

**Realtime Machine Learning based In-Cycle Control of Homogeneous  
Charge Compression Ignition**

by

David Carl Gordon

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Mechanical Engineering  
University of Alberta

© David Carl Gordon, 2023

# Abstract

Homogeneous Charge Compression Ignition (HCCI) has the potential to significantly reduce engine out oxides of nitrogen ( $\text{NO}_x$ ) emissions, while maintaining a high fuel efficiency compared to existing lean-burn spark ignition engines. HCCI is characterized by compression induced autoignition of a lean homogeneous air-fuel mixture. The challenge with HCCI combustion is the high cyclic variation due to the lack of direct ignition control leading to cycles with high emissions. Advances in control strategies and microcontrollers mean that it is now possible to experimentally test these control strategies for HCCI combustion. In this thesis two advanced control strategies will be developed and experimentally implemented on a single cylinder research engine.

To control HCCI combustion it is necessary to combine fast actuators, in-cycle control and model based cycle-to-cycle control. Using a high speed ignition system which can add energy to the cylinder in under  $10 \mu\text{s}$ , an in-cycle controller was developed and experimentally tested. The proposed controller was able to successfully reduce the standard deviation of combustion phasing and indicated mean effective pressure (IMEP) by up to 34% and 28%, respectively. Based on the success of the in-cycle controller, it was coupled with a cycle-to-cycle controller for the experimental implementation of a multiscale controller. When compared to the single actuation of the in-cycle control strategies, the advantages of MIMO controller showed a significant improvement in the prediction and prevention of misfire cycles. This control approach provided a clear increase in indicated efficiency from 28.09 % to 29.42 %, improved IMEP from 2.88 bar up to 3.02 bar and helped to stabilize the operation point and a

reduction of standard deviation for IMEP and CA50 by more than 65 % was achieved.

Three models have been developed for determining the emissions and performance of HCCI. These models cover a wide range of modeling strategies from physics based kinetics modeling to fully machine learning based black-box models. The first model investigated was a physical kinetics model which is able to provide a breakdown of the chemical species in the cylinder. An offline kinetics model was successfully created and showed that the chemical components of the cylinder could be estimated using a relatively simple kinetics model with 34 species and 36 reactions. To improve the real-time capability of the model, a machine learning (ML) model was then investigated. The first ML model developed is a support vector machine (SVM) model with the goal of determining the effect of different ML approaches and feature set selection on the model quality for HCCI emissions prediction. A linear and a non-linear SVM model were compared to a traditional artificial neural network (ANN) model. This comparison showed for a small data set that SVM based models were more robust to changes in feature selection and better able to avoid local minimums compared the ANN leading to a more consistent model prediction. Finally a transient engine performance and emissions model for HCCI was created using a deep neural network (DNN) containing a long short term memory (LSTM) layer. This model requires significantly more data than the SVM model. However, due to the recurrent neural network the model is able to capture time dependencies in the data. This resulted in an accurate model for transient engine operation with an error less than 5 % for all four model outputs. Of the models developed and tested the LSTM model showed the greatest accuracy while preserving a simple model structure to allow for real-time implementation.

Using the developed LSTM based DNN model a nonlinear model predictive controller (NMPC) has been designed and experimentally tested. To successfully implement the NMPC, the open-source package `acados` which enables the integration of

embedded solvers for nonlinear optimal control has been used. The NMPC online optimization was tested on various processors to determine which provided the required realtime turnaround time. Of the six devices tested, all showed the possibility to meet the real-time requirements. Therefore to keep implementation costs low a Raspberry Pi 400 was chosen to test on the engine testbench. The implementation of the `acados` NMPC on the Raspberry Pi 400 has been experimentally shown to follow an IMEP reference with an RSME of 0.133 bar on the HCCI engine. The NMPC was also able to observe constraints and keep the combustion phasing close to the target value.

# Preface

This thesis is an original work by David Carl Gordon. The research presented in this thesis is the combination of engine testing results collected at the University of Alberta Internal Combustion Engine Laboratory led by Dr. Bob Koch and from data collected at the Teaching and Research Area for Mechatronics in Mobile Propulsion (MMP) at RWTH Aachen led by Prof. Jakob Andert. This work is the result of a close collaboration between the University of Alberta and RWTH Aachen University.

Each chapter is fully or partially based on published or submitted papers in peer-reviewed journals and conference proceedings [1–14].

The multiscale control work was completed in collaboration with Julian Bedei at RWTH Aachen University. The joint work presented took advantage of his novel data generation method and data he collected.

For the real-time implementation of the model predictive controller, the embedded programming was completed using the open-source software called `acados` in collaboration with Alexander Winkler and Eugen Nuss from RWTH Aachen University and with `acados` development support from Jonathan Frey at the University of Freiburg.

The development of the NMPC and integration of the LSTM was completed in collaboration with Armin Norouzi.

# Acknowledgments

First and foremost, I would like to thank my thesis supervisor Dr. Charles (Bob) Koch. Without his guidance and financial support the work completed over the past four years would not have been possible. The academic freedom he provided allowed me to pursue my own area's of interest and really develop my own connections with international research groups and local companies. I would also like to thank my co-supervisor Jakob Andert for allowing me to visit RWTH and for all the support you have provided.

I must then thank my family, as without the full support of parents Laurie & Kathy Gordon and sister Danielle I would not have been able to complete my PhD. I am also extremely grateful for the support of my Grandparents who are always available for a visit and a coffee when I needed a break.

The support and friendship of my University of Alberta colleagues must be acknowledged. Armin Norouzi provided both the technical support which enabled the completion of my thesis and having someone to chat with is greatly appreciated. I would also like to acknowledge Jakub McNally, Javad Kheyrollahi, Masoud Aliramazani, Khashayar Ebrahimi and Bradley Gibeau who have provided support throughout my PhD.

Next, I would like to acknowledge my colleagues and friends at RWTH Aachen University who have made my stays in Germany so enjoyable and have helped to ensure each stay was productive. Thank you Julian Bedei, Alexander Winkler, Patrick Schaber, Gero Blomeyer, Bastian Lehrheuer, Max Wick, and Christan Wouters. Also

a special thanks to Frederick Ratz who has put up with my crazy requests for experimental testing and for always ensuring the engine is operational.

# Table of Contents

<b>1</b>	<b>Introduction and Background</b>	<b>1</b>
1.1	Homogeneous Charge Compression Ignition (HCCI) . . . . .	3
1.1.1	Combustion Metrics . . . . .	5
1.1.2	Emissions . . . . .	11
1.1.3	Combustion Stability . . . . .	11
1.2	HCCI Control Strategies . . . . .	14
1.2.1	Actuators for control . . . . .	14
1.2.2	Traditional engine control strategies . . . . .	16
1.2.3	Advanced control strategies . . . . .	17
1.2.4	HCCI performance and emissions modeling . . . . .	20
1.2.5	Timing constraints . . . . .	24
1.3	Problem Identification and Proposed Solutions . . . . .	27
1.3.1	In-cycle controller . . . . .	28
1.3.2	Model-based controller . . . . .	28
1.4	Thesis Outline and Contributions . . . . .	30
1.4.1	Thesis outline . . . . .	30
1.4.2	Contributions . . . . .	32
<b>2</b>	<b>Experimental Setup</b>	<b>34</b>
2.1	University of Alberta SCORE . . . . .	34
2.1.1	Data acquisition and engine control systems . . . . .	38



2.1.2	Emission measurement . . . . .	38
2.2	RWTH Aachen SCRE . . . . .	40
2.2.1	Emissions measurement . . . . .	45
2.2.2	Data acquisition and engine control systems . . . . .	45
2.2.3	Rapid Ignition System . . . . .	47
2.3	Summary of chapter . . . . .	47
<b>3</b>	<b>In-Cycle HCCI Engine Control using Rapid Ignition System</b>	<b>49</b>
3.1	Pressure based control strategy . . . . .	51
3.2	Heat release based control strategy . . . . .	53
3.3	Ignition timing constraints . . . . .	54
3.4	Combustion stability . . . . .	56
3.5	Emission benefits . . . . .	60
3.6	Summary of chapter . . . . .	64
<b>4</b>	<b>Combined In-Cycle and Cycle-To-Cycle HCCI Engine Control</b>	<b>65</b>
4.1	Selection of Feedback Variables . . . . .	66
4.1.1	Main Combustion (MC) . . . . .	66
4.1.2	Intermediate Compression (IC) . . . . .	67
4.1.3	Selection of Operating Variables . . . . .	68
4.2	Design of Multiple Input Multiple Output (MIMO) Controller using Artificial Neural Network (ANN) based Inverse Models . . . . .	69
4.2.1	ANN Modeling Approach . . . . .	69
4.2.2	ANN Model Results . . . . .	72
4.3	Controller Results and Validation . . . . .	75
4.4	Summary of chapter . . . . .	83
<b>5</b>	<b>HCCI Emission and Performance Modeling</b>	<b>84</b>
5.1	White-Box Chemical Kinetics Modeling . . . . .	85

5.1.1	Mechanism . . . . .	85
5.1.2	Combustion model . . . . .	86
5.1.3	Model performance . . . . .	91
5.1.4	Simulink model . . . . .	93
5.1.5	FPGA model . . . . .	96
5.2	Machine Learning Methods: Steady State Models . . . . .	98
5.2.1	Support Vector Machine . . . . .	100
5.2.2	Artificial Neural Network (ANN) . . . . .	106
5.2.3	Feature Selection: Physical insights . . . . .	107
5.2.4	ML Emission Model Comparison . . . . .	112
5.2.5	Feature Selection . . . . .	116
5.2.6	Optimization and model training time . . . . .	118
5.2.7	Chosen model performance . . . . .	119
5.3	Deep Neural Network - Long Short Term Memory: Transient Model .	123
5.3.1	Experimental Data Generation . . . . .	128
5.3.2	LSTM Model Training . . . . .	130
5.4	Summary of chapter . . . . .	140
<b>6</b>	<b>Nonlinear Model Predictive Control Experimental Implementation for HCCI</b>	<b>142</b>
6.1	HCCI NMPC Creation . . . . .	145
6.1.1	Nonlinear State-Space Representation . . . . .	145
6.1.2	Optimal Control Problem . . . . .	147
6.2	Implementation of NMPC in Simulation . . . . .	150
6.3	Experimental NMPC Implementation . . . . .	156
6.3.1	<code>acados</code> NMPC on MABX II . . . . .	156
6.3.2	<code>acados</code> Embedded Hardware Calculation Time Comparison .	159
6.3.3	Realtime testing of <code>acados</code> on Raspberry Pi 400 . . . . .	161

6.3.4	LSTM based NMPC Development Time . . . . .	165
6.4	Summary of chapter . . . . .	170
<b>7</b>	<b>Conclusions</b>	<b>172</b>
7.1	In-cycle HCCI Control using Rapid Spark . . . . .	173
7.2	Multiscale HCCI Control . . . . .	174
7.3	HCCI Performance and Emission Modeling . . . . .	175
7.4	Nonlinear Model Predictive Control Experimental Implementation . .	177
7.5	Future Work . . . . .	178
	<b>Appendix A: Ph.D. Publications</b>	<b>197</b>
A.1	Peer Reviewed Journal Papers . . . . .	197
A.2	Peer Reviewed Conference Papers in Proceedings . . . . .	200
A.3	Technical Posters . . . . .	203

# List of Tables

1.1	HCCI Control Actuators . . . . .	15
2.1	University of Alberta - Single cylinder research engine parameters . .	36
2.2	Rapid prototyping ECU Specifications . . . . .	39
2.3	RWTH Aachen University - Single cylinder research engine parameters	42
2.4	Accuracy of emissions measurement system . . . . .	45
3.1	Combustion stability improvement due to feedforward rapid spark con- troller. . . . .	60
4.1	Performance of ANN based cycle-to-cycle and in-cycle inverse process models . . . . .	74
4.2	Controller results split into approximately 200 cycles intervals . . . .	76
5.1	Range of HCCI engine input parameters . . . . .	99
5.2	Features $u_1 - u_{26}$ for the five different feature sets . . . . .	111
5.3	Criteria for method selection . . . . .	116
5.4	Criteria for feature selection . . . . .	118
5.5	Selected training options for proposed deep network to predict HCCI performance and emissions . . . . .	131
5.6	RMSE and normalized RMSE of DNN model vs Experiment . . . . .	133
6.1	Constraint Values . . . . .	150
6.2	Processor NMPC execution time for HCCI MiL simulation . . . . .	160

6.3 Two-way communication from dSPACE MABX II to Raspberry Pi 400  
and back for various communication protocols . . . . . 161

# List of Figures

1.1	Normalized heat release showing $CA_{10-90}$ and $CA_{50}$ . . . . .	9
1.2	Effect of varying NVO duration on cylinder pressure from [15]. . . . .	10
1.3	Distinct cyclic variation in the in-cylinder pressure . . . . .	12
1.4	Distinct cyclic variation in combustion phasing $CA_{50}$ return map . . .	13
1.5	HCCI control variables and available calculation time . . . . .	25
1.6	Calculation time for model based controllers . . . . .	26
1.7	Schematic of the thesis organization . . . . .	31
2.1	University of Alberta SCRE schematic . . . . .	35
2.2	Hinged electromagnetic valve schematic for University of Alberta SCRE	37
2.3	Electromagnetic valve schematic . . . . .	41
2.4	RWTH Aachen SCRE schematic . . . . .	43
3.1	Cyclic variability in the cylinder pressure during HCCI combustion .	51
3.2	Correlation between upcoming combustion phasing and cylinder pressure	52
3.3	Correlation between upcoming combustion phasing and cumulative heat release . . . . .	53
3.4	Selected threshold of 0J heat release for start of combustion using cor- relation between upcoming combustion phasing and cumulative heat release . . . . .	54
3.5	Feedforward spark control structure . . . . .	55
3.6	Ignition discharge time based on control signal at $5^\circ$ CA aTDC . . .	56

3.7	Combustion phasing, $CA_{50}$ , stability improvement with in-cycle controller based on heat release at $5^\circ$ CA aTDC. The controller is activated after 500 cycles. Here, $n = 1500$ 1/min, NVO= $133^\circ$ CA, IMEP = 4.0 bar. . . . .	57
3.8	IMEP stability improvement with in-cycle spark controller based on heat release at $5^\circ$ CA aTDC. . . . .	58
3.9	Effect of tested spark controller based on heat release on the combustion phasing return map . . . . .	59
3.10	Comparison of nitrogen oxide emissions between control strategies. Error bars represent two standard deviations. . . . .	61
3.11	Comparison of unburnt hydrocarbon emissions between control strategies. Error bars represent two standard deviations. . . . .	62
3.12	Comparison of carbon monoxide emissions between control strategies. Error bars represent two standard deviations. . . . .	63
4.1	Multi-scale control strategy for HCCI combustion . . . . .	66
4.2	Inverse model structure to describe the HCCI cycle-to-cycle and in-cycle (MC) scale according to the split autoregressive process . . . . .	71
4.3	Modeling results: Probability density function of model error for cycle-to-cycle and in-cycle scale . . . . .	73
4.4	$CA_{50}$ , IMEP, heat release in IC $Q_{IC}$ and operating variables $m_{W, IC}$ , $m_{W, MC}$ and $t_{F, MC}$ for different controller concepts . . . . .	77
4.5	$CA_{50}$ return map for different controller concepts compared to open-loop operation . . . . .	78
4.6	Structure of model free MS-SISO control approach and model free C2C-SISO control approach . . . . .	79
4.7	Standard deviation and mean value of $CA_{50}$ and IMEP of different control approaches in comparison to the uncontrolled case . . . . .	80

5.1	Schematic display of the combustion kinetics model highlighting the three separate components. . . . .	87
5.2	Comparison between calculated and measured (average over 250 cycles) pressure for a post processed single zone model applying the Tsushima mechanism in MATLAB and Cantera . . . . .	91
5.3	Gas composition calculated by a single zone model using the Tsushima mechanism, complete conversion and over stoichiometric conditions. . . . .	92
5.4	Measured pressure and iHC mass fraction versus CAD for the post processed model . . . . .	94
5.5	Relative error between the MATLAB and the Simulink pressure trace for different resolutions of $T$ in the $k_f$ lookup table. . . . .	95
5.6	Temperature calculated with an external pressure trace at the minimal FPGA step size $\Delta t_{\text{FPGA}} = 3.125\text{e-}7$ s. . . . .	97
5.7	Schematic of data collection and proposed emission modeling using PSO-based LSVM and NLSVM . . . . .	100
5.8	SVM regression and support vectors (based on [117]) . . . . .	101
5.9	Comparison between $R^2$ and Normalized RMSE values for CO for NLSVM and LSVM with benchmark ANN method . . . . .	113
5.10	Comparison between $R^2$ and NRMSE values for HC for NLSVM and LSVM with benchmark ANN . . . . .	115
5.11	Comparison between $R^2$ and NRMSE values for $\text{NO}_x$ for NLSVM and LSVM with benchmark ANN . . . . .	115
5.12	Comparison between $R^2$ and NRMSE values for $\text{CO}_2$ for NLSVM and LSVM with benchmark ANN . . . . .	116
5.13	Optimization and evaluation time comparison between LSVM, NSVM, and ANN on an Intel i7-8700K PC . . . . .	119



5.14	Actual vs experiment - HCCI emission model. . . . .	122
5.15	Long-Short-Term Memory (LSTM) cell structure schematics . . . . .	123
5.16	Structure of proposed deep neural network model for engine perfor- mance and emission modeling. . . . .	125
5.17	Computational graph of proposed deep network . . . . .	127
5.18	PRBS data generation on single cylinder research engine for DNN model training . . . . .	134
5.19	Distribution of engine inputs using PRBS data generation . . . . .	135
5.20	Distribution of consecutive cycles where all inputs are constant during PRBS . . . . .	136
5.21	Distribution of engine outputs resulting from PRBS data generation .	137
5.22	Loss versus epochs for the proposed deep neural network model . . .	138
5.23	Validation for LSTM-based DNN model vs. experimental data . . . .	139
6.1	Development process for creation of online NMPC for HCCI engine control . . . . .	144
6.2	<code>acados</code> offline NMPC simulation showing controller outputs and pre- dicted engine response . . . . .	154
6.3	<code>acados</code> offline NMPC simulation showing NMPC solution results: a) Calculation time of NMPC b) SQP iterations required for NMPC con- vergence c) Cost resulting from eqn 6.15 . . . . .	155
6.4	Block diagram of LSTM-NMPC structure running on MABX II . . . .	157
6.5	Realtime cycle to next cycle NMPC experimental results: Showing IMEP reference tracking performance with one cycle delay. . . . .	158
6.6	Block diagram of the split LSTM-NMPC structure running on a Rasp- berry PI 400 and MABX II . . . . .	162
6.7	Experimental cycle to cycle NMPC implementation on Raspberry Pi 400: Showing IMEP reference tracking performance . . . . .	166

6.8	Model prediction from LSTM model during experimental implementation on Raspberry Pi 400. a) CA50 b) IMEP . . . . .	167
6.9	LSTM cell and hidden states during experimental implementation on Raspberry Pi 400 . . . . .	168
6.10	acados NMPC performance results on Raspberry Pi 400 . . . . .	169

# Abbreviations & Acronyms

<b>AFR</b>	Air-Fuel Ratio.
<b>AI</b>	Artificial Intelligence.
<b>ANN</b>	Artificial Neural Network.
<b>ARX</b>	AutoRegressive EXogenous.
<b>ATAC</b>	Active Thermo-Atmosphere Combustion.
<b>aTDC</b>	after Top Dead Center.
<b>BB</b>	Black-Box.
<b>BM</b>	Benchmark.
<b>BNN</b>	Bayesian Neural Network.
<b>bTDC</b>	before Top Dead Center.
<b>CA</b>	Crank Angle.
<b>CA50</b>	Crank angle where fifty percent of heat energy has been released [CAD].
<b>CAD</b>	Crank Angle Degree.
<b>CAN</b>	Controller Area Network.
<b>CAS</b>	Combustion Analysis System.

<b>CFD</b>	Computational Fluid Dynamics.
<b>CFS</b>	Criterion for Feature Selection [-].
<b>CI</b>	Compression Ignition.
<b>CMS</b>	Criterion for Methods Selection [-].
<b>CO</b>	Carbon Monoxide.
<b>CO2</b>	Carbon Dioxide [%].
<b>COM</b>	Control-Oriented Model.
<b>DDPG</b>	Deep Deterministic Policy Gradient.
<b>Deep RL</b>	Deep Reinforcement Learning.
<b>DNN</b>	Deep Neural Networks.
<b>DOI</b>	Duration of Injection [ms].
<b>DPG</b>	Deterministic Policy Gradient.
<b>ECU</b>	Engine Control Unit.
<b>EGR</b>	Exhaust Gas Recirculation.
<b>ELM</b>	Extreme Learning Machine.
<b>EMVT</b>	Electro-magnetic Valve Train.
<b>EVC</b>	Exhaust Valve Closing [CAD].
<b>EVO</b>	Exhaust Valve Opening [CAD].
<b>FC</b>	Fully Connected Layer.

<b>FPGA</b>	Field Programmable Gate Array.
<b>FS</b>	Feature Set.
<b>FTIR</b>	Fourier-Transform Infrared Spectroscopy.
<b>GA</b>	Genetic Algorithm.
<b>GPR</b>	Gaussian Process Regression.
<b>HC</b>	Hydrocarbon.
<b>HCCI</b>	Homogeneous Charge Compression Ignition.
<b>IC</b>	Intermediate Combustion.
<b>ICE</b>	Internal Combustion Engine.
<b>ILC</b>	Iterative Learning Control.
<b>IMEP</b>	Indicated Mean Effective Pressure [bar].
<b>IP</b>	Internet Protocol.
<b>IVC</b>	Intake Valve Closing [CAD].
<b>IVO</b>	Intake Valve Opening [CAD].
<b>KTT</b>	Karush-Kuhn-Tucker.
<b>LLNL</b>	Lawrence Livermore National Laboratory.
<b>LPV</b>	Linear Parameter Varying.
<b>LQR</b>	Linear Quadratic Gaussian.
<b>LSTM</b>	Long-Short Term Memory.

<b>LSVM</b>	Linear Support Vector Machine.
<b>LTC</b>	Low Temperature Combustion.
<b>MC</b>	Main Combustion.
<b>MiL</b>	Model in the Loop.
<b>MIMO</b>	Multiple Input Multiple Output.
<b>ML</b>	Machine Learning.
<b>MPC</b>	Model Predictive Control.
<b>MPRR</b>	Maximum Pressure Rise Rate [bar/CAD].
<b>MSE</b>	Mean Square Error.
<b>NARX</b>	Nonlinear AutoRegressive EXogenous.
<b>NLSVM</b>	Nonlinear Support Vector Machine.
<b>NMPC</b>	Nonlinear Model Predictive Control.
<b>NO<sub>x</sub></b>	Nitrogen Oxide [ppm].
<b>NRMSE</b>	Normalized Root Means Square Error [%].
<b>NVO</b>	Symmetric Negative Valve Overlap [CAD].
<b>PCC</b>	Person Correlation Coefficient.
<b>PDF</b>	Probability Density Function.
<b>PID</b>	Proportional Integral Derivative.
<b>PRBS</b>	Pseudo-random Binary Sequence.

<b>PRF</b>	Primary Reference Fuel.
<b>PSO</b>	Particle Swarm Optimization.
<b>QP</b>	Quadratic Programming.
<b>RBF</b>	Radial Basis Function.
<b>RDE</b>	Real Driving Emissions.
<b>RIS</b>	Rapid Ignition System.
<b>RMSE</b>	Root Mean Square of Error [-].
<b>RNN</b>	Recurrent Neural Network.
<b>RON</b>	Research Octane Number.
<b>RPM</b>	Revolutions Per Minute [1/min].
<b>SACI</b>	Spark Assist Compression Ignition.
<b>SCRE</b>	Single Cylinder Research Engine.
<b>SI</b>	Spark Ignition.
<b>SISO</b>	Single Input Single Output.
<b>SMC</b>	Sliding Mode Controller.
<b>SOI</b>	Start of Fuel Injection.
<b>SQP</b>	Sequential Quadratic Programming.
<b>SVM</b>	Support Vector Machine.
<b>SVR</b>	Support Vector Regression.

<b>TDC</b>	Top Dead Center.
<b>UDP</b>	User Datagram Protocol.
<b>UHC</b>	Unburnt Hydrocarbons [ppm].
<b>VHDL</b>	Very high speed integrated circuit Hardware Description Language.
<b>VVT</b>	Variable Valve Timing.



# Chapter 1

## Introduction and Background <sup>1</sup>

Global energy consumption is increasing due to rapid population growth and economic development caused by the industrial revolution [17, 18]. According to the International Energy Agency, Carbon Dioxide (CO<sub>2</sub>) emissions from the transportation sector grew by 92% from 1990 to 2020 and global CO<sub>2</sub> emission levels are expected to rise by 29% by the end of 2035 [19]. Additionally, the International Energy Agency predicts that between 2020 and 2035 8.6 billion metric tons will be emitted [19]. In Canada from 2000 to 2017, the CO<sub>2</sub> emission level has increased by 19.2% with 44% of that increase caused by the transportation sector [20]. Natural Resources Canada reports that of the 69% of primary energy converted to secondary energy, 21% is devoted to the transportation sector. Of this amount, gasoline, diesel fuel, and aviation fuels contribute the most: 58%, 28%, and 10% respectively [20].

This has caused governments around the world to introduce more stringent environmental legislation for all sectors including transportation. Automobile manufacturers are responding by developing new propulsion methods in an attempt to meet these targets. Currently, the G-20 countries account for 90% of the global vehicle sales and 17 of these countries follow the European emissions legislation which has been updated to Euro 6 in 2015 [21]. The Euro 6 environmental legislation further reduces nitrogen oxide (NO<sub>x</sub>) emissions from 0.18 to 0.08 g/km over the previous

---

<sup>1</sup> This chapter is partially based on [1, 15, 16]

Euro 5 regulations [22]. These new regulations are challenging to meet using current engines and require expensive exhaust gas aftertreatment systems such as selective catalytic reduction, lean NO<sub>x</sub> traps and others [21, 23]. Additionally, the recent implementation of the Euro 6d emissions standards implemented in January 2021 have not changed the limits defined in the Euro 6 standard but rather added the need to comply with real driving emission (RDE) testing. Ambient conditions, traffic, and driver behavior can significantly affect RDE test results. The challenge with the RDE legislation is that engines must operate cleanly under all conditions. Engine design and calibration are thus much more challenging [24]. Complying with RDE legislation is such a large shift from previous testbench certification testing where current engine calibration methods are becoming unworkable. To solve this problem intelligent engine emission control strategies are now being deployed. These strategies take advantage of Artificial Intelligence (AI) and optimal control strategies to integrate the advanced after-treatment systems which are needed to meet the new regulations. Some AI approaches give the system the ability to modify engine calibration during real (on road) testing by adapting the controller based on the actual measured engine operating condition. This is useful to meet RDE requirements.

One way to reduce emissions is to lower the engine out emissions and boost the engine efficiency. Low temperature combustion methods to improve the engine out emission levels while also improving engine fuel efficiency is being actively investigated [25]. Homogeneous Charge Compression Ignition (HCCI) is a combustion method which utilizes a premixed homogeneous charge of fuel and air which is auto-ignited from the rising temperature and pressure during compression [26]. HCCI has shown promising engine out NO<sub>x</sub> emission reduction, however, the lack of a direct timing control is a major control and modeling challenge [27]. The speed of the combustion process also restricts the type of control strategies which can be tested due to time constraints of current microprocessors.

HCCI was first introduced in 1979 under the name Active Thermo-Atmosphere Combustion (ATAC) [28]. This combustion method has an emission reduction potential which is desired due to new more stringent environmental regulations. HCCI is a combustion method that combines characteristics of Spark Ignition (SI) and Compression Ignition (CI) engines. Similar to SI engines, the air and fuel mixture is homogeneous at the the point of ignition. However, instead of utilizing an electrical spark to initiate combustion, an HCCI engine relies on the compression of the fuel and air mixture to increase the cylinder temperature and pressure above the auto-ignition point of the fuel. As the mixture is homogeneous, the combustion process begins at many points in the combustion chamber leading to a heat release that is much faster than both SI and CI engines [29]. This rapid heat release and the high compression ratios used allow for combustion that is closer to the ideal Otto cycle and therefore offer thermodynamic efficiency benefits [30]. As HCCI combustion offers many advantages including engine out emission reductions, fuel flexibility, and a thermal efficiency increase, there are also drawbacks including ignition timing challenges, increased unburnt hydrocarbon and carbon monoxide emissions, as well as a limited operating range [26].

The current state of the art research that has been performed in the area of HCCI engine control with a focus on real-time engine control strategies is reviewed in this chapter. This will help to provide the motivation for the work presented in this thesis and allow for the main contributions to be defined. Finally, the organization and structure of the thesis will be presented.

## **1.1 Homogeneous Charge Compression Ignition (HCCI)**

Homogeneous Charge Compression Ignition (HCCI) has shown promising engine out emissions reductions, however, the lack of a direct timing control is a major control

and modeling challenge [3, 27, 31]. Furthermore, increased Hydrocarbon (HC) and Carbon Monoxide (CO) emissions have also been observed [9]. The main combustion mechanism for HCCI is compression induced autoignition of a premixed charge, leading to a high dependency on the in-cylinder gas mixture properties.

HCCI is a part-load combustion method, which is characterized by lean Low Temperature Combustion (LTC). HCCI has the potential to significantly reduce  $\text{NO}_x$  emissions by as much as 99% when compared with current stratified lean-burn gasoline combustion [32–34]. With this reduction in engine-out emissions, expensive exhaust after-treatment systems can be reduced or simplified [35–37]. HCCI utilizes rapid multi-site combustion, this in combination with reduced wall heat losses from LTC leads to thermodynamic and fuel efficiency benefits of up to 30% over current gasoline engines [38]. HCCI combustion allows for the use of any fuel that can be vaporized in air and auto-ignited using the cylinder compression. This opens the possibility to use non-standard fuels instead of gasoline or diesel fuel. Therefore, various standard biofuels including methanol and ethanol [39, 40] along with some newer biofuels including rapeseed methyl ester [41], isopentanol [42], dimethyl carbonate [43], butanol isomers [44] and others can be effectively burned in an HCCI engine. There is also the possibility to burn fuels containing water such as wet-ethanol, which is ethanol that has not been completely distilled to save processing energy [45].

HCCI combustion is defined by compression induced autoignition and is therefore highly dependent on the in-cylinder temperature, pressure and fuel mixture after compression. The lack of a direct combustion timing control method like spark timing in SI engines or injection timing in traditional CI engines is a major challenge of this combustion method. Ignition timing is strongly dependent on temperature and can be altered through intake air heating, increased compression ratio or Exhaust Gas Recirculation (EGR) [29, 46–48]. When using internal EGR as a means to provide the thermal energy required to achieve autoignition, a strong coupling between cycles

can exist when the engine is operated near the misfire limit [49, 50]. Current cycle-based control strategies only stabilize a small section of the operation range, which is insufficient for practical engine deployment [51]. Therefore, a stabilizing controller which works over a wide range of operating conditions is needed to reduce the cyclic variations in combustion phasing and load, and prevent a spontaneous shift from stable to unstable operation. Previous work in this area has been investigated by [27, 35, 52–55].

This section will focus on introducing the basics of HCCI combustion and some of the challenges of controlling the process.

### 1.1.1 Combustion Metrics

There are various combustion metrics that are useful in describing not only HCCI combustion but all internal combustion engine combustion processes. These metrics will be used throughout this thesis and will be defined below.

The indicated mean effective pressure (IMEP) is used to describe the average pressure in the combustion cylinder during a cycle and is used to describe the load output of the engine. The variation in the IMEP,  $\sigma_{\text{IMEP}}$ , is used to describe the stability of the combustion where a low variation in IMEP is a smooth running engine that is producing the same power output every cycle. The IMEP can be calculated using [56]:

$$\text{IMEP} = \frac{\text{work output per cylinder per mechanical cycle}}{\text{swept volume per cylinder}} = \frac{1}{V_d} \int p_{\text{cyl}} dV \quad (1.1)$$

Where  $V_d$  is the swept cylinder volume and  $dV$  is the change in cylinder volume. The indicated work can be calculated using the enclosed area of a pressure vs volume ( $p$ - $V$ ) trace, which can easily be calculated from the measured cylinder pressure and engine position. IMEP is normalized to the number of cylinders or cylinder volume making it a good comparison between different engines, however, it does not consider the frictional losses of the engine. Then by defining the limits of integration around

both the main combustion from intake valve closing (IVC) until exhaust valve opening (EVO) and for the negative valve overlap (NVO) period from exhaust valve closing (EVC) till intake valve opening (IVO) the IMEP of each phase of the cycle can be determined using:

$$\text{IMEP}_{\text{main}} = \frac{1}{V_d} \int_{\text{IVC}}^{\text{EVO}} p_{\text{cyl}} dV \quad (1.2)$$

$$\text{IMEP}_{\text{NVO}} = \frac{1}{V_d} \int_{\text{EVC}}^{\text{IVO}} p_{\text{cyl}} dV \quad (1.3)$$

The pressure rise rate,  $dp/d\varphi$  [bar/deg], of an engine gives a description of how quickly the cylinder pressure is changing as a function of crank angle. At high pressure rise rates engine damage can occur and prolonged operation at high pressure rise rates can lead to engine failure [57]. Additionally, at elevated pressure rise rates the acoustic combustion noise is undesirable for vehicle operator comfort. However, a higher the pressure rise rate more closely resembles the ideal Otto cycle where the quick combustion leads to efficiency benefits [56]. The pressure rise rate changes throughout the combustion event, however, its maximum value is used to represent the worst condition for that cycle and is given by:

$$dp/d\varphi = \max \left| \frac{P_{\text{cyl}}(i) - P_{\text{cyl}}(i-1)}{\varphi(i) - \varphi(i-1)} \right| \quad (1.4)$$

Where  $P_{\text{cyl}}(i)$  and  $P_{\text{cyl}}(i-1)$  are the current and previous cylinder pressure, respectively.

The air-to-fuel ratio (AFR) is an important combustion metric that is used describe the mass of air,  $m_a$ , that is brought into the cylinder in relation to the mass of injected fuel,  $m_f$ , and is given by:

$$\text{AFR} = \frac{m_a}{m_f} \quad (1.5)$$

For a given fuel chemistry it is possible to calculate the exact amount of fuel required for a given amount of air for complete combustion, this is called the stoichio-

metric AFR. The ratio of  $AFR_{act}$ , to the stoichiometric AFR,  $AFR_{stoic}$ , is defined as the equivalence air-fuel ratio  $\phi$  and lambda  $\lambda$ , which are given by:

$$\lambda = \frac{1}{\phi} = \frac{AFR_{act}}{AFR_{stoic}} \quad (1.6)$$

A value of lambda less than 1 which is rich operation where there is excess fuel for the amount of oxygen while a value of lambda greater than 1 is lean engine operation where more oxygen is present for the given fuel amount. For HCCI combustion the engine is operated in the lean operation conditions which is required to allow significant cylinder charge dilution to help slow the extremely rapid combustion. Additionally, this lean operation also provides fuel reduction benefits when compared to traditional stoichiometric SI.

The heat release from the combustion of the fuel is calculated from the measured incylinder pressure. Then using the heat realease it is possible to calculate the start, duration and end of combustion as well as the combustion phasing  $CA_{50}$ . The general equation for the pressure analysis is given by [56]:

$$\frac{dQ_b}{d\varphi} = -mc_v \frac{dT}{d\varphi} + \frac{dQ_w}{d\varphi} - p \frac{dV}{d\varphi} + \frac{dQ_{f,e}}{d\varphi} + \frac{dQ_{H_2O,e}}{d\varphi} + \sum_i (h_i - u_i) \frac{dm_i}{d\varphi} \quad (1.7)$$

To determine the heat release, both the combustion and NVO recompression phase are considered where the transferred masses into the cylinder is zero. By neglecting the blow-by mass flow[58] and assuming the mixture preparation takes place before start of combustion, simplifies eqn 1.7 to become:

$$\frac{dQ_b}{d\varphi} = -mc_v \frac{dT}{d\varphi} + \frac{dQ_w}{d\varphi} - p \frac{dV}{d\varphi} \quad (1.8)$$

By applying the ideal gas simplifications to the equation of state and differentiating by  $\varphi$  results in:

$$p \frac{dV}{d\varphi} + V \frac{dp}{d\varphi} = mR \frac{dT}{d\varphi} + mT \frac{dR}{d\varphi} + RT \frac{dm}{d\varphi} \quad (1.9)$$

Substituting eqn 1.9 into eqn 1.8 then yields the temperature gradient:

$$\frac{dT}{d\varphi} = \frac{1}{mR} \left\{ p \frac{dV}{d\varphi} + V \frac{dp}{d\varphi} - mT \frac{dR}{d\varphi} - RT \frac{dm}{d\varphi} \right\} \quad (1.10)$$

By applying the previously made assumption that the mass gradient is zero, the equation 1.10 simplifies to the characteristic equation for the pressure trace analysis:

$$\frac{dQ_b}{d\varphi} = -\frac{c_v}{R} \left( V \frac{dp}{d\varphi} - mT \frac{dR}{d\varphi} \right) - \left( 1 + \frac{c_v}{R} \right) p \frac{dV}{d\varphi} + \frac{dQ_w}{d\varphi} \quad (1.11)$$

By assuming a constant gas composition  $dR/d\varphi = 0$  and using the gas relations  $R = c_p - c_v$  and  $\gamma = c_p/c_v$ , the rate of heat release can be written as:

$$\frac{dQ_b}{d\varphi} = -\frac{1}{\gamma - 1} V \frac{dp}{d\varphi} - \frac{\gamma}{\gamma - 1} p \frac{dV}{d\varphi} + \frac{dQ_w}{d\varphi} \quad (1.12)$$

Finally, by integration of the rate of heat release rate over the crank angle duration of interest (from start to end of combustion) yields:

$$Q_f(\varphi) = \int_{\varphi_{SOC}}^{\varphi_{EOC}} dQ_f(\varphi) \quad (1.13)$$

To describe the timing and duration of combustion the crank angle for 50% net heat release,  $CA_{50}$ .  $CA_{50}$  is defined as the crank angle when 50% of the integrated total heat release is reached as shown in Figure 1.1.  $CA_{50}$  describes the timing of the combustion process and variations in  $CA_{50}$  influence the combustion efficiency and temperature which in turn significantly changes engine out emissions. The  $CA_{50}$  is often a controlled variable for HCCI engine control strategies as a specific  $CA_{50}$ , which varies between 5-8 CAD aTDC, is desired for maximum power and efficiency [35]. Similar to IMEP, the variation in  $CA_{50}$  is useful in describing the cyclic variability of the HCCI combustion process. Not only is the combustion timing important but the length of the combustion event is also valuable and is given by the burn duration as the length in crank angle it takes for the combustion to proceed from 10 to 90% of the complete heat release,  $CA_{10-90}$ . For this work it is defined as the time in crank angle degrees for the combustion to consume between 10% and 90% fuel mass,  $CA_{10-90}$  as depicted in Figure 1.1.



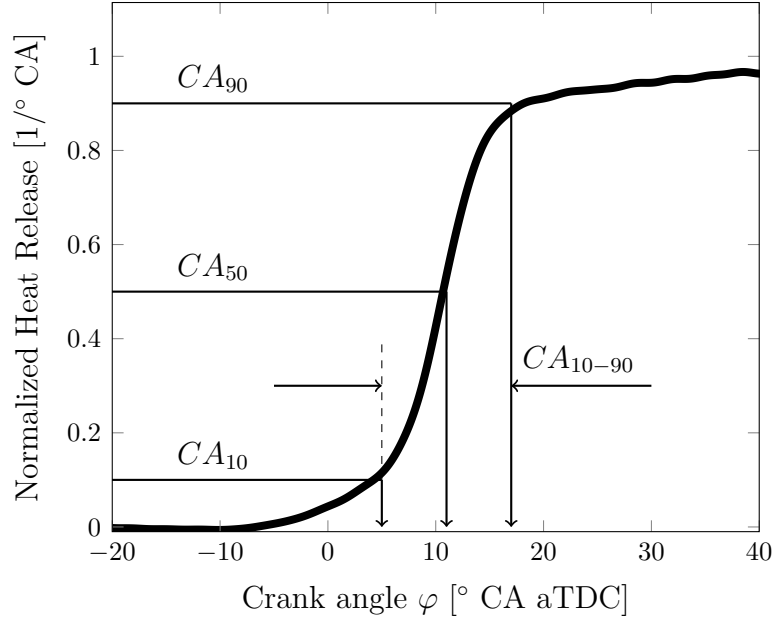


Figure 1.1: Normalized heat release showing  $CA_{10-90}$  and  $CA_{50}$  (adapted from [15])

### Exhaust Gas Recirculation

To provide enough thermal energy for the HCCI auto-ignition process to begin it is often necessary to retain some exhaust gas to increase the temperature of the trapped cylinder charge. External or internal exhaust gas recirculation (EGR) are often used [59]. External EGR involves routing the hot exhaust from the exhaust manifold back into the intake manifold. Cooling of the exhaust gas occurs and the ability to change the amount of EGR on a cyclic basis when using an external EGR system is not possible [60].

When the exhaust gas remains in the combustion chamber or only leaves the cylinder briefly into either the intake or exhaust manifolds before being returned into the combustion chamber is referred to as internal EGR [61]. When the exhaust is pushed into the intake / exhaust port and then drawn back into the cylinder this is referred to as intake or exhaust EGR [61]. This method allows changes in EGR on a cyclic basis, however, it requires that one (or both) of the valves remain slightly open as the piston passes top dead center (TDC). To accommodate this valve strategy it is necessary to

prevent a collision with the valves so often deep valve pockets are machined into the piston. This limits the piston shape to designs which provide inefficient in-cylinder mixing. This type of internal EGR also leads to increased thermal losses as the hot exhaust gas is drawn through the valves twice. There is also increased pumping losses as the air must be pushed out of the cylinder and drawn back in past the valves which act as a flow restriction.

Combustion chamber EGR is where the exhaust valve closes before the piston reaches TDC and the intake valve opens late (after gas exchange TDC) called negative valve overlap (NVO). This keeps the exhaust gas in the cylinder leading to reduced pumping losses compared to the other EGR strategies. To minimize the rebreathing of exhaust gas the valve timings are varied symmetrically around TDC. By increasing the NVO duration the amount of exhaust gas that is trapped in the cylinder is increased as shown in Figure 1.2.

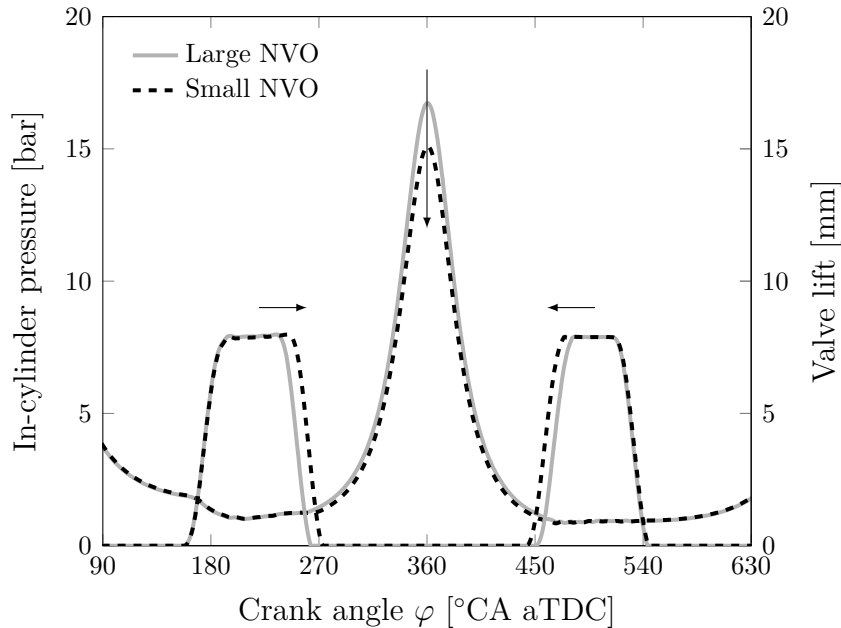


Figure 1.2: Effect of varying NVO duration on cylinder pressure from [15].

### 1.1.2 Emissions

HCCI offers engine out emission benefits compared to SI and CI combustion engines. The rapid combustion or short burn duration of HCCI helps keep the cylinder temperature low during the combustion process which helps to avoid the  $\text{NO}_x$  formation temperature [56]. As HCCI combustion is a lean homogeneous combustion there are no fuel rich areas which greatly reduces the production of soot emissions. The lean burn and low temperature combustion process does increase the amount of unburned hydrocarbon (uHC) and carbon monoxide (CO) emissions, especially at low engine loads [62]. The increased uHC are due to the low combustion temperature and flame extinction at the cold cylinder walls [63]. The oxidation of CO to  $\text{CO}_2$  is a relatively slow process and the reaction rate further decreases with low temperatures, which are found in HCCI combustion [63]. Both uHC and CO emissions, can be easily handled using an aftertreatment oxidation catalyst assuming the exhaust temperature is sufficiently high [56, 62].

### 1.1.3 Combustion Stability

Unlike SI or CI engines where ignition is controlled directly through an electrical spark or fuel injection, for HCCI the conditions in the cylinder and chemical kinetics of the fuel dictate when combustion begins. No direct method to control combustion timing leads to a limited operating range. At low loads misfire limits HCCI operation, while high pressure rise rates and peak cylinder pressures restrict high loads [29, 51].

The lack of direct ignition control can also lead to large cyclic variations with correspondingly poor combustion stability [52]. Experimentally measured in-cylinder pressure signals of three consecutive cycles are shown in Figure 1.3 as a function of engine crank angle. Gas exchange TDC occurs at  $360^\circ$  crank angle (CA) and the increase in in-cylinder pressure is due to the internal NVO. While combustion TDC is at  $720^\circ$  CA and the pressure rise is due to charge compression and fuel combustion.

Here cycle 1 is a good representation of a standard HCCI cycle with a normal combustion phasing of  $12^\circ\text{CA}$  after top dead center (aTDC), it is then followed by cycle 2 which can be considered a misfire with no combustion occurring. The incomplete combustion residual fuel is transferred to the next cycle through internal EGR. As the combustion phasing is very late, the in-cylinder temperature increases which increases the temperature of the exhaust gas transferred to cycle 3. There is also the possibility that during the negative valve overlap (NVO) recompression a portion of the residual fuel ignites (as seen in cycle 3) and leads to a further temperature increase of the residual exhaust gas. The result is an increase in the temperature of the fresh air charge and the temperature after compression. This leads to an early combustion phasing with a high pressure rise rate. These results are consistent with other researchers [50, 64, 65]. It is thus essential to consider the coupling between subsequent combustion events in HCCI since it is an auto ignition process that depends on the in-cylinder conditions.

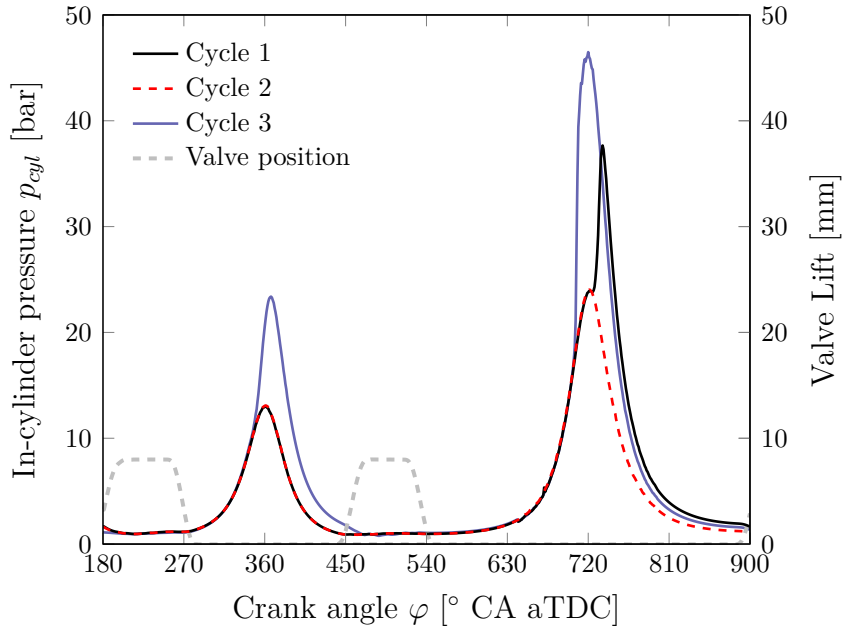


Figure 1.3: Distinct cyclic variation in the in-cylinder pressure  $p_{cyl}$  trace at  $n = 1500$  1/min, IMEP = 4.0 bar from [15].

To understand the coupling between combustion cycles, a return map of  $\text{CA}_{50}$  is

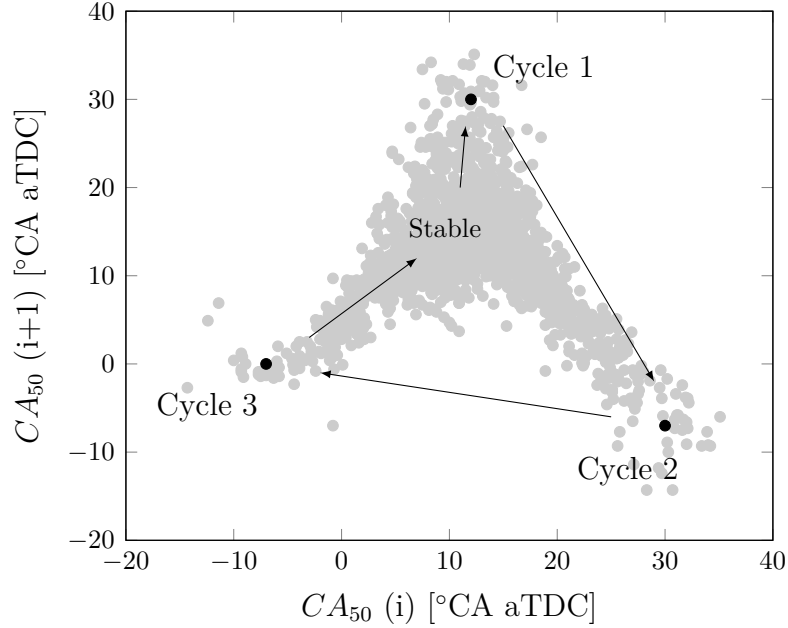


Figure 1.4: Distinct cyclic variation in combustion phasing  $CA_{50}$  return map at  $n = 1500$  1/min, IMEP = 4.0 bar from [15].

shown in Figure 1.4. A return map is used to show the relationship between the combustion phasing of the current cycle,  $CA_{50}(i)$ , and of the following one,  $CA_{50}(i + 1)$ . When the two consecutive cycles are not correlated, the return map shows a stochastic scatter of data points around the center of the return map. The spread of the data points can be used to represent the stochastic variation from cycle to cycle [49]. When a distinct pattern or branching can be seen on the return map as is the case in Figure 1.4, a direct coupling between cycles exists [55]. To effectively stabilize combustion the spread of the data points and distinct ‘V’ should be reduced. Return maps are often used in application where a short prediction horizon is desired. As they only provide correlation information for a few cycles.

The HCCI  $CA_{50}$  cyclic variation shown in Figure 1.4 is highly undesired as an early combustion phasing results in high pressure rise rates leading to increased combustion noise and possible engine damage [66, 67]. Overall, high cyclic variation of combustion also tends to reduce thermal efficiency and increase exhaust emissions [68]. The cycles with a very late combustion phasing (partial combustion) lead to an increase in HC

and CO emissions due to the unburnt fuel and low combustion temperatures. The very early and rapid combustion cycles have increased combustion temperatures leading to increase NO<sub>x</sub> emissions from these cycles. The combustion stability of an HCCI engine has a significant impact on its emission levels and maintaining combustion stability has been the focus of much research.

## 1.2 HCCI Control Strategies

This section will provide an overview of previous work regarding engine control strategies applied to various traditional combustion processes with a focus on current research covering HCCI combustion control.

### 1.2.1 Actuators for control

To stabilize HCCI combustion various control strategies utilizing a wide range of actuators to change the in-cylinder state have been both tested in simulation and experimentally implemented. Table 1.1 shows several control actuators that have been developed and tested for HCCI combustion. Each method has its advantages and disadvantages.

Intake air heating has been successful for achieving HCCI combustion but the response time of the heater makes it unsuitable for cycle by cycle control [29]. External exhaust gas recirculation (EGR) is generally too slow to change the cylinder conditions on a cycle-by-cycle basis and introduces another source of thermal energy loss, however, it can be used in combination with other control strategies [47, 69]. Internal EGR in the form of NVO through Variable Valve Timing (VVT) is a promising control method for both cycle-by-cycle and in-cycle control methods [70–72]. Using VVT also offers reduced throttling losses as power output can be regulated by diluting the incoming air charge with high EGR rates, however, they require a VVT system to be added to the engine [71]. Variable compression ratio systems have shown they

Table 1.1: HCCI Control Actuators (adapted from [63])

Control Variable	Actuator
Intake temperature [29]	Electric heating
Exhaust gas recirculation [47, 69–72]	External EGR
	Variable valve timing
	Exhaust rebreathing
	Exhaust throttling
Compression ratio [46]	Mechanical variable compression ratio
Mixture composition [40, 73, 74]	Dual fuel
	Pilot injection
	NVO recompression injection
Temperature [75]	Water injection
Spark assist [47]	Spark plug
	High energy spark system

have the ability to change the compression ratio of the cylinder on a cycle-by-cycle basis [46]. Two fuels with different ignition properties has shown potential for HCCI operation, however, this method requires two fuel sources and two separate injection systems [40, 73, 74]. Various fuel injection strategies have shown increased combustion timing control as well as reduced emissions [76]. Water injection had been shown to decrease the pressure rise rate and delay combustion by reducing cylinder temperature, however, like the dual fuel approach two injection systems are required and often large amounts of water are used [75]. Spark assist is the utilization of an electrical spark to initiate HCCI combustion which increases the cylinder temperature and pressure so the remaining fuel air mixture can auto-ignite. This control strategy has shown useful is extending the upper operating limit of HCCI but at the cost of increased  $\text{NO}_x$  emissions [47].

### 1.2.2 Traditional engine control strategies

In the automotive industry, current Engine Control Units (ECUs) utilize (mostly) feed-forward controllers. These controllers utilize a look-up table that relate parameters such as commanded load or speed to specific controller outputs including fueling, intake pressure and valve phasing. Often these tables are stacked with other tables containing correction factors for various external conditions such as weather or specific engine operation states. These look-up tables are then used in a feedforward controller which enables fast changes in operating points. The feedforward control is then often coupled with a feedback controller which take a measurement of the current engine operation to provide error compensation. Normally the feedback controller in engine control is a Proportional Integral Derivative (PID) controller or a slow adaptation table. The first major addition of feedback control for combustion control is the application of lambda regulation for SI engines where the current stochiometric ratio is measured using an oxygen sensor in the exhaust stream which is then used to adjust the engine fueling to correct for variation in the injectors or intake system [56].

These look-up tables (or calibration maps) are generated on a test bench through extensive testing to ensure the engine performs optimally while enhancing power output, fuel economy, and durability. As emissions regulations become increasingly more stringent, the development required for the creation of these calibration maps is also becoming extremely complicated and time-consuming [77, 78]. PID controller gains are tuned using the procedure of parameter optimization and fine tuned using trial-and-error. The optimization process, also known as engine calibration, results in finding look-up table values and controller gains [77, 78]. Due to increasing requirements for low-fuel consumption and emissions, the number of calibration parameters have increased substantially, making manual test-bench calibration difficult and time-consuming.



### 1.2.3 Advanced control strategies

Beyond the feed-forward look-up table based control strategies traditionally used in the automotive industry there has been a trend in both university research and industry to implement more advanced engine control strategies for some aspects of the engine controller. These advanced engine control methods are focused in two main areas: 1) utilizing high speed calculations on specialized engine controllers combined with fast actuators to allow for in-cycle control or 2) taking advantage of the increased computational performance of the processors on the market to implement various model based control strategies. Each of these areas will be described in more detail below.

#### **In-cycle engine controllers**

Using the low latency and parallel computing advantages of a field programmable gate array (FPGA) has increased their use in demanding in-cycle control applications where calculation time is limited. For example, FPGA hardware is used to provide closed loop control for the position of pneumatic valves [79]. Where the control is experimentally implemented and the valve control is both stable and able to provide a variety of valve strategies.

For HCCI control, FPGA hardware is used to calculate the heat release in real time (calculation time of less than 0.02 CAD at 1200 rpm) which is then used for HCCI combustion control [80]. The FPGA has also been used for a wide range of cylinder pressure based calculations including wall heat losses and IMEP and others [81]. To prevent an overshoot of IMEP in an HCCI engine where unburnt fuel is transferred from the previous cycle an FPGA controller is used to correct the amount of injected fuel in the next cycle [82]. An FPGA is used to calculate the real-time combustion metrics,  $CA_{50}$  and IMEP, which are then used for closed-loop control in [83]. The real-time calculated values were compared to offline post-processing results with ac-

ceptable results. FPGA hardware has also been used to implement a physical based gas exchange model that is used to calculate the cylinder state including in-cylinder pressure, temperature and trapped mass each  $0.1^\circ$  CA in [84].

To control HCCI over a wide operating range, in-cycle individual control interventions have become necessary to stabilize the combustion process [85]. Such interventions also aim to ensure low emissions [9] and high combustion efficiency [10]. Characteristic cycle-to-cycle dynamics or the reduction of the overall fuel consumption are the focus of [86] and [87], respectively. These approaches have been implemented using rule based-model free control. Thus, the potential of in-cycle control [88] by fast manipulation of variables, such as water injection [3, 10] and actively controlled glow plugs [89], has been demonstrated. Linear control laws for a single input and a single output variable (SISO) to stabilize the combustion are also demonstrated in [10].

These in-cycle control strategies have shown great potential to improve the HCCI combustion process. Combining in-cycle control with effective HCCI actuators (Table 1.1) can further improve HCCI combustion control. In addition, nesting fast in-cycle control with slower cycle-to-cycle control can also provide HCCI combustion stability improvements. This gap of in-cycle control with novel actuators and model based control for HCCI combustion stability improvement identified in literature will be investigated in this thesis.

### **Model based controllers**

Model based controllers for cycle to cycle control are the subject of this section. Unlike the in-cycle control strategies that utilize an FPGA most model based controllers take advantage of traditional microprocessor based ECU's which are readily available and in use in current production vehicles. As each generation of processors become more computationally powerful and energy efficient there is an increase capability available for engine control strategies. The calculation speed available on processors

limit the control strategy to cycle-to-cycle HCCI control. These cycle-by-cycle controllers consider the previous cycle for control interaction in the subsequent cycle. Therefore a prediction, often model based, of the upcoming combustion phasing is needed. Controllers used to track engine load and speed requests within the HCCI operating range and have been successfully implemented in cycle-to-cycle control [72].

Control strategies that have been simulated and experimentally tested which include feedforward and feedback control [90], model predictive controllers (MPC) [91–93], and iterative learning controllers (ILC) [40]. All of these strategies control the HCCI process and each have advantages and disadvantages.

For internal combustion engine applications in general, several model-based controllers have been used for engine feedback control, such as the Linear Quadratic Regulator (LQR) controller [94], Linear Quadratic Gaussian (LQG) controller [95], Sliding Model Controller (SMC) [96–98], Adaptive [98, 99], and Model Predictive Control (MPC) controller [100, 101]. Among these model-based controllers, MPC is one of the most promising for dealing with the highly constrained nonlinear system of ICEs [102]. MPC can provide an optimal real-time solution for meeting multi-objective goals while addressing system and operational constraints. New variants of MPC utilize optimization solvers and packages that are suitable for the real-time operation of time-critical systems [100, 101].

MPC, has been increasingly used in industry during the past four decades, has the following five main advantages: (1) it implicitly considers constraints on state, input, and output variables, (2) it provides closed loop control performance and stability for optimal problems with constraints, (3) it exploits the use of a future horizon while optimizing the current control law, (4) it offers the possibility of both offline and real-time implementations, and (5) it is capable of handling uncertainty in the system’s parameters, delays, and non-linearity in the model [103].

Model based optimal control techniques take advantage of speed increases of real-

time control hardware to allow for significant improvements in system modeling, which has led to an increase in interest from various researchers over the past two decades. MPC is a model based controller that is an effective control strategy widely used in a range of applications from the chemical process industry to the other industries such as automotive, power and energy systems, manufacturing, aerospace, healthcare, finance and others [104]. All of these applications take advantage of the ability of MPC to provide an optimal control solution while allowing for the implementation of constraints on system states and controller outputs.

The models for the MPC controllers can be designed from various modeling methods including physics based models (white box), Machine Learning (ML) models based on experimental data (black box) as well as a mix of the physics and experimental data (gray box) [105]. Each of these models have their advantages and disadvantages. However, independent of the type of the model, one challenge with MPC is the controller sensitivity to model uncertainty and the required model computational time for online optimization. Often a trade-off exists, where improving model accuracy leads to increased model complexity and processing power requirements and these complex models exhibit nonlinear behavior requiring a more complicated control law such as Nonlinear MPC (NMPC) [106, 107].

#### **1.2.4 HCCI performance and emissions modeling**

Due to the complexity of combustion phenomena and the high number of subsystems in ICE, physical-based model development is time-consuming and may become nonlinear and non-convex [102]. To capture the behavior of HCCI combustion, various simulation models including stochastic, multi-zone and physical models have been developed to predict the gas exchange and combustion processes [65, 86, 108, 109]. Detailed physical models are typically too computationally intensive for use in real-time engine applications and so are often linearized around a specific operating point

for implementation in processor based engine controllers [91, 110]. In addition, the accuracy of the physics-based method is typically reduced, mainly due to linearization or model-order reduction techniques. Data-driven or black-box models that use measurement data directly for training ML methods are an alternative approach for modeling. These models could be as accurate as 3D Computational Fluid Dynamics (CFD) physical models and require significantly less processing time for implementation of model-based controllers in ECUs, however, they do require significant effort for offline training [111–113].

Despite the many advantages of data-driven black-box models, there are two main drawbacks: i) they are complex and could run a high risk of overfitting, particularly when a large number of features are used [111–113], and ii) because they do not contain physical models, their accuracy will decrease when the physics underlying the model change [114, 115].

To meet current and the upcoming emission regulations a detailed understanding of HCCI engine emission formation is essential. However, predicting the exact value of engine out emissions using conventional physics-based models is still a challenge for engine researchers due to the complexity of combustion and emission formation modeling [114]. This has led researchers to consider machine learning (ML) based methods which help to provide an accurate model while minimizing the computational requirements [116].

ML techniques have been widely used for addressing engine performance, emission modeling and control [117, 118]. To this end, different ML methods have been tested and used for HCCI performance, combustion phasing, and emission modeling using an Artificial Neural Network (ANN) [119–123], Extreme Learning Machine (ELM) [124–126], Bayesian Neural Network (BNN) [127], Deep Neural Networks (DNN) [128], and Least Squared Support Vector Machine (LS-SVM) [100, 101, 129]. Among these methods, most researchers have focused on the prediction of engine performance,

consisting of IMEP and  $CA_{50}$  [100, 101, 121, 124–126, 129] while a limited number of researchers have studied emission prediction [119, 120, 122, 130]. ANN has been the ML method that has been widely used for emission and performance prediction for SI and CI engines [122, 131–133]. This has led researchers to consider ANN the baseline ML method for engine modeling and control implementation. However, to create an accurate model ANN requires a large data set which requires significant engine testing time and results in high testing costs.

One of the most powerful machine learning methods that has shown remarkable accuracy in the prediction of ICE emissions and performance is Support Vector Machine (SVM) [11, 115, 117, 131, 134]. SVM is a machine learning approach which has been used for both classification and regression problems [135, 136]. By providing the SVM with a set of input and output pairs, it approximates a hyperplane to retrieve a pattern that exists between given inputs and the corresponding outputs. For HCCI, SVM has been used to predict combustion phasing, misfire, and high pressure rise rates [137]. For example, it has accurately predicted  $CA_{50}$  with an error of 1.9% for transient load changes [137], and cyclic combustion variability [138]. Transit Linear Parameter Varying (LPV) based models were developed to predict  $CA_{50}$  and IMEP [100, 101, 129]. The accurate prediction capabilities and low computational requirements of SVM has proven it is a powerful technique for predicting the complex and highly nonlinear phenomena provided the motivation to apply this strategy to emission formation in HCCI engines and compare the results to ANN. SVM has been used to predict the performance and emissions of SI [139, 140] and diesel [11, 115, 117, 134] engines but, to date, has not been comprehensively investigated for HCCI emissions prediction.

In ML, hyperparameter tuning, is often difficult due to local minima. In ANN, a grid search for the number of neurons and number of hidden layers is usually used to find optimal hyperparameters. Depending on the depth of the network, a

random search could be added during optimization [120]. Metaheuristic approaches were also used to tune ANN hyperparameters such as Particle Swarm Optimization (PSO) [141, 142] and Genetic Algorithm (GA) [143]. Compared with GA, PSO is a relatively new heuristic search method based on collaborative behavior and swarming in biological populations. Both GA and PSO are population-based search approaches that depend on information sharing among their population members. Although PSO and GA have a similar performance in terms of the accuracy of the solution, it has been proven that PSO is computationally more efficient, and requires fewer parameters that need to be defined for optimization [144, 145]. In SVM, there are three main parameters to tune which are tolerated error, kernel function parameters, and regularization coefficient.

A Recurrent Neural Network (RNN) for ML-based NMPC is structurally similar to a feedforward neural network with the exception of backward connections used to handle sequential inputs [146]. The advantage of the RNN is its computational efficiency which is the result of parameter sharing. However, RNN cannot accurately capture any long-term dependencies of the model as the prediction is solely based on recent steps. This can also be described as the “vanishing gradient”, with the contribution of earlier steps becoming increasingly small. The challenge with RNN is the lack of long term memory; however, memory cells can be introduced to help solve this problem. The most well-known form of these long-term memory cells is the Long Short-Term Memory (LSTM) cell [146].

Combining LSTM and NMPC, denoted as LSTM-NMPC, has shown its potential in optimal temperature set-point planning for energy efficient buildings [147], steam quality of thermal power units [148], and for motion prediction of surrounding vehicles in an autonomous vehicle [149]. With the success in these previous applications, LSTM-NMPC is now being considered for systems requiring fast time steps such as control of an internal combustion engine. To implement this, embedded programming

techniques are required.

The structure of most of the previous work in ML-based MPC for ICEs has been linear [100, 101, 121, 150] or a non-linear model that has been linearized [106, 124, 151]. Only a few previous researchers have explored a nonlinear data-driven structure [150, 152, 153]. Among these works, only one study was found that explores ML-based MPC control of a CI engine where a linearized model of nonlinear ML-based model was used to design and implement a controller [151]. Further, for considering real-time implementation only linear models such as linear parameter varying (LPV) [100] or linearized model [124] have been successfully implemented experimentally. All previous ML-based nonlinear MPC (NMPC) have only been implemented in simulation [152, 153]. For HCCI, No previous experimental work has been found considering data-driven-based NMPC using DNN with a LSTM layer.

### 1.2.5 Timing constraints

It is important to consider the computational complexity when considering HCCI control strategies. This applies to both the in-cycle controllers, which have an extremely limited calculation window available, as well as the model based control strategies. Figure 1.5 shows some possible in-cycle control actuation that can be used to control HCCI combustion. This figure considers two water injection events (blue drop) one before the NVO recompression and one during the main compression. Two fuel injection events (green drop) are considered as well as two spark interactions (yellow ‘lightning bolt’) one at the traditional timing and one late spark. By taking the in-cylinder state at EVC for an engine speed of 1500 rpm, the available calculation time before control actuation is approximately 0.22 ms for the start of the pre-fuel ( $t_{EVC \rightarrow SOI_{fuel}}$ ) or pre-water injection ( $t_{EVC \rightarrow SOI_{water}}$ ) to influence the NVO period. This timing constraint is not feasible using a traditional microprocessor based ECU but can be easily achieved when using a FPGA based ECU. From IVC until a spark actuation



( $t_{IVC \rightarrow spark_{main}}$ ) is only 14.4 ms at 1500 rpm is available. For these control actions the limited calculation time means that an FPGA must be used instead of a conventional cycle-to-cycle ECU.

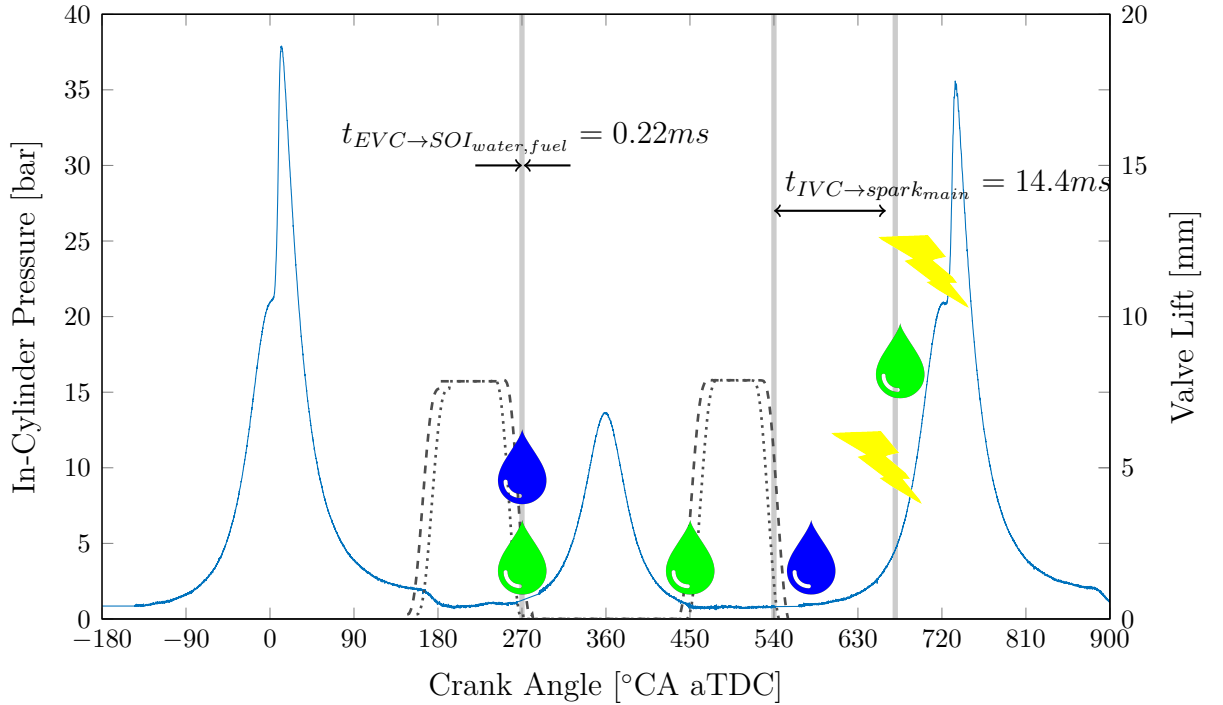


Figure 1.5: HCCI control variables and available calculation time. Blue drop represents a water injection, green drop is a fuel injection event, while a spark actuation is denoted by the ‘lightning bolt’.

The calculation window for a cycle-to-cycle controller is dependent on the speed of the engine. At 1200 rpm, a complete four stroke engine cycle lasts 100 ms while at 1800 rpm only 66.7 ms are available. For the HCCI tests in this thesis an engine speed of 1500 rpm is used which has a duration of 80 ms. Cylinder state calculations from the last cycle must be complete before starting the calculation of any model based controller. Figure 1.6 shows the time available for cycle-to-cycle model based control if changes to the NVO duration are too be made. Here the FPGA based cylinder state calculation are completed at 60 CAD aTDC and the control actuation to the valve timing is required at 260 CAD aTDC leaving 200 CAD for the calculation (22 ms at 1500 rpm). To meet these real-time timing requirements, a computationally efficient

algorithm is needed to provide feedback from one engine cycle to the next cycle for any model based control strategies.

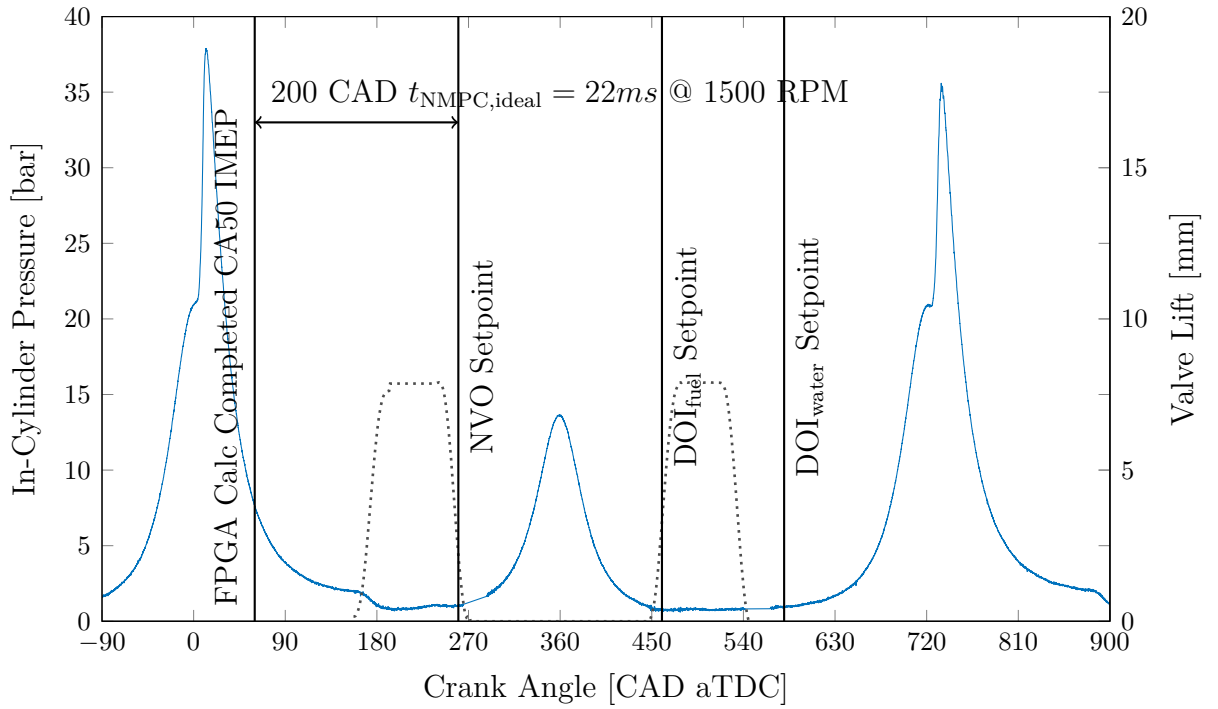


Figure 1.6: Calculation time for model based controllers

### 1.3 Problem Identification and Proposed Solutions

To simulate a steady state operating point the engine is operated in a conditioned environment that keeps rotational speed, load, intake pressure and temperature, oil and coolant temperature and exhaust pressure constant. Only one load and speed is selected to reduce the experimental space and demonstrate the proposed emission model.

Based on the literature review in the previous section, it is well understood that HCCI has a narrow operating range and has high efficiency and low emissions within specific operating conditions [25, 154, 155]. In vehicles, the limited load range is not as critical in hybrid and electric range extender applications. These require only a few efficient load and speed operating points as the electric part of the propulsion systems are used to handle any transient loads. Therefore, the ability of HCCI to greatly reduce engine out emissions as well as lower fuel consumption compared to traditional combustion engines is beneficial. Even with all the benefits of HCCI, there are still challenges that need to be overcome to allow for a wide adoption of HCCI within industry. Research over the past 40 years has led to improvements in controlling HCCI. However, the combustion stability of the HCCI process still requires improvement to allow for a wider operating range of both load and engine speed. The vast improvement of engine control hardware, provides a platform to implement improved advanced control strategies for HCCI. Two main methods to improve the combustion stability are apparent from current research:

1. In-cycle control strategies requiring actuators and high speed controllers.
2. Model based control strategies utilizing physics based white-box models black-box machine learning based models and combinations (gray box models).

Each of these two control strategies has their own advantages and disadvantages which will be discussed in this thesis.

### 1.3.1 In-cycle controller

The main objective of these controllers is to affect the combustion process within the same combustion cycle. This requires high speed FPGA based engine models and rapid actuators to quickly intervene into the combustion process. To improve HCCI combustion stability, large cyclic variations need to be reduced. Previous work considered the addition of direct water injection to improve the combustion stability of the HCCI process [4, 15, 84]. This work showed a significant improvement in combustion stability as the water injection was able to help prevent the early combustion event that would normally follow a misfire cycle. The challenge with this control strategy is nothing could be done about misfire cycles. Therefore, this thesis will look to specifically address these misfire or extremely late combustion cycles by adding a high energy spark actuator. Similar to the previous work [15], this high speed spark will be controlled using a FPGA to enable rapid controller calculations as shown in Chapter 3.

Another area that could be improved for the control of HCCI is combining various previously tested real-time control strategies and have multiple control actions within a single cycle. Combining these timescales within a cycle and cycle-to-cycle allows for multi-scale control to even further improve HCCI combustion stability. This multi-scale control strategy will be further investigated in this thesis in Chapter 4.

### 1.3.2 Model-based controller

As mentioned in Section 1.2, MPC is a promising control strategy for the highly constrained nonlinear ICE combustion process. The use of a moving prediction horizon allows control optimization while considering constraints on state, input and output variables. MPC can also handle multi-variable systems. Since ICE is highly constrained, MPC is an ideal control strategy. The ability of MPC to inherently enforce constraints can significantly reduce calibration and development time. Flexible han-

dling of uncertainty, delays, and non-linearity in the model and the possibility of real-time implementation are other advantages of MPC that make it applicable in the automotive industry [103]. These MPC features have resulted in MPC being investigated widely in the automotive industry. However, MPC requires additional computing and memory resources compared with classic control and an accurate model is also needed.

As the name suggests, a MPC controller has a model of the process embedded and the performance of the MPC is directly dependent on the accuracy of this model. The models for the MPC controllers can be designed from various modeling methods including physics based models (white box) [156], Machine Learning (ML) models based on experimental data (black box) [157] as well as a mix of the physics and experimental data (gray box) [158]. Each of these models have their advantages and disadvantages [114]. For all types of models, two challenges with MPC are: the controller sensitivity to model uncertainty, and the required online optimization time. Often a trade-off exists, where improving model accuracy leads to increased model complexity and these complex models exhibit nonlinear behavior requiring a more complicated control law such as Nonlinear MPC (NMPC).

Joint work focusing on control of diesel engines [2, 159], has shown that NMPC is a useful method to control a diesel engine while meeting constraints. The current work in literature and experience with diesel NMPC control provided the motivation to investigate the experimental implementation of NMPC for HCCI. The experience gained from diesel combustion control is applied to HCCI which is an unstable combustion process.

In this thesis, three different modeling strategies for MPC are investigated: 1) A physics based white-box model considering the combustion kinetics; a black box support vector machine model; and finally a transient long-short term memory model. Since these three model strategies are quite different, Chapter 5 will present the HCCI

modeling process. After comparing the performance of the three models in simulation, the LSTM-NMPC was found to exhibit the best characteristics and thus was experimentally implemented to test the online performance of the NMPC on the dSPACE MicroAutoBox II (MABX II) prototype engine controller. The online results are described in detail in Chapter 6. The processing power of the MABX II was insufficient for cycle-to-cycle control using the developed NMPC. Therefore, the LSTM-NMPC is then implemented on an external processor to allow for true cycle-to-cycle control with detailed results in Chapter 6.3.2.

## 1.4 Thesis Outline and Contributions

### 1.4.1 Thesis outline

This thesis is organized into seven main chapters. Figure 1.7 shows the control structure tested and where each part is developed in this thesis. The goal of this thesis is to develop a control strategy to improve the performance and emissions of HCCI combustion by considering all of the parts of the traditional control loop including: 1) actuation, 2) fast sensing, 3) in-cycle HCCI control, 4) model based control with constraints. While considering the entire control system one of the main goals of the thesis work was to develop a control strategy that allowed for a significant reduction in the engine calibration effort while allowing for efficient engine control. The chapters of this thesis are as follow:

- **Chapter 1** provides background, motivation, and main contributions of this thesis.
- **Chapter 2** presents details on the two experimental engines used in this work and the actuators and control systems tested.
- **Chapter 3** presents details on the experimental implementation of in-cycle engine control using a rapid ignition system.

- **Chapter 4** presents the combination of in-cycle and cycle-to-cycle controllers for a combined multiscale control approach..
- **Chapter 5** provides three different emissions and engine performance modeling methods. One white-box kinetics model, one support vector machine black-box steady-state model and one deep neural network transient engine model.
- **Chapter 6** presents the development of the online NMPC utilizing the transient ML engine model which has been experimentally tested.
- **Chapter 7** presents overall conclusions from the work presented as well as future work.

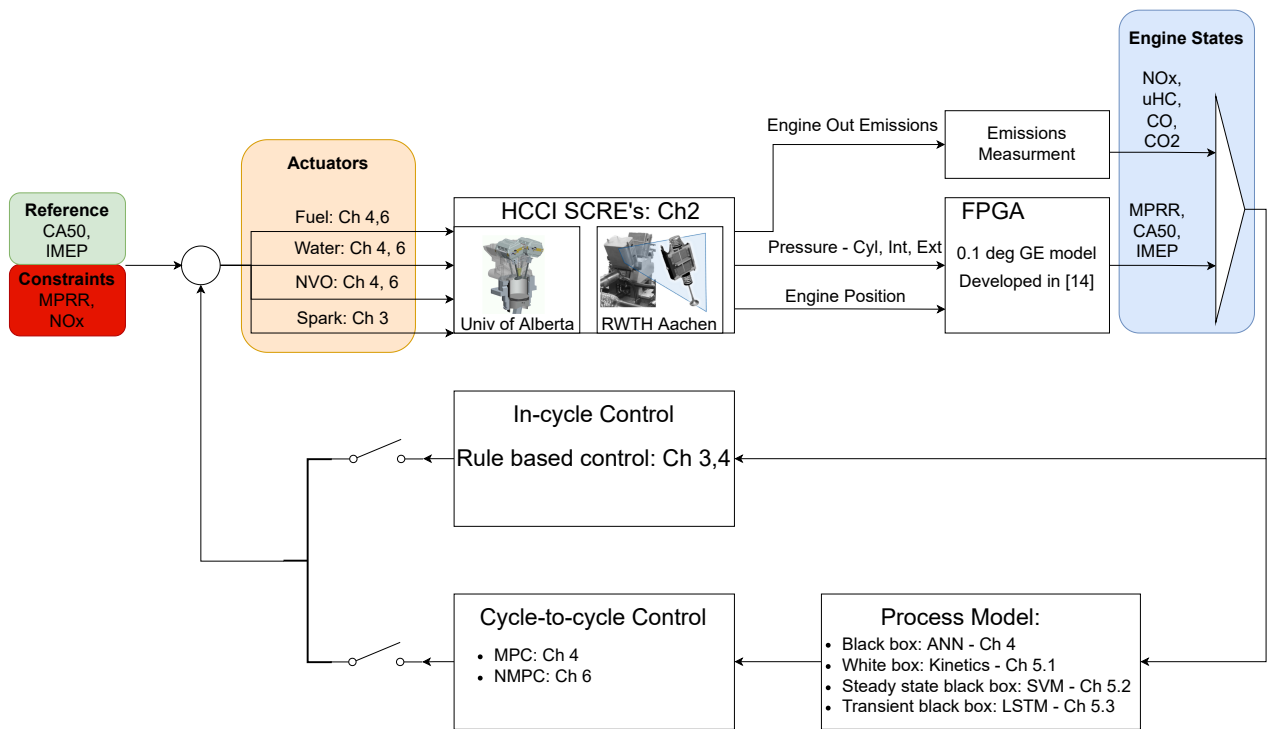


Figure 1.7: Schematic of the thesis organization - contributions in the area of: 1) actuation, 2) fast sensing, 3) in-cycle HCCI control, 4) model based control with constraints.

## 1.4.2 Contributions

A summary of the main contributions of this thesis are (publications resulting from each chapter are noted. A full list of publications can be found in Appendix A ):

- **Chapter 2: Experimental setup**
  - Upgrading up a single cylinder research engine to allow for FPGA based engine control strategies to allow for experimental implementation of in-cycle control.
  - Integration of prototype novel fast acting high energy spark actuation experimentally to develop novel in-cycle HCCI control strategies using spark.
- **Chapter 3: In-cycle controller development and experimental validation using a rapid igniton system**
  - Development and experimental validation of in-cycle controller using a high energy spark to reduce emissions and improve HCCI stability [8, 9].
  - Experimental implementation of multi-scale control to combine an in-cycle and cycle-to-cycle controller to improve HCCI stability. The developed controller improved the combustion stability by 18% compared the the previous in-cycle controller [5].
- **Chapter 4: Combination of in-cycle and cycle-to-cycle controller for development and experimental validation using a multiscale control approach**
  - Experimental implementation of multi-scale control to combine an in-cycle and cycle-to-cycle controller to improve HCCI stability. The developed controller improved the combustion stability by 18% compared to the previous in-cycle controller [5].



- **Chapter 5: HCCI emissions and performance modeling**
  - A white-box kinetics model for real-time calculation of cylinder chemical composition is developed for the FPGA hardware which is suitable for in-cycle control.
  - A black-box support vector machine model for HCCI emissions prediction is developed for cycle-to-cycle control. This will allow for steady-state emissions to be considered in an MPC [13].
  - A deep neural network model containing a long short term memory (LSTM) units for transient HCCI performance and emission modeling that is suitable for MPC implementation [2].
- **Chapter 6: Experimental implementation of NMPC for HCCI stability and emissions improvement**
  - A novel approach to augment LSTM in the NMPC problem (LSTM-NMPC) by augmenting LSTM hidden and cell state into nonlinear optimization problem is described [14].
  - State of the art realtime NMPC implementation on a real system using LSTM. Executed on low-cost hardware .
- Method to develop realtime NMPC controller while significantly reducing calibration effort.

# Chapter 2

## Experimental Setup <sup>1</sup>

In this chapter, details covering the specifics of the two experimental single cylinder research engines (SCRE) are presented. The first engine is located at the University of Alberta located in Edmonton, Alberta, Canada. The second SCRE is located at the Teaching and Research Area Mechatronics in Mobile Propulsion at RWTH Aachen University located in Aachen, Germany. These two engines are both setup for testing of HCCI combustion and some similarities as well as a few major differences. The work in this thesis focuses on these two experimental engines as their differences allowed for different control and modeling of HCCI to be performed.

### 2.1 University of Alberta SCRE

The first experimental test engine is located at the University of Alberta which is shown schematically in Figure 2.1. This SCRE is outfitted with a hinged Electromagnetic valve train (EMVT) (shown in Figure 2.2) is used to collect experimental data for this thesis. The EMVT system utilize a hinged plate between two electromagnetic coils which are actively controlled using a dSPACE 1103 [160]. The valve position is estimated by integrating the magnetic flux of the coil not in use. This allows for closed loop control of the valve position to allow the valves to follow a defined opening profile. The flexibility of the valve train allows for engine operation

---

<sup>1</sup> This chapter is partially based on [8, 15]

with a wide array of valve strategies including NVO. The engine geometry and testing conditions are listed in Table 2.1.

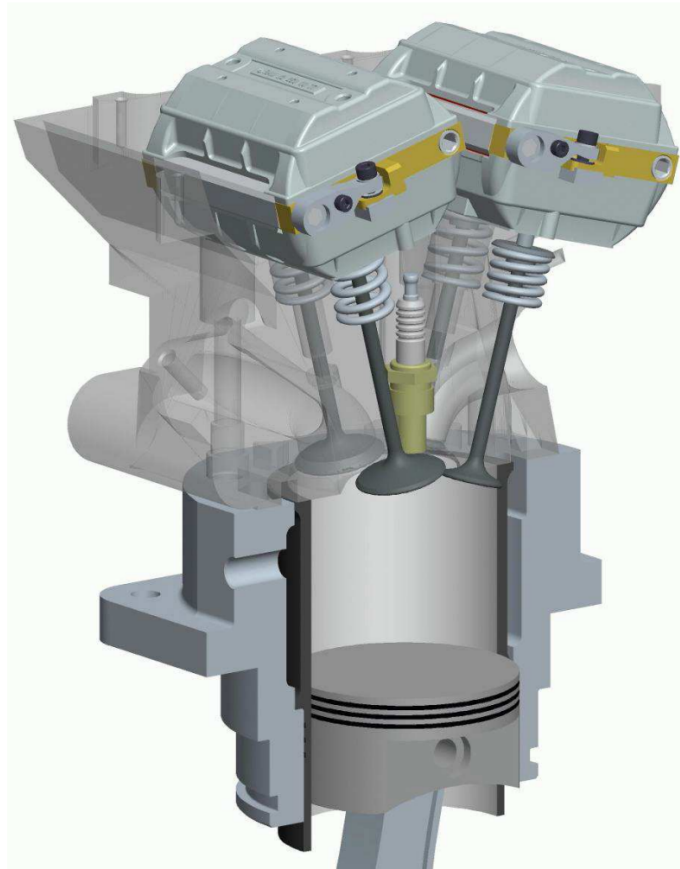


Figure 2.1: University of Alberta SCRE schematic [160].

Fuel is injected into the SCRE intake port using two conventional fuel injectors, which allows for a variable n-heptane to iso-octane ratio. Thus, the research octane number (RON) of the injected fuel mixture can be adjusted between 0 (n-heptane) and 100 (iso-octane) cycle by cycle as desired. The HCCI process is controlled by the VVT system. To minimize compression losses the timings for EVC and IVO are always symmetrical to TDC. EVO and IVC are fixed at 165 CAD and 585 CAD. The injection into the intake manifold is fixed at 350 CAD for all experiments conducted for in this thesis.

In-cylinder pressure is measured using a Kistler 6041 piezoelectric pressure transducer. The intake and exhaust manifold pressures are measured using Kistler 4045-A5

Table 2.1: University of Alberta - Single cylinder research engine parameters

Parameter	Value
Displacement volume	0.657 l
Clearance volume	0.0509 l
Stroke	88.9 mm
Bore	97 mm
Compression ratio	13.9:1
Engine speed	840 rev/min
No. of valves (In/Ex)	2/2
Valve train	Hinged EMVT
Connecting rod length	159 mm
Crank shaft radius	44.5 mm
Piston area	7400 mm <sup>2</sup>
Oil and coolant temperature	90 °C
Fuel rail pressure	3 bar
Intake temperature	31 °C
Intake / exhaust pressure	910 mbar

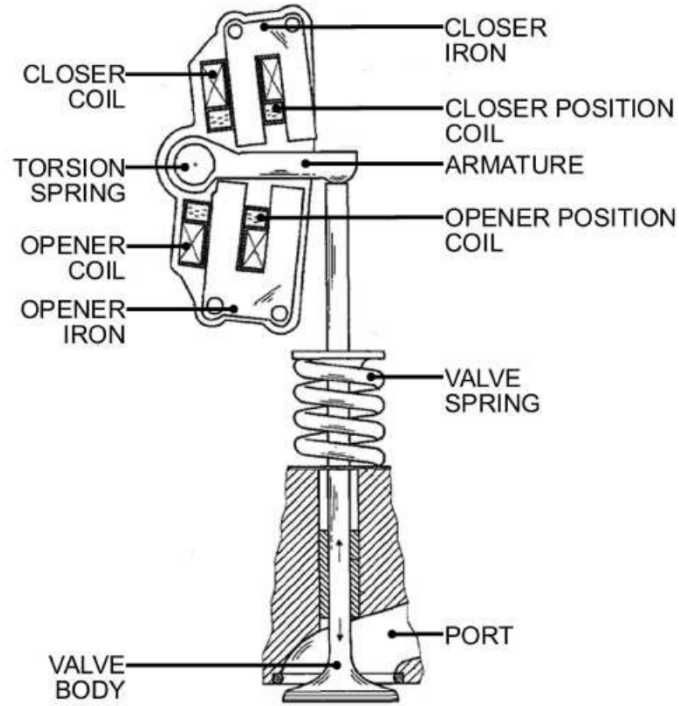


Figure 2.2: Hinged electromagnetic valve schematic for University of Alberta SCRE [161].

piezoresistive pressure transducers. The angular position of the crank is measured using a 0.1 CAD resolution optical encoder. All of these signals are input to the dSPACE MABX II prototyping ECU [162].

The engine is operated in a conditioned environment that keeps rotational speed, intake pressure and temperature, oil and coolant temperature and exhaust pressure constant. The active inputs to the HCCI process are injected fuel mass, injection timing and valve timings. All experiments are done at an engine speed of 840 rpm and all temperatures are conditioned to limit additional variations as specified in Table 2.1. The air-fuel ratio is varied as well as the power output.

The engine coolant is run through a flat plate heat exchanger which is connected to the building cooling water system. This helps to remove heat from the engine while in operation. The engine coolant temperature can also be increased using a electric heater which is controlled using the Taylor Dynopro2 system.

The oil is supplied to the engine through an external oil conditioning system which filters, heats and pressurizes the oil which is supplied to the engine. The oil temperature is regulated along with the oil pressure which is set to 3.5 bar  $\pm$ 0.1 bar for all experiments in this work.

### 2.1.1 Data acquisition and engine control systems

A National Instruments Data Acquisition System (DAQ), model PCIe-6351, was used to record cylinder pressure at a 0.1°CA resolution for use in offline post-processing analysis. Additionally, the in-cylinder and manifold pressure measurements are simultaneously input to the Field Programmable Gate Array (FPGA) board in the prototyping ECU. The FPGA board is located within the dSPACE MABX II prototyping ECU where detailed information is provided in Table 2.2. The MABX II contains two main boards: a CPU and FPGA. The CPU (ds1401) was used to implement the controllers developed in this work.

The Xilinx Kintex-7 FPGA contained within the MABX II was used to calculate combustion metrics in real-time. IMEP, CA<sub>50</sub> and MPRR are calculated and are transferred from the FPGA to CPU for use as inputs for control. Details regarding the real-time calculation of these properties can be found in [81, 163].

### 2.1.2 Emission measurement

To measure engine-out emission emissions, an electrochemical NO<sub>x</sub> sensor, and MKS Fourier-Transform Infrared Spectroscopy (FTIR) were used. The sensors sample directly from the engine exhaust pipe.

A production amperometric NO<sub>x</sub> sensor (ECM-06-05) was used in the experiments. All the sensor working parameters were set using the sensor control module (*ECM-NOxCANt* P/N: 02-07). The sensor control module was connected to the dSPACE MABX II via a Controller Area Network (CAN) to record the NO<sub>x</sub> emissions.

The exhaust gas measurement is done using a MKS MultiGas 2030 FTIR Sys-

Table 2.2: Rapid prototyping ECU Specifications

	Parameter	Specification
Processor	dSPACE <sup>®</sup> 1401	IBM PPC-750GL
	Speed	900 MHz
	Memory	16 MB main memory
I/O	dSPACE <sup>®</sup> 1511	
	Analog input	16 Parallel channels
	Resolution	16 bit
	Sampling frequency	1 Msps
	Analog output	4 Channels
	Digital input	40 Channels
	Digital output	40 Channels
FPGA	dSPACE <sup>®</sup> 1514	Xilinx <sup>®</sup> Kintex-7
	Flip-flops	407600
	Lookup table	203800
	Memory lookup table	64000
	Block RAM	445
	DSP	840
	I/O	478

tem [164]. The FTIR analyzer was used to measure the concentration of various chemical components in the exhaust gas. The FTIR spectrometer passes an infrared beam through a gas sample, obtains the interference pattern of the gas, and identifies the gas composition based on the absorption spectrum of the gas constituents. The sample exhaust gas passed through two heated filters (*Flexotherm Flex*) connected by sample lines (*Flexotherm*) heated to 191°C to avoid water vapor condensation in the sample gas. The sample data was collected at a 5 Hz frequency using the mks Series 2000 MultiGas analyzer software version 10.1.

Finally, the air-fuel ratio  $\lambda$  is measured by a Bosch series-build oxygen sensor powered by ECM electronics to convert the sensor measurement to an analog signal which is provided to the dSPACE MABX II.

## 2.2 RWTH Aachen SCRE

The SCRE located at RWTH Aachen University is also outfitted with a fully variable EMVT. This engine has been specifically designed to operate with the EMVT system. The flexibility of the valve timing allows for any desired valve timing to be implemented and changed on a cyclic basis if desired. Engine operation with combustion chamber exhaust gas re-circulation through NVO is useful for HCCI combustion and can be easily achieved with this setup. This allows for a wide operating range of HCCI combustion timings to be implemented. The FEV Europe GmbH (FEV) designed EMVT system is depicted in Figure 2.3 [165, 166]. This figure shows an armature which is located between two electromagnetic coils, one to provide opening force and one to close the valve. Two valve springs provided the restoring force to return the valve to its resting position of half open. The position of the electromagnetic valves are measured using FEV conductive lift measurement sensors. The position of the valve is used for closed-loop control of the valve position, which will be discussed later. This EMVT system differs from the one at the University of Alberta (described in section 2.1) as the linear motion of these valves can be directly measured using a position sensor, however, the linear motion when compared to the hinged actuation of the UofA system results in longer travel distances and thus higher valve landing velocities. The schematic of the valves and engine head is shown in Figure 2.4.

One challenge with the EMVT systems is that to protect the valves against a possible collision with the piston. To achieve this, the piston must be manufactured with deep valve pockets to allow space between fully open valves and the piston face at Top Dead Center (TDC). This is the worst case and these engine are called free



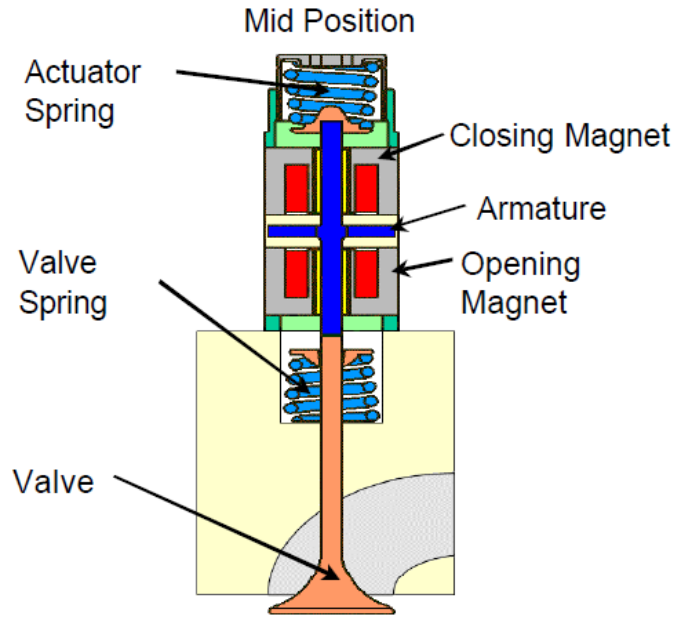


Figure 2.3: Electromagnetic valve schematic from [167].

running. The deep piston pockets lower the cylinder compression ratio to 12:1. The engine specifications and testing conditions are defined in Table 2.3.

The dynamometer controller is set to hold speed constant despite changing engine torque for all experiments in this work. The engine speed is regulated to a set value and the torque produced by the engine is measured using a HBM U2A/200kg load cell. The angular position of the engine is measured using a Heidenhain ROD 430 encoder which has a 0.1 degree angular resolution.

The intake air is conditioned before it reaches the engine. Using two Roots-type superchargers powered by a variable frequency drive controlled electric motor the intake pressure can be adjusted up to 1.8 bar boost. The temperature of the intake air is also controllable using an in-line air heater to provide various intake temperatures and is controlled using the ADAPT system described later. The volume of air entering the engine is measured using an Aerzen gas flow meter which provides an overall flow rate entering the engine.

The exhaust back pressure can also be controlled using a pneumatically powered

Table 2.3: RWTH Aachen University - Single cylinder research engine parameters

Parameter	Value
Displacement volume	0.499 L
Stroke	90 mm
Bore	84 mm
Compression ratio	12:1
Engine speed	1500 rev/min
No. of valves (In/Ex)	2/2
Valve train	Linear EMVT
Max. valve lift (In/Ex)	8 mm/8 mm
Valve angle (In/Ex)	22.5°/22.5°
Valve diameter (In/Ex)	32 mm/26 mm
Intake air pressure	1013 mbar
Exhaust pressure	1013 mbar
Oil and coolant temperature	90 °C
Engine speed	1500 rpm
Fuel rail pressure	100 bar
Intake temperature	50 °C

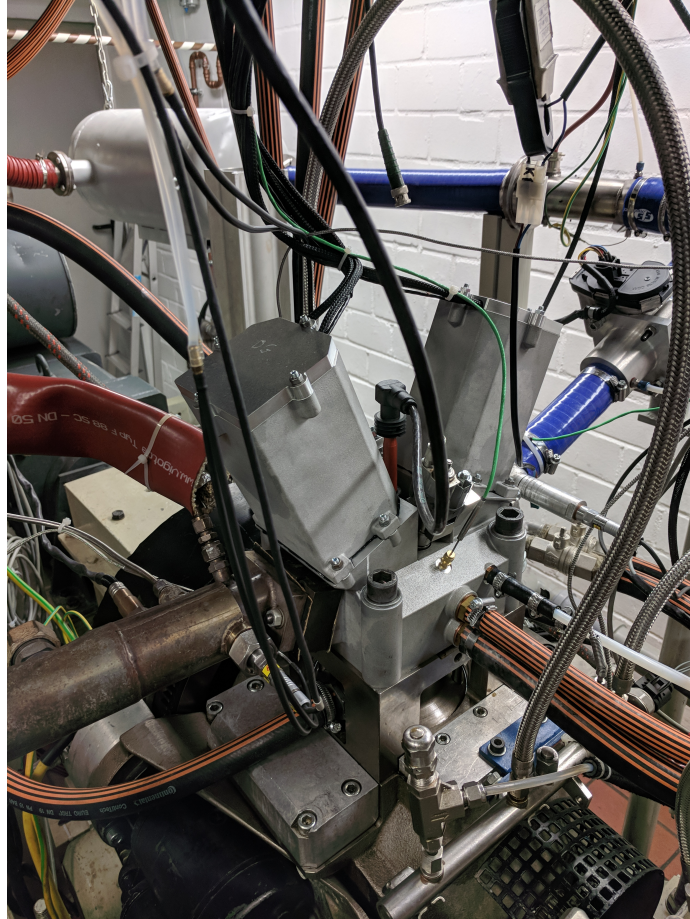


Figure 2.4: RWTH Aachen SCRE photo [168].

valve. For this work the back pressure in the exhaust system is regulated to  $1013 \text{ mbar} \pm 25 \text{ mbar}$  for all experiments.

The in-cylinder pressure is measured using a Kistler 6041 piezoelectric pressure transducer that is mounted between the intake and exhaust valves. The intake and exhaust manifold pressures are measured using Kistler 4045A piezoresistive pressure transducers. To convert the output from the three pressure sensors to a voltage, three Kistler charge amplifiers are used. Each charge amplifier and pressure transducer are calibrated as a pair to reduce the amount of measurement error.

The fuel used for all testing in this work is conventional European RON 96 gasoline containing 10% ethanol. The fuel is pressurized first using a low pressure fuel pump up to 6 bar and then up to 100 bar using a high pressure pump before injection

directly into the combustion chamber. The fuel pressure is measured using a rail pressure transducer. The amount of fuel injected is measured using a FEV Seppeler fuel measurement system. This system can only be used for steady state fueling measurements, as it requires multiple cycles to record the time to inject a known volume of fuel. However, it can also be used to calibrate an injector function to determine the fuel injected for a known injection duration for in-cycle measurements. The fuel injector used is a BMW 13537589048 piezoelectric outward-opening hollow cone injector. This injector is powered by Siemens VDO injection controller which is used to provide the necessary voltage to open the injector.

Similar to the fuel injection system, the distilled water that is directly injected into the combustion chamber is first pressurized to 6 bar using a low pressure pump. Then an electronically controlled high pressure pump is used to pressurize the water to injection pressure ranging from 15-175 bar. This pressure is measured using a rail pressure transducer and is used for closed-loop control of the rail pressure. The direct injection of water is provided through a BMW 13537589048 piezoelectric outward-opening hollow cone injector. This injector is powered using VEMAC VAPA power electronics which provide the required voltage to open the injector. It also allows for customization of the injection opening profile, however, for this work the opening profile was not adjusted.

The engine coolant is run through a flat plate heat exchanger which is connected to the building cooling water system. This helps to remove heat from the engine while in operation. The engine coolant temperature can also be increased using an electric heater which is controlled using the ADAPT system. For all tests the coolant temperature is maintained at  $90\text{ }^{\circ}\text{C} \pm 2^{\circ}\text{C}$ .

The oil is supplied to the engine through an external oil conditioning system which filters, heats and pressurizes the oil which is supplied to the engine. The oil temperature is regulated to  $105\text{ }^{\circ}\text{C} \pm 2^{\circ}\text{C}$  which helps to ensure any water in the oil is

evaporated. The oil pressure is set to 5.4 bar  $\pm$ 0.2 bar for all experiments in this work.

### 2.2.1 Emissions measurement

An emission measurement system is used to determine the composition of exhaust gas. The two measurement devices used are an Eco Physics CLD700REht for NO and NOx measurement and a Rosemount NGA 2000 which provides measurements of unburnt hydrocarbons, carbon monoxide, carbon dioxide, and oxygen concentration. The specifications of the emission measurement system is provided in Table 2.4.

Table 2.4: Accuracy of emissions measurement system

Gas	Maximum	Detection level	Resolution	Accuracy
NOx	10000 ppm	0.1 ppm	0.1 ppm	1% of reading
uHC	5%	0.04 ppm	0.1 ppm	1% of reading
CO (low)	2500 ppm	0.1ppm	0.1ppm	1% of reading
CO (high)	10%	0.1%	0.1%	1% of reading
CO2	18%	0.1%	0.1%	1% of reading
O2	25%	0.1%	0.1%	1% of reading

### 2.2.2 Data acquisition and engine control systems

Engine control is provided by three main systems. The first system is an A&D ADAPT system which provides control of engine operating parameters including load, speed, oil and water temperature and pressure, exhaust back pressure, and intake temperature. This system also is used for low speed data acquisition which is described in the following section.

The main engine control is provided by a dSPACE MicroAutoBox II (MABX) prototyping ECU which contains both microprocessor and FPGA modules. This

ECU is used for the implementation of all control algorithms to provide control of the valves, ignition, water and fuel injection timing and water rail pressure. The software that is run on the MABX is compiled using Matlab Simulink. This makes implementing and testing controllers easier and provides a user friendly interface. Another advantage of using the MABX II is that it is connected to a computer running dSPACE ControlDesk which provides access to the variables in the control algorithms and allows for adaption the engine controller during engine operation. More detailed technical specifications are provided in Table 2.2. The same MABX II is used for both the engine at the University of Alberta and RWTH Aachen University which allows easy code transfer.

A dSPACE DS1103 is used to provide closed-loop control of the EMVT system. The DS1103 runs an FEV created C-code based controller which uses the valve position sensors to determine the correct valve opening and closing currents based on the desired valve opening and closing times from the MABX. The DS1103 is also connected to the same computer running dSPACE ControlDesk as the MABX II, this provides access to enable the starting sequence for the valves and allows for the maximum current applied to the EMVT's to be adjusted if necessary.

Data collection for offline processing is recorded by two systems. The cyclic data is recorded using an FEV Combustion Analysis System (CAS) which records cylinder and manifold pressures along with engine angle, measured injection and ignition current, lambda, valve position, and other engine data at a 0.1 CAD resolution. The averaged data from all recorded cycles such as air flow, fuel consumption, emissions composition, and various temperatures including intake, exhaust, coolant, oil, and cylinder head temperature are recorded using the A&D ADAPT system. This data is recorded at a sampling rate of 10 Hz and averaged for the complete measurement set. This data is used for post-processed analysis of the combustion process.

The dSPACE ControlDesk computer is used to record results which were calculated

on the processor and FPGA modules of the MABX II. This data is used for the development and evaluation of the control applications running on the MABX II. The extraction of values from the FPGA to the MABX processor is limited to prevent overruns. This does not affect the operational speed of the FPGA but it does limit the data collected from the FPGA.

### **2.2.3 Rapid Ignition System**

The rapid ignition system (RIS) is designed for providing the ability to have a spark begin immediately after the ignition trigger signal is received. To achieve such a fast response, a high-frequency current system is utilized. Traditional ignition systems utilize stored magnetic energy by charging a coil which is used to provide a spark between the gap of the plug. However, these conventional ignition systems require significant charging time (typically 3-10 ms) when compared to the desired control action speed needed here. In the RIS, the high frequency current supplied to the primary coil from the power supply unit generates an alternating magnetic field, and the field yields alternating high voltage on the secondary coil immediately. Thus the RIS does not require charging time, allowing for the spark to begin within approximately 5-10  $\mu\text{s}$ , which is approximately 1000 times shorter than that of conventional systems which have charging times between 1500-4000  $\mu\text{s}$  [169]. The RIS has sufficient bandwidth for the proposed control strategy.

## **2.3 Summary of chapter**

This chapter provided the details of the two experimental SCRE's used for the data generation and experimental testing of the developed controllers in this thesis. The details on the operating conditions and engine control systems and data acquisition systems have been presented. The main difference between the two SCRE's is the displacement volumes and the fueling systems. With one engine having a port-fuel

injection system and the other having direct injectors. The engine controllers on both engines are the same dSPACE MABX II prototyping ECU allowing for testing of controllers at both locations.



## Chapter 3

# In-Cycle HCCI Engine Control using Rapid Ignition System <sup>1</sup>

This chapter focuses on the design and implementation of a high speed spark actuator to implement an in-cycle HCCI combustion controller. The main objective of this controller is to affect the combustion process within the same combustion cycle. To accomplish this high speed FPGA based engine models and rapid actuators to quickly intervene into the combustion process are used. To improve HCCI combustion stability, large cyclic variations need to be reduced. The cyclic variability of  $CA_{50}$  is used. Figure 1.3 shows the experimentally measured cylinder pressure signal of three consecutive cycles. Cycle 1 is typical of a standard cycle with a  $CA_{50}$  of  $12^\circ$  CA aTDC. It is then followed by cycle 2 which can be considered a incomplete combustion with a very late combustion phasing. Then, due to the incomplete combustion, residual fuel is transferred to the next cycle (cycle 3) through internal EGR. As the combustion phasing is very late in cycle 2, the in-cylinder temperature increases which increases the temperature of the exhaust gas transferred to cycle 3. There is also the possibility that during the NVO recompression a portion of the residual fuel ignites (as seen in cycle 3) and leads to a further temperature increase of the residual exhaust gas. Additionally, combustion during the NVO recompression increases the pre-reactions leading to an increase in  $H_2O_2$  formation. This increase

---

<sup>1</sup> This chapter is based on [8, 9]

in temperature and combustion radicals leads to an early combustion phasing with a high pressure rise rate as in cycle 3. An early combustion phasing is not desired as the high pressure rise rate leads to increased combustion noise and possible engine damage [66, 67]. Overall, high cyclic variation of combustion also tends to reduce thermal efficiency and increase exhaust emissions [68].

The return map for the combustion phasing,  $CA_{50}$  is shown in Figure 1.4. A return map is used to show the relationship between the combustion phasing of the current cycle,  $CA_{50}(i)$ , and of the next one,  $CA_{50}(i + 1)$ . In stable operation, two consecutive cycles are not correlated so the return map would show random scatter around the combustion phasing mean and the spread of the data points represents the stochastic variation from cycle to cycle [49]. However, when a distinct pattern or branching can be seen on the return map, as in Figure 1.4, a coupling or relation between cycles exists. To effectively stabilize combustion the spread of the data points and distinct ‘V’ should be reduced.

In previous work, the cyclic variation of HCCI has been reduced by preventing the early combustion following a misfire (cycle 3 in Figure 1.3) by using direct water injection to cool the trapped exhaust gas which retards combustion phasing back to the desired value [3, 7, 10]. However, this control strategy leaves cycle 2 unaffected and the engine experiences a misfire resulting in higher emissions and lower efficiency. The use of water injection adds the requirement for a second injection system and the requirement for a high quality water source for implementation in production vehicles [75]. To prevent a complete misfire during cycle 2 in Figure 1.3 a spark interaction is presented in this chapter. This spark will be used only when necessary. This spark will not be active during ‘normal’ HCCI combustion to help preserve the  $NO_x$  benefits of HCCI combustion by

keeping the combustion temperature low. The rapid ignition system presented in Section 2.2.3 provides a method to provide a rapid intervention to the combustion process in the means of adding additional energy through a spark to help initiate the auto-ignition process. The following section presents the development of the high speed in-cycle control strategy used to help stabilize the combustion process.

### 3.1 Pressure based control strategy

The high cyclic variability of HCCI combustion can be seen when examining the range of in-cylinder pressure of 200 cycles at a specified crank angle as seen in Figure 3.1. In this figure, the motoring pressure (cylinder pressure without combustion) can be seen where all the cycles overlap when late combustion occurs. This difference between measured cylinder pressure compared to the baseline motoring pressure provides a method to determine if auto-ignition has started.

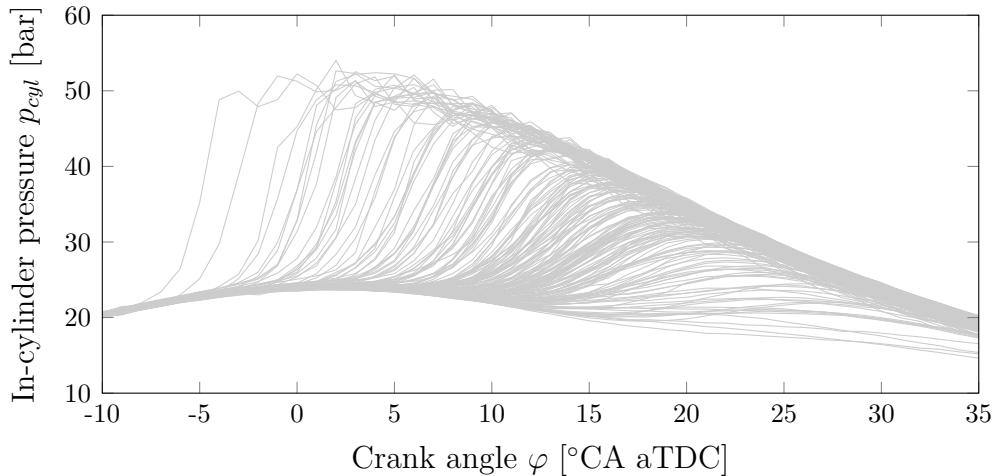


Figure 3.1: Cyclic variability in the cylinder pressure during HCCI combustion at  $n = 1500$  1/min, IMEP = 4.0 bar

The large variation in cylinder pressure presented in Figure 3.1 shows that at a given crank angle there is a large variation in cylinder pressure which is typical of

HCCI combustion. By selecting a specific CA and then examining the correlation of the in-cylinder pressure with the combustion phasing of that cycle are shown in Figure 3.2. When considering the correlation between cylinder pressure and the combustion phasing of that cycle at an engine angle before TDC ( $-10$  or  $-5^\circ$  CA aTDC) is it very difficult to observe anything other than cycles with very early combustion phasings. This is not useful for spark control where distinguishing late cycles from early ones is the goal. However, when considering the cylinder pressure after  $0^\circ$  CA aTDC a clear correlation between the combustion phasing and the cylinder pressure is apparent. This has limited applications for the spark to have maximum impact on the combustion it should be as early as possible. At  $5^\circ$  CA aTDC a clear distinction between cycles with a combustion phasing later than  $10^\circ$  CA aTDC and earlier than  $10^\circ$  CA aTDC can be made. These cycles ( $CA_{50} > 10^\circ$  CA aTDC) are considered as having a late combustion phasing and spark support is needed.

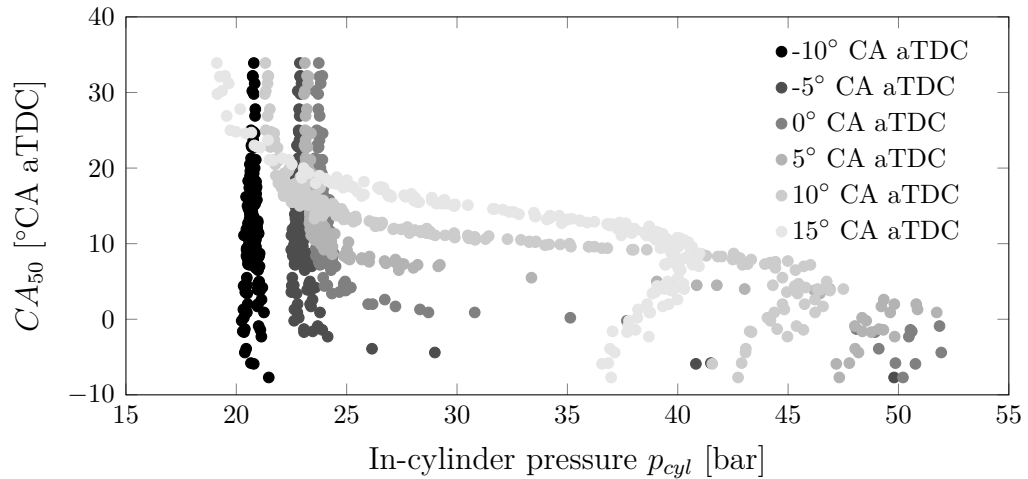


Figure 3.2: Correlation between upcoming combustion phasing and cylinder pressure measured at a specified crank angle aTDC.  $n = 1500$  1/min, IMEP = 4.0 bar

At  $5^\circ$  CA aTDC, if the cylinder pressure has not exceeded 26 bar it is assumed that the combustion will be late or misfire and a spark is required. There will be

some cycles when the spark will be activated even though the ignition timing is only slightly retarded at around  $12^\circ$  CA aTDC, however, it is important to actuate the spark as early as possible to ensure enough time for the flame to propagate through the fuel/air mixture.

### 3.2 Heat release based control strategy

The increase in cylinder pressure due to the start of the combustion process, can also be shown using the accumulated heat release. The advantage of using heat release compared with measured pressure is that a very large change in heat release values are observed at the start of the combustion process. So, the effect of signal noise can be reduced. Figure 3.3 shows a similar result to the pressure based correlation (Figure 3.2) but now the heat release at a specified angle aTDC is correlated to the combustion phasing of that cycle.

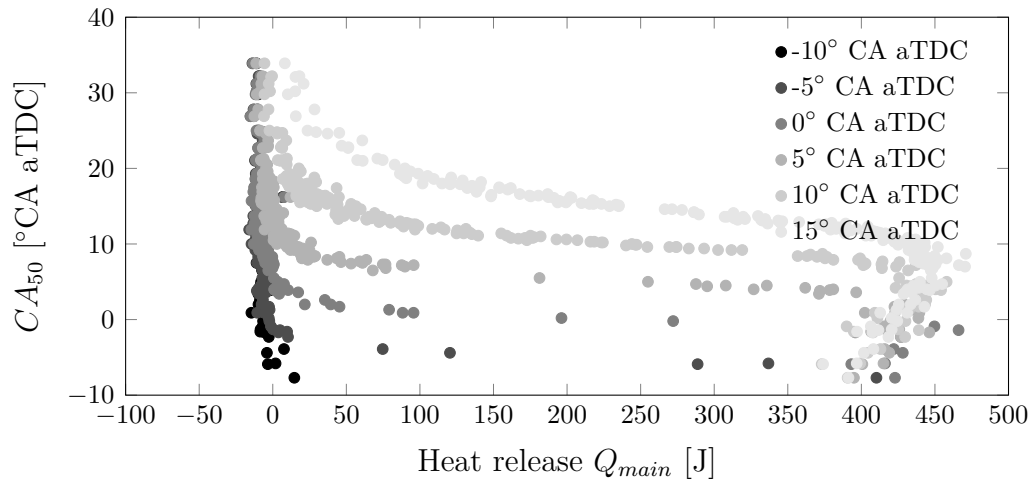


Figure 3.3: Correlation between upcoming combustion phasing and cumulative heat release at a specified angle aTDC.  $n = 1500$  1/min, IMEP = 4.0 bar

Again angles before  $0^\circ$  CA do not provide any information on the current state of the combustion. Similar to the pressure correlation,  $5^\circ$  CA aTDC is selected as the timing when to check if combustion has begun. If not the spark is activated.

Only the correlation at  $5^\circ$  CA aTDC near the transition area of 0 J heat release is shown in Figure 3.4. A heat release threshold of 0 J is selected as a physical representation of the start of combustion. Below this threshold a combustion phasing of approximately  $15^\circ$  CA aTDC or later is predicted.

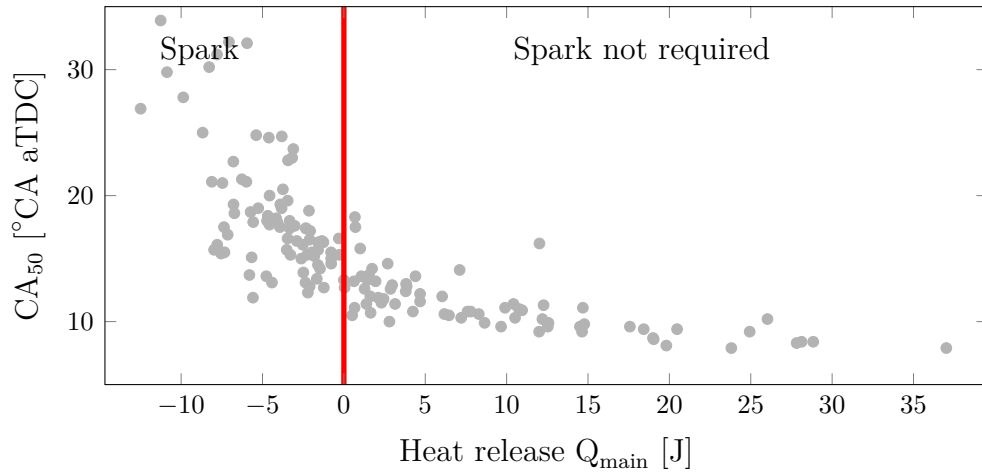


Figure 3.4: Selected threshold of 0J heat release for start of combustion using correlation between upcoming combustion phasing and cumulative heat release at  $5^\circ$  CA aTDC

### 3.3 Ignition timing constraints

The above two control strategies are then implemented on the FPGA side of the engine controller as shown schematically in Figure 3.5. The proposed feedforward control strategy takes the cylinder pressure or cumulative heat release at an engine angle of  $5^\circ$  CA aTDC and compares it to the specified threshold (discussed in the previous section). If it is determined that the combustion has yet to begin the rapid ignition system (RSI), that was presented in Section 2.2.3, is activated for the spark to begin.

With HCCI combustion the cylinder state is very close to the auto-ignition point of

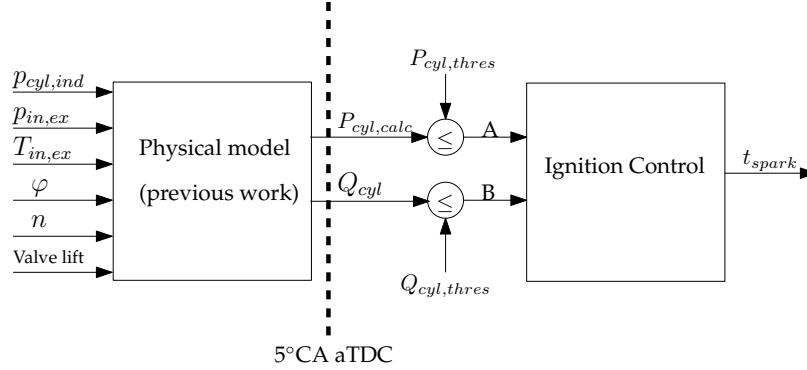


Figure 3.5: Feedforward spark control structure

the fuel and air mixture even in cases of late combustion or misfire. Therefore only a small amount of additional energy is required for the auto-ignition process to begin. However, after the piston reaches TDC the expanding cylinder volume lowers the cylinder pressure and increases the amount of additional energy required for combustion to begin. Therefore, it is desirable to have the spark occur as early as possible.

The FPGA is used as it is capable of in-cycle control. Using the rapid calculation rate of the FPGA allows for the calculation of the entire gas exchange process including heat release using the existing model that has been developed in [84]. The gas exchange process is calculated on the FPGA every  $0.1^\circ$  CA which provides the heat release rate in  $3.525 \mu\text{s}$  [84]. Therefore the engine can be operated up to 4728 rpm before the resolution of the heat release rate can not be maintained within  $0.1^\circ$  CA. As HCCI is a low-mid speed combustion method that is generally limited to around 3000 rpm the proposed control strategy can be applied to the entire HCCI operating range [108].

The total calculation of the heat release and the controller comparison is completed with a delay of only 301 FPGA samples (or  $3.76 \mu\text{s}$  for the FPGA used in this study). When combined with the high speed of the RIS ignition system the spark begins in  $5\text{-}10 \mu\text{s}$  ( $0.045\text{-}0.09^\circ$  CA at 1500 1/min) after the engine reaches

5° CA aTDC. Therefore when using an FPGA based model and RIS system the spark can occur when the cylinder state is close to auto-ignition and provide the maximum time for the fuel to burn before the exhaust valve opens. Using the proposed control strategy with traditional ignition systems (which require around 3-10 ms for coil charging) would significantly reduce the amount of time for the fuel to burn as shown in Figure 3.6.

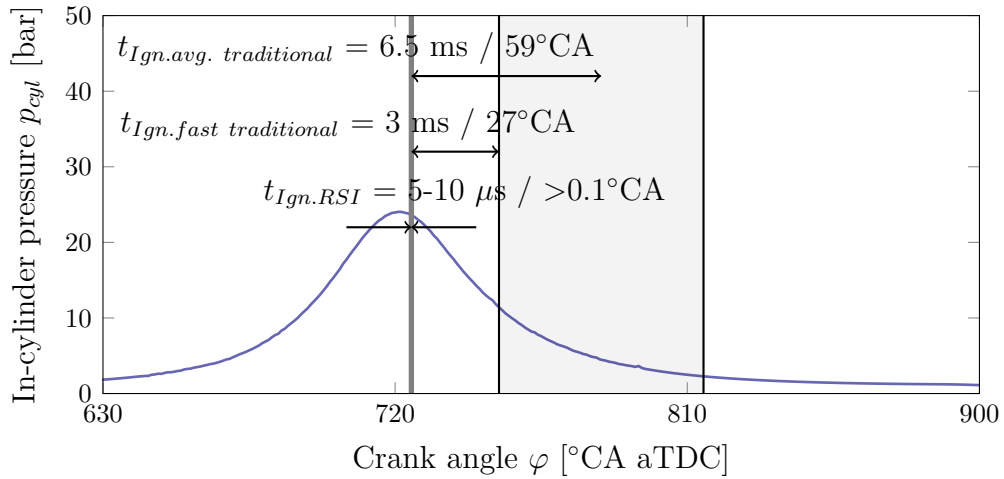


Figure 3.6: Ignition discharge time based on control signal at 5° CA aTDC. Gray area represents variation in traditional ignition systems

When using a FPGA the total number of logical gates and memory on the board are limited. So when additional controllers are implemented it is important to determine the required additional resources. The proposed controller consumes under 1 % of the available flip-flops and look-up tables on the Xilinx FPGA while the gas exchange model uses approximately 65% [84].

### 3.4 Combustion stability

The proposed feedforward controller was tested, using both the in-cylinder pressure and heat release as control input, on the RWTH SCORE (defined in Section 2.2). The goal is to reduce the cyclic variation as shown in Figure 1.3. The performance of the



controller in stabilizing the combustion phasing is shown in Figure 3.7. The CA50 for 1000 consecutive cycles are plotted with the controller being activated after 500 cycles. The variation in combustion phasing after cycle 500 is significantly reduced when the controller is enabled as shown by the reduction in standard deviation of CA50 from 6.89 to 5.3 ° CA or a reduction of 23%.

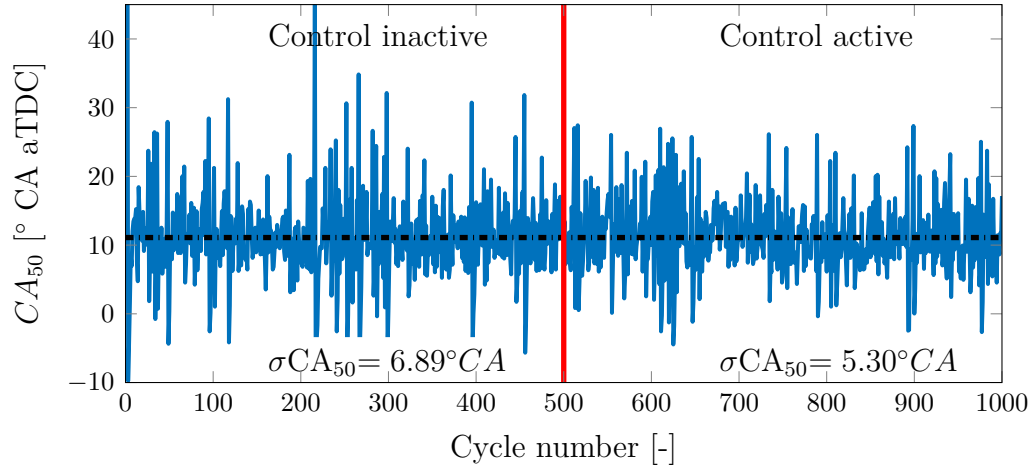


Figure 3.7: Combustion phasing,  $CA_{50}$ , stability improvement with in-cycle controller based on heat release at 5° CA aTDC. The controller is activated after 500 cycles. Here,  $n = 1500$  1/min, NVO= 133° CA, IMEP = 4.0 bar.

During uncontrolled operation there are eight cycles with a combustion phasing later than 30° CA aTDC (considered a misfire). After the controller is activated no cycles with a combustion phasing greater than 30° CA aTDC occur. This demonstrates the controller eliminates misfire cycles. During the controlled operation, the most extreme retarded combustion phasing is 27.4° CA aTDC which occurs at cycle 516. Since the spark is fired at 5° CA aTDC it takes 22° CA for 50% of the fuel to burn. This is still a long delay which helps to justify the need in having the spark as early as possible.

Figure 3.7 also shows that the number of early combustion phasing cycles have been greatly reduced even though the spark does not directly effect these cycles.

This reduction in cycles with a very advanced combustion phasing is attributed to the fuel being burnt in the desired cycle instead of being transferred to subsequent cycles. Thus the proposed control method reduces the coupling between subsequent cycles.

The torque (IMEP) produced as a function of cycle for the controller inactive and then activated is shown in Figure 3.8. With the spark controller inactive, there are numerous cycles with a low IMEP having a very late and inefficient combustion. With the spark controller activated there is a significant reduction in the number of cycles with a very low IMEP. The controller action results in a reduction of the standard deviation of IMEP from  $\sigma_{\text{IMEP}} = 0.312$  bar to  $\sigma_{\text{IMEP}} = 0.222$  bar. This represents a 28.9% reduction in the standard deviation of IMEP showing a significant improvement in the combustion stability of HCCI.

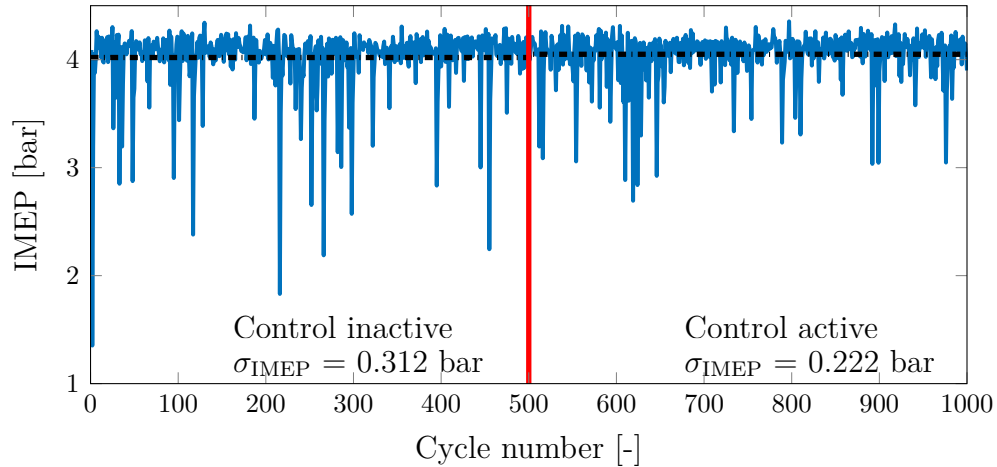


Figure 3.8: IMEP stability improvement with in-cycle controller based on heat release at  $5^\circ$  CA aTDC. The controller is activated after 500 cycles. Here,  $n = 1500$  1/min, NVO =  $133^\circ$  CA, IMEP = 4.0 bar.

The spark controller effectiveness to decouple subsequent cycles is depicted in Figure 3.9. When the controller is inactive (red squares) the characteristic ‘V’ shaped spread of data (presented in Figure 1.4) is observed. When the controller is acti-

vated (blue circles) the arms of the return map are significantly reduced down to a more stochastic circular distribution of data. The developed controller is effective in eliminating the cycles with an extremely retarded combustion phasing which then prevents subsequent cycles with a very advanced combustion phasing. However some cyclic coupling does remain.

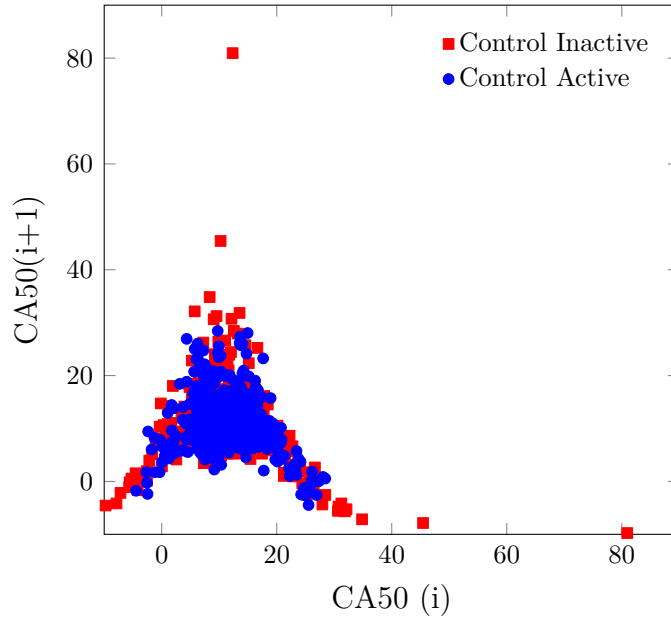


Figure 3.9: Effect of tested spark controller based on heat release on the combustion phasing return map

A summary of the tested control strategies at various operating points is listed in Table 3.1. For all the operating points tested an increase in the average IMEP was found for active control. The control strategy improves the average combustion efficiency over 500 cycles by delivering more power at a constant fueling. Very little change in the average combustion phasing is observed. The controller is preventing both very retarded and advanced combustion phasing cycles.

The standard deviation of both IMEP and combustion phasing are significantly reduced when the controller is activated. This reduction demonstrates the effectiveness of the controller to stabilize HCCI combustion and will result in a smoother engine operation. The reduction in average peak pressure rise rate  $\overline{(dp/d\varphi)}$  is also shown

Table 3.1: Combustion stability improvement due to feedforward rapid spark controller. Control input of cylinder pressure (pres) and heat release (HR) are both shown.

Control input	IMEP [bar]		CA <sub>50</sub> [°CA]		$\Delta\sigma_{\text{IMEP}}$ [%]	$\Delta\sigma_{\text{CA}_{50}}$ [%]	$\Delta\overline{dp/d\varphi}$ [%]	Interactions [cycles]
	Off	On	Off	On				
Pres	2.86	2.89	9.4	9.6	-21.2	-12.9	-1.0	117/500
Pres	4.03	4.04	11.0	11.1	-13.8	-2.5	-1.4	25/500
HR	2.71	2.72	9.4	9.4	-5.0	-1.9	-1.6	71/500
HR	3.92	3.96	11.0	10.8	-25.2	-34.3	-0.8	121/500
HR	4.02	4.05	11.1	11.1	-28.9	-23.0	-1.9	86/500

in Table 3.1. This improvement is relatively small since the proposed controller does not directly reduce the pressure rise rate and is only active for a small number of the 500 cycles.

### 3.5 Emission benefits

In general, the NO<sub>x</sub> benefits of HCCI occur due to the rapid multi-site low temperature combustion. This is reduced when a spark is used to initiate the combustion process. However, with the addition of a spark to an HCCI engine results in an improvement in combustion stability. Since the spark creates a single point of ignition that propagates through the cylinder and increases the local gas temperature to above the NO<sub>x</sub> formation temperature. The proposed HCCI control strategy is expected to increase NO<sub>x</sub> emissions. The proposed control strategy should reduce the uHC emissions related to misfire by providing a spark to help the combustion process begin, while preserving the benefit of not having a spark when not needed. The proposed ignition control strategy was compared with the cases of no spark

interaction (HCCI) and having a spark every cycle at  $5^\circ$  CA aTDC spark assist compression ignition (SACI). For each ignition strategy three measurements were performed for 30 s then averaged.

Figure 3.10 shows that the expected increase  $\text{NO}_x$  and NO emission was observed when the spark was always used. The nitrogen oxide emissions are similar between the HCCI case and the HCCI controller operation. This result is expected as the spark is only used in approximately 10% of cycles and in these cycles the combustion occurs very late and as the cylinder gas is expanding which results in a combustion temperature below the  $\text{NO}_x$  formation temperature.

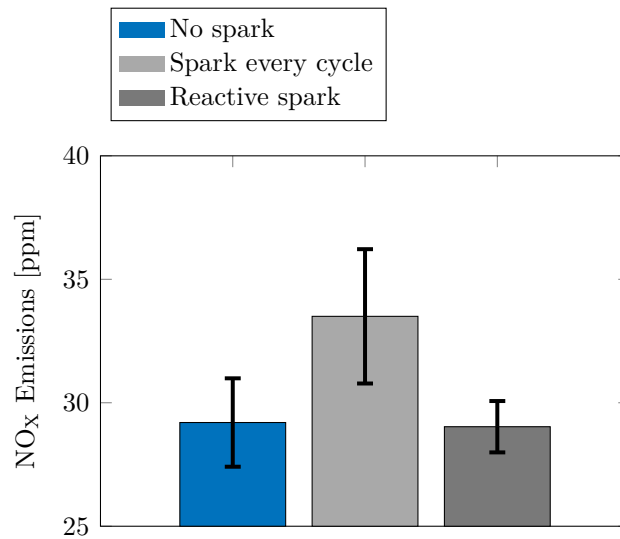


Figure 3.10: Comparison of nitrogen oxide emissions between control strategies. Error bars represent two standard deviations.

The unburnt hydrocarbon emissions for these cases can be seen in Figure 3.11. When the controller is enabled the lowest HC emissions are observed. Compared the HCCI case (No spark) an improvement of 6.6% is achieved. However, the case where the spark occurs every cycle actually has the highest HC emissions. When a spark is always a hotter more complete combustion for every cycle is expected. However, the

very late spark timing leads to SACI combustion which often has a very high cyclic variability at very late spark timings [170, 171]. This high cyclic variability leads to unstable combustion and increased HC emissions. Similarly, the carbon monoxide emissions are also increased in the SACI case as seen in Figure 3.12 which matches the HC emission results.

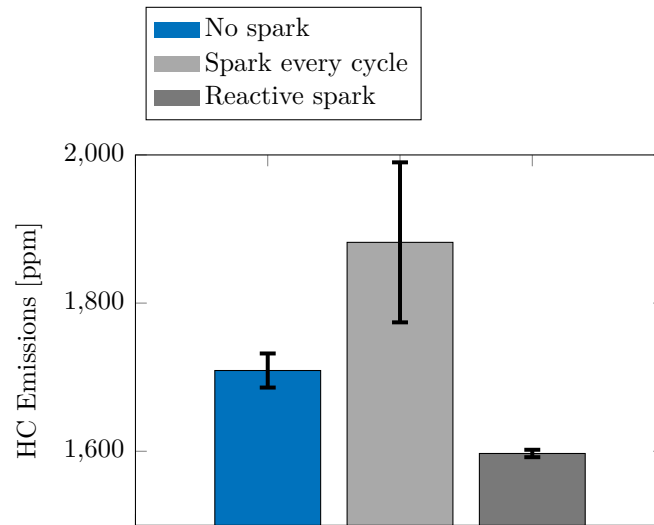


Figure 3.11: Comparison of unburnt hydrocarbon emissions between control strategies. Error bars represent two standard deviations.

The high cyclic variability of the SACI case as seen by the large standard deviation of the emissions was observed. This indicates that when the spark was used every cycle the combustion stability is decreased compared to the pure HCCI operating condition. This decrease in combustion stability is attributed to an increase in the amount of fuel burnt using flame propagation compared to the autoignition of HCCI. The high experimental uncertainty (large standard deviation) is due to the statics of the limited number of tests performed. However, the basic benefit of the proposed control strategy is demonstrated. The impact to the engine emission output, further research is needed to allow for a larger data set. Further tests would allow more conclusive statistical data on the relationship of the spark input on the engine out emissions.

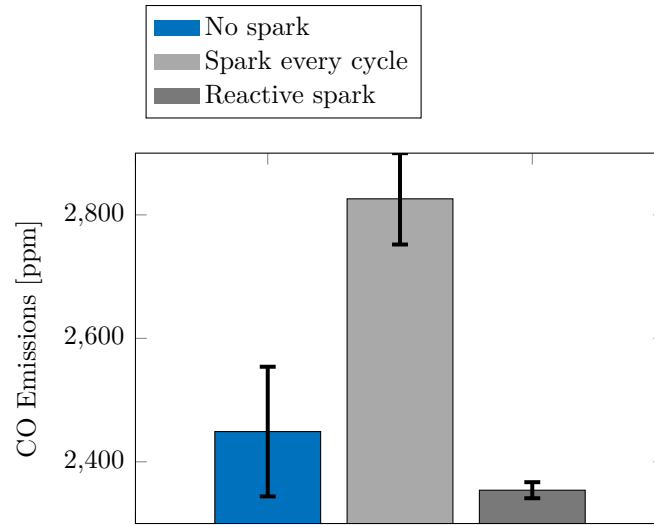


Figure 3.12: Comparison of carbon monoxide emissions between control strategies. Error bars represent two standard deviations.

## 3.6 Summary of chapter

In this chapter, the rapid ignition system was combined with the fast calculation rate of an FPGA based engine controller to experimentally test an ignition controller. This controller utilizes one of cylinder pressure or heat release to determine if the auto-ignition process of HCCI has started. The proposed controller was experimentally and was able to successfully reduce the standard deviation of combustion phasing and IMEP at multiple HCCI operating points. The controller required a spark in approximately 10% of cycles at the operating points tested. This led to nitrogen oxide levels similar to the pure HCCI case while showing an improvement of 6.6% in unburnt hydrocarbon emission and a reduction of 4.0% of Carbon Monoxide emissions over the pure HCCI operation.



# Chapter 4

## Combined In-Cycle and Cycle-To-Cycle HCCI Engine Control <sup>1</sup>

In-cycle control approaches, like the rapid ignition controller described in Chapter 3, utilize a control intervention within a cycle and have shown great potential to stabilize HCCI. The described approaches have been implemented using a model-free, rule-based control. In addition to the work presented in Chapter 3, the potential of in-cycle control [88], such as water injection [3, 10], and actively controlled glow plugs [89] have been demonstrated. In many cases, linear single input single output (SISO) control is used to stabilize the combustion. The chosen operating variables show a great potential to stabilize the combustion, despite only using one input variable to the in-cycle controller. For enhanced controller performance and to fully stabilize HCCI it is expected that both the cycle-to-cycle and in-cycle timescales need to be considered for multiscale control. In this section, the modeling approach of a split autoregressive process (Figure 4.1) is utilized [10].

The split autoregressive process describes the autoregressive character of HCCI by separating the process into the main combustion (MC) and intermediate compression (IC). Two separate feedback loops are used due to the timescale separation. For each individual cycle  $i$ , the behavior of the sub-processes can be described depending on

---

<sup>1</sup> This chapter is based on [5, 10]

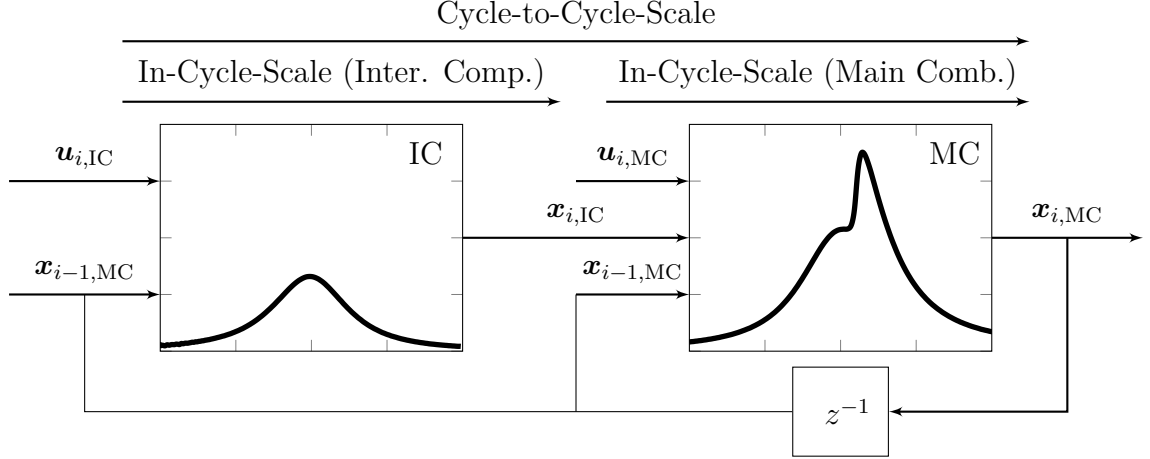


Figure 4.1: Multi-scale control strategy for HCCI combustion based on Wick et. al [10]

certain feedback variables  $\mathbf{x}_{i,MC}$ ,  $\mathbf{x}_{i,IC}$  such as CA50. Cyclic input variables for both timescales  $\mathbf{u}_{i,MC}$ ,  $\mathbf{u}_{i,IC}$  such as water- or fuel injection duration are considered.

Thus, in this section the potential of a controller on multiple timescales will be presented.

In the following section, the feedback variables  $\mathbf{x}_{i-1}$  from each cycle are discussed such that an adequate representation of the process is ensured. Furthermore, operating variables  $\mathbf{u}_i$  to control the process are identified.

## 4.1 Selection of Feedback Variables

### 4.1.1 Main Combustion (MC)

For the MC, the feedback variables  $\mathbf{x}_{i,MC}$  the accumulated heat release  $Q_{MC}$  and the combustion phasing CA50 (eq. 4.1) have been selected as:

$$\mathbf{x}_{i,MC} = \begin{pmatrix} Q_{MC} \\ CA50 \end{pmatrix} \quad (4.1)$$

The heat release  $Q_{MC}$  results from the integral of the heat release rate, which can be determined using the 1<sup>st</sup> law of thermodynamics based on the measured in-cylinder pressure  $p(\phi)$ . Thus,  $Q_{MC}$  is dependent on the thermo-chemical in-cylinder

state. In addition, CA50 is found to provide information on the combustion timing and thus on the pressure trace of the entire cycle. In several studies, CA50 is used to represent the chemical or thermodynamic state. For example, the influence of the mixture composition and the intake temperature at various  $\text{SOI}_F$  to maintain a constant CA50 have been investigated [172]. The influence of the in-cylinder state has also been found to have a significant impact on the following IC and MC. Thus CA50 was used as feedback and the target variable for a linear MPC approach [173]. In this work, it is assumed that there is a sufficient representation of the in-cylinder state by the chosen feedback variables heat release  $Q_{\text{MC}}$  and combustion phasing CA50.

#### 4.1.2 Intermediate Compression (IC)

For the IC, the feedback variables  $\mathbf{x}_{i,\text{IC}}$  are calculated from the measured cylinder pressure and are defined as follows:

$$\mathbf{x}_{i,\text{IC}} = \begin{pmatrix} Q_{\text{IC}} \\ m_{\text{F,res}} \\ m_{\text{W,IC}} \end{pmatrix} \quad (4.2)$$

The heat release during IC,  $Q_{\text{IC}}$ , influences the resulting in-cylinder state. Combustion during IC changes the chemical and thermodynamic state resulting in a change to the measured in-cylinder pressure trace, which changes  $Q_{\text{IC}}$ . This change in  $Q_{\text{IC}}$  has a strong effect on the subsequent MC, so that  $Q_{\text{IC}}$  should be taken into account for the developed control approach. On its own  $Q_{\text{IC}}$  is not sufficient to represent the in-cylinder state. Therefore, the residual fuel mass  $m_{\text{F,res}}$  is also chosen as a controller input variable of the IC. Since the unburnt residual fuel mass trapped in the combustion chamber causes an enrichment of the mixture if the fuel injection duration remains constant. The amount of the residual fuel is one possible variable to represent the mixture composition and thus the in-cylinder state after the gas exchange. Additionally, the water injection during IC,  $m_{\text{W,IC}}$ , is included as a controller input

variable for the MC. If water is injected during the IC, it cools down the mixture and has a direct influence on the thermodynamic and chemical cylinder state before MC.

### 4.1.3 Selection of Operating Variables

In addition to the feedback variables of the process  $\mathbf{x}_{i-1}$  and  $\mathbf{x}_i$ , the in-cylinder state after IC is also dependent on cycle individual operating variables  $\mathbf{u}_i$ . The selected operating variables are derived from experience and previous work [10, 82]. These variables have a strong influence on the thermodynamic and chemical in-cylinder state and are:

$$\mathbf{u}_{i,MC} = \begin{pmatrix} t_{F, MC} \\ m_{W, MC} \end{pmatrix} \quad (4.3)$$

$$\mathbf{u}_{i,IC} = m_{W, IC} \quad (4.4)$$

for the MC and for the IC.

Water injection during IC  $m_{W, IC}$  and MC  $m_{W, MC}$  can be used to effectively delay the start of combustion through both the cooling and charge-diluting effects [10]. This can prevent early combustion phasing in the MC and reduces the risk of unwanted self-ignition during IC.

Adjusting the fuel injection duration  $t_{F, MC}$  can be used to account for any residual fuel in the cylinder and is used to keep the total fuel mass for each cycle constant in order to prevent mixture enrichment and the resulting high pressure gradient during combustion [82]. Previous work indicated that it is necessary to measure these operating variables  $\mathbf{u}_i$  and their interaction with the above defined feedback variables on multiple timescales. Those variables are used to represent the thermo-chemical charge state but are much less complex than the full charge state to enable real-time computation.

The variables selected are found to be a sufficient representation of the complex HCCI combustion process and are real-time capable.

## 4.2 Design of Multiple Input Multiple Output (MIMO) Controller using Artificial Neural Network (ANN) based Inverse Models

Data driven models have been used to model complex non-linear systems and their success has been used to provide the motivation to investigate them in this work.

Overall,  $n_k = 7300$  cycles were recorded with 75 % being used for training of the ANN. The remaining 25 % of the data serves as a validation data set. For example, data driven models are useful to demonstrate novel measurement methodologies [174]. This model type is well suited to capture the nonlinear process behavior. Previous work provided a range of network sizes and network typologies for identifying the HCCI process behavior to achieve a promising model and control results [174]. Using the same topology, multilayer perceptrons with 3 hidden layers and 13 neurons each are applied in the next section.

### 4.2.1 ANN Modeling Approach

A nonlinear control policy  $f$  is fit to the data through inversion of the controlled system. To achieve this, a ANN is used to generate the inverse process models, where the cause-effect-relationship is reversed. Then the required control interventions can be determined from the desired behavior of the system. This means that the required operating variable interventions  $\hat{\mathbf{u}}_i$ , are estimated with a function  $f$  on the basis of the current state, represented by the feedback variables  $\mathbf{x}_{i-1}$ , and the required target state of the next cycle  $\mathbf{x}_{i,\text{target}}$  as:

$$\hat{\mathbf{u}}_i = f(\mathbf{x}_{i-1}, \mathbf{x}_{i,\text{target}}) \quad (4.5)$$

During the ANN model training a cost function  $J$  is minimized, based on the mean square error (MSE). So the following optimization problem is solved:

$$\min(J) = \min\left(\frac{1}{n_k} \sum_{k=1}^{n_k} \|(\overline{\mathbf{u}}_{i,k} - \hat{\mathbf{u}}_{i,k})\|^2\right) \quad (4.6)$$

$$\min(J) = \min\left(\frac{1}{n_k} \sum_{k=1}^{n_k} \|(\overline{\mathbf{u}}_{i,k} - f(\overline{\mathbf{x}}_{i-1,k}, \overline{\mathbf{x}}_{i,k}, W, B))\|^2\right) \quad (4.7)$$

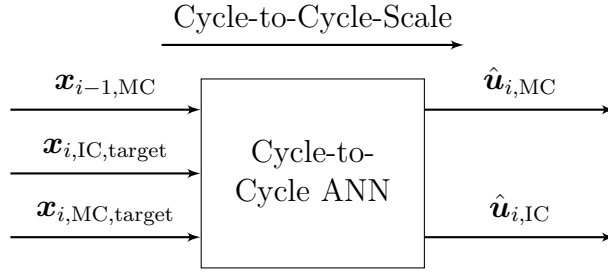
Where  $k$  is the  $k$ -th training data set, the overlined variables are normalized for the training process of the ANN to account for different orders of magnitude.  $W$  is the weight matrix and  $B$  is the bias matrix of the ANN. This means that during the training process of the ANN the matrix  $W$  and the matrix  $B$  are tuned to achieve a minimum MSE over all training data sets  $n_k$ . After training the ANN, the solution of the optimization problem is given by the ANN. Thus, the application of these models in control algorithms leads to an explicit NMPC based on inverse process models. Such controllers have already been used in other applications, such as diesel airpath control [175] or the control of chemical reactors [176].

Explicit MPC does not need real-time optimization on the engine control unit (ECU) and requires only low computational power [176]. The explicit MPC based on the ANN model, has then been ported to real-time hardware.

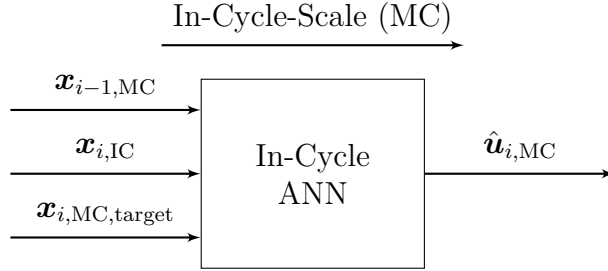
To address the cyclic coupling of the HCCI process (Figure 4.1) in the model approach, two models are trained accordingly, for the cycle-to-cycle and the in-cycle timescales respectively. The structure of the applied models, including the input and output variables, is shown in Figure 4.2.

Clearly the control model inputs must be available at the appropriate time in the real-time system. Thus for the cycle-to-cycle model, the execution takes place after the end of the expansion (190 °CA aTDC), for the in-cycle model after the end of the IC phase (412 ° CA aTDC). For the cycle-to-cycle model only the feedback variables of the last cycle  $\mathbf{x}_{i-1,MC}$  are available at execution time are shown in Figure 4.2a.

In the cycle-to-cycle model, the required operating variables including  $\hat{\mathbf{u}}_{i,IC}$  (eq. 4.4) and  $\hat{\mathbf{u}}_{i,MC}$  (eq. 4.3) are required to achieve a desired target state of the



(a) Cycle-to-cycle ANN for estimation of operating variables on the cycle-to-cycle scale  $\hat{\mathbf{u}}_{i,IC}$ ,  $\hat{\mathbf{u}}_{i,MC}$



(b) In-cycle ANN for estimation of operating variables on the in-cycle (MC) scale  $\hat{\mathbf{u}}_{i,MC}$

Figure 4.2: Inverse model structure to describe the HCCI cycle-to-cycle and in-cycle (MC) scale according to the split autoregressive process

entire cycle  $\mathbf{x}_{i,target}$ . The target state of the MC  $\mathbf{x}_{i,MC,target}$  (eq. 4.1) and IC  $\mathbf{x}_{i,IC,target}$  must be defined as model inputs.

Since the feedback variables of the IC  $\mathbf{x}_{i,IC}$  are already known after the IC and can therefore be included in the in-cycle ANN for the MC as an additional input. The target state for the MC  $\mathbf{x}_{i,MC,target}$  must still be set and since the IC is completed, the operating variables  $\hat{\mathbf{u}}_{i,IC}$  are available. These MC variables affect the MC so they are used as an input into the model. For this purpose, they are included in the feedback variables from the IC  $\mathbf{x}_{i,IC}$  in order to consider them. Using the aforementioned variables, only the required operating variables for the MC  $\hat{\mathbf{u}}_{i,MC}$  are estimated.

No model for the in-cycle control of IC is used since this model would set the operating variables in such a way that a target state after IC is reached without considering the target state of the MC. For example, a cycle-specific operating variable intervention by an IC model can prevent heat release in the IC. Since such an IC

model has no knowledge about the influence of the intervention on the MC. An over-compensation with a negative influence on the next MC could then occur. This could lead to control deviations that cannot be rejected by the in-cycle (MC) model and a lower overall control performance can be expected. Thus an in-cycle model for IC is not used. For the multiscale control concept that is introduced later, the operating variables calculated with the C2C-ANN for MC are discarded and recalculated with additional information using the in-cycle-ANN. This provides a consistent comparison between the applied controlled strategies.

### 4.2.2 ANN Model Results

How the trained ANN output differs on the cycle-to-cycle and in-cycle timescale is examined. As described above, the in-cycle model is expected to have a significant improvement in comparison to the cycle-to-cycle model. To evaluate this, the error  $E$  is defined as the difference between the ANN output  $\hat{\mathbf{u}}_i$  and the measured variables  $\mathbf{u}_i$ .

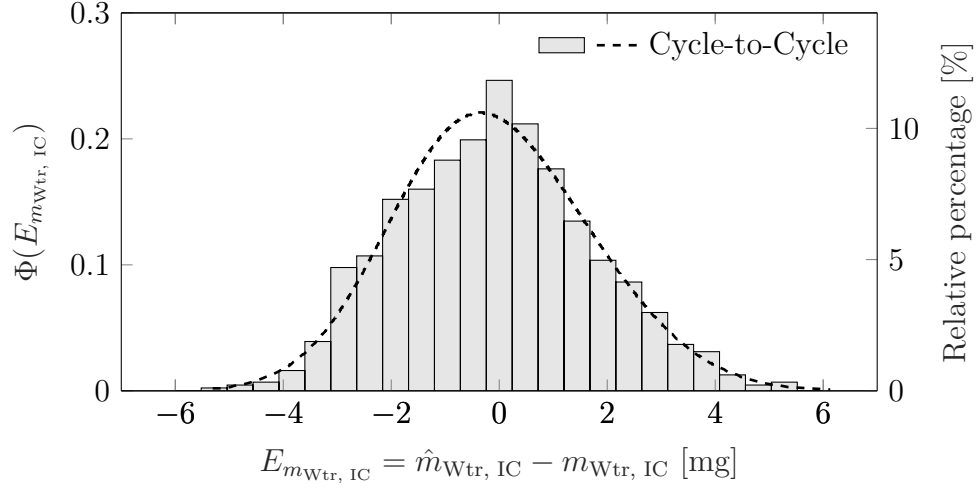
$$\mathbf{E} = \hat{\mathbf{u}}_i - \mathbf{u}_i \quad (4.8)$$

The probability density function (PDF)  $\Phi$  for each element of  $E$  with the following property:

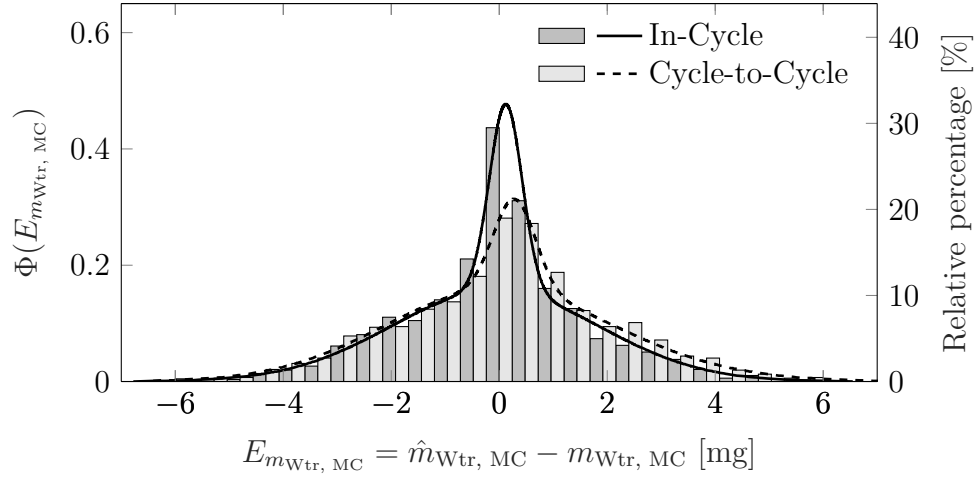
$$\int_{-\infty}^{\infty} \Phi(E) dE = 1 \quad (4.9)$$

The PDF  $\Phi$  as a function of the error  $E$  for all three elements of  $\mathbf{u}_i$  are shown in Figure 4.3. The probability distribution of the error is also displayed as a percentage using the bars. The results for the cycle-to-cycle models are indicated by dashed lines and light bars and for in-cycle models by the solid lines and dark bars respectively. In addition, Table 4.1 lists MSE and the Pearson correlation coefficient (PCC)  $r$  for all operating variables for both model timescales. Both Table 4.1 and Figure 4.3 are based on the validation data set which is not used during the training process.

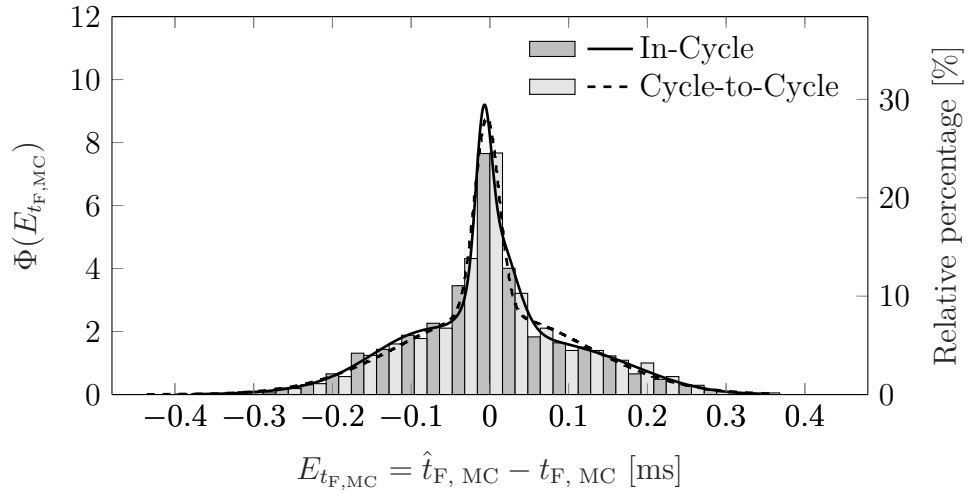




(a) Probability density function of required water pre-injection mass  $\hat{m}_{W, IC}$



(b) Probability density function of required water main injection mass  $\hat{m}_{W, MC}$



(c) Probability density function of required fuel injection duration  $\hat{t}_{F, MC}$

Figure 4.3: Modeling results: Probability density function of model error for cycle-to-cycle and in-cycle scale

Table 4.1: Performance of ANN based cycle-to-cycle and in-cycle inverse process models

Variable	Quality measure	Cycle-to-cycle	In-cycle	$\Delta\%$
$\hat{m}_{W, IC}$	MSE	3.35 mg <sup>2</sup>	-	-
	PCC $r$	0.731	-	-
$\hat{m}_{W, MC}$	MSE	3.73 mg <sup>2</sup>	2.93 mg <sup>2</sup>	-21.5%
	PCC $r$	0.709	0.807	+13.8%
$\hat{t}_{F, MC}$	MSE	0.010 ms <sup>2</sup>	0.011 ms <sup>2</sup>	+6.9%
	PCC $r$	0.710	0.721	+1.6%

For the predicted water pre-injection mass  $\hat{m}_{W, IC}$  (Figure 4.3a), which is only estimated in the cycle-to-cycle model, the PDF shows a characteristic Gaussian distribution. 48.81% of the cycles have a maximum error of 1.2 mg, 75.23% of the cycles are within a window of 2.16 mg. An MSE of 3.348 mg<sup>2</sup> is obtained and the PCC is  $r = 0.731$ .

For the water injection mass during the MC  $\hat{m}_{W, MC}$ , a completely different probability distribution is observed (Figure 4.3b). Near zero, the PDF becomes sharply peaked, which indicates a significant accumulation of the error in this window. Towards the outside, the curve becomes considerably flatter. This behavior is observed for the cycle-to-cycle as well as for the in-cycle model, whereas the peak is more distinctive for the latter. The in-cycle model tends to smaller errors and 60.42% of the cycles are within an error window of 1.2 mg. For the cycle-to-cycle model, only 50.80% is obtained. This results in a reduction in MSE of 21.45% and an increase in PCC of 13.82% in favor of the in-cycle model.

The PDF of the model error for the fuel injection duration  $\hat{t}_{F, MC}$  (Figure 4.3c) shows a similar characteristic as for the water main injection  $\hat{m}_{W, MC}$ , whereas the

peak around zero is even more significant in this case. In contrast to the water injection, there is only a slight difference of the PDF of cycle-to-cycle and in-cycle model for the fuel injection. This means, the required fuel mass can already be predicted quite accurately on a cyclic base. Nevertheless, a better control performance is expected from the in-cycle model, since the overall model performance including the water injection is significantly better. The error window of 0.096 ms contains 72.30% of the cycles for the cycle-to-cycle model and 72.75% for the in-cycle one. For the cycle-to-cycle model, an MSE of 0.0102 ms<sup>2</sup> is obtained, which is slightly better than the MSE of the in-cycle model of 0.0109 ms<sup>2</sup>. However, for the in-cycle model a small increase of the PCC  $r$  of 1.55% is observed.

The results yield the following conclusion: For the cycle-to-cycle model, a relatively large number of operating variables  $\hat{\mathbf{u}}_i$  must be computed using only the feedback variables  $\mathbf{x}_{i-1,MC}$ . This leads to an increased uncertainty in the cycle-to-cycle model. However, it is obvious that the in-cycle model performance for the estimated water injection can be significantly increased compared to the cycle-to-cycle one.

This is attributed to an improved representation of the thermodynamic cylinder state by consideration of the additional information from the IC. Thus, the model error is significantly decreased on the in-cycle scale.

### 4.3 Controller Results and Validation

To test the developed controllers the RWTH SCRE, defined in Section 2.2 is used. Then the validation measurements were performed by splitting a measurement of 995 cycles into 5 intervals of about 200 cycles each. Table 4.2 provides an overview of the distribution of the different control approaches used during the measurement.

Figure 4.4 provides an overview of CA50, IMEP and the heat release ( $Q_{IC}$ ) during IC. As well the applied operating variables, the injected water mass during IC (IC  $m_{W, IC}$ ) and MC ( $m_{W, MC}$ ) and the fuel injection duration ( $t_{F, MC}$ ) from the differ-

Table 4.2: Controller results split into approximately 200 cycles intervals

Phase	Cycle	Abbreviation	Control approach
1	1-197		Controller disabled
2	198-401	C2C-SISO	Linear model free cycle-to-cycle SISO
3	402-620	MS-SISO	Linear model free multiscale SISO
4	621-805	C2C-MIMO	ANN cycle-to-cycle MIMO
5	806-995	MS-MIMO	ANN multiscale MIMO

ent control approaches are compared to the base uncontrolled case. Additionally, the indicated thermal efficiency  $\eta_i$  is noted. The intervals for each controller evaluation, as defined in Table 4.2, are indicated by the dashed black lines. In addition, the color of each interval varies from a light gray for the uncontrolled case to black for the MS-MIMO controller. Furthermore, the CA50 return map for all control approaches, in the same color scheme, is shown in Figure 4.5.

### Phase 1: Controller Disabled:

During the uncontrolled case in phase 1, there is a strong variance of the CA50 with a maximum value of 27.45 °CA aTDC observed and a minimum value of -10.95 °CA aTDC. Likewise, IMEP shows significant outliers, which indicate the combustion instability of the selected operating point. In addition, the typical CA50 return map can be seen in Figure 4.5 for each control approach as a reference in the background.

### Phase 2 and 3: Reference SISO-Controllers:

For the evaluation of the MIMO control concepts (discussed in Phase 4 and 5), the model-free rule-based SISO controls from preliminary work are used for comparison.

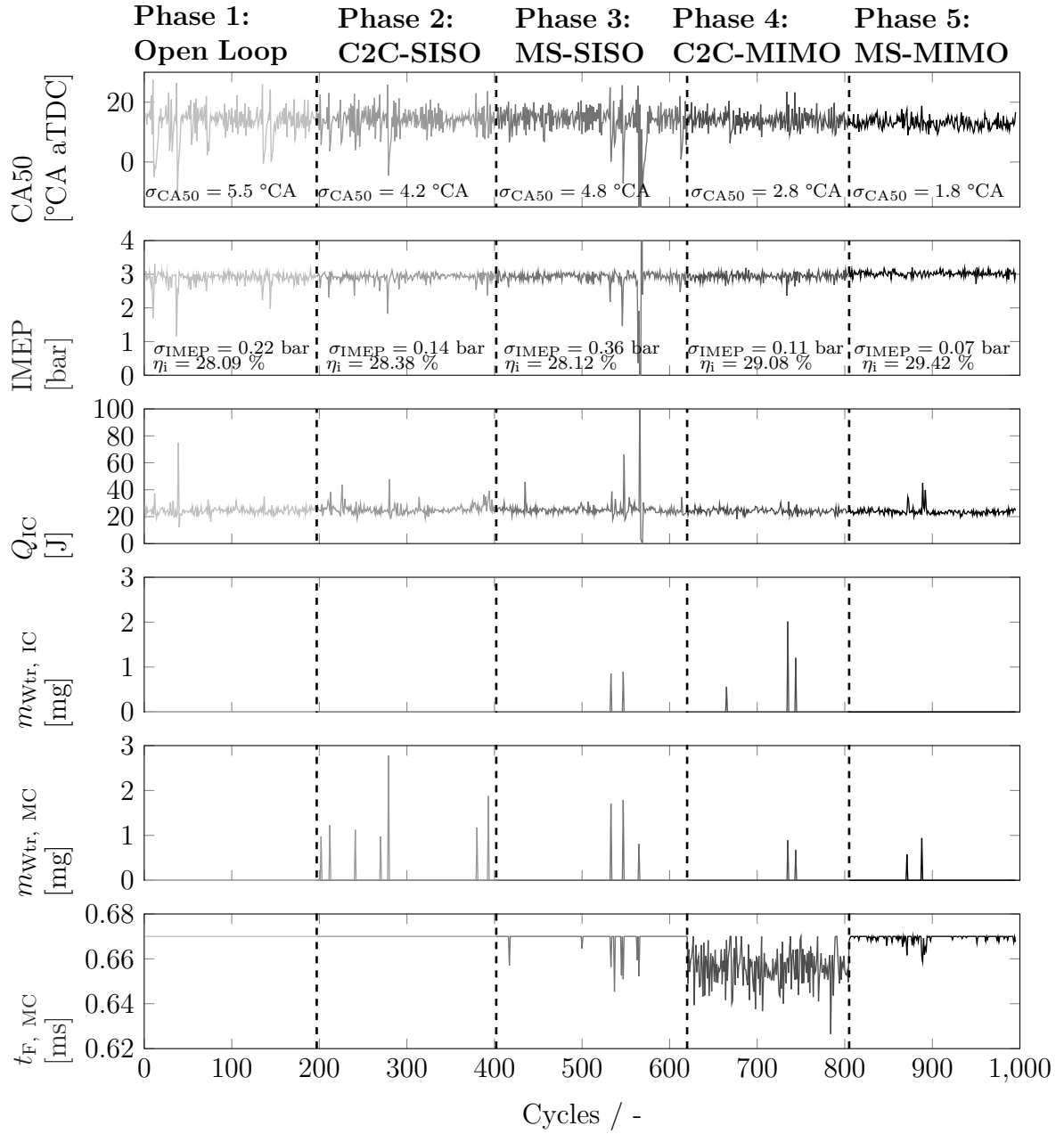
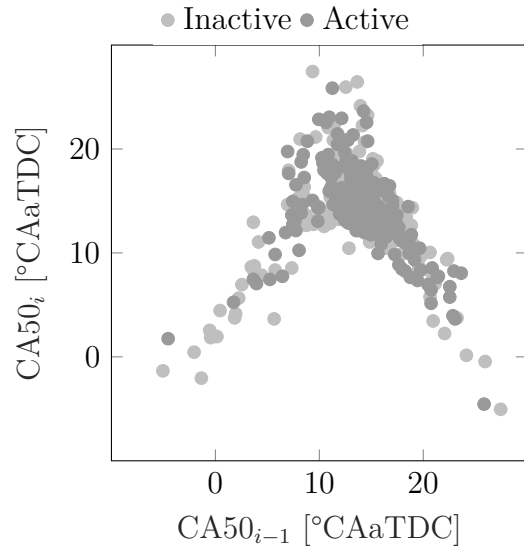
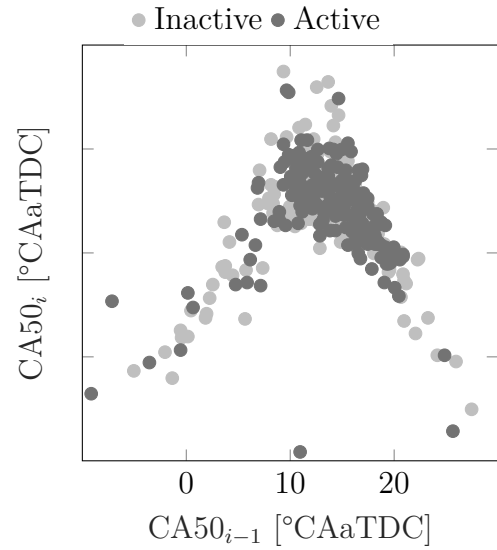


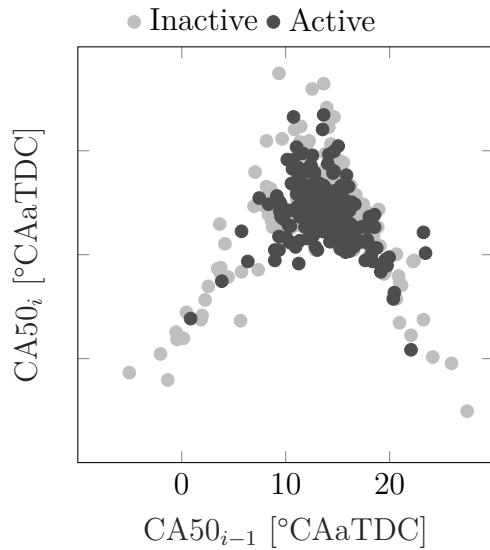
Figure 4.4: CA50, IMEP, heat release in IC  $Q_{IC}$  and operating variables  $m_{W, IC}$ ,  $m_{W, MC}$  and  $t_{F, MC}$  for different controller concepts @ IMEP=3 bar,  $n=1500 \text{ min}^{-1}$



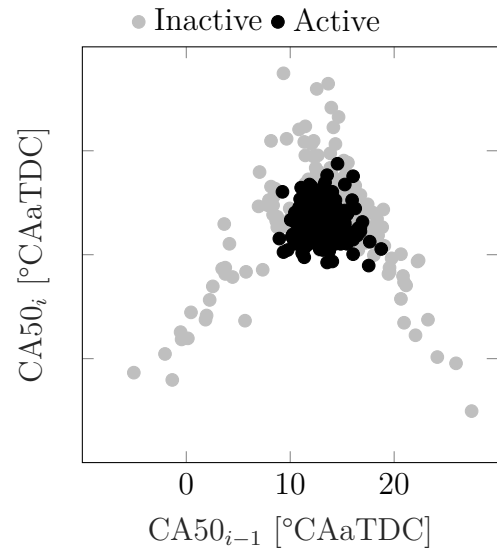
(a) Linear model free cycle-to-cycle SISO



(b) Linear model free multiscale SISO



(c) ANN based cycle-to-cycle MIMO



(d) ANN based multiscale MIMO

Figure 4.5: CA50 return map for different controller concepts compared to open-loop operation @ IMEP=3 bar,  $n=1500 \text{ min}^{-1}$

The Phase 2 and 3 concepts utilize cycle individual direct water [10] and fuel [82] injections. The applied control approaches are shown schematically in Figure 4.6.

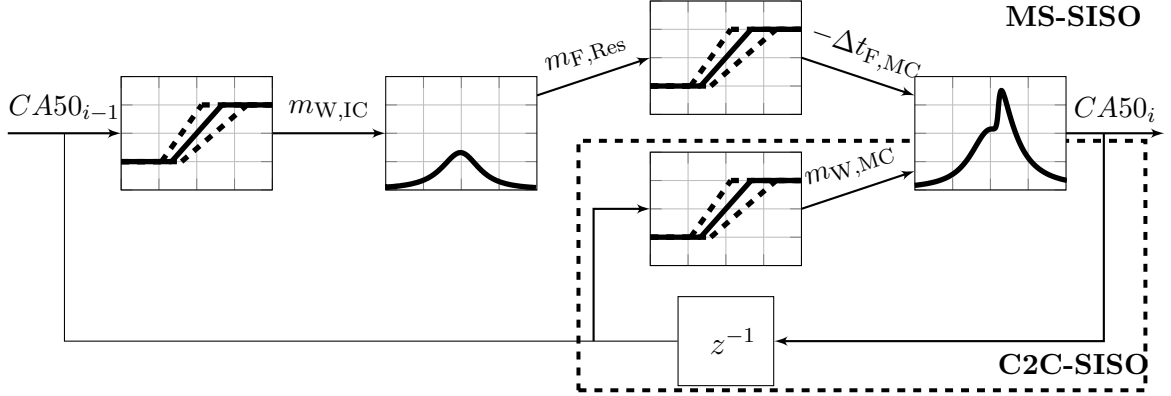
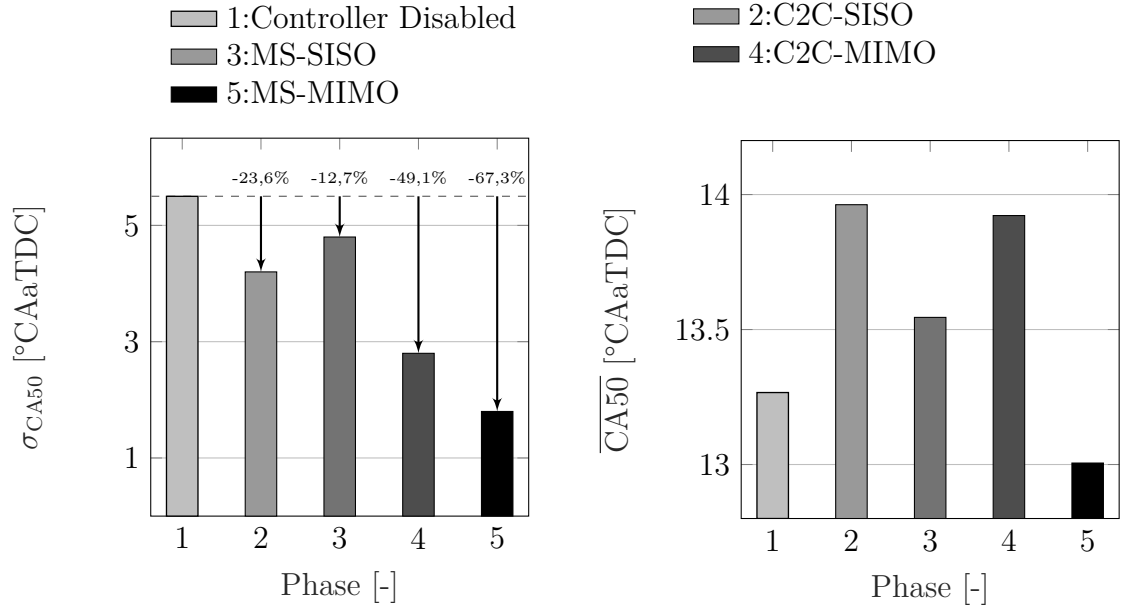


Figure 4.6: Structure of model free MS-SISO control approach and model free C2C-SISO control approach (dashed box)

In phase 2, the C2C-SISO control [10] is applied. As shown in the dashed box in Figure 4.6 the operating variable  $m_{W,MC}$  is determined using a linear control law dependent on the CA50 of the previous combustion cycle. The intention is to prevent an early combustion phasing, which is expected after a late combustion, by delaying the combustion using direct water injection.

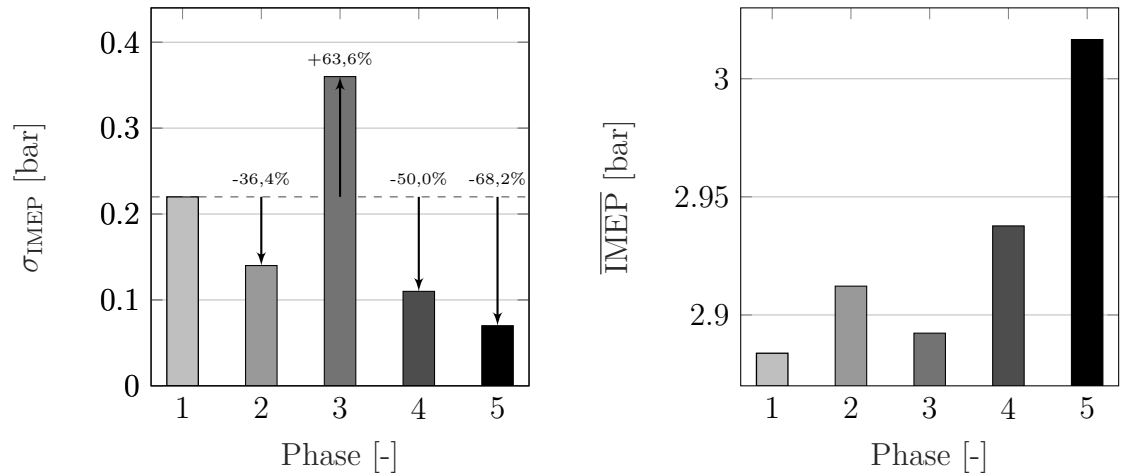
In phase 3 the MS-SISO controller combines the main findings of the preliminary work. Here, two independent water injections[10]  $m_{W,IC}$  and  $m_{W,MC}$  are used as the control variables, which are dependent on the CA50 of the previous combustion cycle. Additionally, the fuel injection duration[82]  $t_{F,MC}$  is adjusted such that possible residual fuel mass  $m_{F,Res}$  is compensated by shorter injections to keep the overall fuel mass constant.

As shown in fig 4.4 in phase 2 the C2C-SISO approach is capable of decreasing the variance in CA50 and IMEP. However, there are still significant outliers. The combination of multiple rulebased control laws in phase 3 results in several early combustion cycles and an increase of  $\sigma_{CA50}$  of 10.9 % compared to the C2C-SISO approach (cf. Figure 4.7a).



(a) Standard deviation of CA50

(b) Mean value of CA50



(c) Standard deviation of IMEP

(d) Mean value of IMEP

Figure 4.7: Standard deviation and mean value of CA50 and IMEP of different control approaches in comparison to the uncontrolled case @ IMEP=3 bar,  $n=1500 \text{ min}^{-1}$



However, it was shown in preliminary work, that those operating variables have a high potential for stabilization, the high non-linearity of the process results in insufficient controller performance of rulebased SISO approaches. Thus, in phase 4 and 5 non-linear model-based methods are investigated.

**Phase 4: ANN Based Cycle-to-Cycle C2C-MIMO Controller:**

The following two approaches use MIMO controllers with ANNs from Figure 4.2a and 4.2b to match the high non-linear process behavior. Phase 4 in Figure 4.4 shows the results of the C2C-MIMO controller.

For model based concepts in phase 4 and 5 the following target values are set:

$$\mathbf{x}_{i,\text{target}} = \begin{pmatrix} CA50_{i,\text{target}} \\ Q_{i,\text{MC},\text{target}} \\ Q_{i,\text{IC},\text{target}} \end{pmatrix} = \begin{pmatrix} 8 \text{ }^\circ\text{CA aTDC} \\ 480 \text{ J} \\ 20 \text{ J} \end{pmatrix} \quad (4.10)$$

Contrary to the SISO concepts, it is apparent that this ANN based approach more frequently applies the adaption of fuel injection duration. Overall, the standard deviation of CA50 and IMEP were reduced by 49.1% and 50.0% respectively (Figure 4.7) using the C2C-ANN. Additionally, less fuel is injected and the mean IMEP is increased to 2.94 bar, showing a clear improvement of the indicated efficiency to  $\eta_i = 29.08 \%$  compared to  $\eta_i = 28.09 \%$  for the uncontrolled case.

Nevertheless, as shown in Figure 4.4, there are two outliers in CA50 in cycles 734 and 743 which are directly counteracted by an additional water injection in cycles 735 and 744. Thus, water injections are generally applied following very late combustion phasing with  $CA50 > 20^\circ\text{CA aTDC}$ .

Cycles 735 and 744 correspond to the both extreme right data points in the CA50 return map in Figure 4.5c. The effect of the water injection is obvious as the data points are shifted upwards compared to the reference cycles, approaching the target for CA50. Thus a rapid, early combustion is prevented. The point cloud spread

becomes tighter showing the stabilization of the process compared to the SISO approaches.

#### **Phase 5: ANN Based Multiscale MS-MIMO Controller:**

In Figure 4.4, the MS-MIMO concept is applied in phase 5 showing a slightly different behavior than the C2C-MIMO concept. Here, the controller interventions are mainly performed by adjusting the fuel injection. However, the variation in fuel mass ( $\Delta t_{F,MC}$ ) is significantly lower than what was observed for the C2C-MIMO controller. The water injection before the IC is no longer applied at all and only in two cases water injection is applied for the MC following late combustion cycles. The two water injection interventions match with the two peaks of  $Q_{IC}$ , meaning that this information is taken into account by the in-cycle-ANN. Furthermore, the trace shows no peaks with  $Q_{IC} > 75$  J. This confirms the assumption made during the measurement that a higher heat release in the IC can be suppressed by cycle individual control interventions.

This MS-MIMO approach results in a further reduction of the process fluctuations in CA50 and IMEP. A reduction of the standard deviation for CA50 and IMEP of 67.3% and 68.2% is achieved. Even compared to the C2C-MIMO approach, a significant reduction in standard deviation for CA50 and IMEP of 35.7% and 36.4% is obtained. Additionally, there is a significant increase in IMEP to 3.02 bar while shifting the indicated thermal efficiency further up to 29.42 % leading to an 1.2 % higher efficiency than for the C2C-MIMO approach. In Figure 4.5d, the point cloud is further reduced showing even more stable combustion than the C2C-MIMO approach.

Since the two model-based MIMO controllers use the same cycle-to-cycle-ANN for control, the difference in controller interventions can only be based on the in-cycle ANN. The latter model has considerably more information about the process and is therefore noticeably better at predicting which operating variables are required for the MC. This clearly shows the relevance for multiscale control approaches for HCCI

using a second feedback loop within each cycle.

Only a single HCCI operating point was examined but for a real vehicle the methodology needs to be applied over a wider operating range. This would require additional measurements and retraining of the ANN which is beyond the scope of this thesis. Nevertheless, the methodology has a high potential over the whole HCCI operating range, as it provides a database with a very high dynamic and increased variance to address the high non-linearity of HCCI.

## 4.4 Summary of chapter

To further improve the combustion stability of HCCI, the in-cycle controller was expanded with the addition of cycle-to-cycle controller. Here a cycle-to-cycle model for the entire cycle and a in-cycle model was created for several time scales according to the split autoregressive process. By taking into account additional feedback variables from the intermediate combustion, it was shown that the model quality can be significantly increased and the validation error reduced. These MIMO models were then each implemented in a control structure, and ensuring the real-time capability. A C2C-SISO control approach presented in [15] and a new MS-SISO controller with linear control rule were used for a baseline comparison. Both of the MIMO controllers significantly stabilize the operating point compared to the SISO controllers. Specifically when compared to the C2C-MIMO control, the MS-MIMO controller reduces the standard deviation of IMEP and CA50 by 18.2%.

# Chapter 5

## HCCI Emission and Performance Modeling <sup>1</sup>

Accurate models are essential to model based control strategies. Due to the complex combustion process and the large variation in fuel properties emission models are often complex and difficult to implement in model based control. Finding an accurate emission model for the HCCI combustion process is one goal of this chapter. It is also important that these emission models are suitable for real-time implementation on the engine controller. These models can then be embedded in a MPC to predict the engine out emissions and to allow the controller to consider emissions. Finding this balance between modeling accuracy and quick computational time resulted in these two approaches:

1. high speed FPGA based kinetics model
2. blackbox machine learning models

Each of these two models took a different approach to provide a model with high accuracy and the possibility of rapid calculation. The goal of the first approach was to take advantage of the parallel computing power of the FPGA to get an accurate real-time physics based emission calculation. While the second approach took advantage of data driven machine learning models that are quick to execute in real-time but require

---

<sup>1</sup> This chapter is partially based on [13, 177]

significant training data. These methods will be presented in the next sections.

## 5.1 White-Box Chemical Kinetics Modeling

Real-time measurement of in-cylinder conditions, such as gas temperature and composition are extremely difficult to directly measure and often require advanced optical systems [178]. A detailed HCCI model can estimate these emission states and can then be used for estimating other states used in a controller. This has been found essential for implementing reliable closed loop controllers [7]. Based off a measured in-cylinder pressure trace a multi-zone kinetics model determines gas temperatures, compositions and combustion phasing, providing the information needed for a controller to reduce emissions. A simpler single zone model is not capable to calculate pollutant emissions but reduces the calculation effort drastically. Temperature and combustion phasing can still be predicted and used as in-cycle control input [9, 179]. This section describes the translation of a kinetics mechanism to a real-time implementation on an FPGA. Here, a preexisting detailed chemical combustion mechanism is validated for a single cylinder research engine (SCRE), extending an existing gas exchange model, that is available both in post processing [180] as well as real-time operation [15, 163], providing the initial conditions to the combustion model. This section is structured into four subsections. First the mechanism is presented, then the combustion model is described and its performance analyzed. In the third part the transfer of the model to Simulink is explained and mathematical problems for porting the model to the FPGA are described. The performance of the FPGA model is described in the last step.

### 5.1.1 Mechanism

For real-time combustion modeling a chemical kinetic mechanism describing HCCI combustion is needed. Gasoline is made up out of several hundred chemical com-

pounds and therefore too complex for subsequent real-time implementation [181, 182]. Instead, primary reference fuel (PRF) mechanisms that model iso-octane and n-heptane, are used. The mechanism used here is from Tsurushima [183] which is derived from the work of the Lawrence Livermore National Laboratory (LLNL) [184]. A process has been developed that has a modular form to allow most mechanisms in Chemkin [185] or Cantera [186] format to be used. The Tsurushima mechanism consists of 34 species and 48 reactions, of which 46 are elementary and 2 are three-body reactions. The mechanism is based on a kinetic model for PRF from Tanaka [187] with a very simple high temperature mechanism and a modified low temperature model. The mechanism has been validated using a shock tube and gas sampling in an HCCI engine, and is known to have reduced precision in simulating ignition timings due to its small size.

### 5.1.2 Combustion model

The combustion model describing the in-cylinder conditions can be separated in three parts:

- geometry
- thermodynamics
- reaction kinetics

The interactions between the three parts of the kinetics model describing the in-cylinder state are summarized as shown in Figure 5.1. These equations represent a physics based model of the combustion process and can be used to calculate the in-cylinder state.

Geometry is the driving factor for the compression ignition and solely consists of extensive values. The combustion itself in the reaction kinetics part is described entirely intensive and the thermodynamic part links the two together. Two main

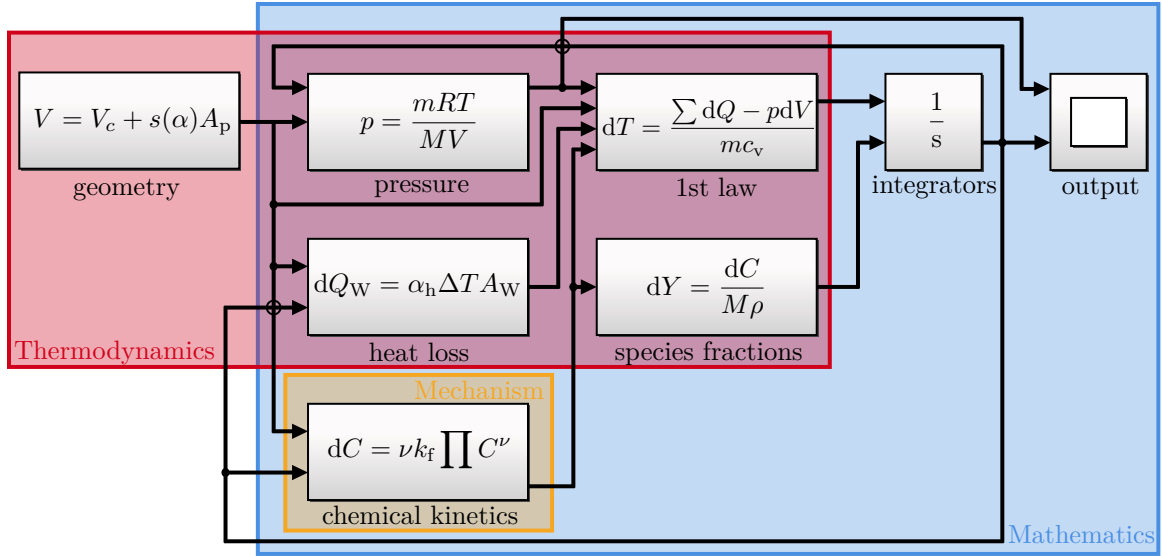


Figure 5.1: Schematic display of the combustion kinetics model highlighting the three separate components.

versions of the model are developed. The first one calculates the pressure internally while the second one is based on measured combustion chamber pressure.

## Geometry

The geometry of an internal combustion engine can be represented by:

- piston position  $s(t)$
- piston speed  $\dot{s} = \frac{ds(t)}{dt}$
- cylinder volume  $V(t)$
- change in volume  $dV(t)$
- crank angle  $\alpha$
- connecting rod length  $r$

The piston equation which describes  $s(t)$  as a function of the crank angle  $\alpha$  and

connecting rod to crank ratio  $\lambda_s$  is:

$$s = r \cdot \left[ 1 - \cos(\alpha) + \frac{1}{\lambda_s} \cdot \left( 1 - \sqrt{1 - \lambda_s^2 \cdot \sin^2(\alpha)} \right) \right] \quad (5.1)$$

With the known geometric parameters bore diameter  $D_{bore}$  and clearance volume  $V_c$  the volume can be calculated as:

$$V = V_c + s \cdot \frac{D_{bore}^2}{4} \cdot \pi \quad (5.2)$$

The derivative of equation 5.1 with respect to time results in the following equation:

$$\dot{s} = \frac{ds}{dt} = r \cdot \sin(\alpha) \cdot \frac{\lambda_s \cdot \cos(\alpha)}{\sqrt{1 - \lambda_s^2 \cdot \sin^2(\alpha)} + 1} \cdot \quad (5.3)$$

For the change in volume equation 5.2 is derived with respect for time by using  $ds$  from equation 5.3.

$$\dot{V} = \frac{ds}{dt} \cdot \frac{D_{bore}^2}{4} \cdot \pi \quad (5.4)$$

## Thermodynamics

The thermodynamic part of the model describes the intensive properties:

- temperature  $T$
- pressure  $p$
- density  $\rho$

based off of extensive values from geometry and mass  $m$ . Pressure and density are calculated using the ideal gas law assumption:

$$p = \frac{m \cdot R \cdot T}{M \cdot V} \quad (5.5a)$$

$$\rho = \frac{p \cdot M}{R \cdot T} \quad (5.5b)$$

The pressure is only needed when when the model is used to predict future values and in that case the measured pressure trace can not be used. Temperature is calculated using the first law of thermodynamics

$$\Delta U = \sum Q - \sum W \quad (5.6)$$



For a closed system, the first term is:

$$\frac{dU}{dt} = m \cdot c_{v,m} \cdot \frac{dT}{dt} . \quad (5.7)$$

Substituting eqn 5.7 into eqn 5.6 by solving for  $\frac{dT}{dt}$  results in:

$$\frac{dT}{dt} = \frac{\frac{dQ_{\text{chem}}}{dt} - \frac{dQ_{\text{wall}}}{dt} - p \cdot \frac{dV}{dt}}{m \cdot c_{v,m}} . \quad (5.8)$$

Where chemical energy, heat transfer through the walls and piston work have been incorporated using the wall heat transfer model as [188]:

$$\frac{dQ_{\text{wall}}}{dt} = a_h \cdot (T - T_w) \cdot (\pi \cdot D_{\text{bore}} \cdot s + 2 \cdot A_{\text{piston}}) . \quad (5.9a)$$

$$a_h = \frac{\left(\left(\frac{ds}{dt} + 1.4\right) \cdot p \cdot 10^{-5}\right)^{0.8} \cdot 130}{T^{0.4} \cdot V^{0.06}} \quad (5.9b)$$

and heat release  $dQ_{\text{chem}}$ :

$$\frac{dQ_{\text{chem}}}{dt} = \sum_i u_i \cdot \frac{dY_i}{dt} \quad (5.10)$$

which is driven by the change in mass fractions  $Y_i$  as well as the internal energies  $u_i$  both provided by the reaction kinetics.

## Reaction kinetics

The reaction kinetics calculate the species concentration and require the chemical, thermodynamic and kinetic properties of the mixture to describe the chemical properties of the gas. Heat capacity ( $c_v$ ) and internal energy ( $u$ ) are based on NASA polynomials [189] while the molar masses ( $M$ ) are constant for each species that only change for the mean value but never for one component. The reaction kinetics mechanism is the change in species concentration for species  $i$  in a reaction with all species  $n$  is:

$$\frac{dC_i}{dt} = \nu_i \cdot k_f \prod_n C_n^{\nu_n} . \quad (5.11)$$

The empirical factor  $k_f$  describes the speed of the reaction that is temperature dependent and is determined using the Arrhenius equation:

$$k_f = A_f \cdot T^b \cdot \exp\left(\frac{-E_a}{R \cdot T}\right). \quad (5.12)$$

Some reactions have both a forward and reverse reactions that are simultaneously occurring, and thus is accounted for in the combustion mechanism to maintain thermodynamic equilibrium. To obey equilibrium conditions, Cantera handles these reverse reactions by introducing an equilibrium factor  $K_C$ :

$$K_C = \frac{k_f}{k_b} \quad (5.13)$$

and uses that as a starting point to calculate the reverse (backwards) speed factor  $k_b$ .  $K_C$  can also be calculated from the free Gibbs energy:

$$\Delta g = \sum_i [\nu_{\text{prod},i} \cdot (h_i - T \cdot s_i) - \nu_{\text{reac},i} \cdot (h_i - T \cdot s_i)] \quad (5.14)$$

resulting in:

$$K_C = \left(\frac{p_0}{R \cdot T}\right)^{\sum \nu_{\text{net}}} \cdot \exp\left(\frac{-\Delta g}{R \cdot T}\right). \quad (5.15)$$

Using  $K_C$  to calculate  $k_b$  provides the retarding of the progression rate caused by the reverse reaction without violating the equilibrium condition [190]. Species fractions as needed for equation 5.10 are correlating to concentration as follows.

$$\frac{dY_i}{dt} = \frac{dC_i}{dt} \cdot \frac{M_i}{\rho} \quad (5.16)$$

The mechanism describes both the components properties as well as a set of reactions for the combustion of iso-octane and n-heptane which is initially implemented in the

open source Cantera software. The Tsurushima mechanism only utilizes two types of reactions, elementary and three-body. The elementary reactions follow the form of equation 5.11. To describe a reaction with a third and inert collision partner, an additional concentration  $C_m$  describing the rest of the mixture with all species  $n$ :

$$C_m = \sum_n \epsilon_n \cdot C_n . \quad (5.17)$$

The efficiency vector  $\epsilon_n$  is accounting for oscillation frequencies that can make a lighter molecule absorb more kinetic energy.

### 5.1.3 Model performance

A single zone model is calculated offline using MATLAB and Cantera which is capable of simulating the combustion process. Inputs to the calculation are the initial conditions provided by the sensors and gas exchange model. The calculated cylinder

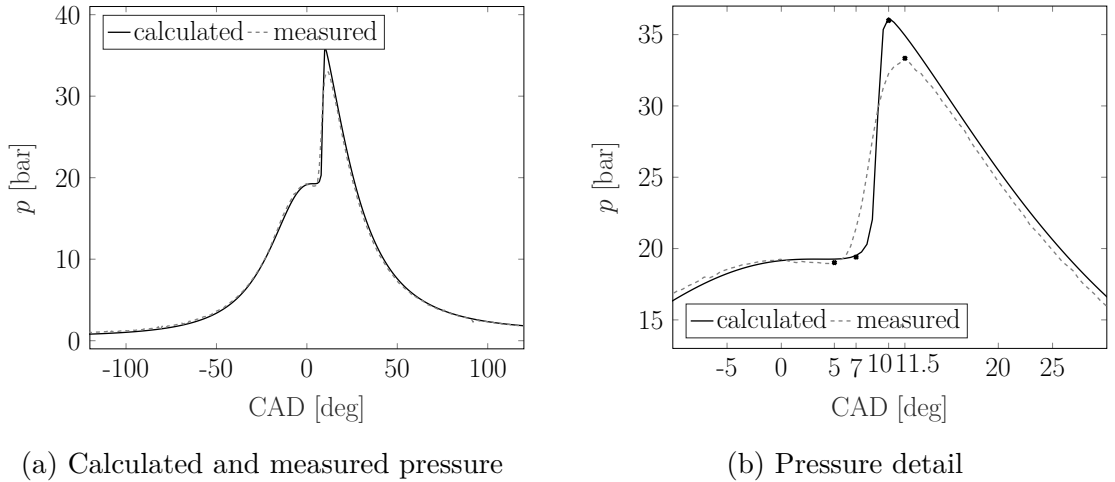
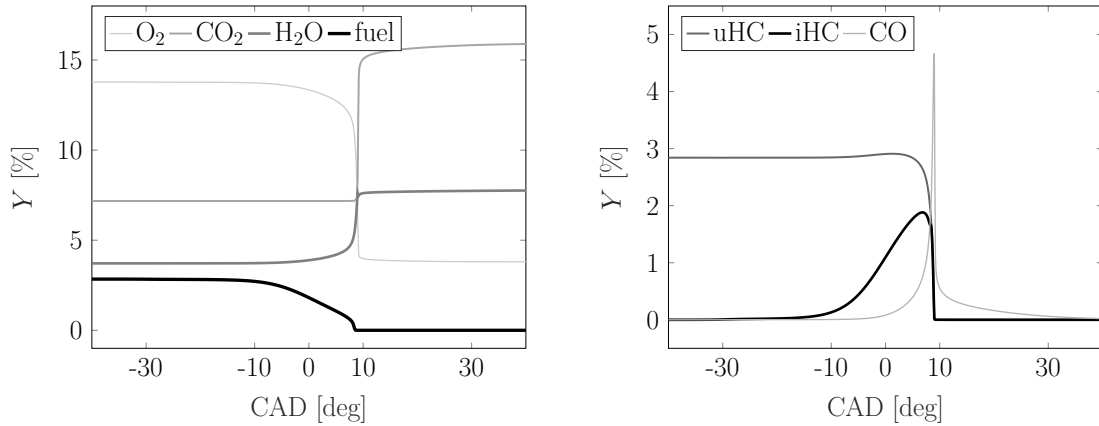


Figure 5.2: Comparison between calculated and measured (average over 250 cycles) pressure for a post processed single zone model applying the Tsurushima mechanism in MATLAB and Cantera

pressure is displayed in Figure 5.2a and compared to the average measured cylinder pressure (over 250 engine cycles). The slight discrepancy between the measured and the calculated pressure trace during the compression is attributed to a slight initial

condition mismatch between the model and measurement. During combustion (detail Figure 5.2b) three main differences were observed. The measured combustion starts earlier, shows a less steep pressure rise rate and does not reach as high peak pressure as the simulation. This difference is attributed to the simplification of the simulation model - the whole combustion chamber is simulated as one zone at only one temperature. So the combustion temperature is the same throughout the cylinder and is either taking place in its entirety, with a 100% conversion efficiency, or not at all in the model.



(a) Global reaction: fuel,  $O_2$ ,  $CO_2$  and  $H_2O$

(b) Intermediates: HC and CO

Figure 5.3: Gas composition calculated by a single zone model using the Tsurushima mechanism, complete conversion and over stoichiometric conditions.

This is a simplification of the experimental combustion since the model has no delay through colder or less fuel rich zones in the cylinder, neither is there transport mechanisms taking intermediate products out of the combustion. This directly leads to the steeper pressure rise caused by a faster combustion in the model and is also the cause for the higher peak pressure [191]. The model assumption of complete combustion releases more energy that causes a higher pressure. In Figure 5.3 the species fractions  $O_2$ ,  $CO_2$ ,  $H_2O$ , fuel, HC and CO from the Tsurushima mechanism are displayed around TDC. The combustion occurs starting at -10 CAD and proceeds to completion at 9 CAD aTDC. The concentrations of the products  $CO_2$  and  $H_2O$

rise during combustion. Because of the residual gas from the last cycle is trapped inside the cylinder,  $\text{CO}_2$  and  $\text{H}_2\text{O}$  concentrations are not zero before combustion.  $\text{O}_2$  is consumed during the combustion, leading to a drop in concentration. After the simulated combustion is over at 9 CAD there is still oxygen left in the cylinder since the operating point is lean. The hydrocarbon species fractions in Figure 5.3b are further separated into overall unburned hydrocarbons uHC including everything that has at least one C-atom and one H-atom, and intermediate hydrocarbons iHC that excludes the two fuels iso-octane and n-heptane. At the beginning of the combustion a rise in overall uHC concentration can be seen. This is not an indicator for a growth in hydrocarbons, which would contradict mass conservation, but rather displays iHC having bigger molecular weights leading to bigger mass fractions. HC concentrations are zero at 9 CAD marking the end of combustion, while CO continues to oxidization to  $\text{CO}_2$  until 30 CAD. The lack of transport mechanisms in a single zone model make them unsuitable to accurately simulate emissions apart from final products of a complete combustion (compare to Figure 5.3). Figure 5.4 shows the measured pressure as well as intermediate hydrocarbon mass fraction. The quantitative values of the two traces is not of interest here, but rather their timing in relation to each other. The calculation of iHC, either based on the measured pressure trace or based on an internally calculated pressure, allows to predict combustion phasing 10 to 15 CAD ahead. This knowledge will be useful for the application of a rapid spark system avoiding misfires in real-time.

#### **5.1.4 Simulink model**

MATLAB combined with Cantera was found to be a flexible and modular way to switch mechanisms and initial conditions easily. These programs are built to run offline on a desktop computer which utilizes a high clock rate processor, allowing for operation with great flexibility and in a wide range of applications. However,

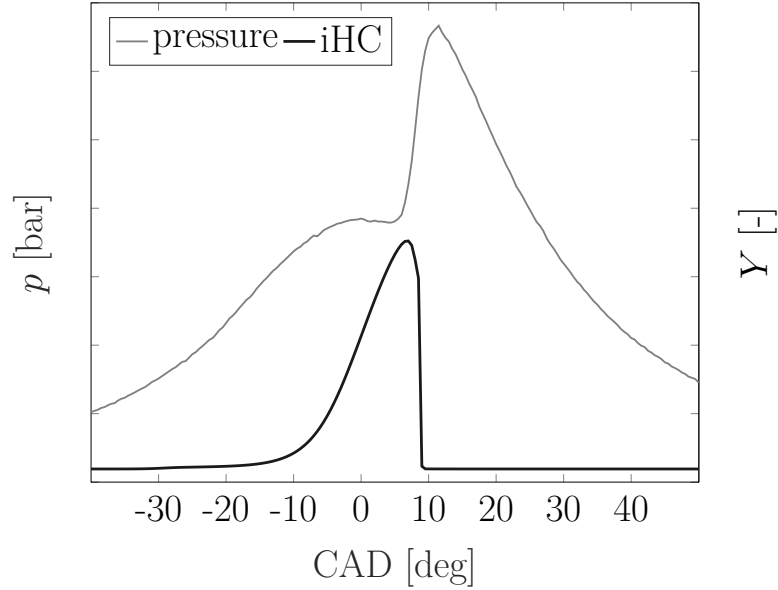


Figure 5.4: Measured pressure and iHC mass fraction versus CAD for the post processed model

these processor based calculations are orders of magnitude too slow for a real-time application on an engine. Real-time calculations must be completed in the available time and require significant optimizations when porting the offline model. These real-time calculations are therefore very specific and not meant to allow for flexibility. The goal is a highly specialized and optimized model, that is capable of meeting the strict real-time timing requirements. A FPGA provides the general capabilities needed to run a specialized set of equations at high rates. The whole offline model, including the chemical kinetics calculations from Cantera, has to be ported using the Xilinx System Generator Blockset, a Simulink based programming language that allows for compiling *very high speed integrated circuit hardware description language* (VHDL) running on the FPGA. The steps to port the Cantera and Matlab based model (presented in Section 5.1.2) to Simulink is described.

In the first step, the post-processed model is replicated exactly in Simulink. Meaning that the exponential equations 5.12 and 5.15 are calculated by Simulink. Simulink is capable of handling exponents but shows reduced efficiency. FPGAs are based on

fixed point operations and therefore are inherently inefficient and imprecise when calculating exponents. In a second step these equations have to be substituted by lookup tables.  $k_f$  as a function of temperature can be directly translated in a lookup table following equation 5.12. Figure 5.5 shows the relative error between the solution of the Simulink model and the solution of the MATLAB model. The peak itself results from a relatively small deviation in combustion phasing that results from the error tolerances of the Simulink ODE solver. This error will not propagate further since the discretized model is no longer using a variable step size ODE solver. The error

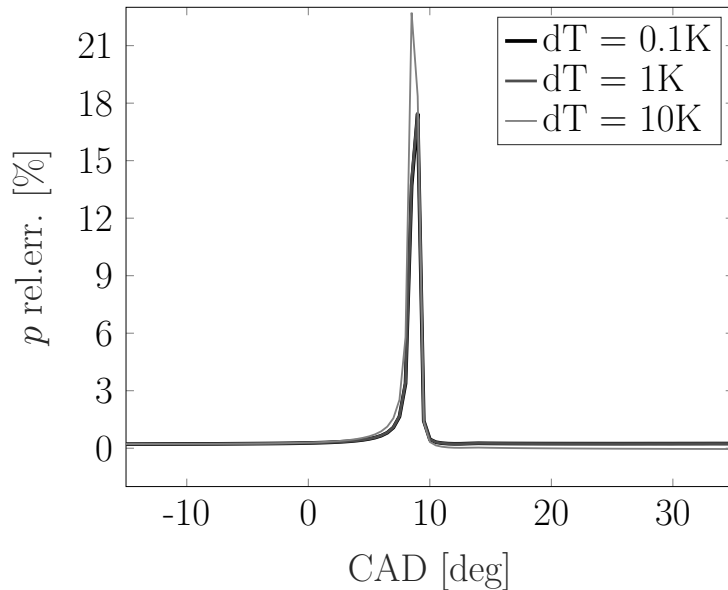


Figure 5.5: Relative error between the MATLAB and the Simulink pressure trace for different resolutions of  $T$  in the  $k_f$  lookup table.

for 3 different resolutions on the temperature axis of the  $k_f$  lookup table is also shown in Figure 5.5. It can be seen that for a resolution less than 1 K no significant improvement in error is observed. The exponential in equation 5.15 are dependent on temperature and on gas composition through Gibbs free energy. For the Tsurushima mechanism with 34 species the lookup table would need 36 dimensions which is not practical as the maximum lookup table size in Simulink is 30 dimensions. Simplifications must be made to avoid this. Which are the equilibrium factor  $K_C$  which is

provided in 2 separate lookup tables of 2 dimensions, temperature and  $K_C$ . One table is for a typical lean unburned mixture and the other for a burned mixture.

### 5.1.5 FPGA model

A variable step size solver is not suitable for real-time implementation as a fixed time step is used for the combustion model on the FPGA. Unlike processor based computers, that are capable of doing a broad variety of tasks in any order, a FPGA performs a fixed set of calculations. Those calculations are all completed at the exact same time and truly parallel with a set clock rate. This unique feature makes the FPGA ideal for combustion simulations where the calculations and the reactions progression must be calculated at the same time. To calculate all reactions at the same time in Simulink, but without generating 34 sets of blocks, a quadratic matrix was used for the stoichiometric factors in equation 5.11. That causes a problematic exponential for FPGA implementation. Additionally, most of the time, this exponential results in a power of zero anyways. This is avoided by precalculating the reaction exponential by setting the stoichiometric exponents that are known. Therefore, only the participating concentrations are connected to the reaction block.

As the FPGA has a fixed clock rate and therefore requires the solution step size to be predefined. The first approach was to run the model using the smallest time step, identified in the variable time step offline simulink model. This was found to be  $1.6373e-10$  s. This would result in the FPGA solver starting the calculation at IVC with the smallest needed step size ( $\Delta t_{\min}$ ) that is necessary to calculate the combustion around TDC, where the changes in the ODE are the steepest. A discrete solver that is fast enough to keep up with the real process could align with the measured pressure at the pressure sensors resolution of a tenth of a CAD, and would finish the simulation and provide emissions immediately after combustion. In addition to that it would provide the SOC before there is any significant or measurable heat



release. A mildly optimized model has been generated for the FPGA requiring 25 sequential logical calculations per solver step. Thus the minimal time step possible on the FPGA is:  $\Delta t_{FPGA} = 25 \cdot f_{clock}$ , with the clock rate  $f_{clock} = 12.5$  ns. That results in a minimal step size of  $3.125e-7$  s which is 1908 times the minimal time step of the ode23tb Matlab solver of  $1.6374e-10$  s. Figure 5.6 shows the calculated temperature of a discrete solver with the step size  $\Delta t_{FPGA}$ . During one rotation at 840 rpm the

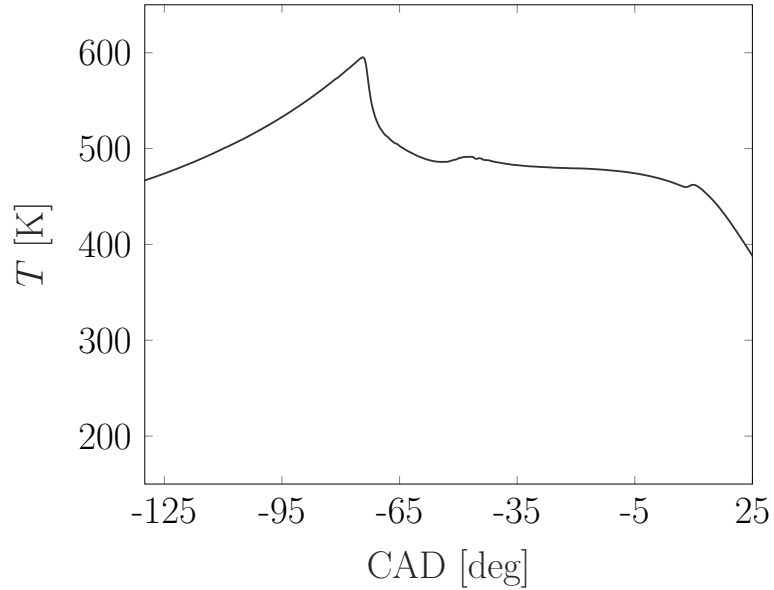


Figure 5.6: Temperature calculated with an external pressure trace at the minimal FPGA step size  $\Delta t_{FPGA} = 3.125e-7$  s.

FPGA can calculate 228,576 steps which equals 0.0523% of the steps needed for a step size of  $\Delta t_{min} = 1.6374e-10$  s. This means the current FPGA hardware is only able to calculate 0.1886 CAD of combustion during one revolution of the engine. Since a model with a resolution of 0.1 CAD is needed, this meant that with the current FPGA hardware, it was not possible to run a fully white-box kinetics model in real-time. With future improvements in the clock rate of FPGA boards it could be possible to implement a realtime whitebox kinetics based model for engine control.

## 5.2 Machine Learning Methods: Steady State Models

The FPGA based physical kinetics model developed in the previous section provided detailed information of the chemical composition of the cylinder throughout the combustion process, however, even with the parallel computation power of an FPGA it was not possible to calculate the emissions in real-time. In order to get an emission model that can be used within a MPC, ML methods are explored to find an accurate emission model of the HCCI process that are suitable for realtime implementation. Data is necessary for the machine learning methods using and will be described next.

It is well understood that HCCI has a narrow operating range and has high efficiency and low emissions within specific operating conditions [25, 154, 155]. In vehicles, the limited load range is not as critical in hybrid and electric range extender applications. These require only a few efficient load and speed operating points as the electric part of the propulsion systems are used to handle any transient loads. To simulate a steady state operating point the engine is operated in a conditioned environment that keeps rotational speed, load, intake pressure and temperature, oil and coolant temperature and exhaust pressure constant. Only one load and speed is selected to reduce the experimental space and demonstrate the proposed emission model.

Active input factors to the HCCI combustion process include injected fuel mass, injection timing and valve timings. These variables were chosen to be varied as they significantly affect the combustion process and the resulting engine out emissions at a given load and speed operating point. They also strongly influence the combustion stability of the HCCI process which has a significant effect on the engine out emissions [3, 7]. As the HCCI process is sensitive to operating conditions, a relatively small change in an input parameter can result in a significant change in engine output parameters. These engine input parameters have also been explored in previous

work regarding HCCI emissions modeling [132]. The variation in engine inputs is summarized in Table 5.1.

Table 5.1: Range of HCCI engine input parameters

Engine Input	Min	Max	Mean
SOI [deg aTDC]	455.9	495	473.4
$m_f$ [mg]	2.7	2.96	2.86
NVO [deg]	173	201.6	187.7

The process of creating a machine learning based emission model can be broken into two main steps. First, the required data for emission modeling of the HCCI engine was collected as discussed in Section 2. Figure 5.7 schematically shows the data collection and emission modeling. During data collection, fuel amount ( $m_f$ ), negative valve overlap duration (NVO), and start of fuel injection (SOI) are the main inputs while intake pressure ( $P_{in}$ ) and intake temperature ( $T_{in}$ ) are held constant during data collection. The power output of the engine is represented by IMEP and the operating points are chosen to keep the output load approximately constant for all tests with an IMEP of  $3.5 \pm 0.2$  bar. Holding IMEP constant while varying  $m_f$ , SOI, and NVO results in varying combustion efficiency and a trade-off between different emissions.

For these tests,  $\lambda$ , CO, CO<sub>2</sub>, NO<sub>x</sub>, and HC emission were measured. All manipulated and conditioned input and engine output variables in the emission modeling section are used as inputs to the data-driven system. Using unsupervised learning, different feature sets by interpolation of these inputs are created, and these features are the main inputs of the PSO-based SVM method. In this study, both Nonlinear SVM (NLSVM) and Linear SVM (LSVM) are considered for emission modeling using different feature sets, and PSO is used to optimize the hyperparameters of both the

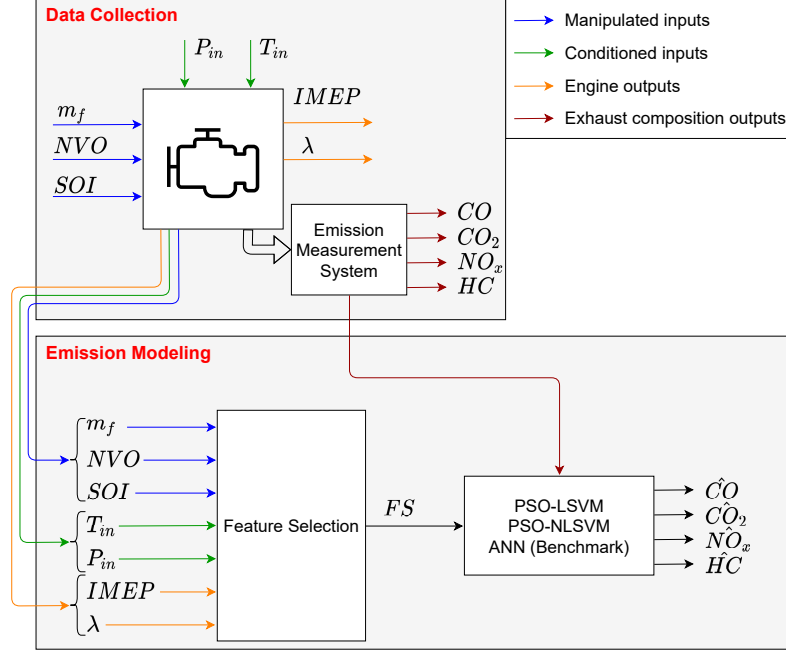


Figure 5.7: Schematic of data collection and proposed emission modeling using PSO-based LSVM and NLSVM

LSVM and NSVM. This method was compared with the benchmark Artificial Neural Network (ANN) emission modeling [120].

### 5.2.1 Support Vector Machine

The main idea of the regression form of SVM, also called Support Vector Regression (SVR) is to find an optimal hyperplane,  $\mathbf{y}(\mathbf{u}_i)$ , such that  $\mathbf{y}(\mathbf{u}_i)$  is as flat as possible and it has the maximum deviation of  $\epsilon$  for all training data [192]. In other words, the optimization problem is to find the flattest function with the maximum error tolerance  $\epsilon$ . Therefore, the optimal hyperplane which describes the training data,  $\{\mathbf{u}_i, \mathbf{z}_i\}$ , can be defined as:

$$\mathbf{y}(\mathbf{u}_i) = \mathbf{w}^T \mathbf{u}_i + \mathbf{b} \quad (5.18)$$

where  $\mathbf{u}_i$  and  $\mathbf{z}_i$  are input and target of the training data and  $\mathbf{w}$  and  $\mathbf{b}$  are found by solving the SVM algorithm for regression problems. The optimization problem to

find the optimum hyperplane  $\mathbf{y}(\mathbf{u}_i)$  is defined as:

$$\begin{aligned} & \text{Minimize: } \frac{1}{2} \|\mathbf{w}\|_2^2 \\ & \text{Subject to: } \begin{cases} \mathbf{z}_i - \mathbf{w}^T \mathbf{u}_i - \mathbf{b} \leq \epsilon \\ \mathbf{w}^T \mathbf{u}_i + \mathbf{b} - \mathbf{z}_i \leq -\epsilon \end{cases} \quad i = 1, \dots, n \end{aligned} \quad (5.19)$$

where the flattest function is achieved by minimizing  $\frac{1}{2} \|\mathbf{w}\|_2^2$  and the tolerance is achieved by solving for the defined constraints. A schematic of SVM regression is shown in Figure (5.8) where the main objective of SVM is shown as the orange line estimating a proper function by the maximum deviation of  $\epsilon$ .

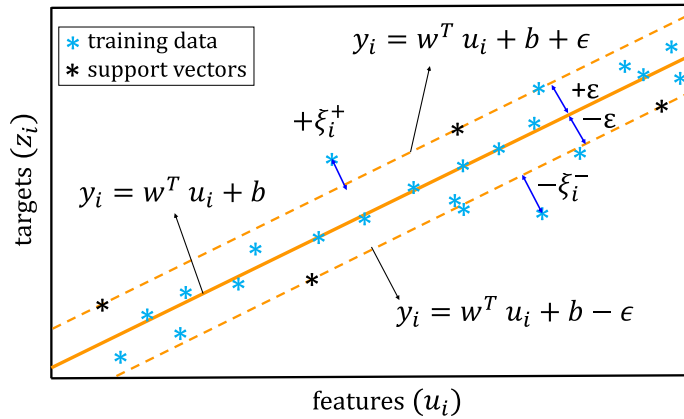


Figure 5.8: SVM regression and support vectors (based on [117])

For those data points within a defined tolerance ( $\epsilon$ ),  $\mathbf{y}(\mathbf{u}_i)$  has been found such that it predicts all pairs of learning data within a defined error. If all the data points lay within the defined tolerance, the optimization problem is feasible. However, occasionally the algorithm cannot converge within the defined constraints and the current optimization problem becomes infeasible. To overcome the infeasibility of Eq. (5.19), a penalty variable ( $\zeta_i$ ) or so called slack variable has been added to the original optimization problem as:

$$-\epsilon - \zeta_i^- \leq \mathbf{z}_i - \mathbf{y}_i \leq \epsilon + \zeta_i^+ \quad (5.20)$$

To consider these penalty variables, the Soft Margin Loss Function (SMLF) has been added to optimization problem which is defined as [193]:

$$\begin{aligned}
& \text{Minimize: } \frac{1}{2} \|\mathbf{w}\|_2^2 + C \sum_{i=1}^n (\zeta_i^+ + \zeta_i^-) \\
& \text{Subject to: } \begin{cases} \mathbf{z}_i - \mathbf{w}^T \mathbf{u}_i - \mathbf{b} \leq \epsilon + \zeta_i^+ \\ \mathbf{w}^T \mathbf{u}_i + \mathbf{b} - \mathbf{z}_i \leq \epsilon + \zeta_i^- \\ \zeta_i^-, \zeta_i^+ \geq 0 \end{cases} \quad i = 1, \dots, n
\end{aligned} \tag{5.21}$$

where  $C$  is a regulatory parameter to set the trade off between tolerated error and the smoothness of the model.

To consider constraints in the optimization problem the Lagrangian function is calculated as

$$\begin{aligned}
L = & \frac{1}{2} \|\mathbf{w}\|_2^2 + C \sum_{i=1}^N (\zeta_i^- + \zeta_i^+) \\
& - \sum_{i=1}^N \alpha_i^+ (-\mathbf{z}_i + \mathbf{y}_i + \epsilon + \zeta_i^+) - \sum_{i=1}^N \mu_i^+ \zeta_i^+ \\
& - \sum_{i=1}^N \alpha_i^- (\mathbf{z}_i - \mathbf{y}_i + \epsilon + \zeta_i^-) - \sum_{i=1}^N \mu_i^- \zeta_i^-
\end{aligned} \tag{5.22}$$

where  $\alpha_i^+$ ,  $\alpha_i^-$ ,  $\mu_i^+$ , and  $\mu_i^-$  are the non-negative Lagrangian Multipliers. The Lagrangian is solved by calculating the partial differential with respect to the optimization variables as

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \rightarrow w = \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) \mathbf{u}_i \tag{5.23a}$$

$$\frac{\partial L}{\partial \mathbf{b}} = 0 \rightarrow \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) = 0 \tag{5.23b}$$

$$\frac{\partial L}{\partial \zeta_i^+} = 0 \rightarrow \alpha_i^+ + \mu_i^+ = C \tag{5.23c}$$

$$\frac{\partial L}{\partial \zeta_i^-} = 0 \rightarrow \alpha_i^- + \mu_i^- = C \tag{5.23d}$$

where Eq. (5.23a), Eq. (5.23b), Eq. (5.23c) are SVM expansion, bias constraints, and the box constraint, respectively [117]. By substituting Eqs. (5.23a)-(5.23d) into Eq. (5.22) the Quadratic Programming (QP) problem can be defined by

$$\begin{aligned}
\text{Minimize: } L &= \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (\alpha_i^+ - \alpha_i^-)(\alpha_j^+ - \alpha_j^-) \mathbf{u}_i^T \mathbf{u}_j \\
&\quad - \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) \mathbf{z}_i + \epsilon \sum_{i=1}^N (\alpha_i^+ + \alpha_i^-) \\
\text{Subject to: } &\begin{cases} \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) = 0 \\ 0 \leq \alpha_i^+ \leq C \\ 0 \leq \alpha_i^- \leq C \end{cases}
\end{aligned} \tag{5.24}$$

which can be used in a compact version following [194]:

$$\begin{aligned}
\text{Minimize: } &\frac{1}{2} \alpha^T \mathcal{H} \alpha + f^T \alpha \\
\text{Subject to: } &A_{eq} \alpha = B_{eq}
\end{aligned} \tag{5.25}$$

where

$$\alpha = \begin{bmatrix} \alpha^+ \\ \alpha^- \end{bmatrix}, \quad \mathcal{H} = \begin{bmatrix} H & -H \\ -H & H \end{bmatrix}, \quad f = \begin{bmatrix} -\mathbf{z}_i + \epsilon \\ \mathbf{z}_i + \epsilon \end{bmatrix}, \tag{5.26}$$

$$H = [\mathbf{u}_i^T \mathbf{u}_j], \quad \mathbf{A}_{eq} = [1 \dots 1 \quad -1 \dots -1], \quad \mathbf{B}_{eq} = [0]$$

In order to calculate  $\mathbf{b}$ , the Karush-Kuhn-Tucker (KKT) conditions are used where [195, 196]:

$$\alpha_i^+ (-\mathbf{z}_i + \mathbf{y}_i + \epsilon + \zeta_i^+) = 0 \tag{5.27a}$$

$$\alpha_i^- (\mathbf{z}_i - \mathbf{y}_i + \epsilon + \zeta_i^-) = 0 \tag{5.27b}$$

$$\mu_i^+ \zeta_i^+ = (C - \alpha_i^+) \zeta_i^+ \tag{5.27c}$$

$$\mu_i^- \zeta_i^- = (C - \alpha_i^-) \zeta_i^- \tag{5.27d}$$

must be fulfilled at the optimum point. Based on these equations, only five following cases are possible as

$$\alpha_i^+ = \alpha_i^- = 0 \quad (5.28a)$$

$$0 < \alpha_i^+ < C, \alpha_i^- = 0 \quad (5.28b)$$

$$0 < \alpha_i^- < C, \alpha_i^+ = 0 \quad (5.28c)$$

$$\alpha_i^+ = C, \alpha_i^- = 0 \quad (5.28d)$$

$$\alpha_i^- = C, \alpha_i^+ = 0 \quad (5.28e)$$

To find the support vector, where  $|\mathbf{z}_i - \mathbf{y}_i|$  is exactly equal to  $\epsilon$ , only  $0 < \alpha_i^+ < C$ ,  $\alpha_i^- = 0$  and  $0 < \alpha_i^- < C$ ,  $\alpha_i^+ = 0$  must be fulfilled. Therefore,  $\mathbf{b}$  can be calculated as

$$\mathbf{b} = \frac{1}{|S|} \sum_{i \in S} (\mathbf{z}_i - \mathbf{w}^T \mathbf{u}_i - \text{sign}(\alpha_i^+ - \alpha_i^-) \epsilon) \quad (5.29)$$

where  $S$  represents the support vector set based on Eqs. 5.28b and 5.28c as:

$$S = \{ i \mid 0 < \alpha_i^- + \alpha_i^+ < C \} \quad (5.30)$$

Therefore, by solving Eqs. 5.29 and 5.25 and substituting  $\mathbf{w}$  and  $\mathbf{b}$  into Eq. 5.18,  $\mathbf{y}(\mathbf{u}_i)$  is obtained as [117]:

$$\begin{aligned} \mathbf{y}(\mathbf{u}) &= \sum_{i=1}^N (\alpha_i^+ - \alpha_i^-) \mathbf{u}_i^T \mathbf{u} \\ &+ \frac{1}{|S|} \sum_{i \in S} (\mathbf{z}_i - \mathbf{w}^T \mathbf{u}_i - \text{sign}(\alpha_i^+ - \alpha_i^-) \epsilon) \end{aligned} \quad (5.31)$$

### Kernel tricks

The structure of the dot product in Eq. (5.31) is a simple linear kernel which fails to capture any nonlinear behavior of the process. The inner product of Eq. (5.31),  $\mathbf{u}_i^T \mathbf{u}_j$ , can be replaced by a nonlinear kernel  $K(\mathbf{u}_i, \mathbf{u}_j)$ . By replacing the linear kernel with a



nonlinear kernel, using so called kernel tricks, brings nonlinear pattern recognition at a reasonable computational cost [197]. In this study the RBF (Radial basis function) kernel function is used as

$$K(\mathbf{u}_i, \mathbf{u}_j) = \exp\left(-\frac{\|\mathbf{u}_i - \mathbf{u}_j\|_2^2}{2\sigma^2}\right) \quad (5.32)$$

where  $\sigma$  is the Gaussian variance and  $\|\cdot\|_2$  is the two norm. Therefore, the prediction function,  $\mathbf{y}$  is calculated as [197]:

$$\mathbf{y}(\mathbf{u}) = \sum_{i=1}^n (\alpha_i^+ - \alpha_i^-) K(\mathbf{u}_i, \mathbf{u}) + \mathbf{b} \quad (5.33)$$

This study examines different interpolations of different features that also play the precise role of the polynomial kernel.

### **Hyperparameters optimization: Particle Swarm Optimization (PSO)**

To calculate the hyperparameters for both the LSVM and the NLSVM, Particle Swarm Optimization (PSO) has been used. The LSVM and NLSVM hyperparameters are  $(C_{\text{LSVM}}, \epsilon_{\text{LSVM}})$  and  $(C_{\text{NLSVM}}, \epsilon_{\text{NLSVM}}, \sigma)$ , respectively. PSO is an optimization method that optimizes a candidate solution iteratively with regard to the given cost or merit function [198, 199]. To train the SVM models, a total of 70 engine operating points were available for the University of Alberta engine presented in Section 2.1. Then 80% of the data was used for training, 10% for cross-validation, and 10% as test data. Cross-validation data is used to tune the hyperparameters of the optimization methods. The cost function to find the LSVM and NLSVM is defined based on the Mean Square Error (MSE) of training and cross validation datasets. Hence, the hyperparameter calculation is defined as the following optimization problem:

$$\begin{aligned}
[C_{\text{LSVM}}, \epsilon_{\text{LSVM}}] &= \arg \min \left( \frac{1}{n_{tr}} \sum_{i=1}^{n_{tr}} (z_{tr,i} - y_{tr,i})^2 \right. \\
&\quad \left. + \frac{1}{n_{cv}} \sum_{i=1}^{n_{cv}} (z_{cv,i} - y_{cv,i})^2 \right) \\
[C_{\text{NLSVM}}, \epsilon_{\text{NLSVM}}, \sigma] &= \arg \min \left( \frac{1}{n_{tr}} \sum_{i=1}^{n_{tr}} (z_{tr,i} - y_{tr,i})^2 \right. \\
&\quad \left. + \frac{1}{n_{cv}} \sum_{i=1}^{n_{cv}} (z_{cv,i} - y_{cv,i})^2 \right)
\end{aligned} \tag{5.34}$$

Where  $C_{\text{LSVM}}$  and  $C_{\text{NLSVM}}$  are the regulatory parameters for linear SVM and non-linear kernel SVM, respectively. The index  $tr$  and  $cv$  represent training and cross-validation data set and  $n$  denotes number of data points in the data-set (i.e.  $n_{tr}$  is number of training data points). The tolerated error for linear SVM is  $\epsilon_{\text{LSVM}}$  and for nonlinear kernel SVM is  $\epsilon_{\text{NLSVM}}$ . The target data and prediction data are illustrated by  $z$  and  $y$ , respectively and  $\sigma$  is the Gaussian variance of RBF kernel. The PSO algorithm was used to solve for the hyperparameters. The PSO-based SVM algorithm is shown in Algorithm 1 and Algorithm 2 for linear and RBF kernel of SVM, respectively. The number of particles in the swarm set for both the LSVM and NLSVM model is set to 200 while the maximum iteration number is limited to 400 and 600 for LSVM and NLSVM, respectively.

### 5.2.2 Artificial Neural Network (ANN)

To compare the developed SVM to a traditional ML method, the proposed methods will be compared to the conventional ANN methods presented in literature. A feed-forward backpropagation network with single hidden layer and 15 neurons in each hidden layer using Levenberg-Marquardt backpropagation training method has been used in this study. This model with the same structure and number of neurons was previously developed for a single cylinder HCCI Ricardo engine [120]. This is a relatively shallow network which was chosen as there is a limited amount of data

---

**Algorithm 1:** PSO-based linear kernel SVM algorithm

---

**Result:** HCCI emission model:  $\mathbf{y}(\mathbf{u})$   
training data set:  $\{\mathbf{u}, \mathbf{z}\}$ ;  
splitting data set: training  $\{\mathbf{u}_{tr}, \mathbf{z}_{tr}\}$ , cross-validation  $\{\mathbf{u}_{cv}, \mathbf{z}_{cv}\}$ , and test  $\{\mathbf{u}_{ts}, \mathbf{z}_{ts}\}$   
;  
Random hyperparameters:  $C_{\text{LSVM}}, \epsilon_{\text{LSVM}}$ ;  
Run Quadratic Programming of Eq. 5.25 to calculate  $\alpha^-$  and  $\alpha^+$ ;  
Calculate support vector sets based on Eq. 5.30;  
Calculate model based on random hyperparameters using Eq. 5.31;  
Set PSO options:  $\text{MaxIterations}^1$ ,  $\text{MaxStallIterations}^2$ ,  $\text{FunctionTolerance}^3$ ,  
and  $\text{SwarmSize}^4$ ;  
**while**  $i \in \text{MaxIterations}$  or  $d_s$  over  $\text{MaxStallIterations} \geq$   
 $\text{FunctionTolerance}$  **do**  
    Calculate cost function:  
     $J(C_{\text{LSVM}}, \epsilon_{\text{LSVM}}) = \left(\frac{1}{n_{tr}} \sum_{i=1}^{n_{tr}} (z_{tr,i} - y_{tr,i})^2 + \frac{1}{n_{cv}} \sum_{i=1}^{n_{cv}} (z_{cv,i} - y_{cv,i})^2\right)$ ;  
    Run PSO algorithm to minimize  $J(C_{\text{LSVM}}, \epsilon_{\text{LSVM}})$  and find hyperparameters;  
    Update hyperparameters:  $C_{\text{LSVM}}, \epsilon_{\text{LSVM}}$ ;  
    Run Quadratic Programming of Eq. 5.25 to calculate  $\alpha^-$  and  $\alpha^+$ ;  
    Calculate support vector sets based on Eq. 5.30;  
    Calculate model based on Updated hyperparameters using Eq. 5.31;  
     $i = i + 1$   
**end**

1. Maximum number of iterations for optimization (= 400), 2. Positive integer (= 20), 3. non-negative scalar: Iterations end when the relative change in cost function value over the last  $\text{MaxStallIterations}$  iterations is less than  $\text{FunctionTolerance}$  (=  $1e - 6$ ), 4. Number of particles in the swarm (= 200), 5. Relative change
- 

available. The model training has been completed using the same parameters as used in [120].

### 5.2.3 Feature Selection: Physical insights

A steady state emissions model is developed to predict the steady-state HCCI engine emissions values of carbon dioxide ( $\text{CO}_2$ ), carbon monoxide (CO), unburnt hydrocarbons (HC), and nitrogen oxides ( $\text{NO}_x$ ). The structure of the model is defined by Eq. (5.18) where  $\mathbf{w}$  and  $\mathbf{b}$  are obtained by solving the SVM algorithm for a given training data set,  $\{\bar{\mathbf{u}}, \mathbf{z}\}$ . Here,  $\bar{\mathbf{u}}$  is the normalized Feature Set (FS). In total five different feature sets are tested. The training target set,  $\mathbf{z}$ , is defined based on measured steady-state  $\text{CO}_2$ , CO, HC, and  $\text{NO}_x$  values. To develop the model, 70 experimental

---

**Algorithm 2:** PSO-based RBF kernel SVM algorithm

---

**Result:** HCCI emission model:  $\mathbf{y}(\mathbf{u})$   
training data set:  $\{\mathbf{u}, \mathbf{z}\}$ ;  
splitting data set: training  $\{\mathbf{u}_{tr}, \mathbf{z}_{tr}\}$ , cross-validation  $\{\mathbf{u}_{cv}, \mathbf{z}_{cv}\}$ , and test  $\{\mathbf{u}_{ts}, \mathbf{z}_{ts}\}$   
;  
Random hyperparameters:  $C_{LSVM}, \epsilon_{LSVM}, \sigma$ ;  
Run Quadratic Programming of Eq. 5.25 to calculate  $\alpha^-$  and  $\alpha^+$ ;  
Calculate support vector sets based on Eq. 5.30;  
Calculate model based on random hyperparameters using Eq. 5.32 and 5.33;  
Set PSO options: *MaxIterations*<sup>1</sup>, *MaxStallIterations*<sup>2</sup>, *FunctionTolerance*<sup>3</sup>,  
and *SwarmSize*<sup>4</sup>;  
**while**  $i \in \text{MaxIterations}$  or  $d_s$  over  $\text{MaxStallIterations} \geq$   
*FunctionTolerance* **do**  
    Calculate cost function:  
     $J(C_{LSVM}, \epsilon_{LSVM}, \sigma) = \left(\frac{1}{n_{tr}} \sum_{i=1}^{n_{tr}} (z_{tr,i} - y_{tr,i})^2 + \frac{1}{n_{cv}} \sum_{i=1}^{n_{cv}} (z_{cv,i} - y_{cv,i})^2\right)$ ;  
    Run PSO algorithm to minimize  $J(C_{LSVM}, \epsilon_{LSVM}, \sigma)$  and find  
    hyperparameters;  
    Update hyperparameters:  $C_{LSVM}, \epsilon_{LSVM}, \sigma$ ;  
    Run Quadratic Programming of Eq. 5.25 to calculate  $\alpha^-$  and  $\alpha^+$ ;  
    Calculate support vector sets based on Eq. 5.30;  
    Calculate model based on Updated hyperparameters using Eq. 5.32 and 5.33;  
     $i = i + 1$   
**end**

1. Maximum number of iterations for optimization (= 600), 2. Positive integer (= 20), 3. non-negative scalar: Iterations end when the relative change in cost function value over the last *MaxStallIterations* iterations is less than *FunctionTolerance* (=  $1e-6$ ), 4. Number of particles in the swarm (= 200), 5. Relative change
- 

data points are available where 56 points (80 %) are used to train the model and 14 (20 %) points to test the model.

Due to the lack of direct ignition control in HCCI, unlike conventional spark ignition in gasoline engines, the start of combustion depends on the in-cylinder conditions including pressure, temperature and gas mixture. However, these factors can only be influenced indirectly. The inputs used to set cylinder conditions and therefore affect the combustion are: Negative Valve Overlap (NVO) in CAD; Injected Fuel Mass per cycle ( $m_f$ ) in mg; and Start of Injection (SOI) in deg aTDC.

Symmetric NVO is used to change the percentage of fresh air and exhaust gas within the cylinder, called internal Exhaust Gas Recirculation (EGR). This changes both the amount of oxygen in the cylinder as well the temperature of the cylinder

charge. Generally, a lean mixture is desired for reduced  $NO_x$  emissions, however, at very lean mixtures the fuel flammability limit of the fuel may be exceeded leading to combustion instability which results in high cyclic variability and increased HC emissions. NVO also impacts the cylinder temperature after compression, where a higher cylinder temperature results in an earlier auto-ignition process. The injected fuel mass directly changes the amount of fuel that is added to the cylinder. However, it is important to note that a fraction of the unburnt fuel is transferred between cycles due to the trapped EGR within the cylinder. The amount of transferred fuel changes depending on the combustion efficiency of the last cycle and the EGR rate (determined by the NVO duration) [84]. The start of injection impacts the mixture homogeneity and can lead to stratified mixtures. This can change the start of combustion and the emissions levels. To test a wide range of cylinder conditions before combustion these parameters were varied (NVO,  $m_f$ , and SOI).

Additional operating factors that are held as constant as possible are: intake temperature ( $T_{in}$ ); intake pressure ( $P_{in}$ ); and indicated mean effective pressure (IMEP) which is representative of applied engine load. IMEP,  $T_{in}$ , and  $P_{in}$  are variable inputs to the HCCI process but they were held constant for this measurement set to reduce the number of input parameters. However, these three factors ( $T_{in}$ , and  $P_{in}$ ) are included to account for unwanted fluctuations. The measured lambda value ( $\lambda$ ) is used for modeling, although, it can also be accurately estimated using measured intake air flow and injected fuel demand or calculated using an online gas exchange model making it a causal variable which is useful for future control applications [84].

Now different feature sets are used to train data driven models. The first feature set uses 7 inputs to create a linear model, L7. The first extension of this feature set is considering cross correlations between the variables ( $m_f \times NVO$ ,  $m_f \times SOI$ ,  $NVO \times SOI$ ,  $\lambda \times NVO$ ,  $\lambda \times SOI$ ,  $\lambda \times m_f$ ) resulting in the L13 feature set. The cross correlations with  $T_{in}$ ,  $P_{in}$ , and IMEP are not taken into account as to not over interpret the effect

of possible fluctuations. Then higher order correlations are also considered by adding the squares of the input variables, FS S14. Additionally, 2 more FS are added (S20 and S26) that consider the square of the cross correlations. Details of the five FS's can be found in table 5.2. From a machine learning point of view, these FS's plays the exact role of a polynomial feature set. The only difference is that the redundant higher dimensional feature has been removed based on physical insight expertise.

Table 5.2: Features  $u_1 - u_{26}$  for the five different feature sets L7 – S26.  $u_1 - u_7$  are linear features,  $u_8 - u_{14}$  are squared features,  $u_{15} - u_{20}$  are cross correlations, and  $u_{21} - u_{26}$  are the squared cross correlations. L stands for linear and S stands for squared

[Features $u_1 - u_{26}$ for the five different feature sets L7 – S26]					
name →	L7	L13	S14	S20	S26
feature ↓					
$u_1 = m_f$	x	x	x	x	x
$u_2 = \text{NVO}$	x	x	x	x	x
$u_3 = \text{SOI}$	x	x	x	x	x
$u_4 = T_{\text{in}}$	x	x	x	x	x
$u_5 = P_{\text{in}}$	x	x	x	x	x
$u_6 = \text{IMEP}$	x	x	x	x	x
$u_7 = \lambda$	x	x	x	x	x
$u_8 = m_f^2$			x	x	x
$u_9 = \text{NVO}^2$			x	x	x
$u_{10} = \text{SOI}^2$			x	x	x
$u_{11} = T_{\text{in}}^2$			x	x	x
$u_{12} = P_{\text{in}}^2$			x	x	x
$u_{13} = \text{IMEP}^2$			x	x	x
$u_{14} = \lambda^2$			x	x	x
$u_{15} = m_f \times \text{NVO}$		x		x	x
$u_{16} = m_f \times \text{SOI}$		x		x	x
$u_{17} = \text{NVO} \times \text{SOI}$		x		x	x
$u_{18} = \lambda \times \text{NVO}$		x		x	x
$u_{19} = \lambda \times \text{SOI}$		x		x	x
$u_{20} = \lambda \times m_f$		x		x	x
$u_{21} = (m_f \times \text{NVO})^2$					x
$u_{22} = (m_f \times \text{SOI})^2$					x
$u_{23} = (\text{NVO} \times \text{SOI})^2$					x
$u_{24} = (\lambda \times \text{NVO})^2$					x
$u_{25} = (\lambda \times \text{SOI})^2$					x
$u_{26} = (\lambda \times m_f)^2$					x

As the dimensions and the range of the features are quite different, all of the features should be normalized to improve the training performance [200]. Here the

min-max normalization method is used to normalize the features

$$\bar{\mathbf{u}} = \frac{\mathbf{u} - \min(\mathbf{u})}{\max(\mathbf{u}) - \min(\mathbf{u})} \quad (5.35)$$

All of features from Table 5.2 are normalized for ANN and SVM methods to eliminate relative orders of magnitude difference between the features. By solving the SVM algorithm for the training data set,  $\{\bar{\mathbf{u}}, \mathbf{z}\}$ , the approximate function,  $\mathbf{y}_{ss}$  is obtained to predict the steady-state values of CO<sub>2</sub>, CO, HC, and NO<sub>x</sub>.

### 5.2.4 ML Emission Model Comparison

To illustrate the method, the model for the CO emissions will be discussed in detail with the other emissions being similar. The recorded data points are randomly split into three sections where 80% of the collected data is used as training data to develop the models. Then 10% of the data is used for model cross-validation. The remaining 10% of the data is called test data and is allocated for assessment of the models where the same data is used for assessing all models including LSVM, NLSVM, and ANN. To do this, the randomly chosen data points for each of the three data sets is then kept constant between all models and feature sets to allow for a fair comparison. Training and cross-validation data sets are used to train the model and calculate hyperparameters. In LSVM and NLVM, the PSO algorithm is employed to calculate the hyperparameters by solving the optimization problem of Eq. (5.34). The same training and cross-validation data are used to train an ANN model using Levenberg-Marquardt algorithm.

To rate the model quality the coefficient of determination ( $R^2$ ) is used. It is defined by

$$R^2 = 1 - \frac{\sum(z_i - y_i)^2}{\sum(z_i - \bar{z})^2} \quad (5.36)$$

with  $z_i$  being a measured value in the data set,  $y_i$  being the models response to the accompanying  $z_i$  and  $\bar{z}$  being the mean of the measured data. A value of  $R^2$



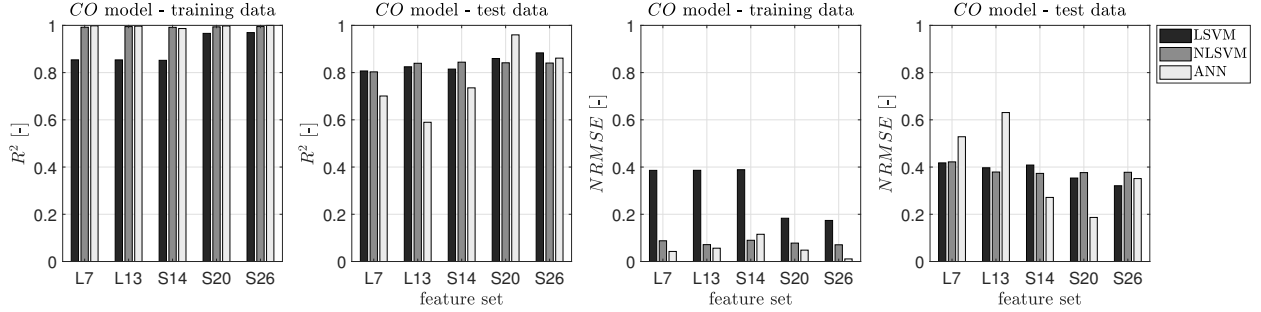


Figure 5.9: Comparison between  $R^2$  and Normalized RMSE values for CO for NLSVM and LSVM with benchmark ANN method designed based on [120] in dependence of the different feature sets.

value closer to 1 is judged as a better model fit to the data. The  $R^2$  estimate of the relationship between the dependent variables based on an independent variable may fail to tell the goodness of fit. Therefore, the Normalized Root Mean Square Error (NRMSE) is used to capture the error between the model and actual values. The Normalized version of RMSE is used to remove the dependency of RMSE to scale output and generalize the model easily. NRMSE is defined by  $\frac{RMSE}{\sigma}$  where  $\sigma$  is the standard deviation and  $RMSE$  is defined as  $RMSE = \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y})^2}{n}}$  where  $y_i$  is experimental value and  $\hat{y}$  is predicted value. This criteria provides a good representation of how far the model prediction is away from the real data. Therefore, the lower the NRMSE the closer the model is to the real value. Both of these methods help to quantify the model fit.

Figure 5.9 shows both the  $R^2$  and NRMSE values for the training and test data for the CO model. As expected the  $R^2$  and NRMSE values are the best for the training data as the models were trained on this data set. As the model has never been trained on the testing data this reduced prediction accuracy is expected and provides the best representation of the model fit.

When comparing the coefficients of determination ( $R^2$ ) of the LSVM, NLSVM and ANN models in Figure 5.9 a few key differences can be seen. First, when only considering the training data the NLSVM and ANN models result in an improved  $R^2$

value over the simplified LSVM model. Although this does not result in a significantly improved model prediction performance when given the test data. Actually, the LSVM outperforms the ANN model in most feature sets. Showing that the ANN model can suffer from over fitting which is not seen with the simple LSVM model when presented with unknown training data. This problem with a small network, such as conventional ANN, can be reduced when using large datasets; however, when limited data is available SVM shows a better prediction capability [117].

When comparing the different feature sets, all the three models result in fairly consistent prediction accuracy irrespective of the number of features. This is especially true for both linear and non-linear SVM models which only vary by 12% and 14%  $R^2$  as the number of features is increased from 7 to 26. This is likely the result of the SVM algorithm always converging to the global minimum while the ANN model can converge to a local minimum as seen by the decrease in ANN model performance going from L7 to L13. The convergence of the ANN model is highly dependent of the initial choice of weights and bias values. This guarantee of global convergence is one the major advantages of the SVM method [115, 117, 134, 201]. The main reason for global optimization is that SVM uses Quadratic programming, which includes optimizing a function according to linear constraints. As ANN uses Gradient descent, it makes ANN sensitive to randomization of weights parameters. This means that if initial weights put cost function close to a local minimum, the accuracy of the model will never increase past a certain threshold [131]. To avoid this, each ANN model is trained in a loop with multiple randomization values, where the randomization is reset until it reaches acceptable accuracy.

The findings from CO emissions can then be extended to HC,  $\text{NO}_x$ , and  $\text{CO}_2$  as shown in Figures 5.10-5.12. The trends seen between modeling methods vary slightly between specific emissions as expected due to the physical differences in their production mechanism. To do this, a Criterion for Methods Selection (CMS),  $J_{CMS}(R^2)$ , is

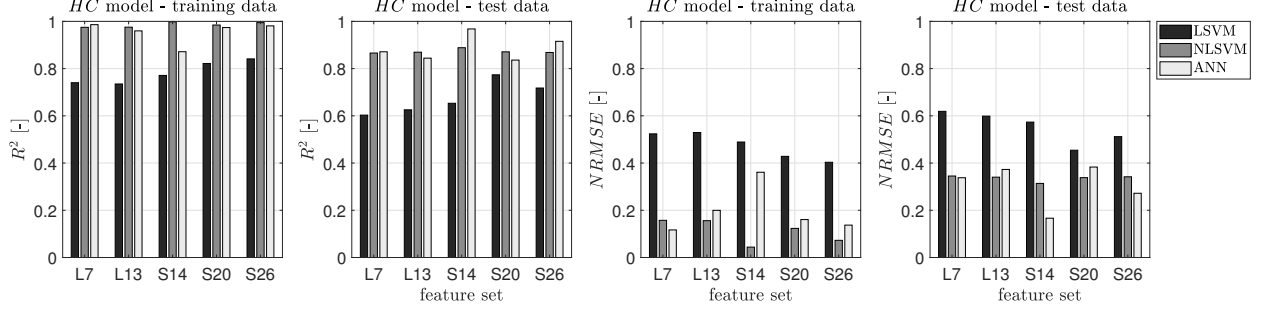


Figure 5.10: Comparison between  $R^2$  and NRMSE values for HC for NLSVM and LSVM with benchmark ANN method designed based on [120] in dependence of the different feature sets.

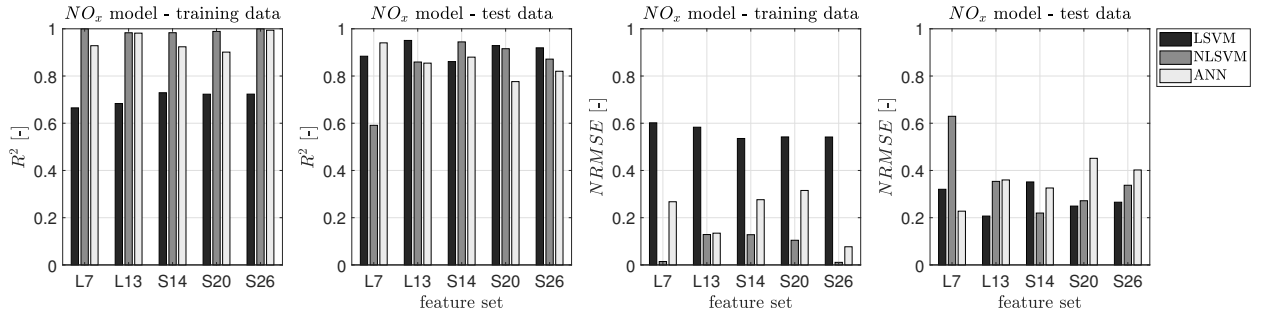


Figure 5.11: Comparison between  $R^2$  and NRMSE values for  $\text{NO}_x$  for NLSVM and LSVM with benchmark ANN method designed based on [120] in dependence of the different feature sets.

defined as

$$J_{CMS}(R^2) = \bar{R}_{(FS)}^2 - \sigma(R_{(FS)}^2) \quad (5.37)$$

where  $\sigma(R_{(FS)}^2)$  is standard deviation of  $R^2$  and  $\bar{R}_{(FS)}^2$  is average value of  $R^2$  for selected feature set, L7, L13, S14, S20, and S26. Table 5.3 shows criterion for method selection,  $J_{CMS}(R^2)$ . This represents the lower bound of one standard deviation of uncertainty of the model fit. This helps to select a model with the best fit while ensuring the robustness of the model to changing feature sets. The goal is to have the value close to 1. Here the best model fit score is highlighted in green and the worst is shown in red.

Here 3 of the 4 emissions are best represented using the NLSVM model and the other is best fit using LSVM. This shows that the SVM based models provide a stable

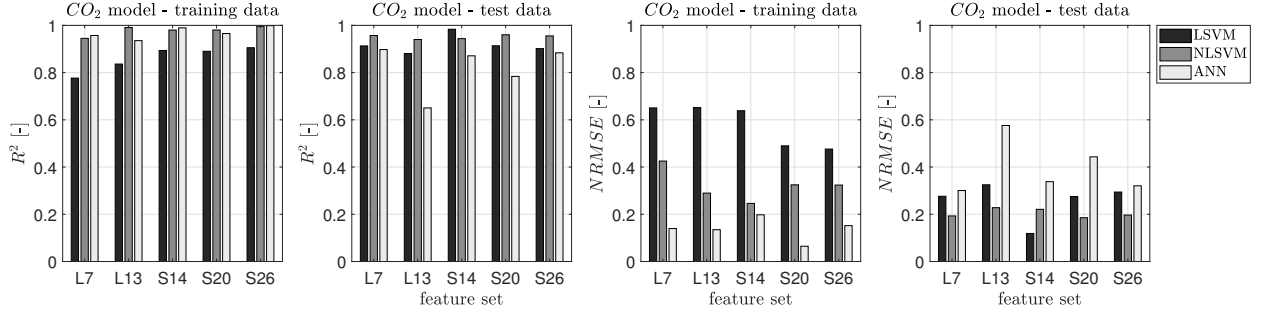


Figure 5.12: Comparison between  $R^2$  and NRMSE values for  $\text{CO}_2$  for NLSVM and LSVM with benchmark ANN method designed based on [120] in dependence of the different feature sets.

Table 5.3: Criteria for method selection

$J_{CMS}(R^2)$	LSVM	NLSVM	ANN
CO	0.809	0.818	0.641
HC	0.612	0.864	0.838
$\text{NO}_x$	0.877	0.710	0.799
$\text{CO}_2$	0.884	0.944	0.725

prediction over the range of feature sets considered. A detailed analysis of the feature set will be performed next.

## 5.2.5 Feature Selection

One important aspect to training the ML methods is the proper feature selection. It is important to include any features that have a correlation to the outputs of interest. However, the addition of extra features increase the model complexity and training time which is undesirable for real-time model implementation. Figure 5.9 shows the effect of feature selection on the model performance for CO emissions. The number of features in each feature set increases from left to right.

The best  $R_{train}^2$  value in all cases occurs for FS L7 ( $R_{ANN}^2 = 0.999$ ), while  $R_{test}^2$  is maximized at S20 ( $R_{ANN}^2 = 0.959$ ) for the ANN model. The  $R^2$  values are all very close for the training data at approximately  $R^2 \approx 0.98$ , however, a significant

difference can be seen between the  $R^2$  values of the test dataset. Generally as more features are added model performance improves as seen in Figure 5.9 in the test data for the ANN model. As the feature set is increased from L7 to S26 a continued increase can be seen, with the exception of L13 which has a decreased model performance with the training data using the ANN model. Improved model performance does not necessarily result from increased features.

For CO emissions the best model performance on the test data occurs when using the ANN model with S20 feature set. However, for simplified control purposes the L7 feature set using the NLSVM model provides good a prediction capability with only a 15.6% reduction in model fit,  $R_{test}^2$ . As the main goal is to provide a real-time model for control applications this simplified and robust NLSVM prediction model is the desired choice for CO emissions prediction.

This feature analysis was extended to HC,  $\text{NO}_x$ , and  $\text{CO}_2$  and is shown in Figures 5.10–5.12. To compare the increased feature sets to the base feature set ( $FS = L7$ ) a percent accuracy increase in  $R^2$  value is defined, as Criterion for Feature Selection (CFS),  $J_{CFS}(R^2)$ , as:

$$J_{CFS}(R^2) = \frac{R_{FS}^2 - R_{L7}^2}{R_{L7}^2} \times 100\% \quad (5.38)$$

This provides the relative increase in performance compared to the simplest model, which includes lambda as a feature, for the model type selected previously. Table 5.4 shows the improvement based on different feature sets. Here the simplest model is chosen that provides a significant increase in prediction performance ( $\Delta R^2 > 2\%$ ).

Overall, proper feature selection is required to gain the maximum model performance. This does not mean including any and all features but rather a proper feature exploration and selection is required. A suitable trade-off between accuracy and model complexity results in the emission model for control purposes for CO, HC,  $\text{NO}_x$ , and  $\text{CO}_2$  are NLSVM-L13, NLSVM-L7, LSVM-L13, and NLSVM-L7, respectively. For

Table 5.4: Criteria for feature selection

	feature set	NLSVM- CO	NLSVM- HC	LSVM- NO <sub>x</sub>	NLSVM- CO <sub>2</sub>
$J_{CFS}(R^2)$	<b>FS = L13</b>	4.53%	0.41%	7.56%	-1.75%
	<b>FS = S14</b>	5.12%	2.60%	-2.57%	-1.38%
	<b>FS = S20</b>	4.78%	0.59%	5.09%	0.34%
	<b>FS = S26</b>	4.66%	0.28%	4.02%	-0.17%
	<b>chosen FS</b>	<b>L13</b>	<b>L7</b>	<b>L13</b>	<b>L7</b>

this system, more features does not necessarily result in better model prediction performance. Additionally, based on the data collected there is not a single modeling method that should be used for all emissions.

### 5.2.6 Optimization and model training time

As the purpose of the proposed emissions models is hardware implementation, it is necessary to evaluate the computational time requirements. There are two different computational times of interest: 1) the offline one-time model training time and 2) the online deployment time to execute the model. The training computational time for optimization of the hyperparameters is shown in Figure 5.13a and the evaluation of the model based on the optimized hyperparameters for the *CO* model is shown in Figure 5.13b. As shown in Figure 5.13a, PSO-based NSVM requires more optimization time than LSVM. Part of this increase is because more optimization variables need to be determined using PSO compared with LSVM. As shown, ANN has a optimization time that is between NSVM and LSVM. For the ANN model, the optimization time includes multiple ANN training runs to reduce the effect of the randomized starting weights as described in Section 5.2.4. However, in addition to this optimization time the ANN model also requires a grid search between the number of neurons and the hidden layer size result in a significant computation time. However, as in this thesis, the structure of the ANN is chosen based on a benchmark model for comparison purposes based on [120] so a grid search is not required. The optimization part of

modeling, even for the ANN grid search, does not affect the real-time implementation for two main reasons: 1) in real-time, only the already trained model is evaluated, 2) even with online learning, i.e., updating model in real-time, the model will be updated based on optimized hyperparameters.

The model evaluation time is based on pre-optimized hyperparameters and this evaluation time plays a crucial role in real-time implementation. As shown in Figure 5.13b, LSVM needs 67% and 32% lower evaluation time compared to ANN and NLSVM, respectively. NLSVM also takes 52% lower computation time than ANN. These results can be repeated for the other  $NO_x$ ,  $CO_2$ , and  $HC$  models which result in an average reduction in evaluation time for the LSVM model of 64% and 28% compared to the ANN and NLSVM models, respectively. On average for the 4 emissions the NLSVM requires 45% lower evaluation time than the ANN model.

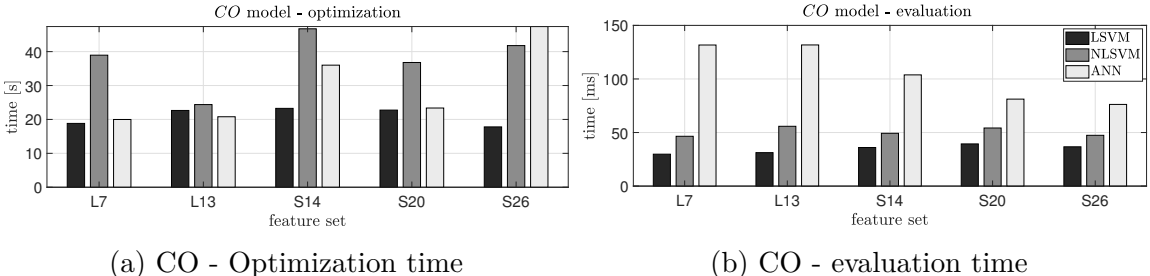


Figure 5.13: Optimization and evaluation time comparison between LSVM, NSVM, and ANN on an Intel i7-8700K PC

### 5.2.7 Chosen model performance

The model type and feature set selected in the previous sections for each of the four emissions are evaluated compared against the experimental data. Figures 5.14a-5.14d show the prediction performance of the selected models along with a  $\pm 5\%$  band shown in red.

The  $CO_2$  model has all predicted values within the  $\pm 5\%$  error bands. For the CO, HC and  $NO_x$  models there is 56%, 97%, and 56% of the data points within the

error bands, respectively. For the CO model there is a relatively large spread in the cross-validation and test data. However, as there is a large spread in the CO levels over the testing data points the model is still able to provide the modeling trends.

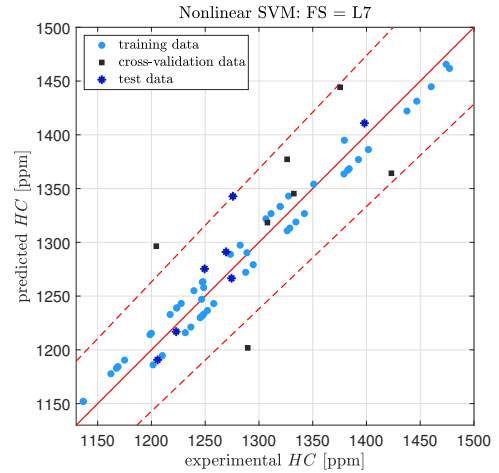
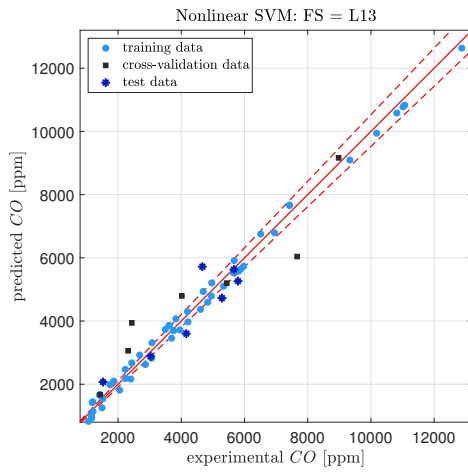
The NO<sub>x</sub> model has a larger spread in all of the data points. This could be a result of the low level of NO<sub>x</sub> emissions from 35-70 ppm and the stochastic variation in the HCCI combustion that is not captured in the steady state modeling. As only a single cycle or a few cycles within a measurement can greatly increase the average emissions levels a transient emission model is needed.

The effect of different machine learning approaches and feature sets on the model quality for HCCI emissions prediction was described in this section. The goal was to select an accurate and simple emission model for future real-time control implementation. First, linear and non-linear SVM models were compared to a traditional ANN model. This comparison showed for a small data set that SVM based models were more robust to changes in feature selection and better able to avoid local minimums compared to ANN leading to a more consistent model prediction. For each of the four emissions examined the best model type was determined by taking the highest average  $R^2$  value less the variance in  $R^2$  over the various feature sets. This led to the NLSVM being selected for 3 of the emissions and LSVM for NO<sub>x</sub> prediction.

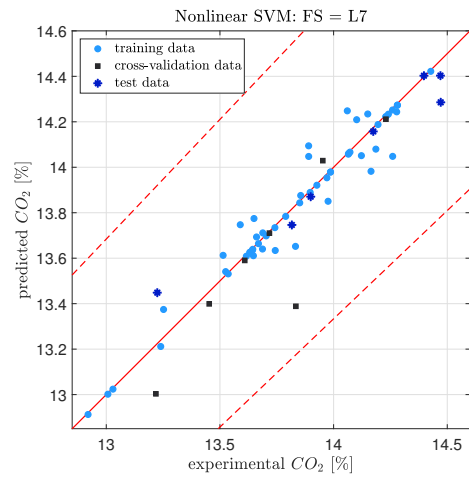
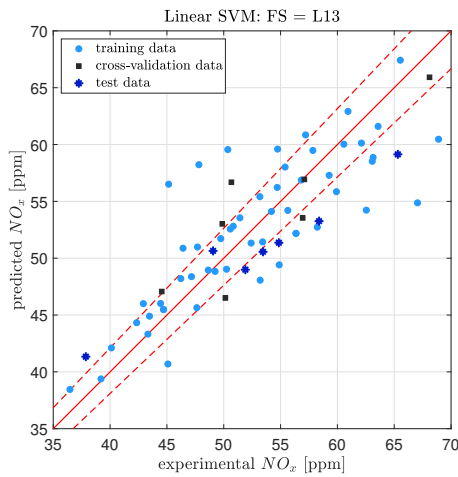
Then the individual feature sets were examined. The base feature sets were extended by multiplying individual features together to explore in-feature interactions. By comparing the individual features, with the base feature set (L7), the feature set with an improved accuracy taking into account the increase in model complexity was chosen. The steady-state emission models are chosen for control purposes with the goal of implementing in MPC. For CO, HC, NO<sub>x</sub>, and CO<sub>2</sub> the chosen models are NLSVM-L13, NLSVM-L7, LSVM-L13, and NLSVM-L7, respectively. The NO<sub>x</sub> and CO models have the largest prediction error while the HC and CO<sub>2</sub> models are quite accurate. The NO<sub>x</sub> model produced the least accurate results however it was still



able to capture the trends in  $\text{NO}_x$  production.



(a) CO - NLSVM - L13 - actual vs experiment (b) HC - NLSVM - L7 - actual vs experiment



(c) NOx - LSVM - L13 - actual vs experiment (d) CO2 - NLSVM - L7 - actual vs experiment

Figure 5.14: Actual vs experiment- HCCI emission model. Dashed red line represent  $\pm 5\%$  of experimental data value

## 5.3 Deep Neural Network - Long Short Term Memory: Transient Model

In this section, the dynamic response of an HCCI engine will be modeled using a Long Short-Term Memory (LSTM) based model. In this section IMEP, CA50, MPRR and  $\text{NO}_x$  concentration will be modeled. These are chosen as the first three are calculated cycle by cycle using the measured in-cylinder pressure on the FPGA (see Figure 1.7) and  $\text{NO}_x$  is measured using a fast  $\text{NO}_x$  sensor as described in Chapter 2.

The LSTM cell is the most well-known form of RNN with long-term memory cells that are able to predict outputs while considering a long-term dependency. In comparison to basic RNNs, LSTMs employ a hidden state that is divided into two components: i) the short-term state  $h(k)$ , and ii) the long-term state  $c(k)$  as shown in Fig. 5.15. The long-term state goes across the network and initially enters the forget gate and is multiplied by  $f(k)$ . Each time step adds new values (memories) to the input gate  $i(k)$ . As a result, some data is added and some is deleted at each time step [146].

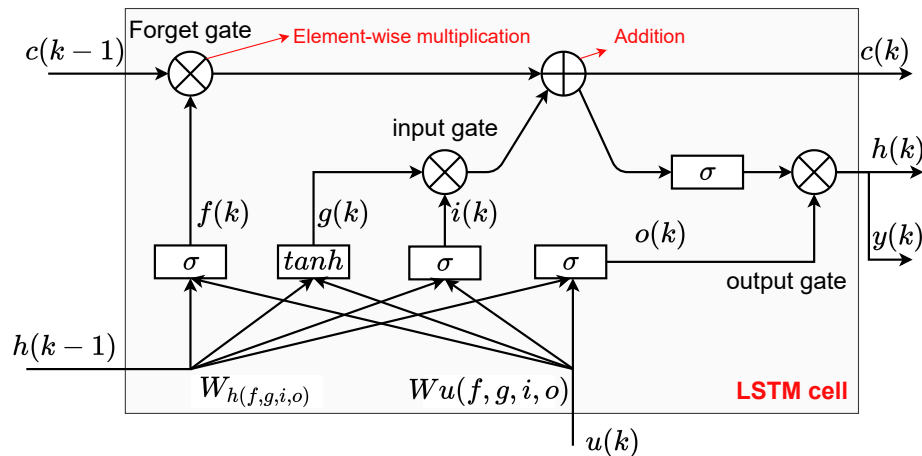


Figure 5.15: Long-Short-Term Memory (LSTM) cell structure schematics

To model the HCCI engine-out emissions, a deep neural network with seven hidden layers, including 6 Fully Connected (FC) layers and one LSTM layer, is proposed as shown in Fig. 5.16. The input of this model is the duration of injection for both fuel

( $DOI_{\text{fuel}}(k)$ ) and water injection ( $DOI_{\text{water}}(k)$ ), the negative valve overlap duration ( $NVO(k)$ ) which controls the amount of air in the cylinder, and two inputs from the previous cycle. These inputs are the indicated mean effective pressure  $IMEP(k-1)$  and the combustion phasing angle ( $CA50(k-1)$ ). These previous inputs are included in the model to provide a feedback effect of two of the measurable engine performance outputs. These two combustion metrics are chosen as IMEP is the requested load of the engine and the performance of the controller to track IMEP will be evaluated. CA50 is also an important combustion parameter is strongly related to the other combustion characteristics including MPRR, engine efficiency and emissions. The outputs of this model are the indicated mean effective pressure ( $IMEP(k)$ ), Nitrogen Oxides ( $NO_x(k)$ ), combustion phasing angle for 50% heat release ( $CA50(k)$ ) and Maximum Pressure Rise Rate ( $MPRR(k)$ ).

This model structure is based on preliminary diesel combustion engine control testing as well as for hydrogen diesel dual fuel operation [2, 202]. To capture the nonlinearity of the HCCI combustion process with the LSTM, more LSTM hidden units are needed which result in an increased number of hidden states and cell states. As the hidden and cell states are also states in the MPC optimization the increased number of states leads to an increased computational cost for the MPC controller to find an optimal solution during real-time implementation. In preliminary work, where an NMPC was implemented in real-time for diesel combustion control, doubling LSTM hidden units resulted in approximately doubling computational turnaround time from 63 ms to 130 ms. For engine control, this calculation time must be within one engine cycle time for cycle-by-cycle control [1]. Similarly, for HCCI combustion, it is necessary to keep the model calculation real-time capable. To minimize the number of LSTM hidden units, Fully Connected (FC) layers are added before and after the LSTM layer to boost the network's capacity for estimating the engine's nonlinearity without significantly increasing the number of hidden and cell states.

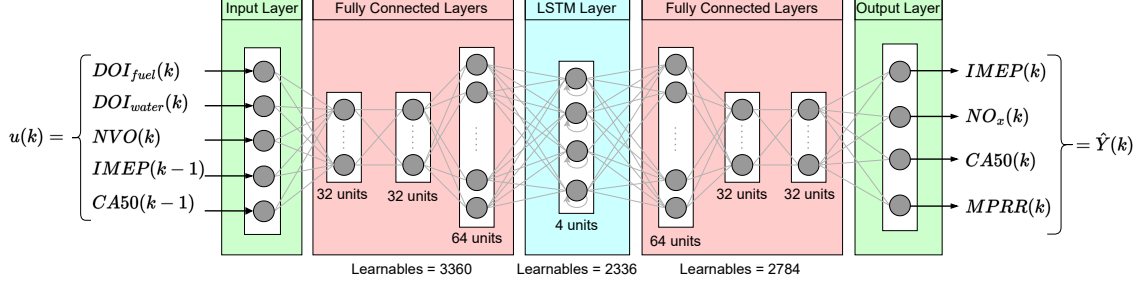


Figure 5.16: Structure of proposed deep neural network model for engine performance and emission modeling. LSTM: Long-short term memory, DOI: duration of injection, IMEP: indicated mean effective pressure, MPRR: maximum pressure rise rate, CA50: combustion phasing for 50% heat release

To use this network inside a Nonlinear MPC (NMPC), a function using forward propagation is needed. To perform forward propagation, first the LSTM and FC layers computations are evaluated. A computational graph (Fig. 5.17) clarifies how the equations of the model are obtained. The LSTM computations are

$$i(k) = \sigma (W_{u,i}^\top u(k) + W_{h,i}^\top h(k-1) + b_i) \quad (5.39a)$$

$$f(k) = \sigma (W_{u,f}^\top u(k) + W_{h,f}^\top h(k-1) + b_f) \quad (5.39b)$$

$$g(k) = \tanh (W_{u,g}^\top u(k) + W_{h,g}^\top h(k-1) + b_g) \quad (5.39c)$$

$$o(k) = \sigma (W_{u,o}^\top u(k) + W_{h,o}^\top h(k-1) + b_o) \quad (5.39d)$$

$$c(k) = f(k) \odot c(k-1) + i(k) \odot g(k) \quad (5.39e)$$

$$h(k) = o(k) \odot \tanh (c(k)) \quad (5.39f)$$

where  $(W_{u,(f,g,i,o)})$  are the weight matrices applied to the input vector  $u(k)$  and  $W_{h,(f,g,i,o)}$  are weight matrices of the previous short-term state  $h(k)$ . In this equation,  $\odot$ , is an element-wise multiplication and  $b_{(f,g,i,o)}$  are the biases. In Eq. 5.39,  $i(k)$  is the input gate,  $f(k)$  is the forget gate,  $g(k)$  is the cell candidate,  $o(k)$  is the output gate,  $c(k)$  is the cell state, and  $h(k)$  is the hidden state. Two activation functions are used in Eq. 5.39 which are given as: i)  $\tanh(z)$  activation function:

$$\tanh(z) = \frac{e^{2z} - 1}{e^{2z} + 1} \quad (5.40)$$

ii)  $\sigma(z)$  activation function:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \quad (5.41)$$

These activation functions are used to introduce non-linearity into the otherwise linear layers. A FC layer equation with Rectified Linear Unit (ReLU) activation function is defined as

$$z_{\text{FC}}(k) = \text{ReLU}(W_{\text{FC}}^T u(k) + b_{\text{FC}}) \quad (5.42)$$

where ReLU activation function is defined as

$$\text{ReLU} = \begin{cases} 0 & \text{if } z \leq 0 \\ z & \text{if } z > 0 \end{cases} \quad (5.43)$$

The computational graph of this part of the network is schematically depicted in Figure 5.17. Based on this figure and using Eq. 5.39 and Eq. 5.42 the model equations are:

$$z_{\text{FC1}}(k) = \text{ReLU}(W_{\text{FC1}}^T u(k) + b_{\text{FC1}}) \quad (5.44a)$$

$$z_{\text{FC2}}(k) = \text{ReLU}(W_{\text{FC2}}^T z_{\text{FC1}}(k) + b_{\text{FC2}}) \quad (5.44b)$$

$$z_{\text{FC3}}(k) = \text{ReLU}(W_{\text{FC3}}^T z_{\text{FC2}}(k) + b_{\text{FC3}}) \quad (5.44c)$$

$$z_{\text{FC4}}(k) = \text{ReLU}(W_{\text{FC4}}^T h(k) + b_{\text{FC4}}) \quad (5.45a)$$

$$z_{\text{FC5}}(k) = \text{ReLU}(W_{\text{FC5}}^T z_{\text{FC4}}(k) + b_{\text{FC5}}) \quad (5.45b)$$

$$y(k) = W_{\text{FC6}}^T z_{\text{FC5}}(k) + b_{\text{FC6}} \quad (5.45c)$$

where  $W_{\text{FC},i}$  and  $b_{\text{FC},i}$  are the weights and biases of the fully connected layer where  $i \in \{1, 2, 3, 4, 5, 6\}$  and  $W_{u,(f,g,i,o)}$  are the weight matrices of the input vector  $u(k)$  and  $W_{h,(f,g,i,o)}$  are the weight matrices to the previous short-term states  $h(k)$ .

For training with experimental data, a standard cost function [146] of this network is defined as

$$J(W, b) = \frac{1}{m} \sum_{k=1}^m \mathcal{L}(\hat{y}(k), y(k)) + \frac{\lambda}{2m} \sum_{l=1}^L \|W^{[l]}\|_2^2 \quad (5.46)$$

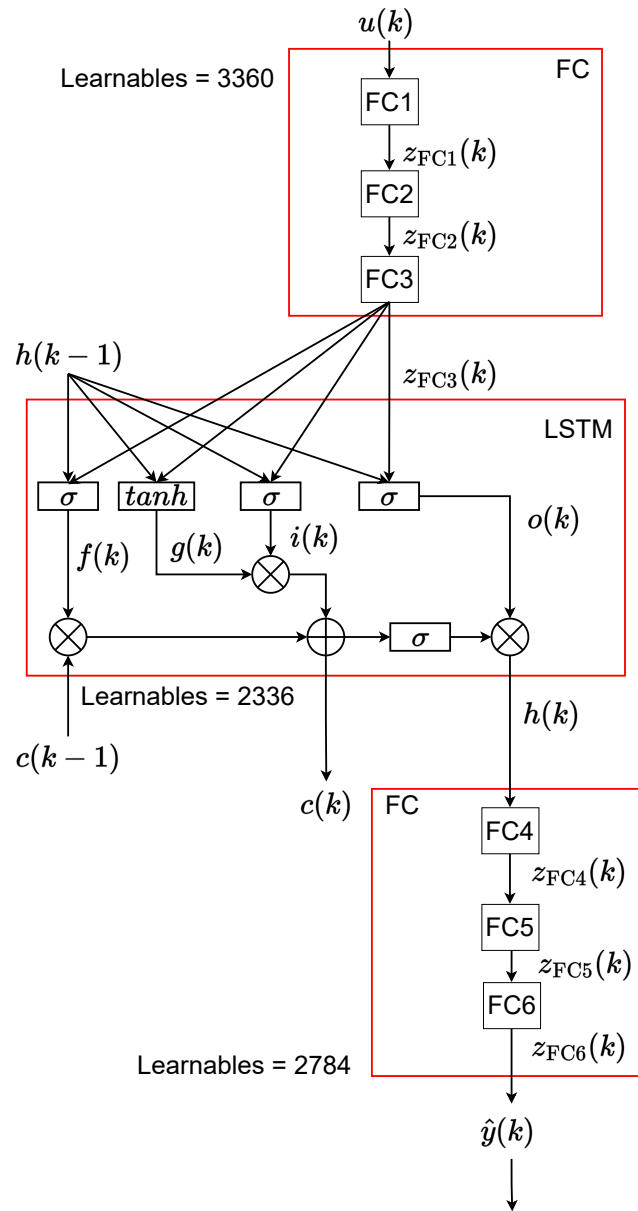


Figure 5.17: Computational graph of proposed deep network– FC: Fully Connected, LSTM: Long-Short Term Memory

where  $\mathcal{L}(\hat{y}(k), y(k))$  is the loss function,  $m$  is number of data points,  $\hat{y}(k)$  is measured output and  $y(k)$  is predicted output. A Mean Squared Error (MSE) loss function is used that is

$$\mathcal{L}(\hat{y}(k), y(k)) = \frac{1}{m} \sum_{k=1}^m (\hat{y}(k) - y(k))^2 \quad (5.47)$$

In Eq. 5.46,  $\lambda$ , is the regularization coefficients and  $\|W^{[l]}\|_2^2$  is the Euclidean norm which is defined as

$$\|W^{[l]}\|_2^2 = \sum_{i=1}^{n^{[l]}} \sum_{j=1}^{n^{[l-1]}} (w_{ij}^{[l]})^2 \quad (5.48)$$

### 5.3.1 Experimental Data Generation

To develop this deep network, which has 8480 learnable parameters, a large data set was collected using the RWTH engine (details in Section 2.2). The experimental single cylinder HCCI engine was run for 65,000 consecutive cycles, and the inputs including DOI of fuel, DOI of water, and NVO duration are changed randomly using a Pseudo-random binary sequence (PRBS). This PRBS is used to change both the amplitude and frequency of these three combustion inputs. However, as HCCI is a highly sensitive combustion process, the inputs could not be changed completely randomly over the entire operating range of interest (2-4 bar IMEP). Therefore the bounds of the PRBS signal are manually restricted during data collection to ensure that combustion is possible. For example as the range on fuel DOI is increased the NVO duration is decreased to allow more air into the cylinder to mix with the additional fuel and the range on water DOI is increased to help reduce the maximum pressure rise rate. This prevents combinations of a long NVO duration (minimum intake air) and maximum fuel injection which would result in undesirable rich combustion. This also prevents a large water injections when the engine is operating at very low fuel amounts and subsequently low load. At these low loads, a large water injection would completely extinguish the combustion. Using previous engine map-



ping experience, bounds on the PRBS signal were defined over a wide range of inputs while ensuring viable HCCI combustion [3, 15]. The PRBS inputs to the SCORE for data generation are plotted verses engine cycle in Figure 5.18a.

The resulting engine performance and emission outputs from the PRBS input to the system for the 65,000 cycles can be seen in Figure 5.18b. The engine outputs show the expected relations such as the increase in  $\text{NO}_x$  emissions as the IMEP and MPRR increases. When looking at all 65,000 cycles it is clear that there is a significant variation in the engine outputs resulting from the PRBS input as well as from the inherent stochastic variation in the HCCI combustion process.

To better visualize the variation in engine inputs and to confirm that an adequately large input and frequency range to the HCCI process has been performed, a histogram of all HCCI inputs are shown in Figure 5.19. Here the both the variation in the value and the number of consecutive cycles the input is applied for is shown. For the fuel injection duration (shown in Figure 5.19a) the variation is approximately Gaussian centered at 0.7 ms. A similar distribution can be seen for the NVO duration (Figure 5.19e) where the spread of data is a little more evenly distributed for NVO durations between 180 and 230 CAD. The distribution of water injection amount is heavily skewed to low (or no) water injection as seen in Figure 5.19c. This is a result that water could not be randomly injected and for low engine loads no water was injected as for any random water injection durations below 0.85 ms the water injection was set to zero as the injector is not able to consistently open for extremely short durations. For the number of consecutive cycles for each input step the values vary evenly between 75 and about 200 cycles. It should be noted that each input was changed independently of each other so while the fuel injection was kept constant the NVO and water injection could also change. Therefore the number of cycles where all inputs remained constant was quite short, with an average of 48 cycles, as shown in Figure 5.20.

The resulting variation in the engine outputs can also be shown in Figure 5.18b. An approximately 2 bar load variation of IMEP can be seen in Figure 5.21a. The IMEP distribution only approximates a Gaussian distribution as there is slightly more cycles with 2.4, 2.6 and 3.1 bar as shown by the spikes in the histogram. These spikes are attributed to the PRBS input signal not being Gaussian. A large load range for HCCI combustion is obtained especially considering no external intake air path variations were made such as changing the boost pressure or intake temperature. The variation in CA50, is approximately Gaussian. As shown in Figure 5.21b, where the average combustion phasing is approximately 6 CAD aTDC which is desirable as the most efficient combustion phasing traditionally lies in the range of 5-8 CAD aTDC [56]. Similarly, Figure 5.21d shows a Gaussian distribution of the maximum pressure rise rate centered at 5 bar/CAD. Keeping the MPRR below about 8 bar/CAD helps to ensure long engine life and quiet combustion. Finally, the NO<sub>x</sub> emission histogram (Figure 5.21c) shows that HCCI generally produces low NO<sub>x</sub> emissions with the majority of cycles producing around 40 ppm of NO<sub>x</sub> emissions. However, there are still cycles with up to 400 ppm of NO<sub>x</sub> emissions. These high emitting cycles must be avoided to fully take advantage of the low emissions of HCCI combustion. Overall, the collected engine data has acceptable variation over the HCCI operating range of interest to allow for the creation of a data driven ML model that is accurate over the desired operating range.

### 5.3.2 LSTM Model Training

Since the data generated above has sufficient variation in amplitude and frequency it is now used for the data driven LSTM model. Before beginning the training process, the various model options must be chosen and are summarized in Table 5.5. To train this ML model, the Adam algorithm was used in the MATLAB Deep Learning Toolbox<sup>®</sup>. The loss function vs epoch for the proposed deep network is presented in

Figure 5.22. Additionally, the validation loss function converges to the training loss function, suggesting that neither overfitting nor underfitting has occurred [146]. It is also possible to see that within relatively few iterations (approximately 50 epochs) the model training approaches the minimum after 3450 epochs.

Table 5.5: Selected training options for proposed deep network to predict HCCI performance and emissions

<b>Name</b>	<b>Value</b>
Optimizer	Adam
Maximum Epochs	3500
Mini batch size	512
Learn rate drop period	1000 Epochs
Learn rate drop factor	0.5
L2 Regularization	10
Initial learning rate	0.001
Validation frequency	64 iteration
Momentum	0.9
Squared gradient decay	0.99

With the chosen network, an acceptable loss is achieved during the training process. Now the model prediction is checked for the validation data and is compared to the experimental values as shown in Figure 5.23. Where the final 10,000 cycles (of the 68,000 total cycles) are used for validation of the developed model. The comparison between the model and experimental data show that for all four model outputs an acceptable prediction is possible. The developed LSTM model is able to accurately capture steps and transients for both engine performance outputs (IMEP, CA50 and MPRR) as well as for engine-out emissions ( $\text{NO}_x$ ). Some differences between the model and experiment, specifically for the IMEP and CA50 prediction are observed as a couple misfire cycles (or extremely late combustion cycles) that the model is unable to predict located between cycle 1000 and 1900. Since misfire cycles are

inherently difficult to predict the LSTM model struggles. It can be seen that the prediction for both IMEP and CA50 look like filtered values which is partially due to the large inherent variation in HCCI combustion that can not be predicted using the inputs provided to the model. However, overall the model prediction is quite good and the transient performance of the model is also quite good as shown by the tracking performance of the NO<sub>x</sub> emissions.

The accuracy of this model for each output can be summarized as shown in Table 5.6. For accuracy, the Root Mean Square Error (RMSE) and Normalized Root Mean Square Error (NRMSE) are used which is defined as

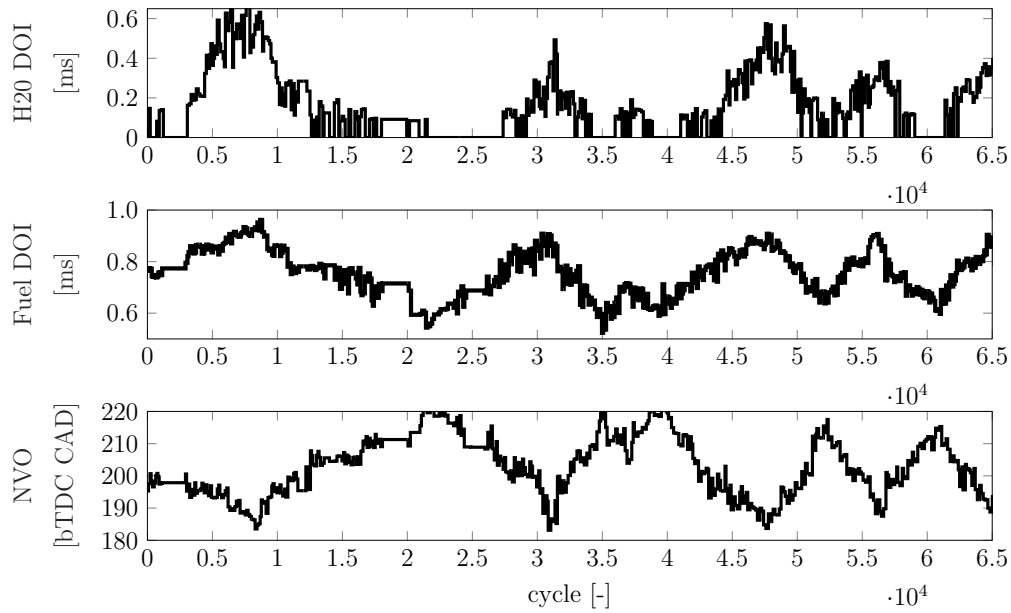
$$\text{RMSE} = \sqrt{\frac{\sum_{k=0}^{N-1} (\hat{y}(k) - y(k))^2}{N}} \quad (5.49)$$

$$\text{NRMSE} = \frac{\text{RMSE}}{y_{\max} - y_{\min}} \quad (5.50)$$

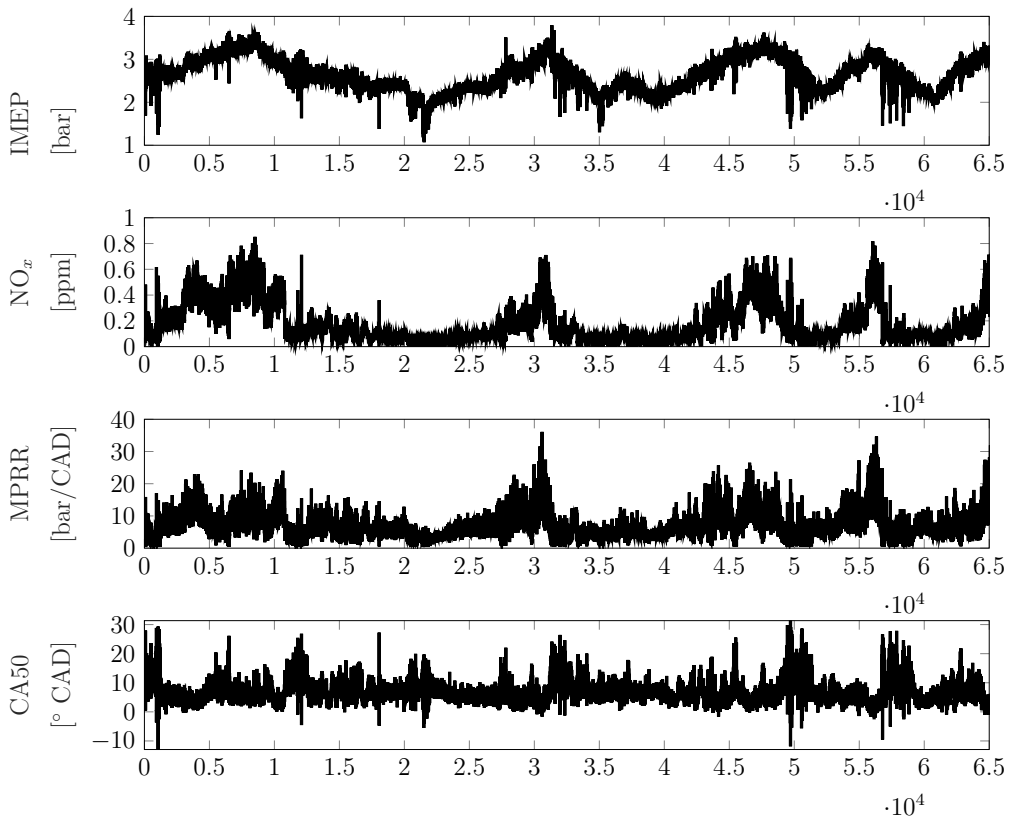
As presented in Table 5.6, MPRR is the most difficult parameter for the model to predict as shown by the 4.7% error in validation while other outputs are predicted with less than 4% error. The prediction of IMEP is quite good with an error of less than 2.8% for both training and test data. Even the NO<sub>x</sub> emission prediction, which is generally more difficult to predict compared to the engine performance parameters, is within 4% for the validation data. The model prediction accuracy could be improved by adding more hidden and cell states to the LSTM layer; however, this is not guaranteed as the HCCI process contains significant stochastic behavior that is difficult to model. Additionally, by adding more states, a significant increase in the computational time of the model on the real-time hardware occurs. Therefore, this model has been improved only by adjusting the number of hidden units of the fully connected layers. As the developed model utilizes only 4 cell and 4 hidden states and provides an accuracy of under 5% error it will be used for the NMPC design in the next chapter.

Table 5.6: RMSE and normalized RMSE of DNN model vs Experiment– RMSE: root mean square error, IMEP: Indicated mean effective pressure, FQ: Fuel Quantity. PM: Particle matter, MPRR: maximum pressure rise rate

	<b>Unit</b>	<b>Training</b>	<b>Validation</b>
$y_{\text{IMEP}}$	[bar]	0.074	0.077
	[%]	2.7	2.8
$y_{\text{NO}_x}$	[ppm]	18	16
	[%]	4.2	3.8
$y_{\text{MPRR}}$	[bar/CAD]	1.6	1.7
	[%]	4.4	4.7
$y_{\text{CA50}}$	[CAD]	1.5	1.6
	[%]	3.4	3.6

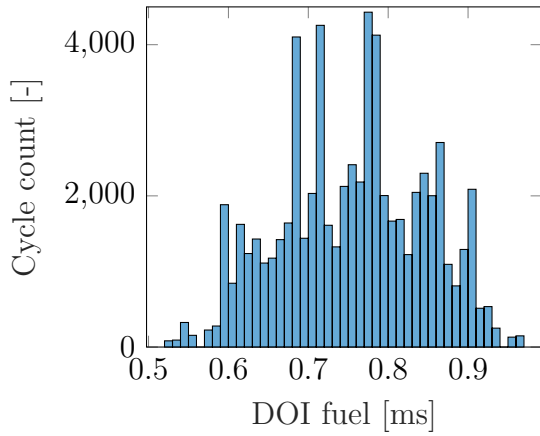


(a) PRBS inputs

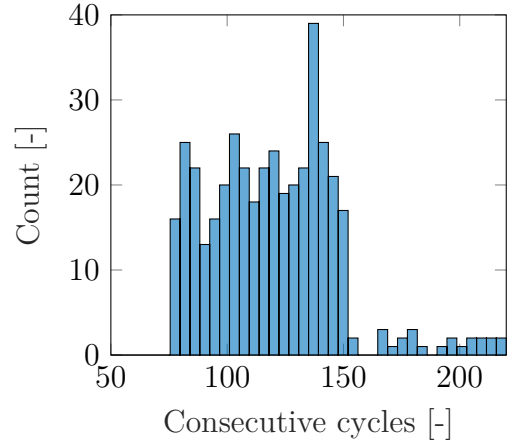


(b) Resulting engine outputs

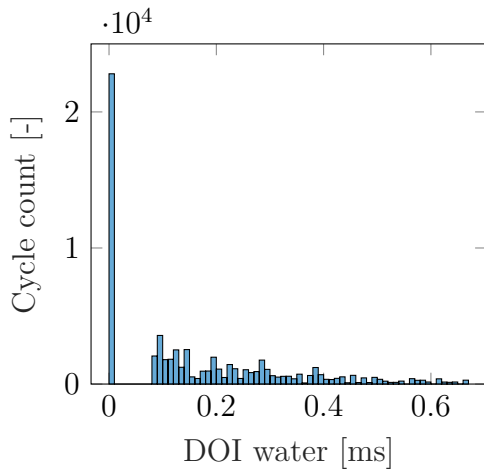
Figure 5.18: PRBS data generation on single cylinder research engine for DNN model training



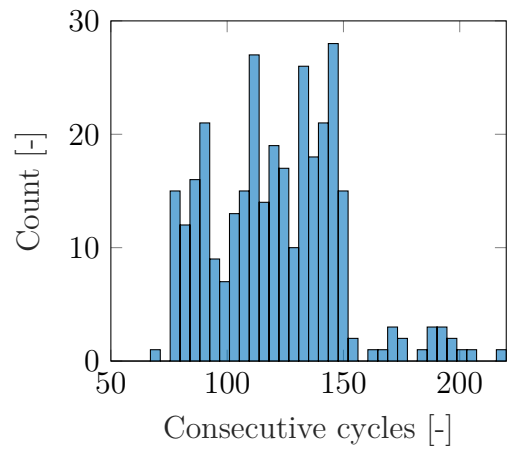
(a) Fuel injection duration cycle distribution



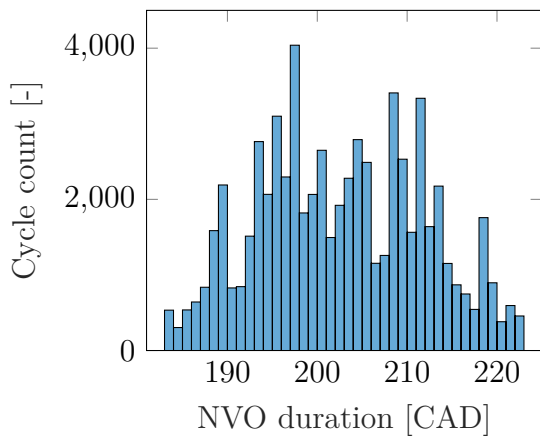
(b) Fuel injection duration consecutive cycle distribution



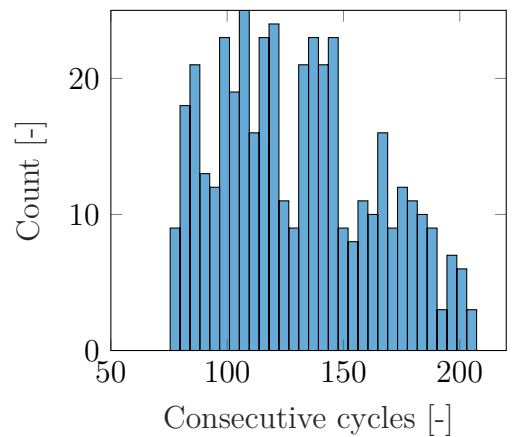
(c) NO<sub>x</sub> Water injection duration cycle distribution



(d) Water injection duration consecutive cycle distribution



(e) NO<sub>x</sub> Water injection duration cycle distribution



(f) Water injection duration consecutive cycle distribution

Figure 5.19: Distribution of engine inputs using PRBS data generation

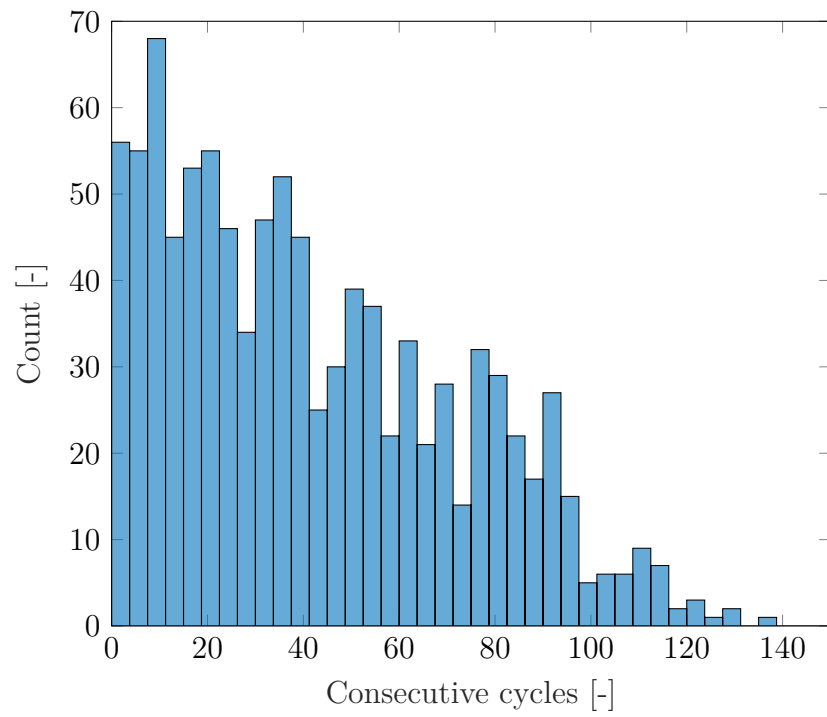
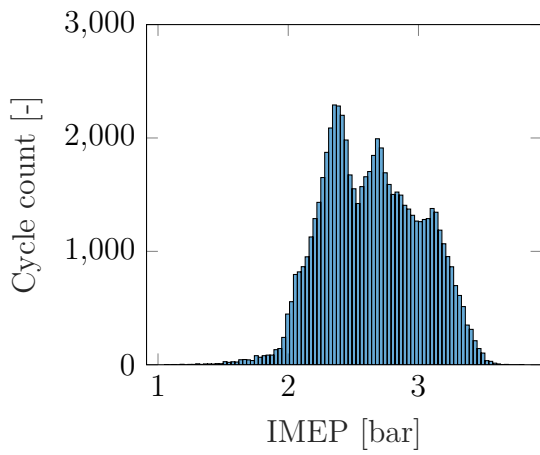
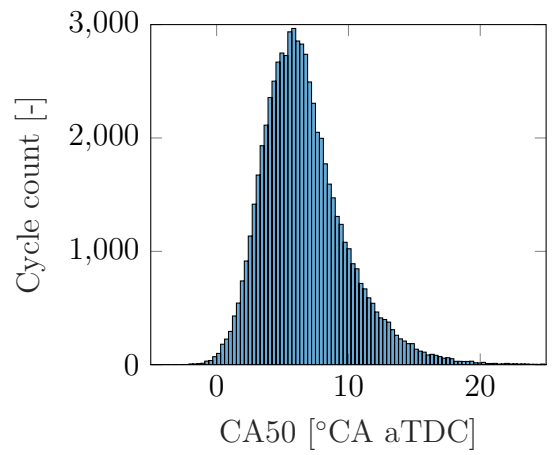


Figure 5.20: Distribution of consecutive cycles where all inputs are constant during PRBS

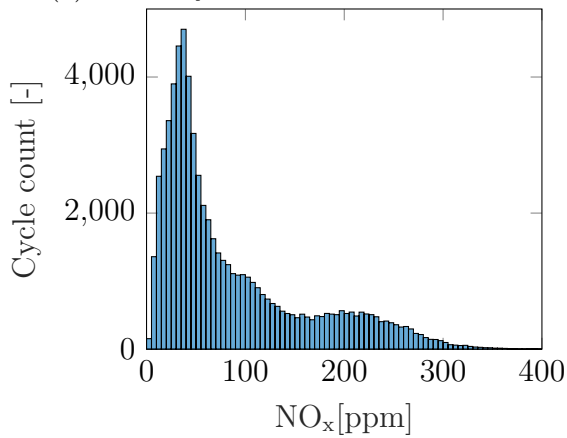




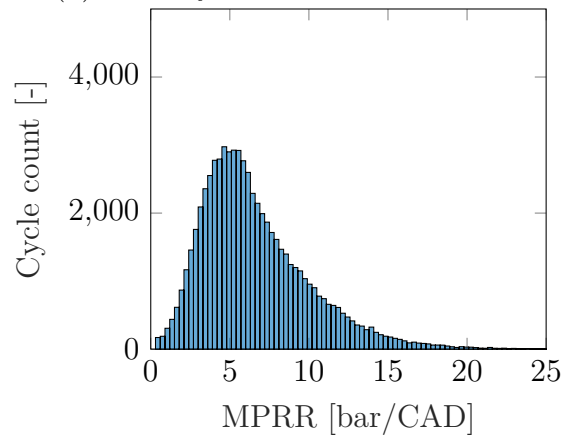
(a) IMEP cycle distribution



(b) CA50 cycle distribution



(c) NO<sub>x</sub> cycle distribution



(d) MPRR cycle distribution

Figure 5.21: Distribution of engine outputs resulting from PRBS data generation

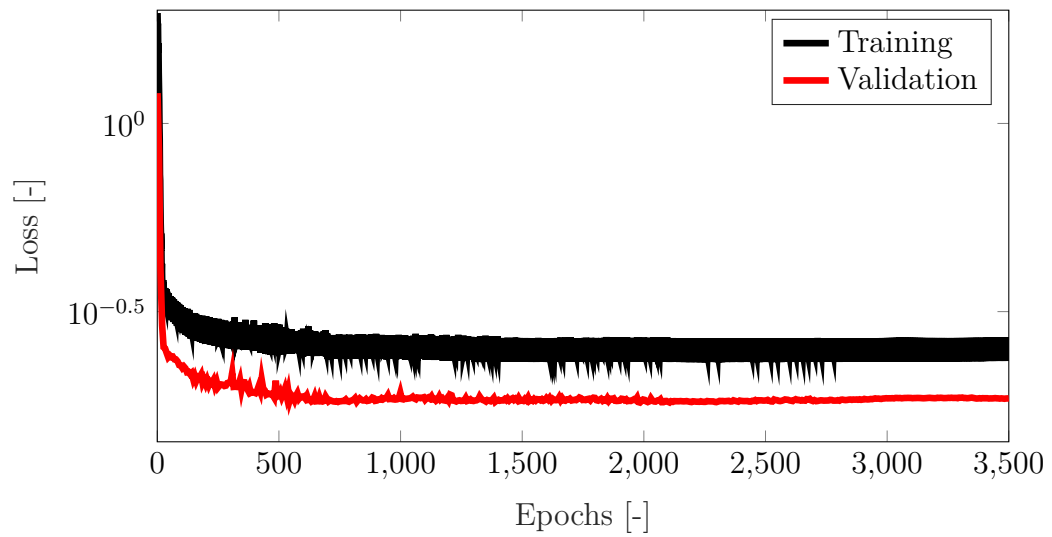


Figure 5.22: Loss versus epochs (number of passes of the entire training dataset) for the proposed deep neural network model

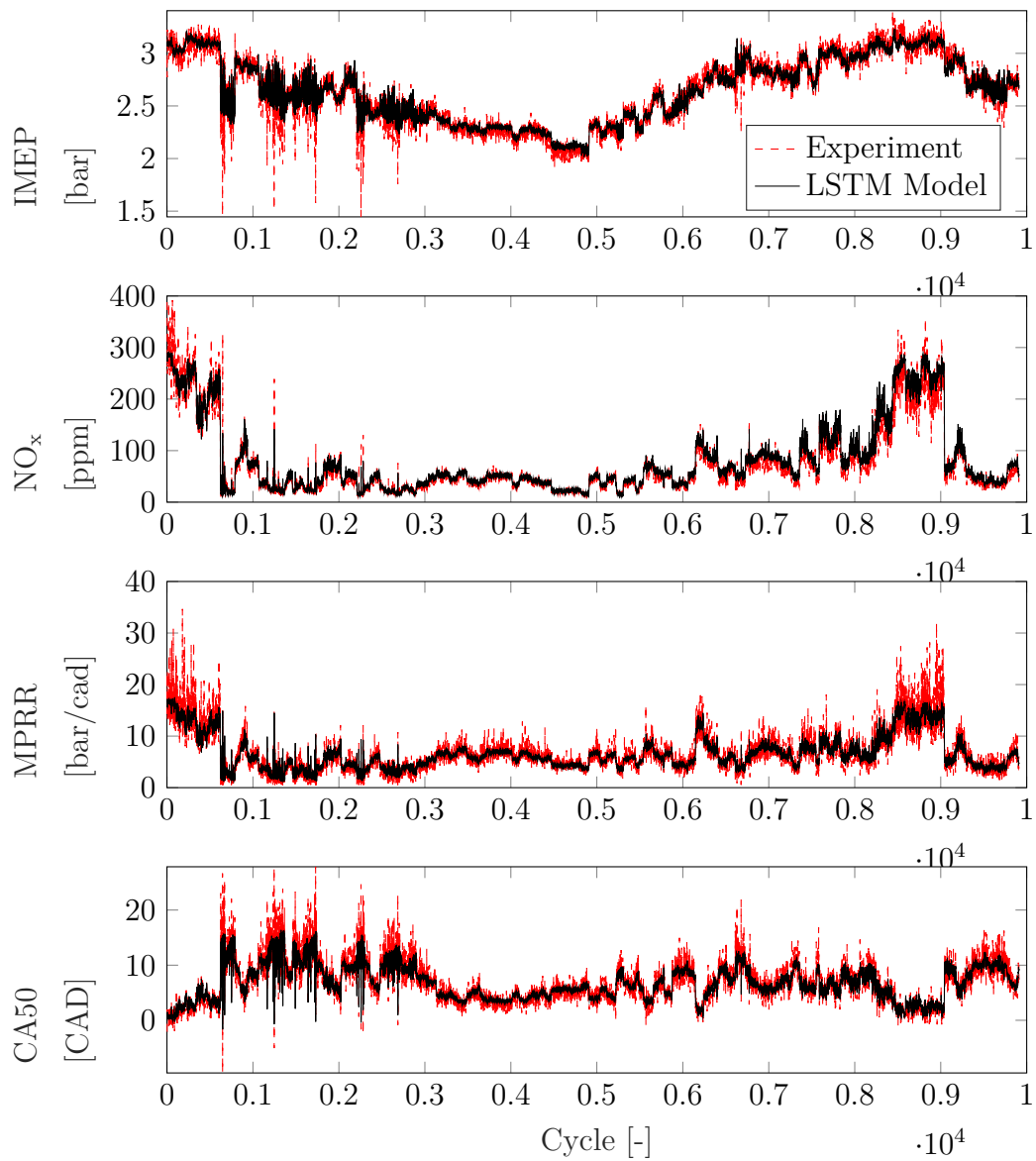


Figure 5.23: Validation for LSTM-based DNN model vs. experimental data – a) – Indicated mean effective pressure (IMEP), b) –Nitrogen Oxides (NO<sub>x</sub>), c) – Maximum pressure rise rate (MPRR), d) –Combustion phasing angle (CA50)

## 5.4 Summary of chapter

This chapter described two different approaches to modeling the emissions from HCCI combustion. The first was a physics based approach where the combustion was modeled using a chemical kinetic model of the fuel in one combustion zone. Conservation of energy is used for the cylinder temperature and the ideal gas law is used for the cylinder pressure. When the chemical reactions are considered directly and the reaction progression rates are calculated a detailed representation of the cylinder state at each crank angle is known. Even the kinetics model with a reduced number of reactions requires a step size of approximately  $1.6e-10$ s which is about 2000 times slower than what is possible when using current FPGA hardware.

The second approach focused on machine learning based models which require a large amount of computation to train (offline) but are much more computationally efficient to implement in real-time. Specifically, the effect of different machine learning approaches and feature sets on the model quality for HCCI emissions prediction was shown. The goal of this investigation was to select an accurate model while also selecting the simplest model that still has an acceptable prediction capability for real-time control implementation in an MPC. For stationary data, a linear and a non-linear SVM models were compared to a traditional ANN model. This comparison showed for a small data set that SVM based models were more robust to changes in feature selection and are better able to avoid local minimums compared to ANN leading to a more consistent model prediction. For each of the four emissions examined the best model type was determined by taking the highest average  $R^2$  value less the variance in  $R^2$  over the various feature sets. This led to the NLSVM being selected for 3 of the emissions and LSVM for  $\text{NO}_x$  prediction.

Unlike the kinetics model, that implements the physics of the combustion process directly, for the ML models the opportunity to apply knowledge of the system is in the individual feature selection. The base feature sets were chosen by considering

known inputs to the combustion process. These were extended by multiplying the individual features together to explore in-feature interactions. By comparing the individual features with the base feature set (L7) the feature set with an improved accuracy that is acceptable given the increase in model complexity was chosen. In this study, the emission models chosen for control purposes for CO, HC, NO<sub>x</sub>, and CO<sub>2</sub> are NLSVM-L13, NLSVM-L7, LSVM-L13, and NLSVM-L7, respectively. The NO<sub>x</sub> and CO models have the largest prediction error while the HC and CO<sub>2</sub> models are quite accurate. The NO<sub>x</sub> model produced the least accurate results however it was still able to capture the trends in NO<sub>x</sub> production.

The next ML model tested was a long short term memory model surrounded by fully connected layers to provide a transient process model unlike the quasi-stationary model resulting from the SVM. This model requires significantly more data than the SVM model, however, due to the recurrent neural network the model is able to capture time dependencies in the data. This resulted in an accurate model with an error less than 5% for all four model outputs (IMEP, CA50, MPRR and NO<sub>x</sub>). Of the models developed and tested the LSTM model showed the greatest accuracy while preserving a simple model structure to allow for real-time implementation. For this reason, the LSTM ML model will be used within a model predictive controller in the next chapter.

# Chapter 6

## Nonlinear Model Predictive Control Experimental Implementation for HCCI <sup>1</sup>

The development of the nonlinear model predictive controller (NMPC) using the LSTM model developed in Section 5.3 for HCCI performance and emissions is the next step. The development of the NMPC and its experimental implementation is described in this chapter. In NMPC, the current control input is computed by solving a nonlinear program at each sample time. The underlying optimization problem consists of a model based prediction of the system's behavior, starting from the current state. The selected cost function minimizes engine-out  $\text{NO}_x$  emissions while simultaneously trying to reduce fuel and water consumption while tracking a requested output load (IMEP) and combustion phasing (CA50) for an HCCI engine. The optimization also must consider constraints placed on control action and states of the model.

The development of control algorithm in real-time on the dSPACE hardware through MATLAB/ Simulink is demonstrated in this section. The overall development process is schematically shown in Figure 6.1. Four main steps are followed to get a real time NMPC. The first two steps are the collection of engine data for model development and validation are described Section 5.3. The third step is the

---

<sup>1</sup> This chapter is partially based on [1, 2, 16, 202]

offline simulation and tuning of the NMPC where the objective (cost) function and physical constraints are defined and tuned to get acceptable controller performance. The fourth and final step is the transition of the MPC optimizer to the embedded processor using `acados`. In this chapter, the NMPC has been implemented in MATLAB/Simulink using the open source package `acados` [203].

As the success of the real-time NMPC is highly dependent on the calculation time of the online NMPC this chapter also investigates the performance of `acados` running on various embedded processors and the communication with the main engine controller the dSPACE MABX II. Finally the results on the online NMPC are presented.

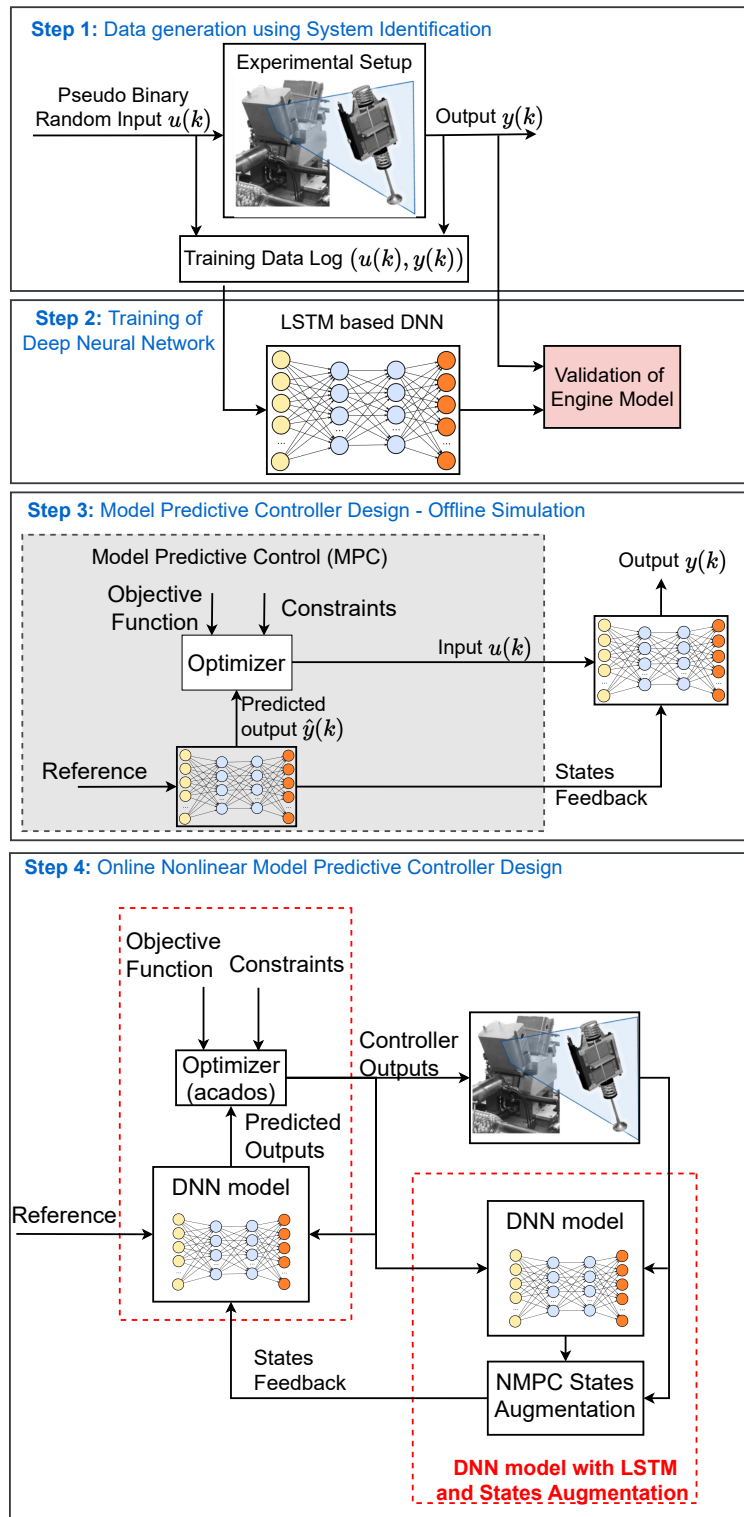


Figure 6.1: Development process for creation of online NMPC for HCCI engine control



## 6.1 HCCI NMPC Creation

### 6.1.1 Nonlinear State-Space Representation

The model structure of the Deep Neural Network as shown in Figure 5.17, derived in the section 5.3, must first be formulated using a nonlinear state-space representation to allow for integration into `acados`. The developed model is the sequence of fully connected layers applied to the inputs of the network (Eq. 5.44), an LSTM layer and a set fully connected layers applied to the output of the model (Eq. 5.45). The network is defined as:

$$z_{\text{FC3}}(k) = f_{\text{FC,in}}(u(k)) \quad (6.1)$$

$$\begin{bmatrix} c(k) \\ h(k) \end{bmatrix} = f_{\text{LSTM}}(c(k-1), h(k-1), z_{\text{FC3}}(k)) \quad (6.2)$$

$$y(k) = f_{\text{FC,out}}(h(k)) \quad (6.3)$$

Now, eliminating the intermediate value  $z_{\text{FC3}}(k)$  results in

$$\begin{bmatrix} c(k) \\ h(k) \end{bmatrix} = f(c(k-1), h(k-1), u(k)) \quad (6.4)$$

$$y(k) = f_{\text{FC,out}}(h(k)) \quad (6.5)$$

The hidden and cell states are now grouped in an overall state vector  $x(k) = [c(k-1), h(k-1)]^\top$ , the resulting system of equations becomes

$$x(k+1) = f(x(k), u(k)) \quad (6.6)$$

$$y(k) = f_{\text{FC,out}}(x(k+1)) \quad (6.7)$$

is close to the standard nonlinear state-space representation of a dynamic system. However, in the standard formulation, the output depends only on the current state  $x(k)$  and input  $u(k)$ . One way to make the LSTM based network conform to the standard form is to adapt the definition of the engine cycle, accounting for the output

after the control actions to the following cycle. Here, the next state in the output function is replaced by its definition, resulting in a nonlinear output function

$$x(k+1) = f(x(k), u(k)) \quad (6.8a)$$

$$\begin{aligned} y(k) &= f_{\text{FC,out}}(f(x(k), u(k))) \\ &= g(x(k), u(k)) \end{aligned} \quad (6.8b)$$

with  $x(k)$  the model states from above,  $y(k)$  the model outputs, and  $u(k)$  the model inputs. These are:

$$x(k) = \begin{bmatrix} c(k-1) \\ h(k-1) \end{bmatrix} \in \mathbb{R}^8, \quad (6.9a)$$

$$y(k) = \begin{bmatrix} y_{\text{IMEP}}(k) \\ y_{\text{CA50}}(k) \\ y_{\text{NO}_x}(k) \\ y_{\text{MPRR}}(k) \end{bmatrix} \in \mathbb{R}^4, \quad (6.9b)$$

$$u(k) = \begin{bmatrix} y_{\text{IMEP}}(k-1) \\ y_{\text{CA50}}(k-1) \\ u_{\text{DOI,fuel}}(k) \\ u_{\text{DOI,water}}(k) \\ u_{\text{NVO}}(k) \end{bmatrix} \in \mathbb{R}^5. \quad (6.9c)$$

The model is dependent on the IMEP and CA50 from the previous cycle as inputs for the next cycle.

A positive definite weighting matrix for the control inputs would thus force an unnecessary compromise between tracking the output and the manipulated variables. By introducing the change of manipulated variables as new inputs [204, 205], the positive definite weighting matrix now only drives the change to be zero, posing no

conflict in reaching the desired output setpoints.

$$\underbrace{\begin{bmatrix} x(k+1) \\ u(k) \end{bmatrix}}_{\tilde{x}(k+1)} = \underbrace{\begin{bmatrix} f(x(k), u(k-1) + \Delta u(k)) \\ u(k-1) + \Delta u(k) \end{bmatrix}}_{\tilde{f}(\tilde{x}(k), \Delta u(k))} \quad (6.10a)$$

$$\underbrace{\begin{bmatrix} y(k) \\ u(k-1) \end{bmatrix}}_{\tilde{y}(k)} = \underbrace{\begin{bmatrix} g(x(k)) \\ u(k-1) \end{bmatrix}}_{\tilde{g}(\tilde{x}(k))}. \quad (6.10b)$$

Both the absolute inputs as well as their rate of change can be penalized in the cost function.

### 6.1.2 Optimal Control Problem

Given Eq. 6.10, the discrete Optimal Control Problem (OCP) is defined as follows

$$\begin{aligned} \min_{\substack{\Delta u_0, \dots, \Delta u_N \\ \tilde{x}_0, \dots, \tilde{x}_N \\ \tilde{y}_0, \dots, \tilde{y}_N}} & \sum_{i=0}^N \|r_i - \tilde{y}_i\|_Q^2 + \|\Delta u_i\|_R^2 \\ \text{s.t.} & \quad \tilde{x}_0 = \begin{bmatrix} x(k), & u(k-1) \end{bmatrix}^\top, \\ & \quad \tilde{x}_{i+1} = \tilde{f}(\tilde{x}_i, \Delta u_i) \quad \forall i \in \mathbb{H} \setminus N, \\ & \quad \tilde{y}_i = \tilde{g}(\tilde{x}_i, \Delta u_i) \quad \forall i \in \mathbb{H}, \\ & \quad u_{\min} \leq F_u \cdot \tilde{u}_k \leq u_{\max} \quad \forall i \in \mathbb{H}, \\ & \quad y_{\min} \leq F_y \cdot \tilde{y}_k \leq y_{\max} \quad \forall i \in \mathbb{H} \end{aligned} \quad (6.11)$$

where  $\mathbb{H} = \{0, 1, \dots, N\}$ . The subscripts  $i$  indicate that the variables are part of the internal computations of the NMPC controller, whereas  $x(k)$  and  $u(k-1)$  are the actual model's current state and the previously applied control input, respectively. In this formulation, the nonlinear output function is stated as part of the constraints by introducing the augmented output as an optimization variable, allowing a linear least squares cost function. After augmenting the controls into the state vector the NMPC states and outputs can be written as

$$x(k) = \begin{bmatrix} y_{\text{IMEP}}(k) \\ y_{\text{CA50}}(k) \\ y_{\text{NO}_x}(k) \\ y_{\text{MPRR}}(k) \\ c(k-1) \\ h(k-1) \\ \text{DOI,fuel}(k) \\ \text{DOI,water}(k) \\ \text{NVO}(k) \end{bmatrix} \in \mathbb{R}^{15}, \quad (6.12a)$$

$$u(k) = \begin{bmatrix} \Delta \text{DOI,fuel} \\ \Delta \text{DOI,fuel} \\ \Delta \text{NVO} \end{bmatrix} \in \mathbb{R}^3. \quad (6.12b)$$

The reference  $\tilde{r}_i$  and the weighting matrix  $Q$  are selected such that deviations from the requested IMEP and CA50 are penalized while minimizing  $\text{NO}_x$  emissions as well as the amount of injected fuel and water

$$\tilde{r}_i = \begin{bmatrix} r_{\text{IMEP},i}, & r_{\text{CA50},i}, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0 \\ 0, & 0, & 0, & 0, & 0 & & & & & & \end{bmatrix}^T, \quad (6.13)$$

$$Q = \text{diag}(q_{\text{IMEP}}, q_{\text{CA50}}, q_{\text{NO}_x}, q_{\text{MPRR}}, 0, 0, 0, 0, 0, 0, 0, 0, 0, r_{\text{DOI,fuel}}, r_{\text{DOI,water}}, 0). \quad (6.14)$$

The specific cost function  $J$  is specified as

$$J = \sum_{i=0}^N \underbrace{\|r_{\text{IMEP},i} - y_{\text{IMEP},i}\|_{q_{\text{IMEP}}}^2 + \|r_{\text{CA50},i} - y_{\text{CA50},i}\|_{q_{\text{CA50}}}^2}_{\text{Reference Tracking}} + \dots \quad (6.15)$$

$$\underbrace{\|y_{\text{NO}_x,i}\|_{q_{\text{NO}_x}}^2}_{\text{Emission Reduction}} + \dots \quad (6.16)$$

$$\underbrace{\|u_{\text{DOI,fuel},i}\|_{r_{\text{DOI,fuel}}}^2 + \|u_{\text{DOI,water},i}\|_{r_{\text{DOI,water}}}^2}_{\text{Fuel / Water consumption reduction}} + \dots \quad (6.17)$$

$$\underbrace{\|\Delta u_i\|_R^2}_{\text{Oscillation Reduction}} \quad (6.18)$$

The weighting matrix  $R$  is a diagonal matrix with positive elements defined as

$$R = \text{diag}(r_{\Delta u_{\text{DOI,fuel}}}, r_{\Delta u_{\text{DOI,water}}}, r_{\Delta u_{\text{NVO}}}) \quad (6.19)$$

One advantage of NMPC is the ability to impose constraints on inputs and outputs.  $F_u$  and  $F_y$  are diagonal matrices in eqn 6.11 with ones at the locations of bounded outputs and inputs.

The IMEP of the engine is limited in order to keep the engine running near the operating range that was used to develop the LSTM model. The load range of 1-6 bar IMEP is also chosen as this is the range in which open loop testing has shown that HCCI is possible using the naturally aspirated intake conditions. The limits on CA50 imposed are a range of 0-17 CAD aTDC which removes any extremely early and late cycles without over constraining the NMPC optimization. The limits on  $\text{NO}_x$  are used to constrain peak emissions levels and can be set to meet emission standards. A limit of 300 ppm for  $\text{NO}_x$ , was selected based on the data collected during the PRBS system identification (shown in Figure 5.18b) that shows high emissions over 300 ppm are possible. Controlling the maximum pressure rise rate (MPRR) is crucial in combustion engines to ensure quiet and safe engine operation at various engine loads. MPRR is the rate at which the pressure increases in the cylinder and the maximum permissible MPRR is engine and application-dependent. If MPRR is too high, the increased pressure rise rates can lead to engine damage and

acoustically loud combustion. However, a higher MPRR is closer to the ideal Otto cycle where all the heat is released instantaneously and therefore is a more efficient combustion. Here, a high limit of 15 bar/CAD (for a production engine) constraint is implemented.

Constraints are also imposed on the DOI for both the fuel and water injections. These constraints are set approximately 50% higher than the maximum value that was used during PRBS data generation. This is to allow the controller flexibility but also ensuring the NMPC solution is within an acceptable range.

Table 6.1 summarizes the implemented limitations on outputs and manipulated variables. In future testing these specified limits could be modified to meet legislation or design constraints. Here, to investigate the potential of the method, not overly restrictive constraints were chosen.

Table 6.1: Constraint Values

Lower bound	Variable	Upper bound
1 bar	$y_{IMEP}$	6 bar
0 CADaTDC	$y_{CA50}$	17 CADaTDC
0 ppm	$y_{NO_x}$	500 ppm
0 bar/CAD	$y_{MPRR}$	15 bar/CAD
0 ms	$u_{DOI,fuel}$	1.50 ms
0 ms	$u_{DOI,water}$	1.00 ms
150 CAD	$u_{NVO}$	360 CAD

## 6.2 Implementation of NMPC in Simulation

The control algorithm development using MATLAB/ Simulink is demonstrated in this section. As the engine combustion cycle repeats, depending on engine speed, the available computation time for the NMPC controller is dependent on the speed of the engine. At 1500 rpm, one engine cycle lasts 80 ms. To meet the real-time require-

ments, a computationally efficient algorithm is needed to provide feedback from one engine cycle to the next cycle. The NMPC controller is implemented in MATLAB/Simulink using the open source package `acados` [206], since simulation results [202] indicate that it outperforms MATLAB’s MPC toolbox in terms of computation time, both with `fmincon` and the `FORCES PRO` [207, 208] backend.

The plant model is passed to `acados` through the discrete dynamics interface, as no discretization is required. For computation of the Hessian in the underlying Sequential Quadratic Programming (SQP) algorithm, the Gauss-Newton approximation is used by selecting the non exact Hessian option. The resulting Quadratic Problems (QPs) within the SQP algorithm are solved using the Interior Point (IP) [209] based QP solver `hpipm` [210], that is provided by the `acados` package [206]. Although when compared to the Active Set based QP solvers such as `qpOASES` [211], `hpipm` using the IP method has a higher computation time on average. However, worst case peaks, which are critical for feasible realtime implementation are lower [210].

The OCP in Eq. 6.11 leads to a band diagonal structure in the matrices of the QPs within the SQP algorithm, which can be exploited by `hpipm`. Fully condensing the problem improves the computation time compared to passing the sparse but high dimensional problem formulation. The difference in runtime is attributed to the state vector having a higher dimension than the control input vector and the modeled engine dynamics only require small prediction horizons of three cycles. The resulting OCP structure allows the fully condensed problem formulation to take full advantage of the condensation benefits [204, 205]. In this work, partial condensing using the `hpipm` solver has been used. Full condensing using the `hpipm` solver has also been tested and showed very little improvement in either controller performance or solver time.

Tuning of the weights, setting the number of allowed SQP iterations and determining the prediction horizon are done by means of model in the loop simulations, where

the NMPC controller is run compared to the internal plant model. Through these stimulative studies, the allowed number of SQP iterations is limited to five while the prediction horizon is set to three. Previous work has shown that the cycle-to-cycle dependency of HCCI lasts approximately two cycles so a prediction horizon of three cycles is sufficient [7].

For the offline NMPC design, the controller is tuned to follow a reference for requested load (IMEP) for a set of six steps that are within the range that the LSTM model has been trained. Figure 6.2 shows the final tuned NMPC resulting from the offline-model-in-the-loop testing. The first three sub figures show the controller actions for the fuel and water injection amounts along with the NVO duration. The commanded engine inputs are within the expected values for the all three inputs and they are all within the constraints set in Table 6.1.

Figure 6.2d-g show the engine outputs predicted by the engine model. It can be seen that the NMPC is able to closely track the requested IMEP with just a slight delay after each step change in the reference. This slight delay is caused by the NMPC cost function, eqn 6.15, which also has weights on the rate of change of the outputs to prevent oscillation which has the effect of damping the response of the controller. The reference for the  $\text{NO}_x$  emissions has been set to 0 in order to minimize the emissions and its easy to see that the predicted  $\text{NO}_x$  emissions are well below the constraint of 500 ppm applied to the system. The reference for MPRR has been set to 3 bar/CAD as ideally the pressure rise rate needs to be minimized for engine life span and acoustic noise reduction however higher pressure rise rates are often coupled with improved efficiency and therefore the target is set to a realistic value. Finally, the reference for CA50 is set to 5 CAD aTDC to keep the engine running at an efficient operating point. Overall, NMPC is able to keep the combustion phasing at the setpoint quite well with only very slight deviation of under 1 CAD during the steps in IMEP. The one step at cycle 700 when the load is increased significantly results in a slightly larger



overshoot in CA50 of 1.8 CAD as the controller tries to minimize the cost function for all the variables.

The NMPC performance characteristics can be seen in Figure 6.3 where the calculation time of the NMPC running on a Intel i7-9700 PC can be seen. The calculation time in simulation on the PC is much faster than the required NMPC turnaround time. The number of SQP iterations required to achieve convergence can be seen in Figure 6.3b where it can be seen that for most cycles the solver takes 1 iteration. With a few spikes up to the maximum of three iterations especially after IMEP steps are applied. Finally, the cost has been calculated according to eqn 6.15 where it can be seen that during transients the cost spikes due to the deviation between the reference and actual state. However, the cost reaches a steady value within a few cycles as the controller is able to bring the system to the reference. As the model used for the simulation has no stochastic behavior the NMPC finds a solution for each IMEP step and remains at that operating point.

Overall the controller performance and solution times in the offline simulation looks very promising and can now be ported for use on an embedded processor for real-time engine control testing.

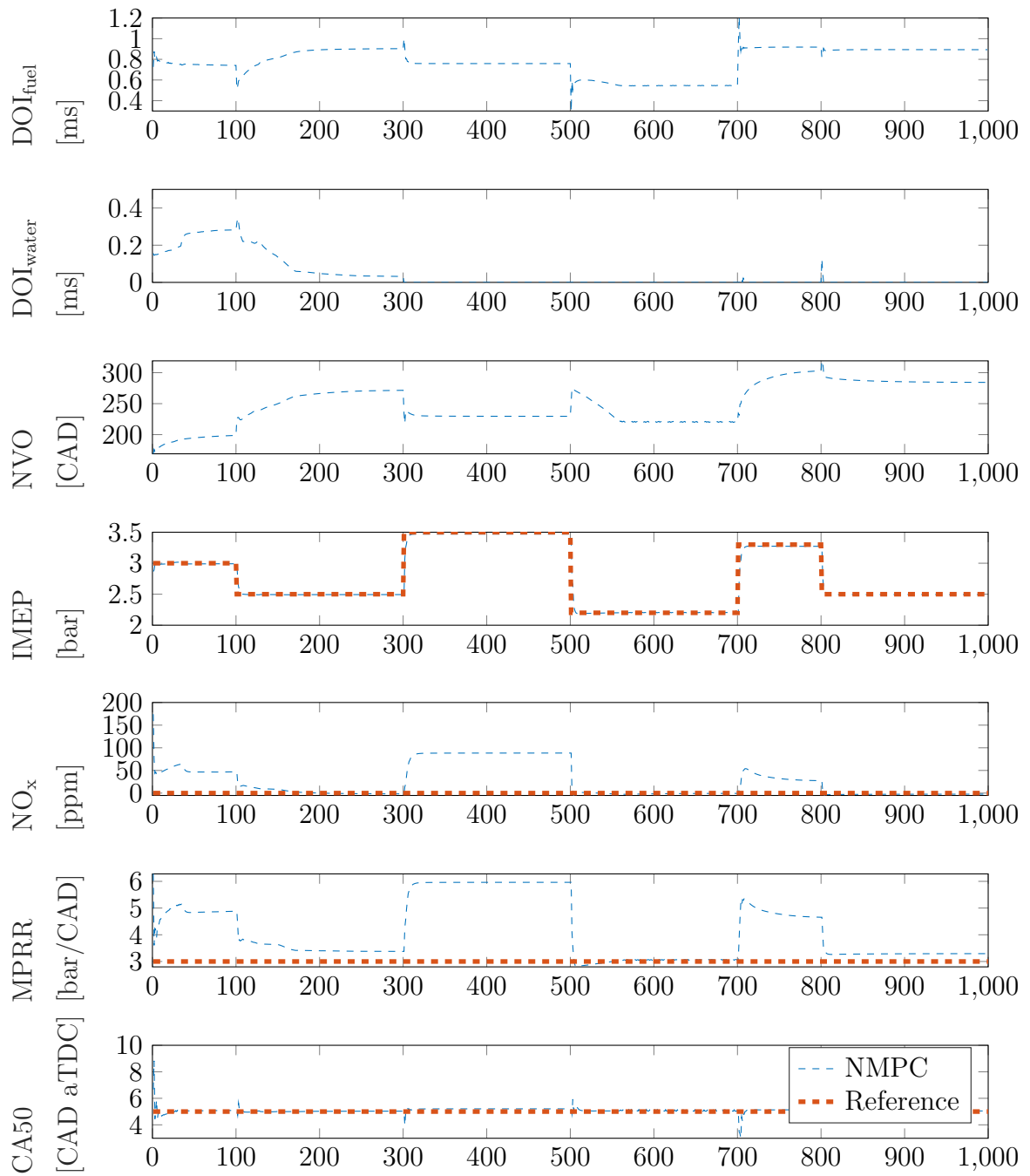


Figure 6.2: *acados* offline NMPC simulation showing controller outputs and predicted engine response

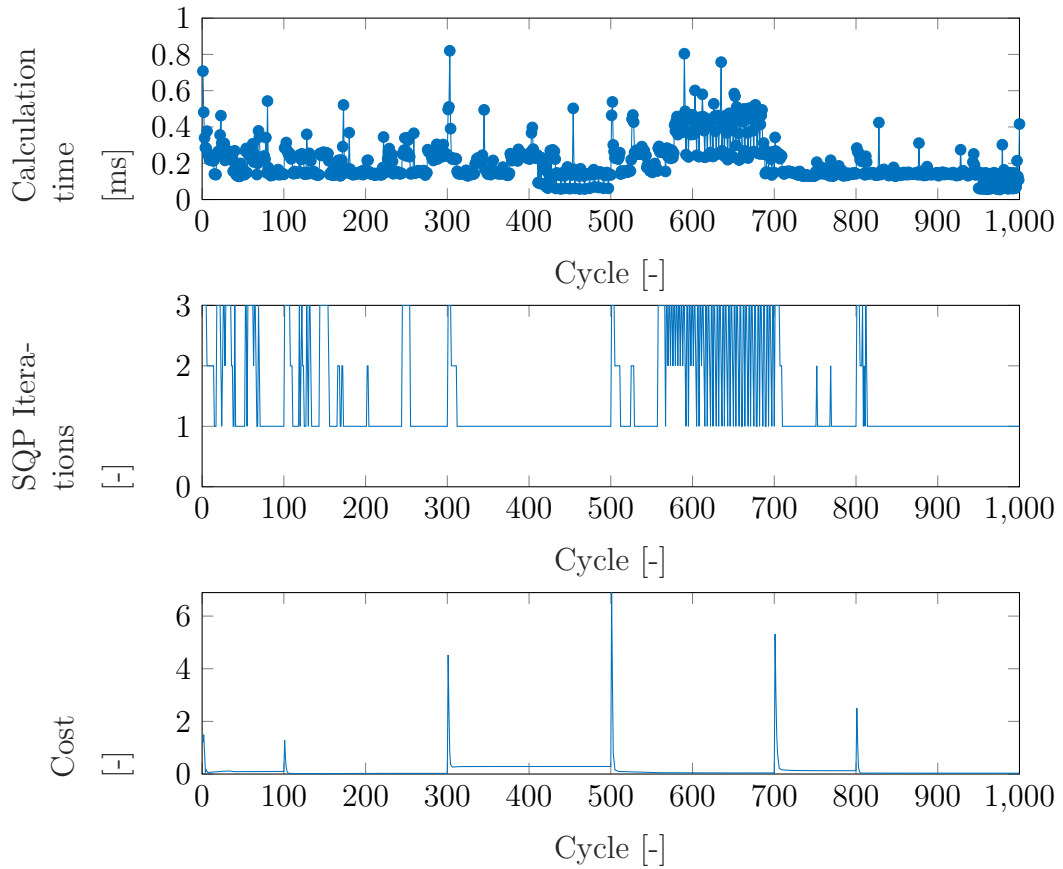


Figure 6.3: `acados` offline NMPC simulation showing NMPC solution results: a) Calculation time of NMPC b) SQP iterations required for NMPC convergence c) Cost resulting from eqn 6.15

## 6.3 Experimental NMPC Implementation

### 6.3.1 `acados` NMPC on MABX II

After tuning the NMPC offline, the algorithm is directly deployed to the embedded processor of the MABX II. The required cross-compiled libraries of `acados` can be obtained by following the “Embedded Workflow” in the `acados` documentation [212]. Figure 6.4 illustrates the implementation of the `acados` NMPC running on the MABX II. Here the measured states provided by the real-time FPGA calculation (developed in [15]) of MPRR, CA50 and IMEP along with the measured  $\text{NO}_x$  emissions are augmented with the LSTM states (also see schematic in Figure 1.7). The LSTM states are the current cell and hidden states which are estimated by the derived DNN model that is running in parallel to the real engine. All of these inputs are then fed to the NMPC for online optimization.

On the MABX II engine controller, the NMPC shows a maximum turnaround time of 56.0 ms and an average of 43.3 ms which is too slow to allow for cycle-to-cycle control, as NMPC turn around of 22 ms is needed, as shown in Figure 1.6. However, to test the controller concept the LSTM model was retrained with the one cycle delay in the model and the cycle-to-next-cycle NMPC was experimentally implemented as shown in Figure 6.5. For the online control performance testing shown, the NMPC is able to follow the requested IMEP steps while keeping both the  $\text{NO}_x$  emissions and MPRR below the specified constraints. When compared to the simulation results, the online NMPC only used a single cycle of water injection during the increase in load step where the water is used to help prevent a large pressure MPRR when the fuel is greatly increased to provide the requested power. The one challenge with the implemented controller is there is a significant variation in the CA50 during the load steps. The average CA50 follows the target value of 7 CAD aTDC, however, the cycle to cycle variations are quite large with an oscillation of around 8 CAD. These large oscillations in CA50 are likely the result of the one cycle delay and the controller not

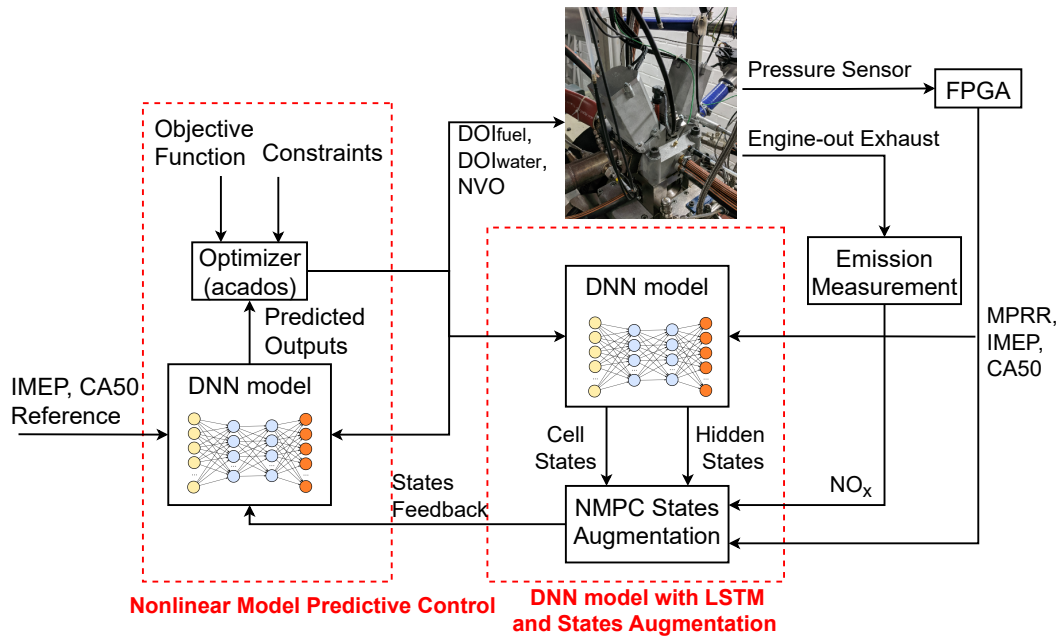


Figure 6.4: Block diagram of LSTM-NMPC structure running on MABX II - IMEP: Indicated mean effective pressure,  $\text{NO}_x$ : Nitrogen Oxides, MPRR: Maximum pressure rise rate, DOI: Duration of injection, CA50: Combustion phasing angle. Prediction horizon of 3.

being able to directly affect the next cycle. Longer time scale oscillations are also present in the  $\text{NO}_x$  emissions and matching oscillations are also visible in the MPRR especially between cycles 95-120. These are likely due to the large cycle to cycle change in fueling and will be resolved in true cycle-to-cycle control as implemented next.

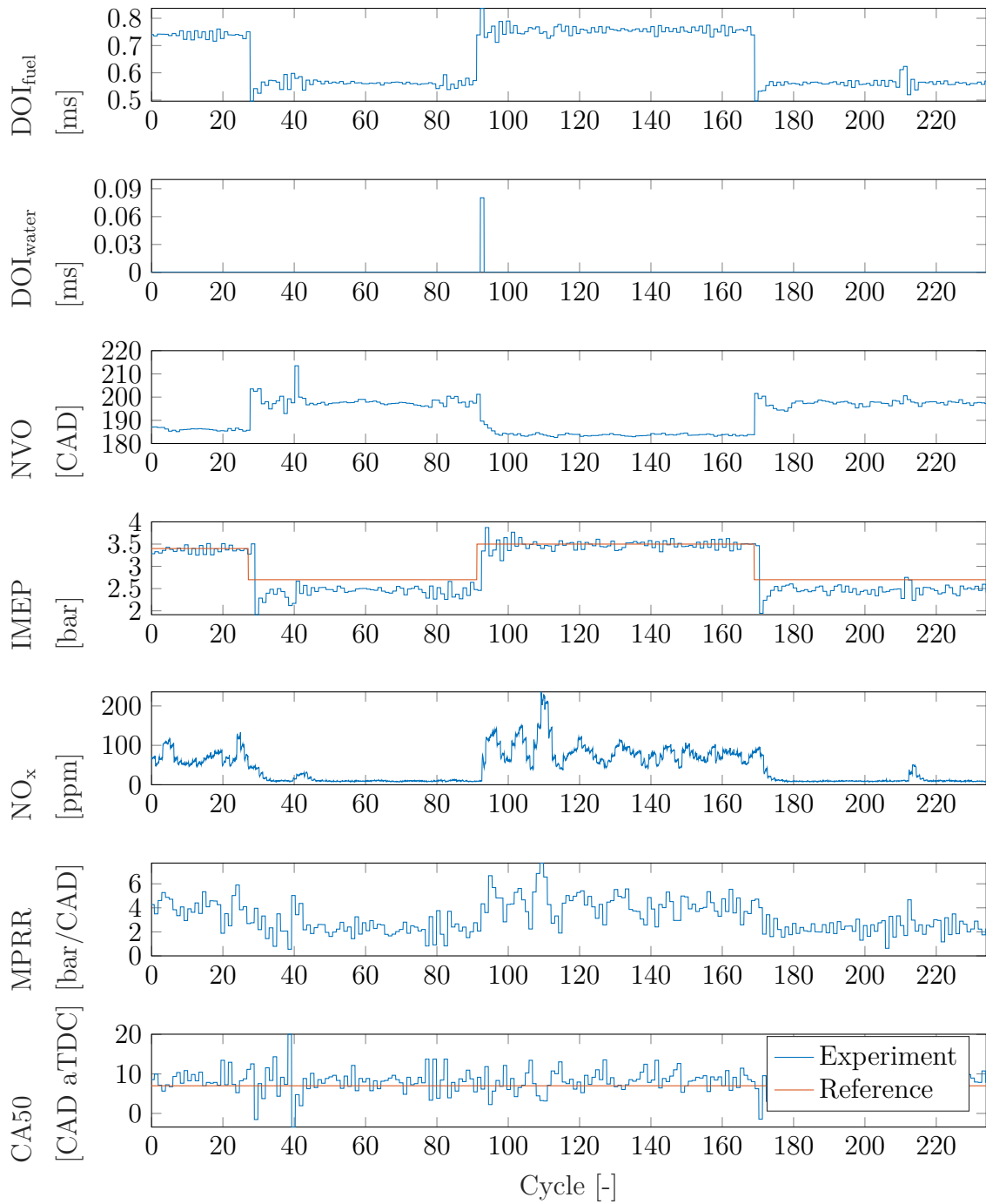


Figure 6.5: Realtime cycle to next cycle NMPC experimental results: Showing IMEP reference tracking performance with one cycle delay.

### 6.3.2 `acados` Embedded Hardware Calculation Time Comparison

Due to the processor speed of the dSPACE MABX II (IBM PPC 750GL single core CPU at 900 MHz) and the time constraints of the HCCI process, it was not possible to complete the calculation of the NMPC in the required 22 ms and therefore it is necessary to evaluate the computational performance of the `acados` implementation on additional embedded hardware. To evaluate the potential of `acados` on a wide variety of hardware the dSPACE MABX II was compared to six other systems. This benchmark was performed for future research projects.

A wide range of systems were tested including the high performance industry standard engine controller development hardware the dSPACE SCALEXIO labbox which contains an Intel Core i7-6820EQ. Three inexpensive Raspberry Pi prototype boards were also tested as they have shown excellent computation performance and were recommended by the `acados` developers. The three Raspberry Pi boards tested were the Pi3, a Pi 4B and the Pi 400. Finally, two windows based personal computers (PC's) were also evaluated, one laptop with an Intel Core i5-8365UE processor and a desktop with an Intel Core i7-9700. The various processors were tested by compiling the `acados` libraries for their specific processors and testing the calculation time using a model in the loop (MiL) simulation. The average calculation time on the various devices for the NMPC calculation is shown in Table 6.2.

The MiL results show that of all the tested devices significantly outperform the baseline dSPACE MABX II which is expected as the MABX II has the oldest and slowest clock rate processor of the devices tested. When looking at calculation times of the various devices tested it can be seen that the calculation time scales with the clock rate of the CPU tested. All of the devices tested (other than the baseline MABX II) are able to complete the NMPC execution in under the 22 ms time constraint presented in Section 6.1. For this work, the Raspberry Pi 400 is selected

Table 6.2: Processor NMPC execution time for HCCI MiL simulation

System	Calculation Time [ms]
<b>dSPACE MABX2 (IBM PPC 750GL @ 900 MHz (baseline)</b>	<b>35</b>
Raspberry Pi 3 Broadcom BCM2837 @ 1.2 GHz	13
Raspberry Pi 4B Broadcom BCM2711 @ 1.5 GHz	6.5
<b>Raspberry Pi 400 Broadcom BCM2711 @ 2.2 GHz (selected)</b>	<b>4</b>
dSPACE SCX Intel Core i7-6820EQ @ 2.8 Ghz	3.8
Laptop Intel Core i5-8365UE @ 1.6/4.1 GHz	1.3
Desktop Intel Core i7-9700 @ 3.0/4.7 GHz	0.7

for further testing for online engine control. At a cost of \$100 Canadian Dollars the NMPC calculation performance only consumes 18% of the available 22 ms. This additional computation space will allow for increase model complexity if needed for future testing.

The main challenge of running the NMPC optimization on an external processor is the requirement that the communication time is also included in the total time for the NMPC calculation to complete. The dSPACE MABX II will still be the main engine controller which performs the in-cylinder state calculation on the FPGA and still provides the actuator commands to the fuel and water injector along with controlling the valve timings. Therefore the two way communication time from the MABX II to the Raspberry Pi 400 and back to the MABX II must be determined. For this task two communication standards have been tested including controller area network (CAN) and User Datagram Protocol (UDP). The CAN network was tested as it is a well defined bus communication standard that is already widely used in the automotive industry which allows communication over two data wires with all devices on the bus network. The UDP protocol was tested as it is one of the core internet protocol (IP) standards that is commonly used for large and fast data transfer between PC's. The communication time over both of these protocols from the MABX II to the Raspberry



Pi 400 and back to the MABX II is tested on isolated networks where the only devices on the network are the two devices being tested and there is a simple counter with an 8 byte message on the MABX II that is sent to the Pi and then returned to the MABX II. This test should provide the best case communication times and the results on the experimental implementation may vary depending on the number of devices on the network and the actual message size needed. The test results are shown in Table 6.3.

Table 6.3: Two-way communication from dSPACE MABX II to Raspberry Pi 400 and back for various communication protocols

Communication Standard	2 way time [ms]
CAN	4-6
UDP	1

The communication tests show that the return time for a message over the CAN network varies between 4-6 ms while the communication over UDP results in the message being returned in 1 ms. The increased communication time of the CAN interface may also be higher as the Raspberry Pi needs to communicate with an external MCP2517FD chip or the delay could be due to the implementation of the CAN communication in the Matlab Simulink interface. However, the UDP communication time of 1 ms is acceptable and the expected complete NMPC calculation and communication time on the Raspberry Pi 400 is 5 ms which is still well below the 22 ms calculation window. Therefore UDP communication will be used for subsequent testing.

### 6.3.3 Realtime testing of acados on Raspberry Pi 400

As discussed in the previous section, running the NMPC on a separate embedded processor has shown in simulation the potential to greatly improve the calculation rate to allow for cycle-to-cycle control without a one cycle delay. The chosen system

is the Raspberry Pi 400 which communicates with the main engine controller the dSPACE MABX II using UDP. The NMPC structure and which parts are running where is schematically shown in Figure 6.6. Essentially the NMPC calculation on the Raspberry Pi is standalone from the main engine controller other than receiving the current measured states (MPRR, IMEP, CA50 and NO<sub>x</sub> emissions) and IMEP reference the entire NMCP is executed on the Raspberry Pi. Both the optimizer and DNN LSTM model are calculated on the Pi where the calculated setpoint for the next cycle is fed to the MABX II.

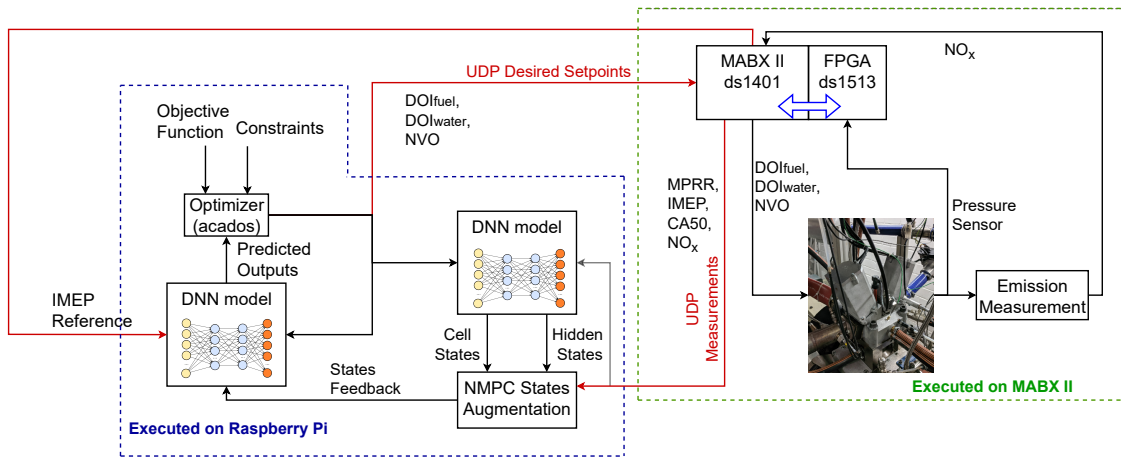


Figure 6.6: Block diagram of the split LSTM-NMPC structure running on a Raspberry Pi 400 and MABX II

This decoupled structure also allows for the NMPC on the Pi to be updated without re-flashing the MABX II and allows for implementing changes in the NMPC or acados optimizer without restarting the engine during experimental testing. The modular design of this structure allows for the NMPC to be executed on any external processor that can interface with the MABX II over UDP. Figure 6.7 shows the experimental testing of the designed LSTM based NMPC. Again a reference IMEP is provided to the controller where a requested load is to be followed. Additionally, a reference CA50 is again provided to keep the combustion phasing at the efficient operation that

comes with operation at 6 CAD aTDC. Again the constraints specified in Table 6.1 are implemented.

The IMEP tracking performance of the implemented controller is quite good with an RMSE of 0.133 bar while being able to keep the CA50 at the setpoint of 6 CAD aTDC with an RSME of 1.83 CAD. As expected the duration of fuel injection corresponds with the requested IMEP. This trend is also seen in the NVO duration where with increased fueling a reduced NVO is requested to allow more air into the cylinder. Interestingly, again similar to the NMPC results on the MABX II the NMPC (Figure 6.5) there is only a single cycle where water is desired during an increased load step. This result is different from the offline simulation, shown in Figure 6.2, where for the first two load steps water was requested even after the step, however, the experimental results do match the second half of the offline simulation where it was seen that water was only requested during IMEP changes.

The constraints applied to the system are obeyed for almost all of the cycles as there is only one cycle (cycle 562) where the constraint for MPRR of 15 bar/CAD is slightly exceeded with a cycle having a MPRR of 17 bar/CAD. This same cycle results in NO<sub>x</sub> emissions of 305 ppm which is slightly above the constraint specified (300 ppm). These constraint violations are only slightly over the specified thresholds and therefore are likely due to a model-plant mismatch. This results from the model predicting a slightly lower MPRR and NO<sub>x</sub> emissions than the experimental measurement.

The prediction of CA50 and IMEP from the LSTM model can be seen in Figure 6.8. The model prediction is similar to the results of the offline model training results shown in Figure 5.23 where both the predicted CA50 and IMEP appear damped compared to the experimental values. However, for both engine performance metrics the model is able to capture the trends quite accurately. For IMEP the model is easily able to capture the trends quite well with an RSME of 0.09 bar while for CA50 the RSME is 1.27 CAD.

One advantage of the LSTM model used in the NMPC implementation is the ability for the cell and hidden states within the LSTM to vary depending on the current output from the engine. This allows the model to adapt as the engine changes and allows the LSTM to adapt while the controller is active. The change in the LSTM states can be seen in Figure 6.9. These states are not physical states but rather states resulting from the structure of deep neural network model used. The adaptation of the states can be seen by the slow change in the states after the steps in IMEP are applied. Specifically a large adaptation in states can be seen at the step applied at cycle 350. Some of the states end up adapting more than others which remain relatively constant during a specific desired IMEP.

The performance of the `acados` NMPC executed on the Raspberry Pi can be evaluated as shown in Figure 6.10. The average execution time for the NMPC is 1.18ms and all the cycles take below 1.4 ms for the NMPC calculation for the 650 cycles tested. The calculation time is about 1 ms slower than the offline simulation results (Figure 6.3) which took place on a PC, however, this is still well below the 22 ms available and shows that if needed the model complexity could be increased when using the Pi 400 as an external processor. As for the required SQP iterations, the limit of three iterations is hit on 71% of the 650 engine cycles which was not the case for the offline simulation. This could be of the variation presented by the real engine and the need for an increase number of iterations to deal with the stochastic behavior of the HCCI combustion process. To check for further improvement in the performance of the NMPC the number of iterations allowed could be increased from three in future testing. Finally the value of the cost function (eqn 6.15) can be seen in Figure 6.10c. When compared the the offline simulation, shown in Figure 6.2, the value of the cost is similar during transients where the spike reach a value of around 3-5. However for a constant IMEP reference the value reaches approximately zero in simulation while in the experiment the cost remains around 0.5. This is a result of

using the same model in the NMPC and for the plant model in the simulation.

Overall, the performance characteristics of the NMPC executed on the Raspberry Pi allows for cycle to cycle control which provides good load tracking and obeys constraints. This shows the potential of experimental use of real-time NMPC using a deep neural network model.

### **6.3.4 LSTM based NMPC Development Time**

When compared to the extremely time and resource intensive of developing a traditional look-up table based control strategy the LSTM based NMPC provides a method to allow for rapid controller development. Often one of the most significant drawback of blackbox based models is the required training time for the model itself. For the work presented, the data collected (65,000 engine cycles) took around one and a half hours of testbench time to collect the required continuous data set. Using this data the overall model training time varies based on the hardware used for the training purpose and the model settings selected. For the model presented in Section 5.3 and using the training settings given in Table 5.5 using an Intel I7-12700K based PC with a NVIDIA RTX 3090Ti the training took between two and three hours depending on the dataset used. Then taking advantage of the the flexibility of `acados` allows for the model to be implemented in around one to two hours depending how many inputs and outputs to the controller are desired. Therefore, using the process developed in this thesis a model based controller can be implemented in under six hours which includes data collection, model training and controller execution. This is significantly less than the time for traditional engine control development which can save time and reduce calibration effort. Another advantage of the proposed control development method is the flexibility of the process which can be applied to other dynamic and highly constrained systems.

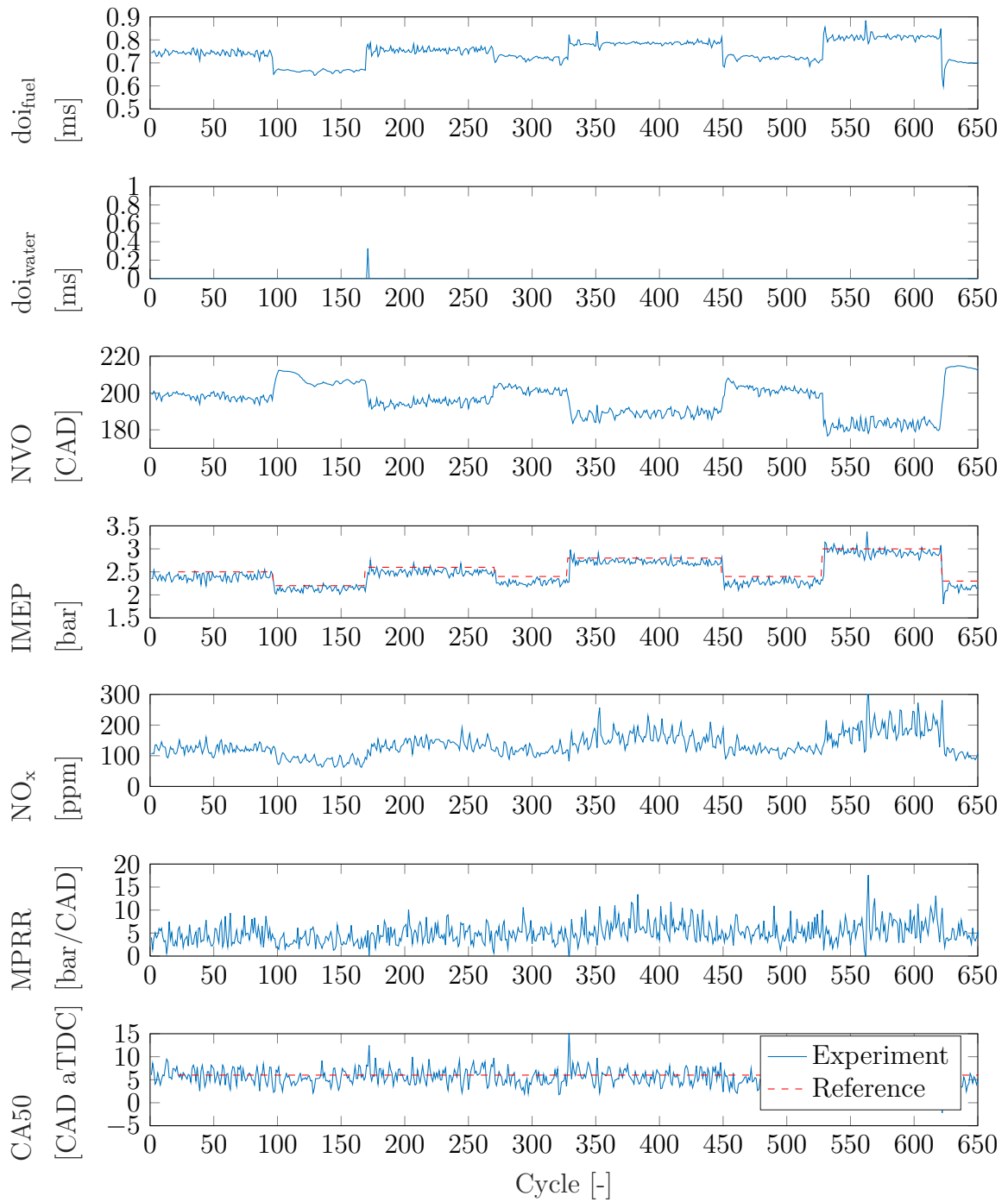


Figure 6.7: Experimental cycle to cycle NMPC implementation on Raspberry Pi 400: Showing IMEP reference tracking performance

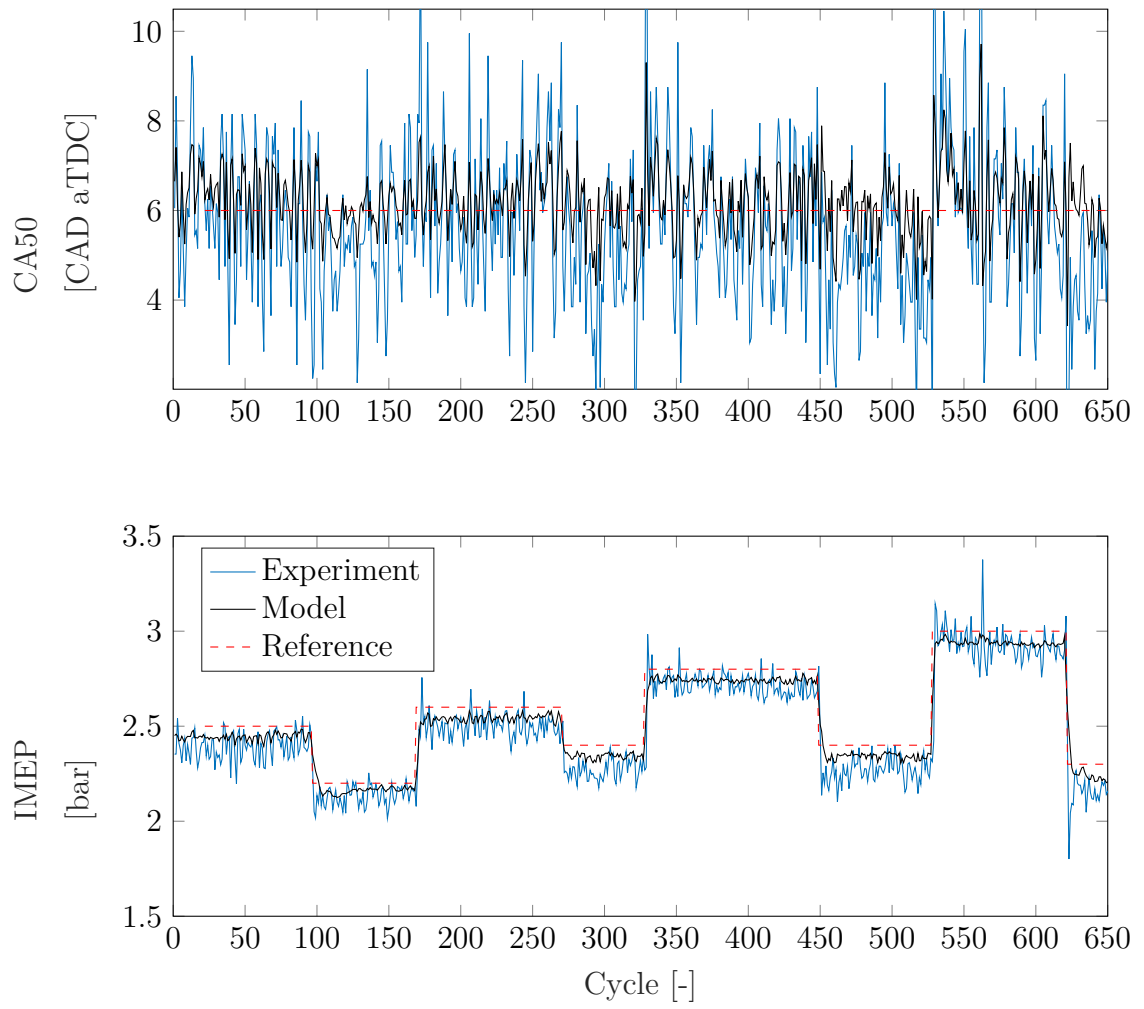


Figure 6.8: Model prediction from LSTM model during experimental implementation on Raspberry Pi 400. a) CA50 b) IMEP

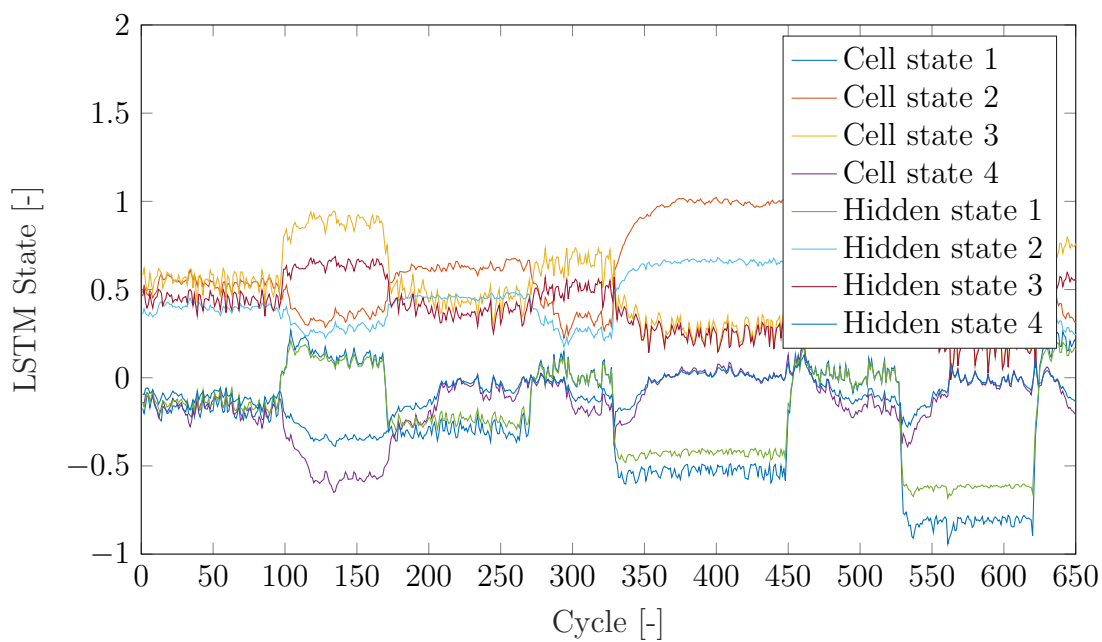


Figure 6.9: LSTM cell and hidden states during experimental implementation on Raspberry Pi 400



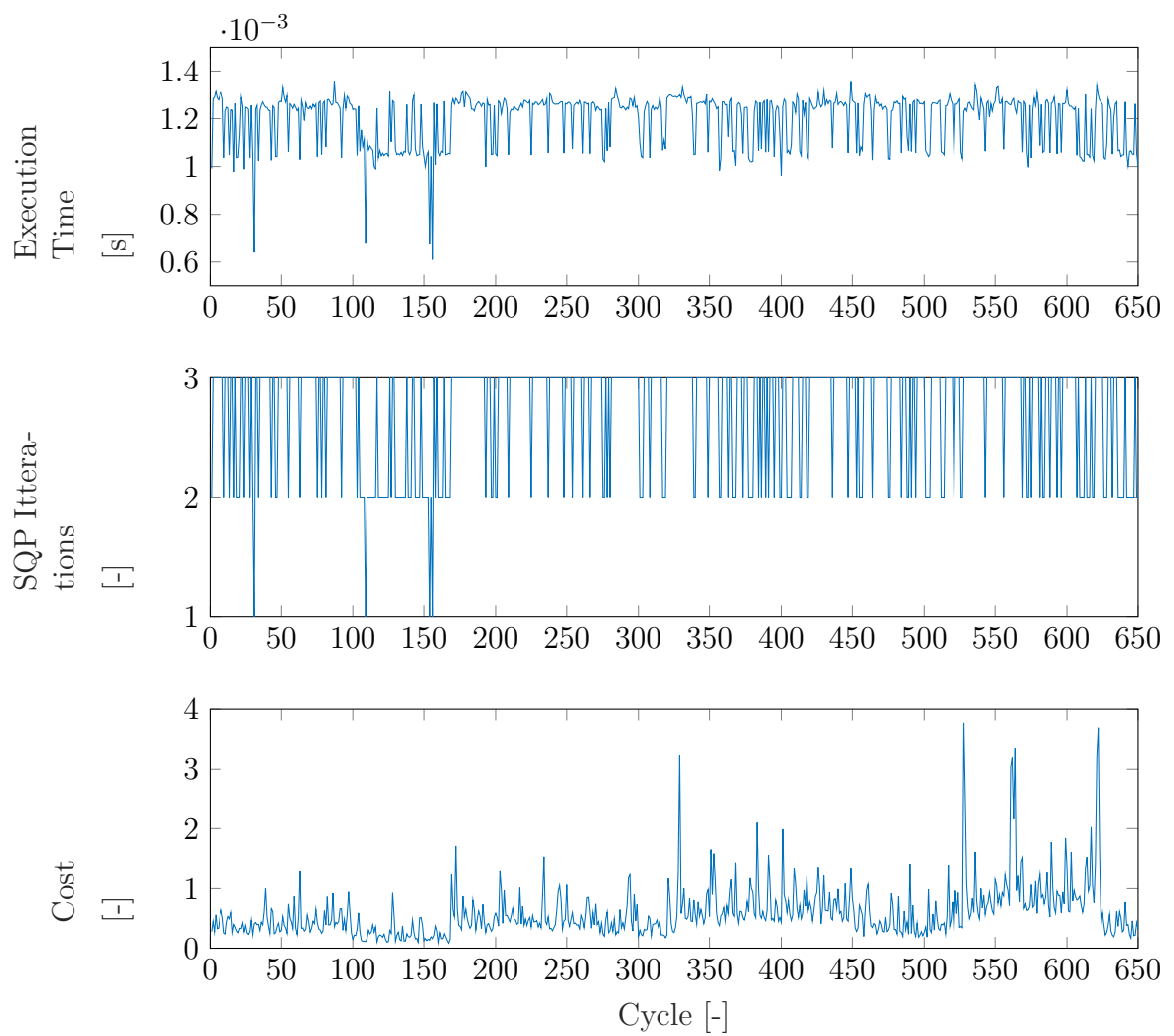


Figure 6.10: acados NMPC performance results on Raspberry Pi 400

## 6.4 Summary of chapter

The developed NMPC has been successfully experimentally implemented in real-time using the `acados` embedded solvers for nonlinear optimal control. As the NMPC requires an accurate and efficient model, a data driven deep neural network utilizing a long short term memory (LSTM) has been developed and implemented for real-time HCCI engine performance and emission prediction. The developed LSTM model with four cell and four hidden states was first tested in an offline simulation which allowed for tuning of the cost function weights. The offline simulation showed acceptable control performance.

The NMPC with a prediction horizon of three, was then successfully implemented on a dSPACE MABX II but the IBM PPC 750GL was not able to complete the online optimization within the 22 ms requirement and required adjustment to the model to allow for cycle-to-next-cycle implementation. Using this model with an added delay allowed a proof of concept of the NMPC to be successfully experimentally tested where IMEP reference tracking was possible, however, the system suffered from significant oscillation due to the controller inability to directly interact with the subsequent cycle.

For true cycle-to-cycle control, the next step was to find an external processor that would decrease the calculation time of the NMPC to allow for real-time implementation. Of the six devices tested, all showed the possibility to meet the real-time requirements even with a delay of 1 ms introduced due to the UDP communication between the MABX II and external system. Therefore to keep implementation costs low a Raspberry Pi 400 was chosen to test on the engine testbench. This will allow excellent flexibility for future NMPC engine control with machine learning based models.

The implementation of the `acados` NMPC on the Raspberry Pi 400 has been experimentally shown to follow an IMEP reference with an RSME of 0.133 bar. The

tested controller was also able to observe constraints and keep the combustion phasing around the target value. Another significant benefit of the developed control strategy is the ability to develop a controller extremely quickly. By collecting a reasonable amount of engine data it is possible to train the LSTM based model and implement a NMPC on the hardware in significantly less time than traditional table based engine control.

# Chapter 7

## Conclusions

This thesis examined the experimental application of advanced control strategies to improve the stability and reduce the emissions of homogeneous charge compression ignition combustion. The first set of control strategies presented in Chapter 3 focused on real-time in-cycle control strategies. These control strategies utilized high speed actuators, such as high speed spark and direct water injection to quickly influence the in-cylinder state. The parallel computational power of a FPGA is used to calculate combustion metrics and to drive high speed actuation during the engine cycle. Improvements in combustion stability and a reduction in emissions in HCCI engines was achieved.

Then by combining an in-cycle controller with a cycle-to-cycle controller allowed for a multiscale control approach to be developed and experimentally tested in Chapter 4. Using an artificial neural network (ANN) to model the combustion process the purely in-cycle controller was compared with the multiscale control approach. Experimental testing of the multiscale controller resulted in a significant stability increase over the pure in-cycle controller.

The next advanced control strategy focused on using a more traditional engine control architecture (processor based control) but incorporated Machine Learning (ML) for engine modeling, control, and emission reduction. First the development of a physics based (white-box) model to capture the combustion kinetics was created

as a baseline to compare the ML models. Then the application of ML to steady state emission modeling was presented in Section 5.2 by modeling engine-out Nitrogen Oxide, Hydrocarbon, Carbon Monoxide, and Carbon Dioxide emissions using a Support Vector Machine. Then, a third model utilizing a deep neural network with a long short term memory (LSTM) model allowed for the creation of a dynamic model for HCCI emissions and performance was presented in Section 5.3. This model provided a transient HCCI model capable of predicting engine out emissions and engine performance for a range of engine loads from 2.5-4.0 bar IMEP.

The integration of ML and Model Predictive Control (MPC) was examined in Chapter 6. The engine model needed for MPC was a ML model which was feasible for realtime implementation. The design and experimental implementation of the MPC on a prototyping engine control unit as well as on various other external processors was presented. A short summary of the main results from each chapter are described next.

## **7.1 In-cycle HCCI Control using Rapid Spark**

This thesis provided the development and experimental implementation of an in-cycle control strategy. The controller took advantage of the addition of a new high speed ignition system with the ability to add energy to the cylinder in under  $10 \mu\text{s}$  ( $0.09^\circ \text{ CAD}$  at  $1500 \text{ rev/min}$ ). The in-cylinder state calculated every  $0.1^\circ \text{ CAD}$  using a detailed gas exchange model on the FPGA provides the real time heat release and cylinder state which are used as an input for the controller. Then considering the in-cylinder state at various crank angles it was determined that by considering the cylinder state at  $5 \text{ cad aTDC}$  provides sufficient information to determine if the auto-ignition process has started or if additional energy is required. Using this information, two controller inputs are considered: the cylinder pressure; and accumulated heat release at  $5 \text{ cad aTDC}$ . Both controllers were able to successfully reduce the stan-

standard deviation of combustion phasing and IMEP at multiple HCCI operating points. In one operating point, before the controller is activated there was eight cycles with a combustion phasing later than  $30^\circ$  CAD aTDC (considered a misfire). After the controller is activated there are no cycles with a combustion phasing greater than  $30^\circ$  CAD aTDC, showing the controller is able to completely eliminate misfire cycles. The controller commanded a spark in approximately 10% of cycles at the operating point tested. Additionally, the addition of the reactive spark system has led to nitrogen oxide levels similar to the pure HCCI case while showing an improvement of 6.6% in unburnt hydrocarbon emission over pure HCCI. This work has been published in [8, 9].

## 7.2 Multiscale HCCI Control

To combine the in-cycle controller benefits with a secondary cycle-to-cycle controller a multi-scale control approach was developed. The first time scale is cycle-to-cycle where the combustion phasing of one cycle is used to determine the amount of water to inject during the following NVO recompression phase. Then using the heat released in the NVO recompression in combination with the data from the previous main combustion allows for determining the amount of water to inject (if any) into the main compression phase as well if the fueling amount should be adapted. When this information is used in an artificial neural network (ANN) it can be inverted and used to directly influence the HCCI combustion. This control approach provided a clear increase in indicated efficiency from 28.09 % to 29.42 %, improved IMEP from 2.88 bar up to 3.02 bar and helped to stabilize the operation point with reductions of standard deviation for IMEP and CA50 by more than 65 %. The advantages of MIMO controller over the single actuation in-cycle controller showed a significant improvement in the prediction and prevention of misfire cycles. This work has been published in [5].

## 7.3 HCCI Performance and Emission Modeling

For a model predictive controller the first step was to identify a suitable model for HCCI. This thesis presented three models for capturing the HCCI combustion process. These models cover the range of modeling strategies from fully physics based kinetics modeling to fully machine learning based black-box models. The first model investigated was a one zone skeletal physical kinetics model which is able to provide a breakdown of the chemical species in the cylinder. The goal for using this model was to utilize the parallel computational capability of the FPGA to propagate forward in time the chemical process reactions, since these reactions are mostly independent they can be solved simultaneously. An offline kinetics model was successfully created and showed that the chemical components of the cylinder could be estimated using a relatively simple kinetics model with 34 species and 36 reactions. However, when the model was transferred to the FPGA for realtime implementation at an engine speed of 840 rpm the current FPGA hardware is only able to calculate 0.1886 CAD of combustion during one revolution of the engine. Since a model with a resolution of 0.1 CAD is needed this meant that with the current FPGA hardware, it was not possible to run a fully white-box kinetics model in real-time.

To speedup the model, a machine learning (ML) model was then investigated. The challenge with ML based models is that the model quality is strongly based on the data used to train the model. The advantage of ML models are the relatively straight forward implementation of the model in real-time. The first ML model developed is a support vector machine model used to predict HCCI engine out emissions. This model predicted the Nitrogen Oxide, Carbon Monoxide, Carbon Dioxide and unburnt Hydrocarbon emissions from the HCCI engine. The effect of different machine learning approaches and feature set selection on the model quality for HCCI emissions prediction was the result of this work. First, linear and non-linear SVM models were compared to a traditional ANN model. This comparison showed that for a small data

set that SVM based models were more robust to changes in feature selection and were better able to avoid local minimums compared to ANN. In this case the SVM model has a more consistent model prediction. For each of the four emissions examined the best model type was determined by taking the highest average  $R^2$  value less the variance in  $R^2$  over the various feature sets. This led to the NLSVM being selected for 3 of the emissions and LSVM for  $\text{NO}_x$  prediction. Then, the individual feature sets were examined. The base feature sets were extended by multiplying individual features together to explore feature interactions. By comparing the individual features with the base feature set (L7) the feature set with an improved accuracy but also considers the increase in model complexity was chosen. In this thesis, the emission models chosen for control purposes for CO, HC,  $\text{NO}_x$ , and  $\text{CO}_2$  are NLSVM-L13, NLSVM-L7, LSVM-L13, and NLSVM-L7, respectively. The  $\text{NO}_x$  and CO models have the largest prediction error, of  $R^2 = 69\%$  and  $R^2 = 74\%$ , respectively. While the HC and  $\text{CO}_2$  models are quite accurate with an  $R^2 = 84\%$  and  $R^2 = 86\%$ , respectively. Even the  $\text{NO}_x$  model, which produced the least accurate prediction, was still able to capture the trends in  $\text{NO}_x$  production. This work has been published in [13].

Finally a transient engine performance and emissions model for HCCI was created using a deep neural network containing a long short term memory (LSTM) layer. This model requires significantly more data than the SVM model, however, due to the recurrent neural network the model is able to capture time dependencies in the data. This resulted in an accurate model which is able to provide a model for transient engine operation with an error less than 5% for all four model outputs (IMEP, CA50, MPRR and  $\text{NO}_x$ ). Of the models developed and tested, the LSTM model showed the greatest accuracy while preserving a simple model structure to allow for real-time implementation. MPRR is the most difficult parameter for the model to predict as shown by the 4.7% error in validation while other outputs are predicted with less than 4% error. The prediction of IMEP is quite good with an error of less than 2.8%



for both training and test data. Even the  $\text{NO}_x$  emission prediction, which is generally more difficult to predict compared to the engine performance parameters, is within 4% for the validation data. The modeling process was applied to diesel combustion and has been published in [1, 2].

## 7.4 Nonlinear Model Predictive Control Experimental Implementation

Using the developed LSTM based DNN model a nonlinear model predictive controller (NMPC) has been designed and experimentally tested. To successfully implement the NMPC, the open-source package `acados` which enables the integration of embedded solvers for nonlinear optimal control has been used. Using the LSTM model with four cell and four hidden states was first tested in an offline simulation which enabled the offline tuning of the controller and allowed for controller feasibility testing. The NMPC was then successfully implemented on a dSPACE MABX II where the IBM PPC 750GL. However, only cycle to next cycle control was implemented as the NMPC optimization was not able to complete the NMPC calculations within the engine cycle time (22 ms) requirement. However, even with this one cycle delay, the NMPC was able to provide acceptable IMEP reference tracking although some oscillation due to the controller inability to directly interact with the subsequent cycle.

Then the NMPC online optimization was tested on various external processors to determine which provided the calculation rate needed. Of the six various devices tested all showed the possibility to meet the real-time requirements even with a delay of 1-2 ms introduced due to the UDP communication between the MABX II and NMPC external processor. To keep NMPC costs low a Raspberry Pi 400 was chosen to test on the engine testbench. The implementation of the `acados` NMPC on the Raspberry Pi 400 did not have a cycle delay and has been experimentally shown to follow an IMEP reference with an RSME of 0.133 bar. The tested controller was also

able to observe constraints and keep the combustion phasing around the target value. The experimentally tested NMPC has an average execution time of 1.18 ms and a maximum time of 1.4 ms. With the current controller setup the controller complexity could be further increased if needed to allow for a more detailed calculation as only 10% of the cycle time is used.

Another significant benefit of the developed LSTM based NMPC control strategy is the ability to develop a controller extremely quickly. By collecting a reasonable amount of engine data (around 65,000 cycles in this case) it is possible to train the LSTM based model and implement a NMPC on the engine controller in significantly less time than traditional table based engine control strategies.

## 7.5 Future Work

This work involved the components of: modeling, control development, actuator integration, and the analysis of realtime hardware capability. The components were implemented on a complex experimental system, the HCCI engine, which has actuator and sensor constraints to protect the engine. As such, the successful implementation of a new controller is time consuming and beyond the scope of this thesis. However, possible ways to extend the work presented in this thesis include:

- Combining the rapid ignition controller with the multi-scale control strategy to add another actuator for the controller to use.
- Including more emissions in the DNN LSTM model to allow for constraints on a wide array of emissions.
- Use the steady state SVM model to provide an estimation of uHC emissions from NOx emissions.
- Transfer learning of the developed model on the RWTH engine to the UofA engine.

- Use the developed realtime LSTM MPC method on a variety of systems such as Hydrogen-Diesel dual fuel combustion and electric motor thermal derating.
- Use the developed realtime LSTM MPC method for control of HCCI utilizing other zero-carbon fuels such as ammonia or hydrogen.
- Including additional inputs to the LSTM model such as: exhaust gas temperature (useful to set constraints on exhaust temperatures for aftertreatment system), ion current (add more information on the chemical state), burn duration (to allow for better shaping of heat release) or others.
- Test control strategy with additional disturbances and check sensitivity to operating point.

# Bibliography

- [1] A. Norouzi *et al.*, “Deep learning and nonlinear model predictive control integration for compression ignition engine emission reduction,” 2022.
- [2] A. Norouzi *et al.*, “Deep learning based model predictive control for compression ignition engines,” *Control Engineering Practice*, vol. 127, p. 105 299, 2022.
- [3] D. Gordon *et al.*, “Development and experimental validation of a field programmable gate array–based in-cycle direct water injection control strategy for homogeneous charge compression ignition combustion stability,” *International Journal of Engine Research*, Apr. 2019, doi:10.1177/1468087419841744.
- [4] D. Gordon *et al.*, “Development and experimental validation of an FPGA based in-cycle direct water injection control strategy for HCCI combustion stability,” in *Symposium for Combustion Control*, 2018, pp. 195–206.
- [5] J. Bedei *et al.*, “Dynamic measurement with in-cycle process excitation of hcci combustion: The key to handle complexity of data-driven control?” *International Journal of Engine Research*, Feb. 2022.
- [6] K. Ebrahimi, D. Gordon, P. Canteenwalla, and C. R. Koch, “Evaluation of ASTM d6424 standard for knock analysis using unleaded fuel candidates on a six cylinder aircraft engine,” *International Journal of Engine Research*, 2021.
- [7] C. Wouters *et al.*, “Evaluation of the potential of direct water injection in HCCI combustion,” in *2019 JSAE/SAE Powertrains, Fuels and Lubricants*, ser. SAE Technical Paper Series, SAE International400 Commonwealth Drive, Warrendale, PA, United States, 2019.
- [8] D. Gordon *et al.*, “Hcci combustion stability improvement using a rapid ignition system,” in *Symposium for Combustion Control*, Jun. 2019.
- [9] D. Gordon *et al.*, “Homogeneous charge compression ignition combustion stability improvement using a rapid ignition system,” *International Journal of Engine Research*, vol. 21, no. 10, pp. 1846–1856, 2020.
- [10] M. Wick *et al.*, “In-cycle control for stabilization of homogeneous charge compression ignition combustion using direct water injection,” *Applied Energy*, 2019.

- [11] A. Norouzi, D. Gordon, M. Aliramezani, and C. Koch, "Machine learning-based diesel engine-out nox reduction using a plug-in pd-type iterative learning control," in *2020 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE, Aug. 2020, pp. 450–455.
- [12] D. Gordon *et al.*, "Negative valve overlap peak pressure based in-cycle control for HCCI combustion using direct water injection," in *Proceedings of Combustion Institute - Canadian Section Spring Technical Meeting*, Kelowna, Canada, May 13-16, 2019.
- [13] D. Gordon *et al.*, "Support vector machine based emissions modeling using particle swarm optimization for homogeneous charge compression ignition engine," *International Journal of Engine Research*, p. 14 680 874 211 055 546, 2021.
- [14] D. C. Gordon *et al.*, "End-to-end deep neural network based nonlinear model predictive control: Experimental implementation on diesel engine emission control," *Energies*, vol. 15, no. 24, 2022.
- [15] D. Gordon, "HCCI modeling and control strategies utilizing water injection," M.S. thesis, University of Alberta, Dec. 2018.
- [16] D. Gordon, A. Norouzi, S. Shahpouri, M. Shahbakhti, and C. R. Koch, "Deep reinforcement learning for emission control in diesel engines," 2022.
- [17] D. G. Kessel, "Global warming—facts, assessment, countermeasures," *Journal of Petroleum Science and Engineering*, vol. 26, no. 1-4, pp. 157–168, 2000.
- [18] I. Yusri, A. A. Majeed, R. Mamat, M. Ghazali, O. I. Awad, and W. Azmi, "A review on the application of response surface method and artificial neural network in engine performance and exhaust emissions characteristics in alternative fuel," *Renewable and Sustainable Energy Reviews*, vol. 90, pp. 665 –686, 2018.
- [19] A. Dewangan, A. Mallick, A. K. Yadav, and R. Kumar, "Combustion-generated pollutions and strategy for its control in CI engines: A review," *Materials Today: Proceedings*, vol. 21, pp. 1728–1733, 2020, International Conference on Mechanical and Energy Technologies.
- [20] S. Davis and R. G. Boundy, "Transportation Energy Data Book: Edition 39," Oak Ridge National Lab.(ORNL), Oak Ridge, TN (United States), Tech. Rep., 2021.
- [21] M Williams and R Minjares, "A technical summary of Euro 6/VI vehicle emission standards," *International Council for Clean Transportation (ICCT)*, Washington, DC, accessed July, vol. 10, p. 2017, 2016.

- [22] E. Parliament and the Council of the European Union, “Regulation (EC) No 715/2007 of the European Parliament and of the Council of 20 June 2007 on type approval of motor vehicles with respect to emissions from light passenger and commercial vehicles (Euro 5 and Euro 6) and on access to vehicle repair and maintenance information.,” *Official Journal of the European Union L Series.*, vol. 171, 2007.
- [23] M. Weiss *et al.*, “Will Euro 6 reduce the nox emissions of new diesel cars?—insights from on-road tests with portable emissions measurement systems (PEMS),” *Atmospheric Environment*, vol. 62, pp. 657–665, 2012.
- [24] EuroVI, “Commission regulation (EU) 2016/646 of 20 april 2016 amending regulation (EC) NO692/2008 as regards emissions from light passenger and commercial vehicles (Euro 6),” in *Euro 6 regulation*, 2016.
- [25] R. H. Stanglmaier and C. E. Roberts, “Homogeneous charge compression ignition (HCCI): benefits, compromises, and future engine applications,” SAE Technical Paper, Tech. Rep., 1999.
- [26] M. Yao, Z. Zheng, and H. Liu, “Progress and recent trends in homogeneous charge compression ignition (HCCI) engines,” *Progress in Energy and Combustion Science*, vol. 35, no. 5, pp. 398–437, 2009.
- [27] M. Shahbakhti and C. Koch, “Characterizing the cyclic variability of ignition timing in a homogeneous charge compression ignition engine fuelled with n-heptane/iso-octane blend fuels,” *International Journal of Engine Research*, vol. 9, no. 5, pp. 361–397, 2008.
- [28] S. Onishi, S. H. Jo, K. Shoda, P. Do Jo, and S. Kato, “Active thermo-atmosphere combustion (ATAC)-a new combustion process for internal combustion engines,” SAE Technical Paper, Tech. Rep., 1979.
- [29] R. K. Maurya and A. K. Agarwal, “Experimental study of combustion and emission characteristics of ethanol fuelled port injected homogeneous charge compression ignition (HCCI) combustion engine,” *Applied Energy*, vol. 88, no. 4, pp. 1169–1180, 2011.
- [30] J. B. Heywood *et al.*, *Internal combustion engine fundamentals*. McGraw-hill New York, 1988, vol. 930.
- [31] M. Fathi, O. Jahanian, and M. Shahbakhti, “Modeling and controller design architecture for cycle-by-cycle combustion control of homogeneous charge compression ignition (HCCI) engines – a comprehensive review,” *Energy Conversion and Management*, vol. 139, pp. 1–19, 2017.
- [32] H. Breitbach, A. Waltner, T. Landefeld, and C. Schwarz, “Lean-burn stratified combustion at gasoline engines,” *MTZ Worldwide*, vol. 74, no. 5, pp. 10–16, 2013.

- [33] P. Langen, T. Melcher, S. Missy, C. Schwarz und E. Schünemann, “Neue BMW Sechs- und Vierzylinder-Ottomotoren mit High Precision Injection und Schichtbrennverfahren,” in *28. Internationales Wiener Motorensymposium 2007*.
- [34] A. Waltner, P. Lückert, U. Schaupp, E. Rau, R. Kemmler, and R. Weller, “Die Zukunftstechnologie des Ottomotors: strahlgeführte Direkteinspritzung mit Piezo- Injektor,” in *27. Internationales Wiener Motorensymposium 2006*, pp. 24–43.
- [35] H. Zhao, *HCCI and CAI engines for the automotive industry* (Woodhead Publishing in mechanical engineering). Boca Raton FL and Cambridge, England: CRC Press and Woodhead Pub., 2007.
- [36] P. Guibert, C. Morin, and S. Mokhtari, “Verbrennungs-steuerung durch Selbstzündung,” *MTZ - Motortechnische Zeitschrift*, vol. 65, no. 2, pp. 122–130, 2004.
- [37] C. Stan and P. Guibert, “Verbrennungssteuerung durch Selbstzündung,” *MTZ - Motortechnische Zeitschrift*, vol. 65, no. 1, pp. 56–62, 2004.
- [38] I. Adcock, “Ice breaker! Mazda’s Skyactiv-X using SpCCI,” *SAE Automotive Engineering*, 2017.
- [39] R. Maurya and A. Agarwal, “Experimental investigations of performance, combustion and emission characteristics of ethanol and methanol fueled HCCI engine,” *Fuel Processing Technology*, vol. 126, pp. 38–48, 2014.
- [40] C. E. Slepicka, “Iterative learning control for fuel robust HCCI,” M.S. thesis, University of Alberta, 2016.
- [41] A. Tsolakis, A. Megaritis, and D. Yap, “Application of exhaust gas fuel reforming in diesel and homogeneous charge compression ignition (HCCI) engines fuelled with biofuels,” *Energy*, vol. 33, no. 3, pp. 462–470, 2008.
- [42] Y. Yang, J. Dec, N. Dronniou, and B. Simmons, “Characteristics of isopentanol as a fuel for HCCI engines,” *SAE International Journal of Fuels and Lubricants*, vol. 3, no. 2, pp. 725–741, 2010.
- [43] B. Rajesh Kumar and S. Saravanan, “Partially premixed low temperature combustion using dimethyl carbonate (DMC) in a DI Diesel engine for favorable smoke/NOx emissions,” *Fuel*, vol. 180, pp. 396–406, 2016.
- [44] J. Mack, D. Schuler, R. Butt, and R. Dibble, “Experimental investigation of butanol isomer combustion in homogeneous charge compression ignition (HCCI) engines,” *Applied Energy*, vol. 165, pp. 612–626, 2016.
- [45] J. H. Mack, S. M. Aceves, and R. W. Dibble, “Demonstrating direct use of wet ethanol in a homogeneous charge compression ignition (HCCI) engine,” *Energy*, vol. 34, no. 6, pp. 782–787, 2009.
- [46] G. Haraldsson, P. Tunestål, B. Johansson, and J. Hyvönen, “HCCI combustion phasing with closed-loop combustion control using variable compression ratio in a multi cylinder engine,” *SAE Technical Papers 2003-01-1830*, 2003.

- [47] L. Manofsky, J. Vavra, D. Assanis, and A. Babajimopoulos, “Bridging the gap between HCCI and SI: Spark-assisted compression ignition,” *SAE Technical Papers 2011-01-1179*, 2011.
- [48] M. J. Atkins and C. R. Koch, “The effect of fuel octane and diluent on homogeneous charge compression ignition combustion,” *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 219, no. 5, pp. 665–675, 2005. eprint: <https://doi.org/10.1243/095440705X11202>.
- [49] A. Ghazimirsaeid and C. R. Koch, “Controlling cyclic combustion timing variations using a symbol-statistics predictive approach in an HCCI engine,” *Applied energy*, vol. 92, pp. 133–146, 2012.
- [50] B. Lehrheuer *et al.*, “A study on in-cycle combustion control for gasoline controlled autoignition,” in *SAE Technical Paper*, SAE International, Apr. 2016.
- [51] S. Saxena and I. D. Bedoya, “Fundamental phenomena affecting low temperature combustion and HCCI engines, high load limits and strategies for extending these limits,” *Progress in Energy and Combustion Science*, vol. 39, no. 5, pp. 457–488, 2013.
- [52] E. Hellstrom, J. Larimore, S. Jade, A. G. Stefanopoulou, and L. Jiang, “Reducing cyclic variability while regulating combustion phasing in a four-cylinder HCCI engine,” *IEEE Transactions on Control Systems Technology*, vol. 22, no. 3, pp. 1190–1197, 2014.
- [53] J. Larimore, “Experimental analysis and control of recompression homogeneous charge compression ignition combustion at the high cyclic variability limit,” Dissertation, The University of Michigan, Ann Arbor, 2014.
- [54] A. Vaughan, “Adaptive machine learning for modeling and control of non-stationary, near chaotic combustion in real-time,” Dissertation, University of Michigan, Ann Arbor, MI 48109, USA, 2015.
- [55] A. Vaughan and S. V. Bohac, “A cycle-to-cycle method to predict HCCI combustion phasing,” in *ASME 2013 Internal Combustion Engine Division Fall Technical Conference*, American Society of Mechanical Engineers, 2013, V001T03A026–V001T03A026.
- [56] R. Stone, *Introduction to internal combustion engines*. Palgrave Macmillan, 2012.
- [57] Z. Wang, H. Liu, and R. D. Reitz, “Knocking combustion in spark-ignition engines,” *Progress in Energy and Combustion Science*, vol. 61, pp. 78 –112, 2017.
- [58] G. P. Merker, C. Schwarz, and R. Teichmann, “Grundlagen Verbrennungsmotoren,” *Simulation der Gemischbildung, Verbrennung, Schadstoffbildung und Aufladung*, vol. 4, 2009.



- [59] A. Cairns and H. Blaxill, “The effects of combined internal and external exhaust gas recirculation on gasoline controlled auto-ignition,” SAE Technical Paper, Tech. Rep., 2005.
- [60] H. Wei, T. Zhu, G. Shu, L. Tan, and Y. Wang, “Gasoline engine exhaust gas recirculation—a review,” *Applied energy*, vol. 99, pp. 534–544, 2012.
- [61] G. H. Abd-Alla, “Using exhaust gas recirculation in internal combustion engines: A review,” *energy conversion and management*, vol. 43, no. 8, pp. 1027–1042, 2002.
- [62] J. E. Dec, “Advanced compression-ignition engines—understanding the in-cylinder processes,” *Proceedings of the Combustion Institute*, vol. 32, no. 2, pp. 2727–2742, 2009.
- [63] V. Rapp, N. Killingsworth, P. Therkelsen, and R. Evans, “Lean-burn internal combustion engines,” in *Lean Combustion (Second Edition)*, Second Edition, Boston: Academic Press, 2016, pp. 111–146.
- [64] J. Larimore, E. Hellström, J. Sterniak, L. Jiang, and A. G. Stefanopoulou, “Experiments and analysis of high cyclic variability at the operational limits of spark-assisted HCCI combustion,” in *2012 American Control Conference (ACC)*, 2012, pp. 2072–2077.
- [65] E. Nuss, D. Ritter, M. Wick, J. Andert, D. Abel, and T. Albin, “Reduced order modeling for multi-scale control of low temperature combustion engines,” in *Active Flow and Combustion Control 2018*, ser. Notes on Numerical Fluid Mechanics and Multidisciplinary Design, R. King, Ed., vol. 141, Cham: Springer, 2019, pp. 167–181.
- [66] J. A. Eng, “Characterization of pressure waves in HCCI combustion,” in *SAE Powertrain & Fluid Systems Conference & Exhibition*, SAE International, 2002.
- [67] R. K. Maurya, *Characteristics and control of low temperature combustion engines: employing gasoline, ethanol and methanol*. Springer, 2017.
- [68] E. Hellström *et al.*, “Understanding the dynamic evolution of cyclic variability at the operating limits of HCCI engines with negative valve overlap,” *SAE International Journal of Engines*, vol. 5, no. 3, pp. 995–1008, 2012.
- [69] L. Shi, Y. Cui, K. Deng, H. Peng, and Y. Chen, “Study of low emission homogeneous charge compression ignition (HCCI) engine using combined internal and external exhaust gas recirculation (EGR),” *Energy*, vol. 31, no. 14, pp. 2665–2676, 2006.
- [70] K. Stapf, D. Seebach, F. Fricke, S. Pischinger, K. Hoffmann, and D. Abel, “CAI-Engines: modern combustion system to face future challenges,” in *SIA Int. Conference—The Spark Ignition Engine of the Future, Strasbourg*, 2007.
- [71] D. Cleary and G. Silvas, “Unthrottled engine operation with variable intake valve lift, duration, and timing,” in *SAE World Congress & Exhibition*, SAE International, 2007.

- [72] K. Ebrahimi, C. Koch, and A. Schramm, “A control oriented model with variable valve timing for HCCI combustion timing control,” in *SAE Technical Paper*, SAE International, Apr. 2013.
- [73] R. V. Klikach, “Investigation and analysis of RCCI using NVO on a converted spark ignition engine,” M.S. thesis, University of Alberta, 2018.
- [74] S. L. Kokjohn, R. M. Hanson, D. A. Splitter, and R. D. Reitz, “Experiments and modeling of dual-fuel HCCI and PCCI combustion using in-cylinder fuel blending,” *SAE International Journal of Engines*, vol. 2, no. 2, pp. 24–39, Nov. 2009.
- [75] M Thewes *et al.*, “Water injection-high power and high efficiency combined,” in *25th Aachen Colloquium Automobile*, 2016, pp. 345–380.
- [76] J. Benajes, S. Molina, A. García, E. Belarte, and M. Vanvolsem, “An investigation on RCCI combustion in a heavy duty diesel engine using in-cylinder blending of diesel and gasoline fuels,” *Applied Thermal Engineering*, vol. 63, no. 1, pp. 66–76, 2014.
- [77] L. Guzzella and C. Onder, *Introduction to modeling and control of internal combustion engine systems*. Springer Science & Business Media, 2009.
- [78] R. Isermann, “Engine modeling and control,” *Berlin: Springers Berlin Heidelberg*, vol. 1017, 2014.
- [79] S. Trajkovic, A. Milosavljevic, P. Tunestål, and B. Johansson, “FPGA controlled pneumatic variable valve actuation,” in *SAE 2006 World Congress & Exhibition*, SAE International, Apr. 2006.
- [80] C. Wilhelmsson, P. Tunestål, and B. Johansson, “FPGA based engine feedback control algorithms,” eng, in *FISITA 2006 World Automotive Congress*, JSAE, 2006.
- [81] J. Pfluger, J. Andert, H. Ross, and F. Mertens, “Rapid control prototyping for cylinder pressure indication,” *MTZ Worldwide*, vol. 73, no. 11, pp. 38–42, 2012.
- [82] M. Wick, B. Lehrheuer, T. Albin, J. Andert, and S. Pischinger, “Decoupling of consecutive gasoline controlled auto-ignition combustion cycles by field programmable gate array based real-time cylinder pressure analysis,” *International Journal of Engine Research*, vol. 19, no. 2, pp. 153–167, 2018.
- [83] J. K. Arora and M. Shahbakhti, “Real-time closed-loop control of a light-duty RCCI engine during transient operations,” in *SAE Technical Paper*, SAE International, Mar. 2017.
- [84] D. Gordon *et al.*, “Development and experimental validation of a real-time capable FPGA based gas-exchange model for negative valve overlap,” *International Journal of Engine Research*, 2018.

- [85] B. Lehrheuer, B. Morcinkowski, S. Pischinger, and M. Nijs, “Low temperature gasoline combustion – potential, challenges, process modeling and control,” in *Active Flow and Combustion Control 2014*, ser. Notes on numerical fluid mechanics and multidisciplinary design, R. King, Ed., vol. 127, Cham: Springer, 2015, pp. 163–179.
- [86] D. Ritter, J. Andert, D. Abel, and T. Albin, “Model-based control of gasoline-controlled auto-ignition,” *International Journal of Engine Research*, vol. 19, no. 2, pp. 189–201, 2018.
- [87] T. Albin, D. Ritter, R. Zweigel, and D. Abel, “Hybrid multi-objective mpc for fuel-efficient pcci engine control,” in *2015 European Control Conference (ECC)*, pp. 2583–2588.
- [88] J. Andert, M. Wick, B. Lehrheuer, C. Sohn, T. Albin, and S. Pischinger, “Autoregressive modeling of cycle-to-cycle correlations in homogeneous charge compression ignition combustion,” *International Journal of Engine Research*, vol. 7, no. 4, p. 146 808 741 773 104, 2017.
- [89] B. Lawler, J. Lacey, O. Güralp, P. Najt, and Z. Filipi, “Hcci combustion with an actively controlled glow plug: The effects on heat release, thermal stratification, efficiency, and emissions,” *Applied Energy*, vol. 211, pp. 809–819, 2018.
- [90] K. Ebrahimi, A. Schramm, and C. R. Koch, “Feedforward/feedback control of HCCI combustion timing,” in *2014 American Control Conference*, Jun. 2014, pp. 831–836.
- [91] K. Ebrahimi and C. Koch, “Model predictive control for combustion timing and load control in HCCI engines,” in *SAE Technical Paper 2015-01-0822*, SAE International, Apr. 2015.
- [92] S. M. Erlien, A. F. Jungkunz, and J. C. Gerdes, “Multi-cylinder HCCI control with cam phaser variable valve actuation using model predictive control,” in *Proceedings of the 2012 ASME Dynamic Systems and Control Conference*, vol. 2, 2013, pp. 823–832.
- [93] N. Ravi, H.-H. Liao, A. F. Jungkunz, A. Widd, and J. C. Gerdes, “Model predictive control of HCCI using variable valve actuation and fuel injection,” *Control Engineering Practice*, vol. 20, no. 4, pp. 421–430, 2012.
- [94] J. D. López, J. J. Espinosa, and J. R. Agudelo, “LQR control for speed and torque of internal combustion engines,” *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 2230–2235, 2011, 18th IFAC World Congress.
- [95] R. Pfeiffer, G. Haraldsson, J.-O. Olsson, P. Tunest Al, R. Johansson, and B. Johansson, “System identification and LQG control of variable-compression HCCI engine dynamics,” in *Proceedings of the 2004 IEEE International Conference on Control Applications, 2004.*, IEEE, vol. 2, 2004, pp. 1442–1447.

- [96] A. Norouzi, K. Ebrahimi, and C. R. Koch, “Integral discrete-time sliding mode control of homogeneous charge compression ignition (hcci) engine load and combustion timing,” *IFAC-PapersOnLine*, vol. 52, no. 5, pp. 153–158, 2019, 9th IFAC Symposium on Advances in Automotive Control AAC 2019.
- [97] M. R. Amini, M. Shahbakhti, S. Pan, and J. K. Hedrick, “Discrete adaptive second order sliding mode controller design with application to automotive control systems with model uncertainties,” in *2017 American Control Conference (ACC 2017)*, IEEE, 2017, pp. 4766–4771.
- [98] J. S. Souder and J. K. Hedrick, “Adaptive sliding mode control of air–fuel ratio in internal combustion engines,” *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, vol. 14, no. 6, pp. 525–541, 2004.
- [99] E. Lavretsky and K. A. Wise, “Robust adaptive control,” in *Robust and adaptive control*, Springer, 2013, pp. 317–353.
- [100] B. K. Irdmousa, S. Z. Rizvi, J. M. Velni, J. Naber, and M. Shahbakhti, “Data-driven modeling and predictive control of combustion phasing for RCCI engines,” *American Control Conference (ACC 2019)*, pp. 1–6, 2019.
- [101] L. A. Basina, B. K. Irdmousa, J. M. Velni, H. Borhan, J. D. Naber, and M. Shahbakhti, “Data-driven modeling and predictive control of maximum pressure rise rate in RCCI engines,” in *IEEE Conference on Control Technology and Applications (CCTA 2020)*, IEEE, 2020, pp. 94–99.
- [102] A. Norouzi, H. Heidarifar, M. Shahbakhti, C. R. Koch, and H. Borhan, “Model predictive control of internal combustion engines: A review and future directions,” *Energies*, vol. 14, no. 19, 2021.
- [103] W. H. Kwon and S. H. Han, *Receding horizon control: model predictive control for state models*. Springer Science & Business Media, 2006.
- [104] J. H. Lee, “Model predictive control: Review of the three decades of development,” *International Journal of Control, Automation and Systems*, vol. 9, no. 3, p. 415, 2011.
- [105] M. Aliramezani, C. R. Koch, and M. Shahbakhti, “Modeling, diagnostics, optimization, and control of internal combustion engines via modern machine learning techniques: A review and future directions,” *Progress in Energy and Combustion Science*, vol. 88, p. 100967, 2022.
- [106] D. Liao-McPherson, M. Huang, S. Kim, M. Shimada, K. Butts, and I. Kolmanovsky, “Model predictive emissions control of a diesel engine airpath: Design and experimental evaluation,” *International Journal of Robust Nonlinear Control*, vol. 30, no. 17, pp. 7446–7477, 2020.
- [107] S. Di Cairano, J. Doering, I. V. Kolmanovsky, and D. Hrovat, “Model predictive control of engine speed during vehicle deceleration,” *IEEE Transactions on Control Systems Technology*, vol. 22, no. 6, pp. 2205–2217, 2014.

- [108] J. Andert, M. Wick, B. Lehrheuer, C. Sohn, T. Albin, and S. Pischinger, "Autoregressive modeling of cycle-to-cycle correlations in homogeneous charge compression ignition combustion," *International Journal of Engine Research*, vol. 19, no. 7, pp. 790–802, 2018.
- [109] B. Morcinkowski, "Simulative Analyse von zyklischen Schwankungen der kontrollierten ottomotorischen Selbstzündung," Dissertation, RWTH Aachen University, Aachen, 2015.
- [110] S. Choi, M. Ki, and K. Min, "Development of an on-line model to predict the in-cylinder residual gas fraction by using the measured intake/exhaust and cylinder pressures," *International Journal of Automotive Technology*, vol. 11, no. 6, pp. 773–781, 2010.
- [111] X. V. Nguyen, J. Chan, S. Romano, and J. Bailey, "Effective global approaches for mutual information based feature selection," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 2014, pp. 512–521.
- [112] M. A. Hall, "Correlation-based Feature Selection for Machine Learning. PhD thesis. The University of Waikato (1999),"
- [113] Z. M. Hira and D. F. Gillies, "A review of feature selection and feature extraction methods applied on microarray data," *Advances in bioinformatics*, vol. 2015, 2015.
- [114] M. Bidarvatan, V. Thakkar, M. Shahbakhti, B. Bahri, and A. Abdul Aziz, "Grey-box modeling of HCCI engines," *Applied Thermal Engineering*, vol. 70, no. 1, pp. 397–409, 2014.
- [115] A. Mohammad, R. Rezaei, C. Hayduk, T. O. Delebinski, S. Shahpouri, and M. Shahbakhti, "Hybrid physical and machine learning-oriented modeling approach to predict emissions in a diesel compression ignition engine," in *2021 SAE World Congress, 2021-01-0496*, SAE International, 2021.
- [116] S. Shahpouri, A. Norouzi, C. Hayduk, R. Rezaei, M. Shahbakhti, and C. R. Koch, "Hybrid machine learning approaches and a systematic model selection process for predicting soot emissions in compression ignition engines," *Energies*, vol. 14, no. 23, 2021.
- [117] A. Norouzi, M. Aliramezani, and C. R. Koch, "A correlation-based model order reduction approach for a diesel engine NOx and brake mean effective pressure dynamic model using machine learning," *International Journal of Engine Research*, vol. 22, no. 8, pp. 2654–2672, 2021.
- [118] M. Yu, X. Tang, Y. Lin, and X. Wang, "Diesel engine modeling based on recurrent neural networks for a hardware-in-the-loop simulation system of diesel generator sets," *Neurocomputing*, vol. 283, pp. 9–19, 2018.

- [119] S. Javed, Y. S. Murthy, R. U. Baig, and D. P. Rao, “Development of ann model for prediction of performance and emission characteristics of hydrogen dual fueled diesel engine with jatropha methyl ester biodiesel blends,” *Journal of Natural Gas Science and Engineering*, vol. 26, pp. 549–557, 2015.
- [120] J. Rezaei, M. Shahbakhti, B. Bahri, and A. A. Aziz, “Performance prediction of HCCI engines with oxygenated fuels using artificial neural networks,” *Applied Energy*, vol. 138, pp. 460–473, 2015.
- [121] Y. Bao, J. Mohammadpour Velni, and M. Shahbakhti, “An online transfer learning approach for identification and predictive control design with application to RCCI engines,” in *Dynamic Systems and Control Conference*, ser. Dynamic Systems and Control Conference, American Society of Mechanical Engineers, vol. 84270, Oct. 2020, V001T21A003.
- [122] H. Bendu, B. Deepak, and S Murugan, “Application of GRNN for the prediction of performance and exhaust emissions in HCCI engine using ethanol,” *Energy Conversion and Management*, vol. 122, pp. 165–173, 2016.
- [123] T. Gangopadhyay, A. Locurto, J. B. Michael, and S. Sarkar, “Deep learning algorithms for detecting combustion instabilities,” in *Dynamics and Control of Energy Systems*, A. Mukhopadhyay, S. Sen, D. N. Basu, and S. Mondal, Eds. Singapore: Springer Singapore, 2020, ch. 13, pp. 283–300.
- [124] V. M. Janakiraman, X. Nguyen, and D. Assanis, “An ELM based predictive control method for HCCI engines,” *Engineering Applications of Artificial Intelligence*, vol. 48, pp. 106–118, 2016.
- [125] V. M. Janakiraman, X. Nguyen, and D. Assanis, *Nonlinear model predictive control of a gasoline hcci engine using extreme learning machines*.
- [126] A. Vaughan and S. Bohac, “Real-time, adaptive machine learning for non-stationary, near chaotic gasoline engine combustion time series,” *Neural Networks*, vol. 70, pp. 18–26, 2015.
- [127] Y. Bao, J. M. Velni, and M. Shahbakhti, “Epistemic Uncertainty Quantification in State-Space LPV Model Identification Using Bayesian Neural Networks,” *IEEE Control Systems Letters*, vol. 5, no. 2, pp. 719–724, 2020.
- [128] H. Yaşar, G. Çağıl, O. Torkul, and M. Şişci, “Cylinder pressure prediction of an HCCI engine using deep learning,” *Chinese Journal of Mechanical Engineering*, vol. 34, no. 1, pp. 1–8, 2021.
- [129] A. Raut, B. Irdmoussa, and M. Shahbakhti, “Dynamic modeling and model predictive control of an RCCI engine,” *Control Engineering Practice*, vol. 81, pp. 129–144, 2018.
- [130] W. Pan, M. Korkmaz, J. Beeckmann, and H. Pitsch, “Nonlinear identification modeling for PCCI engine emissions prediction using unsupervised learning and neural networks,” SAE Technical Paper, Tech. Rep., 2020.

- [131] X. Niu, C. Yang, H. Wang, and Y. Wang, "Investigation of ANN and SVM based on limited samples for performance and emissions prediction of a CRDI-assisted marine diesel engine," *Applied Thermal Engineering*, vol. 111, pp. 1353–1364, 2017.
- [132] E. Ahmed, M. Usman, S. Anwar, H. M. Ahmad, M. W. Nasir, and M. A. I. Malik, "Application of ANN to predict performance and emissions of SI engine using gasoline-methanol blends," *Science Progress*, vol. 104, no. 1, p. 00 368 504 211 002 345, 2021.
- [133] A. N. Bhatt and N. Shrivastava, "Application of artificial neural network for internal combustion engines: A state of the art review," *Archives of Computational Methods in Engineering*, pp. 1–23, 2021.
- [134] M. Aliramezani, A. Norouzi, and C. R. Koch, "Support vector machine for a diesel engine performance and NOx emission control-oriented model," *21st IFAC World Congress in Berlin, Germany*, 2020.
- [135] V. Vapnik and A. Lerner, "Generalized portrait method for pattern recognition," *Automation and Remote Control*, vol. 24, no. 6, pp. 774–780, 1963.
- [136] V. Vapnik and A. Chervonenkis, "A note on class of perceptron," *Automation and Remote Control*, vol. 24, 1964.
- [137] V. M. Janakiraman, X. Nguyen, J. Sterniak, and D. Assanis, "A system identification framework for modeling complex combustion dynamics using support vector machines," in *Informatics in Control, Automation and Robotics*, Springer, 2014, pp. 297–313.
- [138] V. M. Janakiraman, X. Nguyen, J. Sterniak, and D. Assanis, "Identification of the dynamic operating envelope of HCCI engines using class imbalance learning," *IEEE transactions on neural networks and learning systems*, vol. 26, no. 1, pp. 98–112, 2014.
- [139] E. Gani and C. Manzie, "Indicated torque reconstruction from instantaneous engine speed in a six-cylinder SI engine using support vector machines," SAE Technical Paper, Tech. Rep., 2005.
- [140] G Najafi *et al.*, "SVM and ANFIS for prediction of performance and exhaust emissions of a SI engine with gasoline–ethanol blended fuels," *Applied Thermal Engineering*, vol. 95, pp. 186–203, 2016.
- [141] H. Bendu, B. Deepak, and S. Murugan, "Multi-objective optimization of ethanol fuelled HCCI engine performance using hybrid GRNN–pso," *Applied Energy*, vol. 187, pp. 601–611, 2017.
- [142] X. Li, S. Wu, X. Li, H. Yuan, and D. Zhao, "Particle swarm optimization-support vector machine model for machinery fault diagnoses in high-voltage circuit breakers," *Chinese Journal of Mechanical Engineering*, vol. 33, no. 1, pp. 1–10, 2020.


- [143] M Taghavi, A Gharehghani, F. B. Nejad, and M Mirsalim, “Developing a model to predict the start of combustion in HCCI engine using ANN-ga approach,” *Energy Conversion and Management*, vol. 195, pp. 57–69, 2019.
- [144] R. Hassan, B. Cohanim, O. De Weck, and G. Venter, “A comparison of particle swarm optimization and the genetic algorithm,” in *46th AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference*, 2005, p. 1897.
- [145] G. N. Kouziokas, “Svm kernel based on particle swarm optimized vector and bayesian optimized svm in atmospheric particulate matter forecasting,” *Applied Soft Computing*, vol. 93, p. 106410, 2020.
- [146] A. Géron, *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O’Reilly Media, 2019.
- [147] B.-K. Jeon and E.-J. Kim, “LSTM-Based Model Predictive Control for Optimal Temperature Set-Point Planning,” *Sustainability*, vol. 13, no. 2, 2021.
- [148] Q. Wang, L. Pan, and K. Y. Lee, “Improving Superheated Steam Temperature Control Using United Long Short Term Memory and MPC,” *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 13345–13350, 2020, 21st IFAC World Congress.
- [149] X. Tang, G. Zhong, K. Yang, J. Wu, and Z. Wei, “Motion planning framework for autonomous vehicle with lstm-based predictive model,” in *2021 5th CAA International Conference on Vehicular Control and Intelligence (CVCI)*, 2021, pp. 1–5.
- [150] S Batool, J Naber, and M Shahbakhti, “Data-Driven Modeling and Control of Cyclic Variability of an Engine Operating in Low Temperature Combustion Modes,” *IFAC-PapersOnLine*, 2021, Modeling, Estimation and Control Conference (MECC 2021).
- [151] A. S. Ira *et al.*, “A machine learning approach for tuning model predictive controllers,” in *15th International Conference on Control, Automation, Robotics and Vision (ICARCV 2018)*, 2018, pp. 2003–2008.
- [152] S. Wang, D. Yu, J. Gomm, G. Page, and S. Douglas, “Adaptive neural network model based predictive control for air–fuel ratio of SI engines,” *Engineering Applications of Artificial Intelligence*, vol. 19, no. 2, pp. 189–200, 2006.
- [153] Y. Hu, H. Chen, P. Wang, H. Chen, and L. Ren, “Nonlinear model predictive controller design based on learning model for turbocharged gasoline engine of passenger vehicle,” *Mechanical Systems and Signal Processing*, vol. 109, pp. 74–88, 2018.
- [154] K. Lee, S. Cho, N. Kim, and K. Min, “A study on combustion control and operating range expansion of gasoline HCCI,” *Energy*, vol. 91, pp. 1038–1048, 2015.



- [155] X. Duan, M.-C. Lai, M. Jansons, G. Guo, and J. Liu, "A review of controlling strategies of the ignition timing and combustion phase in homogeneous charge compression ignition (HCCI) engine," *Fuel*, vol. 285, p. 119 142, 2021.
- [156] J. Zheng, W. Yang, D. L. Miller, and N. P. Cernansky, "A skeletal chemical kinetic model for the hcci combustion process," *SAE Transactions*, pp. 898–912, 2002.
- [157] B. Bahri, M. Shahbakhti, and A. Aziz, "Real-time modeling of ringing in HCCI engines using artificial neural networks," *Energy*, vol. 125, pp. 509 –518, 2017.
- [158] M Bidarvatan, V Thakkar, M Shahbakhti, B Bahri, and A. A. Aziz, "Grey-box modeling of hcci engines," *Applied Thermal Engineering*, vol. 70, no. 1, pp. 397–409, 2014.
- [159] A. Norouzi *et al.*, "Machine Learning Integrated with Model Predictive Control for Imitative Optimal Control of Compression Ignition Engines," *arXiv preprint arXiv:2204.00142*, 2022.
- [160] R. R. Chladny, "Modeling and control of automotive gas exchange valve solenoid actuators," Ph.D. dissertation, University of Alberta, 2007.
- [161] Pat. DE10033595C1, 2001.
- [162] <https://www.dspace.com/en/inc/home/products/hw/micautob/microautobox2.cfm>, 2020.
- [163] D. Gordon *et al.*, "Development and experimental validation of a real-time capable field programmable gate array–based gas exchange model for negative valve overlap," *International Journal of Engine Research*, vol. 930, p. 146 808 741 878 849, 2018.
- [164] <https://www.mksinst.com/f/multigas-2030-ftir-continuous-gas-analyzer>, 2020.
- [165] F. Pischinger and P. Kreuter, *Arrangement for electromagnetically operated actuators*, US Patent 4,515,343, Jul. 1985.
- [166] F Pischinger *et al.*, "Electromechanical variable valve timing," *Automotive Engineering International*, 1999.
- [167] *Operation of the electromechanical variable valve train - EMVT closed loop engine*, FEV Motorentechnik, Jul. 2002.
- [168] B. Lehrheuer, D.-I. R. Savelsberg, P. D.-I. J. Andert, and D.-I. M. Thewes, "Potential of in-cycle combustion control for the auto ignition process of gasoline engines," in *Symposium for Combustion Control*, 2015.
- [169] DENSO, DENSO. (), [Online]. Available: <https://www.denso-technic.com/everything-you-need-to-know-about-ignition-coil-charge-up#:~:text=Howlongdoesacoil,millisecondsformanymoderncoils..>

- [170] N. Prakash, J. B. Martz, and A. G. Stefanopoulou, “A phenomenological model for predicting the combustion phasing and variability of spark assisted compression ignition (saci) engines,” in *ASME 2015 Dynamic Systems and Control Conference*, American Society of Mechanical Engineers, 2015, V001T11A004–V001T11A004.
- [171] V. Natarajan, V. Sick, D. Reuss, and G. Silvas, “Effect of spark-ignition on combustion periods during spark-assisted compression ignition,” *Combustion Science and Technology*, vol. 181, no. 9, pp. 1187–1206, 2009.
- [172] L. Xu, X.-S. Bai, C. Li, P. Tunestål, M. Tunér, and X. Lu, “Combustion characteristics of gasoline dci engine in the transition from hcci to ppc: Experiment and numerical analysis,” *Energy*, vol. 185, pp. 922–937, 2019.
- [173] K. Ebrahimi and C. R. B. Koch, “Real-time control of hcci engine using model predictive control,” in *2018 Annual American Control Conference (ACC)*, IEEE, 27.06.2018 - 29.06.2018, pp. 1622–1628.
- [174] M. Wick, J. Bedei, J. Andert, B. Lehrheuer, S. Pischinger, and E. Nuss, “Dynamic measurement of hcci combustion with self-learning of experimental space limitations,” *Applied Energy*, vol. 262, p. 114 364, 2020.
- [175] M. Huang, K. Zaseck, K. Butts, and I. Kolmanovsky, “Rate-based model predictive controller for diesel engine air path: Design and experimental evaluation,” *IEEE Transactions on Control Systems Technology*, vol. 24, no. 6, pp. 1922–1935, 2016.
- [176] K. Kiš and M. Klaučo, “Neural network based explicit mpc for chemical reactor control,” *Acta Chimica Slovaca*, vol. 12, no. 2, pp. 218–223, 2019.
- [177] G. Blomeyer, “Physics based in-cycle combustion model for real-time online emissions calculation on a prototype homogenous charge compression ignition (hcci) engine,” M.S. thesis, RWTH Aachen University, Nov. 2020.
- [178] T. Berg, O. Thiele, S. Seefeldt, and R. Vanhaelst, “Measurement of in-cylinder mixture formation by optical indication,” *MTZ worldwide*, vol. 74, no. 6, pp. 26–30, 2013.
- [179] J. Wang, J. Bedei, J. Deng, J. Andert, D. Zhu, and L. Li, “Detection of transient low-temperature combustion characteristics by ion current – the missing link for homogeneous charge compression ignition control?” *Applied Energy*, p. 116 299, 2020.
- [180] C. Wouters, “Development and Validation of a Model-Based Control for HCCI Combustion using Direct Water Injection,” *Eindhoven University of Technology*, 2018.
- [181] Jerzembeck, Peters, Pepiot-Desjardins, and Pitsch, “Laminar burning velocities at high pressure for primary reference fuels and gasoline: Experimental and numerical investigation,” *Combustion and Flame*, pp. 292–301, 2009.
- [182] “A semi-detailed chemical kinetic model of a gasoline surrogate fuel for internal combustion engine applications,” *Fuel*, vol. 113, pp. 347–356, 2013.

- [183] T. Tsurushima, “A new skeletal prf kinetic model for hcci combustion,” *Proceedings of The Combustion Institute - PROC COMBUST INST*, vol. 32, pp. 2835–2841, Dec. 2009.
- [184] Curran, Gaffuri, Pitz, and Westbrook, “A comprehensive modeling study of iso-octane oxidation,” in *Combustion and Flame*. 2002, vol. 129, pp. 253–280.
- [185] <https://www.ansys.com/products/fluids/ansys-chemkin-pro>, 2020.
- [186] “<https://cantera.org/>” (2020), [Online]. Available: <https://cantera.org/>.
- [187] S. Tanaka, F. Ayala, and J. Keck, “Combustion flame,” pp. 467–481, 2003.
- [188] G. F. Hohenberg, “Advanced approaches for heat transfer calculations,” SAE Technical paper, Tech. Rep., 1979.
- [189] D. O. Allison, “Polynomial approximations of thermodynamic properties of arbitrary gas mixtures over wide pressure and density ranges,” *NASA Langley Research Center Hampton*, 1972.
- [190] N. Peters, “Technische Verbrennung,” pp. 39–42, 2010.
- [191] S. B. Fiveland and D. N. Assanis, “Development of a two-zone hcci combustion model accounting for boundary layer effects,” *SAE Transactions*, vol. 110, pp. 1030–1044, 2001.
- [192] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [193] A. J. Smola and B. Schölkopf, “A tutorial on support vector regression,” *Statistics and computing*, vol. 14, no. 3, pp. 199–222, 2004.
- [194] R. Bellman, “The theory of dynamic programming,” *Bulletin of the American Mathematical Society*, vol. 60, no. 6, pp. 503–515, 1954.
- [195] W. Karush, “Minima of functions of several variables with inequalities as side constraints,” *M. Sc. Dissertation. Dept. of Mathematics, Univ. of Chicago*, 1939.
- [196] H. W. Kuhn and A. W. Tucker, *Nonlinear programming, in (j. neyman, ed.) proceedings of the second Berkeley symposium on mathematical statistics and probability*, 1951.
- [197] M. Aliramezani, A. Norouzi, and C. R. Koch, “A grey-box machine learning based model of an electrochemical gas sensor,” *Sensors and Actuators B: Chemical*, p. 128 414, 2020.
- [198] A. Norouzi, M. Masoumi, A. Barari, and S. Farrokhpour Sani, “Lateral control of an autonomous vehicle using integrated backstepping and sliding mode controller,” *Proceedings of the Institution of Mechanical Engineers, Part K: Journal of Multi-body Dynamics*, vol. 233, no. 1, pp. 141–151, 2019.
- [199] A. Norouzi, H. Adibi-Asl, R. Kazemi, and P. F. Hafshejani, “Adaptive sliding mode control of a four-wheel-steering autonomous vehicle with uncertainty using parallel orientation and position control,” *International Journal of Heavy Vehicle Systems*, vol. 27, no. 4, pp. 499–518, 2020.

- [200] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” *arXiv preprint arXiv:1502.03167*, 2015.
- [201] S. Shahpouri, A. Norouzi, C. Hayduk, R. Rezaei, M. Shahbakhti, and C. R. Koch, “Soot emission modeling of a compression ignition engine using machine learning,” *IFAC-PapersOnLine*, 2021, Modeling, Estimation and Control Conference (MECC 2021).
- [202] A. Norouzi *et al.*, “Integration of deep learning and nonlinear model predictive control for emission reduction of compression ignition combustion engines: A simulation study,” *arXiv preprint arXiv:2204.00139*, 2022.
- [203] R. Verschueren *et al.*, *Acados: A modular open-source framework for fast embedded optimal control*, 2019.
- [204] M. Diehl, H. J. Ferreau, and N. Haverbeke, “Efficient numerical methods for nonlinear mpc and moving horizon estimation,” in *Nonlinear Model Predictive Control: Towards New Challenging Applications*, L. Magni, D. M. Raimondo, and F. Allgöwer, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 391–417.
- [205] G. Frison, D. Kouzoupis, J. Jørgensen, and M. Diehl, “An efficient implementation of partial condensing for nonlinear model predictive control,” Dec. 2016, pp. 4457–4462.
- [206] R. Verschueren *et al.*, “Acados – a modular open-source framework for fast embedded optimal control,” *Mathematical Programming Computation*, Oct. 2021.
- [207] A. Domahidi and J. Jerez, *Forces professional*, Embotech AG, <https://embotech.com/FORCES-Pro>, 2014–2019.
- [208] A. Zanelli, A. Domahidi, J. Jerez, and M. Morari, “Forces nlp: An efficient implementation of interior-point methods for multistage nonlinear nonconvex programs,” *International Journal of Control*, vol. 93, no. 1, pp. 13–29, 2020.
- [209] S. Mehrotra, “On the implementation of a primal-dual interior point method,” *SIAM Journal on optimization*, vol. 2, no. 4, pp. 575–601, 1992.
- [210] G. Frison and M. Diehl, “Hpipm: A high-performance quadratic programming framework for model predictive control,” *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 6563–6569, 2020.
- [211] H. J. Ferreau, C. Kirches, A. Potschka, H. G. Bock, and M. Diehl, “Qpoases: A parametric active-set algorithm for quadratic programming,” *Mathematical Programming Computation*, vol. 6, no. 4, pp. 327–363, 2014.
- [212] *Acados* .

# Appendix A: Ph.D. Publications

## A.1 Peer Reviewed Journal Papers

1. **D.C. Gordon**, A. Norouzi, A. Winkler, J. McNally, E. Nuss, D. Abel, M. Shahbakhti, J. Andert, C. R. Koch, End-to-End Deep Neural Network Based Nonlinear Model Predictive Control: Experimental Implementation on Diesel Engine Emission Control, *Energies*, 2022.
  - **My contributions:** Experimental implementation of NMPC using *acados* on dSPACE MABX II. Drafted sections of manuscript and revised entire submission.
2. A. Norouzi, S. Shahpouri, **D.C. Gordon**, A. Winkler, E. Nuss, D. Abel, J. Andert, M. Shahbakhti, C. R. Koch, Deep learning based model predictive control for compression ignition engines, *Control Engineering Practice*, 2022.
  - **My contributions:** Implemented LSTM model developed by Armin in Simulink using *acados*. Setup of simulation environment to test NMPC. Drafted sections of manuscript and revised entire submission.
3. K. Ebrahimi, **D.C. Gordon**, P. Canteenwalla, C.R. Koch, Evaluation of ASTM D6424 standard for knock analysis using unleaded fuel candidates on a six cylinder aircraft engine, *International Journal of Engine Research*, 2022.
  - **My contributions:** Created figures and analyzed results using data collected by K. Ebrahimi. Literature review for inclusion of existing standards. Drafted

manuscript and responsible for all revisions.

4. J. Bedei, M. Oberlies, P. Schaber, **D.C. Gordon**, E. Nuss, L. Li, J. Andert, Dynamic measurement with in-cycle process excitation of HCCI combustion: The key to handle complexity of data-driven control?, *International Journal of Engine Research*, 2022.
  - **My contributions:** Created online gas exchange model for calculation of in-cylinder state. Revised entire submission.
5. **D.C. Gordon**, A. Norouzi, G. Blomeyer, J. Bedei, M. Aliramezani, J. Andert, C.R. Koch, Support Vector Machine Based Emissions Modeling using Particle Swarm Optimization for Homogeneous Charge Compression Ignition Engine, *International Journal of Engine Research*, 2021.
  - **My contributions:** Collected experimental data and processed data for use in ML algorithms. Identified useful combustion features and necessary model outputs. Drafted manuscript and revised entire submission.
6. **D.C. Gordon**, C. Wouters, S. Kinoshita, M. Wick, B. Lehrheuer, J. Andert, S. Pischinger, C.R. Koch, Homogeneous charge compression ignition combustion stability improvement using a rapid ignition system, *International Journal of Engine Research*, 2020.
  - **My contributions:** Adapted rapid igniton system to engine. Designed and collected experimental data. Analysis of data and creation of all figures. Drafted manuscript and revised entire submission.
7. C. Wouters, **D.C. Gordon**, T. Ottenwalder, Wick, B. Lehrheuer, J. Andert, S. Pischinger, Evaluation of the Potential of Direct Water Injection in HCCI Combustion, *SAE Technical Paper*, 2019.

- **My contributions:** Implemented the controller on the FPGA and ran the engine to collect the experimental data. I processed the data and provided Mr. Wouters with the results. Drafted one section of the manuscript and revised entire submission.
8. **D.C. Gordon**, C. Wouters, M. Wick, B. Lehrheuer, J. Andert, C.R. Koch, S. Pischinger, Development and experimental validation of a field programmable gate array based in-cycle direct water injection control strategy for homogeneous charge compression ignition combustion stability. *International Journal of Engine Research*, 2019.
    - **My contributions:** Creation of the injection controller on the FPGA hardware. Ran experimental testing on the engine and analysis of data. Drafted the manuscript and revised entire submission.
  9. M. Wick, J. Bedei; **D.C. Gordon**, C. Wouters, B. Lehrheuer, E. Nuss, J. Andert, C.R. Koch, In-cycle control for stabilization of homogeneous charge compression ignition combustion using direct water injection. *Applied Energy*, 2019. Implementation of the controller on the FPGA hardware. Ran experimental testing on the engine and analysis of data. Drafted 2 sections of the manuscript.
  10. **D.C. Gordon**, C. Wouters, M. Wick, B. Lehrheuer, J. Andert, C.R. Koch, S. Pischinger, Development and experimental validation of a real-time capable FPGA based gas-exchange model for negative valve overlap. *International Journal of Engine Research*, 2018.
    - **My contributions:** Ported the entire gas exchange process to the FPGA. Performed all offline simulations and ran experimental testing on the engine and compared online and simulation results. Drafted the manuscript and revised

entire submission.

## A.2 Peer Reviewed Conference Papers in Proceedings

1. A. Norouzi, S. Shahpouri, **D.C. Gordon**, A. Winkler, E. Nuss, M. Shahbakhti, and C. R. Koch, Machine Learning Integrated with Model Predictive Control in Compression Ignition Combustion Engines Control, *10<sup>th</sup> Symposium on Advances in Automotive Control (AAC2022)* August 2022, Columbus, Ohio.
  - **My contributions:** Created first draft of manuscript and provided analysis of results. Created presentation for conference.
2. **D.C. Gordon**, J. McNally, A. Norouzi, and C. R. Koch, Experimental Investigation of Hydrogen Diesel Dual Fuel Combustion, *Aachen Hydrogen Colloquium*, May 2022, Aachen Germany.
  - **My contributions:** Ran experimental testing and analysis of results. Created visualizations and gave presentation at conference.
3. J. McNally, **D.C. Gordon**, A. Norouzi, M. Shahbakhti, and C. R. Koch, Experimental Study of Hydrogen Diesel Dual Fuel Engine Characterization, *Canadian Society for Mechanical Engineering International Congress (CSME)*, June 2022, Edmonton, Canada.
  - **My contributions:** Ran experimental testing and analysis of results. Created visualizations and revised submitted abstract.
4. A. Norouzi, S. Shahpouri, **D.C. Gordon**, A. Winkler, E. Nuss, D. Abel, M. Shahbakhti, and C. R. Koch, Deep Learning and Nonlinear Model Predictive Control Integration for Compression Ignition Engine Emission Reduction,



*Canadian Society for Mechanical Engineering International Congress (CSME)*,  
June 2022, Edmonton, Canada.

- **My contributions:** Simulink implementation of NMPC. Created visualizations.
5. **D.C. Gordon**, A. Norouzi, S. Shahpouri, M. Shahbakhti, and C. R. Koch, Deep Reinforcement Learning for Emission Control in Diesel Engines, *Canadian Society for Mechanical Engineering International Congress (CSME)*, June 2022, Edmonton, Canada.
    - **My contributions:** Ran experimental testing and analysis of results. Created visualizations and wrote abstract. Gave conference presentation.
  6. A. Norouzi, **D.C. Gordon**, M. Aliramezani, C.R. Koch, Machine Learning-based Diesel Engine-Out NOx Reduction Using a plug-in PD-type Iterative Learning Control, *4<sup>th</sup> IEEE Conference on Control Technology and Applications (CCTA 2020)*, Montreal, QB, Canada, August 2020.
    - **My contributions:** Ran experimental testing and analysis of ILC results. Provided first draft of manuscript and provided revisions.
  7. C. Wouters, **D.C. Gordon**, M. Wick, B. Lehrheuer, J. Andert, S. Pischinger, Evaluation of the Potential of Direct Water Injection in HCCI Combustion, *2019 JSAE/SAE Powertrains, Fuels and Lubricants International Meeting*, Kyoto, Japan, August 2019.
    - **My contributions:** Designed testing plan to highlight impact of water injection on HCCI. Ran experimental testing and analysis of results. Wrote results section and created visualizations.

8. **D.C. Gordon**, C. Wouters, S. Kinoshita, M. Wick, B. Lehrheuer, J. Andert, S. Pischinger, C.R. Koch, HCCI Combustion Stability Improvement Using a Rapid Ignition System, *4th Symposium for Combustion Control (SCC 2019)*, Aachen, Germany, June 2019.
  - **My contributions:** Adapted rapid ignition system to work on test engine. Designed testing plan and ran experimental tests to show benefit of rapid ignition. Completed analysis of results and created visualizations. Wrote manuscript and gave conference presentation.
9. **D.C. Gordon**, C. Wouters, S. Kinoshita, M. Wick, B. Lehrheuer, J. Andert, S. Pischinger, C.R. Koch, Negative valve overlap peak pressure based in-cycle control for HCCI combustion using direct water injection *Proceedings of Combustion Institute - Canadian Section Spring Technical Meeting (CICS2019)*, Kelowna, Canada, May 2019.
  - **My contributions:** Designed testing plan to highlight impact of water injection on HCCI. Ran experimental testing and analysis of results. Created visualizations and gave conference presentation.
10. **D.C. Gordon**, C. Wouters, S. Kinoshita, M. Wick, B. Lehrheuer, J. Andert, S. Pischinger, C.R. Koch, Development and Experimental Validation of an FPGA Based In-Cycle Control Strategy for HCCI Combustion Stability *3rd Symposium for Combustion Control (SCC2018)*, Aachen, Germany, June 2018.
  - **My contributions:** Ported gas exchange model to the FPGA and ran experimental testing. Compared simulation and online implementation. Created visualizations and gave conference presentation.

## A.3 Technical Posters

1. **D. Gordon**, A. Norouzi, C.R. Koch, AI-based Advance Control Methods for next generation combustion engines, *2021 Future Energy Systems Research Symposium*, Sept 20, 2021, Edmonton, Canada.
2. M. Aliramezani, A. Norouzi, **D. Gordon**, C.R. Koch, Emission reduction of internal combustion engines with advanced control and machine learning techniques, *Future Energy Systems Real World Industry Mixer*, Feb 20, 2020.
3. **D. Gordon**, A. Norouzi, M. Aliramezani, C.R. Koch, Combustion Control Research – *University of Alberta, Canadian Graduate Engineering Consortium*, Sept 2019