

# Machine Learning Integrated with Model Predictive Control for Imitative Optimal Control of Compression Ignition Engines

Armin Norouzi\* Saeid Shahpouri\* David Gordon\*  
Alexander Winkler\*\* Eugen Nuss\*\*\* Dirk Abel\*\*\*  
Jakob Andert\*\* Mahdi Shahbakhti\* Charles Robert Koch\*

\* *Department of Mechanical Engineering, University of Alberta,  
Edmonton, Canada*

\*\* *Teaching and Research Area Mechatronics in Mobile Propulsion,  
RWTH Aachen University, Aachen, Germany*

\*\*\* *Institute of Automatic Control, RWTH Aachen University,  
Germany*

---

**Abstract:** The high thermal efficiency and reliability of the compression-ignition engine makes it the first choice for many applications. For this to continue, a reduction of the pollutant emissions is needed. One solution is the use of Machine Learning (ML) and Model Predictive Control (MPC) to minimize emissions and fuel consumption, without adding substantial computational cost to the engine controller. ML is developed in this paper for both modeling engine performance and emissions and for imitating the behaviour of a Linear Parameter Varying (LPV) MPC. Using a support vector machine-based linear parameter varying model of the engine performance and emissions, a model predictive controller is implemented for a 4.5 L Cummins diesel engine. This online optimized MPC solution offers advantages in minimizing the NO<sub>x</sub> emissions and fuel consumption compared to the baseline feedforward production controller. To reduce the computational cost of this MPC, a deep learning scheme is designed to mimic the behavior of the developed controller. The performance in reducing NO<sub>x</sub> emissions at a constant load by the imitative controller is similar to that of the online optimized MPC, however, the imitative controller requires 50 times less computation time when compared to that of the online MPC optimization.

*Keywords:* Diesel engines, Linear Parameter Variable Model, Machine learning, Support Vector Machine, Model Predictive Control, Deep Learning

---

## 1. INTRODUCTION

Compression Ignition (CI) engines are commonly used in various transportation applications, from public transportation to personal vehicles. The diesel engine has a high thermal efficiency, long lifetime as well as fuel economy advantages at full-load and part-load conditions when compared to spark ignition engines (Ortner and Del Re, 2007). Although diesel engines have many advantages and are commonly used, they play a significant role in the environmental pollution problems worldwide (Reşitoğlu et al., 2015). Hybridization and electrification are proceeding rapidly for passenger vehicles, but the uptake has been slow for commercial heavy-duty trucks due to the high battery costs and decreased cargo capacity (Heid et al., 2017).

Real Driving Emissions (RDE) requirements have been implemented beginning with Euro 6d. The challenge with RDE requirements are that test results are significantly affected by various external factors including the ambient conditions, traffic, and driver behavior. Based on RDE

legislation, engines should operate cleanly under all conditions, which makes engine design and calibration much more challenging (Norouzi et al., 2021b). Complying with RDE legislation is a significant shift from the previous legislation. Intelligent engine control strategies that take advantage of Machine Learning (ML) methods and utilize model-based control are one way to make substantial progress toward meeting the rigorous emission regulations.

Model Predictive Control (MPC) has been shown to be an effective model based control strategy. This is the result of MPC considering constraints on inputs, outputs, and states. Additionally, MPC offers optimal performance control by utilizing a future horizon while optimizing the current control law (Liao-McPherson et al., 2020; Stewart and Borrelli, 2008; Winkler et al., 2021). MPC has been used for CI control of various different engines from light to heavy duty and for broad range of applications (Liao-McPherson et al., 2020; Zhao et al., 2014; Yin et al., 2020; Liu et al., 2021). Despite the many benefits of MPC, it has a high computational cost and the performance depends on the accuracy of the embedded model. There is a trade-off based on the complexity of the model between prediction accuracy and computation time. This is the

\* Corresponding Author: Armin Norouzi-(e-mail: norouziy@ualberta.ca).

main reason that physics-based models, such as detailed 3D combustion models, which offer a high prediction accuracy, have seen limited MPC implementation due to high computational times. These models first need to be simplified or linearized to be feasible for MPC implementation (Stewart and Borrelli, 2008; Norouzi et al., 2019).

A state-space Linear Parameter Varying (LPV) model is capable of providing an accurate combustion model while utilizing a simpler structure (Irdmoussa et al., 2019). LPV is a state-space system that updates the system matrices based on scheduling parameters. The LPV model can fill the gap between complex nonlinear models and inaccurate but simple linear models. It employs an array of linear models characterized by scheduling variables. The LPV model has been used in internal combustion control within an MPC framework in literature for both CI and Spark Ignition (SI) engines (Irdmoussa et al., 2019; Basina et al., 2020).

The traditional use of an LPV model requires the evaluation of a complex model in a grid of scheduling parameters; however, ML provides a systematic way to create a state-space LPV model directly from measurement data. SVM-LPV (Rizvi et al., 2015) uses a Least-Square Support Vector Machine (LS-SVM) framework to update state-space matrices. This method has been previously applied to Internal Combustion (IC) engine performance modeling (Irdmoussa et al., 2019; Basina et al., 2020). In this paper, a Support Vector Machine-based state-space LPV algorithm (SVM-LPV) is used to model engine performance and the engine-out  $\text{NO}_x$  emissions to show the capability of the SVM-LPV technique. To the best of authors' knowledge, this paper presents the first study to use this technique for emission modeling and control of an IC engine. To make this algorithm reach the optimum solution, a Bayesian optimization has been added to optimize the hyperparameters of the SVM.

Modeling of the engine is the first use of ML in this paper. Then cloning the behaviour of an MPC is the second use of ML presented in this paper which is called learning, approximate, or imitation MPC. This learning MPC utilizes a deep neural network to imitate the behaviour of the designed MPC and it has been used successfully in vehicle dynamics (Zhang et al., 2019), and the heating, ventilation, and air conditioning (HVAC) industry (Toub et al., 2019). In this case it has been shown to provide a significant computational time reduction in comparison with traditional online optimization of MPC. However, to the authors' knowledge this imitative LPV-MPC has not been used for IC engine control.

Based on the current literature, this paper has the following contributions:

- (1) Machine Learning based modeling for MPC design
  - (a) An SVM-LPV model is adapted to develop a linear parameter-varying model for engine-out  $\text{NO}_x$  emissions and engine performance metrics
  - (b) An LPV MPC based on a ML model to minimize engine-out emissions and fuel consumption while maintaining the same output torque performance and comparison with a benchmark model (Cummins calibrated ECU-based GT-power model)

- (2) Machine Learning based control to reduce computational time of MPC
  - (a) Developed imitation based controller using deep neural network to clone behavior of LPV-MPC to reduce the computational time of optimization

## 2. ENGINE SIMULATION MODEL

A 4.5-liter medium-duty Cummins Diesel engine is the focus of this work. The Cummins QSB4.5 160 Diesel engine is a turbocharged inter-cooled engine meeting Tier 3 emissions legislation. A dSPACE MicroAutoBox II is used to control the intake manifold pressure, engine speed, load, injected fuel amount, and fuel rail pressure. A Kistler piezoelectric pressure sensor is used to measure the in-cylinder pressure. Engine-out Nitrogen Oxides ( $\text{NO}_x$ ) emissions from the engine are measured using a Bosch sensor with ECM electronics (P/N: 06-05). Further experimental setup details can be found in (Norouzi et al., 2021a, 2020). The engine is used to parameterize the simulation model.

A Detailed Physical Model (DPM) of the real engine was created using GT-Power to test the proposed control strategies in an Engine Simulation Model (ESM) platform. This model contains several chemical and physical sub-models that are used to simulate the complex combustion and gas exchange processes. This experimentally validated model is from our previous studies with an accuracy of  $\pm 5.8\%$ ,  $\pm 4.6\%$ , and  $\pm 18.1\%$  in the prediction of the maximum in-cylinder pressure, intake manifold pressure, and  $\text{NO}_x$ , respectively (Shahpouri et al., 2021b,a).

## 3. METHODOLOGY

In this paper, two methods of combining ML and MPC are presented consisting of ML-based modeling and ML imitation control which tested in simulation. An overview of the methodology is depicted in Fig. 1. First, randomly generated inputs are fed into the ESM and the output engine performance is recorded for control modeling. A Least-square Support Vector Machine based Linear Parameter-Varying (SVM-LPV) model is then developed using the input-output data. This model is used for the design of the LPV-MPC controller. Finally, this MPC controller is used to train the ML based imitation controllers. To assess, the LPV-MPC controller performance it is compared to a Linear Autoregressive with Extra Input (ARX) based linear MPC using a GT-power/MATLAB/SIMULINK co-simulation.

## 4. MODELING

### 4.1 Support Vector Machine based Linear Parameter Varying (LPV) Model

The LPV model is defined as:

$$\begin{aligned} \mathbf{x}(k+1) &= A(p(k))\mathbf{x}(k) + B(p(k))\mathbf{u}(k) \\ \mathbf{y}(k) &= C(p(k))\mathbf{x}(k) + D(p(k))\mathbf{u}(k) \end{aligned} \quad (1)$$

Where the state matrices (A,B,C and D) are a function of scheduling parameters,  $p(k)$ . An SVM-based algorithm

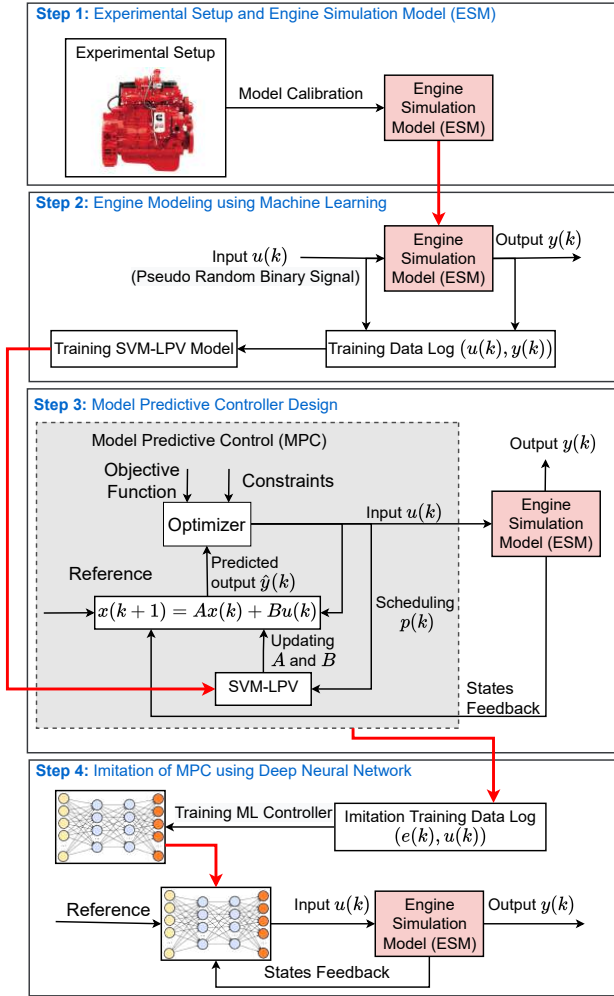


Fig. 1. Modeling and controller design procedure based on Engine Simulation Model (ESM)

is used to update the state matrices. In this study  $u(k)$ ,  $x(k)$ , and  $y(k)$  are defined as

$$\begin{aligned} u(k) &= [FQ(k) \text{ SOI}(k) \text{ VGT}(k)]^T, \\ x(k) &= [T_{\text{out}}(k) \text{ } P_{\text{man}}(k) \text{ } \text{NO}_x(k)]^T, \\ y(k) &= [T_{\text{out}}(k) \text{ } \text{NO}_x(k)]^T, \end{aligned} \quad (2)$$

where  $FQ(k)$  is Fuel Quantity (FQ),  $\text{SOI}(k)$  is the Start Of main Injection (SOI),  $\text{VGT}(k)$  is the Variable Geometric Turbine (VGT) rate,  $T_{\text{out}}$  is the engine output torque,  $P_{\text{man}}$  is the intake manifold pressure, and  $\text{NO}_x(k)$  is the engine-out Nitrogen Oxides ( $\text{NO}_x$ ) emissions. The SVM-LPV algorithm developed in (Rizvi et al., 2015) is adapted for this specific problem. Additionally, to tune the hyper-parameters of the SVM-LPV, a Bayesian optimization is implemented. It is assumed that the output of the model is equal or partially equal to the states of the system and that the system states are measurable. Thus, the matrix  $C$  is not scheduled, and matrix  $D$  is identically zero. Then, the model can be simplified as

$$\mathbf{x}(k+1) = A(p(k)) \mathbf{x}(k) + B(p(k)) \mathbf{u}(k) \quad (3)$$

where  $u(k) \in \mathbb{R}^{n_u}$ ,  $x(k) \in \mathbb{R}^{n_x}$ , and  $p(k) \in \mathbb{R}^{n_p}$  are inputs, states, and scheduling parameter at  $k$ , and  $A \in \mathbb{R}^{(n_x \times n_x)}$  and  $B \in \mathbb{R}^{(n_x \times n_u)}$  are state-space model matrices ( $n_x$  and  $n_u$  are the number of states and manipulated variables,

respectively). To formulate our problem in an SVM framework,  $A(p(k))$  and  $B(p(k))$  can be written as

$$A(p(k)) = W_1 \phi_1(p(k)), \quad B(p(k)) = W_2 \phi_2(p(k)) \quad (4)$$

where  $\phi_1 \in \mathbb{R}^{(n_x \times n_h)}$  and  $\phi_2 \in \mathbb{R}^{(n_u \times n_h)}$  are the high dimension feature space. Substituting Eq. 4 into Eq. 3 results in

$$\mathbf{x}(k+1) = \underbrace{[W_1 \quad W_2]}_{\mathbf{W}} \underbrace{\begin{bmatrix} (\phi_1(p(k)) \mathbf{x}(k))^T \\ (\phi_2(p(k)) \mathbf{u}(k))^T \end{bmatrix}}_{\Phi(k)^T} \quad (5)$$

The residual error of modeling,  $e(k) \in \mathbb{R}^{n_x}$ , is defined as

$$e(k) = W \Phi(k)^T - \mathbf{x}(k+1) \quad (6)$$

and it can be added to Eq. 5 as

$$\mathbf{x}(k+1) = W \Phi(k)^T + e(k) \quad (7)$$

The LS-SVM cost function is then defined as

$$\text{Minimize: } \frac{1}{2} \|W\|_2^2 + \frac{1}{2} \sum_{j=1}^N e(j)^T \gamma e(j) \quad (8)$$

$$\text{Subject to: } \mathbf{x}(j+1) = W \Phi(j)^T + e(j)$$

Where  $N$  is the number of training samples used for modeling and  $j$  is the discrete sample time defined from 1 to  $N$ . In this LS-SVM formulation,  $\gamma$  is a diagonal matrix of size  $n_x$  that acts as the regularization parameters. The Lagrangian function can then be calculated based on Eq. 8 as

$$\begin{aligned} L(W) &= \frac{1}{2} \|W\|_2^2 + \frac{1}{2} \sum_{j=1}^N e(j)^T \gamma e(j) \\ &\quad - \sum_{j=1}^N \alpha_j^T (W \Phi(j)^T + e(j) - \mathbf{x}(j+1)) \end{aligned} \quad (9)$$

where  $\alpha_j^T \in \mathbb{R}^{n_x}$  are the discrete time Lagrange multipliers. To find the optimum  $W$ , the derivatives of the Lagrangian, Eq. 9, with respect to optimization variables must be zero as

$$\frac{\partial L}{\partial W} = 0 \rightarrow W = \sum_{j=1}^N \alpha_j \Phi(j) \quad (10a)$$

$$\frac{\partial L}{\partial e} = 0 \rightarrow \alpha = \gamma e \quad (10b)$$

$$\frac{\partial L}{\partial \alpha} = 0 \rightarrow x(j+1) = W \Phi(j)^T + e(j) \quad (10c)$$

Substituting Eqs. 10a and 10c into Eq. 7 results in

$$x(k+1) = \sum_{j=1}^N \alpha_j \underbrace{\Phi(j) \Phi(k)^T}_{[\Omega]} + \gamma^{-1} \alpha(k) \quad (11)$$

where  $\Phi(j) \Phi(k)^T$  is the kernel matrix,  $[\Omega]$ , and can be defined as

$$\begin{aligned} [\Omega] &= x(j)^T K(p(j), p(k)) x(k) \\ &\quad + u(j)^T K(p(j), p(k)) u(k) \end{aligned} \quad (12)$$

where  $K(p(j), p(k))$  is a nonlinear kernel function. Usually, a Radial Basis Function (RBF) kernel,  $K_{RBF}$ , is used as the kernel function, which is defined as

$$K_{RBF}(p(j), p(k)) = \exp\left(-\frac{\|p(j) - p(k)\|^2}{2\sigma}\right) \quad (13)$$

where  $\sigma$  is a free parameter that is tuned during the hyperparameter optimization and  $\|p(j) - p(k)\|^2$  is the L<sub>2</sub>

norm between the two feature vectors. Writing Eq. 11 in a compact notation yields

$$X = \alpha \Omega + \gamma^{-1} \alpha \quad (14)$$

where  $X = [x(1) \dots x(N)]$ . Solving this equation for  $\alpha$  results in

$$\alpha = (I_N \odot \gamma^{-1} + \Omega^T \odot I_{n_x})^{-1} X \quad (15)$$

where  $I_N$  and  $I_{n_x}$  indicate the identity matrix in the dimension of training sample size by  $\mathbf{x}$  size and  $\odot$  is element-wise or Kronecker product. By calculating  $\alpha$ , the state-space model matrices can be calculated as

$$\begin{aligned} A(p(k)) &= \sum_{j=1}^N \alpha_j \mathbf{x}(j)^T K_{RRBF}(p(j), p(k)) \\ B(p(k)) &= \sum_{j=1}^N \alpha_j \mathbf{u}(j)^T K_{RRBF}(p(j), p(k)) \end{aligned} \quad (16)$$

where the  $j$  index shows the data used in the training set. The model is developed using the training set of  $x(j)$   $j \in (1, 2, \dots, N)$  and  $u(j)$   $j \in (1, 2, \dots, N)$ . Additionally, the scheduling parameter,  $p$  is also given in the training set as  $p(j)$   $j \in (1, 2, \dots, N)$ .

#### 4.2 Bayesian Hyperparameters Optimization

The SVM-LPV model has two main hyperparameters:  $\gamma$ , the regularization coefficient, and  $\sigma$  the free parameters of the kernel. The cost function of the hyperparameter optimization is defined as

$$J(\gamma, \sigma) = \frac{1}{N_{CV}} \sum_{i=1}^{N_{CV}} (\hat{X}(i) - X(i))^2 \quad (17)$$

where  $\hat{\mathbf{x}}(i)$  is the modeled output and  $\mathbf{x}(i)$  is the measured states and  $N_{CV}$  is the validation dataset that is used for optimizing the parameters. Bayesian Optimization utilizes Bayes Theorem to direct a search of a global optimization problem. The cost function versus iteration number for 100 iterations of the Bayesian optimization is shown in Fig. 2. In this figure, the Bayesian optimization approaches to the global optimum after 76 iterations. The

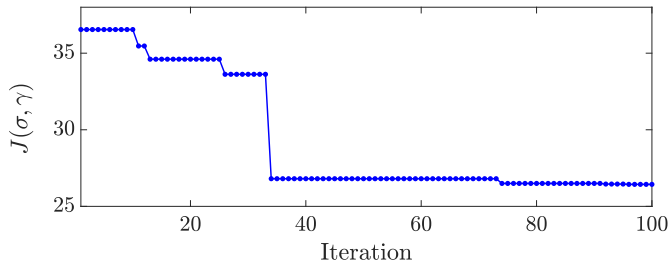


Fig. 2. Bayesian optimization results for LPV-SVM model parameter optimization showing the cost function ( $J$ ) values versus the integration number

Bayesian-optimized SVM-LPV can capture all states with an accuracy of 7.3%, 1.1%, and 1.9% for  $\text{NO}_x$ ,  $T_{\text{out}}$ , and  $P_{\text{man}}$  respectively when using the training data. As this model will be used for the Model Predictive Controller (MPC), the accuracy of this model for a new data-set is critical. The SVM-LPV is compared to a linear state-space model called Autoregressive with Extra Input (ARX) that

is commonly used for system identification. The ARX-based discrete-time state-space model of the Diesel engine emissions and performance is trained using the same training dataset as the SVM-LPV resulting in:

$$\begin{aligned} A &= \begin{bmatrix} 0.7286 & 7.1252 & -0.0019 \\ 0.0002 & 0.9859 & 8.9878 \times 10^{-6} \\ -0.6105 & 33.94287 & 0.9076 \end{bmatrix} \\ B &= \begin{bmatrix} 1.2639 & -1.0899 & 1.0084 \times 10^{-5} \\ -0.0007 & 0.0014 & -1.01397 \times 10^{-5} \\ 2.9360 & -8.2453 & -0.0106 \end{bmatrix} \\ C &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (18)$$

The linear ARX and SVM-LPV model are both run simultaneously in one simulation where both models are compared against the ESM. Both models are evaluated using unseen test data. Fig. 3 shows the model comparison where both models have a high accuracy within 5% normalized root mean square error (NRMSE) for estimating the output torque. However, the linear model fails to provide an accurate estimation for intake manifold pressure and  $\text{NO}_x$  emissions. For the intake manifold pressure, the SVM-LPV model has significantly better estimation than the linear model where the NRMSE is 0.95% in comparison to 14.81% for the linear model. For  $\text{NO}_x$ , the SVM-LPV estimates with less than 7% error. While the linear ARX model is unable to accurately capture the emission level resulting in a NRMSE of 32.3%. Next, using the developed SVM-LPV model, an MPC combustion controller will be designed.

## 5. MODEL PREDICTIVE CONTROLLER DESIGN

### 5.1 Controller Design

The controller's objective in this study is to minimize engine-out  $\text{NO}_x$  emissions and fuel consumption while maintaining the desired engine output torque. The cost function  $J(\mathbf{u}(\cdot|k))$  of the finite horizon optimal control problem (OCP) with horizon length  $N_p$  is defined as

$$\begin{aligned} J(\mathbf{u}(\cdot|k), s(k)) &= \\ & \sum_{i=0}^{N_p-1} \left[ \underbrace{\|T_{\text{out}}(k+i) - T_{\text{out, ref}}(k+i)\|_{w_{T_{\text{out}}}}^2}_{\text{Torque output tracking}} \right. \\ & + \underbrace{\|\text{NO}_x(k+i)\|_{w_{\text{NO}_x}}^2}_{\text{NO}_x \text{ minimizing}} + \underbrace{\|FQ(k+i)\|_{w_{FQ}}^2}_{\text{fuel consumption minimizing}} \\ & + \underbrace{\|u(k+i|k) - u(k+i-1|k)\|_{w_{\Delta u}}^2}_{\text{control effort penalty}} \\ & \left. + \underbrace{w_s s(k)^2}_{\text{Constraint violation penalty}} \right] \end{aligned} \quad (19)$$

where

$$\|\cdot\|_w^2 = [\cdot]^T w [\cdot] \quad (20)$$

where  $s(k)$  is a slack variable that is added to the cost function to allow slight violation of the constraints. In this equation,  $w_v$ ,  $v \in [T_{\text{out}}, \text{NO}_x, FQ, \Delta u, s]$  are the MPC weights. The optimization decision,  $\mathbf{u}(\cdot|k)$ , is defined as

$$\mathbf{u}(\cdot|k) = [u(k|k)^T \ u(k+1|k)^T \ \dots \ u(k+N_p-1|k)^T]^T \quad (21)$$

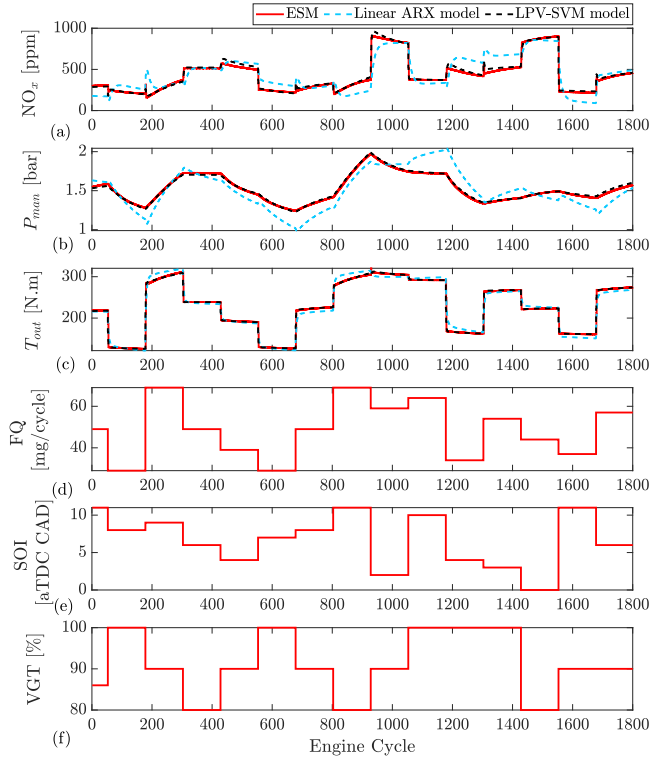


Fig. 3. Linear ARX, LPV-SVM and ESM comparison for engine-out emissions and performance using ESM co-simulation: a) engine-out  $\text{NO}_x$ , b) intake manifold pressure ( $P_{\text{man}}$ ), c) engine output torque ( $T_{\text{out}}$ ), d) Fuel quantity ( $FQ$ ), e) Start of injection ( $SOI$ ), f) Variable Geometry Turbine ( $VGT$ ) rate

Based on the defined cost function, the Optimal Control Problem (OCP) solved at each discrete-time instance, (an engine cycle) is

$$\min_{\substack{\mathbf{u}(\cdot|k), \\ s(k)}} J(\mathbf{u}(\cdot|k), s(k)) \quad (22a)$$

$$\text{s.t.} \quad x(0) = \bar{x}(0), \quad (22b)$$

$$x(k+1) = f(x(k), u(k)) \quad \forall k \in [0, N_p - 1], \quad (22c)$$

$$\underline{x} \leq x_k \leq \bar{x} \quad \forall k \in [0, N_p], \quad (22d)$$

$$\underline{u} \leq u_k \leq \bar{u} \quad \forall k \in [0, N_p - 1] \quad (22e)$$

Where  $f(x(k), u(k))$  are based on Eq. 3 which is the SVM-LPV model of the system and  $x$  are the states of model. A five step prediction horizon is used due to the dynamics of the  $\text{NO}_x$  emissions and a single step control horizon is used. For the linear MPC,  $A$  and  $B$  are constant matrices. The optimization is subject to the constraints listed in Table 1. The 500 ppm maximum  $\text{NO}_x$  is chosen as the upper limit to keep the emissions below the maximum experimentally measured  $\text{NO}_x$  output from the Cummins calibrated engine controller. However, this constraint can be adjusted depending on emission legislation. The limit on  $FQ$  is used as a safety constraint. Limits in  $SOI$  are added to avoid early combustion which can lead to combustion noise and late combustion which can lead to low thermal efficiency and high exhaust gas temperatures. The turbocharger characteristic map is used to set the  $VGT$  limit.

## 5.2 Controller Results

As the LPV model is developed at a constant speed of 1500 rpm, the ESM is also simulated at that speed. The developed MPC based on the SVM-LPV model, the linear ARX-based linear MPC and benchmark (BM) ESM calibrated ECU based on the Cummins production ECU are compared and shown in Fig. 4. Except for a slight violation in  $\text{NO}_x$  constraints (for example at engine cycle 600), both controllers are able to keep  $\text{NO}_x$  emissions below the specified constraint. For the  $\text{NO}_x$  emissions, both the linear MPC and BM controller have higher overall emissions levels than the LPV-MPC controller. This is likely due to the simple linear model used in the linear MPC that is unable to capture the non-linear  $\text{NO}_x$  formation trends.

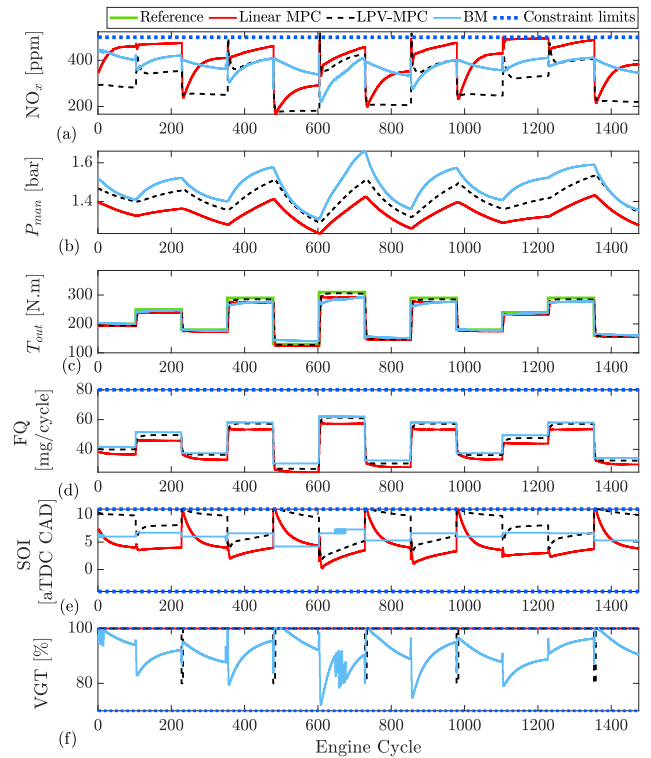


Fig. 4. Linear MPC, LPV-MPC and Benchmark comparison at 1500: a) engine-out  $\text{NO}_x$ , b) intake manifold pressure ( $P_{\text{man}}$ ), c) engine output torque ( $T_{\text{out}}$ ), d) fuel quantity ( $FQ$ ), e) Start of injection ( $SOI$ ), f) Variable Geometry Turbine ( $VGT$ ) rate

As shown, both the LPV-MPC and Linear MPC (LMPC) tend to generate late injection timings which cause a reduction in peak combustion temperature resulting in lower  $\text{NO}_x$  levels. However, this late combustion phasing can result in lower thermal efficiency and higher fuel consumption. For this reason an upper bound is added for  $SOI$ . The output torque ( $T_{\text{out}}$ ) tracking performance

Table 1. Constraint Values

Min Value ( $\underline{x}, \underline{u}$ )	Variable ( $x, u$ )	Max Value ( $\bar{x}, \bar{u}$ )
0 ppm	$\text{NO}_x$	500 ppm
10 mg/cycle	$FQ$	80 mg/cycle
-2 aTDC CAD	$SOI$	11 aTDC CAD
70 %	$VGT$	100 %

is within 5% for all three controllers. The feed-forward controller does fail to reach the target torque and remains slightly below the set-point for each step.

The LPV-SVM model contains a gain scheduling matrix  $A$  and  $B$  which are dependent on the inputs SOI and FQ. The scheduling parameters as a function of inputs for matrix  $A$  are shown in Fig. 5 which shows that the relationship between the model inputs and the scheduling parameters are non-linear for the Diesel combustion process. This non-linearity of the gain scheduling variables of the LPV-SVM model are an advantage of using the LPV model for the combustion instead of using only a few points of linearization. The gain scheduling matrix  $B$  is similarly nonlinear.

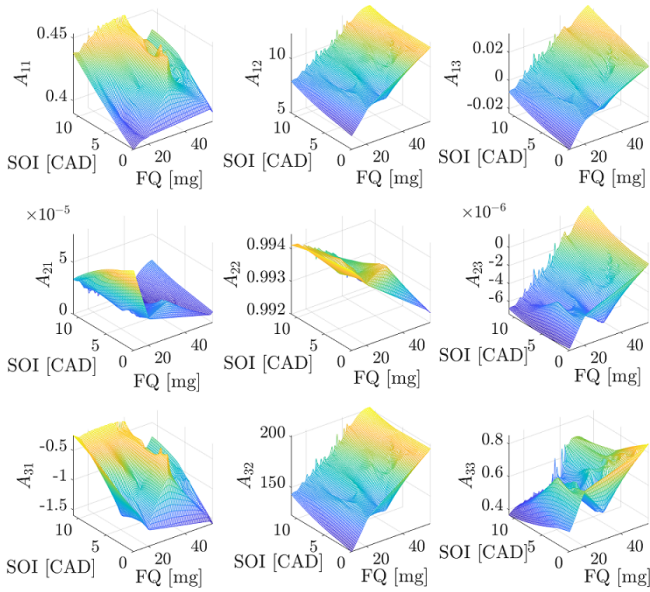


Fig. 5. “A” matrix elements for the LPV-SVM model at the engine speed of 1500 rpm

The LPV model is developed at a constant speed of 1500 rpm. To evaluate the controller robustness, each controller is tested at an engine speed of 1200 rpm. As shown in Fig. 6, both the LMPC and LPV-MPC perform significantly better than BM. Here the benchmark controller tends to advance injection timing at lower speeds which results in significant increases in  $\text{NO}_x$  emissions. Due to the increased accuracy of the LPV model, LPV-MPC performs slightly better. In the next section MPC will be replaced by imitation ML to reduce computation.

## 6. IMITATION OF MPC USING A DEEP NEURAL NETWORK

Machine learning was used to model the system in section 5. Here ML is used to replace the MPC with a learning controller called imitative LPV-MPC. The goal is to avoid the high computational time of MPC, that requires solving MPC optimization online. Instead, a function, in this case a deep network, is trained to approximate the MPC and can be deployed with a much lower computational cost.

The schematic of imitative LPV-MPC was previously shown in Section 4 of Fig. 1 (step 4). First, the LPV-MPC are implemented using the ESM in simulation. Then

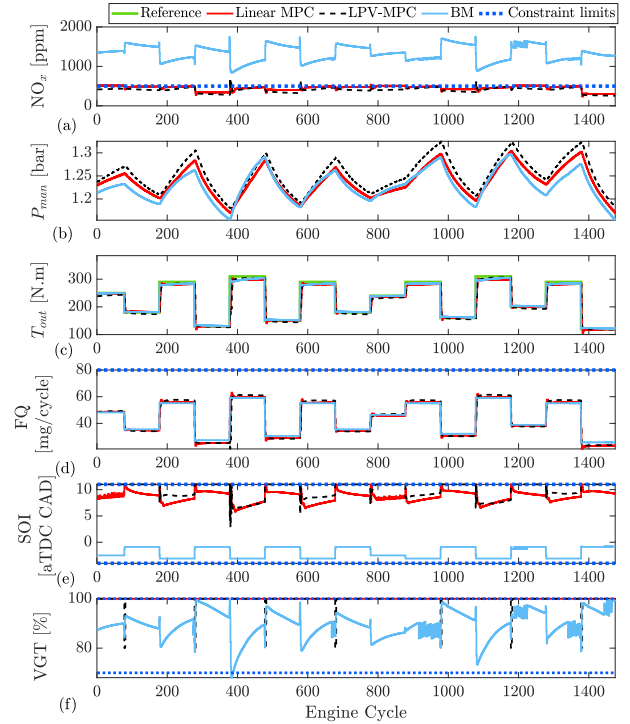


Fig. 6. Linear MPC, LPV-MPC and Benchmark comparison at 1200 rpm: a) engine-out  $\text{NO}_x$ , b) intake manifold pressure ( $P_{man}$ ), c) engine output torque ( $T_{out}$ ), d) Fuel quantity ( $FQ$ ), e) Start of injection ( $SOI$ ), f) Variable Geometry Turbine ( $VGT$ ) rate

the controller input and outputs are recorded, and a deep neural network, including a Long-Short-Term Memory (LSTM) layer, are used to mimic the behavior of the MPC as shown schematically in Fig. 7. The inputs of this network are engine output torque, the error in output torque ( $e_{T_{out}}$ ), engine-out  $\text{NO}_x$ , intake manifold pressure  $P_{man}$ , and engine speed  $n_{rpm}$  and the outputs are fuel quantity ( $FQ$ ), start of injection ( $SOI$ ), and  $VGT$ . This network includes four main layers where the first, third and fourth layer are fully connected (FC) layers with a layer size (neurons) of 32. The second layer is an LSTM layer with the same layer size. The reason for using an FC layer around the LSTM to create a deep network is to increase the complexity of the model without increasing the number of hidden and cell states of the entire network. Finally, the online MPC is replaced with the designed imitative LPV-MPC.

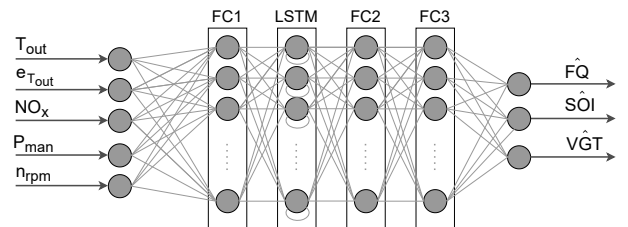


Fig. 7. Structure of proposed network for imitation of LPV-MPC

To train the imitative controller network, the LPV-MPC is evaluated for 2000 seconds at different engine speeds with randomly generated output torque targets. Of these

2000 seconds, 1600 seconds are devoted to training and the remainder to validation. To train this network, a mini-batch size of 512, initial learn rate of 0.01, and a learn rate drop period of 200 Epochs with a drop factor of 0.5 is used. In this training, an  $L_2$  Regularization with a value of 0.8 is used. The training accuracy for FQ, SOI, and VGT is 4.3%, 6.3%, and 8.3% while for the validation data an accuracy of 4.3%, 8.9%, 10.3% is observed. To test the imitative LPV-MPC, the controller is tested on a previously unseen reference and compared to the LPV-MPC in Fig. 8. The results show that the imitative controller can successfully clone the behaviour of the LPV-MPC and generate approximately optimal control without performing an online optimization.

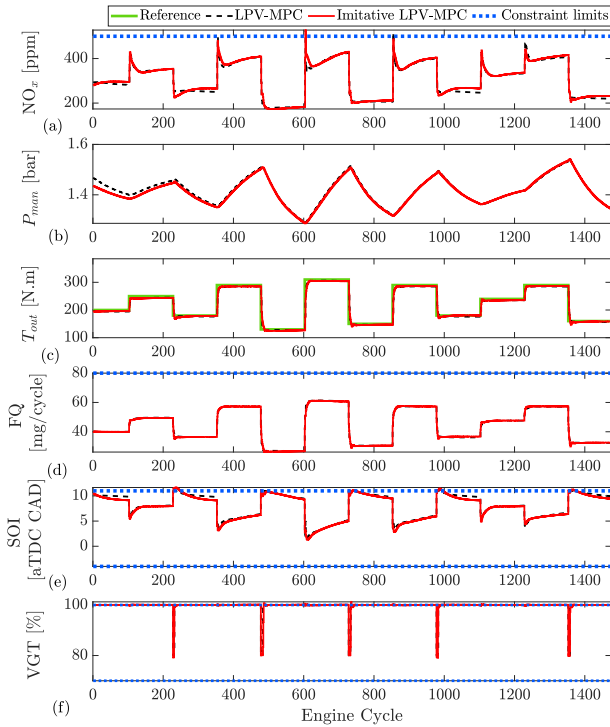


Fig. 8. LPV-MPC and imitative LPV-MPC comparison in 1500 rpm: : a) engine-out  $NO_x$ , b) intake manifold pressure ( $P_{man}$ ), c) engine output torque ( $T_{out}$ ), d) Fuel quantity ( $FQ$ ), e) Start of injection ( $SOI$ ), f) Variable Geometry Turbine ( $VGT$ ) rate

The performance of the controllers at engine speeds of 1500 and 1200 rpm are compared to the baseline model and are summarized in Table 2. As the LPV controller was designed only based on constant speed data at 1500 rpm, the LPV-MPC and imitative controller’s performance show the controllers robustness changing speeds. Here a significant  $NO_x$  emissions reduction can be seen for all the controllers over the baseline model except for the LMPC model at 1500 rpm which can be attributed to the use of a simplified linear model. In addition to the reduced emissions for all controllers, they are able to maintain or improve fuel consumption compared to the baseline. This demonstrates the advantage of the optimized controllers over the calibration based baseline. One disadvantage of the developed models is an increase in load tracking error in comparison to the baseline model at 1500 rpm. However, this 2% discrepancy in load tracking results in significant

emission reduction of 18-70% and fuel consumption reduction of 1-10%.

The imitative LPV-MPC controllers provide similar improvements to the full MPC controllers over the baseline model, while providing significantly improved computational times. As presented in Table 2, the imitative controllers are 50 and 77 times faster than online MPC optimization at 1500 and 1200 rpm, respectively. All these simulations are carried out on a computer equipped with Intel Core i7-6700K processor with 32.0 GB of RAM. This computation requirement with respect to online MPC optimization makes future real-time implementation of this controller feasible.

## 7. SUMMARY AND CONCLUSIONS

This paper presents the integration of machine learning and model predictive control for both modeling and controller implementation for a diesel engine application. First, a support vector machine based linear parameter varying model is developed to design an LPV-MPC. When comparing the results of linear and LPV models for newly generated inputs, the LPV model showed better prediction accuracy for all engine outputs. Using these models a linear MPC and an LPV-MPC are designed.

Then, the LPV-MPC is implemented and the controller input and output data are collected from the MPC and used to train a deep neural network. By replacing the full online MPC with a deep network the aim is to reduce the computational time of the MPC. After testing the imitative LPV-MPC controller at two different engine speeds, the imitative controller performs very closely to the online optimized MPC but with a significant reduction in the processing time. In addition, the MPC and imitative models showed significant improvements in  $NO_x$  emissions and a reduction in fuel consumption while providing similar load following capabilities as the feed-forward production controller. Both the LPV-MPC and imitative controller are able to reduce  $NO_x$  emissions by 18-70% while reducing fuel consumption by 1-10% compared to the Cummins production controller and the imitative controller requires 1/50 the computation time compared to online MPC optimization.

## ACKNOWLEDGMENTS

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: The research was performed as part of the Research Group (Forschungsgruppe) FOR 2401 “Optimization based Multiscale Control for Low Temperature Combustion Engines,” which is funded by the German Research Association (Deutsche Forschungsgemeinschaft, DFG) and with Natural Sciences Research Council of Canada Grant 2022-03411. Partial funding from Future Energy Systems at the University of Alberta is also gratefully acknowledged.

## REFERENCES

Basina, L.A., Irdmoua, B.K., Velni, J.M., Borhan, H., Naber, J.D., and Shahbakhti, M. (2020). Data-driven modeling and predictive control of maximum pressure

Table 2. Percentage of improvement for proposed MPC and imitative LPV-MPC with respect to the Benchmark for engine speeds of 1500 and 1200 rpm

1500 rpm				
	NO <sub>x</sub> [%]	FQ [%]	load error [%]	Average time per cycle [ms]
LMPC	+6.71	-10.00	-1.04	1.17
LPV-MPC	-18.98	-3.48	+0.99	1.69
Imitative LPV-MPV	-18.05	-3.70	+1.03	0.03
1200 rpm				
	NO <sub>x</sub> [%]	FQ [%]	load error [%]	time per cycle [ms]
LMPC	-66.20	-1.23	-0.98	1.93
LPV-MPC	-69.77	0.00	-0.88	1.61
Imitative LPV-MPV	-70.48	-1.85	-3.56	0.03

- rise rate in RCCI engines. In *IEEE Conference on Control Technology and Applications (CCTA 2020)*, 94–99. IEEE. doi:10.1109/CCTA41146.2020.9206358.
- Heid, B., Hensley, R., and Knupfer, S. (2017). What’s sparking electric-vehicle adoption in the truck industry?
- Irdmousa, B.K., Rizvi, S.Z., Velni, J.M., Naber, J., and Shahbakhti, M. (2019). Data-driven modeling and predictive control of combustion phasing for RCCI engines. *American Control Conference (ACC 2019)*, 1–6. doi:10.23919/ACC.2019.8815269.
- Liao-McPherson, D., Huang, M., Kim, S., Shimada, M., Butts, K., and Kolmanovsky, I. (2020). Model predictive emissions control of a diesel engine airpath: Design and experimental evaluation. *International Journal of Robust Nonlinear Control*, 30(17), 7446–7477. doi:10.1002/rnc.5188.
- Liu, Z., Dizqah, A.M., Herreros, J.M., Schaub, J., and Haas, O. (2021). Simultaneous control of NO<sub>x</sub>, soot and fuel economy of a diesel engine with dual-loop EGR and VNT using economic MPC. *Control Engineering Practice*, 108, 104701. doi:10.1016/j.conengprac.2020.104701.
- Norouzi, A., Aliramezani, M., and Koch, C.R. (2021a). A correlation-based model order reduction approach for a diesel engine NO<sub>x</sub> and brake mean effective pressure dynamic model using machine learning. *International Journal of Engine Research*, 22(8), 2654–2672. doi:10.1177/1468087420936949.
- Norouzi, A., Ebrahimi, K., and Koch, C.R. (2019). Integral discrete-time sliding mode control of homogeneous charge compression ignition (HCCI) engine load and combustion timing. *IFAC-PapersOnLine*, 52(5), 153–158. doi:10.1016/j.ifacol.2019.09.025.
- Norouzi, A., Gordon, D., Aliramezani, M., and Koch, C.R. (2020). Machine Learning-based Diesel Engine-Out NO<sub>x</sub> Reduction Using a plug-in PD-type Iterative Learning Control. In *2020 IEEE Conference on Control Technology and Applications (CCTA)*, 450–455. IEEE. doi:10.1109/CCTA41146.2020.9206277.
- Norouzi, A., Heidarifar, H., Shahbakhti, M., Koch, C.R., and Borhan, H. (2021b). Model predictive control of internal combustion engines: A review and future directions. *Energies*, 14(19). doi:10.3390/en14196251.
- Ortner, P. and Del Re, L. (2007). Predictive control of a diesel engine air path. *IEEE transactions on control systems technology*, 15(3), 449–456. doi:10.1109/TCST.2007.894638.
- Reşitoğlu, İ.A., Altinişik, K., and Keskin, A. (2015). The pollutant emissions from diesel-engine vehicles and exhaust aftertreatment systems. *Clean Technologies and Environmental Policy*, 17(1), 15–27.
- Rizvi, S.Z., Mohammadpour, J., Tóth, R., and Meskin, N. (2015). An IV-SVM-based approach for identification of state-space LPV models under generic noise conditions. In *2015 54th IEEE Conference on Decision and Control (CDC)*, 7380–7385. IEEE.
- Shahpouri, S., Norouzi, A., Hayduk, C., Rezaei, R., Shahbakhti, M., and Koch, C.R. (2021a). Hybrid machine learning approaches and a systematic model selection process for predicting soot emissions in compression ignition engines. *Energies*, 14(23). doi:10.3390/en14237865.
- Shahpouri, S., Norouzi, A., Hayduk, C., Rezaei, R., Shahbakhti, M., and Koch, C.R. (2021b). Soot Emission Modeling of a Compression Ignition Engine Using Machine Learning. 54(20), 826–833. doi:10.1016/j.ifacol.2021.11.274.
- Stewart, G. and Borrelli, F. (2008). A model predictive control framework for industrial turbodiesel engine control. In *2008 47th IEEE Conference on Decision and Control*, 5704–5711. IEEE. doi:10.1109/CDC.2008.4739384.
- Toub, M., Shahbakhti, M., Robinett, R.D., and Aniba, G. (2019). MPC-trained ANFIS for Control of MicroCSP Integrated into a Building HVAC System. In *2019 American Control Conference (ACC)*, 241–246. IEEE.
- Winkler, A., Frey, J., Fahrbach, T., Frison, G., Scheer, R., Diehl, M., and Andert, J. (2021). Embedded real-time nonlinear model predictive control for the thermal torque derating of an electric vehicle. *IFAC-PapersOnLine*, 54(6), 359–364. doi:https://doi.org/10.1016/j.ifacol.2021.08.570. 7th IFAC Conference on Nonlinear Model Predictive Control NMPC 2021.
- Yin, L., Turesson, G., Tunestål, P., and Johansson, R. (2020). Model predictive control of an advanced multiple cylinder engine with partially premixed combustion concept. *IEEE/ASME Transactions on Mechatronics*, 25(2), 804–814. doi:10.1109/TMECH.2020.2969853.
- Zhang, X., Bujarbaruah, M., and Borrelli, F. (2019). Safe and near-optimal policy learning for model predictive control using primal-dual neural networks. In *2019 American Control Conference (ACC)*, 354–359. IEEE.
- Zhao, D., Liu, C., Stobart, R., Deng, J., Winward, E., and Dong, G. (2014). An explicit model predictive control framework for turbocharged diesel engines. *IEEE Transactions on Industrial Electronics*, 61(7), 3540–3552. doi:10.1109/TIE.2013.2279353.