Model-Guided Data-Driven Optimization and Control for Internal Combustion Engine Systems

Qingyuan Tan
University of Windsor

Follow this and additional works at: https://scholar.uwindsor.ca/etd

Recommended Citation
https://scholar.uwindsor.ca/etd/7625

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000ext. 3208.
Model-Guided Data-Driven Optimization and Control for Internal Combustion Engine Systems

By

Qingyuan Tan

A Dissertation
Submitted to the Faculty of Graduate Studies through the Department of Electrical and Computer Engineering in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy at the University of Windsor

Windsor, Ontario, Canada

2018

©2018 Qingyuan Tan
Model-Guided Data-Driven Optimization and Control for Internal Combustion Engine Systems

by

Qingyuan Tan

APPROVED BY:

________________________________________
C.R. Koch, External Examiner
University of Alberta

________________________________________
J. Tjong,
Department of Mechanical, Automotive & Materials Engineering

________________________________________
M. Saif,
Department of Electrical and Computer Engineering

________________________________________
B. Shahrrava,
Department of Electrical and Computer Engineering

________________________________________
X. Chen, Co-Advisor
Department of Electrical and Computer Engineering

________________________________________
M. Zheng, Co-Advisor
Department of Mechanical, Automotive & Materials Engineering

Nov. 28, 2018
DECLARATION OF ORIGINALITY

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone’s copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office and that this thesis has not been submitted for a higher degree to any other University or Institution.
ABSTRACT

The incorporation of electronic components into modern Internal Combustion, IC, engine systems have facilitated the reduction of fuel consumption and emission from IC engine operations. As more mechanical functions are being replaced by electric or electronic devices, the IC engine systems are becoming more complex in structure. Sophisticated control strategies are called in to help the engine systems meet the drivability demands and to comply with the emission regulations.

Different model-based or data-driven algorithms have been applied to the optimization and control of IC engine systems. For the conventional model-based algorithms, the accuracy of the applied system models has a crucial impact on the quality of the feedback system performance. With computable analytic solutions and a good estimation of the real physical processes, the model-based control embedded systems are able to achieve good transient performances. However, the analytic solutions of some nonlinear models are difficult to obtain. Even if the solutions are available, because of the presence of unavoidable modeling uncertainties, the model-based controllers are designed conservatively.

The data-driven control algorithms, on the other hand, are the strategies that rely solely on the relationships between the control input and the measured system output. By utilizing the measurement only, data-driven algorithms require no prior-information of the system. The algorithms do not need system models for the development of the controllers. Data-driven algorithms are good alternatives for solving optimization problems when the construction of accurate models of the systems are difficult, costly, or time-consuming. Though robust, because of the lack of system knowledge, data-driven algorithms constantly suffer from slow convergence during optimization. Moreover, the demand for readily available measurement also makes the algorithms, if not impossible, difficult to foresee and prevent degradation of system performance.

In this dissertation, the Model-Guided Data-Driven, MGDD, algorithms are proposed and applied to the optimization and control of systems related to IC engine operations. Compared to the conventional model-based algorithms, the proposed methods have relaxed requirements on the accuracy and the complexity of the system models. The
knowledge provided by the models become guidance for the data-driven algorithms. As a result, the transient performance of conventional data-driven optimization is improved. By adjusting the signal flow between the system models and the data-driven algorithms, different MGDD algorithms are proposed for different applications.

Finally, simulations and experimental results on the optimization of injection timing for a Compression Ignition, CI, engine and the control of an Electronic Throttle Body, ETB, are provided in this work to evaluate the effectiveness of the proposed MGDD algorithms.

*Keywords:* Model-guided, data-driven, optimization, feedback, IC engine, experiment, simulation.
DEDICATION

To mom and dad:

It's impossible to thank you adequately for everything you've done, from loving me unconditionally to raising me as a person of integrity. You have instilled virtues and taught me how to celebrate and embrace life. I could not have asked for better parents or role-models in this life.
ACKNOWLEDGMENTS

With great honor, I am sincerely grateful to my supervisors, Dr. Xiang Chen and Dr. Ming Zheng, for their support and guidance throughout my Ph.D. studies at the University of Windsor. Their wisdom and enthusiasms for research have inspired me to face every challenge encountered throughout the development of this dissertation. Without the encouragement and guidance from them, the completion of this work would have been impossible.

I have received enormous support from Dr. Ying Tan from the University of Melbourne and my dear colleagues at the University of Windsor during the entire course of my Ph.D. study. I feel very grateful and would like to express my deepest appreciation to Dr. Meiping Wang, Dr. Prasad Divekar, Dr. Ruili Dong, Dr. Shui Yu, Dr. Xuebo Zhang, Dr. Xiao Yu, Dr. Tongyang Gao, Dr. Xiaoye Han, Dr. Tadonori Yanai, Dr. Marko Jeftic, Dr. Shouvik Dev, Zhenyi Yang, Mark Ives, Geraint Bryden, Christopher Aversa, Hua Zhu, Navjot Sandhu, Divyanshu Purohit, Simon Leblanc and Linyan Wang.

I am also grateful for the support from the University of Windsor, AUTO 21, Canada Research Chair Program, Canada Foundation of Innovation, Ontario Innovation Trust, Natural Sciences and Engineering Research Council of Canada.

Qingyuan Tan

Windsor, Ontario, Canada

Nov. 2018
# TABLE OF CONTENTS

DECLARATION OF ORIGINALITY .......................................................... iii
ABSTRACT ......................................................................................... iv
DEDICATION ....................................................................................... vi
ACKNOWLEDGMENTS ....................................................................... vii
LIST OF TABLES ................................................................................. xi
LIST OF FIGURES .............................................................................. xii
NOMENCLATURE ................................................................................ xv

CHAPTER 1. INTRODUCTION................................................................. 1
  1.1 Model-Based Optimization and Control .......................................... 1
  1.2 Data-Driven Optimization and Control .......................................... 4
  1.3 Scope of Work ........................................................................... 5
  1.4 Dissertation Significance ............................................................. 6
  1.5 Dissertation Outline .................................................................. 7

CHAPTER 2. OPTIMIZATION OF INTERNAL COMBUSTION ENGINE
SYSTEMS AND ELECTRONIC THROTTLE CONTROL ......................... 9
  2.1 Optimization of Internal Combustion Engine Systems .................... 9
    2.1.1 Design of Experiments Based Optimization .......................... 9
    2.1.2 Data-Driven Based Optimization ....................................... 12
    2.1.3 Model for a Compression Ignition Engine ......................... 13
  2.2 Electronic Throttle Control ......................................................... 19
    2.2.1 Existing Challenges ............................................................ 21
    2.2.2 Existing Solutions ............................................................... 22

CHAPTER 3. DATA-DRIVEN OPTIMIZATION AND CONTROL .......... 25
  3.1 Simultaneous Perturbation Stochastic Approximation .................. 25
  3.2 Iterative Learning Control ......................................................... 28
  3.3 Extremum Seeking Control ....................................................... 30

CHAPTER 4. EXTREMUM SEEKING CONTROL OF INJECTION TIMING IN A
COMPRESSION IGNITION ENGINE USING A SOFT SENSOR .............. 36
  4.1 Problem Formulation and Extremum Seeking Control Using a Soft Sensor .... 37
    4.1.1 Problem Formulation ......................................................... 37
    4.1.2 Extremum Seeking Control Using a Soft Sensor .................. 40
  4.2 Experimental Setup .................................................................. 41
  4.3 Simulation and Experimental Results .......................................... 44
    4.3.1 Simulation Results ............................................................. 44
4.3.2 Experimental Validation ................................................................. 51
4.4 Summary .......................................................................................... 55

CHAPTER 5. MODEL-GUIDED EXTREMUM SEEKING CONTROL OF INJECTION TIMING IN A COMPRESSION IGNITION ENGINE ................................................................. 56
5.1 Model-Guided Extremum Seeking Control ............................................. 56
5.2 Engine Model Validation ..................................................................... 60
5.3 Optimization of Injection Timing in a Compression Ignition Engine ........ 63
  5.3.1 Applicable Conditions for Model-Guided Extremum Seeking Control .... 63
  5.3.2 Model-Guided Extremum Seeking Control ........................................ 64
  5.3.3 Design of Performance Function .................................................... 65
  5.3.4 Simulation Study ........................................................................... 66
  5.3.5 Experimental Study ....................................................................... 71
5.4 Summary .......................................................................................... 75

CHAPTER 6. MODEL-GUIDED EXTREMUM SEEKING CONTROL OF INJECTION TIMING IN A COMPRESSION IGNITION ENGINE WITH ONLINE MODEL CALIBRATION ......................................................................................................... 76
6.1 Model-Guided Extremum Seeking Control with Online Model Calibration .... 76
6.2 Experimental Study ........................................................................... 78
  6.2.1 Model Online Calibration ............................................................... 78
  6.2.2 Model-Based Optimization Using Online Calibrated Model .............. 79
  6.2.3 Engine Injection Timing Optimization ............................................. 80
6.3 Summary .......................................................................................... 81

CHAPTER 7. MODEL-GUIDED DATA-DRIVEN PREDICTIVE CONTROL FOR THE ACTUATION OF AN ELECTRONIC THROTTLE BODY ................................................................. 83
7.1 Modeling of an Electronic Throttle Body ................................................. 84
  7.1.1 First-Principle Model .................................................................... 84
  7.1.2 Pseudo-Wiener Model .................................................................. 89
7.2 Model Identification and Model Validation ............................................ 91
  7.2.1 Levenberg-Marquardt Algorithm .................................................... 92
  7.2.2 Experimental Setup .................................................................... 94
  7.2.3 Identification and Validation of First-Principle Model ....................... 96
  7.2.4 Identification and Validation of Pseudo-Wiener Model .................... 102
7.3 Model-Guided Data-Driven Predictive Control Strategy ......................... 112
7.4 Experimental Study ........................................................................... 116
  7.4.1 Determination of Prediction Horizon ............................................. 116
7.4.2 Experimental Results ................................................................. 119
7.5 Summary ......................................................................................... 129
CHAPTER 8. CONCLUSIONS AND FUTURE PERSPECTIVES ................. 130
  8.1 Conclusions ..................................................................................... 130
  8.2 Future Perspectives .......................................................................... 131
REFERENCES .......................................................................................... 132
VITA AUCTORIS ..................................................................................... 147
LIST OF TABLES

Table 4-1. Gas analyzers .............................................................................................................. 43
Table 4-2. Model parameters ........................................................................................................ 48
Table 4-3. Summary of test results ................................................................................................ 55
Table 7-1. Parameters for the first-principle ETB model ......................................................... 85
Table 7-2. Identified parameters for the first-principle ETB model ........................................... 99
Table 7-3. Identified parameters for the Pseudo-Wiener models .............................................. 109
Table 7-4. MSE for the experiments shown in Figure 7-32 ..................................................... 125
Table 7-5. MSE for the experiments shown in Figure 7-33 ..................................................... 126
LIST OF FIGURES

Figure 1-1. Main contribution........................................................................................................... 7
Figure 1-2. Dissertation outline ........................................................................................................ 8
Figure 2-1. The model-based DOE steps ......................................................................................... 11
Figure 2-2. Torque-based throttle opening estimation........................................................................ 20
Figure 2-3. Schematic of an ETB .................................................................................................... 21
Figure 2-4. ETB valve opening affected by the flow of air ................................................................. 21
Figure 2-5. Rate-dependent characteristics of an ETB system ......................................................... 22
Figure 3-1. Structure for perturbation-based ESC .......................................................................... 31
Figure 4-1. The ESC structure using soft sensor prediction ............................................................. 41
Figure 4-2. Schematic of the single-cylinder research engine platform ........................................... 42
Figure 4-3. Schematic of the fueling system ..................................................................................... 43
Figure 4-4. Working mechanism of the soft sensor ......................................................................... 45
Figure 4-5. Comparison of steady-state responses among different measurements ................. 45
Figure 4-6. Comparison of transient responses between different sensors................................. 46
Figure 4-7. Maximum cyclic bulk gas temperature applied for simulation studies................. 47
Figure 4-8. Simulation results: application of feedback with no delay ........................................ 49
Figure 4-9. Simulation results: application of feedback with delay .............................................. 49
Figure 4-10. Simulation results: application of a slower perturbation signal ............................... 50
Figure 4-11. Test 1: ESC test results for optimizing the thermal efficiency only ...................... 52
Figure 4-12. Test 2: ESC test starts from a different SOI value ..................................................... 53
Figure 4-13. Test 3: ESC test results for optimizing the thermal efficiency and NOx ........... 54
Figure 4-14. Test 4: ESC test results using the soft sensor ............................................................ 54
Figure 5-1. Schematic for a model-guided ESC structure ............................................................. 60
Figure 5-2. Comparisons between measurement and model output ............................................ 62
Figure 5-3. Model-guided ESC for the optimization of injection timing in a CI engine .......... 64
Figure 5-4. Performance function ................................................................................................ 66
Figure 5-5. Modeled rate of heat release ....................................................................................... 67
Figure 5-6. The effect of model accuracy on model-guided ESC ................................................. 67
Figure 5-7. The effect of model accuracy on the model-guided ESC ......................................... 68
Figure 5-8. Simulation under transient conditions: ESC without model guidance ............... 70
Figure 5-9. Simulation under transient conditions: application of model-guided ESC... 70
Figure 5-10. The measured relationship between ignition delay and SOI.................. 71
Figure 5-11. Engine test results: model-guided ESC.............................................. 72
Figure 5-12. Engine test results: ESC without model guidance .............................. 73
Figure 5-13. Transient engine test: ESC optimization without model guidance ....... 74
Figure 5-14. Transient engine test: Model-guided ESC ........................................... 75
Figure 6-1. Model-guided ESC structure with online model calibration..................... 77
Figure 6-2. Online model calibration........................................................................ 79
Figure 6-3. Gradient descent optimization............................................................... 80
Figure 6-4. Engine test results ................................................................................... 81
Figure 7-1. First-principle ETB model ................................................................. 84
Figure 7-2. Coulomb friction .................................................................................... 86
Figure 7-3. Nonlinear spring.................................................................................... 87
Figure 7-4. Gear backlash......................................................................................... 88
Figure 7-5. The Wiener structure model................................................................. 89
Figure 7-6. Schematic of the ETB test setup............................................................ 95
Figure 7-7. Schematic of the steady flow bench....................................................... 95
Figure 7-8. Control signal (blue) and ETB position (red) .......................................... 98
Figure 7-9. ETB actuated above the limp-home....................................................... 99
Figure 7-10. ETB actuated below the limp-home..................................................... 99
Figure 7-11. Parameter identification for the first-principle ETB mode ................. 100
Figure 7-12. Identified first-principle ETB model..................................................... 100
Figure 7-13. Model validation ................................................................................. 101
Figure 7-14. Poor matching between measurement and model output ................. 102
Figure 7-15. Excitation signal for model identification: above limp-home position. 105
Figure 7-16. Identified Pseudo-Wiener model: above limp-home position ............ 106
Figure 7-17. Data-driven Pseudo-Wiener model: above limp-home position ........ 107
Figure 7-18. Excitation signal for model identification: below limp-home position.. 108
Figure 7-19. Data-driven Pseudo-Wiener model: below limp-home position ........ 108
Figure 7-20. Validation of the PWA model: without air flow condition ................. 110
Figure 7-21. Validation of the PWA model: with air flow condition ....................... 111
Figure 7-22. Validation of the PWB model: without air flow condition ...................... 112
Figure 7-23. Data-driven ETB model ........................................................................ 112
Figure 7-24. The MGDD predictive control algorithm ............................................. 113
Figure 7-25. Reference trajectory ............................................................................. 117
Figure 7-26. Tracking test results: values of the weighting factors are set equal ........ 118
Figure 7-27. Tracking test results: values of the weighting factors are set differently... 119
Figure 7-28. Effect of different performance function designs .................................. 120
Figure 7-29. Control of ETB under air flow conditions ............................................. 120
Figure 7-30. MGDD control structure using two models .......................................... 121
Figure 7-31. Control of ETB: step reference across the limp-home position .......... 122
Figure 7-32. Step tracking comparison: no air flow condition .................................. 125
Figure 7-33. Step tracking comparison: with air flow condition ............................... 126
Figure 7-34. Tracking of sinusoidal references: no flow condition ......................... 127
Figure 7-35. Step tracking comparison: reference crosses the limp-home position .... 128
Figure 7-36. Step tracking comparison: reference crosses the limp-home position .... 129
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>BBM</td>
<td>Black-box model</td>
</tr>
<tr>
<td>CAD</td>
<td>Crank angle degree</td>
</tr>
<tr>
<td>CAN</td>
<td>Controller area network</td>
</tr>
<tr>
<td>CFD</td>
<td>Computational fluid dynamic</td>
</tr>
<tr>
<td>CI</td>
<td>Compression ignition</td>
</tr>
<tr>
<td>DC</td>
<td>Direct current</td>
</tr>
<tr>
<td>DEM</td>
<td>Discrete event model</td>
</tr>
<tr>
<td>DI</td>
<td>Direct injection</td>
</tr>
<tr>
<td>DOC</td>
<td>Diesel oxidation catalyst</td>
</tr>
<tr>
<td>DOE</td>
<td>Design of experiments</td>
</tr>
<tr>
<td>DPF</td>
<td>Diesel particulate filer</td>
</tr>
<tr>
<td>ECU</td>
<td>Engine control unit</td>
</tr>
<tr>
<td>EGR</td>
<td>Exhaust gas recirculation</td>
</tr>
<tr>
<td>EPA</td>
<td>Environmental protection agency</td>
</tr>
<tr>
<td>EMF</td>
<td>Electromotive force</td>
</tr>
<tr>
<td>ESC</td>
<td>Extremum seeking control</td>
</tr>
<tr>
<td>ETB</td>
<td>Electronic throttle-body</td>
</tr>
<tr>
<td>EVO</td>
<td>Exhaust valve opening</td>
</tr>
<tr>
<td>FPM</td>
<td>First-principle model</td>
</tr>
<tr>
<td>FPGA</td>
<td>Field programmable gate array</td>
</tr>
<tr>
<td>GBM</td>
<td>Grey-box model</td>
</tr>
<tr>
<td>HC</td>
<td>Hydrocarbon</td>
</tr>
<tr>
<td>HCCI</td>
<td>Homogeneous charge compression ignition</td>
</tr>
<tr>
<td>IC</td>
<td>Internal combustion</td>
</tr>
<tr>
<td>IMEP</td>
<td>Indicated mean effective pressure</td>
</tr>
<tr>
<td>IPODs</td>
<td>Injector power drivers</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>IVC</td>
<td>Intake valve closing</td>
</tr>
<tr>
<td>LHV</td>
<td>Lower heating value</td>
</tr>
<tr>
<td>LMA</td>
<td>Levenberg-marquardt algorithm</td>
</tr>
<tr>
<td>LNT</td>
<td>Lean NO\textsubscript{x} trap</td>
</tr>
<tr>
<td>LUT</td>
<td>Look-up table</td>
</tr>
<tr>
<td>LTI</td>
<td>Linear time-invariant</td>
</tr>
<tr>
<td>MGDD</td>
<td>Model-guided data-driven</td>
</tr>
<tr>
<td>MPC</td>
<td>Model predictive control</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean squared error</td>
</tr>
<tr>
<td>MVM</td>
<td>Mean value model</td>
</tr>
<tr>
<td>NO\textsubscript{x}</td>
<td>Oxides of nitrogen – NO and NO\textsubscript{2}</td>
</tr>
<tr>
<td>PC</td>
<td>Personal computer</td>
</tr>
<tr>
<td>PID</td>
<td>Proportional-integral-derivative</td>
</tr>
<tr>
<td>PM</td>
<td>Particulate matter</td>
</tr>
<tr>
<td>PWA</td>
<td>Model that covers the ETB response above the limp-home position</td>
</tr>
<tr>
<td>PWB</td>
<td>Model that covers the ETB response below the limp-home position</td>
</tr>
<tr>
<td>PWM</td>
<td>Pulse width modulation</td>
</tr>
<tr>
<td>RPM</td>
<td>Revolutions per minute</td>
</tr>
<tr>
<td>RT</td>
<td>Real-time</td>
</tr>
<tr>
<td>SMC</td>
<td>Sliding model control</td>
</tr>
<tr>
<td>SI</td>
<td>Spark ignition</td>
</tr>
<tr>
<td>SISO</td>
<td>Single-input-single-output</td>
</tr>
<tr>
<td>SOC</td>
<td>Start of combustion</td>
</tr>
<tr>
<td>SOI</td>
<td>Start of injection</td>
</tr>
<tr>
<td>TDC</td>
<td>Top dead center</td>
</tr>
<tr>
<td>TTL</td>
<td>Transistor-transistor logic</td>
</tr>
<tr>
<td>UHC</td>
<td>Unburned hydrocarbon</td>
</tr>
<tr>
<td>VVA</td>
<td>Variable valve actuation</td>
</tr>
</tbody>
</table>
VRFT Virtual reference feedback tuning

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta$</td>
<td>Control input</td>
</tr>
<tr>
<td>$\theta^*$</td>
<td>Optimum control input</td>
</tr>
<tr>
<td>$\hat{\theta}^*$</td>
<td>Optimum control input for the model</td>
</tr>
<tr>
<td>$p$</td>
<td>In-cylinder pressure [bar]</td>
</tr>
<tr>
<td>$J$</td>
<td>Performance function [-]</td>
</tr>
<tr>
<td>$g_r$</td>
<td>Adaptation gain [-]</td>
</tr>
<tr>
<td>$z$</td>
<td>Forward time shift operator [-]</td>
</tr>
<tr>
<td>$A$</td>
<td>Amplitude of the perturbation signal [-]</td>
</tr>
<tr>
<td>$\omega$</td>
<td>Frequency of the perturbation signal [rad/s]</td>
</tr>
<tr>
<td>$u$</td>
<td>Global input [-]</td>
</tr>
<tr>
<td>$y$</td>
<td>System output [-]</td>
</tr>
<tr>
<td>$\hat{y}$</td>
<td>Model output [-]</td>
</tr>
<tr>
<td>$\mathcal{R}$</td>
<td>Set of real numbers [-]</td>
</tr>
<tr>
<td>$S_A$</td>
<td>Amplitude of the excitation signal [-]</td>
</tr>
<tr>
<td>$S_{off}$</td>
<td>Offset of the excitation signal [-]</td>
</tr>
<tr>
<td>$\varphi$</td>
<td>Engine crank angle [$^\circ$]</td>
</tr>
<tr>
<td>$\omega_{RPM}$</td>
<td>Engine speed [RPM]</td>
</tr>
<tr>
<td>$m_{fuel}$</td>
<td>Cyclic injected fuel mass [mg/cycle]</td>
</tr>
<tr>
<td>$Z$</td>
<td>Vector of performance output [-]</td>
</tr>
<tr>
<td>$V$</td>
<td>Cylinder volume [L]</td>
</tr>
<tr>
<td>$l$</td>
<td>Connecting rod length [mm]</td>
</tr>
<tr>
<td>$a$</td>
<td>Crank radius [mm]</td>
</tr>
<tr>
<td>$V_s$</td>
<td>Cylinder swept volume [L]</td>
</tr>
<tr>
<td>$V_c$</td>
<td>Cylinder clearance volume [L]</td>
</tr>
<tr>
<td>$T$</td>
<td>In-cylinder bulk gas temperature [K]</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>Ratio of specific heats [-]</td>
</tr>
</tbody>
</table>
\( N_{im} \) Total moles of in-cylinder gas before combustion [\text{mol}]

\( \bar{R} \) Universal gas constant [\text{J/mol} \cdot \text{K}]

\( p_{in} \) Intake manifold pressure [\text{kpa}]

\( T_{in} \) Intake temperature [\text{K}]

\( \eta_V \) Cylinder volumetric efficiency [-]

\( \delta Q_{hr} \) Rate of heat release [\text{J/CAD}]

\( \delta Q_{ht} \) Rate of wall heat transfer [\text{J/CAD}]

\( Q_{hrMAX} \) Maximum rate of heat release [\text{J/CAD}]

\( A_s \) Surface area of the combustion chamber [\text{mm}^2]

\( B \) Cylinder bore [\text{mm}]

\( \bar{S}_p \) Mean piston speed [\text{m/s}]

\( p_{mot} \) Instantaneous motored cylinder pressure [\text{bar}]

\( C_1 \) and \( C_2 \) Wall heat transfer coefficient [-]

\( T_{wall} \) Combustion chamber wall temperature [\text{K}]

\( \varphi_{CA50} \) Crank angle of 50\% accumulated heat release [\text{CAD}]

\( LHV \) Lower heating value [\text{MJ/kg}]

\( \varphi_{CD} \) Combustion duration [\text{CAD}]

\( p_{ime} \) Indicated mean effective pressure [\text{bar}]

\( dp_{max} \) Peak pressure rise rate [\text{bar/CAD}]

\( \lambda_b \) and \( c \) Parameters for gradient descent optimization [-]

\( x_i \) Normalized cumulative heat release [-]

\( \sigma_1 \) and \( \sigma_2 \) Coefficients for the Wiebe function [-]

\( \varphi_{SOI} \) Crank angle for the start of fuel injection [\text{CAD}]

\( u_{duty}, u_d \) Duty cycle of the PWM signal [-]

\( u_{dr} \) Voltage output from the motor driver [\text{V}]

\( u_{arm} \) Voltage applied across the DC motor [\text{V}]

\( K_{dr} \) Gain of the motor driver circuit [\text{V}]

\( K_{arm} \) Equivalent gain between \( u_{arm} \) and \( i_{arm} \) [\text{A/V}]
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_m$</td>
<td>Torque constant</td>
<td>$[N\cdot m/A]$</td>
</tr>
<tr>
<td>$K_{emf}$</td>
<td>Electromotive force constant</td>
<td>$[V]$</td>
</tr>
<tr>
<td>$K_v$</td>
<td>Viscous friction gain</td>
<td>$[N\cdot m\cdot s/rad]$</td>
</tr>
<tr>
<td>$K_p$</td>
<td>Proportional gain</td>
<td>[-]</td>
</tr>
<tr>
<td>$K_I$</td>
<td>Integral gain</td>
<td>[-]</td>
</tr>
<tr>
<td>$K_D$</td>
<td>Differential gain</td>
<td>[-]</td>
</tr>
<tr>
<td>$N_p$</td>
<td>Length of prediction horizon</td>
<td>[-]</td>
</tr>
<tr>
<td>$i_{arm}$</td>
<td>Armature current</td>
<td>[A]</td>
</tr>
<tr>
<td>$T_d$</td>
<td>Torque caused by Coulomb friction</td>
<td>$[N\cdot m]$</td>
</tr>
<tr>
<td>$T_m$</td>
<td>DC motor torque output</td>
<td>$[N\cdot m]$</td>
</tr>
<tr>
<td>$T_f$</td>
<td>Equivalent friction torque</td>
<td>$[N\cdot m]$</td>
</tr>
<tr>
<td>$T_s$</td>
<td>Equivalent spring torque</td>
<td>$[N\cdot m]$</td>
</tr>
<tr>
<td>$T_l$</td>
<td>Torque output from the gearbox</td>
<td>$[N\cdot m]$</td>
</tr>
<tr>
<td>$T_{pt}$</td>
<td>Torque caused by the pre-tension of the limp-home spring</td>
<td>$[N\cdot m]$</td>
</tr>
<tr>
<td>$J_{eq}$</td>
<td>Equivalent inertia of the ETB system</td>
<td>$[kg\cdot m^2]$</td>
</tr>
<tr>
<td>$\omega_{etb}$</td>
<td>Angular velocity of the ETB valve plate</td>
<td>$[rad/s]$</td>
</tr>
<tr>
<td>$\theta_{LH}$</td>
<td>Angular limp-home position</td>
<td>[$^\circ$]</td>
</tr>
<tr>
<td>$\theta_{etb}$</td>
<td>Angular position of the ETB valve plate</td>
<td>[$^\circ$]</td>
</tr>
<tr>
<td>$\theta_{etb_max}$</td>
<td>Maximum angular opening position of the ETB valve plate</td>
<td>[$^\circ$]</td>
</tr>
<tr>
<td>$k_s$</td>
<td>Spring gain</td>
<td>$[N\cdot m/^\circ]$</td>
</tr>
<tr>
<td>$C_d$</td>
<td>Discharge coefficient</td>
<td>[-]</td>
</tr>
<tr>
<td>$\rho_{air}$</td>
<td>Density of air at 20 $^\circ$C</td>
<td>$[kg/m^3]$</td>
</tr>
<tr>
<td>$A_{plate}$</td>
<td>Cross-sectional area of the orifice plate</td>
<td>$[m^2]$</td>
</tr>
<tr>
<td>$A_{upstream}$</td>
<td>Cross-sectional area upstream of the orifice plate</td>
<td>$[m^2]$</td>
</tr>
</tbody>
</table>
CHAPTER 1. INTRODUCTION

The optimization and control of the internal combustion engine systems are realized through either the model-based or the data-driven algorithms. In this chapter, some of the popular model-based and the data-driven algorithms are reviewed. The advantages, as well as the limitations of the referred model-based and the data-driven algorithms are presented. To meet the growing demands of performance improvements for the internal combustion engine systems, a new category of the optimization/control algorithm is proposed, i.e. the model-guided data-driven algorithm. The significance of the new algorithm is also addressed. Finally, the outline of the dissertation is presented.

1.1 Model-Based Optimization and Control

With the addition of sensors and electronic actuators, the performance improvements of the modern Internal Combustion, IC, engine systems have become optimal multi-objective problems [1-4]. The interconnected dynamics within the IC engine systems require the precise adjustment and coordination of most accessible system inputs [1-4]. In general, the applied inputs for an IC engine system are bounded – either physically bounded, such as the control for an Electronic Throttle Body, ETB, is limited by the rated specifications of the embedded Direct Current, DC, motor [5-8], or bounded by the purposes, such as the operating ranges for fuel injection timings in a diesel engine that are confined for achieving the desired engine performances [7][9-12].

For most optimization and control problems related to IC engines, one of the many optimization ‘objectives’ is chosen as the target while the rest of the objectives become the constraints for the optimization [4]. In the past, suitable model-based algorithms have been attempted to solve such problems. Zhao et al. have presented a mean-value model that is linearized at multiple operating points of a diesel engine [10]. The model is then applied towards diesel engine air-path management using the Model Predictive Control, MPC. In the study presented by Bengtsson et al. [13], the steady-state nonlinear relationship between multiple control inputs and the combustion phase of a diesel engine is estimated by a fourth-order polynomial. The inverse of this nonlinear model is introduced to compensate for the system nonlinearities. The MPC algorithm utilizes the
identified model to minimize the fuel consumption and the exhaust emission of the diesel engine. Isermann et al. have proposed a model-based calibration approach for the optimization of exhaust emission from a diesel engine using a mean-value engine model [14].

For the model-based optimization and control approaches, the controller is designed based on the model parameters with the assumption that the model output can predict the real system behavior accurately [15]. However, in reality, the modeling process is the development of mathematical expressions that approximate the system characteristics. In the approximations, some errors are inevitable - unmodeled system dynamics always exist within a model. As a result, the closed-loop control system using the model-based algorithms can be less robust and sometimes unsafe [13][14][16][17].

To keep the advantages of the model-based approaches and, at the same time, improve the robustness against the existed model uncertainties, robust control strategies have been developed. For the robust control strategies, the parameters of a system model are applied for controller design while the output from the controller is conservative in order to preserve the robustness of the closed-loop system [18][19]. As a result, the transient performances of the system are, in general, sacrificed [18]. Moreover, because a clear and practical description of ‘model uncertainty’ is currently not available, the design and deployment of robust control algorithms in real applications are hurdled [15][19].

Besides the development of control strategies, significant efforts have been made regarding model accuracy improvement for the emulation of different processes within IC engine systems. The Computational Fluid Dynamic, CFD, models are the ones that have the capacity to provide detailed spatially-resolved and temporally-resolved internal fluid field information for an IC engine system [20]. The CFD models have been utilized to simulate different air motions and chemical kinetics within Spark Ignition, SI, engines [21][22]. The air motion, combustion, and emission generation from Compression Ignition, CI, engines have also been investigated using the CFD models [23][24]. Though the CFD models can provide high fidelity predictions of the real system responses, it is impractical to use the CFD models for control-oriented applications because of the extensive computations required for the model operation [8][25].
On the other hand, control-oriented models are computationally efficient models feasible for real-time applications [26]. Control-oriented models have been broadly classified into three categories [8][26][27]:

1. **First-Principle Models** or FPMs yield physical insights of the real systems;
2. **Black-Box Models** or BBMs rely on experimental data alone and do not require any prior knowledge of the physical principles of a system;
3. **Grey-Box Models** or GBMs are partially based on first-principle together with the inclusion of a few tuning parameters. Experimental results are required for the identification of these parameters.

The FPMs usually contain extensive details of the real system. However, the development of such models are time-consuming and require a thorough understanding of the system mechanisms [3][8][26][28]. Moreover, to tune the model parameters for an FPM hence its output can match the measured system response usually leads to aggregated model parameters [26]. The numerical values of the aggregated model parameters usually include the effect of unconsidered and distributed dynamics of the real system [26]. As a result, the parameters are usually not accurate [26]. From the above discussion, it is evident that the construction of a valid control-oriented FPM is challenging.

Compared to the FPMs, the BBMs and the GBMs usually have simple structures but heavily rely on experimental results for model parameter identification. The system is excited within its operating ranges by specially designed signals, as the system responses are recorded [27][29]. By applying different model identification techniques, the model parameters are determined, thus the model output reproduces the system behavior [27][29]. However, for nonlinear systems, the identified models usually demonstrate satisfactory accuracy only within the vicinity of the operating point where the model parameters are identified [26][27][29]. For an IC engine, this is usually around the calibrated operating points in the steady-state Look-Up Tables, LUTs [30]. The identified BBM or GBM models also have limited robustness to disturbances. If the disturbances have affected the model output to depart significantly from the real measurement [26], recalibration is required for the BBM or GBM models.
From the discussion above, it is apparent that the integrity of the system model is important to the quality of the model-based optimization or control applications. Unfortunately, the construction of an accurate and computationally efficient system model within bounded operating ranges, if not impossible, is very challenging [8]. Therefore, the model-free optimization and control algorithms have emerged.

### 1.2 Data-Driven Optimization and Control

The progress achieved in data-acquisition technologies have made it feasible and straightforward for the sampling of a massive amount of measurement data from a target system [31][32]. The acquired data can be either collected online during the system operation or stored for post analysis [31][33]. With the availability of massive amount of data in real-time, the development of data-driven algorithms for the optimization and control of real systems has become practical.

The data-driven control and optimization algorithms are built directly on sets of collected input-output system data [31-34]. Compared to model-based control designs, the data-driven methods have one distinct feature: the fixed structure of the data-driven controller is neither dependent on the system model nor to any previous knowledge of the system [34]. As a result, the twin-born problem between the unmodeled system dynamics and the robustness of model-based control does not exist for data-driven algorithms [15][33][34].

For the data-driven optimization and control algorithms, the numerical values of the controller parameters are tuned based on large batches of measurements obtained from the system [31-34]. This is also different from the adaptive control in which the controller parameters are usually determined by a few samples or even from a single pair of input-output data [34]. Most of the data-driven algorithms are built on the concept of optimization for different performance criteria where system performance is measured by the 2-norm of certain sampled signals [34-39]. These optimization problems are later solved either through the ‘one-shot’ approach [40-42], i.e. with a single batch of data sampled at a single operating point [34], or through the iterative computations [43][44].

The Virtual Reference Feedback Tuning, VRFT, is a representative of the ‘one-shot’ data-driven control algorithms [40][41]. The VRFT is usually applied to determine
controller parameters for the Linear Time-Invariant, LTI, systems [33][34]. By introducing a virtual reference signal, the parameters of the optimum controller can be derived [33][34][40][41]. Though the VRFT may avoid the construction of a system model, it suffers from a few drawbacks [15]: (1) the parameters of the controller are tuned offline, which limits the robustness of the controller; (2) the insufficient data-set requirement may cause miss interpretation of the system dynamics.

The data-driven approaches using iterative computations are recursions, and are capable of generating successive approximations towards the global optimum of the designed performance function [15][34]. Compared to the ‘one-shot’ methods, the iterative data-driven algorithms are the more popular strategies that have been applied to a number of engineering applications related to IC engine systems. Zhang et al. [45] and Rezaeizadeh et al. [46] have applied the iterative data-driven algorithms to the optimization of spark timing for SI engine systems. Min et al. [47] have developed an iterative data-driven algorithm to identify the best actuation profiles for both an Exhaust Gas Recirculation, EGR, valve and a set of nozzle vanes of a Variable Geometry Turbocharger, VGT, in a diesel engine system. Ishizuka et al. [48] have implemented an iterative data-driven algorithm to tune the parameters for a NOx, oxides of nitrogen – NO and NO2, model thus to improve the model predictions of NOx quantity from a diesel engine system.

Because of the model-free nature, the data-driven algorithms cannot foresee the system performance using model prediction. As a result, more time and efforts are usually needed for the optimization to converge. The data-driven algorithms are limited to applications with relaxed requirements of system transient performances [15][34][49].

1.3 Scope of Work

In this dissertation, the author attempts to integrate the nonlinear system models with the data-driven algorithms as alternative solutions for the control and optimization of some nonlinear processes in IC engine systems. Necessary assumptions are provided prior to the introduction of each algorithm. Thereon, experimental verifications are given as proof of validation. The proposed solutions are termed the Model-Guided Data-Driven, MGDD, algorithms. Compared to the model-based algorithms, controller parameters of
the MGDD algorithms are not based on the model parameters, whereas the output from complicated nonlinear system models can be directly utilized by the MGDD algorithms. Compared to the data-driven algorithms, the transient performance of the system response can be improved by using available model output.

In this dissertation, the proposed MGDD algorithms have been applied to solve two problems:

(1) **The determination of optimum reference trajectories for IC engine systems;**
(2) **The design of controllers for nonlinear actuators of IC engine systems.**

Three examples related to the optimization of injection timing of a CI engine are first provided as demonstrations of the MGDD algorithms applied in the determination of optimum reference trajectories. The timing of fuel injection is chosen as the tuning parameter by the conventional data-driven optimization algorithm. The optimization is constrained by different performance functions. The MGDD algorithms are formed by integrating a first-principle engine model with the data-driven optimization and provide feedforward or feedback assistances to improve the transient performance of the data-driven optimization.

Then, the actuation of an Electronic Throttle Body, ETB, is presented as a demonstration of using the MGDD algorithm to control a non-smooth and nonlinear actuator. A data-driven Pseudo-Wiener model is proposed in this work to predict the ETB response in real-time. A nonlinear model predictive control structure with the online model update is then introduced to optimize the ETB control input and to achieve the desired system response confined by a performance function.

In the next section, the significance of the dissertation is presented.

**1.4 Dissertation Significance**

The main contribution of this work is the proposition of the MGDD algorithms together with the experimental validation. As shown in Figure 1-1, under the ideal scenario, if accurate, yet computationally efficient, system models are available, the model-based algorithms can be a possible solution for the optimization and control of such systems.
On the other hand, if models are unavailable, the data-driven algorithms become the only applicable solutions for system optimization and control.

Figure 1-1. Main contribution

For situations in which the systems can be described by certain mathematical models that are complicated in structure and suffering from plausible fidelity, the MGDD algorithms become the alternative solutions for the optimization and control of such systems. The models that are not suitable for model-based optimization/control designs, can be utilized to provide feedforward or feedback assistances for the data-driven optimization/control algorithms. As a result, the systems can achieve the desired performances with improved transient system responses, compared to the conventional data-driven algorithms.

1.5 Dissertation Outline

The outline of this dissertation is schematically presented in Figure 1-2. The motivation of the research is first introduced in the present chapter. The preliminary knowledge related to engine parameter optimization and ETB control is introduced in Chapter 2. Three popular data-driven strategies are briefly summarized in Chapter 3. Chapters 2 and 3 provide the motivation and the technologies required for the derivation of the MGDD algorithms. Three different examples on the optimization of injection timing for a CI engine using different MGDD algorithms are presented in Chapters 4, 5, and 6. Chapter 7 demonstrates the actuation of an ETB valve using the MGDD predictive control algorithm. Finally, concluding remarks and future perspectives are given in Chapter 8.
Chapter 1:
Introduction and research motivation

Chapter 2:
Engine parameter optimization and ETB control

Chapter 3:
Data-driven control and optimization

Chapter 4, Chapter 5, Chapter 6
MGDD for the optimization of injection timing of a CI engine

Chapter 7:
MGDD predictive control of an ETB actuation

Chapter 8:
Conclusion and future perspectives

Figure 1-2. Dissertation outline
CHAPTER 2. OPTIMIZATION OF INTERNAL COMBUSTION ENGINE SYSTEMS AND ELECTRONIC THROTTLE CONTROL

Preliminary introductions of the optimization of IC engine parameters and the control of the Electronic Throttle Body, ETB, are provided in this chapter. The challenges for the engine parameter optimization and the ETB control are also addressed.

2.1 Optimization of Internal Combustion Engine Systems

The calibration of an IC engine system is essentially a multi-objective optimization problem [1][2]. The control parameters of an IC engine system are tuned to yield the optimum system performance requirements. These performance requirements are often trade-offs among fuel efficiency, emissions, and reliability [14][50]. Electronic actuators, from simple versions like the ETB in the air-intake path to complex sub-systems with integral intelligence, have been added to the IC engine systems to assist with the achievement of the system performance requirements [51]. In traditional engine calibration processes, extensive time and efforts have been invested to generate Look-Up Tables, LUTs [52][53]. The LUTs are used to store the optimum parameters identified under different engine calibrations [52][53]. Profound knowledge of engine operation principles as well as advanced optimization methods is called in to complete the sophisticated IC engine calibration process [54]. In this chapter, the Design of Experiments, DOE, based optimization and the data-driven optimization are presented.

2.1.1 Design of Experiments Based Optimization

For an on-road vehicle, depending on different operating conditions, the onboard Engine Control Unit, ECU, assigns the optimal parameters or reference trajectories from the LUTs to different actuators based on different system performance requirements [54][55]. The DOE methods have been applied to determine the optimal control parameters on-line or off-line [55-58]. The identified parameters are then stored inside the LUTs at different engine operating conditions [55-58].

The creation of engine LUTs starts with the identification of the control variable limits [54][59-61]. For example, the ranges for fuel injection timing of a CI engine are limited
because of safety and engine performance concerns [7][9-12]. Usually, the minimal sets of experiment points required from engine dynamometer tests can be generated using different DOE methods. These methods include the full factorial design [62] and different optimal design approaches [59][60-62].

The obtained test data, which often include the engine torque, emissions, and fuel consumption measurement, are then applied for engine model identification and validation [55][59-61]. The engine models can be built using regression or neural network structures [59-61]. After a qualified model is obtained, the model-based optimization techniques are used to determine the optimum control parameters. These parameters can keep the engine model running under the desired performance requirements [59-61]. Finally, the optimum parameters are recorded on different LUTs.

From the above descriptions, it is obvious that the system model plays an important role in the DOE based optimization. In real practices, the DOE based optimizations are realized with the considerations of different constraints and budget limits [60]. Despite the differences, all DOE procedures do share a common workflow as summarized in Figure 2-1 [50][60][61]. For the completeness of this work, a brief description of this 7-step workflow is given below:

Step 1. **Plan:** In this step, the objectives of the engine calibration are first defined. Time, budget, and available technical support are taken into consideration for choosing the best experimental strategies. The viable operating boundaries and operational constraints for the control variables are decided. The necessary system responses are also identified to form a reasonable test plan.

Step 2. **Design:** Experimental setup and reasonable design methods, such as D-optimal design, are selected for the tests.

Step 3. **Test:** Experiments that are designed based on the preparations made in Step 1 and Step 2 are carried out on the engine dynamometer test bench. Test data are collected.

Step 4. **Modeling:** Analyses are required to determine the control parameters that impose significant effects on the targeted system responses. Suitable model structures, such
as different regression models, are chosen to build the relationships between the system control inputs and the system outputs. The model parameters are derived based on the experimental data collected in Step 3.

![Diagram of model-based DOE steps]

Figure 2-1. The model-based DOE steps

Step 5. **Model assessment**: The accuracy of the model is assessed. Experiment data other than the ones applied for model parameter identification are used for model validation.
If the model quality cannot meet the fidelity requirements, a redesign of the system model is necessary, i.e. Step 1 to Step 4 is repeated.

Step 6. **Prediction and optimization**: A target performance function is often defined for the model-based optimization using the identified engine model. The optimized control parameters are assessed to validate their robustness under various system operating conditions.

Step 7. **Mapping**: The obtained optimal parameters are mapped on the ECU LUTs. To close the gaps between the calibrated points and improve the transient response of the IC engine, certain smoothing techniques are required to generate the final LUTs.

The LUTs derived through the DOE based optimization have been very effective for IC engine operations. However, the efforts required to obtain the optimum parameters are quite time-consuming [63-65]. Since the workflow requires model-based optimization, the derivation of a reliable engine model is very important. The abundant choices of available model structures and the addition of control actuators for the operation of IC engines make the derivation of the IC engine models challenging [54][61][63-65].

### 2.1.2 Data-Driven Based Optimization

To reduce the effort in DOE based optimization, data-driven algorithms, such as the Extremum Seeking Control, have been attempted as alternatives for the calibration of IC engines. Data-driven algorithms are known for their quickness and directness in locating the optimal parameters [63-68]. The operation of the Extremum Seeking Control, ESC, relies solely on the relationship between the system input and output [63-65][70][71]. This makes the ESC a model-free optimization method. The tedious modeling work required by the DOE based optimization is omitted. In a number of previous studies, the calibrated parameters obtained by the ESC have shown similar or even better results compared to the calibrated parameters obtained through DOE based optimization in terms of different performance requirements [63][65]. Both the engine operation and the ESC optimization can also proceed concurrently in real-time. This simultaneous operation further reduces the strict design requirements proposed in the DOE based optimization, which opens up the possibilities for realizing the optimization of IC engines on-the-fly.
A few examples of the optimization of IC engines using the ESC are presented below:

Popovic et al. [63] have used ESC to optimize the valve timings and advanced spark timing of an SI engine to make sure the engine operates at the best fuel efficiency. Killingsworth et al. [64] optimized the combustion timing of a natural-gas engine using the ESC by changing the temperature of the intake charge. The engine achieves the best fuel efficiency at the selected operating point after the ESC optimization. In the study carried out by Hellstrom et al. [65], the spark timing for an SI engine is optimized online using the ESC to operate at its best thermal efficiency. Different ethanol/gasoline fuel blends, as well as different engine speed/load conditions, are altered during the optimization. Corti et al. [66] have applied ESC to optimize the air-fuel-ratio and the spark timing for an SI engine. A higher Indicated Mean Effective Pressure, IMEP, under constant fueling rate is achieved by the SI engine. Knocking intensity and the temperature at the exhaust manifold are set as the constraints for this optimization task. Haskara et al. [67] have optimized the spark timing of a gasoline engine using ESC when the engine Exhaust Gas Recirculation, EGR, rate is increased without scarifying the combustion stability. In Larsson et al.’s work [68], the spark timing for reaching the highest possible torque output from a gasoline engine is located using the ESC when the EGR ratio of the engine is adjusted.

Although the ESC has shown advantages over the conventional DOE based optimization in the aforementioned examples [63-68], the convergence speed of the ESC is usually limited by the dynamic response of the system. This is caused by the requirement of time-scale separation for averaging with singular perturbation [39][69]. Therefore, the application of the ESC to systems with timely performance concerns would be unsuitable [8]. How to improve the convergence speed of the ESC is, therefore, an interesting research topic.

2.1.3 Model for a Compression Ignition Engine

In the automotive industry, various types of models for describing different processes within an IC engine have been developed [2][8]. These models, though complicated and
highly nonlinear, are usually validated via a large amount of experimental data to attain a certain level of accuracy. Even if it is practically challenging to conduct a direct model-based optimization/control design using some of these existing models, it is possible to extract the output from the models to assist the ESC by providing a reasonable starting point for the optimization. By using the feedforward information from the models, a faster convergence of the ESC optimization can be expected.

Many available IC engine models are classified as Mean Value Models, MVMs [8]. MVMs are mostly control-oriented models represented by nonlinear differential equations in the continuous time-domain [2][8]. The operation of an IC engine differs from a continuously operating thermal engine mainly in two aspects [2][7][8]: (1) the highly transient combustion process; (2) the varying thermodynamic boundary conditions that affect the combustion process. Most existing MVMs focus on the modeling of the thermodynamic boundary conditions. Therefore, the engine combustion is treated as a static effect governed by the different boundary conditions [4][8]. The model parameters of most MVMs are lumped parameters [4][8]. As a result, the output values from the MVMs are considered as averaged effects over multiple IC engine cycles [4][8]. So far, MVMs have been used to model the performance of the IC engines in a number of applications. These include the load estimation of IC engines [72][73], the prediction of air-path dynamics of a turbo-charged IC engine [74], and the approximation of NOx emission of an IC engine [14].

Apart from MVMs, Discrete Event Models, DEMs, are also used to predict IC engine performances under different operating conditions [8][75]. DEMs consider the system operation as a sequence of discrete events and the approaching of a different event marks the change of system states [8][75]. The system is assumed to remain unchanged during two consecutive events [8]. As a result, simulations of DEMs are simply calculated at each event while the processes between the events are neglected [8]. This is the main difference compared to MVMs as continuous time has been broken into small-time sections, hence the states of the DEMs are updated in accordance with the activities that happened within each time section.
For IC engine models, the DEMs are defined in the crank-angle domain, i.e. all the control input and the system states become functions of the engine crank-angle $\varphi$. This is extremely convenient for modeling the performance of IC engines because the reciprocating nature of the IC engine operation makes its analysis convenient as a discrete event process [8].

In this work, a DEM engine model has been selected to support the development of different MGDD algorithms for the optimization of fuel injection timing in a CI engine constrained by different performance requirements. The model is developed based on the first law of thermodynamics applied to the trapped gases in the combustion chamber [28]. Empirically developed sub-models for the heat transfer and combustion processes are incorporated in this model. The details for the DEM model can be found in the work titled ‘Clean combustion control in a compression ignition engine’ [28]. For the completeness of this work, the model is briefly presented in this subsection.

In the model, the crank-angle $\varphi$ has a resolution of $1^\circ$ crank angle. $p_n(\varphi)$ represents the in-cylinder pressure at $\varphi$ within the $n^{th}$ engine cycle. $\theta_n$ is the control input for the model. In the model, $\theta_n$ is the crank angle for the Start of Combustion, SOC, at the $n^{th}$ engine cycle. $u_n$ is the global input for the CI engine model. In this work, $u_n$ comprises of the engine speed, $\omega_{RPM}$, and the cyclic delivered fuel mass, $m_{fuel}$. $V(\varphi)$ is the instantaneous cylinder volume at crank angle $\varphi$.

**Cylinder Volume**

$V(\varphi)$ is expressed as

$$V(\varphi) = V_c + \frac{V_s}{2} \left[ 1 + \frac{l}{a} - \cos(\varphi) - \left( \frac{l}{a} \right)^2 - \frac{\sin(\varphi)}{a} \right]^{1/2}$$

(2-1)

where $l$ is the connecting rod length, $a$ is the crank radius, $V_s$ is the swept volume, and $V_c$ is the clearance volume. The crank radius is half the length of the engine stroke, $s$. 
**In-Cylinder Bulk Gas Temperature**

It is assumed that between $\varphi$ and $\varphi + 1$, the in-cylinder process is a quasi-steady state process. When all gas exchange valves are closed, $T_n$ is the in-cylinder bulk gas temperature, given by:

$$
T_n(p_n, \varphi) = \left[ \frac{p_n(\varphi) \cdot V(\varphi)}{\bar{R} \cdot N_{im}} \right]
$$

(2-2)

where, $\bar{R}$ is the universal gas constant and $N_{im}$ is the total moles of in-cylinder gas when all the gas-exchange valves are closed.

The total moles of the trapped gas are assumed to stay constant during the in-cylinder process. The $N_{im}$ is defined by:

$$
N_{im} = \eta_V \left[ \frac{p_{in} \cdot V_s}{\bar{R} \cdot T_{in}} \right]
$$

(2-3)

where $p_{in}$ is the intake manifold pressure, $T_{in}$ is the intake temperature, and $\eta_V$ is the volumetric efficiency.

**The Rate of Wall Heat Transfer**

For the engine test setup in the author’s lab, the rate of wall heat transfer $\delta Q_{ht}$ is modeled using Woschni’s empirical correlation with the following settings [7]:

$$
\delta Q_{ht}(\varphi, u_n, \theta_n) = h_w(p_n, \varphi) \cdot A_s \cdot [T_n(p_n, \varphi) - T_{wall}]
$$

(2-4)

where

$$
h_w(p_n, \varphi) = C_1 \cdot B^{-0.2} \cdot p_n(\varphi)^{0.8} \cdot T_n(p_n, \varphi)^{-0.55} \cdot w(p_n, \varphi)^{0.8}
$$

(2-5)

and

$$
w(p_n, \varphi) = C_2 \cdot \bar{S}_p + \frac{V_s \cdot T_{ref}}{p_{ref} \cdot V_{ref}} [p_n(\varphi) - p_{mot}(\varphi)]
$$

(2-6)
$A_s$ is the surface area of the combustion chamber, $B$ is the cylinder bore, $\bar{S_p}$ is the mean piston speed, and $p_{mot}$ is the cylinder pressure obtained during engine motoring, i.e. engine rotation with the help of a motor but without the presence of combustion. $T_{ref}$, $p_{ref}$, and $V_{ref}$ are the reference states used for calculating the heat transfer following the combustion event. The wall heat transfer coefficients, $C_1$ and $C_2$, as well as the combustion chamber wall temperature, $T_{wall}$, are calibrated based on the engine geometry. Once calibrated, the coefficients and the wall temperature are kept constant throughout the simulation study. The values of the selected model parameters are later presented in Table 4-2.

**The Rate of Heat Release**

A triangular function is used to approximate the rate of heat release, $\delta Q_{hr}$:

$$\delta Q_{hr}(\varphi, u_n, \theta_n) = \begin{cases} 0 & i \\ (2 \cdot Q_{hrMAX})(\varphi - \varphi_n) & ii \\ (-2 \cdot Q_{hrMAX})(\varphi - \varphi_{CA50}) + dQ_{hrMAX} & iii \\ 0 & iv \end{cases}$$

where,

$$\begin{align*}
&i: \varphi < \varphi_n \\
&ii: \varphi_n < \varphi < \varphi_{CA50} \\
&iii: \varphi_{CA50} < \varphi < [\theta_n + \varphi_{CD}] \\
&iv: \varphi > [\theta_n + \varphi_{CD}] \\
\end{align*}$$

$$Q_{hrMAX} = \frac{m_{fuel} \cdot LHV}{\varphi_{CD}}$$

$\varphi_{CA50}$ is the crank angle of 50% accumulated heat release. $\varphi_{CA50}$ is approximated by:

$$\varphi_{CA50} = \theta_n + \frac{\varphi_{CD}}{2}$$

$\varphi_{CD}$ is the combustion duration. In this work, $\varphi_{CD}$ is assumed as a constant. $LHV$ is the lower heating value of the fuel.
**Engine Work Output**

\[ \delta W_n \] is the work output from the CI engine. \( \delta W_n \) is derived by:

\[
\delta W(p_n, \varphi) = p_n(\varphi) \cdot [V(\varphi) - V(\varphi - 1)]
\] (2-11)

**In-Cylinder Pressure**

It is assumed that the trapped gas is ideal gas, and between \( \varphi \) and \( \varphi + 1 \), the in-cylinder process is a quasi-steady isochoric process. Thus, the change of the in-cylinder bulk gas temperature can be expressed as:

\[
T_n(\varphi + 1) = T_n(\varphi) + \alpha_n \cdot [\delta Q_n(\varphi, u_n, \theta_n) - \delta W_n(\varphi, p_n)]
\] (2-12)

where \( \alpha_n \) is related to the ratio of specific heats, \( \gamma_n \), of the in-cylinder gas mixture and is given by:

\[
\alpha_n = \left[ \frac{\gamma_n - 1}{R \cdot N_{im}} \right]
\] (2-13)

The ratio of specific heats \( \gamma_n \) usually changes with the temperature \( T_n \). To simplify the model computation, in this work, \( \gamma_n \) is approximated as a constant value:

\[
\gamma_n = \gamma_{\varphi_{IVC}}
\] (2-14)

where \( \gamma_{\varphi_{IVC}} \) is the ratio of specific heats of the gas mixture obtained at the crank angle of Intake Valve Closing, IVC. Based on the previous investigation, for the engine test setup in the author’s lab, \( \gamma_{\varphi_{IVC}} = 1.37 \) [76].

\( \delta Q_n(\varphi, u_n, \theta_n) \) is defined as the difference between \( \delta Q_{hr}(\varphi, u_n, \theta_n) \) and \( \delta Q_{ht}(\varphi, u_n, \theta_n) \), i.e. \( \delta Q_n(\varphi, u_n, \theta_n) = \delta Q_{hr}(\varphi, u_n, \theta_n) - \delta Q_{ht}(\varphi, u_n, \theta_n) \).

Therefore, the in-cylinder pressure can be derived as:

\[
p_n(\varphi + 1) = \left[ \frac{T_n(\varphi + 1) \cdot \hat{R} \cdot N_{im}}{V(\varphi)} \right]
\] (2-15)
In this work, the DEM model, described by equations from (2-1) to (2-15), is integrated into different MGDD optimization applications presented in CHAPTER 4 - 6.

2.2 Electronic Throttle Control

The improvement of control strategies for IC engine systems has become a driving factor in automotive innovation and development [4][8]. To meet different performance requirements, all corresponding electrical actuators on an IC engine need to work collectively and accurately track the optimum parameters determined by the calibrated LUTs [4][50][60][61].

The ETB system is a suitable example, in the intake air-path of an SI engine, the ETB is usually installed at the upstream of the intake manifold. The actuation of the ETB opening can affect the quantity of air-flow into the engine cylinder [5][7][77-79]. The quality of the in-cylinder air-fuel mixture is controlled by the ETB actuation. As a result, the engine torque is regulated [80-82].

As shown in Figure 2-2 [80-82], when the accelerator pedal is pressed, a change of the pedal position is detected. The two inputs, i.e. the pedal position and the measured engine speed, are sent to the Accelerator Pedal LUT, where the value for the corresponding engine torque is fitted [80][81]. Together with the measured engine speed, the engine torque is applied as an input for the Mass Airflow LUT to find the desired mass air flow in the intake manifold for the SI engine [80][82]. Through the calculation of a series of mathematical models, the ETB opening can be determined [80-82]. A local controller is presented to make sure the ETB can track the opening position reference derived from the mathematical models.
Fast and accurate ETB actuation can be achieved if the local controller is properly designed. As a result, the SI engine can respond to different torque demands immediately and fulfill the performance requirements [7][8][83].

A schematic of the ETB system is presented in Figure 2-3. The opening of the ETB butterfly valve is governed by a Direct Current, DC, motor through a set of gears. Limp-home spring is also mounted in line with the axle of the butterfly valve. The spring exercises a torque on the throttle plate to pull it back to the default limp-home position. The limp-home position is a fail-safe position of the ETB when motor actuation is not applied. By design, if the ETB system is out of order, the ETB is mechanically kept open with a small angle that allows the driver to ‘limp’ the vehicle to a safe place [77-79]. A pair of redundant position sensors is also embedded in the ETB system. The sensors provide voltage signals that are proportional to the opening of the ETB valve.
2.2.1 Existing Challenges

The control of the ETB opening requires the consideration of the presence of multiple nonlinearities. These nonlinear characteristics are the limp-home spring nonlinearity, the stick-slip friction, and the gear backlash [77-79][84]. The nonlinearities are more noticeable when the ETB is near the fully closed position [85].

Figure 2-4. ETB valve opening affected by the flow of air

Even though the ETB is used to regulate air flow in an SI engine [86], the presence of air movement can disturb the ETB opening. As shown in Figure 2-4, the ETB opening would first increase because of the control input change that is highlighted by the red arrows in the figure. As the test proceeds, the value of control input remains constant and the air flow alone changes the opening of ETB position as pointed out by the black arrow in the...
To improve the system robustness against air flow disturbances, closed-loop control is required.

The ETB system also demonstrates rate–dependent characteristics as shown in Figure 2-5. From the figure, it is obvious that the ETB would respond differently when the frequencies of control inputs are changed. Therefore, the closed-loop controller designed for ETB actuation needs to be adaptive, so the system can perform satisfactorily under dynamic or steady-state conditions alike.

Different solutions have been proposed to achieve the prompt and accurate control of the ETB system.

![Figure 2-5. Rate-dependent characteristics of an ETB system](image)

### 2.2.2 Existing Solutions

Previously, different closed-loop control strategies have been attempted for ETB control [77][88-90]. In general, the methods can be characterized into two categories: Using either the data-driven control strategies or the model-based control strategies.

The Proportional-Integral-Derivative, PID, control is the most popular strategy exercised for the majority of automotive control applications [80]. The presence of a system model is not necessary for the design of PID gains, thus under certain scenarios, the PID control can be considered as a type of data-driven control [15]. Multiple PID gains are usually required for the PID control of ETB [87]. A few examples on the PID controlled ETB are briefly summarized as follows:
Deur et al. [77] have used model-free friction and limp-home compensators to counter the nonlinear behavior of the ETB system. The PID control is then applied to govern the ETB opening. The PID gains are calibrated without utilizing an ETB model. As stated in Eriksson et al.’s work [88], the information of modeled friction is used to counter the nonlinear torque in an ETB. The compensated ETB system then behaves close to a linear system. The corrected ETB system is controlled using a gain scheduling PID method where multiple control gains are calibrated based on different engine operating points. In the study conducted by Jiang et al. [89], a model-free parameter tuning method called iterative feedback tuning is introduced to determine the gains for a PID controller. For demonstration purposes, the obtained PID gains are applied for the ETB actuation. Multiple experiments are required before the desired control parameters can be determined. The gain scheduling is also used for the ETB position control. Corno et al. [90] have proposed a linear data-driven model. The model estimates the relationship between the rate of reference change and the PID gains. The parameters of the model are identified using test data. Based on the design of the reference signals, the PID gains for ETB control are adjusted in advance.

Many of the PID strategies proposed in the literature use the gain scheduling to adapt to the varying operation regime of the ETB. Some researchers have also introduced feedforward or feedback compensations to linearize the ETB system. However, time-consuming efforts are required for the calibration of the multiple sets of PID gains and considerable memory spaces are usually needed for storing the parameters [80][89]. Furthermore, the choices for the design of nonlinear compensators are also abundant [77][84][80][88]. The protocols on how to choose the appropriate compensators are not readily available [84][80][88].

The Sliding Mode Control, SMC, is a model-based robust control approach that has been utilized for the actuation of the ETB [80]. The SMC is first proposed to take care of unknown disturbances, uncertain model parameters, and parasitic system dynamics for nonlinear systems [91-93]. For the SMC control, when the system response reaches the sliding surface in finite time, the system dynamics would be irrelevant to the disturbances and modeling inaccuracies. However, the system dynamics are affected by the design of
the sliding variable [93][94]. This is a desirable property that illustrates the possibility of achieving the desired system performance by tuning the sliding variable [91-94]. Many attempts on the application of the SMC for ETB control have been carried out in the past. A few examples are briefly summarized below:

Pan et al. [95] have designed a continuous time SMC using the back-stepping approach to control the ETB opening. The armature current of the DC motor and the valve angular velocity are estimated using a designed SMC observer. Baric et al. [96] have estimated the unknown disturbances and uncertainties in the ETB system by a multilayer perceptron neural network model. The angular velocity of the butterfly valve is estimated by filtered differentiation of the measured valve position. Based on the output of the neural network model, the SMC controller can be defined. Beghi et al. [97] have proposed a discrete-time SMC controller with an integral function to manage the ETB operation. Instead of using a first-principle model, Beghi et al. have proposed to use a first-order state space MVM to estimate the ETB behavior. A sliding mode observer is used to estimate the state of the designed model. The integral action is introduced to minimize the steady-state error. Horn et al. [98] have also proposed the addition of integrator in the SMC for ETB control in the continuous time domain. Compared to the work conducted by Beghi et al. [97], a first-principle ETB model is used for the design of the controller. Both simulation and experiment test results are presented to demonstrate the effectiveness of the controller.

In most of the cited literature above, gear backlash hysteresis is considered as a model uncertainty. Unfortunately, the majority of the applied SMC controllers lacks a clear definition of how to choose the upper bounds for the unknown model uncertainties. The absent of this definition often makes the tuning of the controller parameters conservative [95-98]. Moreover, the chattering effect always exists within the responses of the feedback system. The chattering effect can be attenuated by smoothing but cannot be eliminated [95-98].

From the above discussion, it is natural to think that an accurate ETB model, if available, can predict the ETB response. As a result, it may be possible to realize lesser conservative controller designs.
CHAPTER 3. DATA-DRIVEN OPTIMIZATION AND CONTROL

Three popular data-driven algorithms that have been applied for the optimization and control of IC engine systems are briefly presented in this chapter. The three algorithms are the Simultaneous Perturbation Stochastic Approximation, SPSA, the Iterative Learning Control, ILC, and the Extremum Seeking Control, ESC.

3.1 Simultaneous Perturbation Stochastic Approximation

Most optimization problems, in the mathematical sense, may be regarded as the search for the minimum, or maximum, of some scalar-valued performance function with respect to a number of controllable parameters. To solve these problems, many existing optimization methods use the gradient of the performance function with respect to the parameters being optimized as guidance to find the optimum solutions [99][100].

Besides the gradient-based strategies, there is a growing trend of developing recursive optimization algorithms that do not directly rely on the gradient of the performance function [100]. Instead, the approximation of the gradient of the performance function derived from the measured system output is utilized for the design of the optimization algorithms [34][100]. One obvious advantage of such algorithms is that the knowledge of the exact relationships between the performance function and the parameters being optimized are not required [100]. This important information cannot be omitted for gradient-based optimization algorithms [99][101].

In gradient-based optimization algorithms, because of the presence of unavoidable measurement noise, further processing is often needed to yield estimations of the real gradient values of the performance function [100]. Since the direct measurement of the gradient of the performance function is usually not available [100], detailed knowledge of the system input-to-output relationship must be available [99][100]. Using these relationships and necessary measurements, the gradient of the performance function can be calculated [99][100]. As a comparison, the optimization algorithms that require approximation of the gradient of the performance function can simply convert the direct measurement of the system output to the values of performance functions
Therefore, the input-to-output relationship of the system is not needed.

The SPSA is one of the optimization algorithms, which is based on the approximation of the gradient of the performance function by utilizing direct measurement of the system output. Details of this algorithm have been summarized by J. C. Spall [100]-[103-105]. The sufficient conditions that guarantee the convergence of the SPSA can be found in Spall’s published papers [103-105]. For the completeness of this work, the SPSA algorithm is briefly described below:

The problem of minimizing a scalar differentiable performance function \( f(\theta) \) is considered in this brief introduction. The control input \( \theta \) is an \( n \)-dimensional vector and the optimum control input is defined as \( \theta^* \). The optimization problem can be translated into the search for \( \theta^* \), which is the solution for \( \frac{\partial f}{\partial \theta_i} = 0 \), where \( i = 1, 2 \ldots n \).

In this introduction, the direct measurement of \( f(\theta) \) is assumed to be available while the direct measurement of \( \frac{\partial f}{\partial \theta_i} \) is not available.

The input \( \theta \) at the \( k \)th iteration of the optimization is expressed as:

\[
\theta_k = \theta_{k-1} - a_k \cdot \hat{g}_k(\theta_{k-1}) \tag{3-1}
\]

In (3-1), \( a_k \) is the updating gain and \( \hat{g}_k \) is the approximation of the gradient of the performance function at \( \theta_{k-1} \). The control input \( \theta_k \) is initialized to \( \theta_0 \) at the first iteration. If the sufficient conditions for the convergence of the SPSA are satisfied [103-105], \( \theta_k \) would converge to \( \theta^* \) when \( k \to +\infty \).

Let \( y(\theta_{k-1}) \) represents the system response to the input \( \theta_{k-1} \) at the \( k \)th iteration. \( c_k \) is a small positive number at the \( k \)th iteration. For the SPSA, all \( n \) elements in \( \theta_{k-1} \) are randomly perturbed simultaneously to obtain two measurements of \( y(\theta_{k-1}) \). The \( i \)th element in \( \hat{g}_k \) can thus be derived by:

\[
\hat{g}_{k,i}(\theta_{k-1}) = \frac{y(\theta_{k-1} + c_k \cdot \Delta_k) - y(\theta_{k-1} - c_k \cdot \Delta_k)}{2c_k \cdot \Delta_{k,i}} \tag{3-2}
\]
To guarantee the convergence of (3-1), values of $a_k$ and $c_k$ should reduce moderately to zero as the optimization proceeds [100][103-105]. Guidance on how to select the values for $a_k$ and $c_k$ have been published [100].

The elements in the $n$-dimensional random perturbation vector $\Delta_k = [\Delta_{k,1}, \Delta_{k,2} \ldots \Delta_{k,n}]^T$, are independent and symmetrically distributed around zero [100][103-105]. The superscript ‘$T$’ represents the transpose of the matrix.

The symmetric Bernoulli ±1 distribution has been widely used as the perturbation vector $\Delta_k$ in the SPSA [106]. The symmetric Bernoulli ±1 distribution has two possible outcomes as $j = 1$ and $j = -1$, where $j = 1$ occurs with the probability of $p_b$ and $j = -1$ occurs with the probability of $1 - p_b$, where $p_b \in (0,1)$. If the elements in $\Delta_k$, (3-2), follow the symmetric Bernoulli ±1 distribution, the probability for each outcome, i.e $j = 1$ and $j = -1$, should be set to 0.5, i.e. $p_b = 0.5$.

A few examples using the SPSA for the optimization of IC engines are briefly presented below:

Ishizuka et al. [107] have tuned the PID gains for the air-path control of a CI engine system adaptively using the SPSA algorithm under lab conditions. During the engine test, it takes around 600 seconds before the optimum control parameters can be identified by the SPSA method. Ishizuka et al. conclude that the optimization method can potentially reduce the effort required for DOE based optimization. Ishizuka et al. [108] have applied the SPSA method for calibrating the parameters of a NOx model. The NOx measurement collected from an actual NOx sensor on a CI engine is used for the calibration. A prominent reduction of differences between the model output and the sensor measurement is observed after 900 seconds of optimization. Popovic et al. [109] have adjusted the timings for the intake valve opening and exhaust valve closing using the SPSA in an SI engine simulation. In this work, the authors have achieved improved engine fuel economy on the addressed SI engine model. It takes around 400 seconds to complete the optimization.
From the above examples, it is obvious that the convergence rate for the SPSA algorithm is slow and may not be suitable for the optimization of systems with parameters that vary quickly over time [15].

3.2 Iterative Learning Control

A human being is often capable of learning from previous mistakes to improve his/her skill from repetitive training. For a system that executes the same task repetitively, is it possible to introduce a similar mechanism that utilizes previous knowledge to improve the system response over time? The ILC can be one answer.

The ILC is a feedforward optimization algorithm that refines the control input to a system that operates iteratively subjected to repeating disturbances [110][111]. The ILC generates its open-loop control through practice and collects tracking errors from previous iterations [110][111]. The control method is anticipatory, thus it compensates for the repeating exogenous disturbances in advance [110-112].

The ILC has shown advantages in certain scenarios. In general, the following postulates, which have been summarized in [112], need to be fulfilled by the target system before the application of the ILC:

- Every iteration ends in a fixed time duration;
- The initial state of the system is set to the same point at the beginning of each iteration;
- The system dynamics are time-invariant and deterministic;
- The system output is measured deterministically.

Details regarding the stability, performance, and robustness of the ILC algorithm can be found in [110-113]. For the completeness of this work, the ILC algorithm for a discrete-time, LTI, Single-Input-Single-Output, SISO, system is briefly described here:

Under discrete time, an LTI, SISO system in the following form is considered:

\[ y_j(k) = P(z) \cdot \theta_j(k) + d(k) \]  

(3-3)
In (3-3), $k$ is the index of time; $j$ is the index of iteration; $z$ is the forward time-shift operator, i.e. $z \cdot x(k) \equiv x(k+1)$; $y_j$ is the system output; $\theta_j$ is the control input; and $d$ is the exogenous disturbance that repeatedly applies to the system at each iteration.

In (3-3), it is assumed that the system $P(z)$ is a proper rational function of $z. P(z)$ has a relative degree of $m$ and is asymptotically stable. For an $N$-sample sequence of inputs,

$$\theta_j(k), \quad k \in \{0,1,...,N-1\}$$

the outputs and the exogenous disturbance of (3-3) can be represented as:

$$y_j(k), \quad k \in \{m,m+1,...,m+N-1\}$$

and

$$d(k), \quad k \in \{m,m+1,...,m+N-1\}$$

It is assumed the desired system output, i.e. reference trajectory, is represented by:

$$R(k), \quad k \in \{m,m+1,...,m+N-1\}$$

the error signal can then be defined by $e_j(k) = R(k) - y_j(k)$.

To improve the response of the system (3-3), i.e. $y_j(k) \rightarrow R(k)$, the ILC algorithm is expressed as:

$$\theta_{j+1}(k) = Q(z)[\theta_j(k) + L(z)e_j(k+1)]$$

(3-8)

$Q(z)$ and $L(z)$ are defined as the Q-filter and the learning function, respectively [111].

Details on how to design $Q(z)$ and $L(z)$ can be found in [110-113].

The expression for the ILC algorithm in (3-8) is non-causal, i.e. $e_j(k+1)$ appears in the derivation of $\theta_{j+1}(k)$. This is an important feature for the ILC [110-113]. Since $e_j(k+1)$ is the error information sampled in the past iteration, the ILC makes such non-causal process possible for the derivation of the control input for the next iteration.

The ILC is not the ideal solution for all scenarios. The presence of non-repeating noise and disturbances would hinder the use of the ILC [111]. In this dissertation, the author has not applied the ILC for the construction of any MGDD algorithm.
In the past, the ILC has been applied to several studies related to IC engine operations. A few examples are presented below:

Min et al. [47] have applied the ILC to provide feedforward control for both the EGR valve and the vane position of a VGT system. By repeatedly training the actuators, e.g. through 11 iterations with each iteration lasting for 50 seconds, the Mass Air Flow, MAF, and the intake pressure of the CI engine can follow each corresponding reference trajectory under the selected acceleration condition. Slepicka et al. [114] have adjusted the injection quantity of iso-octane and n-heptane using the ILC algorithm, thus the IMEP and $\varphi_{CA50}$ of a research engine can track the predefined step trajectories. Slepicka et al. have also stated that at least 3 iterations of learning are applied in this work with each iteration lasting for 60 engine cycles. Heinzen et al. [115] have applied the ILC to control a camless engine valve actuation system for an SI engine. By adjusting the actuation of the on-off valve, the camless system can attain precise cycle-to-cycle tracking of different intake valve profiles. This in turn provides the benefit of achieving flexible power, efficiency, and advanced combustion control for the SI engine system

3.3 Extremum Seeking Control

Essentially, the ESC is a robust optimization method that uses only input-output data flow to achieve the optimal performance as represented by an extremum value of the performance function [39]. Unlike the optimal control strategy, the majority of existing ESC algorithms use time-scale separation techniques to obtain an input-to-steady-state-output mapping [39][69]. The mapping is used for the optimal solution determination for a dynamic system without the help of a system model [39][69]. For example, in the conventional perturbation-based ESC, a dither signal with a relatively low frequency is used to achieve the time-scale separation between the plant dynamics and the ESC optimization [39]. When the optimization assigns a new input, the ESC waits out the dynamics of the system before the measurement of the output is assessed [39][65][69][116][117]. By using the singular perturbation technique and averaging, the perturbation-based ESC can approximate the gradient of the designed performance function [69][116].
The perturbation-based ESC can be characterized into two types:

Type 1. Using a continuous dither signal to probe the steady-state mapping, from which the approximation of the inherent gradient of the performance function is obtained [39];

Type 2. Using ideas of nonlinear programming to exploit the probing dither excitations [117].

Different dither signals have been attempted for the perturbation-based ESC. These include the use of periodic signals [118] such as sinusoidal dither, square wave dither, triangular dither, and stochastic excitations [119]. An example of the Type 1 sine perturbation-based ESC is shown in Figure 3-1 [39][70][116].

![Figure 3-1. Structure for perturbation-based ESC](image)

In Figure 3-1, $z$ is the forward time shift operator; $g_r$ is the adaptation gain; and $A$ is the amplitude of the sine perturbation. The stability and convergence of the ESC are influenced by the choice of $g_r$, $A$, $\omega$ and the shape of the performance function $J(\theta(k))$ close to the extremum [39][116]. The high pass filter, $(z - 1)/(z + h)$, should have a cut-off frequency below the frequency of the perturbation signal $\omega$ [39][64].
Based on the structure shown in Figure 3-1, the measured performance function $J$ is used to derive the control input $u$. A more detailed introduction of the perturbation-based ESC can be found in the book ‘Real-time Optimization by Extremum Seeking Control’ [39]. For the completeness of this work, an intuitive argument that explains the convergence of perturbation-based ESC, Figure 3-1, is provided below [39][120]:

In this explanation, only the single-parameter ESC is considered, i.e. both $\theta(k)$ and $\tilde{\theta}(k)$ are scalars. Furthermore, only a single perturbation signal $A \cdot \cos(\omega \cdot k)$ is applied.

It is assumed that the performance function $J(\theta(k))$ is quadratic and can be expressed as:

$$J(\theta(k)) = f_1 + \frac{f_2}{2}(\theta(k) - \theta^*)^2$$ (3-9)

In (3-9), $f_1$ and $f_2$ are two constants that determine the shape of $J(\theta(k))$; $f_2$ is a positive number; and $\theta^*$ is the optimum control input that leads (3-9) to its minimum.

The variable $\tilde{\theta}(k)$ is defined as:

$$\tilde{\theta}(k) = \theta^* - \bar{\theta}(k)$$ (3-10)

According to Figure 3-1,

$$\theta(k) = \tilde{\theta}(k) + A \cdot \cos(\omega \cdot k)$$ (3-11)

Therefore,

$$\theta(k) - \theta^* = A \cdot \cos(\omega \cdot k) - \bar{\theta}(k)$$ (3-12)

Use the expression of $\theta(k)$ in (3-12) and expand it in (3-9), thus

$$J(\theta(k)) = f_1 + \frac{f_2}{2} \left( A \cdot \cos(\omega \cdot k) - \bar{\theta}(k) \right)^2$$ (3-13)

By expanding (3-13) and applying the trigonometric identity: $2\cos^2(\omega \cdot k) = 1 + \cos(2\omega \cdot k)$. Therefore, (3-13) becomes
\[ J(\theta(k)) = f_1 + \frac{f_2}{2} \ddot{\theta}(k)^2 - A \cdot f_2 \cdot \dot{\theta}(k) \cdot \cos(\omega \cdot k) \]

\[ + \frac{A^2}{2} \cdot f_2 \cdot \cos^2(\omega \cdot k) \]

(3-14)

and

\[ J(\theta(k)) = f_1 + \frac{f_2}{2} \ddot{\theta}(k)^2 - A \cdot f_2 \cdot \dot{\theta}(k) \cdot \cos(\omega \cdot k) + \frac{f_2 \cdot A^2}{4} \]

\[ + \frac{f_2 \cdot A^2}{4} \cdot \cos(2 \cdot \omega \cdot k) \]

(3-15)

The high-pass filter \((z - 1)/(z + h)\) in Figure 3-1 filters out the DC components from \(J(\theta(k))\). As a result, the output from the high-pass filter can be approximated by:

\[ \zeta(\theta(k)) \approx \frac{f_2 \cdot A^2}{4} \cdot \cos(2 \cdot \omega \cdot k) - A \cdot f_2 \cdot \dot{\theta}(k) \cdot \cos(\omega \cdot k) \]

(3-16)

Multiplying \(\zeta(\theta(k))\) with \(A \cdot \cos(\omega \cdot k)\), we arrive at

\[ \xi(\theta(k)) \approx \frac{f_2 \cdot A^3}{4} \cdot \cos(2 \cdot \omega \cdot k) \cdot \cos(\omega \cdot k) \]

\[ - A^2 \cdot f_2 \cdot \dot{\theta} \cdot \cos^2(\omega \cdot k) \]

(3-17)

Using the basic trigonometric identities: 

\[ 2 \cos(2 \cdot \omega \cdot k) \cdot \cos(\omega \cdot k) = \cos(3 \cdot \omega \cdot k) + \cos(\omega \cdot k) \]

and

\[ 2 \cos^2(\omega \cdot k) = 1 + \cos(2 \omega \cdot k) \]

The expression in (3-17) then becomes:

\[ \xi(\theta(k)) \approx \frac{f_2 \cdot A^3}{8} \cdot [\cos(3 \cdot \omega \cdot k) + \cos(\omega \cdot k)] \]

\[ - \frac{A^2 \cdot f_2 \cdot \dot{\theta}}{2} \cdot [1 + \cos(2 \omega \cdot k)] \]

(3-18)

The integrator \(g_f/(z - 1)\) would greatly attenuate the high frequency terms in \(\xi(\theta(k))\). If the filtered term is labeled as \(\xi_f\), then:
\[
\xi_f(\theta(k)) \approx -\frac{A^2 \cdot f_2 \cdot g_r}{2} \cdot \bar{\theta}(k)
\]  
(3-19)

Therefore, the expression for \( \bar{\theta} \) can be approximated as:

\[
\bar{\theta}(k + 1) - \bar{\theta}(k) \approx -\frac{A^2 \cdot f_2 \cdot g_r}{2} \cdot \bar{\theta}(k)
\]  
(3-20)

Replacing \( \bar{\theta} \) in (3-20) with the expression of \( \bar{\theta} \) in (3-10):

\[
\bar{\theta}(k) - \bar{\theta}(k + 1) \approx -\frac{A^2 \cdot f_2 \cdot g_r}{2} \cdot \bar{\theta}(k)
\]  
(3-21)

Thus,

\[
\bar{\theta}(k + 1) \approx \left( 1 + \frac{A^2 \cdot f_2 \cdot g_r}{2} \right) \cdot \bar{\theta}(k)
\]  
(3-22)

In (3-22), the estimation error \( \bar{\theta}(k) \) would decay exponentially when:

1. The value of \( g_r \) is negative and small;

2. The value of \( A \) is small, and \( \left( 1 + \frac{A^2 \cdot f_2 \cdot g_r}{2} \right) < 1 \).

The detailed stability analysis for the perturbation-based ESC is presented in Ariyur et al.’s work [39].

Another category of the ESC that has been reported previously uses tools of parametric self-tuning to find the extremum of the performance function [117][121-125]. The parameters of the performance function are approximated recursively by a polynomial equation using online measurement data. The extremum of the polynomial is then estimated to approximate the extremum of the real performance function.

Though the ESC has been attempted in a number of studies as stated in subsection 2.1.2, the slow convergence rate of the ESC optimization poses serious practical limitations on the application of such a method to physical systems where transient system performance is crucial.
To address this issue, different MGDD optimization structures using the ESC have been proposed in the following 3 chapters. The main difference of the MGDD algorithms in comparison to the conventional ESC is the incorporation of the knowledge of the available system model. This modification helps to improve the convergence speed of the ESC optimization. In this work, only the perturbation-based ESC, Figure 3-1, is applied for the design of different MGDD algorithms.
For most of the ESC optimizations, including the ones referenced in subsection 2.1.2, direct measurement is used to evaluate the performance function during the optimization. Various sensors have been embedded on production CI engine systems and are currently available. These sensors include the temperature sensors for measuring the temperature of the intake-air, micromechanical pressure sensors for measuring the boost pressure at the intake manifold, high-pressure sensors for measuring the fuel rail-pressure, and the inductive engine-speed sensors for measuring the engine rotation speed. A list of the names and operation mechanisms of the onboard sensors are listed in [126]. These sensors are used to register both the system states, e.g. the engine speed, and reference points for engine control purposes, e.g. the accelerator-pedal positions [126].

For some sensor measurements, there exists a large measurement delay associated with either the sensor electronic dynamics or the physical transportation process, such as the response delay of a NOx sensor in a CI engine [127]. The delay in sensor response would normally lead to a longer convergence time for the ESC based optimization. If the delay can be accurately estimated, some delay compensation techniques such as the Smith Predictor can be used as in [128]. But when the delay is difficult to characterize, the slow convergence of the ESC is inevitable.

In this chapter, an MGDD algorithm is applied for the closed-loop optimization of diesel fuel injection timing. A trade-off between the NOx emissions and the thermal efficiency of a single-cylinder CI engine is the constraint for the optimization. Instead of using the NOx sensor measurement directly, a soft sensor is applied. The soft sensor derives the in-cylinder bulk gas temperature using the measured in-cylinder pressure data. The soft sensor and the in-cylinder pressure sensor work together to estimate the real-time values of the designed performance function. Specifically, the maximum in-cylinder bulk gas temperature within one engine cycle and the in-cycle thermal efficiency form the arguments of the performance function for trade-off considerations. The values of bulk
gas temperature represent the averaged in-cylinder temperature and cannot reflect the heterogeneity of the real in-cylinder temperature distribution in a CI engine [7]. In this work, under the selected operating point, a correlation between the value of the bulk gas temperature and the measured NO\textsubscript{x} emission is observed from the experimental study. Because of this observation, the bulk gas temperature is used as an indication for the NO\textsubscript{x} emission.

The inclusion of this soft sensor can improve the convergence speed of the ESC since it avoids large time delays incurred from the use of the NO\textsubscript{x} sensor directly. In this work, the control input, i.e. the crank angle for the Start of Injection, is updated on a cycle-by-cycle basis using the proposed MGDD optimization algorithm. The effectiveness of the proposed method is later demonstrated using experiments.

The remainder of this chapter is organized as follows:

The problem formulation and the proposition of the MGDD algorithm are presented in 4.1. The experimental setup is introduced in 4.2. The simulation and experimental results are presented in 4.3 to demonstrate the effectiveness of the proposed algorithm. Conclusions are made in 4.4.

### 4.1 Problem Formulation and Extremum Seeking Control Using a Soft Sensor

#### 4.1.1 Problem Formulation

Most variables used in this subsection have been defined in 2.1.3.

In the discrete crank angle domain, \(\varphi\), a performance index \(Z_n(\varphi)\) at the \(n^{th}\) engine cycle is presented in this work. The performance index consists of two conflicting objectives:

\[
Z_n(\varphi) = \begin{bmatrix} 1 & 0 \\ 0 & V(\varphi) \cdot (1 - z^{-1}) \end{bmatrix} \cdot x_n(\varphi) 
\]

(4-1)

where

\[
x_n(\varphi) = \begin{bmatrix} T_n(\varphi) \\ p_n(\varphi) \end{bmatrix}
\]

(4-2)
Under a constant fueling rate, the work output from the CI engine should be maximized to achieve a better fuel economy. On the other hand, a lower NO\textsubscript{x} emission requires the combustion flame temperature to stay low [7]. The low flame temperature might adversely affect the engine work output.

It is assumed that between $\varphi$ and $\varphi + 1$, the in-cylinder process is a quasi-steady state process. Therefore, given the measured in-cylinder pressure $p_n(\varphi)$ within the $n^{th}$ engine cycle available, the in-cylinder bulk gas temperature can be derived using (2-2). The maximum in-cylinder bulk gas temperature of the $n^{th}$ engine cycle is defined as:

$$\hat{T}_{\text{max},n} = \max_{\varphi \in [IVC,EVO]} T_n(\varphi) \quad (4-3)$$

$IVC$ represents the crank angle of Intake Valve Closing, and $EVO$ represents the crank angle of Exhaust Valve Opening. In this work, (2-2) and (4-3) form a soft sensor. The soft sensor utilizes the measured $p_n(\varphi)$ to derive the maximum in-cylinder bulk gas temperature as an indication of the change of the NO\textsubscript{x} emission [7][129]. The use of the soft sensor avoids the measurement delay associated with the real NO\textsubscript{x} sensor.

By using the expression of (4-1), the engine work output, $dW_n$ as presented in (2-11), and the maximum in-cylinder bulk gas temperature, $\hat{T}_{\text{max},n}$ as presented in (4-3), at the $n^{th}$ engine cycle are expressed as:

$$dW_n = G_1 \cdot z_n(\varphi) \cdot V_s \quad (4-4)$$

$$\hat{T}_{\text{max},n} = \max_{\varphi \in [IVC,EVO]} [G_2 \cdot z_n(\varphi)] \quad (4-5)$$

Where, $G_1 = [0 \quad 1]$ and $G_2 = [1 \quad 0]$. The engine thermal efficiency $\eta$ obtained at the $n^{th}$ engine cycle is approximated by [28]:

$$\eta_n = \sum_{\varphi \in [IVC,EVO]} \frac{[G_1 \cdot z_n(\varphi)]}{m_{fuel,n} \cdot LHV} \quad (4-6)$$

Therefore, to reach a trade-off between low in-cylinder bulk gas temperature and high engine thermal efficiency, the following performance function for the $n^{th}$ engine cycle is proposed:
\[ J_n = \beta \cdot A_{\hat{T}_{\text{max}} \cdot \max_{\varphi \in [IVC, EVO]} [G_2 \cdot z_n(\varphi)]} + (1 - \beta) \cdot A_\eta \cdot \left\{ 1 - \sum_{\varphi \in [IVC, EVO]} \frac{[G_1 \cdot z_n(\varphi)]}{m_{\text{fuel}_n \cdot LHV}} \right\} \]

Equation (4-7)

\( A_{\hat{T}_{\text{max}}} \) and \( A_\eta \) are elastic coefficients. The elastic coefficients are used to normalize the two arguments in \( J_n \), in order to stay within comparable numerical ranges. In this work, \( A_{\hat{T}_{\text{max}}} \) is set to \( \frac{1}{2000} \) \( K^{-1} \) and \( A_\eta \) is set to \( \frac{1}{0.5} \). As a result, the normalized values of \( A_{\hat{T}_{\text{max}}} \cdot \max_{\varphi \in [IVC, EVO]} [G_2 \cdot z_n(\varphi)] \) and \( A_\eta \cdot \left\{ 1 - \sum_{\varphi \in [IVC, EVO]} \frac{[G_1 \cdot z_n(\varphi)]}{m_{\text{fuel}_n \cdot LHV}} \right\} \) are in the range of \((0.7, 0.9)\). The values of the elastic coefficients are related to the engine operation. These values are determined before the optimization. \( \beta \) is the weighting factor and determines the relative significance of the two arguments.

It is clear that the performance function (4-7) is formed with the cyclic thermal efficiency \( \eta_n \) and the maximum bulk gas temperature \( \hat{T}_{\text{max}_n} \). When the value of the performance function reaches its minimum, an optimum compromise between the desired engine thermal efficiency and the maximum in-cylinder bulk gas temperature is achieved. This optimum scenario is subject to the engine operation and the choice of the weighting factor, \( \beta \).

**Remark 4-1:** The performance function (4-7) evaluates the engine response over one engine cycle. Because of the unmodeled disturbances, a complicated nonlinear static mapping, \( Q_{\text{map}} \), exists between the performance function (4-7) and the control input \( \theta_n \). In this study, \( \theta_n \) is determined as the crank angle for the Start Of Injection, SOI. The performance function for the \( n^{th} \) engine cycle is expressed as:

\[ J_n = f\left( \eta_n(\theta_n), \hat{T}_{\text{max}_n}(\theta_n) \right) = Q_{\text{map}}(\theta_n) \]

Equation (4-8)

The objective of the optimization is to find an optimal control input \( \theta^* \) for the following performance function:
\[ Q_{\text{map}}(\theta^*) = \min_{\theta_n \in [\theta_{\text{min}}, \theta_{\text{max}}]} Q_{\text{map}}(\theta_n) \] (4-9)

Where \( \theta_{\text{min}} \) and \( \theta_{\text{max}} \) are the lower and upper bounds of \( \theta_n \). In this study, the bounds limit the viable crank angle ranges for the SOI. It is noted that the analytical expression of \( Q_{\text{map}} \) is not available. This suggests some on-line techniques are needed to minimize the performance function (4-7).

If the sensor measurement of NO\(_x\) emissions is used directly, the associated gas transportation delays must be considered during the evaluation of the performance function. On the authors’ setup, because of the system configuration, the typical delay for the update of exhaust NO\(_x\) sensor measurement is from 5 seconds to 30 seconds depending on different engine operating conditions. Therefore, the use of this sensor is not feasible for online optimization, as the update speed of the performance function significantly influences the convergence speed of the ESC optimization. Under such a situation, the performance function is only partially measured because of significant unknown delays in NO\(_x\) sensor measurement.

If the performance function is measured on-line, there are many different methods to solve the static optimization problem (4-8). A few methods have been briefly presented in CHAPTER 2. Among them, the ESC is a popular model-free optimization method that has been used in many optimization studies related to IC engine applications [63-68]. In the next subsection, the ESC optimization structure using soft sensor prediction is proposed.

### 4.1.2 Extremum Seeking Control Using a Soft Sensor

The proposed ESC structure is presented in Figure 4-1. When noticeable measurement delay exists within the engine system, this optimization scheme can significantly improve the convergence performance of the conventional ESC, which is one of the author’s main contributions.
As stated in subsection 4.1.1, the performance function $J_n$ utilizes $\hat{T}_{\text{max},n}$ as an indicator for the NOx emissions at the selected operating point. If the NOx measurement from a physical sensor is used, the associated transport delay slows down the ESC optimization [128]. To address this limitation, the soft sensor formed by (2-2) and (4-3) is used to estimate the $\hat{T}_{\text{max},n}$ instead of using direct NOx measurement. By using $\hat{T}_{\text{max},n}$ together with $\eta_n$ the values of (4-7) are calculated.

### 4.2 Experimental Setup

Experiments are conducted on a single-cylinder research CI engine test bench at the author’s laboratory to validate the proposed ESC algorithm. Details of the research platform have been reported previously [28][76][130][131]. For the completeness of this work, in this section, the experimental test setup is briefly described.

The Ford ZSD-420 Duratorq is a 2.0 L, 4-cylinder, common-rail Direct Injection, DI, CI engine. The original engine configuration has been modified by separating one engine cylinder from the other three cylinders to run in a single cylinder mode, i.e. the single cylinder has separated air management and independent fuel injection control. The 3-cylinders-to-1-cylinder configuration enables the investigation of different combustion strategies with the engine coupling to an eddy current dynamometer, which is a non-motoring dynamometer. The 3 cylinders are operated in conventional combustion mode at low load to motor the engine at a stable condition [28][76]. The state of the engine
coolant is controlled continuously by an external conditioning system to maintain a consistent environment for different engine tests [76]. For all the engine tests presented in this work, the coolant temperature is set to 80 °C.

Figure 4-2. Schematic of the single-cylinder research engine platform

A schematic of the research engine platform is presented in Figure 4-2. The intake and the exhaust air-paths of the single-cylinder are separated from the air-paths of the other three cylinders. To simulate the effect of turbocharging, the intake air is supplied to the research cylinder from an oil-free dry air compressor. The volumetric flow rate of the intake air is measured by a Roots flow meter mounted in the intake air-path. An intake surge tank is installed between the Roots flow meter and the intake manifold of the engine for the damping of pressure fluctuations caused by the periodic opening and closing of the intake valves [130].

The gas compositions at both the intake and exhaust manifolds are quantified by different emission analyzers. The gas sampling ports are located at the intake and exhaust manifolds of the research cylinder respectively. A heated sampling line is used to sample the exhaust gas at a temperature of 191 °C according to the requirements of the United
States Environmental Protection Agency, US EPA [132]. The sampled gas first passes through an in-house built conditioning unit to remove particulates and water before it reaches the gas analyzer bench. The gas analyzers are listed in Table 4-1. A production NO\textsubscript{x} sensor for vehicle use is mounted downstream of the exhaust surge tank to measure the NO\textsubscript{x} concentration in the exhaust line. The NO\textsubscript{x} sensor communicates with a personal computer through the Controller Area Network, CAN, protocol. In this work, EGR is not applied during the engine tests. Therefore, the introduction of the engine EGR loop is omitted. Readers can refer to [28][76][130] for more details about the EGR control.

Table 4-1. Gas analyzers

<table>
<thead>
<tr>
<th>Analyzer Type</th>
<th>Measured Species</th>
<th>Analyzer Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Paramagnetic oxygen analyzer</td>
<td>O\textsubscript{2}</td>
<td>CAI 602P</td>
</tr>
<tr>
<td>Heated flame ionization analyzer</td>
<td>Total hydrocarbons</td>
<td>CAI 300M HFID</td>
</tr>
<tr>
<td>Non-dispersive infrared analyzer</td>
<td>CO and CO\textsubscript{2}</td>
<td>CAI 200/300 NDIR</td>
</tr>
<tr>
<td>Chemiluminescence analyzer</td>
<td>NO and NO\textsubscript{2}</td>
<td>CAI 600 HCLD</td>
</tr>
<tr>
<td>Smoke meter</td>
<td>Smoke</td>
<td>AVL Model 415S</td>
</tr>
</tbody>
</table>

Figure 4-3. Schematic of the fueling system

The fuel system for the engine test bench is shown in Figure 4-3. It includes a low-pressure supply circuit that provides fuel to the high-pressure pump. A volumetric flow
detector is used to acquire the fuel consumption. A heat exchanger is installed in the fuel return line to minimize the variation of the fuel supplying temperature.

To enable the application of different fuel strategies to the Ford engine, the National Instruments Real-Time, RT, controllers with embedded Field Programmable Gate Array, FPGA, hardware are configured to provide deterministic control of fuel injection, Figure 4-2. The FPGA is applied to generate Transistor-Transistor Logic, TTL, signals of different pulse patterns to update the injection schedule in real time. The timing of the injection command pulse signal is crank angle-resolved at 0.1 Crank Angle Degree, CAD, while the duration of the injection command is time-resolved in microseconds. The TTL signal is sent to the EFS 8232 solenoid Injector Power Drivers, IPODs. The IPODs are programmed to actuate the injectors with the appropriate voltage/current profiles. In addition, the FPGA is also used to acquire the in-cylinder pressure signals from the in-cylinder pressure sensor. All the control programs are developed under the National Instruments LabVIEW programming environment. In this work, the signals picked for the optimization task are the in-cylinder pressure sensor measurement, the NOx sensor measurement, and the TTL command for SOI control.

4.3 Simulation and Experimental Results

Simulations and engine experiments are carried out to test the effectiveness of the proposed ESC algorithm, subsection 4.1.2. First, the working mechanism and the performance of the soft sensor are evaluated. Then, the effectiveness of the proposed ESC algorithm is validated using real engine experiments.

4.3.1 Simulation Results

The value of $\hat{T}_{max,n}$ is derived by applying the measured in-cylinder pressure trace $p_n(\varphi)$ to (2-2) and (4-3), Figure 4-4. As the value of $\hat{T}_{max,n}$ plays an important role in the proposed ESC optimization scheme, it is important to evaluate whether the $\hat{T}_{max,n}$ can be used to indicate the exhaust NOx emission at the selected engine operating point. In this study, the engine is operated at a constant speed of 1500 RPM. The pressure at the intake manifold is 1.025 bar absolute, and the fueling rate is kept at 4 mg/cycle.
Figure 4-4. Working mechanism of the soft sensor

Figure 4-5. Comparison of steady-state responses among different measurements

Steady-state engine tests are carried out at the selected operating point. As shown in Figure 4-5, $\hat{T}_{max,n}$ captures the change of measured NO$_x$ emission under steady-state testing conditions. When SOI is delayed, measurement from the soft sensor and the NO$_x$ sensor would both reduce. Because the NO$_x$ sensor saturates at 450 ppm, thus measurement from the CAI 600 HCLD NO$_x$ analyzer is also presented in Figure 4-5. From the comparison, at the selected operating point, the soft sensor can estimate the change of NO$_x$ emission under different SOI values.

In this work, NO$_x$ emission is evaluated using parts per million, ppm, because of the good correlation between $\hat{T}_{max,n}$ and the NO$_x$ measured in ppm at the selected engine operating point, Figure 4-5. In practice, the brake-specific units are required for reporting NO$_x$ emissions by the US EPA regulations [132], i.e. $g/bhp \cdot hr$ or $g/kW \cdot hr$. The conversion from ppm to $g/kW \cdot hr$ can be achieved using [130]:

\[ T \]
\[ X_{NO_2} = \frac{Y_{NO_2}}{10^6} \times \frac{M_{NO_2}}{29} \times (\dot{m}_{air} + \dot{m}_{fuel}) \times 3600 / W_b \]  

(4-10)

In (4-10), \( Y_{NO_2} \) is the volumetric concentration of NO\(_2\) in ppm, \( M_{NO_2} \) is the molecular weight of NO\(_2\) in kg/kmol, \( \dot{m}_{air} \) is the intake mass air flow rate in g/s, \( \dot{m}_{fuel} \) is the fuel flow rate in g/s, and \( W_b \) is the engine brake power output in kW.

Figure 4-6. Comparison of transient responses between different sensors

An engine test has also been conducted to compare the transient responses of both the soft sensor and the NO\(_x\) sensor. An increase in fuel injection duration is applied during the test. This is represented by the green curve in Figure 4-6. When the fuel-rail pressure is kept constant at 900 bar, an increase in fuel injection duration causes a rise in fueling rate. In this test, a 275 \( \mu s \) injection duration corresponds to a fueling rate of 2.2 mg/cycle and a 300 \( \mu s \) injection duration corresponds to a fueling rate of 4 mg/cycle.

In Figure 4-6, \( \hat{T}_{max,n} \) responds immediately to the reduction of fueling rate. The measured NO\(_x\) emission also starts to reduce when the fuel injection duration is changed. However, this change in NO\(_x\) measurement takes around 29 seconds to stabilize. The test results show that the delay in NO\(_x\) measurement is remedied using the soft sensor. It is noted that the reading from the soft sensor is only a qualitative representation of the real NO\(_x\) emission under the selected engine operating point. Therefore, the reading from the soft sensor cannot be used to indicate the absolute quantity of the NO\(_x\) emission.
A set of simulations is conducted to demonstrate the importance of the soft sensor for the ESC optimization. The ESC algorithm applied for the simulation is the same as the one presented in Figure 4-1. In the simulation, the real engine system is replaced by the engine model presented in subsection 2.1.3. The in-cylinder pressure generated by the model is applied to (2-2) and (4-3) for the derivation of $\hat{T}_{\text{max}_n}$. The selected values of the model parameters are listed in Table 4-2. These parameters have been verified in [28] and are used to simulate the responses of the research CI engine platform.

In the simulation studies, two different scenarios are created: (1) the $\hat{T}_{\text{max}_n}$ values are updated immediately after each engine cycle; (2) an arbitrary time delay of 14 engine cycles is intentionally created before the $\hat{T}_{\text{max}_n}$ is updated at the $n^{th}$ engine cycle. This is to simulate the scenarios when measurement delay from the NOx sensor is presented, Figure 4-7.
Table 4-2. Model parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\omega_{RPM}$</td>
<td>1500 [ RPM ]</td>
</tr>
<tr>
<td>$m_{\text{fuel}}$</td>
<td>8 [ mg / cycle ]</td>
</tr>
<tr>
<td>$V_c$</td>
<td>0.029 [ L ]</td>
</tr>
<tr>
<td>$V_s$</td>
<td>0.499 [ L ]</td>
</tr>
<tr>
<td>$l$</td>
<td>144 [ mm ]</td>
</tr>
<tr>
<td>$s$</td>
<td>86 [ mm ]</td>
</tr>
<tr>
<td>$\overline{S}_p$</td>
<td>$\frac{2 \cdot \omega_{RPM}}{60}$</td>
</tr>
<tr>
<td>$a$</td>
<td>43 [ mm ]</td>
</tr>
<tr>
<td>$C_1$</td>
<td>2.83 [ - ]</td>
</tr>
<tr>
<td>$C_2$</td>
<td>0.096 [ - ]</td>
</tr>
<tr>
<td>$\bar{R}$</td>
<td>8314.47 [ J / mol · K ]</td>
</tr>
<tr>
<td>$A_s$</td>
<td>$\frac{\pi \cdot B^2}{2} + \frac{\pi \cdot B \cdot s}{2} \left[\frac{1}{a} + 1 - \cos \varphi + \left((\frac{1}{a})^2 - \sin^2 \varphi\right)^{0.5}\right]$</td>
</tr>
<tr>
<td>$\varphi_{CD}$</td>
<td>22 [ CAD ]</td>
</tr>
</tbody>
</table>

1) $\hat{T}_{\text{max,n}}$ without delay:

For the sake of simplicity, the Start of Combustion, SOC, instead of SOI is directly applied as the control input, $\theta_n$, for the simulation studies. The simulation results are presented in Figure 4-8. The blue lines are the original raw data while the black dashed lines are the 5-point average of the raw data. The ESC parameters are set as $A = 3$, $g_r = -2$, and $\omega = 0.2\pi$. $\beta$ is chosen to be 0.3 for the proposed performance function (4-7).
From the results, we can see that the $\theta_n$ converges around 340 CAD and the performance function reaches a value close to 0.817.

In this simulation, a transfer delay of 14 engine cycles is added to the $\hat{T}_{max,n}$ feedback. Compared to the first simulation, the ESC parameters, the weighting factor of the
performance function, and the starting point for the ESC searching are kept unchanged. The simulation results are presented in Figure 4-9. The optimization fails to converge and the value of the performance function increase at the end of the simulation. $\theta_n$ saturates at 340 CAD since the 340 CAD is set as the $\theta_{min}$ for the simulation studies.

To adapt to the delayed $\hat{T}_{max,n}$, the perturbation frequency of the ESC needs to be adjusted. A new perturbation frequency, i.e. $\omega = 0.014\pi$, is applied to the ESC and the results for the new simulation are shown in Figure 4-10.

The ESC optimization manages to converge when the slow perturbation frequency is used. The time required for the ESC convergence is 16572 engine cycles. Since the engine speed is set to 1500 RPM, the optimization has lasted for more than 10 minutes. It is noted that the transportation delay for the NOx sensor on the authors’ engine test platform is much longer than the one applied in the simulation, i.e. in the order of 5 to 30 seconds. The transportation delay is also difficult to characterize as it is also related to the engine operating point. Therefore, a long ESC optimization time is unavoidable. To maintain the engine operating under a steady state for a long period of time is difficult to achieve. As a result, the utilization of this delayed response for the ESC optimization is not practical.

![Figure 4-10. Simulation results: application of a slower perturbation signal](image)

50
4.3.2 Experimental Validation

The optimization of SOI, $\theta_n$, using the proposed ESC structure is implemented on the test engine platform. Four sets of experiments are conducted. All tests are carried out under the same operating conditions to provide a fair ground for the evaluation of the proposed ESC algorithm. The convergence time for each test is calculated from the beginning of the test to the point where the $\theta_n$ varies within 2 CAD.

The operating conditions for all four tests are set as follows:

The fuel injection duration is maintained at 480 $\mu$s. The fuel rail pressure is set to 900 bar. The pressure at the intake manifold is set to 1.025 bar absolute and engine speed is kept at 1500 RPM. No EGR is applied during the engine test.

**Test 1**

In this test, the weighting factor $\beta$ in the performance function is set to zero, (4-7). Therefore, the objective of Test 1 is to maximize the thermal efficiency only.

The test is initiated at the steady-state condition where the fuel injection timing is at an arbitrary value, i.e. 350 CAD, away from the optimum. The experimental results are exhibited in Figure 4-11. The capability of the proposed ESC algorithm is demonstrated as it can gradually drive the SOI timing towards a new steady-state, i.e. the optimal value.

The data is collected at the rate of engine cycles. During the test, the performance function is gradually reduced. The thermal efficiency is optimized after around 800 engine cycles (about 60 seconds). It is also observed that when the best thermal efficiency is reached, the $\theta_n$ oscillates around a fixed mean value, and the $\theta_n$ is around 360 CAD.
In **Test 1**, the parameters of ESC are set as $A = 3$, $g_r = -2$, and $\omega = 0.2\pi$. The tuning parameters are selected to ensure the ESC convergence only. However, a fine tuning of these parameters could also speed up the convergence as suggested in [133].

**Test 2**

To verify that the optimum SOI can be sought by the proposed ESC algorithm for any given initial $\theta_n$ at the selected operating point, the ESC optimization that has been carried out in **Test 1** is repeated with a different initial $\theta_n$. For this test, the initial $\theta_n$ is implemented to be later than the optimal SOI. The test results are shown in Figure 4-12. The ES algorithm successfully locates the same optimum SOI as in **Test 1**. **Test 1** and **Test 2** have demonstrated the effectiveness of the ESC algorithm.
Figure 4-12. Test 2: ESC test starts from a different SOI value

**Test 3**

In this test, the weighting factor $\beta$ is set to 0.3 in the performance function. Furthermore, direct measurement from the NOx sensor is applied to replace $\hat{T}_{\text{max},n}$ in the performance function. The value of the elastic constant, $A_{\hat{T}_{\text{max}}}$, is adjusted to 450 ppm, as 450 ppm is the measurement limit for the NOx sensor.

To use the measured NOx for the ESC optimization, the engine IMEP value is reduced to ~1.8 bar as the NOx sensor saturates at 450 ppm. This is achieved by reducing the injection duration to 300 $\mu$s at 900 bar rail-pressure.

The sampling rate and the ESC parameters are kept the same as the previous tests. The test results are presented in Figure 4-13. Similar to the simulation, the ESC optimization is unable to locate the optimum SOI and the value of the performance function increases. As a comparison, in the next test, the NOx measurement is replaced by $\hat{T}_{\text{max},n}$ from the soft sensor.
Test 4

In this test, the weighting factor $\beta$ is set to 0.3 in the performance function. The measurement from the NO$_x$ sensor is replaced by $\hat{T}_{\text{max},n}$ from the soft sensor to construct the performance function. The engine IMEP is kept at $\sim$1.8 bar, the same as the testing condition presented in Test 3. The ESC parameters are kept unchanged and the test results are presented in Figure 4-14. In the test, the SOI is delayed and the value of the performance function is reduced. Since the engine load is low and the SOI is late, the optimization is stopped after 2590 engine cycles due to the occurrence of engine misfire.
NO\textsubscript{x} and other parameters of interests are measured at both the beginning and the end of all four tests. The parameters are presented in Table 4-3.

Table 4-3. Summary of test results

<table>
<thead>
<tr>
<th>Test</th>
<th>Initial Condition</th>
<th>End of Test</th>
<th>Test Time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( \theta_n )</td>
<td>( \eta_n )</td>
<td>( NO_x )</td>
</tr>
<tr>
<td>Unit</td>
<td>CAD % ppm</td>
<td>CAD % ppm</td>
<td>ppm</td>
</tr>
<tr>
<td>1</td>
<td>350 43.4 867</td>
<td>360 44.4 430</td>
<td>1294</td>
</tr>
<tr>
<td>2</td>
<td>367 41.8 372</td>
<td>360 44.4 421</td>
<td>1080</td>
</tr>
<tr>
<td>3</td>
<td>352 40.5 442</td>
<td>352 39.2 432</td>
<td>3450</td>
</tr>
<tr>
<td>4</td>
<td>351 40.1 445</td>
<td>354 39.4 378</td>
<td>2590</td>
</tr>
</tbody>
</table>

4.4 Summary

An ESC algorithm using a soft sensor is proposed in this chapter. The existence of significant and difficult-to-characterize transportation delay with direct NO\textsubscript{x} measurement hurdles the application of the conventional ESC optimization. A soft sensor is, therefore, used to partially emulate the performance function.

The optimization method identifies a trade-off performance between the thermal efficiency and the maximum bulk gas temperature with the help of on-line in-cylinder pressure measurement. By applying the proposed ESC algorithm, a cycle-by-cycle update of the control input for the CI engine is feasible. The convergence time of the conventional ESC algorithm is also improved. Engine test results further validate the effectiveness of the proposed ESC algorithm.

This chapter provides a viable solution for implementing the ESC when significant and difficult-to-characterize measurement delay is presented in the system. In the next chapter, when measurement delay is negligible, a model-guided ESC algorithm is proposed to speed up the conventional ESC optimization.
CHAPTER 5. MODEL-GUIDED EXTREMUM SEEKING
CONTROL OF INJECTION TIMING IN A COMPRESSION
IGNITION ENGINE

When the measurement delay is negligible, the convergence time required for any
optimization algorithm is dependent on the choice of the initial conditions. If the initial
guess is close to the real optimum point, then less time is required for the optimization to
converge. However, for the conventional ESC, knowledge of the system is unavailable
because of its model-free nature [43][71][72]. As a result, it can not be guaranteed that
the ESC starts near the real optimum point for every trial.

To improve the efficiency of the ESC for IC engine optimization, an ESC structure that
utilizes the model output and the direct measurement is presented in this chapter. The
engine model proposed in subsection 2.1.3 is applied to provide the initialization for the
ESC optimization whenever the engine enters a new operating zone. The model
‘monitors’ the engine operation online and provides the estimated optimal solution for the
performance function. This estimation is a good initial guess for the ESC and improves
the convergence speed of the ESC optimization.

Experiments and simulations are carried out to show the proposed model-guided ESC
algorithm can potentially reduce the convergence time on average compared to the
conventional ESC algorithm. The ESC optimization compensates for the influence of the
external disturbances and the model uncertainties.

In the remainder of this chapter, the model-guided ESC algorithm and the validation of
the engine model are first presented. Then, the effectiveness of the model-guided ESC is
demonstrated using simulations and experiments. Finally, concluding remarks are made.

5.1 Model-Guided Extremum Seeking Control

In this section, two assumptions are made and explained to warrant the applicability of
the model-guided ESC algorithm. For simplicity of presentation, consider the following
single-input-single-output nonlinear static repetitive system:
\[ y_n = g(\theta_n, u_n) \]  \hspace{1cm} (5-1)

where \( \theta_n \) is the system input, \( u_n \) is the global input that decides the operating point of the system, and \( y_n \) is the output from the system at the \( n^{th} \) iteration. The performance function is represented by:

\[ J_n = f(y_n) \]  \hspace{1cm} (5-2)

It is assumed that the nonlinear mappings \( g(\cdot, \cdot) \) and \( f(\cdot) \) are smooth with respect to the input signals. A similar analysis method can be applied to a multi-input-multi-output system [39][66].

Let \( Q_{map} = f \circ g(\theta_n, u_n) = f(g(\theta_n, u_n)) \). The optimization objective is to find \( \theta^* \), based on \( y_n \), and \( Q_{map}(\theta^*) \) reaches an optimum in the \( n = N^{th} \) iteration, i.e. \( \theta_N = \theta^* \).

This condition is always true for any given \( u \in \mathcal{U} \) and \( \theta_n \in \Theta \). \( \mathcal{U} \) and \( \Theta \) are two compact sets in \( \mathcal{R} \). The nonlinear mapping \( Q_{map}(\cdot, \cdot) \) satisfies the following two assumptions:

**Assumption 5-1**: For any given \( u_n \in \mathcal{U} \), there exists a unique \( \theta^* = \theta^*(u_n) \in \Theta \) such that:

\[
\frac{\partial Q_{map}}{\partial \theta_n}(\theta_n, u_n) = 0, \quad \text{if and only if} \quad \theta_n = \theta^* \quad \text{and} \quad \theta_n \in \Theta.
\]

\[
\frac{\partial Q_{map}}{\partial \theta_n}(\theta^* + \zeta, u_n) \cdot \zeta < 0, \quad \forall \zeta \in \Theta \quad \text{and} \quad \zeta \neq 0.
\]

**Assumption 5-1** is a standard assumption in ESC.

**Remark 5-1**: The formulation of this problem is slightly different from the conventional ESC due to the existence of the global input \( u_n \). This global input specifies a particular operating point of the system. Apparently, different values of \( u_n \) may lead to different optimal solutions. The conventional ESC can handle the case when the value of \( u_n \) changes slowly. If \( u_n \) changes fast, ESC algorithms cannot catch up with the varying \( u_n \). The application of the ESC algorithm does not necessarily require the information of the system model \( g(\cdot, \cdot) \). However, if the model of the system, even imprecise, is available, it may help the ESC to achieve fast transitions between two operating points.
Practically, one can use the discrete-time ESC algorithm to address the optimization problem in the iteration domain. If the system is defined in continuous time, it is normally possible to obtain a faster convergence speed by using a faster dither signal, i.e. a larger $\omega$ value. However, for the ESC to search a static mapping under the discrete time domain, the dither signal cannot be under any frequency as the dither signal needs to maintain its periodicity [39]. In the discrete time domain, the sine perturbation signal has the form of $A \cdot \cos(\omega \cdot k)$, Figure 3-1. Where $k$ is the sampling index. To perturb the control input with a periodic signal of zero mean value and nonzero standard deviation [39], the highest frequency, $\omega$, of the sine signal is limited to $\pi$. Therefore, the perturbation frequency has limited influence on the improvement of the convergence speed of the ESC.

If the nonlinear mapping $Q_{map}$ is completely unknown, the discrete perturbation-based ESC algorithm can only achieve a certain convergence rate because the frequency of the dither signal is limited. If an approximated model of the system, $\hat{g}(\theta_n, u_n)$, is available, let $\hat{Q}_{map}: = f \circ \hat{g}(\theta_n, u_n)$ represent an approximated input-to-output map. The map is fully available to the designer. Then the optimal solution of $\hat{Q}_{map}$ is $\hat{\theta}^* = \hat{\theta}^*(u_n)$ for a given $u_n \in \mathcal{U}$. This optimal solution can be computed by using various optimization techniques on-line. The knowledge of the available $\hat{\theta}^*$ can then be used to improve the performance of the ESC algorithm on the real system $g(\theta_n, u_n)$.

For a given $u_n \in \mathcal{U}$, it is denoted that $B_\Delta(u_n) = \max_{\theta_n \in \Theta}|\theta_n - \theta^*|$ where $B_\Delta$ is unknown. We can also define the set $\Theta(u_n) = \{\theta_n: |\theta_n - \theta^*| \leq B_\Delta(u_n)\}$. The next assumption specifies that the model $\hat{g}(\theta_n, u_n)$ or $\hat{Q}_{map}(\theta_n, u_n)$ should have reasonable accuracy or fidelity.

Assumption 5-2: The approximated input-to-output map has certain accuracy such that, for any given $u_n \in \mathcal{U}$, $|\theta^*(u_n) - \hat{\theta}^*(u_n)| \ll B_\Delta(u_n)$. $\theta^*$ is defined in Assumption 1 and $\hat{\theta}^*$ is the optimal solution of $\hat{Q}$.

Remark 5-2: Let us assume that for a given $u_n \in \mathcal{U}$, the unknown optimal value $\theta^*(u_n)$ is within a compact set $\Theta$. Thus, the distance between any $\theta_n(u_n) \in \Theta$ and the unknown optimal value, $\theta^*(u_n)$, is bounded by another compact set $B_\Delta(u_n)$. Any initial value at
the boundary of $B_\Delta(u_n)$ takes the longest time to converge. Assumption 5-2 indicates that when the initial guess is at the boundary of the set $B_\Delta(u_n)$, it converges much slower than the situation when the initial guess is the solution for the model-based optimization. The standard convergence analysis of any iterative process is based on the worst-case scenario. Assumption 5-2 guarantees that when $\hat{\theta}^*(u_n)$ is used to initialize the ESC, the ESC converges much faster than the worst-case scenario.

Assumption 5-2 does not require the model to provide the best initial guess. There is no doubt that a more accurate model leads to a shorter distance between the initial guess $\hat{\theta}^*(u_n)$ and $\theta^*(u_n)$ with less time required for the optimization to converge. Assumption 5-2 points out that compared to the size of the overall search space, the estimated optimal value from the model is relatively closer to the true optimal value. Thus, on average, when the ESC is initialized by the estimated optimal value, the optimization converges faster. If the model is less accurate, as $\hat{\theta}^*(u_n) \in \Theta$, the optimization should not converge any slower than the case when the initial guess is on the boundary of $\Theta$. Although the analysis presented here is rather qualitative, it is noted that reasonably accurate models do exist for some systems under calibrated regions. The different models obtained at various calibration points can be applied to provide an estimate of $B_\Delta$.

If both Assumptions 5-1 and Assumption 5-2 hold, a model-guided ESC scheme can be built up as shown in Figure 5-1. Different from the conventional perturbation-based ESC algorithm, Figure 3-1, a Mode Change Detection block and a Logic block are added in Figure 5-1. These two blocks are used to check whether the operating point of the system is changed, i.e. $u_n \neq u_{n-1}$. If the mode change happens, a new $\hat{\theta}^*(u_n)$ is derived from the model. The $\hat{\theta}^*(u_n)$ is used to reset the ESC as a new initial point. If the current mode remains unchanged, the two blocks will not interfere with the ESC searching. Under this configuration, a sudden change of $u_n$ is tolerable.
Figure 5-1. Schematic for a model-guided ESC structure

**Remark 5-3:** If there is no ESC Controller block, the algorithm shown in Figure 5-1 becomes the traditional open-loop model-based optimization technique, which lacks robustness [15]. On the other hand, if the model is not used, the ESC algorithm may guarantee the robustness, but the convergence performance is limited. Therefore, the proposed model-guided ESC algorithm provides a combination of the available model knowledge with the data-driven optimization to achieve a better system performance.

In this chapter, the author demonstrates the effectiveness of the proposed model-guided ESC algorithm by applying it to optimize the SOI of a research CI engine in terms of the designed performance function.

### 5.2 Engine Model Validation

The engine model presented in subsection 2.1.3 is first validated against measurement at a specific engine operating point. In this chapter, the in-cylinder pressure, $p_n$, is defined as the state variable $x_n$. At crank angle $\varphi$, the state variable is expressed as:
\[ x_n(\varphi) = p_n(\varphi) \]  

\( \varphi \) has a resolution of 1° crank angle. For the model, \( \theta_n \) is the control input and is the SOC at the \( n^{th} \) engine cycle. \( \mathbf{u}_n \) is the global input. \( \mathbf{u}_n \) comprises the engine speed, \( \omega_{RPM} \), and the fueling amount, \( m_{fuel} \).

The chosen parameters for the evaluation of the engine performance are the engine Indicated Mean Effective Pressure, \( p_{ime} \), and the peak pressure rise rate, \( dp_{max} \). At the \( n^{th} \) engine cycle, the two performance indicators are defined as:

\[
p_{ime}(n) = \frac{\sum_{\varphi \in [IVC, EVO]} x_n(\varphi) \cdot V(\varphi) \cdot (1 - z^{-1})}{V_s} \tag{5-4}
\]

\[
dp_{max}(n) = \max_{\varphi \in [IVC, EVO]} (1 - z^{-1}) \cdot x_n(\varphi) \tag{5-5}
\]

It is noted that both \( p_{ime} \) and \( dp_{max} \) are derived from \( x_n(\varphi) \). The acquisition of both \( p_{ime} \) and \( dp_{max} \) are considered to have negligible delay.

Experiments are conducted on the research engine testing bench to validate the model output. The model parameters are the same as those presented in Table 4-2.

A sweep of the control input, \( \theta_n \), is conducted without the addition of EGR. The intake boost pressure is set to 1.5 bar absolute. The engine operates at a constant speed of 1500 RPM.

For the measured in-cylinder pressure traces, the crank angle for 5% of the accumulated heat release [7][28][76][130] is chosen to represent the SOC, \( \theta_n \). The command for the injection duration is held constant at 500 \( \mu \)s. The injection pressure is set to 600 bar. The cyclic fueling rate, \( m_{fuel} \), is 18 mg/cycle. All the values for \( p_{ime} \), defined in (5-4), and \( dp_{max} \), defined in (5-5), are calculated using the measured in-cylinder pressure, \( x_n(\varphi) \) at the \( n^{th} \) engine cycle. The measurement from 200 engine cycles are recorded once the engine enters steady-state under each different \( \theta_n \).

In parallel, simulations under the same testing conditions are conducted using the engine model. Both measurement and model output are presented in Figure 5-2. The results
show that the values of \( p_{\text{ime}} \) and \( d_{\text{p max}} \) are dependent on the change of \( \theta_n \). Standard deviations for both \( p_{\text{ime}} \) and \( d_{\text{p max}} \) are plotted as error bars in Figure 5-2. The error bars for the \( d_{\text{p max}} \) curves are not obviously visible in Figure 5-2B. This is because that the standard deviation values for all \( d_{\text{p max}} \) are less than 0.3% of the corresponding \( d_{\text{p max}} \) values.

Figure 5-2. Comparisons between measurement and model output

As \( \theta_n \) is advanced earlier into the compression stroke, i.e. earlier than 360 CAD in Figure 5-2A, the in-cylinder bulk gas temperature is higher and a larger portion of the fuel energy is dissipated into the cylinder wall in the form of heat transfer [28][76][130]. Therefore, for the same amount of \( m_{\text{fuel}} \), a lower \( p_{\text{ime}} \) is obtained.

When \( \theta_n \) is shifted into the expansion stroke, i.e. later than 360 CAD in Figure 5-2A, the amount of piston work is reduced [7], and the \( p_{\text{ime}} \) value is reduced.

In this work, an optimum \( \theta_n \) is confined to yield the highest thermal efficiency for a fixed \( m_{\text{fuel}} \) by maintaining a constant rail pressure and injection duration. For the results presented in Figure 5-2A, the optimum \( \theta_n \) is close to 359 CAD. The model output captures the trend of the measured \( p_{\text{ime}} \) change. However, discrepancies do exist in the absolute values between measurement and model output.

Besides the thermal efficiency, the peak pressure rise rate, \( d_{p_{\text{max}}} \), is another critical parameter that determines the roughness of combustion in a CI engine [130][134]. From the measurement results in Figure 5-2B, it can be seen that \( d_{p_{\text{max}}} \) gradually decreases as
\( \theta_n \) is shifted from the compression stroke, i.e. earlier than 360 CAD, into the expansion stroke, i.e. later than 360 CAD. The model output can capture the decreasing trend of the measured \( dp_{max} \) as \( \theta_n \) is delayed from an early SOC. In the model, \( \varphi_{CD} \) has been kept constant throughout the simulation, i.e. 22 CAD in this simulation, (2-7). In reality, the combustion duration would change with different \( \theta_n \) values. Under the selected operating point, \( \varphi_{CD} \) varies from 13 CAD to 27 CAD at different \( \theta_n \). This difference between the model settings and the real system is one factor that contributes to the differences in \( dp_{max} \) values between the model output and the measurement. Nevertheless, the model reproduces the experimental trends with sufficient accuracy.

In this study, for a given fuel injection quantity, a higher engine out \( p_{ime} \) and a lower \( dp_{max} \) are desirable for the selected engine operating point.

### 5.3 Optimization of Injection Timing in a Compression Ignition Engine

In this section, the model-guided ESC is presented. The injection timing of a CI engine is optimized using the proposed model-guided ESC algorithm.

#### 5.3.1 Applicable Conditions for Model-Guided Extremum Seeking Control

In general, the model-guided ESC is preferred if the control input, \( \theta_n \), has multiple available choices and the plant model can satisfactorily predict the performances of the real system. For applications related to the operation of CI engines, safe operating boundaries, actuator limits, and reliability constraints are the common factors that determine the values for \( \theta_n \). For the control of SOI for a CI engine, \( \theta_n \) has a flexible and wide selection ranges when a common-rail fuel injection system is employed. Therefore, the convergence rate of the conventional ESC for SOI optimization can vary significantly depending on the chosen initial value for starting the optimization. It is possible that one can choose to start the ESC from a \( \theta_n \) that is far away from the real optimum \( \theta^* \). Provided that a fairly accurate model is applied, the model-guided ESC can supply a reasonable initialization that starts the ESC in the neighborhood of the real optimum \( \theta^* \).
5.3.2 Model-Guided Extremum Seeking Control

The model-guided ESC algorithm, as introduced in Figure 5-1, is adopted for the optimization of SOI, $\theta_n$, for the CI engine test bench, section 4.2. The optimization structure is shown in Figure 5-3. The model-guided mechanism is triggered when the engine operating point is shifted, i.e. when $u_n \neq u_{n-1}$. In this study, the engine operates at 1500 RPM with an injected fuel quantity of ~18 mg/cycle. At the new engine operating point, the optimum control $\hat{\theta}^*$ derived from the engine model is obtained iteratively using the gradient descent algorithm [135]:

$$\hat{\theta}_n = \hat{\theta}_{n-1} - \lambda_b \cdot \frac{f(\hat{\theta}_{n-1} + c, u_n) - f(\hat{\theta}_{n-1} - c, u_n)}{2c} \quad (5-6)$$

In (5-6), $\lambda_b > 0$ determines the step size of the optimization. $c > 0$ is a very small constant number.

![Figure 5-3. Model-guided ESC for the optimization of injection timing in a CI engine](image)

Once $\hat{\theta}^*$ is identified, the parameter is used to initialize the ESC controller. $\hat{\theta}^*$ is used as the very first $\theta_n$ sent to the engine system. The measured feedback is then used to perform the ESC optimization until the system performance function reaches a minimum value. As long as the engine operates under a steady-state condition, i.e. $u_n = u_{n-1}$, the
computation of (5-6) is not triggered, hence the $\theta_n$ is optimized through the conventional perturbation-based ESC algorithm, Figure 3-1.

5.3.3 Design of Performance Function

In this work, the performance target is to drive the IC engine to a reasonably high $p_{ime}$ and a low $d_{p_{max}}$ at a given $u_n$.

$Z_n(\phi)$ is defined as:

$$Z_n(\phi) = \left[ \frac{(1 - z^{-1})}{V_s} \right] \cdot V_s(\phi)$$  \hspace{1cm} (5-7)

Therefore, the performance function that governs the ESC optimization is designed based on (5-4), (5-5), and (5-7) as:

$$f(\theta_n) = \frac{\beta \cdot \max_{\phi \in [IVC,EVO]} [B_1 \cdot Z_n(\phi)]}{A_{d_{p_{max}}}} + \frac{(1 - \beta) \cdot A_{p_{ime}}}{\sum_{\phi \in [IVC,EVO]} [B_2 \cdot Z_n(\phi)]}$$  \hspace{1cm} (5-8)

in which

$$B_1 = \begin{bmatrix} 1 & 0 \end{bmatrix} , \quad B_2 = \begin{bmatrix} 0 & 1 \end{bmatrix}$$

$\beta$ is the weighting factor. $A_{d_{p_{max}}}$ and $A_{p_{ime}}$ are the elastic constants applied to normalize the $d_{p_{max}}$ and the $p_{ime}$ values respectively. The values of the proposed performance function, (5-8), are plotted in Figure 5-4 against the control variable $\theta_n$.

The values of $A_{d_{p_{max}}}$ and $A_{p_{ime}}$ are dependent on the selected operating point. In this study, $A_{d_{p_{max}}}$ is 100 bar/CAD and $A_{p_{ime}}$ is 1 bar under the selected condition. In this the normalized values of $\frac{\max_{\phi \in [IVC,EVO]} [B_1 \cdot Z_n(\phi)]}{A_{d_{p_{max}}}}$ and $\frac{A_{p_{ime}}}{\sum_{\phi \in [IVC,EVO]} [B_2 \cdot Z_n(\phi)]}$ are limited in the range of (0, 0.3). $\beta$ is 0.2 in this study. This indicates that more emphasis is put on the improvement of the thermal efficiency of the engine.
In Figure 5-4, both measurement and simulation results show that the proposed performance function satisfies the requirements in Assumption 5-1 and Assumption 5-2. Therefore, when $\omega_{RPM} = 1500$ RPM, and $m_{fuel} = 18$ mg/engine cycle, the model-guided ESC structure can be applied for the optimization of SOI for the CI engine platform.

### 5.3.4 Simulation Study

First, a simulation study is performed to demonstrate the working mechanism of the model-guided ESC. The Wiebe function [136] is used in an engine model to describe $\delta Q_{hr}$ instead of (2-7):

$$
\delta Q_{hr}(\varphi) = [x_i(\varphi) - x_i(\varphi - 1)] \cdot Q_{hrMAX} \cdot \varphi_{CD}
$$

where

$$
x_i(\varphi) = 1 - exp\left[ -\sigma_1 \cdot \left( \frac{\varphi - \theta_n}{\varphi_{CD}} \right)^{\sigma_2} \right]
$$

(5-10) is the Wiebe function. $x_i$ is the normalized cumulative heat release. $\sigma_1$ and $\sigma_2$ are the coefficients that affect the shape of the heat release. In the simulations, these two values are set as: $\sigma_1 = 3$ and $\sigma_2 = 2$.

The Wiebe function is used to simulate $\delta Q_{hr}$ for the engine model that acts as the target system [7][8][136]. Figure 5-5 shows an example that demonstrates the differences of the $\delta Q_{hr}$ derived from (2-7) and (5-10). For the simulation shown in Figure 5-5, $\theta_n$ is fixed.
at 358 CAD and $\varphi_{CD}$ is set to 22 CAD. In this way, the absolute differences between the two models are minimized. The absolute values between the red and the blue plots are different during the simulated combustion period. However, both plots do share a similar trend in the change of $\delta Q_{hr}$. The $\delta Q_{hr}$ derived from both methods would first increase at an earlier crank angle and then decrease at a later crank angle.

Figure 5-5. Modeled rate of heat release

In the remainder of this subsection, the engine model that utilizes (5-10) is termed the plant model. The engine model that employs (2-7) is termed the simplified model. In this study, the simplified model is applied to locate the $\hat{\theta}^*$ for the initialization of the ESC for the plant model.

Figure 5-6. The effect of model accuracy on model-guided ESC
The operating point is set constant during the simulations, in which $\omega_{RPM}=1500\ RPM$, and $m_{fuel}=18\ mg/cycle$. $\hat{\theta}^*$ is obtained from the simplified model, and $\theta^*$ is acquired from the plant model. The values of $\hat{\theta}^*$ and $\theta^*$ are collected. To investigate the influence of model accuracy on the performance of the model-guided ESC algorithm, the model parameter $\varphi_{CD}$ is varied. For the simplified model, $\varphi_{CD}$ is used in (2-9). For the plant model, $\varphi_{CD}$ is involved in (5-9). For each distinct $\varphi_{CD}$, the values of $\hat{\theta}^*$ and $\theta^*$ are obtained, as shown in Figure 5-6.

Differences always exist between $\hat{\theta}^*$ and $\theta^*$ at different $\varphi_{CD}$ values. This difference is partially caused by the differences in the definition of $\delta Q_{hr}$, Figure 5-5. The largest difference between $\hat{\theta}^*$ and $\theta^*$ is observed when $\varphi_{CD}=22\ CAD$, i.e. a 5.4 CAD difference. The smallest difference occurs when $\varphi_{CD}=26\ CAD$ and the difference is 4.5 CAD. Therefore, $\varphi_{CD}=22\ CAD$ is determined as the case with the worst model accuracy and $\varphi_{CD}=26\ CAD$ is considered as the case with the best model accuracy for this simulation study.

![Figure 5-7](image-url). The effect of model accuracy on the model-guided ESC.
Next, simulations are conducted and the initial $\theta_n$ is varied in the ESC optimization applied to the plant model. Figure 5-7 shows the number of cycles required to achieve the ESC convergence for the plant model starting from different initial $\theta_n$ values. Recall Figure 5-6, when $\varphi_{CD} = 22$ CAD, the $\theta^*$ for the plant model is close to 365 CAD. Therefore, if the initial $\theta_n$ chosen for the ESC optimization is 365 CAD, the optimization converges immediately. As the value of initial $\theta_n$ shifts further away from $\theta^*$, more searching cycles are required before the ESC can converge. The $\hat{\theta}^*$ derived from the simplified model is also highlighted in Figure 5-7 by the red dot, at 361 CAD. Starting from $\hat{\theta}^*$, the convergence time for ESC is around 7 cycles. If the ESC performed on the plant model starts with an initial $\theta_n$ in the range of 348 ~ 370 CAD, the average time required for the optimization to converge is 14.6 cycles. Similarly, when $\varphi_{CD} = 26$ CAD, the $\hat{\theta}^*$ is 357 CAD. The average convergence time required for ESC is 10.6 cycles with an initial $\theta_n$ in the same range, i.e. 348 ~ 370 CAD. When $\hat{\theta}^*$ is the starting point for the ESC searching, only 9 cycles are required for the optimization as highlighted by the red dot in the figure.

If $\hat{\theta}^*$ acquired from the simplified model is used to guide the ESC optimization, a faster convergence can be reached in comparison to the average convergence time for the ESC starting from any $\theta_n$ within the bounded range. However, if ESC starts from a point other than $\hat{\theta}^*$, the convergence speed can be faster if the initial $\theta_n$ is closer to $\theta^*$. This judgement often requires experience and good knowledge about the system behavior. Unfortunately, these pieces of information are not always available to the engineers.

Simulations that demonstrate the transient performance improvement of the ESC optimization are also conducted. In Figure 5-8, the injected fuel quantity $m_{fuel}$ is altered to shift the operating point of the plant model, i.e. the value of $u_n$ is changed. All the other model parameters are kept the same as the ones used for the previous simulation studies. $\varphi_{CD}$ is set to 26 CAD. The conventional ESC is used to adjust $\theta_n$ after every change of $m_{fuel}$. The time required for the ESC convergence are also listed in the figure.
Figure 5-8. Simulation under transient conditions: ESC without model guidance

The same simulation is repeated using the model-guided ESC algorithm. The simulation results are presented in Figure 5-9. Apparently, results for the ESC optimization shown in Figure 5-9 converge much faster after each change in $u_n$. For the conventional ESC optimization, the searching always starts at a $\theta_n$ determined by the $\theta^*$ at the previous $u_n$. Since the ESC is a data driven method, the measurement is the only information that it collects for the derivation of the new control, $\theta_n$. The value of $\theta^*$ obtained at the previous $u_n$ always has some randomness in where it locates, thus the time required for locating $\theta^*$ at the new $u_n$ is not deterministic.

Figure 5-9. Simulation under transient conditions: application of model-guided ESC

When model-guided ESC is applied, the starting point for any optimization is always $\hat{\theta}^*$. Given that the model is an accurate representation of the system and $\hat{\theta}^*$ is close to $\theta^*$. Therefore, the time for the ESC optimization is reduced, and the transient performance of the ESC is more predictable.
5.3.5 Experimental Study

Engine tests are conducted to validate the model-guided ESC approach experimentally. The operating point is set as: $\omega_{\text{RPM}}$ is 1500 RPM and $m_{\text{fuel}}$ is controlled to stay close to 18 $mg/cycle$ with an injection duration of 500 $\mu$s at 900 bar injection pressure. The intake boost pressure is 1.5 bar absolute. No EGR is applied during the engine tests.

The crank angle for SOI, $\varphi_{\text{SOI}}$, is chosen as the control input, $\theta_n$ for the tests. Under the selected operating condition, the crank angle for the start of combustion is related to $\varphi_{\text{SOI}}$ through the ignition delay, $\varphi_{\text{ID}}$, Figure 5-10. $\varphi_{\text{ID}}$ is defined as the difference between the crank angle of the 5% accumulated heat release rate [7][28][76][130] and the $\theta_n$. In this work, to reduce the complicacy of the engine model, the change of $\varphi_{\text{ID}}$ is merely simplified to a constant of 8 CAD, i.e. $\theta_n = \text{SOC} + 8 \text{CAD}$. Therefore, $\hat{\theta}^*$ obtained from the model is directly applied to guide the ESC searching for the optimum $\theta^*$. The performance function, (5-8), is applied for the optimization.

![Figure 5-10. The measured relationship between ignition delay and SOI](image)

The parameters for all the ESC optimizations are set as follows: $A = 2$, $g_r = -3$, and $\omega = 0.2\pi$. Under the selected operating point, the effects caused by the shifting of $\theta_n$ can be tolerated by the CI engine test bench.

The simplified model is employed to guide the ESC during the engine tests. $\varphi_{\text{CD}}$ is selected as 22 CAD for the simplified model. The optimum crank angle for the SOC at this condition was 362 CAD, Figure 5-6. Therefore, $\hat{\theta}^*$ is determined to be 354 CAD as
the starting point for the searching of $\theta^*$. Figure 5-11 shows the engine test results. The plot on the very bottom shows the change of the measured $p_{ime}$ and $dp_{max}$ values during the test. When the variation in the amplitude of $\theta_n$ is within 2 CAD, the optimization is considered to have reached convergence. Therefore, for the model-guided ESC test, the optimization converges in 1600 engine cycles.

![Engine test results](image)

**Figure 5-11.** Engine test results: model-guided ESC

In another two engine tests, the initial $\theta_n$ for the ESC are selected as 352 CAD and 344 CAD respectively. The engine tests are repeated at the same operating point. As shown in Figure 5-12, around 1380 engine cycles are consumed for the ESC to converge when the initial $\theta_n$ is set to 352 CAD. When the ESC optimization starts at 344 CAD, nearly 3500 engine cycles of searching is needed. The measurement results suggest that the model-guided ES does not always provide the best guess to attain the fastest convergence.
ES test that starts from 352 CAD converges faster than the optimization that utilizes model guidance. Nevertheless, when compared to the ESC test with an initial \( \theta_n \) of 344 CAD, the model-guided ESC shows clear advantages in terms of convergence rate.

Figure 5-12. Engine test results: ESC without model guidance

Engine tests that demonstrate the transient performance improvement using the model-guided ESC have also been conducted. The shift of engine operating points is achieved either through the adjustment of engine load, the change of engine speed, or the concurrent alteration of the two parameters. In this study, the fuel injection duration is the only parameter varied during engine operation. The adjustment in injection duration would lead to different \( m_{fuel} \) values and affect the engine load.

Figure 5-13 shows test results with the application of conventional ESC optimization. Model guidance is not utilized in this test. The fuel injection duration has been altered
from 400 μs to 470 μs. \( \omega_{RPM} \) has been kept constant at 1500 RPM throughout the test. The injection duration has first been set to 400 μs. The ESC starts from \( \sim 355 \) CAD. 1764 engine cycles are needed before \( \theta_n \) converges to \( \theta^* \) and \( \theta^* \) is close to 358 CAD. Once converged, the injection duration has been raised to 470 μs to shift the operating point of the engine system. The ESC continues its searching by applying \( \theta_n = 358 \) CAD as the starting point for the ESC at the new operating point. After another 2438 engine cycles, the ESC converges.

![Figure 5-13. Transient engine test: ESC optimization without model guidance](image)

For conventional ESC, the search for \( \theta^* \) at the new operating point always starts at the \( \theta^* \) of the previous operating point. The convergence time required for locating \( \theta^* \) is not deterministic. The initial value for the ESC optimization becomes a parameter that is dependent on the engine operating trajectory. This operating trajectory could change significantly within a limited time.

The model-guided ESC is also applied to locate the \( \theta^* \) under a similar test condition, Figure 5-14. Fuel injection duration has been kept at 400 μs initially. Same as the previous test, the ESC searching starts around 355 CAD. After 1758 engine cycles, the ESC attains its optimum and a shift in engine operating point is executed. The injection duration has been increased to 470 μs and the searching for \( \theta^* \) at the new operating point starts at 359 CAD. 359 CAD is the \( \hat{\theta}^* \) derived from the simplified engine model at the new operating point. The optimization lasts for 708 engine cycles before \( \theta^* \) is located. As \( \hat{\theta}^* \) from the simplified model is used to initialize the ESC, therein the convergence time
for the ESC optimization is less dependent on the operating trajectory of the engine system.

![Figure 5-14. Transient engine test: Model-guided ESC](image)

**5.4 Summary**

In this chapter, an MGDD optimization algorithm, i.e. the model-guided ESC, is proposed for the CI engine optimization. The presence of the system model can improve the convergence rate of the conventional ESC. The model also makes the optimization more deterministic.

In the model-guided ESC algorithm, a system model is computed online. The information extracted from the model is used to reset the ESC optimization based on different system operating point. Simulations and experiments have been carried out to demonstrate the effectiveness of the model-guided ESC. The test results are evidence that the model-guided ESC structure can shorten the convergence time of the conventional ESC optimization. A more predictable searching pattern can also be expected from the proposed model-guided ESC as the starting point for each optimization is determined by the model.

If the system model is not accurate, the online model tuning can be applied in the model-guided ESC structure to improve the accuracy of the model. By tuning the model parameters online, the model integrity is preserved.
CHAPTER 6. MODEL-GUIDED EXTREMUM SEEKING
CONTROL OF INJECTION TIMING IN A COMPRESSION
IGNITION ENGINE WITH ONLINE MODEL CALIBRATION

The model-guided ESC algorithm has been proposed in CHAPTER 5. Compared to the conventional ESC, the new MGDD optimization algorithm has improved the convergence speed and the predictability on transient performance. As stated in Assumption 5-2, the model accuracy is important for the model-guided ESC algorithm. If the system model is not accurate, online model tuning can be introduced to the model-guided ESC algorithm to keep the model integrity high.

In this chapter, the model-guided ESC algorithm with online model calibration is introduced. Thereafter, the effectiveness of the online calibration approach is demonstrated using engine test results. Finally, a summary is made.

6.1 Model-Guided Extremum Seeking Control with Online Model Calibration

The structure for the proposed algorithm is shown in Figure 6-1. The $u_n$ represents the global control inputs that define the engine operating point.

The model is calibrated continuously using the ESC. The ES1 Controller in Figure 6-1 is applied to tune the coefficient $C_1$ in the model, (2-5). The performance function defined for the model calibration is:

$$J(\theta_n) = |\hat{p}_{ime} - p_{ime}|$$

where $p_{ime}$ is the measured IMEP value and $\hat{p}_{ime}$ is the IMEP value acquired from the engine model. Besides $C_1$, all the other model parameters are kept the same during the calibration, Table 4-2.

Whenever a shift in engine operating point takes place, i.e. $u_n(n) \neq u_n(n - 1)$, the mode change detection mechanism is activated, Figure 6-1. The $C_1$ obtained at this instance is applied to update the engine model for the derivation of $\hat{\theta}^*_n$. 

76
In this study, the system control input, \( \theta_n \), is defined as the SOI. As explained in section, 5.1, the \( \hat{\theta}^* \) is obtained using the model-based gradient descent optimization, (5-6). The performance function defined for the gradient descent optimization is:

\[
f(\hat{\theta}_n) = \frac{1}{\hat{p}_{ime}}
\]  

(6-2)

When both the rail fuel pressure and the injection duration are kept constant, an increase in \( p_{ime} \) indicates an improvement of the engine thermal efficiency.

The identified \( \hat{\theta}^* \) is then applied to initialize the ESC optimization, i.e. ES2 Controller in Figure 6-1. The performance function defined for the ESC optimization is:

\[
f(\theta_n) = \frac{1}{p_{ime}}
\]  

(6-3)
Providing the model is accurate, $\hat{\theta}^*_n$ resets $E_{S2}$ in the proximity of $\theta^*$. During the optimization, $E_{S2}$ only compensates the external disturbances if the model has been calibrated by $E_{S1}$.

In the model presented in subsection 2.1.3, the expressions for the rate of heat release, $\delta Q_{hr}$, and the application of a constant specific heat ratio, $\gamma_{\phi_{IVC}}$, are all possible sources that introduce uncertainties to the engine model. As shown in (2-12) and (2-15), an accurate estimation of the piston work increases the accuracy of the estimation of $\hat{p}_{ime}$.

As mentioned in the previous section, $C_1$ is a coefficient that affects the modeled wall heat transfer at different engine operating points. In this work, $C_1$ is tuned for model accuracy improvement, i.e. assuming the value of $C_1$ attributes all model uncertainties even if this is not physically true.

6.2 Experimental Study

Engine tests are carried out on the research CI engine test bench, described in section 4.2. In this section, the tuning of the wall heat transfer coefficient, $C_1$, is demonstrated. This process runs continuously during the engine operation.

6.2.1 Model Online Calibration

The operating point of the CI engine is governed by the engine load and the engine speed. For this engine test, the engine speed is kept constant at 1500 RPM. The engine load can be adjusted by altering the fuel injection duration at a constant injection pressure of 900 bar.

As shown in Figure 6-2, during the engine test, the injection duration is altered from 500 $\mu$s to 450 $\mu$s at the 43rd engine cycle. This change has led to a decrease of engine IMEP from around 5.5 bar to 4 bar, Figure 6-2A. The change in engine operating point requires the calibration of $C_1$ to make sure that the $\hat{p}_{ime}$ value tracks the measured $p_{ime}$.

The $E_{S2}$ is applied to tune the value of $C_1$ in terms of (6-1), Figure 6-2B. Through out the engine test, the absolute value for the average difference between the measured and modeled IMEP is 0.17 bar. The maximum difference between the two values is 0.98
The maximum difference takes place when the engine operating point shifts. The ES$_2$ responds to this sharp change of the performance function by increasing the value of $C_1$ drastically. When the engine is operated at each operating point, the value of $C_1$ is altered continuously by the ES$_2$ to minimize (6-1), Figure 6-1.

![Figure 6-2. Online model calibration](image)

For this experiment, when the injection duration is 500 μs, the calibrated value of $C_1$ is 3.7±0.5. When the injection duration is 450 μs, the calibrated value of $C_1$ is 4.8±0.6.

### 6.2.2 Model-Based Optimization Using Online Calibrated Model

The value of $C_1 = 3.4$ is applied for the engine model when the injection duration is 500 μs. $\hat{\theta}^*_n$ is identified using the gradient descent optimization. Similar to the procedures in subsection 5.3.5, the model identifies the SOC value and then converts it to SOI by adding a constant $\varphi_{1D}$. The constant $\varphi_{1D}$ is 8 CAD in this study.

At a constant fueling rate, the minimization of (6-2) is realized by varying the SOI, $\theta_n$, through the scheduling of injection timing. As the value of (6-2) reaches its minimum, the gradient descent optimization locates the $\hat{\theta}^*_n$ and brings $\hat{p}_{ime}$ to its maximum.
As shown in Figure 6-3, the searching for $\hat{\theta}_n^*$ starts at 352 CAD where the modeled IMEP is 5.1 bar. The equivalent fuel injection duration is 500 μs. The gradient descent optimization delays the value of $\hat{\theta}_n$. As a result, the value of the IMEP increases and the output from (6-2) reduces. When (6-2) reaches its minimum, the optimization terminates and $\hat{\theta}_n$ is 357 CAD. Thus 357 CAD is considered as $\hat{\theta}_n^*$ for the selected operating point. For the model, the value of $\hat{\rho}_{ime}$ is close to 5.4 bar, a 6% increase as compared to the start of the optimization.

![Figure 6-3. Gradient descent optimization](image)

In the following subsection, the derived $\hat{\theta}_n^*$ is applied to reset the ES$_1$ for the derivation of the $\theta^*$ for the engine system when the fuel injection duration is kept at 500 μs.

### 6.2.3 Engine Injection Timing Optimization

Two engine tests are carried out. For this study, the injection duration is set to 500 μs and the injection pressure is kept at 900 bar.

The first engine test is conducted using the conventional ESC optimization, section 3.3. The second engine test uses the proposed model-guided ESC to find the $\theta^*$ in terms of (6-3). Results from both engine tests are shown in Figure 6-4. The derived $\theta^*$ from both tests are close to 357 CAD.
Figure 6-4 shows the engine test results using the conventional ESC. The optimization starts at 367 CAD, which is selected arbitrarily. Based on the measurement, the ESC controller drives the $\theta_n$ to an earlier phasing close to 357 CAD. The measured IMEP value reaches 5.5 bar and the overall time consumed for this optimization is around 750 engine cycles at 1500 RPM.

Figure 6-4B shows the engine test results using the model-guided ESC. The $\hat{\theta}^*_n$ derived from the engine model is 357 CAD. Therefore, 357 CAD is directly applied to initialize the ESC controller for the search of $\theta^*$. For this experiment, the $\hat{\theta}^*_n$ happens to be the same as $\theta^*$, thus minimal time is needed for the searching. The engine IMEP is around 5.5 bar since the very beginning of the optimization.

6.3 Summary

This chapter presents a model-guided ESC structure with the addition of online model calibration. The parameter of an engine model is calibrated continuously to keep the
model integrity high. The model-guided ESC presented in CHAPTER 5 is then applied to optimize the injection timing of a CI engine to achieve the best thermal efficiency at the selected operating point. The calibrated model provides the initialization for the ESC optimization. Engine test results demonstrate the effectiveness of the proposed algorithm.
CHAPTER 7. MODEL-GUIDED DATA-DRIVEN PREDICTIVE CONTROL FOR THE ACTUATION OF AN ELECTRONIC THROTTLE BODY

Different MGDD optimization algorithms are presented in CHAPTER 4, CHAPTER 5, and CHAPTER 6. The proposed MGDD optimization algorithms have shown certain advantages over the conventional ESC. After the optimum parameters are identified by the MGDD optimization, it is crucial for the onboard actuators to make use of the optimum parameters.

An ETB system is an important electronic actuator for an SI engine, which is challenging to control, subsection 2.2.1. To overcome the existing challenges of the ETB control, in this chapter, an MGDD predictive control algorithm is proposed as an alternative solution for ETB actuation.

In this chapter, a data-driven model is first presented to address the rate-dependent hysteresis characteristics of an ETB system. A relatively accurate ETB model provides estimations of the ETB responses and reduces the conservativeness of the controller design. In this work, a data-driven Pseudo-Wiener structure model [137] that includes a linear submodel together with a nonlinear submodel is proposed to estimate the ETB actuation. Compared to the conventional Wiener structure model [138], where the nonlinear submodel is used to estimate the steady-state properties of a system, the proposed model includes a Duham model [139][140] as the nonlinear counterpart to capture the rate-dependent hysteresis of the ETB. To improve the accuracy of the model, the measurement is used to improve the accuracy of the model output. A data-driven optimization structure is then integrated to actuate the ETB to meet the desired performance requirements.

In the remainder of this chapter, both the first-principle ETB model and the proposed data-driven model are identified. The MGDD predictive control structure is then presented. Thereafter, the PID and the SMC control algorithms are applied to actuate the ETB. Comparisons are made between the conventional control algorithms and the new MGDD predictive control algorithm. A brief summary is given in the end.
7.1 Modeling of an Electronic Throttle Body

7.1.1 First-Principle Model

The first-principle ETB model presented in [95][96][98] all share a similar structure as shown in Figure 7-1. This model is built in the continuous time domain. The definitions of the model parameters shown in Figure 7-1 are listed in Table 7-1. The Pulse Width Modulation, PWM, signal is the control signal for driving the DC motor, Figure 2-3. The duty cycle of the PWM signal, \( u_{\text{duty}} \), governs the torque of the DC motor. \( u_{\text{duty}} \) is amplified by an H-bridge driver [141]. The voltage output from the H-bridge driver, \( u_{\text{dr}} \), provides power to the DC motor. A back ElectroMotive Force, EMF, \( u_{\text{emf}} \), exists across the DC motor, which opposes the change of \( u_{\text{dr}} \). The difference between \( u_{\text{dr}} \) and \( u_{\text{emf}} \) is the net voltage applied to the DC motor, \( u_{\text{arm}} \). The armature current, \( i_{\text{arm}} \), of the DC motor is derived by multiplying \( u_{\text{arm}} \) with the armature gain \( K_{\text{arm}} \). The torque output from the DC motor, \( T_m \), is proportional to the amplitude of \( i_{\text{arm}} \). The torque is then transferred from the shaft of the DC motor to the shaft of the butterfly valve through a gearbox, Figure 2-3. The torque output from the gearbox is \( T_l \). Because of the presence of the limp-home spring torque, \( T_s \), and the friction torque, \( T_f \), the final torque exerted on the ETB valve shaft is: \( T_l - T_f - T_s \).

The effect of the back EMF can be neglected [5][96][142] and the torque output from the DC motor is simplified as:

\[
T_m = K_{dr} \cdot K_{arm} \cdot K_m \cdot u_{\text{duty}} \quad (7-1)
\]
Table 7-1. Parameters for the first-principle ETB model

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_{duty}$</td>
<td>Duty cycle of the PWM signal</td>
</tr>
<tr>
<td>$u_{dr}$</td>
<td>Voltage output from the motor driver</td>
</tr>
<tr>
<td>$u_{arm}$</td>
<td>Voltage applied across the DC motor</td>
</tr>
<tr>
<td>$K_{dr}$</td>
<td>Gain of the motor driver circuit</td>
</tr>
<tr>
<td>$i_{arm}$</td>
<td>Armature current</td>
</tr>
<tr>
<td>$K_{arm}$</td>
<td>Equivalent gain between $u_{arm}$ and $i_{arm}$</td>
</tr>
<tr>
<td>$K_m$</td>
<td>Torque constant</td>
</tr>
<tr>
<td>$T_m$</td>
<td>DC motor torque output</td>
</tr>
<tr>
<td>$T_f$</td>
<td>Equivalent friction torque</td>
</tr>
<tr>
<td>$T_s$</td>
<td>Equivalent spring torque</td>
</tr>
<tr>
<td>$T_l$</td>
<td>Torque output from the gearbox</td>
</tr>
<tr>
<td>$J_{eq}$</td>
<td>Equivalent inertia of the ETB system</td>
</tr>
<tr>
<td>$\omega_{etb}$</td>
<td>Angular velocity of the ETB valve plate</td>
</tr>
<tr>
<td>$\theta_{etb}$</td>
<td>Angular position of the ETB valve plate</td>
</tr>
<tr>
<td>$K_{emf}$</td>
<td>Electromotive force constant</td>
</tr>
</tbody>
</table>

In Figure 7-1, the expressions for the three most important complex nonlinearities are not given. These nonlinearities include the friction, the nonlinear limp-home spring, and the gear backlash. The expressions for the three nonlinearities are given below:
A. Friction

Many friction models have been proposed previously to estimate the static, or stick, and kinetic, or slip, friction presented in an ETB system [144][145]. In some of the previous studies [12][95][143], only the kinetic Coulomb friction torque, Figure 7-2, is considered while the static friction is regarded as a model uncertainty. The Coulomb friction model is defined as [143-145]:

\[
T_f = \begin{cases} 
T_d, & \omega_{etb} > 0 \\
0, & \omega_{etb} = 0 \\
-T_d, & \omega_{etb} < 0 
\end{cases} \quad (7-2)
\]

or

\[
T_f = T_d \cdot sgn(\omega_{etb}) \quad (7-3)
\]

where \(T_d\) is a positive constant and \(sgn(\cdot)\) is the signum function defined as:

\[
sgn(x) = \begin{cases} 
1, & x > 0 \\
0, & x = 0 \\
-1, & x < 0 
\end{cases} \quad (7-4)
\]

To improve the prediction of the friction, a viscous friction term is commonly added to the friction model \(T_f\) [6][77]. This friction is proportional to the angular speed of the ETB valve. If the gain of the viscous friction is \(K_v\), the corrected \(T_f\) is expressed as:

\[
T_f = T_d \cdot sgn(\omega_{etb}) + K_v \cdot \omega_{etb} \quad (7-5)
\]
B. Limp-home spring

The nonlinear limp-home spring is designed to restore the butterfly valve to its default limp-home position. The limp-home position, $\theta_{LH}$, is the opening of the ETB when power is not applied to the DC motor [86][88]. The valve plate opening of the ETB, as shown in Figure 7-3, is confined within a bounded range. The $\theta_{etb,max}$ is the maximum ETB opening. The spring gain, $k_s$, is the ratio between the change of the spring torque and the angular speed of the valve plate. $T_{pt}$ is the offset torque caused by the pre-tension of the spring.

The nonlinear spring torque is modeled as [77][78][95][142]:

$$T_s = T_{pt} \cdot \text{sgn}(\theta_{etb} - \theta_{LH}) + k_s \cdot (\theta_{etb} - \theta_{LH}), \quad 0 < \theta_{etb} < \theta_{etb,max} \quad (7-6)$$

![Nonlinear spring](image)

Figure 7-3. Nonlinear spring

C. Gear backlash

A set of gears are used to transfer torque from the DC motor shaft to the axle of the butterfly valve, Figure 2-3. An amount of backlash always exists between the set of gears [95][142]. The output torque from the gearbox, $T_i$, can be derived using the torque of the DC motor, $T_m$. In discrete time, the expression for $T_i$ is:
\[ T_l(k) = \begin{cases} 
  m_{t1} \cdot [T_m(k) - \delta] & T_m(k) > \frac{T_l(k - 1)}{m_{t1}} + \delta \text{ and } T_m(k) > T_m(k - 1) \\
  T_l(k - 1) & \frac{T_l(k - 1)}{m_{t2}} - \delta \leq T_m(k) \leq \frac{T_l(k - 1)}{m_{t1}} + \delta \\
  m_{t2} \cdot [T_m(k) + \delta] & T_m(k) < \frac{T_l(k - 1)}{m_{t2}} - \delta \text{ and } T_m(k) < T_m(k - 1) 
\end{cases} \]  

(7-7)

In (7-7), \( \delta \) is the backlash distance [142]. The values of \( m_{t1} \) and \( m_{t2} \) are determined by the gear ratio [142], Figure 7-4.

(7-7) is also the generalized backlash operator that has been used for the modeling of rate-independent hysteresis [145].

![Diagram of gear backlash]

Figure 7-4. Gear backlash

\( T_l \) is considered bounded [95][142], i.e. \( |T_l(k)| < \rho \). \( \rho \) is the bound for the output torque from the gearbox. In most previous studies [88][95][96][97][98][142], the gear backlash is considered as a model uncertainty in the design of an ETB controller.

As a result, if the effects caused by the back EMF and the gear backlash are neglected [95][96][98], then:

\[ \dot{\omega}_{etb} = \frac{1}{J_{eq}} \cdot [T_m - T_f - T_s] \]  

(7-8)

In (7-8), the backlash of the gearbox is neglected and \( T_m = T_l \). The expressions for \( T_f \) and \( T_s \) can be found in (7-5) and (7-6) respectively.
7.1.2 Pseudo-Wiener Model

The most important aspects of a system are the concepts of system input and output. A system input is the command that affects the performance of the system. A system output is the signal of interest that can be extracted from the system. The disturbances can also impact the way a system behaves. The disturbances are out of the user’s control most of the time. Mathematical expressions that describe the relationships of the system inputs, disturbances, and outputs are called system models. In this subsection, a Pseudo-Wiener structure model is proposed to model an ETB system based on the system input and measured output.

![Figure 7-5. The Wiener structure model](image)

Figure 7-5. The Wiener structure model

The Wiener model, shown in Figure 7-5, consists of a linear dynamic submodel followed by a static nonlinear submodel [137][138]. The system control input, $\theta$, and the model output, $\hat{y}$, are both measurable while the intermediate state, $w$, is often not measurable [137][138][146].

An example that can be described by the Wiener model is the relationship between the weekly training time, $\theta$, for long-distance running athletes and their marathon results, $y$. Let’s assume there is a marathon race every week, and the results of the athletes who have attended the marathon are represented by $y(n)$ in the $n^{th}$ week. The athletes’ marathon results are directly related to their ‘endurance’, $w(n)$, hence $y(n)$ is affected by $w(n)$. The endurance is a variable that cannot be measured directly. Moreover, since $y(n)$ and $w(n)$ both take place at the $n^{th}$ week, the relationship between the two variables is static.

Apparently, the more time, i.e. $\theta(n)$, an athlete spends on training, the higher endurance he/she can achieve. This improvement of endurance is often not an instant result of the athlete’s training taken in the past one week, but is also affected by the trainings taken in the past few weeks, i.e. $\theta(n - 1), \theta(n - 2), \theta(n - 3)$ etc. If an athlete who has been
well trained in the past has reached an endurance of $w(n - 1)$, it is more likely for this athlete to achieve a better endurance after the training in the $n^{th}$ week, i.e. $w(n)$. Therefore, the model structure below is used to estimate the relationship between $\theta$ and $w$ for the athlete in the week $n$:

$$w(n) = \sum_{i=0}^{N} b_i \cdot \theta(n - i) + c \cdot w(n - 1)$$  \hspace{1cm} (7-9)

In (7-9), $N$ is the number for the total weeks of training. $b_i$ is the coefficient for the training time spent in the $i^{th}$ week and is usually a weighting factor. $c$ is the coefficient for the athletes’ endurance achieved in the past week. The expression in (7-9) represents a linear model. This model describes the impact of present and past training on the change of the athlete’s endurance.

Besides the example given above, the Wiener model has been successfully used to model a pH control system [147][148] and a mechanical feed drive system in which a platform is driven by a DC servo motor [137].

For the ETB system, the torque exerted on the axle of the butterfly valve is the sum of the torque from the DC motor and other nonlinear torque listed in subsection 7.1.1. The power is first supplied to the DC motor, which can be approximated by a linear submodel. The torque from the DC motor is exerted on the valve axle where it meets with the nonlinear torque. The nonlinear torque can be described by a nonlinear submodel. The Wiener model is not a first-principle model. Therefore, the model structure and the model parameters do not have clear physical definitions.

In this work, an AutoRegressive-Moving-Average, ARMA, model [149] is chosen as the linear submodel. The submodel is presented in the discrete time domain:

$$w(n) = \sum_{i=1}^{N_1} b_i \cdot \theta(n - i) + \sum_{j=1}^{N_2} c_j \cdot w(n - j)$$  \hspace{1cm} (7-10)
In (7-10), \( N1 \) is the degree of the Moving Average, MA, and \( N2 \) is the degree of the AutoRegressive, AR, terms. \( b_i \) and \( c_j \) are the coefficients for \( \theta(n - i) \) and \( w(n - j) \) respectively. In this study, \( \theta \) is \( u_{duty} \).

A Duhamel model [150][151] is chosen as the nonlinear submodel. Under the discrete time domain, the model is expressed as:

\[
\hat{y}(n) = \hat{y}(n - 1) + \alpha |\Delta w(n)| \left[ f_o \text{sgn}(w(n - 1)) + \sum_{i=1}^{M} f_i (w(n - 1))^{2i-1} - \hat{y}(n - 1) \right] + \sum_{j=1}^{M} g_{j-1} (w(n - j))^{2j-2} \Delta w(n) \tag{7-11}
\]

In (7-11), \( \alpha \), \( f_o \), \( f_i \) and \( g_{j-1} \) are the model coefficients. \( M \) is a positive integer and \( \Delta w(n) = w(n) - w(n - 1) \). Duhamel model is capable of capturing the rate-dependent hysteresis characteristics [140][151] and is attempted for modeling the ETB response, Figure 2-5.

\( \hat{y} \) represents the opening of the ETB valve, \( \hat{\theta}_{etb} \). \( w \) is an intermediate variable, which is the output of (7-10) and the input of (7-11). It is noted that in the Duhamel model, (7-11), both \( \hat{y}(n - 1) \) and \( \hat{y}(n) \) coexist. Therefore, the Duhamel model is a dynamic model. Compare with the conventional Wiener model structure, the proposed model structure uses a linear dynamic model followed by a nonlinear dynamic model to simulate the relationship between the \( u_{duty} \) and the \( \hat{\theta}_{etb} \). Therefore, this model structure is termed as the Pseudo-Wiener model, i.e. (7-10) and (7-11).

In the next section, the model identification is introduced to determine the parameters for both the first-principle ETB model and the Pseudo-Wiener model.

### 7.2 Model Identification and Model Validation

The essence of model identification is to decide the system inputs, system outputs, and disturbances of interests [138]. After a model structure is determined, the system inputs and outputs, sometimes even the disturbances, can be measured. The collected data are then applied for the estimation of the model parameters. The model is later validated.
against alternative sets of data to verify the accuracy of the model. This process can repeat multiple times before a satisfactory model is obtained. In this work, the algorithm applied for the identification of the model parameters is the Levenburg-Marquardt Algorithm, LMA, [152][153][154].

7.2.1 Levenberg-Marquardt Algorithm

The LMA is an iterative technique that minimizes the sum of the squared-error function and is widely adopted to solve nonlinear least square problems [101][154][155]. When the solution is far from the real optimum, the LMA behaves like the gradient descent method, i.e. the parameters are updated in the steepest-descent direction [101][154]. When the solution is in the vicinity of the real optimum, the LMA acts like the Gauss-Newton method, i.e. the sum of the squared-error function is assumed to be locally quadratic [101][154].

In this subsection, a brief description of the LMA is provided. The detailed analysis of the LMA can be found in [101][152][153].

In the remainder of this subsection, vectors and arrays are written in boldface. ‘\( A^T \)’ represents the transposition of array \( A \) and \( \|\cdot\|_2 \) represents the 2-norm.

Let’s assume \( f \) is a function that maps the parameter vector \( A \in \mathbb{R}^n \) to a modeled measurement vector \( \hat{y} = f(A, \theta) \), \( \hat{y} \in \mathbb{R}^m \). \( m, n \) are both positive integers and \( m > n \). \( \theta \) is the input vector, which is kept unchanged during the model identification. Therefore \( \hat{y} = f(A) \) is used to represent \( \hat{y} = f(A, \theta) \).

\( A_0 \) is the initial estimate of the parameter vector. A measurement vector \( y \) is required by the LMA to locate the optimum parameter vector \( A^* \). The \( A^* \) minimizes the squared-error \( e^T e \), where \( e = y - \hat{y} \).

When the value of \( \|\delta\|_2 \) is close to zero, the Taylor series expansion [152] of \( \hat{y} = f(A + \delta) \) at \( A \) is approximated by:

\[
f(A + \delta) \approx f(A) + J\delta = \hat{y} + J\delta
\]

(7-12)
In (7-12), \( J \) is the Jacobian matrix \([156][157]\). The components in \( J \) are:

\[
J_{ij} = \frac{\partial f_i}{\partial A_j}, \quad i \in (1, m), j \in (1, n) \tag{7-13}
\]

The iterative optimization starts from \( A = A_0 \). Within each iteration, the \( \delta \) value that minimizes \( \|y - f(A + \delta)\|_2 \approx \|y - f(A) - J \delta\|_2 = \|e - J \delta\|_2 \) is obtained using the normal equation \([156][157]\):

\[
J^T (e - J \delta) = 0 \tag{7-14}
\]

thus

\[
\delta = (J^T J)^{-1} J^T e \tag{7-15}
\]

In, (7-15), \( J^T J \) is the approximation of the Hessian matrix \([156][157]\) for \( f(\cdot) \). In the LMA, \( J^T J \) is replaced by a square matrix \( N \), i.e.

\[
\delta = N^{-1} J^T e \tag{7-16}
\]

For the square matrix \( N \), the off-diagonal elements are the same as those in \( J^T J \). The diagonal elements in \( N \) are defined as \( N_{jj} = \mu + [J^T J]_{jj} \), where \( j \in (1, n) \). \( \mu \) is a designed parameter called the damping term and \( \mu > 0 \) \([154][155][158]\). Compared to the previous iteration, if the updated parameter vector \( A + \delta \) leads to a reduced \( \|e\|_2 \) value, the \( \delta \) is accepted and the iteration repeats with a decreased \( \mu \) value. Otherwise, \( \mu \) is increased and a new \( \delta \) is derived. This process can repeat multiple times until the \( \delta \) that can reduce the value of \( \|e\|_2 \) is found.

The value of the damping term \( \mu \) changes at each iteration to ensure the decline of \( \|e\|_2 \). Compare with the other diagonal components in \( J^T J \), when the value of \( \mu \) is significantly larger, \( N \) is close to a diagonal matrix. The direction for the change of \( \delta \) is close to the steepest descent direction \([101][154]\). According to (7-16), the magnitude for \( \delta \) is reduced. The inclusion of \( \mu \) is also important when \( J^T J \) is singular \([158]\), i.e. the determinant of \( J^T J \) is zero.
When $\mu$ is significantly smaller than the other diagonal components, the expression of (7-16) is close to (7-15). At this point, the LMA behaves like a Gauss-Newton method [101], where the function is approximated as a quadratic function at $A + \delta$.

The LMA is adaptive because the value of $\mu$ is dependent on the value of $\|\epsilon\|_2$. When the value of $\|\epsilon\|_2$ reduces, the value of $\mu$ would increase. Otherwise, the value of $\mu$ is decreased [101][154][156].

Different strategies for altering the damping term $\mu$ are introduced in [158]. In this work, the Optimization Toolbox from the MathWorks® MATLAB is utilized to perform the LMA [155]. In the program, when $\|\epsilon\|_2$ is reduced, the algorithm sets $\mu_{n+1} = 0.1 \cdot \mu_n$. When $\|\epsilon\|_2$ is increased, $\mu_{n+1} = 10 \cdot \mu_n$ [155].

In this study, the LMA would terminate when one of the following three criteria is met:

1. When $\|\delta_{n+1} - \delta_n\|_2 < 10^{-100} \cdot (1 + \|\delta_n\|_2)$;
2. When $\|f_{n+1} - f_n\|_2 < 10^{-100}$;
3. When 400 cycles of iterations have been performed to save the computational efficiency.

### 7.2.2 Experimental Setup

A brief introduction of the experimental setup is provided in this subsection. The ETB test setup is shown in Figure 7-6. An NI PCI-6229 multifunction I/O device [159] is used for the control and monitoring of the ETB test setup. The analog input channels on the PCI-6229 have a maximum sampling rate of 250 kHz. In this work, the signal from the valve position sensor is sampled by an analog input channel with a sampling rate of 1kHz. The general-purpose counters on the PCI-6229 have a maximum clock frequency of 20 MHz. In this work, the PWM signal is generated from one of the counters. The frequency of the PWM signal is 1kHz. The NI PCI-6229 is embedded on a Personal Computer, PC. The programs for the control of the ETB are developed under National Instruments LabVIEW Environment. A Pololu VNHS2SP30 H-bridge motor driver carrier [160] converts the PWM signal to the power control for the DC motor. The driver supplies the power control to the DC motor periodically with a frequency up to 20 kHz. A 13 VDC voltage supply is required to power the DC motor. A 5 VDC voltage supply is required by the driver.
The ETB valve is designed for a 5.7 L V8 SI engine. The valve plate has a diameter of 80 mm and its limp-home angular position is around 6.4°. The embedded position sensor provides a voltage reading proportional to the opening of the valve. A 0.55 V reading indicates that the valve is fully closed, and 4.85 V indicates that the valve is opened completely.

The ETB is mounted at the upstream of a steady flow bench system, Figure 7-7, where constant air flow can be created to disturb the ETB. A detailed explanation of the steady flow bench can be found in [161].
As shown in the figure, the air flow is generated by four commercially available blower motors, Ametek Lamb 115923, mounted inside the blower box. The motors are powered by 120 VAC with a rated power of 1164 W. Each blower has an independent power switch that enables the flexibility to set the air-flow by turning on a specific number of blower motors. The sharp-edge orifice flow meter has been designed and manufactured according to ISO 5167-1:2003 standards [162]. The pressure difference across the orifice plate is measured and recorded using a Freescale MPXV7025 differential pressure sensor, i.e. the ‘Dif. Pres. Sensor’ in Figure 7-7. The sensor has a measurement range from -25 kPa to 25 kPa that corresponds to an output voltage from 0.2 V to 4.7 V [163]. For non-choked flow, the mass air flow rate, \( Q_{m,air} \), through the orifice nozzle is calculated using the difference pressure, \( \Delta p \), measured across the orifice plate [161]:

\[
Q_{m,air} = C_d \cdot \frac{2 \cdot \Delta p}{\rho_{air}} \cdot \frac{A_{plate}}{\sqrt{1 - \left( \frac{A_{plate}}{A_{upstream}} \right)^2}}
\]  

(7-17)

In (7-17), \( C_d \) is the discharge coefficient whose value is 0.6 in this study. The density of air, \( \rho_{air} = 1.204 \text{ kg/m}^3 \), at 20 °C is used in this study. \( A_{plate} \) and \( A_{upstream} \) represents the cross-sectional area of the orifice plate and the cross-sectional area of the upstream respectively. In this work, the selected cross-sectional area for the orifice plate is 8.55 \( \text{cm}^2 \) and the cross-sectional area of the upstream flow channel is 24.4 \( \text{cm}^2 \). When all four blower motors are turned on, a maximum air flow rate of 100 \( \text{g/s} \) can be reached with a fully opened throttle. In this work, \( Q_{m,air} \) is the same as the air flow rate through the ETB.

### 7.2.3 Identification and Validation of First-Principle Model

In this subsection, the identification of the parameters of the first-principle ETB model is first presented.

Bring (7-1), (7-5) and (7-6) into (7-8). The angular acceleration of the ETB is:
\[
\dot{\omega}_{etb} = \frac{1}{J_{eq}} \left[ K_{dr} \cdot K_{arm} \cdot K_m \cdot u_{duty} - T_d \cdot sgn(\omega_{etb}) - K_v \cdot \omega_{etb} - T_{pt} \cdot sgn(\theta_{etb} - \theta_{LH}) - k_s \cdot (\theta_{etb} - \theta_{LH}) \right]
\]

This can be simplified into:

\[
\dot{\omega}_{etb} = K_1 \cdot u_{duty} - K_2 \cdot sgn(\omega_{etb}) - K_3 \cdot sgn(\theta_{etb} - \theta_{LH}) - K_4 \cdot \omega_{etb} - K_5 \cdot (\theta_{etb} - \theta_{LH})
\]

where:

\[
K_1 = \frac{K_{dr} \cdot K_{arm} \cdot K_m}{J_{eq}} \\
K_2 = \frac{T_d}{J_{eq}} \\
K_3 = \frac{T_{pt}}{J_{eq}} \\
K_4 = K_v \\
K_5 = \frac{k_s}{J_{eq}}
\]

In discrete time, (7-19) becomes:

\[
\omega_{etb}(n) - \omega_{etb}(n-1) = K_1 \cdot u_{duty}(n) - K_2 \cdot sgn(\omega_{etb}(n-1)) - K_3 \cdot sgn(\theta_{etb}(n-1) - \theta_{LH}) - K_4 \cdot \omega_{etb}(n) - K_5 \cdot (\theta_{etb}(n) - \theta_{LH})
\]

The angular speed of the ETB valve cannot be measured directly. Similar to the practice in [96], \(\omega_{etb}(n)\) is approximated using the difference between the valve opening at time \(n\) and \(n-1\):

\[
\omega_{etb}(n) = \theta_{etb}(n) - \theta_{etb}(n-1) = \Delta \theta_{etb}(n)
\]

Therefore, if (7-22) is called into (7-21), the valve opening can be expressed as:
\[
\theta_{ett}(n) \cdot (1 + K_4 + K_5) = K_1 \cdot u_{duty}(n) + (2 + K_4) \cdot \theta_{ett}(n - 1)
- K_2 \cdot \text{sgn}(\Delta \theta_{ett}(n - 1))
- K_3 \cdot \text{sgn}(\theta_{ett}(n - 1) - \theta_{LH})
+ K_5 \cdot \theta_{LH} - \theta_{ett}(n - 2)
\]

(7-23)

In (7-23), \(\theta_{ett}\) is acquired from the valve position sensor. \(u_{duty}\) is the duty cycle control input and \(\theta_{LH}\) is the limp-home position. \(\theta_{LH}\) is 6.4° for the selected ETB.

The goal of the parameter identification is to determine the values for \(K_1, K_2, K_3, K_4,\) and \(K_5\) in (7-23). According to [6][96], the excitation signal \(u_{duty}\) applied for the identification should be a telegraph signal such that the throttle plate is always in motion. This makes the \(T_f\) in the ETB system close to the friction torque described by (7-5) [143-145]. The excitation signal and the corresponding ETB response are plotted in Figure 7-8. The two values chosen for the telegraph input signal are 0.05 and 0.4. To further enrich the frequency spectrum of the excitation signal, a pseudo-random binary sequence with an amplitude of 0.001 is also added to \(u_{duty}\). The duty cycle of the PWM signal, i.e. \(u_{duty}\), is generated randomly. With the selected input signal, the valve is always in motion.

Figure 7-8. Control signal (blue) and ETB position (red)

If the values of the input control signal are lower than 0.3, Figure 7-9 right, or higher than 0.8, Figure 7-9 right, the ETB would stick at certain positions and the proposed model
cannot describe the valve behavior. When the valve is actuated below the limp-home position, this stick phenomenon always shows up when the ETB is moving towards the limp-home position, Figure 7-10 right.

Figure 7-9. ETB actuated above the limp-home

Figure 7-10. ETB actuated below the limp-home

After a few trials and errors, the control signal in Figure 7-8 is selected to serve as the excitation signal for the ETB. The sampled ETB position data are used for model identification, Figure 7-11. The derived parameters are listed in Table 7-2.

Table 7-2. Identified parameters for the first-principle ETB model

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_1$</td>
<td>0.477 m / s$^2$</td>
</tr>
<tr>
<td>$K_2$</td>
<td>0.111 m / s$^2$</td>
</tr>
<tr>
<td>$K_3$</td>
<td>1.302 m / s$^2$</td>
</tr>
<tr>
<td>$K_4$</td>
<td>0.749 1 / s</td>
</tr>
<tr>
<td>$K_5$</td>
<td>0.005 1 / s$^2$</td>
</tr>
</tbody>
</table>
Figure 7-11. Parameter identification for the first-principle ETB mode

The output from the fitted model is plotted against the measurement data in Figure 7-12. The absolute difference between the model output and the measurement data, i.e. $|\theta_{etb} - \hat{\theta}_{etb}|$, is also presented in Figure 7-12.

Figure 7-12. Identified first-principle ETB model
Alternative sets of measured data are then applied for model validation. In Figure 7-13, the data acquired from an excitation signal similar to the one in Figure 7-8 is applied to the identified model. The model output follows the measured data under this scenario.

![Model Validation Diagram](image)

Figure 7-13. Model validation

However, when both the pulse width and the amplitude of the excitation signal are changed to different values, the identified model is unable to capture the system responses, Figure 7-14. Therefore, the identified ETB model is a local model, i.e. the model is only capable of describing the ETB response within a limited operating range.
Figure 7-14. Poor matching between measurement and model output

7.2.4 Identification and Validation of Pseudo-Wiener Model

In this work, the values for $N_1$ and $N_2$ in (7-10) are set as: $N_1 = N_2 = 1$. In (7-11), $M = 2$ is selected. Therefore, the Pseudo-Wiener model is:

$$w(n) = b_1 \cdot \theta(n - 1) + c_1 \cdot w(n - 1)$$  \hspace{1cm} (7-24)

$$\hat{y}(n) = \hat{y}(n - 1)$$

$$+ \alpha \cdot |\Delta w(n)| \cdot [f_0 \cdot sgn(w(n - 1)) + f_1 \cdot w(n - 1)]$$

$$+ f_2 \cdot w(n - 1)^3 - \hat{y}(n - 1)]$$

$$+(g_0 + g_1 \cdot w(n - 1)^2) \cdot \Delta w(n)$$  \hspace{1cm} (7-25)
For an ETB system, the control input $\theta$ is the duty cycle of the PWM signal, $u_{duty}$. The modeled system output, $\hat{y}$, is the angular position of the ETB valve, $\hat{\theta}_{etb}$. (7-24) and (7-25) can then be expressed as:

$$ w(n) = b_1 \cdot u_{duty}(n - 1) + c_1 \cdot w(n - 1) $$

$$ \hat{\theta}_{etb}(n) = \hat{\theta}_{etb}(n - 1) $$

$$ + \alpha \cdot |\Delta w(n)| \cdot \left[ f_0 \cdot sgn(w(n - 1)) + f_1 \cdot w(n - 1) \right. $$

$$ + f_2 \cdot w(n - 1)^3 - \hat{\theta}_{etb}(n - 1) \right] $$

$$ + (g_0 + g_1 \cdot w(n - 1)^2) \cdot \Delta w(n) \hfill (7-27) $$

In this work, instead of using $\hat{\theta}_{etb}$ in radians, the direct voltage readings from the position sensor is used for the model identification. To normalize the input and output of the model within comparable ranges, the input signal is multiplied with a gain of 100 and the position sensor reading is multiplied with a gain of 10. The intermediate variable $w(n)$ does not have a clear physical definition, hence $w(n)$ cannot be directly measured from the ETB system.

For (7-26), with a given initial condition $w(0)$, the output at the $n^{th}$ sampling point is defined as $w\left(n, w(0), b_1, c_1, u_{duty}(\cdot)\right)$. Similarly, for a given $w(n)$ and $w(n - 1)$, the output from (7-27) at the $n^{th}$ iteration is denoted as $\hat{\theta}_{etb}(n, w(n), w(n - 1), A)$, where $A$ is the array of the model parameters: $\alpha, f_0, f_1, f_2, g_0,$ and $g_1$.

The following assumption is proposed before the identification of the model[151].

**Assumption 7-1:** The control input is designed to fully excite all the operation zones of the system and the system is at least locally identifiable.

**Remark 7-1:** Assumption 7-1 guarantees, at least, the local convergence of the model identification problem. Moreover, since the proposed model structure is not physics based, thus the ability to predict the system behavior using the model output is limited. Assumption 7-1 stated that a special excitation signal is required to cover all operating
points of the system. Therefore, by learning the system responses, the model is more likely to provide an accurate prediction of the system behavior globally.

A two-step model identification proposed in [151] is used to determine the parameters for the Pseudo-Wiener model:

**Step 1.** A signal $s(n)$ of length $N_s$ is applied to excite all the modes of dynamics of the ETB system. $s(n)$ is a modulated signal that constitutes of a **pseudo-random binary sequence** plus a **sinusoidal signal with attenuated amplitude and frequency**. The valve position measurement, $\theta_{etb}(s(n))$, is first considered as the output from only the linear submodel, thus the model identification goal is to achieve:

$$
\begin{align*}
\min_{b_1, c_1 \in \mathbb{R}} & \left\| \sum_{n=0}^{N_s-1} \theta_{etb}(n) - w(n, w(0), b_1, c_1, s(\cdot)) \right\|_2 \\
\end{align*}
$$

by tuning $b_1$ and $c_1$. The optimum parameters for the linear submodel are defined as $b^*_1$ and $c^*_1$.

**Step 2.** After the linear submodel is identified, the identification of the nonlinear submodel is initiated. The output from the linear submodel is applied to the nonlinear submodel. In this step, the goal of the identification is to achieve:

$$
\begin{align*}
\min_{A \in \mathbb{R}^{1 \times 6}} & \left\| \sum_{n=0}^{N_s-1} \theta_{etb}(n) - \tilde{\theta}_{etb}(w(n, w(0), b^*_1, c^*_1, s(\cdot))) \right\|_2 \\
& w(n-1, b^*_1, c^*_1, s(\cdot)), A \right\|_2 \\
\end{align*}
$$

by tuning $A$.

The excitation signal chosen in this study is:

$$
\begin{align*}
s(n) &= S_A \cdot e^{-\zeta \cdot n} \cdot \sin(2\pi \cdot f_{max} \cdot e^{-\zeta \cdot n} \cdot n) + S_{off} + \varepsilon(n) \\
\end{align*}
$$

In (7-28), $S_A$ is the amplitude of the signal and $S_{off}$ is the signal offset. $\varepsilon(n)$ represents a pseudo-random binary sequence with the amplitude of 0.001; $f_{max}$ is the maximum frequency of the excitation signal; and $\zeta$ is the attenuation gain.
As shown in Figure 2-5, two distinct hysteresis loops are observed when the ETB is actuated either above or below the limp-home position. Therefore, two Pseudo-Wiener models are used to describe the two hysteresis loops shown in the ETB responses. To identify the Pseudo-Wiener model that covers the ETB actuation above the limp-home position, (7-28) is designed as: $S_A = 0.25$, $f_{max} = 2 \, Hz$, $\zeta = 0.01$, and $S_{off} = 0.26$. This guarantees the ETB can actuates the entire operation range above the limp-home position. The excitation signal is plotted in Figure 7-15.

![Excitation signal](image)

**Figure 7-15.** Excitation signal for model identification: above limp-home position

Following the two-step model identification method, the model parameters can be identified. A comparison between the model output and the measured output is shown in Figure 7-16. The maximum difference between the measurement and model output values can be as significant as 50°.
To improve the model accuracy, modifications are applied to (7-25), i.e.
\[
\hat{\theta}_{etb}(n) = \theta_{etb}(n-1) \\
+ \alpha \cdot |\Delta \omega(n)| \cdot \left[ f_o \cdot sgn(\omega(n-1)) + f_1 \cdot \omega(n-1) + f_2 \cdot \omega(n-1)^3 \right] \\
- \theta_{etb}(n-1) \right) + (g_0 + g_1 \cdot \omega(n-1)^2) \cdot \Delta \omega(n) \tag{7-29}
\]

In (7-29), \(\hat{\theta}_{etb}(n-1)\) is replaced by \(\theta_{etb}(n-1)\). The \(\theta_{etb}(n-1)\) is measured by the position sensor at \((n-1)\), which is available before the start of the \(n^{th}\) iteration. The model becomes a data-driven model as its output is directly derived from the measurement. The output from the modified model is again plotted against the measured output, Figure 7-17. Compare with the results shown in Figure 7-16, the model accuracy is improved.
The method is then applied for the identification of the model that describes the ETB response below the limp-home position. The excitation signal is chosen as:

\[ s(n) = S_A \cdot e^{-\zeta \cdot n} \cdot \sin(2\pi \cdot f_{\text{max}} \cdot e^{-\zeta \cdot n} \cdot n) + S_{\text{off}} + \varepsilon(n) \]  

(7-30)

As the ETB response is highly nonlinear when close to the limp-home, more excitation is preferred close to this position. In (7-30), \( S_A = 0.26 \), \( f_{\text{max}} = 2 \text{ Hz} \), \( \zeta = 0.02 \) and \( S_{\text{off}} = -0.26 \). In this way, the ETB actuates across its entire operation range below the limp-home position. The applied excitation signal is shown in Figure 7-18.
Figure 7-18. Excitation signal for model identification: below limp-home position

The output from the data-driven model is again plotted against the measurement results, Figure 7-19. The identified model parameters are listed in Table 7-3.

Figure 7-19. Data-driven Pseudo-Wiener model: below limp-home position
Table 7-3. Identified parameters for the Pseudo-Wiener models

<table>
<thead>
<tr>
<th>Model Parameters</th>
<th>Above Limp-home</th>
<th>Below Limp-home</th>
</tr>
</thead>
<tbody>
<tr>
<td>$b_1$</td>
<td>0.014</td>
<td>0.170</td>
</tr>
<tr>
<td>$c_1$</td>
<td>0.989</td>
<td>0.865</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>0.027</td>
<td>0.010</td>
</tr>
<tr>
<td>$f_0$</td>
<td>75.800</td>
<td>-2117.380</td>
</tr>
<tr>
<td>$f_1$</td>
<td>-7.539</td>
<td>460.760</td>
</tr>
<tr>
<td>$f_2$</td>
<td>0.006</td>
<td>3.710</td>
</tr>
<tr>
<td>$g_0$</td>
<td>0.831</td>
<td>-12.441</td>
</tr>
<tr>
<td>$g_1$</td>
<td>0.001</td>
<td>0.738</td>
</tr>
</tbody>
</table>

The identified values of $b_1$ are all smaller than 1. This guarantees that the response of the linear submodel is asymptotically stable. In the remainder of this work, the data-driven model for simulating the ETB response above the limp-home position is termed the model PWA. The model that covers the ETB response below the limp-home position is termed the model PWB.

Model validations have also been carried out for PWA and PWB models. First, to verify the PWA model, a slow, i.e. 0.01 Hz, positive control signal and a fast, i.e. 1 Hz, positive control signal have been applied to the ETB. The PWA model is also applied with the same sets of input signals. A comparison between the model output and the measurement results is presented in Figure 7-20. From the results, we can see that the PWA model tracks the system responses well under both fast and slow excitation conditions.
The differences between the model output and the measured output stay within a similar range when air flow is applied to the system, Figure 7-21. All blower motors are turned on to provide a maximum flow of ~100 g/s on the steady flow bench. When disturbed by air flow, the PWA model still estimates the ETB response well. The comparisons show that the model PWA can be applied to simulate the ETB behavior above the limp-home position.
Similarly, the ETB system is commanded with a slow, i.e. 0.01 Hz, negative control signal and a fast, i.e. 1 Hz, negative control signal. Model PWB is commanded with the same control and the comparisons between model output and measurement data are plotted in Figure 7-22. At 0.01 Hz, the differences between the model output and the measurement are small. At 1 Hz, the maximum absolute difference between the model output and system responses has reached 2.5°. Since the operation range below the limp-home position is 0° to 6.4°, a 2.5° difference is almost 40% of the actuation range below the limp-home position. Therefore, the PWB model cannot describe the ETB well, below the limp-home position.
7.3 Model-Guided Data-Driven Predictive Control Strategy

The proposed Pseudo-Wiener model, (7-26) and (7-29), are implemented along with the measured ETB position $\theta_{etb}(n - 1)$ to estimate $\hat{\theta}_{etb}(n)$. $w(n)$ is derived using (7-26). This forms a data-driven model, Figure 7-23.

\[
\begin{align*}
&u_{duty}(n - 1) & \text{ETB} & \theta_{etb}(n - 1) \\
&w(w(n - 1), u_{duty}(n - 1)) & \theta_{etb}(\theta_{etb}(n - 1), w(n), w(n - 1)) & \hat{\theta}_{etb}(n) \\
&[w(n) \\
&[w(n - 1)]
\end{align*}
\]

Figure 7-23. Data-driven ETB model
The data-driven model can predict the ETB response, $\hat{\theta}_{etb}(n)$, based on the available sensor measurement, $\theta_{etb}(n - 1)$, and the output of the linear submodel, (7-26).

By using the data-driven model, an MGDD predictive control algorithm is proposed, Figure 7-24. This structure utilizes the output from a data-driven model to solve an online optimization problem. The performance function is constructed based on the model predictive output and the designed reference trajectory. For an IC engine, the reference trajectories are provided by the calibrated LUTs, as shown in Figure 2-2.

For the simplicity of presentation, in the remainder of this chapter $u_d$ is used to represent $u_{duty}$. The optimization problem at the sampling time $n$ is:

$$\min_{u_{d}} \sum_{n=1}^{N_p-1} J [u_d(n - 1|n), u_d(n - 1|n + 1) ... , u_d(n - 1|N_p - 1), R_{rt}(n|n + 1), R_{rt}(n|n + 2), ..., R_{rt}(n|N_p + n) , \hat{\theta}_{etb}(n|n + 1), \hat{\theta}_{etb}(n|n + 2), ..., \hat{\theta}_{etb}(n|N_p + n)]$$

(7-31)

In (7-31), $N_p$ is the length of the prediction horizon. $R_{rt}$ represents the values defined in the reference trajectory. $u_{d,min}$ and $u_{d,max}$ are the lower and upper bounds for $u_d$. 
In the discrete time domain, the system output sampled at $n$ corresponds to the control input applied at the previous sampling time. In other words, $\theta_{etb}(n)$ is the system output that corresponds to $u_d(n - 1)$. In (7-31), $u_d(n - 1|n)$ represents the $n^{th}$ predicted the control input derived from the system input $u_d(n - 1)$. Similarly, $\hat{\theta}_{etb}(n|n + 1)$ represents the $n + 1^{th}$ predicted the system output derived using $u_d(n - 1)$ and $\theta_{etb}(n)$, both of which are acquired at the $n^{th}$ sampling time. $R_{rt}(n|n + 1)$ represents the designed reference trajectory at the $n + 1^{th}$ sampling time. Because the prediction horizon is $N_p$, the model is calculated for $N_p$ steps ahead of the $n^{th}$ sampling time.

In this work, the control objective is to track the reference trajectory and constrain the change of $u_d$, thus the performance function at $n$ is:

$$J[u_d(n - 1|n), u_d(n - 1|n + 1) ... , u_d(n - 1|n + N_p - 1),$$
$$R_{rt}(n|n + 1), R_{rt}(n|n + 2), ..., R_{rt}(n|n + N_p),$$
$$\hat{\theta}_{etb}(n|n + 1), \hat{\theta}_{etb}(n|n + 2), ... , \hat{\theta}_{etb}(n|n + N_p)]$$

$$= \sum_{k=1}^{N_p} \lambda_{R,k} \cdot [R_{rt}(n|n + k) - \hat{\theta}_{etb}(n|n + k)]^2$$
$$+ \sum_{k=1}^{N_p} \lambda_{u_d,k} \cdot [u_d(n - 1|n + k) - u_d(n - 1|n + k - 1)]^2$$

In (7-32), $\lambda_{R,k}$ and $\lambda_{u_d,k}$ are the weighting factors for the $k^{th}$ prediction. $u_d(n - 1|n) = u_d(n - 1)$ is the initial control input. The rate of $u_d$ change is constrained in (7-32) to prevent a sudden change of the ETB.

For the ease of presentation, in the remainder of this section, the performance function derived at the sampling time $n$ with $N_p$ steps of prediction is represented by $J \left( u_d(n - 1|n + N_p - 1) \right)$.

When $J \left( u_d(n - 1|n + N_p - 1) \right)$ is available, the performance function is optimized online using different optimization methods. These methods include the Gauss-Newton method [101], the LMA [101][154], and the gradient descent method [101][135]. In this
work, the gradient descent method, (5-6), is chosen to solve the optimization problem. The \( u_d(n) \) is derived using:

\[
\begin{align*}
\dot{u}_d(n) &= u_d(n) - \lambda_b \cdot \frac{\partial J(u_d(n-1|n+N_p-1)^{j-1})}{\partial u_d(n)^j} \\
\dot{u}_d(n+1) &= u_d(n + 1)^{j-1} - \lambda_b \cdot \frac{\partial J(u_d(n-1|n+N_p-1)^{j-1})}{\partial u_d(n+1)^{j-1}}
\end{align*}
\]

(7-33)

(7-34)

\[ u_d(n + N_p - 1)^j = u_d(n + N_p - 1)^{j-1} - \lambda_b \cdot \frac{\partial J(u_d(n-1|n+N_p-1)^{j-1})}{\partial u_d(n + N_p - 1)^{j-1}} \]

(7-35)

The superscript \( j \) represents the \( j^{th} \) optimization iteration. This gradient optimization would terminate when either of the following condition is met:

\[
(1) \left\| J(u_d(n-1|n+N_p-1)^{j}) - J(u_d(n-1|n+N_p-1)^{j-1}) \right\|_2 < \delta_g \text{ with } \delta_g \text{ being a very small positive number}; (2) \text{ when the optimization iteration exceeds a predefined time } t_{\text{limit}}. \text{ When the optimization stops at the } j^{th} \text{ iteration, } u_d(n) = u_d(n)^j \text{ is used to update the control input for the ETB system.}
\]

**Remark 7-2:** The above procedure is very similar to the one proposed in *Theorem 2* in [164]. However, as the intermediate variable \( w(n) \) is not measurable, it is difficult to apply a receding horizon observer for the proposed model. Since the linear submodel is stable and the control input is bounded, \( w(n) \) is uniformly bounded. By applying *Theorem 2* in [164], if the model uncertainties are bounded, the proposed algorithm can ensure uniform boundedness within the prediction horizon.
7.4 Experimental Study

Experiments are designed to examine the effectiveness of the proposed MGDD predictive control algorithm for the ETB actuation. Airflow is applied during the flow bench experiments to test the robustness of the MGDD predictive control algorithm. Conventional control algorithms like the PID and the SMC have also been attempted for the ETB control. The performances of the ETB achieved by different control strategies are compared and analyzed in the end.

7.4.1 Determination of Prediction Horizon

The value of the prediction horizon $N_p$ is first determined for the MGDD predictive control algorithm. Only model PWA is studied in this subsection. Similar procedures can be applied for the scenario when model PWB is used for the control derivation.

Based on (7-32), the value of the performance function is dependent on the model output. If the model output can accurately predict the system performance, a larger prediction horizon can be applied. As a result, the updated $u_d(n)$ can adjust the ETB to better yield the designed performance requirements.

However, as shown in Figure 7-16, when the measured ETB position is not applied to correct the model output, the quality of the model prediction is poor. It’s only when $\hat{\theta}_{etb}(n - 1)$ in (7-25) is replaced by $\theta_{etb}(n - 1)$ that the model estimation is improved. In the proposed MGDD predictive control algorithm, during the derivation for $u_d(n)$, the prediction of $\hat{\theta}_{etb}(n|n + 1)$ is based on the available $\theta_{etb}(n)$ value. When the prediction proceeds, i.e. for $k\in[2, N_p]$, $\hat{\theta}_{etb}(n|n + k)$ is derived by $\hat{\theta}_{etb}(n|n + k - 1)$, the new model prediction is derived from the model output at $n + k - 1$. The measurement is not available to correct the model output. Therefore, the model accuracy could deteriorate, Figure 7-16.

To determine the possible length of the prediction horizon, $N_p$, the following tests are designed:
(1) Tracking tests using \( N_p = 1, 2, 3, 4 \) are conducted. The weighting factors in the performance function, (7-32), are set as: \( \lambda_{R,k} = 1/N_p \) and \( \lambda_{u_d,k} = 0 \), where \( k = 1 \ldots N_p \).

(2) The values of the weighting factors are changed for different tracking tests. When \( N_p = 2 \), \( \lambda_{R,2} = 1 \) and \( \lambda_{R,1} = 0 \). When \( N_p = 3 \), \( \lambda_{R,3} = 1 \) and \( \lambda_{R,1} = \lambda_{R,2} = 0 \). \( \lambda_{u_d,k} = 0 \) is set for all the tests, where \( k = 1 \ldots N_p \).

First, the chosen reference trajectory for all the tests is shown in Figure 7-25. Four controllers with \( N_p = 1, 2, 3, 4 \) are designed. The corresponding values for the weighting factors \( \lambda_{R,k} \) are set as: 1, 0.5, 0.33, and 0.25. The values of \( \lambda_{u_d,k} \) are set to zero for \( k = 1 \ldots N_p \).

![Figure 7-25. Reference trajectory](image)

The test results are presented in Figure 7-26. The ETB tracks the reference well until \( N_p \) reaches 4. Persistent oscillation of the valve plate takes place when the 4-step prediction is applied. Therefore, the length of the prediction horizon \( N_p \) is limited to a maximum of three steps.
Next, when the prediction horizon is 2 and 3, the tracking tests using different weighting factors are performed, Figure 7-27. When $N_p = 3$, the ETB fails to track the step reference. For this test, the performance function puts all the weights on the 3rd step of the model prediction, i.e. $\lambda_{R,3} = 1$, and the system control, $u_d(n)$, is derived based on the model output. This suggests that the model can hardly estimate the system output at this point and adding the model output to the prediction horizon would not improve the quality of the derived control. As a result, $N_p = 2$ is determined as the maximum prediction horizon for the proposed predictive control algorithm using the identified Pseudo-Wiener model.
In this work, to keep the model integrity high, one step predictive control is applied to all experiments.

**7.4.2 Experimental Results**

**7.4.2.1 Model-Guided Data-Driven Predictive Control**

In this subsection, the results of ETB actuation using the MGDD predictive control algorithm are presented. First, the control is carried out for the tracking of a step reference using the model PWA. The ETB is actuated above the limp-home position. Two different performance function designs are used in the tests:

- **MGDD 1.** $\lambda_{R,1} = 0.85$ and $\lambda_{u_d,1} = 0.15$ are set for (7-32).

- **MGDD 2.** $\lambda_{R,1} = 1$ and $\lambda_{u_d,1} = 0$ are set for (7-32).

As shown in Figure 7-28, the adjustment of the performance function can affect the ETB response. When MGDD 1 is used for designing the controller, the ETB can track the reference trajectory well. No overshoot is observed during the test. When MGDD 2 is applied for the controller design, compared to the first test, the ETB responds faster to the step change in the reference. However, overshoot effects are also observed during the transients at each step reference change. This comparison demonstrates that the response of the ETB system can be adjusted by tuning the parameters of the performance function.
Figure 7-28. Effect of different performance function designs

Figure 7-29. Control of ETB under air flow conditions

The quasi-steady air flow is then applied to the ETB to test the robustness of the MGDD predictive control against air flow disturbances. One motor in the blower box is turned on during the test to provide an air flow rate of 27 $g/s$, when the ETB is fully open. As
shown in Figure 7-29, when only open-loop control is applied, the holding of the opening position of the ETB is disturbed by the air flow change. When the MGDD controller is introduced in the closed-loop control, the ETB tracks the reference trajectory. The ETB opening remains at the desired reference position under air flow, Figure 7-29.

Next, the ETB valve is commanded to track a reference signal that varies across the limp-home position, 6.4°. Both PWA and PWB models are included in the controller design. The control structure is presented in Figure 7-30.

Figure 7-30. MGDD control structure using two models

Compared to the structure shown in Figure 7-24, a ‘Switch Mechanism’ and a ‘Smoothing Mechanism’ are added. The switch mechanism uses the measured ETB position to determine the appropriate model for the design of the performance function. When $\theta_{etb}(n - 1) < 6.4^\circ$, it guides both the sensor output and the control input to the model PWB, otherwise, the signals are routed to the model PWA. The output from the selected model is sent to update the performance function. The inclusion of the smoothing mechanism,

$$R_{rt}^s(n + 1) = \alpha_s \cdot R_{rt}(n + 1) + (1 - \alpha_s) \cdot R_{rt}(n) \quad (7-36)$$
can reduce the roughness of any sudden change that takes place in the reference signal, $R_{rt}$. This also provides enough time for the switch mechanism to respond. In (7-36), $\alpha_s$ is the weighting factor that adjusts the roughness of $R^s_{rt}$, and $0 < \alpha_s < 1$.

Figure 7-31 shows the control results. When the ETB system actuates above the limp-home position, the system tracks the reference. However, when the ETB actuates below the limp-home position, valve plate oscillation occurs. The oscillations are pointed out by the red arrows in Figure 7-31. When the ETB opening is small, the limp-home effect becomes obvious. More importantly, the identified PWB model cannot capture the valve response very well, Figure 7-22. As a result, the output from the model PWB cannot provide accurate estimations of the ETB behavior. The $u_d$ derived using model PWB has caused the ETB to chatter.

![Graph showing control results](image)

Figure 7-31. Control of ETB: step reference across the limp-home position

### 7.4.2.2 Comparisons between the Proposed Predictive Control and the Conventional Control Algorithms

In this subsection, a conventional PID controller and an SMC controller are developed for the ETB valve actuation.

**PID Controller**

A conventional PID controller is first designed for the ETB control. The PID controller is expressed in the velocity form in discrete time [165]
\[ u_d(n) = u_d(n-1) + K_p \cdot [\varepsilon(n) - \varepsilon(n-1)] + K_i \cdot \varepsilon(n) \]
\[ + K_D \cdot [\Delta \varepsilon(n) - \Delta \varepsilon(n-1)] \] (7-37)

In (7-37), \( K_p \) is the proportional gain, \( K_i \) is the integral gain for the summation term, and \( K_D \) is the derivative gain for the difference term. \( \varepsilon(n) = R_{rt}(n) - \theta_{etb}(n) \) and \( \Delta \varepsilon(n) = \varepsilon(n) - \varepsilon(n-1) \). In this work, the controller gains are selected through trial and error. The selected parameters are as follows: \( K_p = 0.8, K_i = 0.1, \) and \( K_D = 0.05 \).

**SMC Controller**

An SMC controller based on the identified first-principle ETB model is proposed. The controller structure is similar to the ones used in [95][97][98].

First, the sliding surface is defined as:

\[ s_1(n) = c_s \cdot \varepsilon(n) + \Delta \varepsilon(n) \] (7-38)

Take the difference between \( s_1(n) \) and \( s_1(n-1) \), i.e. \( \Delta s_1(n) \), thus:

\[ \Delta s_1(n) = c_s \cdot \Delta \varepsilon(n) + \Delta \varepsilon(n) - \Delta \varepsilon(n-1) \] (7-39)

Since, \( \varepsilon(n) = \theta_{etb}(n) - R_{rt}(n) \), thus (7-39) becomes:

\[ \Delta s_1(n) = (c_s + 1) \cdot (\theta_{etb}(n) - R_{rt}(n)) \]
\[ - (c_s + 2) \cdot (\theta_{etb}(n-1) - R_{rt}(n-1)) \]
\[ + (\theta_{etb}(n-2) - R_{rt}(n-2)) \] (7-40)

Let’s assume the overall model uncertainty is \( d(n) \), and add \( d(n) \) in (7-23):

\[ \theta_{etb}(n) = [K_1 \cdot u_d(n) + (2 + K_4) \cdot \theta_{etb}(n-1)] \]
\[ - K_2 \cdot sgn(\Delta \theta_{etb}(n-1)) - K_3 \cdot sgn(\theta_{etb}(n-1) - \theta_{LH}) \]
\[ + K_5 \cdot \theta_{LH} - \theta_{etb}(n-2)]/(1 + K_4 + K_5) + d(n) \] (7-41)

Replace \( \theta_{etb}(n) \) in (7-40) with the expression in (7-41):
\[ \Delta s_1(n) = (c_s + 1) \cdot \{(K_1 \cdot u_d(n) + (2 + K_4) \cdot \theta_{etb}(n - 1)
\]
\[ - K_2 \cdot \text{sgn}(\Delta \theta_{etb}(n - 1)) - K_3 \cdot \text{sgn}(\theta_{etb}(n - 1) - \theta_{LH})
\]
\[ + K_5 \cdot \theta_{LH} - \theta_{etb}(n - 2))/(1 + K_4 + K_5) + d(n) - R_{rt}(n)\}
\[ -(c_s + 2) \cdot \epsilon(n - 1) + \epsilon(n - 2)\] (7-42)

Therefore, if the control \( u_d(n) \) is designed as:
\[ u_d(n) = \{(1 + K_4 + K_5) \cdot [(c_s + 2) \cdot \epsilon(n - 1) - \epsilon(n - 2)
\]
\[ +(c_s + 1) \cdot R_{rt}(n))/(c_s + 1) - (2 + K_4) \cdot \theta_{etb}(n - 1)
\]
\[ + \theta_{etb}(n - 2) + K_2 \cdot \text{sgn}(\Delta \theta_{etb}(n - 1))\]
\[ + K_3 \cdot \text{sgn}(\theta_{etb}(n - 1) - \theta_{LH}) - K_5 \cdot \theta_{LH})/K_1\]
\[ -[(1 + K_4 + K_5)/[K_1 \cdot (c_s + 1)]] \cdot K_{SMC} \cdot \text{sgn}(s_1(n))\] (7-43)

Replace \( u_d(n) \) in (7-42) with (7-43):
\[ \Delta s_1(n) = -K_{SMC} \cdot \text{sgn}(s_1(n)) + (c_s + 1) \cdot d(n)\] (7-44)

As stated earlier in this work, the model uncertainty, \( d(n) \), is bounded. Let’s assume \( D_{max} \) is the bound, i.e.
\[ |d(n)| \leq D_{max}\] (7-45)

Therefore, when \( K_{SMC} \gg (c_s + 1) \cdot D_{max} > 0\):
\[ \Delta s_1(n) \approx -K_{SMC} \cdot \text{sgn}(s_1(n))\] (7-46)

Thus,
\[ s_1(n) \cdot \Delta s_1(n) \approx -K_{SMC} \cdot |s_1(n)| < 0\] (7-47)

Therefore, the sliding surface is reachable.
However, the value for $D_{\text{max}}$ is unknown in practice. Therefore, trials and errors are required to determine both the values of $c_s$ and $K_{\text{SMC}}$. In this work, $c_s = 1$ and $K_{\text{SMC}} = 5.8$. Please refer to Table 7-2 for the values of the other parameters in the controller. Lastly, the $\text{sgn}(\cdot)$ function in (7-43) is replaced by $\frac{2}{\pi} \cdot \text{arctan}(\cdot)$ to reduce a chattering effect [95].

Comparisons are made between the three different control strategies applied for ETB actuation. Step tracking tests are first conducted when the ETB is only actuated above the limp-home position. Figure 7-32 shows the control results when no air flow is applied to the ETB. The absolute error is defined as: $\epsilon_{\text{abs}} = |R_{rt} - \theta_{etb}|$. This error is used to evaluate the tracking performance. The Mean Squared Error, MSE, values defined by:

$$\epsilon_{\text{MSE}} = \frac{1}{N} \cdot \sum_{i=1}^{N} \left( R_{rt}(i) - \theta_{etb}(i) \right)^2$$  \hspace{1cm} (7-48)

are also calculated for each test. The derived MSE are listed in Table 7-4.

Table 7-4. MSE for the experiments shown in Figure 7-32

<table>
<thead>
<tr>
<th>Control Strategy</th>
<th>MGDD</th>
<th>PID</th>
<th>SMC</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\epsilon_{\text{MSE}}$</td>
<td>$16.1^o^2$</td>
<td>$15.4^o^2$</td>
<td>$31.5^o^2$</td>
</tr>
</tbody>
</table>

Figure 7-32. Step tracking comparison: no air flow condition
In the results, no obvious differences in system performance can be observed when either the PID or the MGDD controller is applied for the ETB actuation. The ETB responds the slowest when controlled by the SMC controller. The MSE value derived from the SMC control test is also the largest of the three.

Next, quasi-static air flow is applied during the ETB step tracking tests. Like the previous experiments, both $\epsilon_{abs}$ and the MSE values are calculated. The test results are plotted in Figure 7-33. The MSE values are listed in Table 7-5.

Table 7-5. MSE for the experiments shown in Figure 7-33

<table>
<thead>
<tr>
<th>Control Strategy</th>
<th>MGDD</th>
<th>PID</th>
<th>SMC</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\epsilon_{MSE}$</td>
<td>14.4°²</td>
<td>18.8°²</td>
<td>31.5°²</td>
</tr>
</tbody>
</table>

![Figure 7-33. Step tracking comparison: with air flow condition](image-url)
Similar to the previous tests, the ETB controlled by the SMC algorithm shows the worst performance. When the ETB is controlled by either the PID or the MGDD algorithm, robustness against the air flow disturbances is observed.

The ETB valve has also been commanded to track sinusoidal references of different frequencies. The chosen references are varying above the limp-home position and the test results are shown in Figure 7-34.

Figure 7-34. Tracking of sinusoidal references: no flow condition
Because the model PWA can predict the system behavior, when the ETB is controlled by the MGDD algorithm, the ETB tracks the reference. Once the ETB is controlled by either the PID or the SMC algorithm, the closed-loop system is not able to keep pace with the change of the reference trajectory when it oscillates at 2 Hz.

Lastly, the tracking tests with reference signals cross the limp-home position are carried out. The experimental results are presented in Figure 7-35. When the MGDD algorithm is applied for the ETB control, the ETB can track the reference trajectory. However, when the ETB is actuated below the limp-home position, poor system performance, such as chattering and overshoot, are still observed.

![Figure 7-35. Step tracking comparison: reference crosses the limp-home position](image)

When the ETB is controlled using the PID algorithm, a similar system performance above the limp-home position is observed compared to the condition when the MGDD predictive control is applied. However, as highlighted by the red dash box, when a sudden drop of the reference value takes place, i.e. from an opening above the limp-home position to an opening below the limp-home position, the ETB fails to track the reference. Since the PID control only relies on the value of $\epsilon$, when $\theta_{etb}$ is above $R_{rt}$, $\epsilon$ is negative. The $u_d$ derived from the PID algorithm would attempt to reduce this difference by decreasing the value of the updated $u_d$.

However, because of the nonlinearity of the ETB, the system enters a dead zone, thus the valve plate is stuck even when the value of $u_d$ is changing. As a result, the duty cycle saturates at -1. This is one of the disadvantages of the PID control designed in this work. The absence of the system model makes the control decision solely based on the
measured output. In Figure 7-36, it is shown that this control saturation is repeatable when the ETB is controlled by the designed PID controller.

The ETB controlled using the SMC algorithm responds slowly. This can be observed when the ETB is close to the limp-home position, Figure 7-35.

![Figure 7-35](image)

Figure 7-36. Step tracking comparison: reference crosses the limp-home position

7.5 Summary

In this chapter, a data-driven Pseudo-Wiener model is proposed to estimate the behavior of an ETB system. To improve the model accuracy, the measured data is used to update the model output. A performance function is then formed using the model output to predict the ETB behavior. The gradient descent optimization is applied to find the optimum control in terms of the designed performance function. This forms the MGDD predictive control algorithm.

The ETB actuation using the MGDD predictive control has shown advantages over the conventional control strategies like the PID and the SMC. However, these advantages rely on the accuracy of the system model. If the model not accurate, the system performance is compromised.
CHAPTER 8. CONCLUSIONS AND FUTURE PERSPECTIVES

8.1 Conclusions

In this work, the author has attempted to combine nonlinear system models with data-driven optimization methods to form different MGDD algorithms. The goal is to achieve efficient optimization and control of nonlinear systems related to IC engine applications.

The main contribution of this work can be summarized as follows:

When significant and time-varying measurement delay presences in the system, available system model with satisfactory accuracy can be applied to predict the system performance and to compensate the measurement delay. The model becomes a soft sensor and the prompt feedback provided by the soft sensor can speed up the convergence rate of the data-driven optimization.

When measurement delay is negligible in the system, the available system model can provide feedforward initialization for the ESC optimization. This forms a model-guided ESC structure. The guidance provided by the model assists the ESC and leads the ESC to a faster convergence speed with more predictable transients as compared to the conventional ESC optimization. If the model accuracy is poor, online model calibration can be added to the model-guided ESC, to improve the model prediction on-the-fly.

Lastly, an MGDD predictive control approach is proposed for the actuation of an ETB system. Instead of a first-principle model, a data-driven Pseudo-Wiener model is selected to simulate the ETB behavior. Compared to the two conventional control strategies, i.e. the PID and the SMC, the MGDD predictive control algorithm has shown obvious advantages for the ETB control. Improved system transient responses and flexible system performance regulations by changing the design of the performance functions can be achieved using the MGDD predictive control algorithms. However, an MGDD predictive control algorithm also put requirements on the accuracy of the system model. The improvement of the integrity of the system model often requires extra efforts.
8.2 Future Perspectives

So far, the MGDD optimization and control algorithms have only been applied to problems such as the optimization of the IC engine operation under steady-states and the control of an SISO system like the ETB.

The goal of the MGDD algorithm development in the future can be the optimization/control of a Multi-Input-Multi-Output system during transient conditions. Sufficient conditions that can ensure the MGDD algorithms to work with performance guarantees, such as stability and robustness, are also in need.
REFERENCES


141


### VITA AUCTORIS

<table>
<thead>
<tr>
<th>NAME:</th>
<th>Qingyuan Tan</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLACE OF BIRTH:</td>
<td>Guilin, Guangxi, China</td>
</tr>
<tr>
<td>YEAR OF BIRTH:</td>
<td>1987</td>
</tr>
<tr>
<td>EDUCATION:</td>
<td>Shanghai Jiaotong University, B.Eng., Shanghai, China, 2010</td>
</tr>
<tr>
<td></td>
<td>University of Toronto, M.A.Sc., Toronto, ON, Canada, 2012</td>
</tr>
<tr>
<td></td>
<td>University of Windsor, Ph.D., Windsor, ON, Canada, 2018</td>
</tr>
</tbody>
</table>