameters. This grid was chosen so as to cover the test cycles used for experimental validation[‡]. We use a dense uniform [§] grid, the gridpoint density was selected by decreasing the gridpoint spacing until the modelling accuracy ceased improving, i.e., until the point of diminishing returns.

2.1 | Open-loop Airpath Modelling

The open-loop airpath model is used by the NMPC controller to estimate the response of the engine to throttle, EGR valve, and VGT commands. We use a data-driven approach to generate models of the EGR rate and intake pressure response to throttle, valve, and VGT actuation. First, we perturb the system around each operating point to generate identification data. Then, at each operating point, ρ_i, for $i = 1, \ldots, 154$, we identify a model of the form,

$$z_{k+1} = \tilde{F}_i + \tilde{A}_i z_k + \tilde{B}_i v_k + \omega_i^T \Phi(y_k), \quad y_k = [z_k^T \; v_k^T]^T, \tag{7}$$

where k is the discrete time index, $\tilde{A}_i$, $\tilde{B}_i$, $\tilde{F}_i$, and ω_i are appropriately sized vectors/matrices of coefficients and $\Phi(x)$ is a basis for $\mathbb{P}_n^2$, the set of all polynomials[¶]. in n variables of degree > 1 and ≤ 2, e.g., $\mathbb{P}_2^2 = span\left\{x_1^2, x_1x_2, x_2^2\right\}$. To ensure the model has correct local properties, the linear portion of the model is identified first using least squares. The nonlinear portion is then identified using the error signal $e_k = z_{k+1} - \tilde{F}(\rho_i) + \tilde{A}(\rho_i)z_k + \tilde{B}(\rho_i)v_k$. Finally, to obtain a model that depends continuously on the operating condition ρ, the coefficients are interpolated between gridpoints using linear interpolation, resulting in a nonlinear parameter varying model of the form

$$z_{k+1} = f_n(z_k, v_k, \rho_k) = \tilde{F}(\rho_k) + \tilde{A}(\rho_k)z_k + \tilde{B}(\rho_k)v_k + \omega(\rho_k)^T \Phi(y_k), \quad y_k = [z_k^T \; v_k^T]^T, \tag{8}$$

where $\tilde{F} : \mathbb{R}^2 \to \mathbb{R}^2$, $\tilde{A}, \tilde{B} : \mathbb{R}^2 \to \mathbb{R}^{2\times 2}$ and $\omega : \mathbb{R}^2 \to |\mathbb{P}_4^2|$. Figure 4 illustrates the result of the fitting procedure and compares the nonlinear model with the same model using only the linear terms. Note that the operating point is varying, demonstrating good model accuracy between gridpoints. In our experience, using interpolation techniques avoids some issues typically associated

[‡]If the controller described in this paper was to be used in a production vehicle the grid would be expanded to cover the entire engine operating region.

[§]Note that it would be more efficient to adapt the spacing of the gridpoints so the density is higher where the dynamics change more rapidly. However, because we are able to gather data quickly, we found the effort required to implement an adaptive scheme was unnecessary.

[¶]Other functions, e.g., Gaussian, neural network, can be used but they will inevitably be more complex to differentiate and/or numerically evaluate.

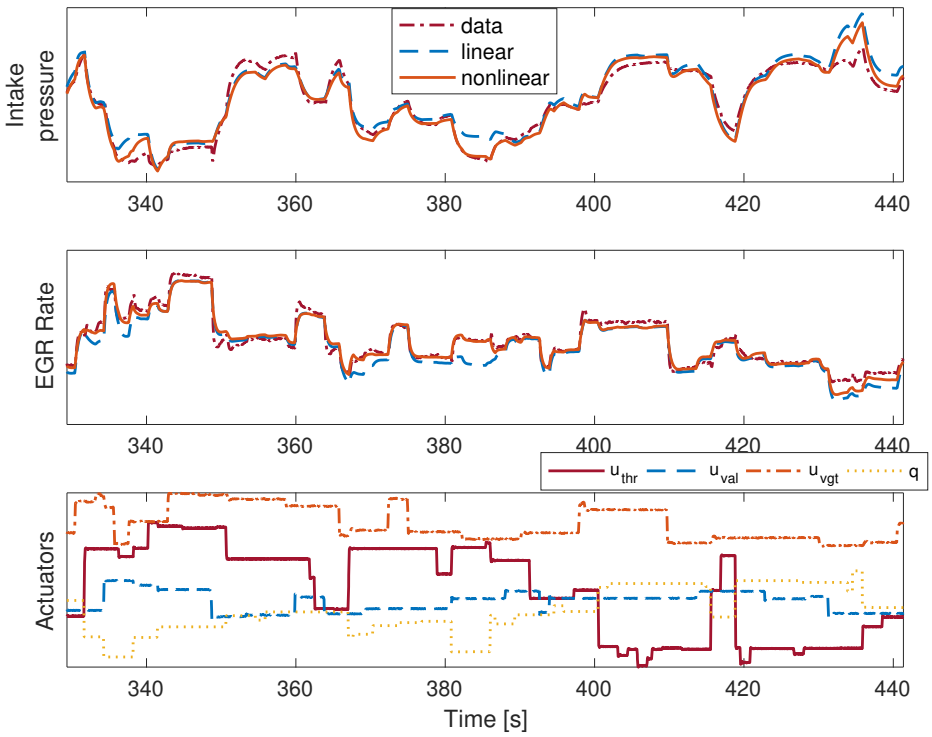

FIGURE 4 Validation of the nonlinear open-loop airpath model at 1800 rpm. The model output matches the validation data well and the inclusion of the nonlinear terms improves overall accuracy.

with region switching strategies e.g., chattering, discontinuity of the feedback signal etc., and does not significantly affect the computational cost of the resulting controllers.

2.2 | Closed-loop Airpath Modelling

The closed-loop airpath prediction model is used by the outer-loop SMPC controller to estimate the response of the inner-loop to EGR target and fueling rate commands. The model is identified from experimental data. We use a linear parameter varying (LPV) model, identified using the local LPV modelling approach[33] in the same manner as the open-loop models. The form of the model is given by:

$$\xi_{k+1} = F(\rho_k) + A(\rho_k)\xi_k + B(\rho_k)u_k, \tag{9}$$

where k is the discrete time index, $A : \mathbb{R}^2 \to \mathbb{R}^{3\times3}$, $B : \mathbb{R}^2 \to \mathbb{R}^{3\times2}$, and $F : \mathbb{R}^2 \to \mathbb{R}^3$ are operating condition dependent matrices. To construct A, B and F, coefficient matrices A_i, B_i and F_i are identified using experimental data at each ρ_i in the grid (6) for all $i = 1, \ldots, 154$ using least squares in the same manner as described in Section 2.1. Linear interpolation is used to compute model coefficients between grid-points. Figure 5 illustrates a typical outcome of the fitting process. The model quality is good over most of the operating range since the inner loop controller tends to "linearize" the closed loop plant from the perspective of the SMPC controller.

The SMPC controller uses models of the closed inner loop, see Figure 7 and Remark 2. To accommodate this, the NMPC controller is calibrated before the data for the SMPC models is gathered, see Section 4.4 for more details. We have observed that the closed-loop performance of the SMPC controller is not very sensitive to the accuracy of the closed-loop airpath model, likely because the presence of the inner loop controller makes the closed-loop airpath dynamics relatively benign. We were able to obtain good performance as long as the time constants of the modelled states were of the correct order of magnitude.

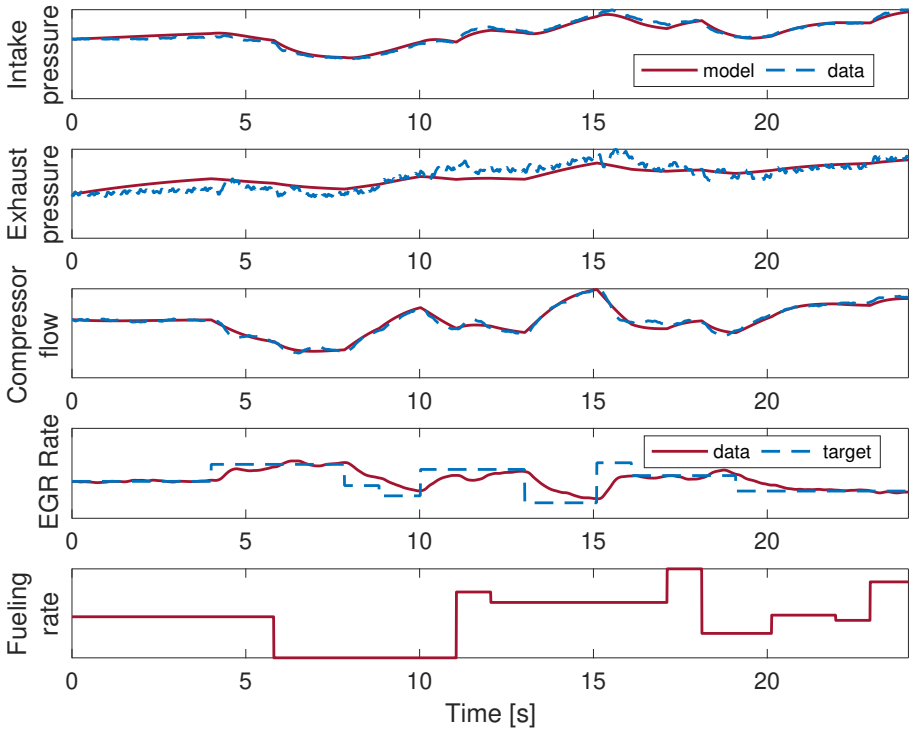

FIGURE 5 Fitting result for a local LTI model of the closed-loop airpath at 1400 rpm. The inclusion of the inner-loop controller leads to approximately linear closed-loop response and, as a result, the linear model is able to accurately capture the system response.

2.3 | Burnt Gas Fraction Modelling

The burnt gas fraction (BGF) model is used by the SMPC to predict the emissions response of the system by tracking the time evolution of the burnt gas fractions in the intake manifold, F_1, and exhaust manifold, F_2. We use the model from Kolmanovsky et al.[34]. Expressed in the form $\dot{\Upsilon} = G(\xi)\Upsilon + b(\xi)q$, the BGF equations can be written as

$$\begin{bmatrix}\dot{F}_1\\ \dot{F}_2\end{bmatrix} = \begin{bmatrix}\frac{-(w_{egr}+w_{th})}{m_1} & \frac{w_{egr}}{m_1}\\ \frac{w_{cyl}}{m_2} & \frac{-(w_{cyl}+w_f)}{m_2}\end{bmatrix}\begin{bmatrix}F_1\\ F_2\end{bmatrix} + \begin{bmatrix}0\\ \frac{1+(A/F)_E}{m_2}\end{bmatrix} c_q N_e q, \quad m_1 = \frac{p_{im}V_{im}}{R_{air}T_{im}}, \quad m_2 = \frac{p_{ex}V_{ex}}{R_{ex}T_{ex}}, \tag{10}$$

where c_q is the constant such that $w_f = c_q N_e q$, R_{air} is the gas constant of air, R_{ex} is the gas constant of the exhaust gas, and V_{im}, V_{ex}, T_{im}, and T_{ex} are the volumes and gas temperatures in the intake and exhaust manifolds. The effective air-fuel ratio, denoted by $(A/F)_E$, quantifies the mass of oxygen consumed per unit fuel and is calibrated as a function of operating condition using exhaust analyzer data. The cylinder flow is estimated as a linear, operating condition dependent function of intake pressure, i.e.,

$$w_{cyl} = a(\rho)p_{im} + b(\rho), \tag{11}$$

the EGR flow is estimated as $w_{egr} = w_{cyl} - w_c$, and the throttle flow, w_{th}, is assumed to be equal to the compressor flow. The BGF equations are stiff so we discretize them using the implicit Euler integration scheme[35]. Since the equations are linear in Υ the update equation can be determined analytically as

$$\Upsilon_{k+1} = (I_{2\times 2} - \Delta\tau_k G(\xi_k, \rho_k))^{-1}(\Upsilon_k + \Delta\tau_k b(\xi_k)\, q_k), \tag{12}$$

where k is the discrete time index, $\Delta\tau_k$ is the integration step size and $I_{2\times 2}$ is an identity matrix. In Figure 6 we compare the BGF model, under the temperature, EGR flow, and throttle flow assumptions, against measurements from a wideband oxygen concentration sensor, situated immediately downstream of the VGT. The model is in good agreement with the measurements; the placement of the sensor induces a filtering effect which accounts for the error in transients.

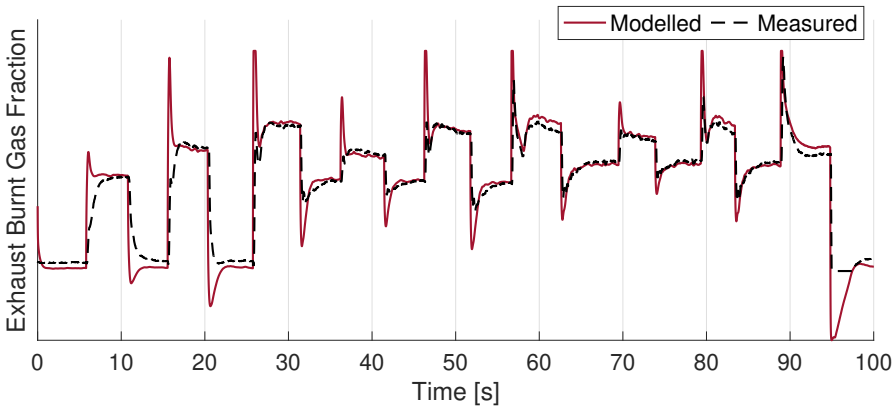

FIGURE 6 Comparison of the burnt gas fraction model against experimental data obtained from a wideband oxygen concentration sensor downstream of the VGT. The model is in good agreement with the measurements; the placement of the sensor induces a filtering effect which accounts for the error in transients.

3 | CONTROL DESIGN

The overall architecture of the controller is shown in Figure 7. The outer loop SMPC controller generates an EGR rate target and a fueling rate input. The fueling rate input is applied directly to the engine while the EGR rate target is passed to the NMPC controller. The NMPC controller tracks EGR rate and intake pressure commands.

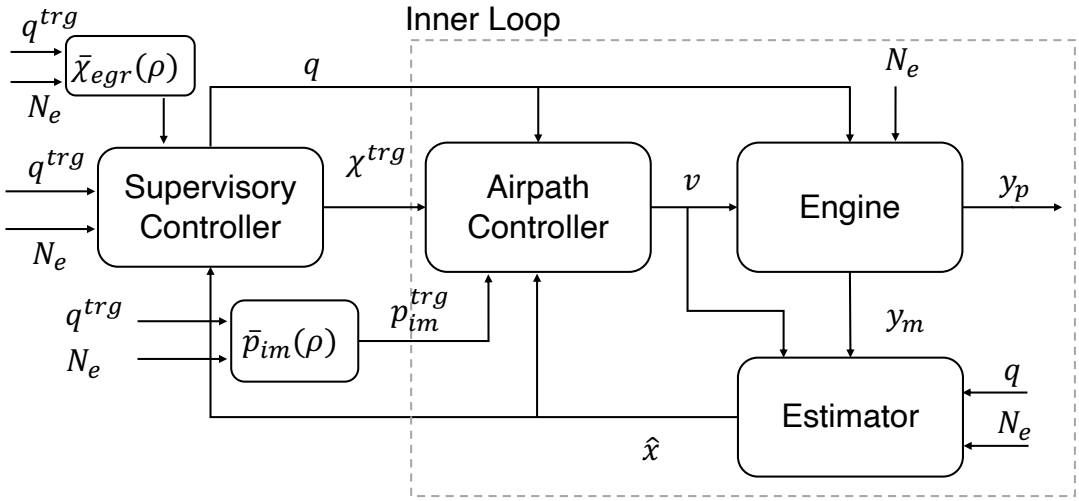

FIGURE 7 A schematic of the MPC control architecture; see Section 1.2 for notation. In this paper the supervisory controller is a supervisory MPC (SMPC) controller and the airpath or inner-loop controller is a nonlinear MPC (NMPC) controller.

In typical engine control strategies, the EGR rate and intake pressure targets ($\bar{\chi}_{egr}$ and $\bar{p}_{im}$) are static functions of the operating condition, i.e., $\bar{\chi}_{egr}(\rho) : \mathbb{R}^2 \to \mathbb{R}$ and $\bar{p}_{im}(\rho) : \mathbb{R}^2 \to \mathbb{R}$, and are implemented using lookup tables. These targets are obtained during engine development and are chosen to be "optimal" in steady state. The steady state maps can be chosen based on a variety of objectives e.g., maximizing fuel economy subject to emissions limits as well as other constraints, e.g., maximum temperature, pressure, etc. Note that there is a complicated tradeoff between fuel economy, NO_x , and PM/THC emissions so determining the steady state maps is nontrivial. A variety of methods have been investigated in the literature for this purpose including constrained optimization techniques[36,37], extremum seeking[38], and model based calibration[39]. Engine manufacturers and suppliers also employ sophisticated proprietary methodologies.

In this paper, we assume that these maps are provided and we focus on transient optimization, i.e., using MPC to shape the transient response of the system as it transitions between operating points. The steady state maps usually neglect dynamic effects, e.g., the intake and exhaust manifold filling dynamics, so transient shaping can have an impact on performance. For example, a significant portion of cumulative emissions production occurs in transients[40], see e.g., Figure 19, and drivability depends on the response speed of the system to fuel commands. Moreover, we have observed that remaining as close as possible to the steady-state targets is a very effective strategy[#], this is reflected in the SMPC formulation in Section 3.2 which minimizes deviation of χ^{trg} from $\bar{\chi}_{egr}$ and q from q^{trg} in a manner similar to a reference governor[29]. Remaining near the optimal steady state targets during transient operation provides good performance in practice (as we demonstrate in Section 5). In a previous paper[23], we explored simultaneous setpoint determination and constraint enforcement using Economic MPC. We found that, in practice, determining optimal setpoints online was slow and lead to poor performance. This motivated our current approach of "precomputing" the setpoints, i.e., using the predetermined setpoint maps $\bar{\chi}_{egr}(\rho)$ and $\bar{p}_{im}(\rho)$.

Most inner-outer loop architectures operate at different rates to minimize interactions between the loops. The SMPC and NMPC controllers operate at the same update rate to allow the SMPC controller to respond to driver fuel requests as quickly as supported by the hardware. This minimizes delays between driver commands and the system response leading to a more responsive vehicle. Further, it allows us to decompose constraint handling (SMPC) and nonlinearity compensation/integral action (NMPC) and makes the architecture modular; the inner loop airpath controller can be of any type, including e.g., PID or MPC. However, choice of inner loop controller will affect the performance of the overall system so using a high performance inner loop controller is advantageous. Finally, these simplifications help make real-time execution computationally tractable.

Remark 2. We manage the possibility of interference between the loops by using closed-loop models that account for the inner loop controller as the SMPC prediction models. This technique is well established in the reference governor literature[29].

Remark 3. The NMPC controller tracks an EGR rate target χ^{trg} and an intake pressure target p_{im}^{trg}. The SMPC controller controls χ^{trg} which is passed to the NMPC controller and q which is applied directly to the engine; it does not control p_{im}^{trg} which is instead obtained from the optimal steady state map, i.e., $p_{im}^{trg} = \bar{p}_{im}(\rho)$. We chose this architecture because the SMPC controller is focused on combustion quality control and combustion quality is most sensitive to χ and q. Future work will include extending the SMPC controller to also control the intake pressure target.

Remark 4. The controller presented in this paper does not use cylinder pressure, NO_x , opacity, or oxygen concentration sensors for feedback. As a result, it can be implemented using only sensors available in a standard production vehicle.

3.1 | Estimator Design

The EGR flow is estimated using a steady state mass balance equation, $w_{egr} \approx w_{cyl} - w_c$, and the cylinder flow is estimated as a function of the operating condition and intake pressure using a static regression map. The EGR rate is calculated as $\chi = \max(0, w_{egr}/w_{cyl})$. The burnt gas fractions are obtained by propagating (10). The normalized fuel-air ratio is estimated as

$$\phi(x, u, \rho) = \frac{w_f}{w_{cyl}(p_{im}, \rho)(1 - F_1)} \left(\frac{A}{F}\right)_s, \tag{13}$$

where $(A/F)_s$ is the stoichiometric air-fuel ratio of the fuel. The cylinder flow is a function of p_{im} and ρ as described in (11) and $w_f = c_q N_e q$ where $c_q > 0$ is a fixed constant that depends on the engine. The normalized fuel-air ratio is strongly correlated with smoke production as illustrated in Figure 8.

The intake temperature, exhaust temperature and the exhaust pressure estimates are obtained from the engine control unit (ECU). Some literature on constructing these estimators includes[41,42,43].

3.2 | Supervisory Controller Design

The objective of the supervisory layer is to enforce safety, and fuel-air ratio constraints to limit smoke during transients as unobtrusively as possible. In this context, safety is ensured by an upper bound on the fuel input and EGR rate to prevent damage to the engine. The following optimal control problem is solved at each sampling instance for the fueling rate and EGR target. The index i runs over the prediction horizon while the index k indicates the sampling instance; the notation $x_{i|k}$ indicates the

[#]We hypothesize that this is because emissions formation occurs in the cylinders at a fast timescale and is essentially a quasi-static function of the airpath states.

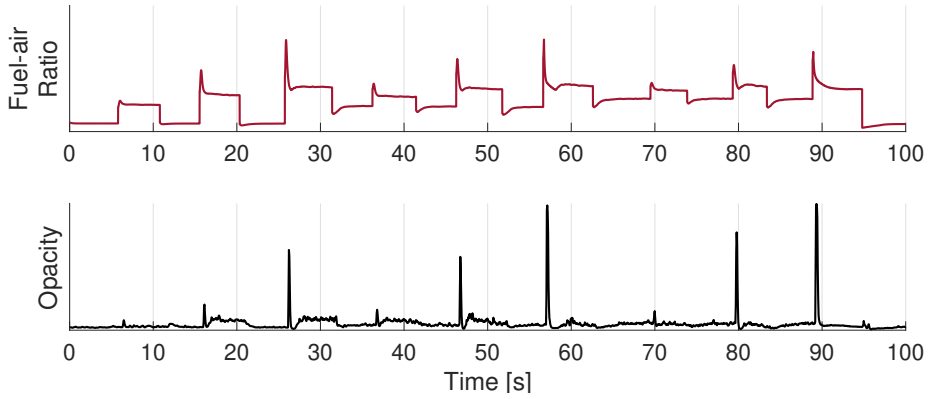

FIGURE 8 The normalized fuel-air ratio is strongly correlated with smoke.

predicted value of x at the ith step in the prediction horizon at time t_k.

$$\min_{s,u,x} \quad J(s,u,\rho_k) = \sum_{i=0}^{N_s} l(u_{i|k}, u_{i-1|k}, \rho_k, s), \tag{14a}$$

$$\text{s.t} \quad x_{i+1|k} = f_s(x_{i|k}, u_{i|k}, \rho_k, \Delta\tau_i),\ i = 0,\ \dots, N_s - 1, \tag{14b}$$

$$\phi(x_{i|k}, u_{i|k}, \rho_k) - \phi_l(\rho_k, w_{c,k}) \le s,\ i = 0,\ \dots, N_s - 1, \tag{14c}$$

$$0 \le \chi^{trg}_{i|k} \le \bar{\chi}_{egr}(\rho_k),\ i = 0,\ \dots, N_s - 1, \tag{14d}$$

$$0 \le q_{i|k} \le q^{trg}_k,\ i = 0,\ \dots, N_s - 1, \tag{14e}$$

$$s \ge 0, \tag{14f}$$

where $u = \begin{bmatrix} u^T_{0|k} & \dots & u^T_{N_s-1|k} \end{bmatrix}^T$, and $x = \begin{bmatrix} x^T_{1|k} & \dots & x^T_{N_s|k} \end{bmatrix}^T$. The quantity $\phi_l(\rho, w_c)$ is the fuel-air ratio limit, it corresponds to when the engine begins to produce visible smoke and is a characteristic of the engine. It is determined experimentally during engine characterization and increases with compressor flow and decreases with engine speed. The fuel-air ratio ϕ is computed using (13). The stage cost function l and prediction model f_s are defined in (15) and (17) given below.

The stage cost function is given by

$$l(u, u^-, \rho, s) = \gamma_s(\chi^{trg} - \bar{\chi}_{egr}(\rho))^2 + \alpha_s(q^{trg} - q) + \beta_s s + ||u - u^-||^2_{R_s},\ \ u = [\chi^{trg}\ \ q]^T, \tag{15}$$

where $\alpha_s, \beta_s, \gamma_s > 0$, and $R_s \succ 0$ are tuning parameters, and reflects tracking objectives for the EGR rate target and fueling rate, a penalty to soften the fuel-air ratio (FAR) constraint to guarantee feasibility, and a damping term ($||x||^2_R = x^T R x$ for $R = R^T \succ 0$). Since the cost function has no dependence on the system outputs, the fuel command and/or EGR rate target are only modified in response to predicted constraint violation, and, as a result, the SMPC controller is a hybrid between an MPC controller and a variant of a reference governor[29]. Together with (14c) and (14f) the slack penalty term, $\beta_s s$, defines an ℓ_1 softened constraint on the fuel-air ratio, which is used to limit smoke. The fuel tracking term, $\alpha(q^{trg} - q)$, which is equivalent to $\alpha_s|q^{trg} - q|$ due to (14e), promotes drivability. The remaining constraints, a lower bound on χ^{trg} and a fueling rate nonnegativity constraint in (14e), make the control constraint set compact. We use linear or 1-norm penalties for both the fuel tracking and fuel-air ratio constraints because they are more robust to ill-conditioning compared to quadratic penalties, ensuring that we are able to reliably solve (14) numerically.

Achieving a sufficiently long prediction horizon to capture the dynamics of interest using a uniform prediction horizon discretization requires a large number of discrete timesteps. Unfortunately, additional timesteps introduce additional decision variables which increases computational complexity. As a countermeasure, we implemented a non-uniform integration timestep in the prediction model. This technique is related to move-blocking[44] and adaptive numerical integration[2]. It was introduced in[45], which also provides stability conditions for the case of continuous time LTI systems. We use the following function to

determine the timestep sizes over the prediction horizon:

$$\Delta\tau_i = \begin{cases} t_s & i \le 2, \\ 6 \cdot t_s & 2 < i \le 4, \\ 40 \cdot t_s & 4 < i \le 8. \end{cases} \tag{16}$$

The total length of the prediction horizon is approximately 1.1 sec and corresponds to $N_s = 8$ steps. The sampling period of the system is approximately 8 msec. The short steps ensure consistency between the model and what is applied, the medium steps capture emissions peaks, and the long steps capture the intake and exhaust pressure responses.

The prediction dynamics (14b) of the supervisory MPC controller are formed by combining the closed-loop airpath and EGR loop models described in Sections 2.2 and 2.3. The prediction model is given by,

$$x_{i+1|k} = f_s(x_{i|k}, u_{i|k}, \rho_k, \Delta\tau_i) = \begin{bmatrix} \xi \\ \Upsilon \end{bmatrix}_{i+1|k} = \begin{bmatrix} F(\rho_k) + \bar{A}(\rho_k, \Delta\tau_i)\xi_{i|k} + \bar{B}(\rho_k, \Delta\tau_i)u_{i|k} \\ [I_{2\times 2} - \Delta\tau_i G(\xi_{i|k}, \rho_k, T_k)]^{-1}(\Upsilon_{i|k} + \Delta\tau_i b q_{i|k}) \end{bmatrix}, \tag{17}$$

which is the concatenation of a downsampled version of (9) and (12). The temperatures $T_k = \begin{bmatrix} T_{im,k} & T_{ex,k} \end{bmatrix}^T$ and operating condition ρ_k are considered constant over the prediction horizon. The downsampled linear model matrices, $\bar{A}(\rho, \Delta\tau)$ and $\bar{B}(\rho, \tau)$ are computed as follows,

$$\bar{A}(\rho, \Delta\tau) = (A(\rho))^{\ell}, \quad \bar{B}(\rho, \Delta\tau) = \left(\sum_{j=0}^{\ell-1} (A(\rho))^j \right) B(\rho), \tag{18}$$

where $\ell = \Delta\tau / t_s$ is the downsampling factor. The constant term F remains unchanged after downsampling.

3.3 | Airpath Controller Design

This section describes the NMPC airpath controller. The NMPC controller presented in this paper includes both feedforward, to provide high performance, and feedback, for robustness, disturbance rejection, and integral action. Both the feedforward and feedback are implemented as separate NMPC controllers. Figure 9 illustrates how they are connected. The feedback NMPC controller uses a rate-based formulation to achieve integral action. The feedforward NMPC controller computes an optimal control action for a nominal plant with a fictitious state $\tilde{z}$. Throughout this section the index i runs over the prediction horizon while the index k indicates the sampling instance; the notation $z_{i|k}$ indicates the predicted value of z at the ith step in the prediction horizon at time t_k, while $\hat{z}_k$ indicates the measured state at time t_k, similarly v_k indicates the control applied to the system at time t_k.

Remark 5. While the airpath controller can be of any type; using a faster controller will lead to better overall performance. In this paper, we use a high performance NMPC controller and show in Section 5 that it outperforms a state of the art in-production airpath controller.

3.4 | Feedback Design

The purpose of the airpath controller is to provide fast tracking of the EGR rate and intake pressure targets computed by the supervisory controller. In addition, due to variations in the engine and plant-model mismatch, integral action is needed to achieve zero offset steady state tracking. We incorporate integral action by employing a rate-based (also known as velocity form) MPC formulation.[‖] The OCP arising from the velocity-form MPC has the following form,

$$\min_{\Delta v} \quad ||z_{N_{fb}} - z_k^{trg}||^2_{Q_{fb}} + \sum_{i=0}^{N_{fb}-1} ||z_{i|k} - z_k^{trg}||^2_{Q_{fb}} + ||\Delta v_{i|k}||^2_{R_{fb}}, \tag{19a}$$

$$\text{s.t.} \quad z_{i+1|k} = z_{i|k} + f_n(z_{i|k}, v_{i-1|k} + \Delta v_{i|k}, \rho_k, \Delta t) - f_n(z_{i-1|k}, v_{i-1|k}, \rho_k, \Delta t), \quad i = 0, \ ..., \ N_{fb} - 1, \tag{19b}$$

$$v_{i|k} = v_{i-1|k} + \Delta v_{i|k}, \quad i = 0, \ ..., \ N_{fb} - 1 \tag{19c}$$

$$z_{0|k} = \hat{z}_k, \quad z_{-1|k} = \hat{z}_{k-1}, \quad v_{-1|k} = \bar{v}_{k-1}, \tag{19d}$$

$$v_{lb} \le v_{i|k} \le v_{ub}, \quad i = 0, \ ..., \ N_{fb} - 1, \tag{19e}$$

[‖] Many strategies for incorporating integral action into MPC are equivalent, see [46,47].

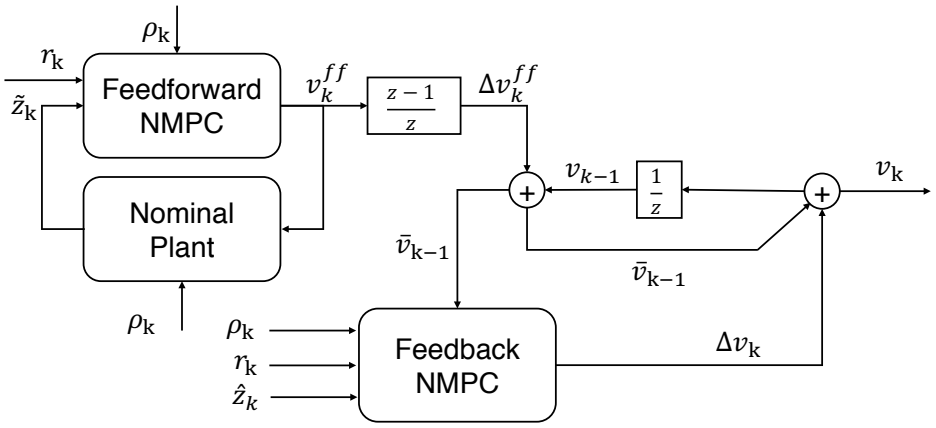

FIGURE 9 A diagram of the feedforward + feedback NMPC architecture. The feedforward NMPC controller is placed in loop with a nominal plant model which is used to generate fictitious measurements, denoted by $\tilde{z}$, as opposed to the real measurements, $\hat{z}$, which are provided to the feedback controller. In the square boxes z stands for the discrete shift operator, i.e., $u_{k+1} = z\,u_k$, and $r = [\chi^{trg} \;\; p_{im}^{trg}]^T$. Note that Δv is computed by the feedback module, v^{ff} is computed by the feedforward module, v is applied to the engine, and $\bar{v}$ is supplied to the feedback module through (19d).

where, $\Delta v = \left[\Delta v_{0|k}^T \; ... \; \Delta v_{(N_{fb}-1)|k}^T\right]^T$, N_{fb} is the receding horizon length, v^{ff} is the control obtained from the feedforward control described in Section 3.5, see Figure 9, v_{ub} and v_{lb} are bounds on the actuator positions, $Q_{fb} \succ 0$, and $R_{fb} \succ 0$ are positive definite weighting matrices, and $f_n(z, v, \rho, \Delta t)$ in (19b) is defined in (8) with the dependence on the sampling period Δt made explicit. The typical choice is to set $\Delta t = t_s$. The variable $\bar{v}_{k-1} = v_{k-1} + v_k^{ff} - v_{k-1}^{ff}$ in (19d) is the nominal control input after feedforward compensation.

To achieve good performance and stability, a common rule-of-thumb for MPC is to use a prediction horizon roughly equal to the slowest time constant of the system. For the DAP these are the pressure dynamics with settling times of approximately 1–2 seconds. With a base sampling rate, t_s, of 8 msec, if $\Delta t = t_s$ a horizon of more than 100 steps would be required, leading to more than 300 optimization variables, which is too large to be solved in real-time for this application. Our strategy for managing the size of the OCP is to utilize move blocking[44]. Typically, move blocking is used to reduce the number of optimization variables in the OCP. However, due to the limited computational resources available in the DAP application, we also seek to reduce the number of nonlinear model evaluations required to propagate the dynamics and to form Jacobian/Hessian matrices.

To this end, our strategy is to increase the discretization timestep of the model, Δt, such that $\Delta t \gg t_s$ while keeping N_{fb} and the form of f_n the same. Note we cannot increase t_s directly because fuel, engine speed, and target input changes, which occur at $t_s \approx 8$ msec, must be accommodated. We parameterize the discretization timestep as an integer multiple of the sampling period, i.e., $\Delta t = d \cdot t_s$; we will refer to $d \in \mathbb{N}$ as the downsampling factor. For the feedback component of the NMPC controller we pick $d = 40$. This approach increases the time horizon of the OCP but creates a mismatch between the sampling period of the controller and discretization timestep of the prediction horizon. It also requires generating models, $f_n(\cdot, \cdot, \cdot, \Delta t)$, with a large sampling time Δt which is challenging in practice due to aliasing effects.

As mentioned above making $\Delta t \neq t_s$ introduces sampling/discretization mismatch. As a result it is no longer clear if we can use $v_{-1} = v(t - \Delta t)$ in (19d) because multiple changes in control would have been applied between times $t - \Delta t$ and t. Furthermore, this is severely inconsistent with the model, (19b), which assumes that a zero-order-hold time of Δt on the control has been and will be applied; if $\Delta t \neq t_s$ this assumption is inaccurate. We address these issues as follows. Since the formulation (19) is rate-based we optimize for the change in control Δv_i rather than v_i directly, this leads to a first order hold assumption applied to the control inputs which more closely matches what is actually applied. Further, we collect (19b) and (19c) into a single rate-based model

$$\bar{z}_{i+1} = \bar{f}(\bar{z}_i, \Delta v_i, \rho_i, \Delta t), \text{ where } \bar{z}_i = \left[\Delta z_i^T \;\; z_i^T \;\; v_{i-1}^T\right]^T, \tag{20}$$

and apply (19d) to $\bar{z}$ instead of z. The next issue which must be addressed is how to obtain (20) when $\Delta t \neq t_s$. We observed that a model of the form (8) cannot be identified from downsampled data directly due to aliasing effects; since we require that $\Delta t = dt_s \gg t_s$, it turns out that for any useful value of d that $1/\Delta t$ is well below the Nyquist frequency. Our solution is to

analytically propagate the LPV portion of (8) with $\Delta v_k = v_k - v_{k-1}$ held constant for d-steps and use the resulting downsampled model to assemble (20). An error model, $\hat{\theta}$, is then introduced to account for higher-order-terms. The resulting model has the following form,

$$\begin{bmatrix} \Delta z_{d(i+1)} \\ z_{d(i+1)} \\ v_{di} \end{bmatrix} = \begin{bmatrix} \tilde{A}(\rho_k) & 0 & 0 \\ \tilde{A}(\rho_k) & I & 0 \\ 0 & 0 & I \end{bmatrix}^d \begin{bmatrix} \Delta z_{di} \\ z_{di} \\ v_{d(i-1)} \end{bmatrix} + \begin{bmatrix} 0 \\ \tilde{F}(\rho_k) \\ 0 \end{bmatrix} + \left(\sum_{j=0}^{d-1} \begin{bmatrix} \tilde{A}(\rho_k) & 0 & 0 \\ \tilde{A}(\rho_k) & I & 0 \\ 0 & 0 & I \end{bmatrix}^j \right) \begin{bmatrix} \tilde{B}(\rho_k) \\ \tilde{B}(\rho_k) \\ I \end{bmatrix} \Delta v_{di} + \hat{\theta}(z_{di}, v_{di-1}, \Delta z_{di}, \Delta v_{di}, \rho_k), \tag{21}$$

where, $\tilde{A}$, $\tilde{B}$, and $\tilde{F}$ are the matrices in (8), and $\hat{\theta}$ is an error model which is identified to data generated by simulating (20) with Δv held constant for d-steps. Note that (8) is associated with $\Delta t = t_s = 8$ msec. Since the linear model can be downsampled analytically and the data used to train $\hat{\theta}$ is generated from noise-free simulations no aliasing effects occur.

Incorporating these changes, the OCP for the move-blocked rate-based MPC is

$$\min_{\Delta v} \; ||z_{dN_{fb}|k} - z_k^{trg}||^2_{Q_{fb}} + \sum_{i=0}^{N_{fb}-1} ||z_{di|k} - z_k^{trg}||^2_{Q_{fb}} + ||\Delta v_{di|k}||^2_{R_{fb}}, \tag{22a}$$

$$\text{s.t.} \quad (21), \quad i = 0, \; ..., \; N_{fb} - 1 \tag{22b}$$

$$z_{0|k} = \hat{z}_k, \quad \Delta z_{0|k} = \hat{z}_k - \hat{z}_{k-1}, \quad v_{-1|k} = \bar{v}_{k-1}, \tag{22c}$$

$$v_{lb} \leq v_{di|k} \leq v_{ub}, \quad i = 0, \; ..., \; N_{fb} - 1, \tag{22d}$$

where $\Delta v = \begin{bmatrix} \Delta v_{0|k}^T \; ... \; \Delta v_{d(N_{fb}-1)|k}^T \end{bmatrix}^T$ and $\bar{v}_{k-1} = v_{k-1} + v_k^{ff} - v_{k-1}^{ff}$. We observed that this method is effective and, for a fixed number of prediction horizon steps N_{fb}, significantly improves the performance of the feedback portion of the NMPC controller compared to setting $\Delta t = t_s$. The results in this paper were generated using $N_{fb} = 6$ and $d = 40$; the resulting time horizon length is approximately 1.9 s. The results presented in this paper utilize the higher order terms $\hat{\theta}$ in (21). However, we would advise in the future against using higher order terms in the feedback MPC. An argument can be made that the feedforward, nonlinear, MPC will keep the system close to the nominal setpoints where linear models are accurate. In addition, using a linear MPC for feedback is advantageous from a verification standpoint.

3.5 | Feedforward Design

The feedback portion of the NMPC controller, described above, incorporates integral action to compensate for model mismatch. It is well known that the addition of integral action can have a detrimental effect on the response speed of the closed-loop system; to compensate we add a feedforward module.

In engine airpath control, a common strategy is to obtain the feedforward control action from various maps and ad hoc logic. In this paper we instead use a model based approach and replace the traditional feedforward table with a function defined by the control action computed by a nonlinear model predictive controller placed in closed-loop with a nominal model of the plant. This is illustrated graphically in Figure 9. The OCP for the feedforward controller has the following form,

$$\min_{v^{ff}} \; ||\tilde{z}_{i|k} - z_k^{trg}||^2_{Q_{ff}} + \sum_{i=0}^{N_{ff}-1} ||\tilde{z}_{i|k} - z_k^{trg}||^2_{Q_{ff}} + ||v_{i|k}^{ff} - \tilde{v}_k||^2_{R_{ff}}, \tag{23a}$$

$$\text{s.t.} \quad \tilde{z}_{i+1|k} = f_n(\tilde{z}_{i|k}, v_{i|k}^{ff}, \rho_k, dt_s), \quad i = 0, \; ..., \; N_{ff} - 1, \tag{23b}$$

$$\tilde{z}_{0|k} = \tilde{z}_k, \tag{23c}$$

$$v_{lb} \leq v_{i|k}^{ff} \leq v_{ub}, \quad i = 0, \; ..., \; N_{ff} - 1, \tag{23d}$$

where N_{ff} is the prediction horizon length, Q_{ff} and R_{ff} are positive definite weighting matrices, $v^{ff} = \begin{bmatrix} v_{0|k}^{ff,T} \; ... \; v_{N_{ff}-1|k}^{ff,T} \end{bmatrix}^T$, f_n is the model of the engine defined in (8) with the dependence on the sampling time made explicit, $z_k^{trg} = [\chi_k^{trg} \;\; \bar{p}_{im}(\rho_k)]^T$ and $\tilde{v}_k$ satisfies $z_k^{trg} = f_n(z_k^{trg}, \tilde{v}_k, t_s)$. The state, $\tilde{z}_{k+1}$, in (23) is obtained by propagating the nominal model,

$$\tilde{z}_{k+1} = f_n(\tilde{z}_k, v_{0|k}^{ff}, \rho_k, \Delta t). \tag{24}$$

The results in this paper used a prediction horizon of $N_{ff} = 6$ and a discretization period of $\Delta t = 6t_s \approx 48$ msec for the feedforward module. No aliasing issues arose while identifying the models.

We found it significantly faster to tune the feedforward MPC instead of calibrating a feedforward map. In addition, the NMPC feedforward controller was able to provide feedforward commands which improved performance by overshooting steady state targets while still considering constraints. Essentially, the NMPC feedforward module is a dynamical system instead of a function like a typical map based feedforward.

Remark 6. The use of a fictitious state, $\tilde{z}$, is necessary to ensure that the NMPC feedforward module is truly a feedforward, i.e., it is independent of the measured state. If the measured state was used as an input to the feedforward module in Figure 9 would be in feedback, this is undesirable since it would interfere with the integral action in the feedback NMPC module.

3.6 | Stability and Feasibility

This section discusses the stability and feasibility of the SMPC and NMPC controllers. Both controllers use nonstandard techniques to achieve the performance required by the application and, as a result, a complete a-priori stability analysis is beyond the scope of this paper. However, existing theory and stability analyses of simplified models were used to guide the controller design process. We also performed exhaustive simulations and hundreds of hours of experimental testing to verify practical stability a-posteriori as is standard practice in engine control applications, see e.g., [48,49]. The testing was rigorous, performed systematically, and based on domain expertise; we didn't observe any issues during our testing campaign.

Moving forward, it would be preferable to incorporate the testing into a framework that allows for probabilistic performance/stability guarantees. A promising direction for future work along these lines is the application of data-driven statistical closed-loop verification techniques, see e.g., [50,51].

3.6.1 | SMPC

For the SMPC controller we will prove recursive feasibility and provide a stability proof for a nominal case where $\Delta\tau_i = t_s$, the horizon is sufficiently long, the OCP is solved exactly, and in the absence of model mismatch. Since SMPC uses a closed loop model of the NMPC controller and engine, this nominal analysis addresses loop interactions. In this section we will use $\mathbb{P}$ to denote the speed and fuel operating range of the engine. We will also denote the corresponding to ρ set of "safe", i.e., no misfires, surge, overpressure, etc., states by $\mathbb{X}(\rho)$. The set of admissible pairs is $\mathbb{Y}(\rho) = \{(x,u) \mid \phi(x,u,\rho) \leq \phi_l(x,\rho),\ 0 \leq \chi^{trg} \leq \bar{\chi}_{egr},\ 0 \leq q \leq q^{trg}\}$ and we define the set of admissible controls as $\mathbb{U}(x,\rho) = \{u \mid (x,u) \in \mathbb{Y}(\rho)\}$. Note that the constraint $x \in \mathbb{X}(\mathbb{P})$ is handled implicitly through careful selection of $\bar{\chi}_{egr}$ and $\bar{p}_{im}$, i.e., through good calibration.

Theorem 1. (Feasibility) For any $\rho \in \mathbb{P}$, $x \in \mathbb{X}(\rho)$ the set $\mathbb{U}(x,p)$ is nonempty, implying that (14) is always feasible.

Proof. Algebraic manipulation of (14c) yields,

$$q \leq q_{max}(\rho,x) = \phi_l(\rho,x)\frac{w_{cyl}(1-F_1)}{N_e c}\left(\frac{A}{F}\right)_s^{-1}, \tag{25}$$

an explicit bound on the fuel. For any $x \in \mathbb{X}$ the right hand side is positive thus $q_{max} \geq 0$ and the constraint $0 \leq q \leq \min(q_{max}, q^{trg})$ is feasible. The constraint $0 \leq \chi^{trg} \leq \bar{\chi}_{egr}(\rho)$ is feasible by construction which completes the proof. □

Theorem 2. (Nominal stability of SMPC) Let the following assumptions hold:

(A1) (Stability of the inner loop) For all steady state admissible $\bar{u} = [\chi^{trg}\ q]$ the inner-loop, with dynamics given by $x^+ = f(x,u,\rho)$, is asymptotically stable about the corresponding equilibrium point $\bar{x}(\bar{u})$ with region of attraction $\mathbb{X}$.

(A2) (Asymptotic controllability) The inner-loop system is asymptotically controllable with respect to (15) in the sense of[2] Assumption 6.5.

Then there exists a horizon length $N^* \in \mathbb{N}$ and a set $\mathbb{X}_s \subseteq \mathbb{X}$ such that if $N \geq N^*$ and $(x_0,u_0) \in \mathbb{X}_s \times \mathbb{R}^2$ then the sequence $\{(x_k,u_k)\} \to (\bar{x},\bar{u})$ as $k \to \infty$ with all $(x_k,u_k) \in \mathbb{Y}(\rho)$.

Proof. See Appendix. □

Note that Theorem 2 only proves the existence of a sufficiently large N^*; we have to assume that the horizon length chosen is long enough. We employ move blocking related techniques to increase N, see Section 3.2. In practice we observed no stability issues, lending some credence to our assumption.

Remark 7. It's difficult to verify (A2) a-priori since obtaining a high accuracy nonlinear model for the engine is extremely challenging. However, to support our assumption, we linearized (17) at each operating condition in the grid defined in (6) and observed that each linearized model was controllable. That is, we checked the controllability of $A_i = \nabla_x f_s(\bar{x}_i, \bar{u}_i)$ and $B_i = \nabla_u f_s(\bar{x}_i, \bar{u}_i)$ for all $i = 1, \dots 154$ where ρ_i is defined in (6), $\bar{u}_i = [\bar{\chi}_{egr}(\rho_i) \;\; q_i^{trg}]$ and $\bar{x}_i$ is the steady state associated with $\bar{u}_i$. This implies that (A2) holds in the vicinity of each operating point see e.g.,[52]. This does not imply that (A2) holds globally but it does support that it holds in our region of interest. Further, in Section 5 we present experimental results that demonstrate the robustness of the SMPC controller a-posteriori.

3.6.2 | NMPC

To achieve the performance required by the application the feedback portion of the NMPC controller uses a variety of non-standard techniques e.g., rate-based formulations, sampling period/prediction horizon discretization mismatch, and move blocking. As a result, a rigorous stability analysis of the controller is challenging and beyond the scope of this work. However, existing theoretical stability results were used to guide the design of the NMPC controller. In particular, we use move-blocking to achieve a sufficiently long prediction horizon.

The NMPC feedback controller does not include a terminal set constraint or terminal penalty, two common methods for establishing the stability of nonlinear model predictive controllers; instead we use the sufficiently long horizon approach. Enforcement of the terminal set constraint consumes computational resources and can cause feasibility issues. Further, since we are unaware of a suitable control Lyapunov function for this system, terminal penalties must be constructed numerically by solving appropriate Ricatti equations at each timestep, adding computational burden. We have also observed that, for this application, terminal penalties can reduce the performance of the controller.

We refer interested readers to[2] Chapter 6 and the references therein for an comprehensive treatment on the stability of NMPC without terminal constraints or penalties. It has been shown (see[2] Theorem 6.24) that provided that the plant is (i) asymptotically controllable, (ii) the stage cost function can be bounded above and below by appropriate comparison functions, and (iii) the horizon is sufficiently long then the nominal closed-loop system is stable. The feedback portion of the NMPC controller uses a strongly convex quadratic stage cost, and exploits move blocking to ensure the prediction horizon (1.52 s in 6 steps) is sufficiently long. We verified that (8) is locally controllable at each operating point in our grid in the same manner as described in Remark 7, i.e., by checking controllability of the identified models at each operating point in the grid. Closed-loop stability is then confirmed in practice through exhaustive simulation studies and experimental testing. Since the NMPC controller only enforces control constraints, feasibility is automatic.

The engine is controllable only in a portion of its state space. Outside of this safe region, the engine may display undesirable behaviour such as surge, misfire, or excessive roughness characterized by torque fluctuations due to poor combustion. The target operating points must be chosen to avoid these unsafe regions. In practice, this is accomplished through the domain expertise** of the engine manufacturer and experienced calibrators who provide a set of nominal operating points for the engine. We take these operating point maps as a given (indeed they are included in the engine control unit software), their safety can be verified by performing fuel step experiments on a dynomometer.

4 | CONTROLLER IMPLEMENTATION

Implementation of a model predictive controller requires an algorithm for (approximately) solving the optimal control problems online. This section outlines suitable strategies for the SMPC controller and for the feedback and feedforward portions of the NMPC controller.

**Engine calibration is a complex topic with an extensive literature, see e.g.,[39,53,54,55], engine manufacturers also have sophisticated internal methodologies. As such, engine calibration is outside the scope of this paper.

4.1 | Supervisory Controller

The SMPC controller OCP, (14), can be compactly represented as

$$\underset{z_k}{min.} \quad J(z_k, \hat{x}_k, \rho_k), \tag{26a}$$

$$s.t \quad c(z_k, \hat{x}_k, \rho_k) \leq 0, \tag{26b}$$

where $z_k = \left[u_k^T \; s_k^T\right]^T \in \mathbb{R}^n$ are the collected primal decision variables, $\hat{x}_k \in \mathbb{R}^5$ is the SMPC state estimate, $\rho_k \in \mathbb{R}^2$ is the operating condition, $J : \mathbb{R}^n \times \mathbb{R}^5 \times \mathbb{R}^2 \to \mathbb{R}$ is the cost function, and $c : \mathbb{R}^n \times \mathbb{R}^5 \times \mathbb{R}^2 \to \mathbb{R}^m$ defines the inequality constraints. All equality constraints are linearized about $\hat{x}_k$ and $u = (0, q_k^{trg})$ is eliminated by substitution using symbolic tools[22]. This nonlinear program can then be approximated by the following quadratic program,

$$\underset{\Delta z_k}{\text{min.}} \quad \frac{1}{2}\Delta z_k^T H_k \Delta z_k + f_k^T \Delta z_k, \tag{27a}$$

$$\text{s.t} \quad A_k \Delta z_k \leq b_k, \tag{27b}$$

where $H_k = \nabla_z^2 J(z_{k-1}, \hat{x}_k, \rho_k) \succeq 0$, $f_k = \nabla_z J(z_{k-1}, \hat{x}_k, \rho_k)$, $A_k = \nabla_z c(z_{k-1}, \hat{x}_k, \rho_k)$, and $b = -c(z_{k-1}, \hat{x}_k, \rho_k)$. We solve one instance of (27) per timestep as in the real-time iteration scheme[56]. This is a type of time-distributed sequential quadratic programming and, under appropriate conditions, convergence to a solution of the nonlinear OCP can be guaranteed over time[57,58]. The SMPC control update is then computed as

$$u_k = \left[z_k(1:2)^T\right]^T, \quad z_k = z_{k-1} + \Delta z_k, \tag{28}$$

where we have used MATLAB notation for indexing.

To guarantee that each QP is convex the nonlinear dynamics are linearized about $\hat{x}_k$ and $u = [0 \; q_k^{trg}]^T$ at each sampling instance before substitution into the cost and constraint functions. We also use the Hessian of the cost function rather than of the Lagrangian which is sometimes referred to as Gauss-Newton Hessian approximation[56]. In addition, due to the presence of slack variables in the original OCP, each QP is guaranteed to be feasible and, due to the penalty on $u_i - u_{i-1}$ in the cost function, (15), the strong second order sufficient conditions always hold. Together these properties ensure that (27) is always solvable and admits a unique solution. The quadratic programs are solved using the Fischer-Burmeister Regularized and Smoothed (FBRS) method which is described in detail in[21].

Remark 8. To guarantee finite-time execution, necessary for real-time implementation, the FBRS algorithm is limited to 5 Newton iterations per sampling period. Each QP is warmstarted with the solution of the QP at the previous timestep. This was sufficient for convergence in almost every case, we observed no degradation of control performance due to the iteration limit. MPC is known to be robust to suboptimality, see[59,60].

4.2 | Airpath Controller

Both the feedforward, (23), and feedback, (22), OCPs can be compactly written in the following form:

$$\underset{x_k}{min.} \quad J(x_k, \hat{z}_k, \rho_k), \tag{29a}$$

$$s.t \quad h(x_k, \hat{z}_k, \rho_k) \leq 0, \tag{29b}$$

where, $x_k \in \mathbb{R}^n$ is the collection of primal decision variables, $\hat{z}_k \in \mathbb{R}^2$ is the state estimate, $\rho_k \in \mathbb{R}^2$ is the operating condition, $J : \mathbb{R}^n \times \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}$, is the cost function and $h : \mathbb{R}^n \times \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}^m$ is a vector of constraints. For both the feedforward and feedback NMPC controllers the inequality constraints are box constraints. For the feedforward,

$$x_k^{ff} = \left[v_{0|k}^{ff,T} \; ... \; v_{N_{ff}-1|k}^{ff,T}\right]^T, \tag{30}$$

and for the feedback

$$x_k^{fb} = \left[\Delta v_{0|k}^T \; ... \; \Delta v_{N_{fb}-1|k}^T\right]^T. \tag{31}$$

All equality constraints are eliminated via symbolic substitution. Since only actuator saturation constraints are considered we chose to implement them via exterior penalties. Consider the exterior quadratic penalty,

$$\gamma : \mathbb{R} \to \mathbb{R}, \quad \gamma(\cdot) = max(0, (\cdot))^2, \tag{32}$$

applying it to the constraints yields an unconstrained OCP:

$$\underset{x_k}{min.} \quad F = J(x_k, \hat{z}_k, \rho_k) + \sum_{j=i}^{m} \sigma_j \gamma(h_j(x_k, \hat{z}_k, \rho_k)), \tag{33}$$

where $\sigma_j > 0$ are the penalty weights and m is the number of inequality constraints. This OCP can be linearized about the solution at the previous timestep, x_{k-1}, to form a quadratic approximation

$$\underset{\Delta x_k}{\text{min.}} \quad \frac{1}{2}\Delta x_k^T G_k \Delta x_k + d_k^T \Delta x_k, \tag{34}$$

where $G_k = \nabla_x^2 F(x_{k-1}, \hat{z}_k, \rho_k)$, and $d_k = \nabla_x F(x_{k-1}, \hat{z}_k, \rho_k)$. The solution of this QP can be computed by solving a linear system of equations, the resulting solution is then computed as $x_k = x_{k-1} + \Delta x_k$. As with the SMPC problem this procedure is a variant of the real-time iteration scheme[56]. Despite only solving one QP per timestep, under appropriate conditions, the real-time iteration scheme converges to the solution of the nonlinear OCP[57,58]. By using exterior penalties we are only able to solve (23) and (22) in an approximate sense. In exchange we solve a single linear system per timestep for each NMPC problem which significantly reduces the computational cost of the controller. We have found that the exterior-penalty based approximations perform well in practice provided the penalty weights are properly tuned.

4.3 | Executable Generation and Execution Time

The MPC controller was implemented on a dSPACE DS1006 rapid prototyping unit (2.6 GHz CPU clock speed) using Simulink 2010b SP2 real-time workshop (RTW). All necessary derivatives for the three optimal control problems, (14), (22), and (23), were calculated using symbolic methods. The tools, collectively referred to as the symbolic control design environment (SCDE)[22], translate symbolic expressions into highly optimized C code (and corresponding S-function templates) which meets the Motor Industry Software Reliability Association (MISRA) standard for embedded computations[61]. Expressions for the cost, constraints, and dynamics were written in the Maple symbolic language then processed and differentiated symbolically. The SCDE was then used to generate S-functions for all necessary derivatives. The QP solver for the SMPC controller was implemented in embedded MATLAB. The QP solvers for the feedforward and feedback NMPC controllers, which are Cholesky factorization routines, were directly implemented in the SCDE.

Execution times on the DS1006 rapid prototyping unit are shown in Table 1. The entire MPC controller takes approximately 1 ms to execute in the worst case, well below the sampling period of 8 ms. The total size of the real-time executable file was 5547 KB. Note that since the number of Newton iterations performed by the SMPC QP solver was fixed at 5 the convergence check was disabled; as a result, the average and maximum execution times are similar.

4.4 | Controller Calibration

Figure 10 provides an overview of the MPC controller calibration process. First the open-loop models are identified as described in Section 2.1, then the NMPC controller is tuned, closed-loop models are identified (Section 2.2) and the SMPC controller is tuned. The final step is drivecycle testing to verify the tuning and identify if any re-tuning is needed.

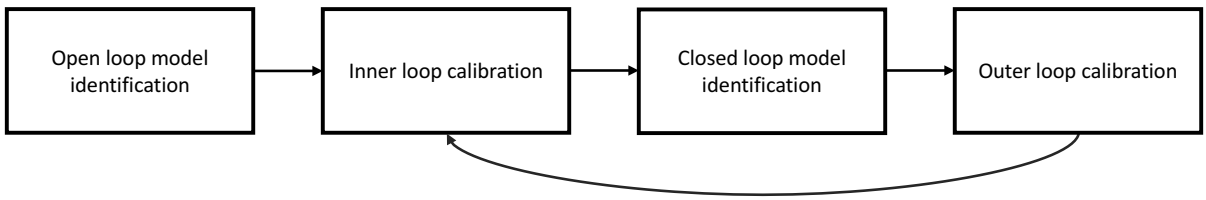

FIGURE 10 Calibration and model identification process for the MPC controllers.

4.4.1 | Feedforward NMPC Controller

The feedforward NMPC controller is tuned in closed-loop simulations with its prediction model using nominal target maps. The fueling rate range is divided evenly into low, medium, and high zones for tuning. At each engine speed the controller is tuned by performing fuel step maneuvers so as to maximize the responsiveness of the EGR rate while avoiding oscillations in either the EGR rate or intake pressure. The result are positive definite weighing matrices $Q_{ff,i}$, $R_{ff,i}$ corresponding to each grid point ρ_i, $i = 1, \dots, 154$. Linear interpolation is used between gridpoints to construct functions $Q_{ff}(\rho)$, $R_{ff}(\rho)$, these are evaluated at the measured operating point then used to construct (23). Only the diagonal terms of Q_{ff} and R_{ff} are nonzero.

4.4.2 | Feedback NMPC Controller

Once the feedforward controller is tuned, the feedback NMPC controller is tuned by performing fuel-step experiments at fixed engine speeds on the engine bench using nominal EGR rate and intake pressure target maps. The fueling rate operating range is evenly divided into low, medium, and high zones for tuning. At each engine speed, fuel step experiments are performed and the EGR rate settling time is reduced as much as possible without inducing oscillations in the intake pressure or EGR rate responses. Cross terms in the weighting matrices are not used.The result are positive definite weighing matrices $Q_{fb,i}$, $R_{fb,i}$ corresponding to each grid point ρ_i, $i = 1, \dots, 154$. Linear interpolation is used between gridpoints to construct functions $Q_{fb}(\rho)$, $R_{fb}(\rho)$, these are evaluated at the measured operating point then used to construct (22). Finally, these tunings are verified through drivecycle tests to ensure they perform well under engine speed variations.

4.4.3 | SMPC Controller

The SMPC controller is tuned after the inner loop controller is calibrated. SMPC has 4 tuning parameters, $\alpha_s, \beta_s, \gamma_s > 0$ and $R_s \succ 0$, which is a 2 by 2 matrix, and is straightforward to tune. Tunning is performed during fuel step experiments at fixed engine speeds. The fuel deviation parameter α_s is fixed to a sufficiently large value to ensure fuel tracking, we used $\alpha_s = 1$. The constraint softening parameter was set to $\beta_s = 1000$, γ_s was set to 0.05 and $R_s = diag(5, 0.1)$ was used.

5 | EXPERIMENTAL RESULTS

The integrated MPC controller was placed in closed-loop with a 2.8L diesel engine on a transient engine dynomometer[††] for experimental validation. The engine is instrumented to measure p_{im}, w_c, and N_e which are used for feedback, estimators are used to reconstruct the rest of the state, see Section 3.1. The signals τ_q, ψ_{NO_x}, ψ_{THC}, and ψ_{OP} are only used for performance evaluation. Other signals are measured but used only for model validation and calibration as described in Section 1.2. In what follows, we compare the NMPC controller with a PID benchmark, analyze step response traces, and present drive cycle results comparing the integrated MPC controller with state-of-the-art industrial equivalent.

5.1 | Step Responses

Before exercising the controller on a more realistic driving pattern, we present some step responses to obtain some insight into typical closed-loop behaviours of the system. First, we compare the NMPC controller with a state-of-the art industrial benchmark (with the supervisor inactive) to showcase the utility of the NMPC controller. Figure 11 compares the responses of the two controllers to a fuel step up, the NMPC controller more closely tracks the EGR rate target leading to reduced NO_x emissions. Figure 12 compares the responses of the two controllers during a fuel-step down, again the NMPC controller more closely tracks the EGR rate target leading to reduced NO_x production.

Second, we illustrate the impact of the NMPC feedforward module. Figure 13 compares the response of the engine, in closed-loop with the NMPC airpath controller with and without the feedforward module, to an acceleration command. Without the feedforward module, the controller's EGR rate tracking performance suffers, it relies on the slow integral action in the feedback module to open the EGR valve in order to increase the EGR rate. This leads to an avoidable increase in NO_x emissions. Figure 14

[††]The focus of this paper is on the engine control algorithms, so for brevity we only provide details about the experimental setup insofar as they relate to the controllers and refer interested readers to the literature [40,62,63] for more details regarding engine testing and instrumentation.

compares the NMPC feedforward module against a standard static map based approach. Note that the EGR valve and EGR throttle actuation is much more aggressive leading to better EGR rate and intake pressure tracking performance.

Finally, we consider the integrated controller. Typical responses with the supervisory controller active are shown in Figure 15. During a tip-in (fuel step up) the fuel-air ratio rapidly increases since fuel is added to the system more quickly than the airflow can be raised to compensate. This causes the controller to predict a fuel-air ratio constraint violation. In response the fueling rate is filtered and the EGR rate target undershoots its steady state value to reduce the fuel-air ratio. Moreover, by adjusting the fuel-air ratio limit, it is possible to limit smoke production.

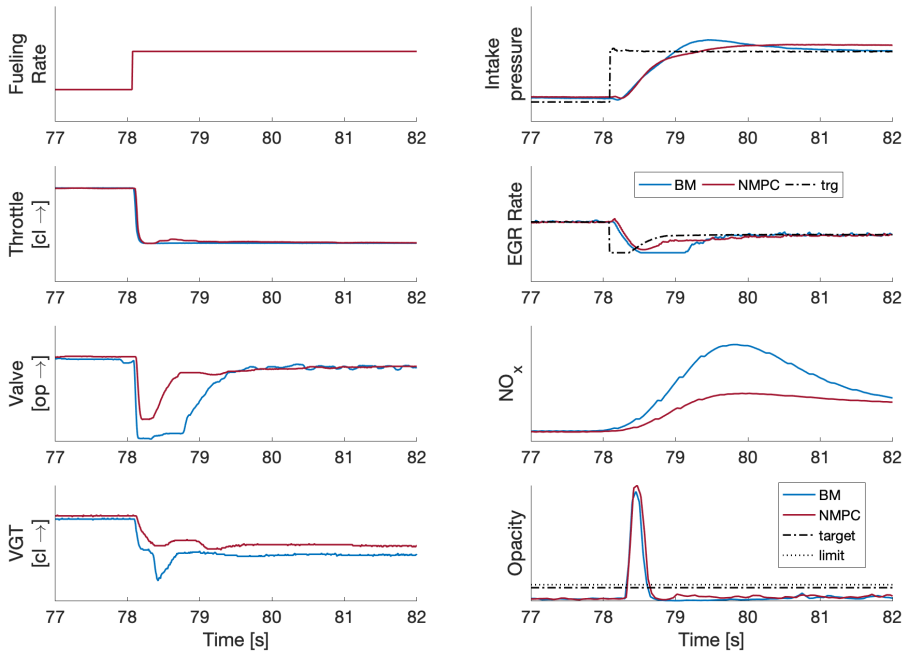

FIGURE 11 A comparison between the benchmark (BM) and NMPC controllers during a fuel step-up maneuver at a constant engine speed. The NMPC controller is able to more closely track the EGR rate target leading to reduced NO_x emissions. The supervisory controller is disabled so opacity limits are not enforced. The y-axes scales have been removed to preserve data confidentiality.

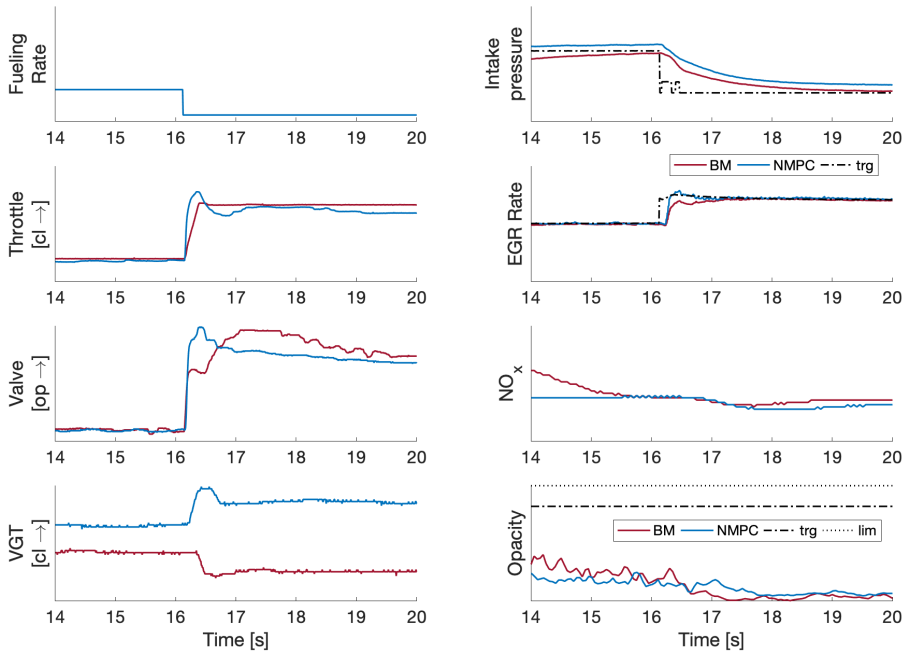

FIGURE 12 A comparison between the benchmark (BM) and NMPC controllers during a fuel step-down maneuver at a constant engine speed. The NMPC controller more closely tracks EGR rate leading to reduced NO_x production. The y-axes scales have been removed to preserve data confidentiality.

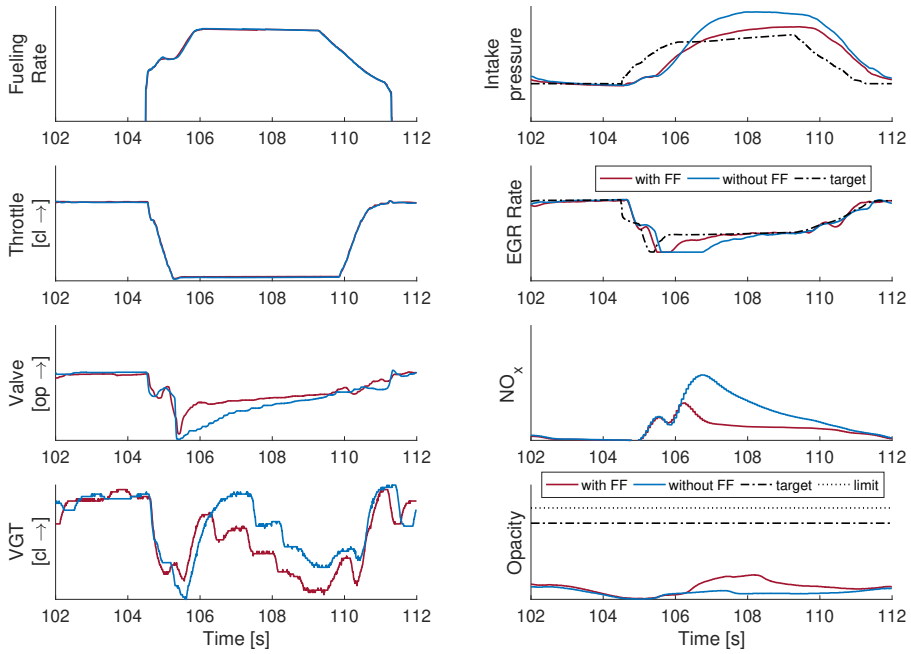

FIGURE 13 A comparison between the NMPC airpath controller with and without the NMPC feedforward module at a constant engine speed. The feedforward module improves EGR rate tracking leading to reduced NO_x emissions. The y-axes scales have been removed to preserve data confidentiality.

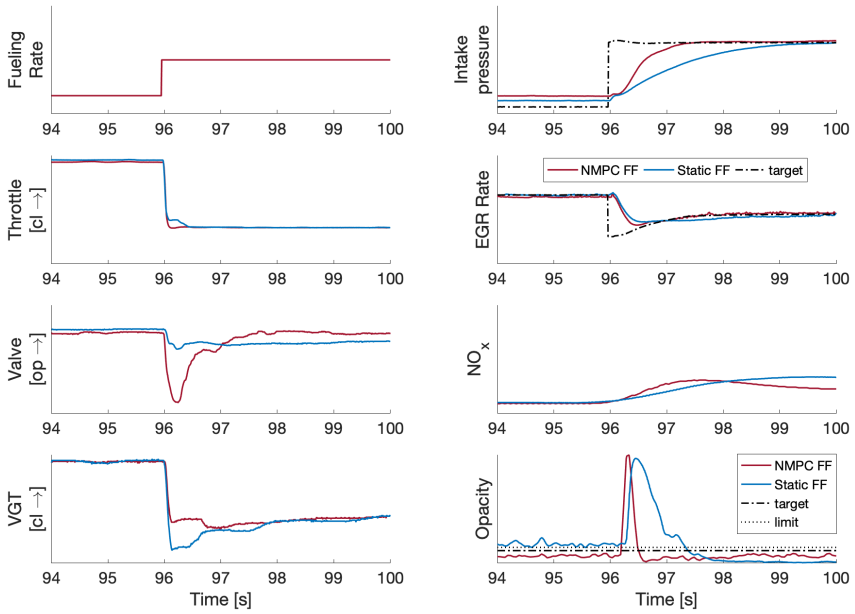

FIGURE 14 A comparison between the NMPC airpath controller with the NMPC feedforward module and a static map based feedforward module. The NMPC FF module leads to improved EGR rate and intake pressure tracking performance. The supervisory controller is disabled so opacity limits are not enforced. The y-axes scales have been removed to preserve data confidentiality.

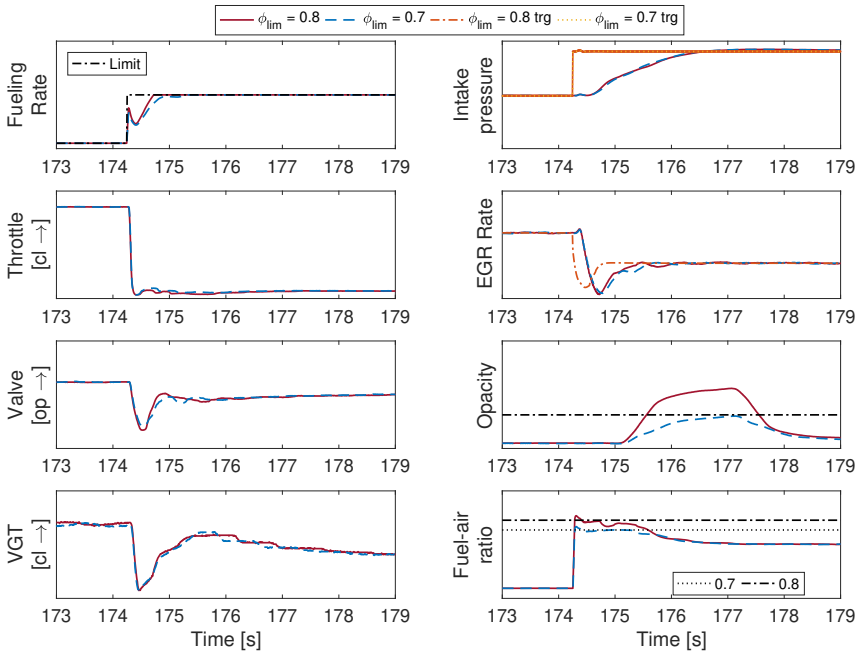

FIGURE 15 Experimental results of two fuel steps at 2400 rpm. Exhaust opacity (smoke) constraints can be enforced by limiting the fuel-air ratio. For example, when $\phi_{lim} = 0.8$ the opacity constraint is violated and reducing ϕ_{lim} to 0.7 eliminates the problem. The fuel-air ratio constraint is slightly violated in both cases due to an unmodelled 3 step delay between the MPC controller and the actuators. The y-axes scales have been removed to preserve data confidentiality.

5.2 | Drive Cycle Validation

This section presents experimental validation results for the integrated MPC controller which it was exercised over the Worldwide Harmonized Light Vehicles Test Cycle (WLTC) and New European Drive Cycle (NEDC) using a simulated vehicle and driver.

Emissions performance was evaluated by estimating the cumulative mass of NO_x and THC emitted during the drivecycle, smoke was evaluated by integrating the exhaust opacity whenever it was above the visible limit[‡‡]. Drivability was evaluated using the root mean square velocity tracking error (RMSE) between the longitudinal speed of the simulated vehicle and its target. Relative difference is defined as

$$\%\ \text{difference} = \frac{\text{DAP MPC} - \text{Benchmark}}{\text{Benchmark}} * 100. \tag{35}$$

5.2.1 | Overall Performance Results

A summary of the results, using the best tunings obtained during testing, is contained in Table 2. Figure 16 represents typical experimental traces when DAP MPC controller is in closed-loop with the engine. The smoke is well controlled, only a handful of spikes approach the boundary of the visible range.

Over the WLTC the the DAP MPC controller was able to significantly reduce cumulative NO_x and THC, compared to a state of the art benchmark controller. The aggressive tuning resulted in slightly worse drivability compared to the benchmark and yielded significant NO_x and THC reductions at the cost of a small increase in smoke production. A more conservative tuning reduced smoke emissions compared to the benchmark but led to smaller NO_x and THC improvements; drivability was adversely affected as well. Over the NEDC the DAP MPC controller slightly increased NO_x and slightly decreased THC. The NEDC cycle is not aggressive enough to trigger fuel limiting or cause visible smoke. The increase in fuel consumption is not large enough to be considered significant as the results are estimated by integrating the commanded fuel signal rather than measured using a fuel meter. The explicit incorporation of fuel-economy into the MPC controller is a topic of future work.

Remark 9. The opacity target is the maximum opacity value allowed during steady state calibration. The opacity limit is when smoke becomes visible, it is desirable to minimize violation of this constraint. The smoke metric used in Table 2 is computed by integrating the opacity signal whenever its higher than the limit.

5.2.2 | Detailed Traces

This section presents detailed experimental traces (Figures 17 to 21) to provide insight into the behaviour of the controller. Figure 17 shows a high speed portion of the WLTC. Between 165 and 170 s, the SMPC controller predicts fuel-air ratio constraint violation and, in response, lowers the EGR rate target and limits the fuel to enforce the constraint. As a result, the exhaust opacity remains below the visible limit. Figure 18 illustrates the importance of the supervisory controller. Without the it, large amounts of visible smoke are produced. When active, the SMPC controller successfully manipulates the fueling rate and EGR rate target to limit smoke production.

Figure 19 shows the response of the DAP MPC controller during an acceleration event. As the vehicle speed increases, three gearshifts occur, each consisting of a tip out (fuel step down) followed by a tip in (fuel step up). In response to each tip in the SMPC controller predicts a fuel-air ratio constraint violation and reduces the EGR rate target, in order to empty the intake manifold of burnt gas, and limits the fuel to enforce the constraint and prevent visible smoke. The target then returns to the steady state EGR rate target as quickly as possible to reduce NO_x . Due to the use of feedforward and feedback in the inner-loop NMPC controller, it is able to use the VGT and EGR valve to track the EGR rate target very accurately. This fast EGR rate tracking ensures that the fuel-air ratio constraint satisfaction is accomplished primarily with EGR valve and VGT actuation, rather than with fuel limiting, which is important for drivability. Figure 20 shows an input-output (most benchmark signals cannot be shown for confidentiality reasons) comparison between the DAP MPC controller and a benchmark industrial controller. The DAP MPC controller reduces transient NO_x by shrinking the spikes that occur after gearshifts. This is possible because (i) the SMPC controller brings the EGR rate target back to its steady state target quickly after a tip in by accurately calculating the fuel-air ratio response using its prediction model and (ii) because the NMPC controller is able to track very fast EGR rate transients by exploiting both the VGT and EGR valve.

[‡‡]The visibility threshold used here is consistent with current calibration guidelines.

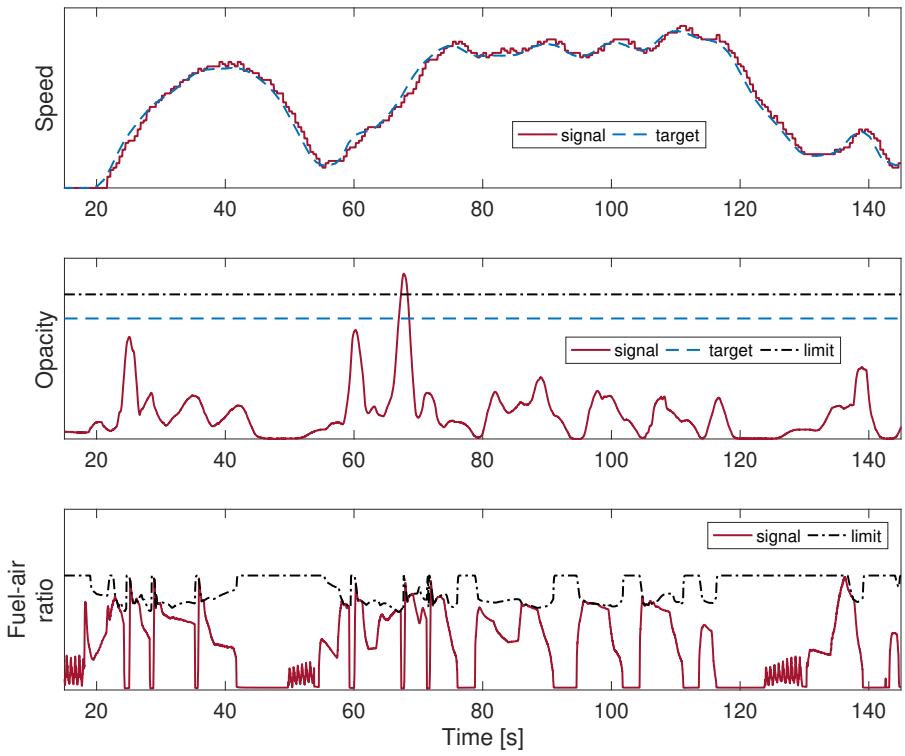

FIGURE 16 A portion of the WLTC with the MPC controller in closed loop with the engine. The vehicle was able to successfully complete the drivecycle despite some fuel limiting. The smoke is well controlled with only a few spikes reaching the edge of the visible range. The y-axes scales have been removed to preserve data confidentiality.

Figure 21 illustrates some issues we observed with the DAP MPC controller. Firstly, between 160 and 165 s the fuel is limited quite severely, despite the exhaust opacity being nowhere near the visible limit. This indicates there is room for improvement in the fuel-air ratio limit map (recall that $\phi_l = \phi_l(N_e, w_c)$). Secondly, between 170 and 180 s the closed-loop system becomes oscillatory which is undesirable in view of increased wear on the actuators and oscillations in the engine torque. This occurs because the controller oscillates near the fuel-air ratio constraint due to mismatch between the EGR rate tracking error predicted by the SMPC controller and the true tracking error. Possible countermeasures for this issue are discussed in Section 6.2.

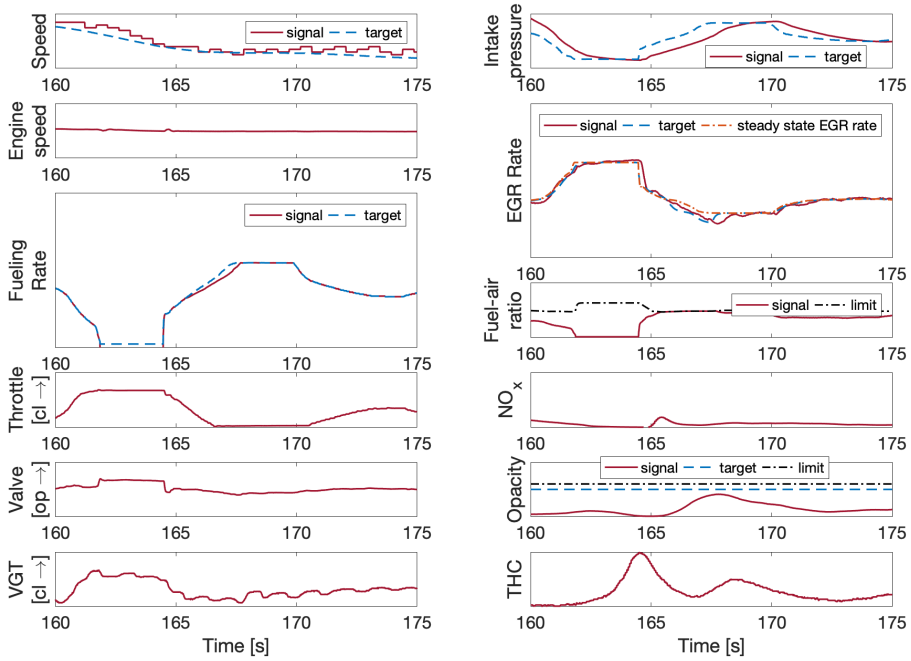

FIGURE 17 A close-up of a high speed potion of the WLTC with the MPC controller in closed-loop with the engine. The supervisory layer prevents visible smoke by coordinating the fuel input and the EGR rate. It causes the EGR rate target to undershoot the steady state optimum, this command is tracked by the NMPC controller, and limits the fueling rate to satisfy the fuel-air ratio constraint and prevent visible smoke. The y-axes scales have been removed to preserve data confidentiality.

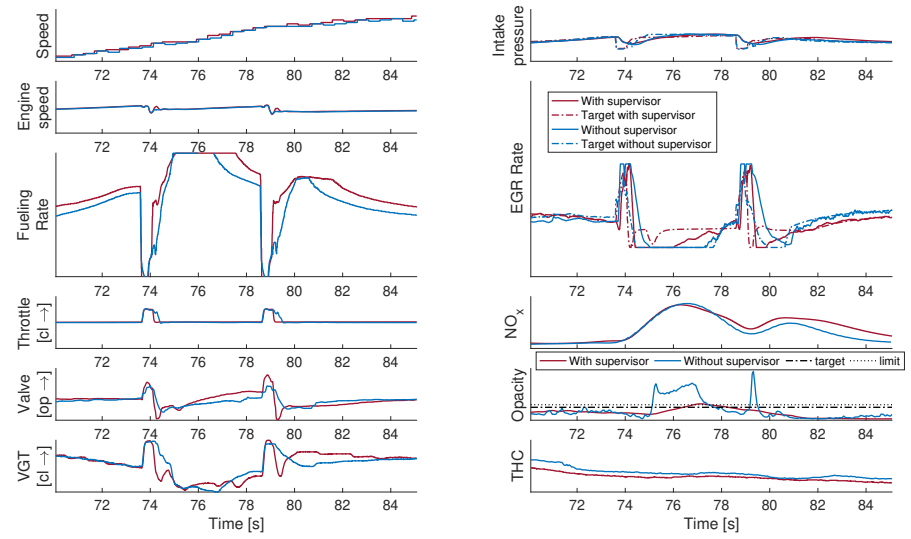

FIGURE 18 A comparison between the MPC controller with and without the supervisory controller. Without the supervisor large amounts of visible smoke is produced; the opacity limit is where the smoke becomes visible. The enforcement of the smoke constraint leads to a torque delay which is undesirable but necessary to prevent excessive smoke. Minimization of this torque delay is part of the SMPC formulation via the fuel tracking term. The y-axes scales have been removed to preserve data confidentiality.

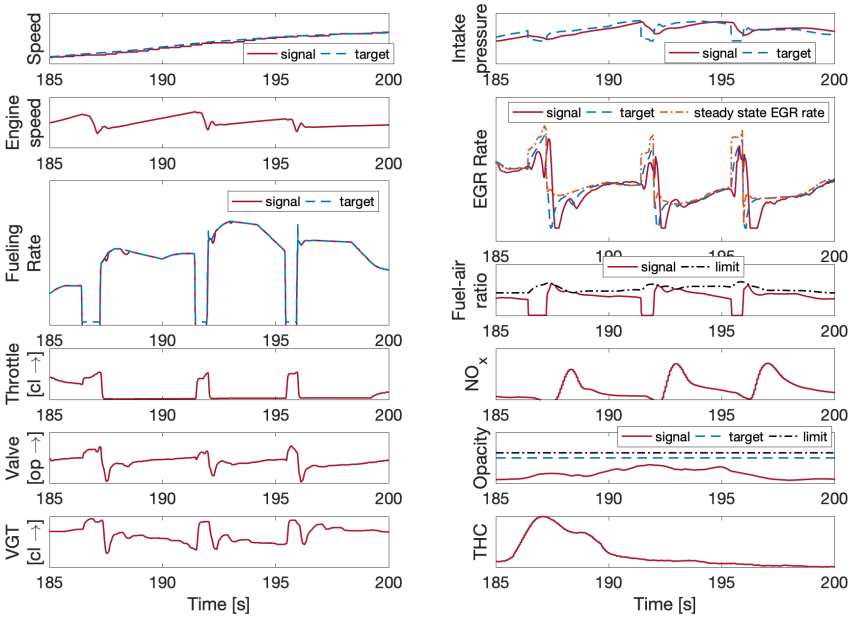

FIGURE 19 A close-up of an acceleration event with the MPC controller in closed-loop with the engine during the WLTC. During the tip ins following the gearshifts the supervisory layer commands the NMPC controller to quickly reduce the EGR rate and limits the fuel to satisfy the fuel-air ratio constraint, preventing visible smoke. The inner loop relies on its feedforward module to track the aggressive commands from the supervisory layer, leading to a very fast EGR rate response which minimizes fuel limiting and NOx production. The y-axes scales have been removed to preserve data confidentiality.

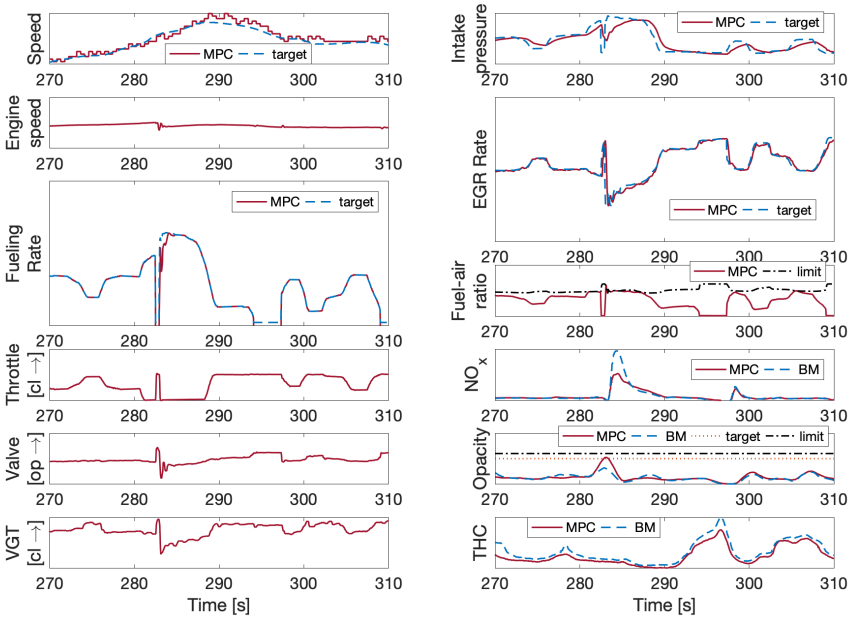

FIGURE 20 A comparison between the MPC and a benchmark (BM) strategy running in closed-loop with the engine during the WLTC. The MPC controller is able to reduce NO_x emissions compared to the benchmark strategy without causing visible smoke or increasing hydrocarbon output. All signals are generated by the MPC controller unless otherwise noted. The y-axes scales have been removed to preserve data confidentiality.

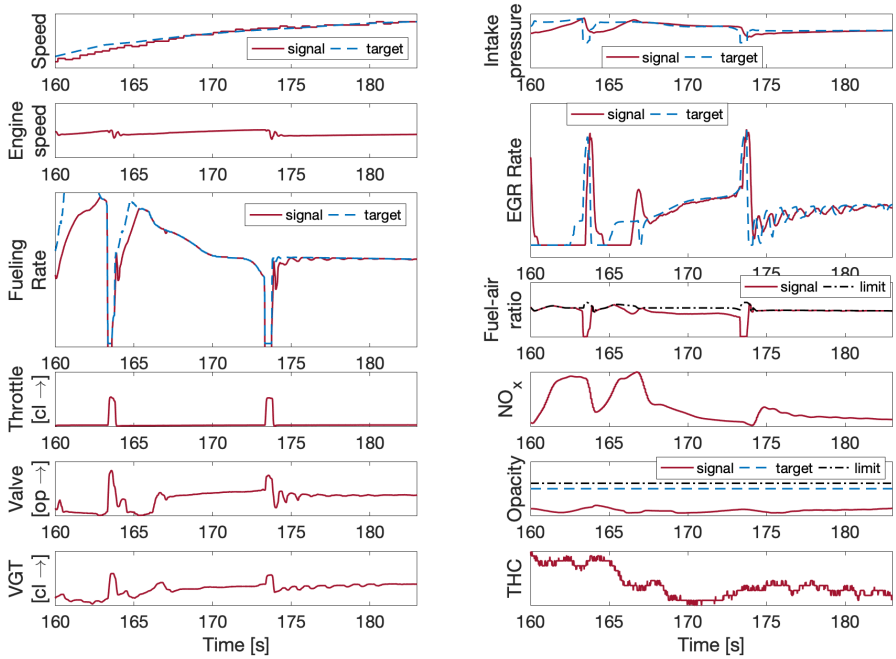

FIGURE 21 A close-up of an acceleration event with the MPC controller in closed-loop with the engine during the WLTC. The MPC controller aggressively limits the fuel despite the low opacity of the exhaust. Oscillation against the fuel-air ratio constraint boundary also occurs which leads to undesirable EGR rate and torque fluctuations. We discuss countermeasures for these oscillations in Section 6.2. The y-axes scales have been removed to preserve data confidentiality.

6 | DISCUSSION

While our proposed MPC controller is promising, more development effort is necessary before it can be considered "production ready". Here we discuss the challenges we observed leading up to and during our experimental campaign and outline directions for continued controller development.

6.1 | Computational Footprint

One of the largest drawbacks of MPC is the computational footprint. Our controller uses the real-time iteration scheme[56], a kind of time-distributed sequential quadratic programming[58], a new QP solver[21], and symbolic code generation tools[22] to reduce the worst case total execution time to around $715\mu s$ on a 2.6 GHz rapid prototyping unit. Assuming a 256 MHz ECU and estimating the computation time using a clock speed scaling analysis [§§] we arrive at an estimate of $0.715ms \cdot \frac{2.6\ GHz}{256\ MHz} = 7.26\ ms$ which is slightly below the current 8 *ms* sampling period. This indicates that our strategy is likely to be implementable in real-time on an appropriate dedicated ECU. However, ECUs must run other modules besides the engine controller so further reductions in the CPU usage are necessary before the controller is production ready.

Since completing the experiments, we have performed preliminary investigations into the use of Kylov methods, specifically the conjugate gradient method[64], for inexactly solving the necessary conditions of (33). We have found that $2-3$ times improvements are possible without noticeably degrading controller performance. We also believe this approach is promising when combined with FBRS[65] to solve (27). Another possible avenue for improvement is the use of optimized mathematical subroutines. Our current implementation uses symbolically generated or handwritten linear algebra and factorization routines. It is well known that specialized routines (e.g., those provided by high performance BLAS or LAPACK libraries) can offer much better performance; they are often optimized at an assembly code level. Further, embedded BLAS libraries have recently begun to appear[66]. We believe that routines of this type for ECUs could help further reduce CPU usage. Finally, we expect ECU hardware to continue to improve over time. In particular, the incorporation of dedicated signal processing units into future ECUs could further reduce computation times.

6.2 | Performance Issues

We observed some performance issues during experimental testing. Specifically, the MPC controller was not able to perfectly control smoke see e.g., Figure 16, and we observed some undesirable oscillations in the fueling rate input, see e.g., Figure 21.

Typically, when using MPC, the best way to improve constraint handling is to improve the prediction models. The least accurate portion of our prediction models is the intake manifold burnt gas fraction F_1 in (10). Since our experimental setup did not have an oxygen sensor in the intake manifold, we were unable to directly validate our F_1 predictions and/or improve the model using data. As our smoke control approach depends directly on F_1 inaccurate predictions of F_1 are the most likely cause of the performance degradation we observed. We suspect that installing the sensor on the testbed (for modelling and validation purposes only, i.e., not for use as feedback) and an associated data collection and modelling effort could significantly improve smoke control. Moreover, as this modelling error forced us to adopt a more conservative smoke limit, we hypothesize that this caused the underperformance we observed on the less aggressive NEDC cycle where the conservatism was not warranted. Note that the benchmark controller's smoke control is also not perfect.

Beyond improving the quality of the models, various modifications to the SMPC controller formulation could be considered. For example, re-parameterizing the fueling rate input in (14) as

$$q = \kappa(q^{trg} - q^{-}) + q^{-} \tag{36}$$

where $\kappa \in [0,1]$ is the new decision variable, q^{-} is the input applied at the previous instance and q^{trg} is the target, will force the fueling rate input to approach the target monotonically and thus remove oscillations. Such a parameterization is often used in reference governors[29].

[§§] We have found clock scaling analyses to be sufficiently accurate for first order computation time estimates due to the relative simplicity of typical ECU computational architectures.

6.3 | Future Perspectives

Our proposed MPC controller offers several advantages over a typical industrial PID based engine control strategy:

- Statement of the control objective is intuitive and can be easily summarized through the OCP formulation;
- The use of prediction models reduces the amount of conservatism needed to handle constraints, leading to performance improvements;
- It straightforwardly handles multiple input and outputs, coupling between them, and nonlinearities;
- It can be calibrated quickly.

It also suffers from some drawbacks:

- It requires an understanding of more advanced control engineering concepts to troubleshoot issues;
- It has a higher computational footprint;
- Construction of the prediction models can be labour intensive.

Overall, we demonstrated that significant ($10-15\%$) emissions reductions are possible without hardware changes[¶¶], e.g., without adding additional sensors or actuators, and that the challenges associated with the computational burden of MPC are not insurmountable, even when using nonlinear MPC. We also demonstrated various techniques such as non-uniform prediction horizons and inner-outer loop MPC control architectures that may be useful in a broader context.

We believe that, as emissions regulations become tighter, MPC has the potential to play a significant role in emissions reduction efforts. However, progressing from demonstrations to mass production would require significant amounts of effort both in terms of modelling, controller improvement and computational footprint reduction, as outlined in the previous two subsections. It would also require creating standardized tools for design, calibration, simulation, modelling and deployment, a significant experimental validation campaign, and a significantly larger team of engineers and researchers. See e.g., the acknowledgments section in[67] which details the deployment of linear MPC in production by a major automotive manufacturer[67].

7 | CONCLUSION

An emissions oriented model predictive controller for a diesel engine airpath has been designed and experimentally validated. Using a combination of a supervisory model predictive controller, for emissions constraint enforcement, and a nonlinear model predictive controller incorporating optimal feedforward and feedback, for nonlinearity compensation, we were able to demonstrate significant reductions in cumulative emissions over the WLTC drive cycle.

By distributing quadratic programming iterations over time we are able to achieve low computation times without compromising performance. However, while the DAP MPC controller is executable real-time using a rapid prototyping unit, more work is required to make this strategy feasible on a production engine control unit. The DAP MPC controller also displayed some undesirable oscillations, produced some visible smoke, and underperformed on gentle driving patterns such as the NEDC. More work is needed to address these issues.

ACKNOWLEDGMENT

The authors would like to thank Dr. Hayato Nakada for his insight and support during all phases of the project, Naoto Tateishi for his help in obtaining the experimental data, and the rest of the Toyota Motor Corporation engineers for their hospitality during our visits.

[¶¶] This can help manufacturers meet emissions targets without major cost increases.

References

1. Rawlings JB, Mayne DQ. *Model predictive control: Theory and design*. Nob Hill Pub . 2009.
2. Grüne L, Pannek J. Nonlinear Model Predictive Control. In: *Nonlinear Model Predictive Control*Springer. 2017 (pp. 45–69).
3. Del Re L, Allgöwer F, Glielmo L, Guardiola C, Kolmanovsky I. *Automotive model predictive control: models, methods and applications*. 402. Springer . 2010.
4. Di Cairano S, Kolmanovsky IV. Real-time optimization and model predictive control for aerospace and automotive applications. In: *2018 annual American control conference (ACC)*IEEE. ; 2018: 2392–2409.
5. Ortner P, Re dL. Predictive control of a diesel engine air path. *IEEE Transactions on Control Systems Technology* 2007; 15(3): 449–456.
6. Stewart G, Borrelli F. A model predictive control framework for industrial turbodiesel engine control. In: *2008 47th IEEE conference on decision and control*IEEE. ; 2008: 5704–5711.
7. Maruyama T, Shimura T, Ejiri A, Ikai Y, Shimotani K. Model predictive control applied to a diesel engine air-path system with dead time. In: *SICE Annual Conference (SICE), 2011 Proceedings of*IEEE. ; 2011: 2628–2633.
8. Zhao D, Liu C, Stobart R, Deng J, Winward E, Dong G. An explicit model predictive control framework for turbocharged diesel engines. *IEEE Transactions on Industrial Electronics* 2014; 61(7): 3540–3552.
9. Zhang Y, Lu G, Xu H, Li Z. Tuneable model predictive control of a turbocharged diesel engine with dual loop exhaust gas recirculation. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* 2018; 232(8): 1105–1120.
10. Huang M, Zaseck K, Butts K, Kolmanovsky I. Rate-based model predictive controller for diesel engine air path: Design and experimental evaluation. *IEEE Transactions on Control Systems Technology* 2016; 24(6): 1922–1935.
11. Iwadare M, Ueno M, Adachi S. Multi-variable air-path management for a clean diesel engine using model predictive control. *SAE International Journal of Engines* 2009; 2(1): 764–773.
12. Huang M, Nakada H, Butts K, Kolmanovsky I. Nonlinear model predictive control of a diesel engine air path: A comparison of constraint handling and computational strategies. *IFAC-PapersOnLine* 2015; 48(23): 372–379.
13. Ortner P, Bergmann R, Ferreau HJ, Del Re L. Nonlinear model predictive control of a diesel engine airpath. *IFAC Proceedings Volumes* 2009; 42(2): 91–96.
14. Herceg M, Raff T, Findeisen R, Allgowe F. Nonlinear model predictive control of a turbocharged diesel engine. In: *2006 IEEE Conference on Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control*IEEE. ; 2006: 2766–2771.
15. Murilo A, Alamir M, Alberer D. A general NMPC framework for a diesel engine air path. *International Journal of Control* 2014; 87(10): 2194–2207.
16. Harder K, Buchholz M, Niemeyer J, Remele J, Graichen K. A real-time nonlinear MPC scheme with emission constraints for heavy-duty diesel engines. In: *American Control Conference (ACC), 2017*IEEE. ; 2017: 240–245.
17. Gelso ER, Dahl J. Air-path control of a heavy-duty EGR-VGT diesel engine. *IFAC-PapersOnLine* 2016; 49(11): 589–595.
18. Harder K, Buchholz M, Niemeyer J, Remele J, Graichen K. Nonlinear MPC with emission control for a real-world off-highway diesel engine. In: *Advanced Intelligent Mechatronics (AIM), 2017 IEEE International Conference on*IEEE. ; 2017: 1768–1773.
19. Broomhead T, Manzie C, Hield P, Shekhar R, Brear M. Economic Model Predictive Control and Applications for Diesel Generators. *IEEE Transactions on Control Systems Technology* 2016.

20. Karlsson M, Ekholm K, Strandh P, Johansson R, Tunestål P. Multiple-input multiple-output model predictive control of a diesel engine. *IFAC Proceedings Volumes* 2010; 43(7): 131–136.

21. Liao-McPherson D, Huang M, Kolmanovsky I. A Regularized and Smoothed Fischer-Burmeister Method for Quadratic Programming with Applications to Model Predictive Control. *IEEE Transactions on Automatic Control* 2018: 1-1. doi: 10.1109/TAC.2018.2872201

22. Walker K, Samadi B, Huang M, Gerhard J, Butts K, Kolmanovsky I. Design Environment for Nonlinear Model Predictive Control. tech. rep., SAE Technical Paper; 2016.

23. Liao-McPherson D, Kim S, Butts K, Kolmanovsky I. A cascaded economic model predictive control strategy for a diesel engine using a non-uniform prediction horizon discretization. In: *2017 IEEE Conference on Control Technology and Applications (CCTA)*; 2017: 979-986.

24. Huang M, Liao-McPherson D, Kim S, Butts K, Kolmanovsky I. Toward Real-Time Automotive Model Predictive Control: A Perspective from a Diesel Air Path Control Development. In: *2018 Annual American Control Conference (ACC)*; 2018: 2425-2430

25. Reşitoğlu İA, Altinişik K, Keskin A. The pollutant emissions from diesel-engine vehicles and exhaust aftertreatment systems. *Clean Technologies and Environmental Policy* 2015; 17(1): 15–27.

26. Guzzella L, Onder C. *Introduction to modeling and control of internal combustion engine systems*. Springer Science & Business Media . 2009.

27. Eriksson L, Nielsen L. *Modeling and control of engines and drivelines*. John Wiley & Sons . 2014.

28. Röpke S, Schweimer G, Strauss T. NOx formation in diesel engines for various fuels and intake gases. tech. rep., SAE Technical Paper; 1995.

29. Garone E, Di Cairano S, Kolmanovsky I. Reference and command governors for systems with constraints: A survey on theory and applications. *Automatica* 2017; 75: 306–328.

30. Vahidi A, Kolmanovsky I, Stefanopoulou A. Constraint handling in a fuel cell system: A fast reference governor approach. *IEEE Transactions on Control Systems Technology* 2006; 15(1): 86–98.

31. Kalabić UV, Buckland JH, Cooper SL, Wait SK, Kolmanovsky IV. Reference governors for enforcing compressor surge constraints. *IEEE Transactions on Control Systems Technology* 2016; 24(5): 1729–1739.

32. Kolmanovsky IV, Jaw LC, Merrill W, Van HT. Robust control and limit protection in aircraft gas turbine engines. In: *2012 IEEE International Conference on Control Applications*IEEE. ; 2012: 812–819.

33. Tóth R. Modeling and identification of linear parameter-varying systems. 2010.

34. Kolmanovsky I, Moral P, Van Nieuwstadt M, Stefanopoulou A. Issues in modelling and control of intake flow in variable geometry turbocharged engines. *Chapman and Hall CRC research notes in mathematics* 1999: 436–445.

35. Davis PJ, Rabinowitz P. *Methods of numerical integration*. Courier Corporation . 2007.

36. Rishavy E, Hamilton S, Ayers J, Keane M. Engine control optimization for best fuel economy with emission constraints. tech. rep., SAE Technical Paper; 1977.

37. Langouët H, Métivier L, Sinoquet D, Tran QH. Optimization for engine calibration. In: *ENGOPT International conference on engineering optimization, Rio de Janeiro, Brazil*Citeseer. ; 2008: 1–5.

38. Popovic D, Jankovic M, Magner S, Teel AR. Extremum seeking methods for optimization of variable cam timing engine operation. *IEEE Transactions on Control Systems Technology* 2006; 14(3): 398–407.

39. Atkinson C, Mott G. Dynamic model-based calibration optimization: An introduction and application to diesel engines. tech. rep., SAE Technical Paper; 2005.

40. Hagena JR, Filipi Z, Assanis DN. Transient diesel emissions: analysis of engine operation during a tip-in. tech. rep., SAE Technical Paper; 2006.

41. Gao Z, Conklin J, Daw CS, Chakravarthy VK. A proposed methodology for estimating transient engine-out temperature and emissions from steady-state maps. *International Journal of Engine Research* 2010; 11(2): 137–151.

42. Fredriksson J, Egardt B. Estimating exhaust manifold pressure in a turbocharged diesel engine. In: *Control Applications, 2002. Proceedings of the 2002 International Conference on*. 2. IEEE. ; 2002: 701–706.

43. Olin PM. A mean-value model for estimating exhaust manifold pressure in production engine applications. tech. rep., SAE Technical Paper; 2008.

44. Cagienard R, Grieder P, Kerrigan EC, Morari M. Move blocking strategies in receding horizon control. *Journal of Process Control* 2007; 17(6): 563–570.

45. Tippett MJ, Tan CK, Bao J. Non-Constant Prediction-Step MPC for Processes with Multi-Scale Dynamics. *IFAC Proceedings Volumes* 2014; 47(3): 3068–3073.

46. Morari M, Maeder U. Nonlinear offset-free model predictive control. *Automatica* 2012; 48(9): 2059–2067.

47. Pannocchia G, Gabiccini M, Artoni A. Offset-free MPC explained: novelties, subtleties, and applications. *IFAC-PapersOnLine* 2015; 48(23): 342–351.

48. Zhu Q, Onori S, Prucka R. An Economic Nonlinear Model Predictive Control Strategy for SI Engines: Model-Based Design and Real-Time Experimental Validation. *IEEE Transactions on Control Systems Technology* 2017(99): 1–15.

49. Albin T, Ritter D, Liberda N, Quirynen R, Diehl M. In-vehicle realization of nonlinear MPC for gasoline two-stage turbocharging airpath control. *IEEE Transactions on Control Systems Technology* 2018; 26(5): 1606–1618.

50. Quindlen JF, Topcu U, Chowdhary G, How JP. Active sampling for closed-loop statistical verification of uncertain nonlinear systems. In: *2018 Annual American Control Conference (ACC)*IEEE. ; 2018: 6259–6265.

51. Quindlen JF, Topcu U, Chowdhary G, How JP. Closed-loop statistical verification of stochastic nonlinear systems subject to parametric uncertainties. In: *2018 Annual American Control Conference (ACC)*IEEE. ; 2018: 5460–5466.

52. Mozyrska D, Pawłuszewicz E, Wyrwas M. Local observability and controllability of nonlinear discrete-time fractional order systems based on their linearisation. *International Journal of Systems Science* 2017; 48(4): 788–794.

53. Atkinson C, Allain M, Zhang H. Using model-based rapid transient calibration to reduce fuel consumption and emissions in diesel engines. tech. rep., SAE Technical Paper; 2008.

54. Nozaki Y, Fukuma T, Tanaka K. Development of a rule-based calibration method for diesel engines. tech. rep., SAE Technical Paper; 2005.

55. Malikopoulos AA, Assanis DN, Papalambros PY. Real-time self-learning optimization of diesel engine calibration. *Journal of engineering for gas turbines and power* 2009; 131(2).

56. Diehl M, Bock HG, Schlöder JP. A real-time iteration scheme for nonlinear optimization in optimal feedback control. *SIAM Journal on control and optimization* 2005; 43(5): 1714–1736.

57. Diehl M, Findeisen R, Allgöwer F, Bock HG, Schlöder JP. Nominal stability of real-time iteration scheme for nonlinear model predictive control. *IEE Proceedings-Control Theory and Applications* 2005; 152(3): 296–308.

58. Liao-McPherson D, Nicotra M, Kolmanovsky I. Time-distributed optimization for real-time model predictive control: Stability, robustness, and constraint satisfaction. *Automatica* 2020; 117: 108973. doi: https://doi.org/10.1016/j.automatica.2020.108973

59. Pannocchia G, Rawlings JB, Wright SJ. Conditions under which suboptimal nonlinear MPC is inherently robust. *Systems & Control Letters* 2011; 60(9): 747–755.

60. Allan DA, Bates CN, Risbeck MJ, Rawlings JB. On the inherent robustness of optimal and suboptimal nonlinear MPC. *Systems and Control Letters* 2017; 106: 68 - 78. doi: https://doi.org/10.1016/j.sysconle.2017.03.005

61. Ward DD. MISRA Standards for Automotive Software. In: *2006 2nd IEE Conference on Automotive Electronics*; 2006: 5-18.

62. Atkins RD. An introduction to engine testing and development. tech. rep., SAE Technical Paper; 2009.

63. Martyr AJ, Plint MA. *Engine testing: theory and practice*. Elsevier . 2011.

64. Kelley C. Iterative methods for linear and nonlinear equations, vol. 16 of Frontiers in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1995. *With separately available software*.

65. Liao-McPherson* D, Kolmanovsky I. The FBstab Algorithm for Model Predictive Control: An Implicit Condensing Approach. In: *To appear at the 2019 IEEE Conference on Decision and Control (CDC)*.

66. Frison G, Kouzoupis D, Sartor T, Zanelli A, Diehl M. BLASFEO: Basic linear algebra subroutines for embedded optimization. *ACM Transactions on Mathematical Software (TOMS)* 2018; 44(4): 42.

67. Bemporad A, Bernardini D, Long R, Verdejo J. Model predictive control of turbocharged gasoline engines for mass production. tech. rep., SAE Technical Paper; 2018.

68. Grimm G, Messina MJ, Tuna SE, Teel AR. Model predictive control: for want of a local control Lyapunov function, all is not lost. *IEEE Transactions on Automatic Control* 2005; 50(5): 546–558.

69. Jiang ZP, Wang Y. A converse Lyapunov theorem for discrete-time systems with disturbances. *Systems & control letters* 2002; 45(1): 49–58.

8 | TABLES

TABLE 1 Summary of problem sizes and execution times. The ECU execution times are estimated for a 256 MHz ECU using clock scaling, see Section 6.1 for more details.

	Average execution time [μs]	Maximum execution time [μs]	Maximum estimated ECU execution time [ms]	Number of decision variables	Number of hard constraints
SMPC	530	550	5.6	17	41
NMPC FF	31	32	0.33	18	0
NMPC FB	127	133	1.4	18	0

TABLE 2 Summary of the results obtained using the DAP MPC controller.

	WLTC aggressive tuning	WLTC conservative tuning	NEDC aggressive tuning
	[% difference]	[% difference]	[% difference]
NO_x	-16	-11	1.4
THC	-14	-4	-2
Fuel	0.67	0.54	0.9
V_{RMSE}	1	4	2
Smoke	24	-49	0

APPENDIX

Proof of Theorem 2

Our goal is to apply[68] Theorem 1 to infer the existence of a Lyapunov function for the closed-loop system. To show this we verify the conditions of the theorem, referred to as (SA1-SA4) in[68]. Since the SMPC OCP has a penalty on Δu, and is thus nonstandard, we work with an equivalent augmented system by defining

$$y = \begin{bmatrix} x \\ u \end{bmatrix}, \text{ and } y^+ = g(y, \Delta u) = \begin{bmatrix} f(x,u) \\ u + \Delta u \end{bmatrix}, \tag{1}$$

which is the so-called input velocity form of the original system. We seek a Lyapunov function which proves the stability of (1). Note that recursive feasibility, i.e., $y_k \in \mathbb{Y}(\rho)$, is automatic thanks to Theorem 1. We consider each condition in turn:

(SA1) Trivially satisfied since the stage cost (15) is semidefinite and continuous, the terminal cost is null.

(SA2) The OCP (14) is guaranteed to be feasible by construction and has a strongly convex objective function, thus (14) has a solution and the infimum is achieved.

(SA3) Consider a candidate detection function $W(y) = V(x) + k(u)$ where $V(x)$ is a Lyapunov function which proves asymptotic stability of the inner-loop and $k(u) = l(u, 0)$. Since the NMPC controller is assumed to stabilize the engine, $V(x)$ can be taken to be its value function. Alternatively, since the inner-loop is stable by assumption, a converse Lyapunov theorem, e.g.,[69], could be invoked. Since V is a Lyapunov function it is bounded above and below by class $\mathcal{K}_\infty$ functions. In addition, $k(u)$ is positive definite, strictly convex with $k(\bar{u}) = 0$, and is Lipschitz continuous on the set, $\{u \mid 0 \le \chi^{trg} \le \max_{\mathbb{P}} \bar{\chi}_{egr}(\rho),\ 0 \le q \le \max_{\mathbb{P}} q^{trg}\}$, with Lipschitz constant L and thus can be bounded from above by a class $\mathcal{K}_\infty$ function. Following the notation of[68] we define our tracking measure as $\sigma(y) = V(x) + k(u)$. Let $\beta_1(\sigma(y)) = \eta_1(||x - \bar{x}||) + \eta_2(||u - \bar{u}||)$, where $\eta_1 \in \mathcal{K}_\infty$ upperbounds $V(x)$ and $\eta_2 \in \mathcal{K}_\infty$ upperbounds $k(u)$. Then $W(y) \le \beta_1(\sigma(y))$ and $\beta_1 \in \mathcal{K}_\infty$. Now consider

$$\begin{aligned} &W(g(y, \Delta u)) - W(y) \\ &= V(f(x,u)) - V(x) + k(u + \Delta u) - k(u) \\ &\le V(x^+) - V(x) + L||\Delta u||_R, \end{aligned}$$

and define

$$\alpha_1(x) = \begin{cases} (V(x) - V(x^+))/V(x), & x \neq \bar{x}, \\ 0 & x = \bar{x}, \end{cases} \tag{2}$$

Since V is a Lyapunov function $\forall x \in \mathbb{X} \setminus \{\bar{x}\}$, $V(x^+) - V(x) < 0,\ V(x) > 0$ and thus $\alpha_1(x) \in [0, 1)$ and $V(x^+) - V(x) + \alpha_1(x)V(x) = 0$. Continuing, using that $V(x^+) - V(x) = -\alpha_1(x)V(x) \le 0$, we have that

$$\begin{aligned} W(g(y, \Delta u)) - W(y) &\le V(x^+) - V(x) + L||\Delta u||_R, \\ &\le -\alpha_1(x)V(x) + (\gamma_1 - \alpha_1(x))k(u) + L||\Delta u||_R, \\ &\le -\alpha_1(x)(V(x) + k(u)) + \gamma_1 k(u) + L||\Delta u||_R, \\ &\le -\alpha(x)\beta_1(\sigma(y)) + \gamma_1 k(u) + L||\Delta u||_R, \end{aligned}$$

for any $\gamma_1 > 1 > \alpha_1(x)$. Since $\alpha_1(x) < 1$, for any $\gamma_2 > 1$, $\alpha_1(x)(V(x) + k(u)) \le \gamma_2\beta_1(\sigma(y))$. Thus $\beta_2(\cdot) = \gamma_2\beta_1(\cdot) \in \mathcal{K}_\infty$ is an appropriate comparison function. The remaining terms can be upper bounded by the function $\gamma_1 l(u, \Delta u) + \gamma_3\sqrt{l(u, \Delta u)}$ for sufficiently large γ_3. Thus we can take $\beta_3(\cdot) = \gamma_1(\cdot) + \gamma_3\sqrt{(\cdot)} \in \mathcal{K}_\infty$ as the final required comparison function. The existence of β_1, β_2, and β_3 verifies the detectability condition[68] Definition 1.

(SA4) Boundedness of the value function can be established using Assumption 2 (Asymptotic controllability) combined with[2] Lemma 6.6. Verifying (SA1-SA4) completes the proof.