MODEL BASED CONTROL AND EFFICIENT CALIBRATION FOR CRANK-TO-RUN TRANSITION IN SI ENGINES

DISSERTATION

Presented in Partial Fulfillment of the Requirements for the Degree Doctor of Philosophy in the Graduate School of The Ohio State University

By

Qi Ma, B.S., M.S.

* * * * *

The Ohio State University

2005

Dissertation Committee:
Stephen Yurkovich, Adviser
Giorgio Rizzoni
Yann G. Guezennec

Approved by

______________________________
Adviser
Electrical & Computer Engineering Graduate Program
ABSTRACT

The latest emission regulations demand drastic reduction of tailpipe hydrocarbon emission of Spark Ignited, Port Fuel Injected (SI PFI) automotive engines during crank-to-run transition. This results in sharply increased calibration effort and associated cost based on the current control architecture of production engines, during this extremely fast transient operating regime under the constraints of substantial emission reduction and good engine startability.

A new approach using model based control is proposed for this problem. The engine start behavior and overall fuel dynamics characterization during crank-to-run transition are thoroughly investigated. A scheduled in-cylinder fresh air charge predictor is constructed. A nonlinear input correction function, invoked prior to engaging fuel dynamics control, is developed. In order to cover a wide range of engine coolant temperatures by means of scheduling, the linear spline modeling technique is applied to air and fuel dynamics modeling, identification and control design. A new criteria, from the class of subspace methods, is introduced to evaluate system order and model quality.

Finally, a predictive fuel dynamics control scheme is realized to overcome individual cylinder fuel dynamics effect by systematically combining the scheduled in-cylinder fresh air charge predictor, the direct inversion of a fuel dynamics model and an inverse correction function. By means of an intelligent mode scheduling of the
in-cylinder fresh air charge predictor with misfire and poor-start detection, a fault tolerant predictive fuel dynamics control results. It has been demonstrated that the calibration effort of start fuel control during engine start and crank-to-run transition is reduced significantly for production inline-4 cylinder engines. In addition to solving this practical problem using model based control, this dissertation research also raises several theoretical questions worthy of further research.

In summary, this dissertation research makes following contributions: 1) individual in-cylinder fresh air charge prediction and individual cylinder fuel dynamics compensation; 2) accommodated misfire and poor-start; 3) significantly reduced calibration effort; 4) introduction of the Linear Parameter Varying Linear Splines (LPV-LSP) technique for gain scheduled modeling, identification and control design; and 5) introduction of a novel method to identify ARMA models using subspace methods.
Dedicated to Linqu and Mulan
ACKNOWLEDGMENTS

My greatest thanks go to my advisor, Prof. Stephen Yurkovich, for constantly providing me valuable advise during the course of my four year period of Ph.d study. Under his full scale tutoring and guidance, I have been enlightened, improved and polished in almost all aspects. What I benefit from him will guide me throughout the rest of my life.

Secondly, I would like to give special thanks to Dr. Kenneth P. Dudek for his advising, inspiration and encouragement during our four-year period of collaboration between CAR and GM Powertrain. His unique and keen understanding in powertrain control taught me how to be a qualified control engineer.

I also wish to thank Prof. Giorgio Rizzoni for supplying me a priceless opportunity to initiate my graduate study at The Ohio State University. Under his supervising and strong recommendation, I was be able to complete my M.S. and follow with Prof. Stephen Yurkovich to pursue my Ph.D. I also wish to thank Prof. Yann G. Guezennec for his guidance during my years at CAR, and for agreeing to be a reader on my dissertation.

In addition, I wish to thank my family for backing me up with strong spiritual support and standing by my side during the whole course of my graduate study.

Lastly, I would like acknowledge General Motors for funding the research project and supporting me to complete the dissertation.
VITA

1996 ........................................B.S. EE, USTB, P. R. China
2002 ........................................M.S. ME, OSU, Columbus, Ohio, USA
2002-present ............................Ph.d EE, OSU, Columbus, Ohio, USA

PUBLICATIONS

Research Publications


FIELDS OF STUDY

Major Field: Electrical & Computer Engineering Graduate Program

Studies in:

Topic 1  Control Theory, Automotive Powertrain Control
Topic 2  Computer Engineering
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>ii</td>
</tr>
<tr>
<td>Dedication</td>
<td>iv</td>
</tr>
<tr>
<td>Acknowledgments</td>
<td>v</td>
</tr>
<tr>
<td>Vita</td>
<td>vi</td>
</tr>
<tr>
<td>List of Tables</td>
<td>x</td>
</tr>
<tr>
<td>List of Figures</td>
<td>xi</td>
</tr>
<tr>
<td>Chapters:</td>
<td></td>
</tr>
<tr>
<td>1. Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Problem Statement and Background</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Historical View and Literature Search</td>
<td>7</td>
</tr>
<tr>
<td>1.2.1 Hardware Based Solutions</td>
<td>8</td>
</tr>
<tr>
<td>1.2.2 Algorithmic Solutions</td>
<td>10</td>
</tr>
<tr>
<td>1.2.3 Simulation and Experimental Studies</td>
<td>14</td>
</tr>
<tr>
<td>1.2.4 Model based Engine Control</td>
<td>18</td>
</tr>
<tr>
<td>1.3 Summary and Outline of Dissertation</td>
<td>23</td>
</tr>
<tr>
<td>2. Modeling</td>
<td>25</td>
</tr>
<tr>
<td>2.1 Linear Splines</td>
<td>26</td>
</tr>
<tr>
<td>2.1.1 Properties of Linear Splines</td>
<td>29</td>
</tr>
<tr>
<td>2.1.2 Implementation Aspects</td>
<td>31</td>
</tr>
<tr>
<td>2.2 Models in Gain Scheduling</td>
<td>33</td>
</tr>
<tr>
<td>2.2.1 LPV in Gain Scheduling</td>
<td>34</td>
</tr>
</tbody>
</table>
2.2.2 LPV-LSP in Gain Scheduling .......................... 37
2.3 Event Based In-Cylinder Fresh Air Charge Models .......... 40
  2.3.1 Choices of In-Cylinder Fresh Air Charge Model .......... 41
  2.3.2 Crank GPO Model .................................. 43
  2.3.3 Crank-to-Run GPO Model ............................... 45
  2.3.4 Run GPO Model ...................................... 47
2.4 Cycle Based Overall Fuel Dynamics Model .................... 48
  2.4.1 Utilized Fuel Fraction ................................ 49
  2.4.2 Nominal Fuel Dynamics ................................. 56
2.5 Summary .................................................... 58

3. Predictive Fuel Dynamics Control .............................. 59
  3.1 Architecture of Predictive Fuel Dynamics Control .......... 59
  3.2 Design Evolution and Lessons Learned ....................... 62
  3.3 Three Step Ahead GPO Prediction For I-4 Engines ............ 76
    3.3.1 Crank GPO Predictor ................................ 78
    3.3.2 Crank-to-Run GPO Predictor ........................... 80
    3.3.3 Run GPO Predictor ................................... 81
    3.3.4 Misfire, Poor-Start GPO Predictor ....................... 81
    3.3.5 Transition Rules .................................... 82
  3.4 Nominal Fuel Dynamics Compensator .......................... 85
  3.5 Inverse Utilized Fuel Fraction ............................... 86
  3.6 Complete Predictive Fuel Dynamics Control ................. 89
    3.6.1 Logic for GPO Assignment ............................. 90
    3.6.2 Fuel Dynamics Control Initial Condition Setup .......... 92
  3.7 Summary .................................................... 93

4. Identification Methodology for Calibration .................... 94
  4.1 Common Tools in Identification ............................ 96
  4.2 Question and Ambiguity ................................... 99
  4.3 Angle View of Linear Dynamic System ......................... 103
  4.4 Examples for Angle between Subspaces ....................... 108
  4.5 Subspace Methods for Linear Model Identification .......... 110
  4.6 Unsolved Issues ......................................... 112
  4.7 Summary .................................................... 113

5. Calibration of Fuel Dynamics Control ......................... 114
  5.1 Calibration of Cylinder Air Rate Predictors ................. 115
    5.1.1 Crank GPO Predictor .................................. 115
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1.2</td>
<td>Crank-to-Run GPO Predictor</td>
<td>117</td>
</tr>
<tr>
<td>5.1.3</td>
<td>Run GPO Predictor</td>
<td>118</td>
</tr>
<tr>
<td>5.1.4</td>
<td>GPO Filter</td>
<td>118</td>
</tr>
<tr>
<td>5.1.5</td>
<td>Misfire and Poor-Start GPO Predictors</td>
<td>119</td>
</tr>
<tr>
<td>5.1.6</td>
<td>Calibration of Transition Rules</td>
<td>122</td>
</tr>
<tr>
<td>5.2</td>
<td>Calibration of Fuel Dynamics Compensator</td>
<td>125</td>
</tr>
<tr>
<td>5.2.1</td>
<td>Law of Mass Conservation</td>
<td>126</td>
</tr>
<tr>
<td>5.2.2</td>
<td>Calibration of $UFF_{20}(ECT)$</td>
<td>128</td>
</tr>
<tr>
<td>5.2.3</td>
<td>Calibration of UFF and Nominal Fuel Dynamics Model</td>
<td>129</td>
</tr>
<tr>
<td>5.2.4</td>
<td>Calibration of Inverse UFF Function</td>
<td>134</td>
</tr>
<tr>
<td>5.3</td>
<td>Performance of Control</td>
<td>135</td>
</tr>
<tr>
<td>5.4</td>
<td>Summary</td>
<td>146</td>
</tr>
<tr>
<td>6</td>
<td>Conclusion</td>
<td>147</td>
</tr>
<tr>
<td>6.1</td>
<td>Summary of Dissertation</td>
<td>147</td>
</tr>
<tr>
<td>6.2</td>
<td>Contributions</td>
<td>150</td>
</tr>
</tbody>
</table>

Appendices:

A. Nomenclature | 153

Bibliography | 156
LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>57</td>
</tr>
<tr>
<td>3.1</td>
<td>62</td>
</tr>
<tr>
<td>5.1</td>
<td>125</td>
</tr>
<tr>
<td>5.2</td>
<td>127</td>
</tr>
<tr>
<td>5.3</td>
<td>130</td>
</tr>
<tr>
<td>5.4</td>
<td>135</td>
</tr>
<tr>
<td>5.5</td>
<td>136</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>FTP75 tailpipe HC emission [37] and Engine-Out HC emission [38]</td>
<td>3</td>
</tr>
<tr>
<td>1.2</td>
<td>Two sequential phases in algorithmic solutions</td>
<td>10</td>
</tr>
<tr>
<td>2.1</td>
<td>Concept of linear splines</td>
<td>27</td>
</tr>
<tr>
<td>2.2</td>
<td>Definitions of dummy variables</td>
<td>28</td>
</tr>
<tr>
<td>2.3</td>
<td>2-dimension linear spline approximation (<a href="http://www.ibiblio.org/e-notes/Splines/Inter.htm">http://www.ibiblio.org/e-notes/Splines/Inter.htm</a>)</td>
<td>29</td>
</tr>
<tr>
<td>2.4</td>
<td>Design procedure for gain scheduled control</td>
<td>37</td>
</tr>
<tr>
<td>2.5</td>
<td>Identification approach for gain scheduled control</td>
<td>38</td>
</tr>
<tr>
<td>2.6</td>
<td>Three distinct phases during engine start and crank-to-run transition</td>
<td>41</td>
</tr>
<tr>
<td>2.7</td>
<td>Two choices of the in-cylinder fresh air charge model</td>
<td>42</td>
</tr>
<tr>
<td>2.8</td>
<td>The intake manifold pressure dynamics after key-on of a production inline-4 cylinder engine</td>
<td>44</td>
</tr>
<tr>
<td>2.9</td>
<td>MAP and GPO decay during crank-to-run transition</td>
<td>46</td>
</tr>
<tr>
<td>2.10</td>
<td>Overall fuel dynamics model</td>
<td>48</td>
</tr>
<tr>
<td>2.11</td>
<td>Raw Injected Fuel Mass and Measured Burned Fuel Mass at -15° C</td>
<td>49</td>
</tr>
<tr>
<td>2.12</td>
<td>RINJ vs. MBFM at four different ECTs</td>
<td>50</td>
</tr>
</tbody>
</table>
CHAPTER 1

INTRODUCTION

1.1 Problem Statement and Background

Obtaining a robust start for a gasoline Spark Ignited (SI) Port Fuel Injected (PFI) automotive engine, together with minimum Hydrocarbon (HC) emission, is extremely difficult because it is a coupled, multi-objective control problem, with the objective to minimize tailpipe HC emission and, meanwhile, to maintain good startability for engine start conditions.

Among several emission species, the tailpipe HC emission generated during engine cold starts is the most significant emission compound subjected to emission regulations, and hence the main efforts for reducing tailpipe emissions are focused on engine-out HC emission reduction during this short operating condition. Due to the direct linkage between high tailpipe HC emission and the quality of the typical (production) cold start fuel control strategy, as well as the efficiency of after-treatment systems, automotive manufactures and research institutes have put significant efforts into improving the hardware conversion efficiency, and refinement of the control strategy.
To evaluate the tailpipe emissions of passenger vehicles, the Federal Test Procedures (FTP) are usually used under the emission ambient condition\(^1\). The vehicle speed profile used in the FTP-75 is given in Figure 1.1.a. The tailpipe HC emission of a production passenger car in the first sampling bag takes up to 76% of the total HC emission over the whole driving cycle in the Tier I emission standard and up to 85% in the Ultra Low Emission Vehicle (ULEV) standard. The pre-catalytic and post-catalytic HC emission during the first 3 seconds of engine operation of a production passenger car at the emission ambient condition is shown in Figure 1.1.b. As shown in both figures, a large amount of HC is emitted at the beginning of the engine operation from Key-On through the first 3 seconds of engine operation due to bad fuel preparation and inactiveness of the Three Way Catalytic converter (TWC). Soon after the first 3 seconds, the HC emission is significantly reduced. After another 10-20 seconds (not shown), the TWC has undergone full “light-up” and the total HC emission during the rest of the driving cycle is a small fraction of the total HC emission of the whole driving cycle. Therefore, it is evident that the key to meeting the latest emission standards is to reduce engine-out HC emission during the first few seconds of engine operation.

As the efforts have been put into the engine-out HC emission reduction for engine start conditions, a significant problem, engine startability, arises simultaneously. The well-known conclusion about this conflict is that the good engine startability and the engine-out HC reduction are two conflicting control objectives. As the emission target is lowered, the tolerance to guarantee the engine startability becomes narrower. As a consequence, a precisely controlled engine start is mandatory so as to achieve the

\(^1\)The emission ambient condition is the condition where the testing vehicle soaks at room temperature (about 25°C) over 24 hours before the emission tests.
minimal engine-out HC emission and to maintain good engine startability at the same
time. As reported by Cheng [38], any abnormal combustion events (such as misfire
or weak fire) in one engine cold start scenario at the emission ambient condition can
lead to an early failure in the emission tests at the ULEV standard. Hence, in order
to achieve the minimum engine-out HC mission, a consistent and robust engine start
must be achieved as a very first step. The task to minimize the HC emission can
only be executed after the engine is robustly started. When the Engine Coolant
Temperature (ECT) becomes lower (say, less than the emission ambient conditions),
good engine startability becomes even more mandatory; although emission tests are
not conducted at such temperatures, customers’ satisfaction and public concern play
vital roles instead.

To achieve consistent and robust engine start together with the minimal engine-
out HC emission is an extremely difficult control task; the difficulties lie in the so-
plicated physical nature of the plant, measurement accuracy and the availability
of a simple and accurate mathematical description of the problem. Some dominant
difficulties are:
1. Start operation of automotive engines is an extremely transient process. It begins from the initial moving of the crankshaft (driven by the starter system) until the engine speed passes the RPM flare and runs stable at either the warm-up speed or the idle speed. The total number of engine combustion events in each successful start is about 80-100 and the duration of the whole start process is approximately 3 seconds. This corresponds to about 20 engine cycles for each cylinder of a multiple cylinder engine. The signature characteristic of engine start is that none of the engine variables, such as engine RPM, Manifold Absolute Pressure (MAP), Cylinder Wall Temperature (CWT), Intake Valve Temperature (IVT), engine friction, gas residual fraction, fuel residual, etc, are at steady state conditions. Moreover, the angular velocity transition of the crankshaft during the crank-to-run transition is the largest in the life of an IC engine, where one combustion event can accelerate the crankshaft speed from 200 RPM to 800 RPM almost instantly.

2. Due to the application of the Port Fuel Injection (PFI) system and the Closed Valve Injection (CVI) configuration, an obvious control causality problem appears: fuel is delivered long before air is breathed into cylinders. How does the control calculate the air quantity beforehand? Although model based prediction schemes have been successfully implemented on running engines [8], these schemes are not guaranteed to work during the crank-to-run transition where large air and RPM transients occur. More importantly, due to the fast (transient) time period at the initial stage of engine operations, feedback control is not possible.
3. The physics behind this transient dynamic process is so profound that only limited understanding has been achieved to date. Well established knowledge about combustion, thermal and fluid dynamics at normal engine operating conditions is difficult to apply at engine start conditions.

4. Some stock measurements may not have enough resolution during engine starts, resulting in embedded system measurement error and consequent control error. In particular, the resolution available for the Equivalence Ratio (EQR) is not adequate for use in the first two engine cycles after first fire (crank-to-run).

5. High variability of air-fuel mixture and combustion qualities prevent us from establishing a sufficient accurate relationship among engine variables. To reduce such impact, a large amount of data is usually required. However, data richness directly links to the cost of testing and the period of control development. Therefore, high variability and data richness are two barriers for analysis and design. As a result, to identify a trustworthy model hidden inside the limited and noisy data is an extremely difficult task. Advanced tools in system identification might be needed.

Due to the above facts, development of the Engine Start Algorithm (ESA) and its calibration process in the past two decades have been based on “trial and error” open loop control methods. As a consequence, the quality of engine start is solely dependent on the amount of calibration effort. As such, the current production ESAs are a result of years of best engineering experience and calibration process refinement, not developed from any control or system theories. It is not so surprising that existing algorithms are very similar, because the current ESA and calibration
process are nothing but “putting a number in to make it work”. Therefore, many current ESAs, including the calibration process, have not incorporated engine dynamics appropriately and efficiently, but in an inconsistent, lengthy and costly way.

Many current ESAs are not model based; that is, because control theory is not employed, the algorithms are calibration intensive (lots of trial and error), and greatly dependent on calibrator skill. Equally important is the fact that there is no possibility for individual cylinder AFR control by such methods. The solution to such control cannot always guarantee consistent results, which is part of the reason for the presence of high HC emission at engine starts. Nevertheless, the corresponding calibration method is dictated by the nature of such control. These issues are amplified greatly in the event of misfire and weak fire. In current control schemes, there typically is no detection and amelioration scheme and the engine is left to recover if possible.

Although more understanding about the engine start behavior has been gleaned by control engineers and engine calibrators, such knowledge does not necessarily yield a more accurate mathematical description of the engine start behaviors; rather, it has resulted in a form of linguistic description and personal experience. Such understanding may help find solutions in the current control architecture but, as stated earlier, it involves tremendous calibration effort and is inherently burdened with human (calibrator) decision errors. In addition, the personal experience and knowledge are encrypted such that they cannot be easily transferred between engine testing/calibration engineers and researchers. It is not surprising that an experienced calibrator for V-8 engines cannot transfer his or her knowledge to V-6 engines, simply because the V-8 and V-6 engines are different enough as defined by the current control architecture. This is not a fault of the engineers but a fault of the current ESAs which restrict
the human intelligence by a low-intelligence control algorithm written in software. Moreover, automotive manufactures usually have several engine families and their variants. It is evident that to engineer the engine start based on the current ESAs is no longer an economical and consistent way to solve the engine cold start problem. Of course, it is also not an economical answer to the emission regulations.

Motivated by the problems as outlined above, this research tackles this difficult problem by means of model based control. The ultimate goal of this research is to reduce the calibration effort and to reduce the chance of abnormal engine starts. To date, this overall problem has not been approached from a control-theoretic viewpoint, which has led to current “calibration-intensive” production algorithms.

As successfully proved in vast industrial applications, model based control, indeed, can incorporate the system dynamics in an effective and systematical way, rendering a new corresponding calibration process. Hence, this research also serves as an exploration of a novel calibration methodology in a realistic practical control problem: What is the calibration process resulting from model based control? By answering this question in this research, a new calibration methodology, “data in, calibration/control out”, will result from this dissertation.

1.2 Historical View and Literature Search

Since the time IC engines were invented and widely used on passenger vehicles, the problems of engine startability and tail pipe emission have been present. In the past, engine startability was secured by sacrificing emissions. In order to pass the latest emission tests successfully, the ESAs must fulfill two tasks simultaneously: 1)
achieve consistent and robust engine start each time \(^2\); 2) achieve a minimal amount of HC emission. In the open literature, however, those two objectives seem to be poorly combined; the bulk of the literature focuses on the solutions after the engine is robustly started, with little attention paid to the design of the ESAs. Some researchers have tried to characterize engine start behaviors from several viewpoints, but complete solutions to the whole engine start problem have not been revealed. In spite of the fact that production ESAs are not available in the public domain, model based control for fast engine warm-up and catalyst light-off have appeared frequently in the open literature.

In addition to the sophisticated physical nature of the problem being faced in this research, several theoretical questions must be answered: 1) Is it possible to identify a model robustly with an appropriate order and manageable complexity given a limited amount of noisy data? 2) What is the optimality measure of the model being identified: prediction error or other optimality measures? 3) Does a model structure exist to cover a wide range of engine coolant temperatures by means of scheduling?

By carrying those questions in mind, an extensive literature search is in the following four subsections: 1) Hardware based solutions; 2) Algorithmic solutions; 3) Simulation and experimental studies; and 4) Gain scheduled control.

1.2.1 Hardware Based Solutions

One popular choice adopted by automotive manufactures to answer the stringent emission regulations lies in hardware based solutions. The main body of the literature found so far focuses on the development of the emission conversion systems

\(^2\)The term consistent and robust engine start means that engine fires robustly due to the first cycle fuel shots and consistently fires without misfire and weak fire in the remaining engine cycles.
to meet ULEV/SULEV/ZLEV\textsuperscript{3} standards. Such development is based on modifying or adding extra hardware to advance the catalytic converter light-off, to store the emission for later conversion, and to improve fuel vaporization efficiency prior to combustion. Therefore, the corresponding hardware designs fall into four main categories based on their functionalities: 1) thin wall design of the catalytic converter; 2) external heated catalysts either by electrical power or by after burner; 3) HC absorbers; 4) fuel atomization technologies. In [35], a thin wall design reduced the thermal inertia of the catalytic converter so as to achieve fast light-off. Nissan [30, 31, 43, 43, 50], Honda [18, 41, 46] and others have successfully used electrically heated catalysts with secondary air supplies to meet ULEV standards. On the other hand, Nissan [30] has met SULEV and On Board Diagnostics (OBD) II requirements with a 2-stage HC trap and an electronically actuated swirl control valve, while in other cases installing in-line HC absorbers into existing converter systems. Some [15, 32] have used afterburners to heat the catalytic converter during engine cold starts. Air-assisted fueling technology [13, 17, 44] is the most common fueling system recommended, although there may be a difference in the preferred injection timing (open-valve or closed-valve) and location of the fuel spray [13]. Honda [29] also used a different intake valves opening timing to increase the swirl in intake stroke so as to improve the fuel vaporization. Fuel atomization is improved due to the pre-vaporized fuel, thereby avoiding the need for a rich mixture at start-up. Hence, port and cylinder wall-wetting and HC emissions are reduced.

The hardware based solutions found so far only focus on modifying or adding extra hardware to reduce the HC emission after the engine is robustly started. It

\textsuperscript{3}ULEV, SULEV and ZLEV refers to Ultra Low Emission Vehicle (ULEV), Super Ultra Low Emission Vehicle (SULEV) and Zero Level Emission Vehicle (ZLEV), respectively.
is clear that a good hardware platform, such as a better performance of TWC, can boost the performance of emission reduction to a higher level, but in order to fully take advantage of the extra degree of freedom for emission control, the corresponding control strategy is very crucial to fully utilize the extra hardware benefits.

1.2.2 Algorithmic Solutions

Another approach to tackle the engine start problem is to adopt algorithmic solutions based on a given hardware platform. The freedom of such an approach is gained by intelligently “wiggling” the control inputs to achieve the same objectives. There are two sequential phases in implementation: 1) Crank-to-Run phase where the ESAs take place; and 2) warm-up phase where the Post Start Algorithms (PAS) come into play. A timing chart is given in Figure 1.2 to show the relationship of the occurrence of those two phases. Although most of the publications in the open literature may have titles containing words such as “...engine start emission reduction...”, actually they typically focus on the second phase rather than the first phase. It is not surprising that the development of ESAs is largely ignored in the public domain.

![Figure 1.2: Two sequential phases in algorithmic solutions](image-url)
Engine Start Algorithms

Here, the term Engine Start Algorithms refers to the algorithms that exclusively focus on the first few seconds of engine operations, with the main goal to start up the engine. Although the ESA seems to be ignored in the public domain, they play an extremely important role to initialize the engine operation. More importantly, algorithm quality directly affects the engine-out HC emission and influences the starting time and the acoustic and tactile feelings from induced vibration. In Figure 1.2, the regime of when the ESAs usually begin and terminate can easily be seen. Generally speaking, the ESA can be affected by the specific engine platform and the position sensing systems, but it is independent enough to be generalized.

As shown in Figure 1.2, the engine start algorithm kicks in only after engine synchronization is established. Earlier than that moment, it seems impossible to shorten the engine starting time. However, if the previous shut down position is known within a reasonable degree of accuracy, the Engine Control Unit (ECU) is actually synchronized with the engine crankshaft all the time, and hence the fuel mass can be delivered earlier and therefore the engine can be started earlier. Moreover, the earlier delivery of fuel also helps fuel evaporation at colder temperatures (observed in [38]), simply because the injected fuel mass gets a longer time to vaporize. Not all engine manufacturers would like to incorporate such features because the accuracy of the previous shut down position depends on the number of teeth on the sensing wheel and the position sensing system. Corresponding error handling is somewhat problematic as well.
Once engine synchronization is achieved with or without the information of the previous shutdown position, batch and sequential fuel injection strategies are commonly used in many commercial applications [37]. The batch fuel injection strategy delivers fuel shots to either all cylinders or one bank of cylinders simultaneously, if a V-6 or V-8 engine is used. Based on the firing order, this scheme does not fully utilize the fuel injected onto the intake valve surface and it may cause open valve injection (OVJ), which usually causes more engine-out HC emission. It has been observed that the injected fuel masses to all cylinders are identical in a batch injection strategy. However, as the engine initiates the first combustion, different cylinders may need different fuel masses due to different intake Manifold Absolute Pressure (MAP) and RPM. This type of fueling strategy may have some contribution to the engine-out HC emission. On the other hand, the sequential fueling strategy is more advanced than the batch injection strategy because it only delivers fuel to the cylinder which may possibly generate the work; but it can take longer time to get the engine started.

Whether in batch injection strategy or in sequential fueling strategy, a common rule for the first engine cycle fueling is [4, 27, 37]: the amount of fuel delivered in the first engine cycle may be several times greater than stoichiometric fuel requirement when the engine is cold (< 50°C). Besides, there is an explicit control causality problem for fuel delivery on gasoline PFI engines: how is the fuel mass calculated for those cylinders who will make power on the RPM ramp, but need to be injected at cranking speed? So far, only Honda [46] has released partial information about their start strategy. In [46], an inverted first order τ-X [34] fuel dynamics model was used in mass production. It was also claimed that the first order τ-X fuel dynamics model is
adequate to describe fuel behavior at start operating conditions. However, one crucial piece of fuel dynamics, as observed by Cheng [38], was not reported.

Post Start Algorithms (PSAs)

After the crank-to-run phase is terminated (required to be very short), control is handed over to the warm-up phase. Compared to the ESAs, the time duration in the PSAs is much longer and the engine operation is much smoother with respect to $\Delta MAP$ and $\Delta RPM$ (that is, changes in MAP and RPM) per engine event. Therefore, tasks such as fast engine warm-up, idle speed regulation, and after-treatment system control can be carried out. Due to the focus of this research, which emphasizes the ESAs rather than PSAs, the relevant literature review presented for PSAs is at the understanding level.

From an algorithmic standpoint, the direct and effective way to reduce engine-out HC emission after engine starts is by bringing the ECT and exhaust temperature to normal operating conditions as quickly as possible. Based on this basic rule, the engine is controlled in such a way so as not to generate the work but to generate a large heating profile to heat up the cylinder wall and after-treatment system. This is usually done by retarding the spark advance with wider throttle openings [46] in lean operations. As implemented by Honda [18, 41, 46] and Volvo [1], controlled Variable Valve Actuation (VVA) or Variable Valve Timing (VVT) may also help to improve combustion efficiency by supplying swirling airflow. As such, feedforward control is almost unavoidable. In order to further speed up the light-up of catalytic converter, the secondary fresh air injection [46] is added to supply an extra heating profile by exhaust fuel oxidation together with the large heat profile by the engine itself. This requires the engine to be operated in an extremely rich condition to guarantee the
complete oxidation in exhaust manifold. Those control methods can advance the light-off time of the catalytic converter within about 20 seconds. After adding an external electric heating source to the catalytic converter, the light-off time may be in the 10 second range [41]. Similar results can be achieved by an exhaust burner as well [50].

**Summary of Hardware and Algorithmic Solutions**

Due to the limited information about the ESAs in the public domain, some engineering experience and common knowledge are compiled in Subsection 1.2.2 without too many references. As stated earlier, the research reported here-in may serve as a pioneer work to unveil the control problem and perhaps supply a novel solution to this short and important engine operation regime. In contrast, approaches covered by the hardware solutions and the PSAs are abundant in the open literature, and the most representative approaches are collected in Subsection 1.2.2.

**1.2.3 Simulation and Experimental Studies**

The automotive industry and research institutes have continued their efforts in studying the fundamental physics behind the problem during engine starts. In the last two decades, the efforts have been intensified due to environmental and customers’ acceptance issues. Loosely speaking, the study can be classified by two directions. One direction stems from the governing equations of the whole process using advanced computation tools which may involve thermal fluid dynamics and combustion theories. For instance, the studies of fuel properties/composition, Computational Fluid Dynamics (CFD) and combustion simulation are benchmark examples in this direction. The other direction emphasizes the experimental exploration, observation,
explanation and verification of the relationship among the engine variables during engine starts. Such direction is, more or less, toward system identification or multivariate correlation approaches which can effectively find the mathematical relationship without too much detail and exact physical insight of the plant. The recent works done by MIT [19, 37] are good examples to illustrate the effectiveness of such an approach.

Automotive Fuel Properties Study and Flow/Combustion Simulation

The foremost difficulty in engine start problems is the complicated evaporation characteristics of the sophisticated composition of automotive fuel. When the fuel spray is mixed with fresh air inside the intake manifold and runners, near the intake port, and inside the cylinders, the contribution of each individual fuel component to the in-cylinder Air Fuel Ratio (AFR) becomes almost impossible to decode. Researchers [14, 45, 53] have tried to characterize vaporization property and have made approximation in laboratory environments and used the simplified manageable properties of the automotive fuel to explain, interpolate, simulate and eventually use in control [2, 52]. The benchmark example of such approximation found so far is to extrapolate the commanded EQR out of the distillation curve of gasoline. Hallett [14, 45] approximated the gasoline composition via a few dominant chemical compounds, without verification; these results were used for years [2]. Despite the weaknesses of the approximation made by Hallett, it facilitates the use of 2-D and 3-D computer simulation to visualize the “reality” of the vaporization and combustion process [2, 53]. In another extreme case, the fuel composition is assumed to be a single chemical component as used in original KIVA code [2], and the simulation results agree very well with laboratory results. However, there is no such good fortune
to have a single component automotive fuel at gas stations; otherwise many problems at engine start would not appear. It is known that automotive fuel composition is able to be specified by a finite number of components, but such a specification is very complicated [2, 14, 45, 53]. If the composition is simplified by more than 10 components, the CFD study is no longer useful to us because it is impractical to give the solution within a reasonable time due to the limited power of computation tools. To this point, the complication in fuel composition forbids us from simulating the fuel properties using only a few simple fuel components in a CFD simulation environment.

The second difficulty of analyzing fuel properties is that the operating conditions of automotive engines are so unique that any experimental setup cannot duplicate them in the laboratory and computer environments. So far, the current fuel property study and flow/combustion simulation are only restricted at a fundamental research level and are not beneficial to control design in production.

**Work on C1-C2**

In this section, work which has appeared on the “C1-C2” problem\(^4\) is discussed. Many researchers have attempted to characterize the fuel life inside the cylinders by means of the direct visualization method, photographic or chronographic methods and laser technology to “see” the fuel puddles/droplets/impingement. However, most experimental setups are based on single cylinder engines or modified from production engines [27, 37]. The observation and conclusion from such experimental apparatus are usually qualitative rather than quantitative. Therefore, it is difficult to extract the control law to combine those observations and conclusions systematically. But

\(^4\)In this dissertation, C1-C2 refers to the first and second engine cycles where cycle #1 starts from the supposed first robust firing.
some recent publications [4, 10, 19] have revealed part of the profound characteristics of the fuel dynamics from the viewpoint of control design. This has never been achieved because of the limited success in measurement technology in the past. The measurement technology used in [4, 19] is to measure or infer the in-cylinder AFR by means of FFID, Fast CO/CO$_2$ and WAFR measurements. Especially, the FFID measurement used in [19] can give a direct indication of the impact of control design. Such indication is very valuable to this research. More importantly, the study reported in [4, 19] is the first of its kind seen in the open literature, to conduct independent engine start tests to study fuel behavior and investigate engine startability by experiments; and it introduced the concept of fuel loss. Because of the significance of this study, the key results in [10, 19] are summarized below:

- **Fuel utilization efficiency**

  “The more you inject, the better chance of robust fire, but with a diminishing return” is the key concept of using fuel utilization efficiency. Fuel utilization efficiency decreases with respect to decreasing ECT. At lower ECTs, such as at $-5^\circ$C, fuel utilization efficiency can reach as low as 10%. This is part of the answer of why much more fuel mass should be injected than required for the stoichiometric condition, so as to make the first robust combustion at engine starts.

- **Influential variables of in-cylinder vapor fuel mass**

  Engine MAP and RPM both affect in-cylinder vapor fuel mass at the first engine cycle. Lower MAPs help to gain more in-cylinder fuel vapor mass. Higher engine RPMs can reduce the in-cylinder vapor fuel mass. When combining MAP and RPM effects together in a robust engine start, the RPM effect dominates.
Therefore, in-cylinder vapor fuel mass, in general, decreases along with the engine RPM rise in a robust start.

- **Effect of residual fuel on the second engine cycle**
  After the first engine cycle is finished by supplying sufficient fuel mass, exhaust residual gas appears in the second engine cycle. This is equivalent to reducing the in-cylinder fresh air charge. Secondly, the previous injected fuel mass is left over and survives in the second engine cycle. Determining the amount of fuel utilized becomes more complicated due to the fact that there is no simple way to access this information because of coupling effects.

- **Deficient EQR measurement by WAFR sensor**
  In addition to the fact that only a limited number of data points can be collected in a successful start, another severe problem of the measurement is the deficient EQR measurement of the Wide range Air to Fuel Ratio sensors (WAFR) during engine Cycle #1 (C1) and Cycle #2 (C2). The WAFR measurement of the first two engine cycles are extremely unreliable due to the mixing effect and the fresh air residual effect.

### 1.2.4 Model based Engine Control

In the past 50 years, a myriad of engine models and subsystem models have been developed. They may serve different purposes and may vary with different degrees of accuracy and complexity. Most of them are informative in that they attempt to characterize how complicated the engine system can be in mathematical sense, but not from a viewpoint of identification and control. Therefore, the engine’s inherent identifiability will always be a problem, given accessible measurements. Not all of
models can be used in control design. In [34, 51], the control oriented engine modeling was systematically reviewed and modified for control design purposes. Although the corresponding identification scheme is not fully covered, it supplied a solid framework of engine modeling in this research. In addition to being able to design the control at a given engine coolant temperature as seen in [34, 51], a modeling architecture which can systematically incorporate the temperature effect into the model as well as control to cover a wide range of engine coolant temperatures is expected.

Control Oriented Models

Given the nature of the core problems in engine starts, control oriented models for fuel dynamics and cylinder air rate estimation are essential. Because of the difficulties inherent to nonlinear model identification, discrete linear models are usually chosen, perhaps appended with some soft nonlinearities. Such consideration is based on well known merits of the linear model identification and control implementation.

Fuel dynamics modeling: The $\tau$-$X$ model [34], multi-puddle fuel dynamics model [40] and fuel utilization efficiency concept [19] are top candidates in this research. But higher order models as shown in [40] require validity of extra conditions. In [40], the model formulation does not originate from experiments, but from an envisioning of the fuel dynamics without concrete data support; also it is surprising that coupling between fuel puddles is not accounted for. Therefore, the identifiability of such a model is questionable. So far, the above candidates are capable in describing the input/output relationship of the fuel dynamics given with standard measurements. However, there is a larger class of linear systems, state space models, suitable for the fuel dynamics. Although there
is no single report in the public domain to describe the parametrization based on the input/output relationship about the fuel dynamics, such a modeling approach is better equipped to handle varieties of modeling problems [33]. No matter which kind of model is selected in this research (input/output transfer function or state space model), the ECT effect must be considered into the model structure. So far, only Honda [46] showed a reasonable form of how ECT enters into the fuel dynamics model.

**Cylinder air rate estimation:** Although often documented in the public domain, various intake manifold pressure dynamics have been developed. Some models can capture the pressure distribution inside the manifold and runners. However, from a control standpoint, the air dynamics must be captured in a non-casual fashion in the control design. Note that the fuel is delivered before the intake stroke starts on a PFI gasoline engine; the fuel quantity has to be calculated based on the forecasted in-cylinder air rate. Such a non-casual control scheme dictates that the prediction error correction method for air rate prediction [8, 11] must be used. Therefore, the Kalman predictor [26] must be at the center of the picture of model based control in this research.

**Gain Scheduling**

Among several suitable methodologies for nonlinear automotive engine control design, gain scheduling or gain scheduled control is the most rigorously developed control methodology. Shamma [36] provided an excellent work in reviewing the recent activities in this area and provided several guaranteed properties. Although it is much harder to analyze the stability and robustness of the gain scheduled control, it is an
effective control design when a few conditions are met. One such well known condition is that, if the scheduling variable is much slower than the dominating dynamics, gain scheduled control is “guaranteed” to be robust. An open loop scheduled control architecture, such as that considered in this research, cannot directly use the fruits from [36]. However, modeling techniques such as the Quasi Linear Parameter Varying (QLPV) methods are worth attention and benefit the fuel dynamics modeling in this research.

Based on the operating conditions at engine starts, the ECT is relatively fixed and it is safe to assume that the temperature of the cylinder wall is not varying too much, even after several combustion events have occurred. Therefore, the linear gain scheduled controller (scheduled by the ECT) and linear parameter varying controller (varying with respect to the ECT) are promising techniques for the problem. In addition, the success of the spline curve fitting and smoothing technique [25, 39] achieved in recent years sheds light on the linear approximation of nonlinear systems. The spline method is an effective modeling tool suitable for linear scheduled modeling, identification and control. In short, any nonlinear system can be approximated by a group of linear systems linked together by piecewise or continuous operating regimes. Such partitions of the operating regime can be viewed as a slope adding modeling scheme. Compared with the QLPV representation of a nonlinear system [36], the linear spline model has much more freedom and flexibility. This is especially true when the mathematical model of the nonlinear system is completely unknown and the QLPV modeling is not possible, wherein the linear spline model can still achieve a reasonable approximation with a specified degree of accuracy. More importantly,
the identification scheme for the linear spline models is very simple. This technique will be discussed in more detail in a subsequence chapter.

**System Identification**

Although the topic of system identification is reviewed very late in this chapter, system identification, indeed, plays a key role in model based control design. Without appropriate system identification tools, it is not possible to trace down the true model structure and corresponding parameters from footprints left by the system, such as input/output data. Though system identification covers broad topics in estimation, signal processing, communication and control [23, 26], if the nonparametric identification methods (such as frequency domain methods) and nonlinear model identification methods are ruled out, the class of methods narrows for linear model identification. Loosely speaking, the methods for linear model identification can be classified by three categories: I) \( AX \approx b \) problem; II) \( AX \approx BY \) problem; III) \( AX \approx 0 \) problem, where \( A, B \) are matrices with more rows than columns and \( b \) is a column vector. The Least Squares (LS) algorithm and Instrumental Variables (IV) methods can handle the problems in category I [23]; subspace methods [33] handle the problems in category II [23, 48]; and the Total Least Squares method (TLS) handles the problems in the last category [47]. Due to the strong conditions required in the IV and TLS methods, it is common practice to convert identification problems into the Auto Regressive with eXtra input (ARX) model identification problem. Though such a practice may not always be justifiable in a statistical sense, it is often accurate enough and certainly easy to implement. In certain situations, such conversion is not a bad choice. As such, there are many choices to make the conversion. In doing so, some choices lose their meaning in either a mathematical or physical sense. For
example, in many control problems, it may be preferable to have the overall control system behave like a unity gain system such that the system output(s) always follow the desired input(s); the control design, whether in forward (or open loop) control or in feedback control, can be viewed as an inversion of the plant. Therefore, within such structures it should be no problem to identify the inverse model for control design directly [16, 24, 28]. For the linear ARX model, presumably converted from any linear discrete model, the difference between the forward and inverse model identification is just a different choice of permutation of data columns. However, the different choices yield different answers. Based on the theory of vector projection, the differences in parameter estimates reflect the concept of what kind of information is lost (or preserved). Interestingly, to this point there are no studies in the literature using this understanding.

1.3 Summary and Outline of Dissertation

In this chapter, the engine start problem during the first few seconds of engine operation is revealed and discussed with respect to identification and control aspects. The scope of the dissertation research is clearly defined. Based on limited information available in the open public domain, the dissertation research performs a leading role in this common, but unsolved, practical control problem.

In Chapter 2, the technique of linear splines is described first, followed by an introduction of a new methodology referred to here as Linear Parameter Varying Linear Spline system (LPV-LSP). After an example using LPV-LSP system in gain scheduled control, detailed engine models for the crank-to-run transition problem are introduced.
In Chapter 3, lessons learned in the dissertation research are introduced at the beginning. The description of the learning development, though highly summarized and condensed, leads to the final control architecture which is the final predictive fuel dynamics control for engine start and crank-to-run fuel control.

Before detailed calibration methods for the predictive fuel dynamics control (introduced in Chapter 3) are introduced, topics in system identification are briefly reviewed in Chapter 4. Promising identification tools, subspace methods, are introduced thereafter. By a clever extension, subspace methods are used in Auto Regressive Moving Average (ARMA) model identification. This extension unifies several optimality measures taken in system identification. Moreover, ambiguities encountered in system identification are addressed.

In Chapter 5, calibration methods for the predictive fuel dynamics control are described with optimality measures using subspace methods. Chapter 6, draws conclusion about this dissertation. Key contributions of this dissertation are outlined, and future work is suggested.
CHAPTER 2

MODELING

In this chapter, models and modeling methodologies are developed and explained for the crank-to-run transition problem. Specifically, event based, in-cylinder fresh air charge models are developed for the crank, crank-to-run, and run regimes. Following this, a cycle-based fuel dynamics model is developed for use in control.

Prior to describing these models, however, the linear spline modeling technique is introduced first. Though readily conceived as a piecewise linear approximation tool, linear splines are, more or less, overlooked in the public domain and detailed descriptions are seldom seen or clearly documented. In response to this omission, an efficient expression of linear splines is described in greater detail with examples. The work in this dissertation along the lines of linear splines for system identification and control is an extension of ideas first brought to us by Kenneth P. Dudek, who is with Advanced Powertrain Systems Control, General Motors Corporation.

Inspired by the mechanism of linear splines, a popular gain scheduled control design tool, classified as a Linear Parameter Varying (LPV) system, is intelligently parameterized by linear splines. The confluence of two systems, which results in the

5The use of linear splines for automotive engine control is considered in the US patent application by K. P. Dudek, in “Application of Linear Splines to Internal Combustion Engine Control”.

25
LPV Linear Splines (LPV-LSP) system, is believed to be a contribution in and of itself for model based gain scheduling, and is especially useful in automotive engine control. This gain scheduled modeling and control design method is ultimately applied to air and fuel loop dynamics modeling, and is used as control for engine start problems being tackled in the dissertation research.

2.1 Linear Splines

Due to its versatility, spline regression has gained much popularity and has been applied throughout the fields of image processing, signal smoothing, CAD visualization, numerical analysis and economics forecasting. Loosely speaking, those curves and surfaces that one can see and manipulate in many common analysis and design software package are “splines”.

Among several types of splines, linear splines are the simplest. The plot of input/output data collected from a physical experiment, shown in Figure 2.1, depicts the fundamental idea behind linear splines. In the plot, the output variable $y$ has an intrinsic nonlinear relationship with the input variable $x$ subject to environmental noise and uncertainty. Linear splines make an approximation by a piecewise continuous curve using a family of straight lines linked together within user pre-defined partitions.

The points where the individual slope changes occur are called knots and the starting point is called the intercept. The knot locations can be varied depending on situations and objectives. In this dissertation, the case in which the knot locations are predefined is mainly discussed. As shown in Figure 2.1, domain of the input variable $x$ is partitioned by several connected intervals. For instance, $x \in [d_0, d_1) \cup [d_1, d_2) \cup …$
\[ \cdots \cup [d_n, d_{n+1}] \]. Based on this example of partition, the following dummy variables can be defined by a loop statement as shown in Figure 2.2.

These knot locations, \(d_1, d_2, \cdots, d_n\), can be treated as new coordinates for the variable \(x\), thereby defining the origin \(\{x_i = x - d_i\}\) for each interval. For example, the origin of \(x_2\) is at \(d_2\). In other words, \(x_i\) can be treated as a positive “perturbation” of \(x\) in domain \(D_i\) whose origin is at \(d_i\) if \(x\) is greater than \(d_i\) (given by Figure 2.1). Given a scalar value of \(x\), linear splines require a vector, \([x_1, x_2, \cdots, x_n]^T\), to complete the whole expression, if the number of partitions is \(n\). In the example, linear splines, which represent a piecewise linear curve, are constructed to cross through data points. Similarly, linear splines can be used to fit a piecewise linear curve (surface) to a perfect curve (surface).
With dummy variables, $x_1, x_2, \ldots, x_n$, any piecewise linear scalar function about $x$, such as the one depicted in Figure 2.1, can be described by a continuous function

$$y = k_0 + \sum_{i=1}^{n} k_i x_i = \phi^T \theta,$$  \hspace{1cm} (2.1)

where $\phi = [1, x_1, \ldots, x_n]^T$ and $\theta = [k_0, k_1, \ldots, k_n]^T$. Such a function can be used to fit continuous lines through discrete data points efficiently using an offset $k_0$ and families of slopes $k_1, k_2, \ldots, k_n$. To identify the offset and slopes given enough discrete data points, an approximate expression may be used:

$$Y \approx \Phi^T \theta,$$  \hspace{1cm} (2.2)

where $\Phi = [1, X_1, \ldots, X_n]^T$, $X_i = [x_i, x_{i+1}, \ldots]^T$ and $Y = [y_1, y_2, \ldots]^T$. Due to linearity of expression (2.2), $\theta$ can be identified using a least squares algorithm.
In general, such a “slope adding scheme” would be a concise characterization of linear splines for the 1-dimension case\(^6\). For higher dimension cases, linear splines represent a simply connected open surface by a family of polyhedrons or “patches”, as shown in Figure 2.3.

\[\text{Figure 2.3: 2-dimension linear spline approximation (http://www.ibiblio.org/e-notes/Splines/Inter.htm)}\]

2.1.1 Properties of Linear Splines

Although polynomial regression holds many nice properties such as smoothness, continuity and differentiability, eventually multi-collinearity arises as terms are added. Moreover, in low dimension cases, polynomial regression is not flexible enough to capture sudden changes in slope, especially with irregular intervals. Although Giarre [3] had demonstrated the time varying modeling capability with polynomials using linear structures, it is limited to a very simple case, as exemplified in [3].

\(^6\)The representation of linear splines is not unique. But the one used in this dissertation is believed to be the most efficient.
However, spline regression does not have these drawbacks. More importantly, linear splines are a simple, flexible and powerful approximation method to fulfill any sophisticated approximation task where polynomial regression fails. Certainly, higher order splines, such as cubic splines, can capture difficult nonlinearity in the data, but the extra benefit obtained by deploying higher order splines diminishes when used for model identification and control development. Although the authors in [25] did not supply a comprehensive summary for linear splines, an overview of the discussion in [25] is compiled into the following properties.

**Simplicity:** Among tools in spline regression, linear splines are simplest and use the least amount of parameters to characterize a complicated relation. For example, given a 1-dimension linear spline model, the slope in any interval/domain can be simply calculated by

\[ k_{D_p} = \sum_{i=1}^{D_p} k_i. \]  

(See Figure 2.1)  

(2.3)

\( D_p \) refers to the \( p^{th} \) interval/domain in the partition of \( D_0 \cup D_1 \cup \cdots \cup D_n \) \((p \leq n)\). Moreover, the intercept at each knot location can be pre-computed and stored for table lookup.

**Efficiency:** Any continuous and monotonic curve can be sufficiently approximated by a batch process. The degree of accuracy can be tuned by adjusting the number of partitions. More importantly, local characterization is not captured by minimizing the fitting error in each single partition, but rather by minimizing the total fitting error cross all of the partitions.

**Flexibility:** The spacing, or interval, between two adjacent knots can be arbitrary.

For some curves with many sharp turns, spacings between two adjacent knots
can be made smaller, whereas at slower varying portions of the curve the spacing can be enlarged (thereby requiring less knots).

**Identifiability:** It can be readily shown that linear splines are easy to identify. A straightforward method is the least squares algorithm. Certainly, each interval between two adjacent knots must contain at least one data point; otherwise, singularity will occur in solving the least squares problem. In addition, the ease of identification implies some degree of robustness. Due to linearity, there are no potential hazards in complicated applications.

It is worth noting that cubic splines win admirable attention in numerical analysis, smoothing and curve fitting. However, it is generally the case that cubic splines are not very useful in modeling and control design. Although, parameters enter linearly in cubic splines, in some system identification tasks with data richness issues, cubic splines may run into serious difficulties in an over-parameterized system. More importantly, cubic splines destroy the linear characterization which is embedded in linear splines; the implication of this will be revealed in later sections.

### 2.1.2 Implementation Aspects

Based on the internal mechanism of linear spline approximation, linear splines store local characteristics of a surface or curve rather than a collection of discrete data points. Therefore, it is a memory-saving algorithm. On the other hand, it is also a computationally efficient algorithm. Though computation speed is a little slower than table lookup, for control design purpose, the computation overhead is not a burden at all. In order to identify a linear spline model, the following representation
is preferred for a 2-dimension case,

\[ z = f(x, y) = K_0 + \sum_{i=1}^{m} p_{i,d_y} x_i + \sum_{j=1}^{n} q_{j,d_x} y_j, \quad (2.4) \]

where \( x \) and \( y \) are the inputs whose domains are partitioned in \( m \) and \( n \) partitions denoted \( x_i \) and \( y_j \), respectively; \( z \) is the output, and \( d_x \) and \( d_y \) are the locations of the sub-domain within which \( x \) and \( y \) are located. As shown in Equation 2.4, the linear spline model in a higher dimension case can be viewed as a summation of 1-dimension cases.

To implement a linear spline model, a different representation is preferred. In the following example, a linear spline model is developed for a 2-dimension case \((z = f(x, y))\) to illustrate the steps in implementation. In the illustration, the knot locations for \( x \) and \( y \) are assumed to be \([0, 1, 2, 3]\) and \([0, 2, 4, 6, 8, 10]\), respectively. The spacings, \( x^p \) and \( y^p \), are 1 and 2 for \( x \) and \( y \), respectively (the equally spaced case). The linear spline model can be written as

\[ z = f(x, y) = K_0 + \sum_{i=1}^{3} p_{i,d_y} x_i + \sum_{j=1}^{5} q_{j,d_x} y_j, \quad (2.5) \]

where spline coefficients \( p_{i,d_y} \) and \( q_{j,d_x} \) are assumed known (identified using a least squares algorithm) and stored as two rectangular matrices \( P \) and \( Q \), respectively, with elements \( p_{i,j} \) and \( q_{j,i} \), respectively. In addition, a rectangular matrix \( G \) is used to store all intercepts at knot locations, in the manner

\[
G = \begin{bmatrix}
d_{1,1} & d_{1,2} & d_{1,3} \\
d_{2,1} & d_{2,2} & d_{2,3} \\
d_{3,1} & d_{3,2} & d_{3,3} \\
d_{4,1} & d_{4,2} & d_{4,3} \\
d_{5,1} & d_{5,2} & d_{5,3}
\end{bmatrix}.
\]

Then, given any inputs such as \((x, y) = (2.1, 4.7)\), the following steps are required to compute \( z \) (note that \( z = f(x, y) \)).
Step 1: Compute

\[
\frac{x}{x^p} = \frac{2.1}{2} = (I_x)\frac{x^p}{1} + r_x = (2)x^p + 0.1,
\]

where \(I_x\) is the quotient (integer), \(r_x\) is the remainder and \(x^p\) is the spacing for \(x\).

Step 2: Compute

\[
\frac{y}{y^p} = \frac{4.7}{2} = (I_y)\frac{y^p}{2} + r_y = (2)(y^p) + 0.35,
\]

similarly, \(I_y\) is the quotient (integer), \(r_y\) is the remainder and \(y^p\) is the spacing for \(y\).

Step 3: The value of \(z\), given \((x, y) = (2.1, 4.7)\), can be calculated by the following formula:

\[
z = K_0 + [G]_{I_x, I_y} + [P]_{I_x, I_y}(r_x) + [Q]_{I_y, I_x}(r_y) \\
= K_0 + d_{2.2} + [P]_{I_x, I_y}(0.1) + [Q]_{I_y, I_x}(0.35),
\]

where the notation \([G]_{i,j}\) refers to the \(i,j\)-th element of the matrix \(G\), and \(P\) and \(Q\), as defined earlier, are two rectangular matrices storing the spline coefficients of the model.

In summary, for a \(N\)-dimension case, \(n\) divides, \(n\) multiplies and one sum are needed to complete calculation. For non-equally spaced partitions, one extra step is needed to complete calculation.

2.2 Models in Gain Scheduling

Taking a step forward, linear splines can be used to describe parameter varying systems. One important usage is in application to Linear Parameter Varying (LPV)
systems in gain scheduled modeling and control. So far, the notion of LPV Linear Splines (or LPV-LSP) has not appeared in the open literature, but the gain scheduled modeling using polynomials has been found to capture the time varying nature of parameters in a LPV system [3]. However, it can be shown that the LPV-LSP structure will do the same job as the LPV-polynomial approach in the example [3], with very good accuracy. However, unlike the polynomial approach, linear splines have less shape restrictions and endorse much more freedom to approximate a nonlinear function, especially the timing varying nature of system matrices in LPV systems.

2.2.1 LPV in Gain Scheduling

Though continuous-time nonlinear models are more popular (in the literature) in analyzing gain scheduled control design due to availability of analyzing tools, discrete nonlinear models are preferred in this dissertation research because of the concern for system identification and control implementation. Thus, in the remainder of this chapter, all models are in the discrete time domain unless otherwise specified.

In general, there are three forms of LPV systems. Given a discrete approximation (not necessarily linear) of a continuous-time nonlinear system or obtained directly from modeling and system identification process [20], we can either use an input/output model such as

\[ y(k) = f(y(k - 1), \ldots, y(k - n), u(k), \ldots, u(k - m), \rho), \]

or a state space model with state and output equations such as

\[
\begin{align*}
\{ & x(k + 1) = f(x(k), u(k), \rho) \\
& y(k) = g(x(k), u(k), \rho). \}
\end{align*}
\]
Here, \( u, x, y \) and \( \rho \) are the system input, state, output and exogenous input, respectively. It it always possible to approximate the discrete models by three linear structures which are:

- **Scheduled Auto Regressive Moving Average (ARMA) model,**
  \[
  y(k) + \sum_{i=1}^{n} \beta_i(\delta)y(k - i) = \sum_{j=0}^{m} \alpha_j(\delta)u(k - j), \tag{2.7}
  \]
  where \( \delta \) is the scheduling variable and \( \delta = g(y(k), u(k), \rho) \). This scheduled model is only associated with the system input and output.

- **Scheduled state space model,**
  \[
  \begin{align*}
  x(k + 1) & = A(\delta)x(k) + B(\delta)u(k) \\
  y(k) & = C(\delta)x(k) + D(\delta)u(k).
  \end{align*}
  \]
  This scheduled model is associated with the system input, state and output by scheduling variable \( \delta = g(y(k), u(k), \rho) \). The scheduled ARMA model can be represented by a scheduled state space model using a minimal realization.

- **Scheduled pole-zero model.** With a slight modification, another description of the ARMA model, the pole-zero model, results but with a different characterization of the model properties:
  \[
  \prod_{i=1}^{n} (z - p_i(\delta))y = k(\sigma) \prod_{j=1}^{m} (z - z_j(\delta))u \tag{2.8}
  \]
  where \( \delta \) is the scheduling variable and \( \delta = g(y(k), u(k), \rho) \).

All forms are considered in \([36, 42]\) and many other publications.

Based on the fundamental principle of gain scheduling, the coupling between the model parameter and input/state/output/exogenous input is temporally disregarded,
and it is typically assumed that the parameter is “frozen”. By this treatment, linear control design tools, such as output feedback, state feedback, pole placement, optimal LQR and so on, can be used for the frozen-in-time LTI system. Therefore, control will be in form of state feedback \( u(k) = K(r(k), \sigma(k))x(k) \) or output feedback \( u(k) = K(r(k), \sigma(k))y(k) \) where \( r \) is the reference signal. By doing so, finite controllers are computed at those predefined representative equilibrium points, or set points. In what follows, controller interpolation and blending are undertaken to generate the global control \[42\]. It is worth noting that, without the notion of the “frozen” parameter model, it is invalid to write down the LPV model as a transfer function, and the classical control design method using transfer functions (such as output feedback \[42\]) cannot be realized.

Controller interpolation and blending are the main tools in controller synthesis for gain scheduling. This step takes the linear control design methodologies into nonlinear realms through scheduling. In general, these controllers are simply calculated from the LPV systems and they usually share the same model structure. Therefore, connecting the controllers parameters by straight lines would be a natural choice in controller synthesis. The blending and interpolation methods do the same thing but with slightly different schemes. If the number of equilibrium points is high enough, there is, indeed, no need to blend the controller; that is, a direct jump from one control to another will do the job adequately \[36\]. In summary, the common design procedure can be summarized as a 4-step design procedure shown in Figure 2.4.
2.2.2 LPV-LSP in Gain Scheduling

With many practical industrial problems, such as the one researched in the dissertation study, it is often extremely difficult to come up with a plant model strictly from first principles (physics based modeling). In many cases where the physics based models are indeed available, the models themselves are too complicated for control design. Therefore, system identification for a certain control bandwidth and a desired accuracy is needed. Due to the nature of system identification, the model structure and system order are up to control designers, given a certain amount of data. The final form of the model will be based on the designer’s preference, understanding of the plant, degree of complexity of control, and implementation. As such, the control design is already baked in the modeling process. Or, it can be concluded that control starts from modeling.

LPV-LSP System

In the dissertation research, one of main goals is to predict cylinder fresh air charge of IC engines during engine start and crank-to-run transition. In order to overcome nonlinear behavior of air dynamics at any engine starting condition, an intelligent modeling technique and corresponding identification scheme are required.
to cover a wide range of engine operating conditions. More importantly, based on years of best engineering experience of engine experts and exhaustive investigation of air loop dynamics, a model scheduled by Intake Air Temperature (IAT), engine RPM and Throttle Position Sensor signal (TPS) could be sufficient to capture the dominant behavior of air loop dynamics at different engine start conditions.

The LPV-LSP system not only supplies a handy tool for modeling, but also simplifies the procedure of control synthesis. Rather than performing the controller interpolation or blending at the third stage of control design, the LPV-LSP system methodology described herein does the synthesis at the stage of the model construction. Such a structure is simple enough to carry the symbolic notions from modeling to control derivation, a great feature of the LPV-LSP method. In other words, by LPV-LSP, the stage of controller synthesis is no longer needed and it is automatically baked in control design at the modeling stage. To summarize the above statement, a flowchart is created and shown in Figure 2.5. As shown, the control design starting from LPV-LSP modeling and corresponding identification becomes a very first step. In the second stage, gain scheduled control design can be undertaken right away. The control stability and performance assessment will be performed at the last step.

![Figure 2.5: Identification approach for gain scheduled control](image-url)

38
LPV-LSP Model Identification

To arrive at a suitable LPV-LSP model, system identification plays a key role by
linking the model quality with the fitting error. Consider a simple LPV-LSP model
structure without concern for how scheduling variables enter into the system model,
such as

\[ y(k) = -\beta_1(\sigma)y(k - 1) - \beta_2(\sigma)y(k - 2) + \alpha_0(\sigma)u(k). \]  

(2.9)

It can be assumed that the coefficients are simple linear spline functions of a scalar
exogenous input \( \rho \) as follows:

\[
\beta_i = \beta_{i,0} + \beta_{i,1}\rho_1 + \beta_{i,2}\rho_2
\]  

(2.10)

\[
\alpha_0 = \alpha_{0,0} + \alpha_{0,1}\rho_1 + \alpha_{0,2}\rho_2.
\]  

(2.11)

When enough measurements are collected, the following approximated expression
results:

\[
Y_k \approx [-Y_{k-1}, -Y_{k-1}\rho_1, -Y_{k-1}\rho_2, -Y_{k-2}\rho_1, -Y_{k-2}\rho_2, -U_k, -U_k\rho_1, -U_{k-1}\rho_2]\theta = \phi^T\theta,
\]  

(2.12)

where \( \theta = [\beta_{0,0}, \beta_{1,1}, \beta_{1,2}, \beta_{2,0}, \beta_{2,1}, \beta_{2,2}, \alpha_{0,0}, \alpha_{0,1}, \alpha_{0,2}]^T \). It is clear that \( \theta \) can be iden-
tified by the least squares algorithm.

The goodness of the above model can be justified by least squares error. Fine
tuning of partitions and restrictions of appearance of certain scheduling variables in
parameters can be used to minimize the total fitting error. After identifying the linear
spline ARMA model, the pole-zero model as well as the corresponding state space
model can be parameterized. However, in the dissertation research, this topic will
not be discussed.
Last Words about LPV-LSP

In this Chapter, the stability assessment of gain scheduled control is not addressed. This is because, by addressing the modeling issues in the framework of LPV-LSP, the modeling error can be largely reduced. In other words, the LPV-LSP model can be representative enough to be considered as a “truth model”; as a consequence, the need for stability assessment is reduced. However, the experimental validation is still required. In general, the issue of stability is still an open research issue.

In order to show the benefit of LPV-LSP in modeling and gain scheduled control design, the dissertation now moves into the actual modeling and control problem. In the remaining sections, an event based in-cylinder fresh air charge model (using linear splines) and a cycle based fuel dynamics model are described.

2.3 Event Based In-Cylinder Fresh Air Charge Models

During engine start and crank-to-run transition, the in-cylinder fresh air charge experiences three distinct modes: crank, crank-to-run and run modes. In each mode, characteristics of the in-cylinder fresh air charge are quite different due to nonlinearity of the volumetric efficiency and throttle air flow at different regimes of crankshaft speed. As a consequence, three in-cylinder fresh air charge models are developed and linked together to describe the overall in-cylinder fresh air charge characteristics during this transient operating regime. Because the engine control is updated at each engine event\(^7\), three sequentially linked in-cylinder fresh air charge models are developed in the event domain.

\(^7\)The engine event is the time instance of when engine piston arrives at about 60° to 75° CA prior to compression Top Dead Center (TDC).
Although the underlying physics of the engine breathing process, could serve as a basic guideline in the model construction, several mathematical relationships and intelligent design choices actually play key roles in the construction of in-cylinder fresh air charge models. However, these choices do reflect physical insight and key characteristics of experiential data.

The modes of the in-cylinder fresh air charge model are defined based on three phases of crankshaft speed development process which are depicted in Figure 2.6.

![Figure 2.6: Three distinct phases during engine start and crank-to-run transition](image)

### 2.3.1 Choices of In-Cylinder Fresh Air Charge Model

In the open literature, it is popular to model the intake manifold pressure dynamics and to compute in-cylinder fresh air charge. In other words, the intake manifold pressure is the true “state” of the model; in-cylinder fresh air charge is computed algebraically from the intake manifold pressure. This choice might be a result of the fact that the intake manifold pressure is a direct measurement, and it is intuitive and straightforward to model. However, this method is not a direct way to address the in-cylinder fresh air charge dynamics; rather, it is an indirect way to generate
the in-cylinder fresh air charge as a byproduct from the dynamics of intake manifold pressure.

In the dissertation research, a direct way is undertaken to model the in-cylinder fresh air charge. Namely, the in-cylinder fresh air charge is the true “state” of the model. The intake manifold pressure measurement, crankshaft speed measurement and Volumetric Efficiency (VE, obtained from steady state operating conditions and stored as a table look-up) are used to assemble the in-cylinder fresh air charge for model construction.

These two choices, direct and indirect methods, sound similar but are quite different. In order to see the differences at the structural level, a diagram, shown in Figure 2.7, supplies a peer-to-peer comparison. In the figure, GPO (derived from “grams per cylinder out of the intake manifold”) refers to in-cylinder fresh air charge, and GPC (derived from “grams per cylinder”) refers to in-cylinder fresh air charge measured at throttle\(^8\).

![Diagram showing two choices of the in-cylinder fresh air charge model](image)

Figure 2.7: Two choices of the in-cylinder fresh air charge model

\(^8\)The acronyms GPO and GPC are used in the automotive industry
Strictly speaking, the true underlying physics inside the intake manifold is a mass balancing process. The process manifests itself by means of the intake manifold pressure which is the only feasible measurement in the research problem. Mathematically, direct and indirect methods are equivalent if the underlying process model is exactly known (though highly nonlinear). However, based on the operating regimes of the problem, measurement setup and sampling scheme, the direct method has proved to be a better choice. The failures and lessons learned using the indirect method are collected and compiled in Chapter 3. In this chapter, only the conclusions are given and listed below:

- The direct method addresses the dominating dynamics.
- The direct method reduces the effect of nonlinearity in the in-cylinder fresh air charge dynamics.
- Because the in-cylinder fresh air charge is the key variable in control calculation, the direct method eases the control integration task significantly.
- The direct method is less sensitive to the error made in the model based predictive control.

In the rest of this section, the detailed GPO models are given for the three transition modes.

### 2.3.2 Crank GPO Model

In crank phase, crankshaft speed is relatively low and the intake manifold pressure can reach equilibrium by the first firing event. As observed on a production inline-4 cylinder engine, intake manifold pressure can reach steady state conditions
within first ten engine events for a wide range of throttle positions. The traces of intake manifold pressure from the production engine in crank phase are plotted in Figure 2.8. As shown, the decay rate of MAP trajectories is influenced by TPS di-

![Intake manifold pressure after Key-On](image)

Figure 2.8: The intake manifold pressure dynamics after key-on of a production inline-4 cylinder engine

rectly. Intuitively, it is viable to build the crank GPO model with TPS dependency because, during engine cranking, volumetric efficiency is simply a fixed number\(^9\) and trajectories of the in-cylinder fresh air charge are simply scaled versions of the intake manifold pressure. In addition, the GPO decay and stabilizing process all indicate a mass balancing process between GPC and GPO. Therefore, GPC should appear in the model equation. However, the crank phase, required to be very short in future engine platforms (much less than two rotations of engine crankshaft), can be efficiently

\(^9\)This is because no inertial or residual charge effects occur.
captured by a second order, unity gained, autonomous (no input) linear difference equation of the form

\[ GPO(k + 1) = \alpha_{CRK} GPO(k) + (1 - \alpha_{CRK}) GPO(k - 1). \] (2.13)

In the above expression, \( \alpha_{CRK} \) is a linear spline function of the throttle position signal so as to capture TPS dependent decay. If \( 1 < \alpha_{CRK} < 2 \), the GPO development process in crank phase is guaranteed to be a stable, monotonically decaying process because the characteristic polynomial, which is \( z^2 - \alpha_{CRK} z - (1 - \alpha_{CRK}) = 0 \), has two roots at \( z = 1 \) and \( z = \alpha_{CRK} - 1 \), respectively. The detailed calibration process for the parameter \( \alpha_{CRK} \) is described in Chapter 5.

### 2.3.3 Crank-to-Run GPO Model

After the first firing event occurs, the GPO dynamics move into crank-to-run mode. The in-cylinder fresh air charge starts to be influenced by the residual gas.

During crank-to-run transition, an interesting phenomena is observed: the intake manifold pressure development is a monotonically decaying process if the engine fires robustly; such a trend will terminate when the bottom of MAP "well" is reached. Moreover, as observed on a production inline-4 cylinder engine, the GPO decay also follows the same trend as the MAP decay which, as depicted in Figure 2.9. More importantly, the GPO decay is not strongly correlated to the values of RPM and TPS. This type of engine behavior is quite similar to the fast decelerating operation in normal driving conditions\(^\text{10}\).

From Figure 2.9, as well as other testing data (not shown), the following conclusions are drawn:

\(^\text{10} \)The throttle is closed but the engine acts as a load and is driven by the vehicle, whereas intake manifold pressure keeps pumping down because of crankshaft acceleration.
Figure 2.9: MAP and GPO decay during crank-to-run transition

Note that: The GPC measurement is not valid during crank-to-run transition. This forbids the GPO model to have mass conservation characteristics.
• The greater the \( TPS \), the slower the decay rate is.

• The greater the \( RPM \), the slower the decay rate is.

The experimental data indicates that a first order autonomous decay model would be sufficient with a TPS and RPM dependent decay rate, given by

\[
GPO(k + 1) = \alpha_{CTR}(TPS(k), RPM(k))GPO(k),
\]

where

\[
\alpha_{CTR}(TPS(k), RPM(k)) = K_0 + \sum_{i=1}^{m} K_i TPS_i + \sum_{j=1}^{n} K_{m+j} RPM_j.
\]

Note that, in the above linear spline expression, the number of parameters is largely reduced because a similar role is played on the MAP/GPO decay rate by TPS and RPM. Note that \( TPS_i \) and \( RPM_j \) are linear splines partitions, as in Equation 2.4, at index (event) \( k \); that is, \( \alpha_{CTR} \) is implicitly a function of index \( k \).

For engines without an Electronic Throttle Control (ETC) system, TPS is fixed during crank-to-run transition. Therefore, it would be sufficient to let \( \alpha_{CTR} \) be a linear spline function of RPM alone. In that case, the crank-to-run GPO model can be simplified as

\[
GPO(k + 1) = \alpha_{CTR}(RPM(k))GPO(k),
\]

where RPM enters the parameter \( \alpha_{CTR} \) as a linear spline function, described by

\[
\alpha_{CTR} = K_0 + \sum_{j=1}^{n} K_j RPM_j.
\]

### 2.3.4 Run GPO Model

When the engine crankshaft speed passes the RPM flare and the intake manifold pressure reaches the bottom of MAP "well", a mass balancing characteristic between
GPC and GPO is nearly achieved. As shown in Figure 2.9, the run mode can be distinguished around event 50. It is approximately at this point that we “hand-off” the air prediction task to whatever scheme is used for the Run regime (detail of the run GPO model is omitted here).

### 2.4 Cycle Based Overall Fuel Dynamics Model

The overall fuel dynamics model, during engine start and crank-to-run transition, consists of two cascaded components and is depicted in Figure 2.10.

![Figure 2.10: Overall fuel dynamics model](image)

In Figure 2.10, UFF refers to the Utilized Fuel Fraction which represents a static, ECT dependent, input correction function, RINJ refers to Raw Injected Fuel Mass, CINJ refers to Corrected Injected Fuel Mass (by UFF) and MBFM refers to Measured Burned Fuel Mass. The main purpose of the UFF in Figure 2.10 is to correct the RINJ so as to yield an intermediate variable, CINJ, such that a mass conservative nominal fuel dynamics model can be constructed. Another purpose of using the static UFF function is to reflect what is indicated by real engine data and to facilitate identification and implementation.
2.4.1 Utilized Fuel Fraction

An important phenomena has been observed when starting the engine at relatively low engine coolant temperatures. The main conclusion of the observation is that a certain amount of the RINJ does not appear in the MBFM, even when the engine RPM almost reaches a steady state condition. Figure 2.11 depicts this characteristic, for test data of a start at -15°C.

![Graph showing Raw Injected Fuel Mass and Measured Burned Fuel Mass at -15°C](image)

Accumulated RINJ over 20 cycles is 4729 mg
Accumulated MBFM over 20 cycles is 800 mg

Figure 2.11: Raw Injected Fuel Mass and Measured Burned Fuel Mass at -15°C

It is believed that the largest portion of the fuel, that “disappeared” in Figure 2.11, is stored temporally in engine subsystems, while the rest escaped through the exhaust. The stored portion of the raw injected fuel mass will eventually appear in the output measurement, but not in the time window of interest. Based on this understanding,
the raw injected fuel mass is not fully utilized during engine starts. As such, the Utilized Fuel Fraction is introduced to represent such an effect.

**UFF near Cycle 20**

The inefficient fuel utilization, as depicted in Figure 2.11, persists all the time when the engine is not fully warmed up. Figure 2.12 strengthens this statement by showing the fuel utilization efficiency at different ECTs. As shown, the ratio between RINJ and MBFM near Cycle 20 becomes larger and larger as the ECT decreases. The trend of the UFF around Cycle 20 is plotted in Figure 2.13 against a regressed quadratic polynomial function with respect to the ECT; the term $UFF_{20}$ in Figure 2.13 refers...
to UFF at Cycle 20. It is evident that the “fuel loss” becomes greater at colder temperatures. This is the key reason of why fuel enrichment requirement during cold start is needed. It becomes crystal clear that, without improvements on the quality of fuel preparation, fuel economy and emission are difficult to improve.

![Trend of UFF20](image)

Figure 2.13: Trend of UFF20 at different ECTs

Though this observation is evident based on the experimental data, the introduction of $UFF_{20}$ to the fuel dynamics model in this dissertation work is unique (to date) to characterize the “fuel loss” concept during the early stages of fuel preparation for SI PFI engines. Such an effect profoundly influences the fuel control strategy at engine cold starts.
Diminishing Return Effect

In addition to the inefficient utilization of raw injected fuel at near steady state conditions when the engine is not fully warmed up, even worse fuel utilization is observed in the first few fueling cycles [19].

Based on the primary objective of the overall fuel dynamics model, to complete the overall model, a linear mass conservative nominal fuel dynamics model must be constructed between CINJ and MBFM. However, the usage of the constant UFF, obtained from the $UFF_{20}$ measurement, only removes the partial nonlinearity appearing near steady state condition$^{11}$; the input/output data (CINJ and MBFM) still does not admit a reasonably good linear relationship. Furthermore, the well-understood nature of linear dynamical systems indicates that the input/output of a mass conservative linear system with finite memory do not stray far away from each other. However, there is still a huge difference between CINJ and MBFM after RINJ is corrected by $UFF_{20}$ (See Figure 2.14, 2.15 and 2.16).

Therefore, without introducing a nonlinear UFF concept, such big differences as shown in Figure 2.15 and 2.16 between CINJ (by the constant UFF) and MBFM can never be reduced. It is now clear that the choice of a nonlinear UFF would be the way to meet the correction requirement. As described in [19], “The more you inject the more production you get from burned fuel, but with a diminishing return” would be a concise summary to characterize the nonlinear correction effect.

$^{11}$Strictly speaking, the engine variables at start conditions never reach steady state. When the nomenclature of “near the steady state” is used, it just means that no big transient appears on the input and output.
Figure 2.14: Evidence of diminishing return effect: Different RINJ profiles

Figure 2.15: Evidence of diminishing return effect: Different CINJ profiles
Test #1: Total first 10 cycles MBFM is 352 mg
Test #2: Total first 10 cycles MBFM is 310 mg
Test #3: Total first 10 cycles MBFM is 342 mg
Test #4: Total first 10 cycles MBFM is 321 mg

Figure 2.16: Evidence of diminishing return effect: Different MBFM profiles

Final Form of UFF

Although, in the public literature, there is no direct indication about the exact form of the nonlinearity required in the nonlinear UFF formation, results from [19] implied some possible directions; after several trials for different functional forms, the following form is chosen in the dissertation research:

\[
CINJ = UFF_{20}(ECT) \left(1 - \frac{2}{\pi} \arctan \left(\frac{RINJ}{\gamma(ECT)}\right)\right) RINJ
\]  

(2.18)

where \( UFF_{20} \) and \( \gamma \) are ECT dependent scalar functions.

Benefit, Indication and Issue of the UFF Formulation

The major benefit gained in choosing the arctangent form in the UFF formulation is threefold:
• True to the actual physical phenomena observed, UFF is a smooth, monotonic, increasing function with respect to the input RINJ, but a decreasing function with respect to the input ECT.

• The single parameter, $\gamma(ECT)$, is used to characterize a shape which meets the correction requirement to capture the “diminishing return” effect. The single ECT dependent parameter eases the calibration process and, in general, admits a robust parameter estimate when data richness is an issue.

• The magnitude of $\gamma(ECT)$ is in the same range of the first indexed RINJ ($RINJ(1)$) in a normal engine start for a given fixed ECT. $\gamma(ECT)$ can therefore be viewed as a weighting parameter for RINJ correction in the first few engine cycles.

The basic correction characteristic of the UFF, which is the diminishing return effect along with increased RINJ, is depicted in Figure 2.17. On the other hand, the diminishing return effect, or forward insensitivity issue, also indicates the strong sensitivity issue in the inverse direction: A small requirement on CINJ requires huge action on RINJ.

![Figure 2.17: Diminishing Return Effect](image-url)
Such a formulation of UFF, though intuitive and true to the experimental data, still requires some modification when used in control implementation. The detailed procedure is introduced in later Chapters.

2.4.2 Nominal Fuel Dynamics

By construction, the nominal fuel dynamics model is formulated as possessing mass conservative (unity gained) linear dynamics. Observed in Figure 2.15 and 2.16, it can be concluded that the system order (depth of memory) would not be very large.

After several trials, the fuel dynamics model structure is selected to take the first order discrete $\tau$-X model form, but the coefficients are scheduled by the ECT; that is,

$$
\begin{align*}
    m_{dep}(k) &= (1 - \alpha)m_{dep}(k - 1) + (1 - X)m_{cinj}(k) \\
    m_{cyl}(k) &= \alpha m_{dep}(k - 1) + Xm_{cinj}(k)
\end{align*}
$$

In the above expression, $m_{dep}$ refers to the deposited fuel mass on the surface of the fuel pass, such as intake valve, top of piston and cylinder wall; $m_{cyl}$ refers to the true fuel mass in vapor form which indeed participates in the combustion process and is measured by MBFM; $m_{cinj}$ is the CINJ.

Facts about the First Fueling Cycle

In order to use the discrete $\tau$-X model for first cycle fuel control, there is an important artifact worth attention: at engine start conditions, the quantity of the first cycle in-cylinder fuel mass is dictated mainly by the parameter $X$ and UFF. In other words, the parameter $X$ can be viewed as a direct feed-through term of the control. It can be shown that the first cycle in-cylinder burned fuel mass ($MBFM(1)$ or $m_{cyl}(1)$) can be calculated by the following equation:

$$
m_{cyl}(1) = MBFM(1) = (X) UFF(RINJ(1)) RINJ(1).
$$

(2.19)
It becomes clear that the coupling between UFF and nominal fuel dynamics is nonlinear; it is not easy to identify them in separate procedures. In addition, this coupling also indicates a constraint of the product, $(X)UFF(RINJ(1))$, in order to achieve engine start.

**One-Pole, One-Zero Representation**

Due to one-to-one correspondence of the $\tau$-$X$ model and the ARMA model, the nominal fuel dynamics also can be written in the following from:

$$m_{cyl}(k) = (1 - \alpha)m_{cyl}(k - 1) + Xm_{cinj}(k) - (X - \alpha)m_{cinj}(k - 1).$$  \hspace{1cm} (2.20)

The unity gain property of the $\tau$-$X$ model is obvious. After several new variables and parameters are defined, the above ARMA model is written as

$$y(k) = -\beta_1(ECT)y(k - 1) + \alpha_0(ECT)u(k) + \alpha_1(ECT)u(k - 1).$$  \hspace{1cm} (2.21)

The reason of doing this substitution is to get a compact expression later on. For cross comparison purposes, one-to-one correspondence of the parameters used in both models is shown in Table 2.1.

<table>
<thead>
<tr>
<th>State space model</th>
<th>ARMA model</th>
</tr>
</thead>
<tbody>
<tr>
<td>$1-\alpha$</td>
<td>$-\beta_1$</td>
</tr>
<tr>
<td>$X$</td>
<td>$\alpha_0$</td>
</tr>
<tr>
<td>$-(X - \alpha)$</td>
<td>$\alpha_1$</td>
</tr>
</tbody>
</table>

Table 2.1: One-to-one correspondence of the model parameters
In control development, which is developed in the next chapter, both models are used simultaneously.

2.5 Summary

In the dissertation research, the development of LPV-LSP system, as described in the first half of this chapter, is believed to be a breakthrough in field of model based engine control. Though the LPV-LSP model is only applied to the GPO model throughout air and fuel loop dynamics modeling, its impact on the GPO prediction is profound and will be elaborated upon in later chapters. Though air and fuel loop models are given without showing detailed development, their construction will become clear in the next chapter, prior to the development of the predictive fuel dynamics control.
CHAPTER 3

PREDICTIVE FUEL DYNAMICS CONTROL

As discussed in Chapter 2, fuel dynamics control during engine cold start and crank-to-run transition mainly relies on open loop compensation. As such, a cylinder air rate prediction and a fuel dynamics compensator are essential. Provided the predicted in-cylinder fresh air charge is used to compute the input of the fuel dynamics compensator, a predictive fuel dynamics control results.

In this chapter, prior to a detailed description of the predictive fuel dynamics control, experiences in the evolution of the control development are reported first, together with detailed analysis. These analyses lead to several key lessons which are believed to be extremely valuable to arrive at the final solution. In what follows, all components of the predictive fuel dynamics control are described in detail. In the description of design development, some concepts already reported in Chapter 2 (such as UFF) are re-visited to illustrate lessons learned in the overall evolution of the crank-to-run control solution.

3.1 Architecture of Predictive Fuel Dynamics Control

The architecture of predictive fuel dynamics control developed in this dissertation research is depicted in Figure 3.1. As shown, the predictive fuel dynamics control
consists of four main components and each individual component has its own sub-problems:

**Multi-Step GPO Prediction:** In this component, the task of cylinder air rate prediction is accomplished. Due to event-based fuel control and PFI configuration, the cylinder air rate must be predicted up to three steps ahead for Inline-4 (I-4) engines and six steps ahead for V-8 engines (measured by the number of engine events) for fuel command calculation. In this dissertation research, a scheme of three step ahead GPO predictions, mainly for I-4 engines, is developed.

**Nominal Fuel Dynamics Compensator:** This component in predictive fuel dynamics control, essentially, is a direct inverse of the nominal fuel dynamics model. By this configuration, open loop control is aimed to achieve approximate unity gained response.

**Inverse Utilized Fuel Fraction:** By inverting the forward UFF function, the output of the nominal fuel dynamics controller is corrected to compensate the lost fuel effect. Together with the nominal fuel dynamics compensator, the inverse UFF function realizes a control such that, by compensating the lost fuel effect, the measured exhaust EQR is very close to the commanded EQR measurement.

**Desired Exhaust EQR:** This element gives several flexible choices in the control calibration process:

- Emissions requirements: It is straightforward to make the exhaust EQR as lean as possible so as to reduce the engine-out HC emission because the predictive fuel dynamics control offers an approximate unity gained
response. The commanded EQR can be a single number or a trajectory, implying freedom in choosing the desired exhaust EQR target.

- Robust starting: The robust engine start can be precisely controlled in the first few cycle fueling events, without enriching the later fueling cycles.
- Plug fouling: Because of individual cylinder fuel dynamics compensation, cast in a predictive control framework, plug fouling can be avoided during enrichment.

Figure 3.1: Architecture of predictive fuel dynamics control

Though readily conceived as an open loop fuel dynamics compensation scheme (Figure 3.1), such an architecture, used during engine start and crank-to-run transition, is quite powerful. Due to the unique characteristics of the problem and the practical implication of the solution, we have very little to compare our results against in this dissertation. We therefore choose to present the research results by reporting, discussing and analyzing the major learnings (through a sequence of design attempts) prior to the introduction of the final form of the predictive fuel dynamics control. By doing so, we present a cohesive development of the final control, and demonstrate that it indeed outperforms other possible/potential approaches.
3.2 Design Evolution and Lessons Learned

As with any significant research problem, several design attempts characterize the four-year study; six of these attempts have changed the direction of this dissertation research and are worth mentioning. Although each could be viewed as a failed attempt, lessons are learned and better understandings of the problem are gained, eventually leading to the final solution. Because of their importance, these six attempts and lessons learned are discussed here and a summary is given in Table 3.1.

<table>
<thead>
<tr>
<th>No</th>
<th>Design Attempts</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Third order MAP/RPM predictor</td>
</tr>
<tr>
<td>2</td>
<td>Virtual AFR sensor</td>
</tr>
<tr>
<td>3</td>
<td>Piecewise UFF Model</td>
</tr>
<tr>
<td>4</td>
<td>MAP/RPM dependent fuel dynamics model</td>
</tr>
<tr>
<td>5</td>
<td>Usage of mass conservation</td>
</tr>
<tr>
<td>6</td>
<td>UFF model without independent UFF(1)</td>
</tr>
</tbody>
</table>

Table 3.1: Evolution to final design

In the following, the six design attempts in this evolution are summarized and the lessons learned are discussed.

1. Third Order MAP/RPM Predictor: A natural choice for predicting GPO is through MAP and RPM prediction and subsequent computation of GPO using tabulated $VE$ such as

$$GPO(k + i|k) = VE (MAP(k + 1|k), RPM(k + i|k)) \frac{MAP(k + i|k)}{IAT(k)},$$
where $MAP_{k+i|k}$ and $RPM_{k+i|k}$ denote $i$-th step ahead MAP and RPM predictions given information at “$k$”, respectively. Such a GPO prediction and computation scheme is depicted in Figure 3.2. Under this prediction structure

![Figure 3.2: Predicting GPO by means of predicted MAP and RPM](image)

and based on experience in constructing the cause and effect relationship, MAP and RPM dynamics are modeled as two third order coupled linear systems. The detailed model equation is shown below:

$$x(k + 1) = Ax(k) + Bu(k),$$  \hspace{1cm} (3.1)

where

$$A = \begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
\alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 & \alpha_5 & \alpha_6 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
\beta_1 & \beta_2 & \beta_3 & \beta_4 & \beta_5 & \beta_6
\end{bmatrix},$$

$$B = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
\gamma_1 & \gamma_2 & \gamma_3 & \gamma_4 & \gamma_5 & \gamma_6 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
\lambda_1 & \lambda_2 & \lambda_3 & \lambda_4 & \lambda_5 & \lambda_6
\end{bmatrix}.$$
$x^T(k) = [MAP(k-2), MAP(k-1), MAP(k), RPM(k-2), RPM(k-1), RPM(k)]$

and $u^T(k) = [TPS(k-2), TPS(k-1), TPS(k), GPC(k-2), GPC(k-1), GPC(k)]$.

The above model is used to describe MAP/RPM dynamics during crank-to-run transition, whereas, when used in crank mode, matrix $B$ is set to zero. In addition, parameters in the above model have the following steady state constraints:

**MAP Dynamics in Steady State:**
\[
\alpha_1 + \alpha_2 + \alpha_3 - 1 = 0, \quad \alpha_4 + \alpha_5 + \alpha_6 = 0,
\gamma_1 + \gamma_2 + \gamma_3 = 0 \quad \text{and} \quad \gamma_4 + \gamma_5 + \gamma_6 = 0.
\]

These constraints guarantee a consistent representation for MAP in steady state.

**RPM Dynamics in Steady State:**
Similarly,
\[
\bar{\beta}_1 + \bar{\beta}_2 + \bar{\beta}_3 = 0, \quad \beta_4 + \beta_5 + \beta_6 - 1 = 0, \quad \lambda_1 + \lambda_2 + \lambda_3 = 0 \quad \text{and} \quad \lambda_4 + \lambda_5 + \lambda_6 = 0.
\]

The above constraints actually imply integral actions for MAP and RPM, respectively. That is, two poles are at $z = 1$ which reflect physical insights: 1) storage effect of intake manifold; 2) integral effect of engine inertia. The key reason for constructing this third order MAP/RPM model is because three step ahead MAP/RPM prediction can be easily realized. Under this model structure, several coupling modes and initial condition setup modes have been tried and results are summarized below:

**Full MAP/RPM coupling mode:** In this mode, $\alpha_4, \alpha_5, \alpha_6, \beta_4, \beta_5$ and $\beta_6$ are not identically zero. However, MAP and RPM are mutually coupled or influenced. In normal engine starts, this coupling is the appropriate way to describe the MAP/RPM dynamics. However, in anomalous engine starts, this coupling effect between MAP and RPM is invalid. In other words, nonlinear effects are not captured in this model structure.
When used as a predictor, future RPM development is difficult to predict accurately in the face of misfire and poor-start. Therefore, bad RPM predictions can adversely affect the MAP prediction, and vise versa. More importantly, during crank-to-run transition, TPS is usually held constant and TPS terms do not play a role in the MAP/RPM dynamics at all; also, GPC measurement during crank-to-run transition is not valid and the GPC terms are dropped. These conditions force the usage of the crank-to-run GPO predictor only relying on matrix $A$. That is, the future MAP depends on

$$MAP(k + 1) = \sum_{i=1}^{3} \alpha_i MAP(k + 1 - i) + \sum_{j=1}^{3} \alpha_{j+3} RPM(k + 1 - j).$$

If $\sum_{j=1}^{3} \alpha_{j+3} RPM(k + 1 - j)$ is not well behaved, MAP prediction is not accurate.

**MAP couples RPM:** $\alpha_4$, $\alpha_5$, $\alpha_6$ are not identically zero. Dynamics of MAP affect RPM, but not the other way around. Although the adverse effect from bad RPM prediction is removed, the problem of initial condition setup enters the problem.

**RPM couples MAP:** $\beta_4$, $\beta_5$, $\beta_6$ are not identically zero. In this mode, problems encountered in the full MAP/RPM mode are retained, therefore it is not used.

**Various TPS to GPC coupling modes:** These modes are not key players in the MAP/RPM prediction.
Initial Condition Setup: In all coupling modes mentioned above, MAP/RPM prediction suffers from problems of initial condition setup. In fact, the impact is more profound than any choice of coupling modes in the face of misfire and poor-start. An effective way to deal with this problem is by adding rules. However, as rules are added, the returns are diminishing. Figure 3.3 details two possible rules for initial condition setup. In this figure, $\Delta MAP$ denotes the change in MAP, and $MAP(k+1|k)$ denotes the value of MAP at $k+1$, given only the value at $k$.

Lesson Learned: Although appropriate for normal “run mode” of operation, this method of GPO prediction for crank-to-run transition fails to address critical issues of robustness and nonlinearity. When questioning the primary goal of MAP/RPM prediction, it is realized that GPO prediction by means of MAP/RPM prediction is not a direct way to address the problem. In contrast, predicting GPO directly using a GPO model may bear fruit (see Figure 3.4), and such an approach will ultimately be used, described later.

2. Virtual AFR Sensor: The motivation for the virtual sensor concept is to address the exhaust EQR measurement problem of the WAFR sensor, discussed in Chapter 1. Figure 3.5 demonstrates inconsistent EQR measurement problems at Cycle 1 (C1) and Cycle 2 (C2) for a typical data stream. From the firing behavior of the engine, the exhaust EQR must be, at least, at or above the lean limit ($EQR \geq 0.7$). However, neither of the two measurements (shown in Figure 3.5) reaches such a threshold. Based on the observation of firing behavior, it is believed that variables associated with combustion may indicate
**Atmosphere pressure:** around 98 kPa

**Intake Manifold Absolute Pressure (MAP)**

\[ MAP(k+1|k) \]

\[ MAP(k) \]

\[ MAP(k-1): \text{artificial} \]

\[ MAP(k-2): \text{artificial} \]

<table>
<thead>
<tr>
<th>Event</th>
<th>k-2</th>
<th>k-1</th>
<th>k</th>
<th>k+1</th>
<th>k+2</th>
<th>k+3</th>
</tr>
</thead>
<tbody>
<tr>
<td>True MAP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1\text{st step MAP prediction}</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>True MAP trajectory</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\[ \Delta MAP = 5\text{kPa} \]

**Scheme 1**

**Scheme 2**

**Figure 3.3:** Two possible schemes for initial condition setup

**Figure 3.4:** Directly predicting GPO
Figure 3.5: Cycle 1 and cycle 2 EQR measurement problem

more accurate EQR information. The following two hypotheses are made: 1) first cycle Indicated Mean Effective Pressure (IMEP) measurements may have a clear relationship with respect to RINJ; 2) changes of RPM over the first two cycles may have a clear relationship with respect to RINJ. In order to verify these hypotheses, 100 hot restart tests are conducted and results are shown in Figure 3.6 and Figure 3.7. As shown in both figures, data scattering makes the above hypotheses invalid. Furthermore, no clear indication about the degree of data scattering at cold start conditions is evident.

Lesson Learned: Based on observed combustion variability as well as the pattern of data scattering obtained in this research, the virtual sensor concept is discarded.

Design 3: Piecewise UFF Formulation. A logical first attempt at the formulation of the UFF model is simply a summation of two functions: 1) a constant
Figure 3.6: First cycle IMEP vs. RINJ

Figure 3.7: First cycle change of RPM vs. RINJ
UFF curve; 2) a quadratic UFF function. This formulation of the UFF model is depicted in Figure 3.8. In order to identify the parameters associated with

![Figure 3.8: A piecewise UFF Model](image)

this UFF formulation, such as the parameter \( \text{offset} \) shown in Figure 3.8, optimization is used in conjunction with the identification of the nominal fuel dynamics model. As such, an initial guess of the starting value for parameter \( \text{offset} \) is used to initiate the optimization routine. However, we were not able to find a consistent optimal value of parameter \( \text{offset} \) over several optimization iterations, with different initial guesses of \( \text{offset} \).

**Lesson Learned:** The optimization cannot return reliable parameter estimates associated with this piecewise UFF function. More specifically, there is basically no way to decide the location of \( \text{offset} \) (see Figure 3.8) using optimization given noisy and sparse data. This lesson led to a simple single parameter nonlinear function which captures the steady state characteristic of UFF as well as the diminishing return effect during crank-to-run transition. The UFF reported in Chapter 2 was the result.
Design 4: MAP/RPM dependent Fuel Dynamics Compensator. In the identification process for the nominal fuel dynamics model, it was found that MAP and RPM both influence parameters of the nominal fuel dynamics model, especially the direct feed-through term. This effect is also confirmed by the experimental studied in [10], [19] and [38]. The general understandings about this effect are:

- The lower the MAP, the greater the value of the direct feed-through term of the nominal fuel dynamics model. This effect may indicate a physical insight: low pressure may help vaporization of the injected fuel mass.

- The greater the RPM, the greater the value of the direct feed-through term of the nominal fuel dynamics model. Intuitively, a greater engine speed can result in a faster air flow rate, therefore benefiting the fuel evaporation.

Under this guideline, the nominal fuel dynamics model (one-pole, one-zero model) is modified as follows:

\[
y(k) = \beta_1(MAP(k - 1), RPM(k - 1), ECT)y(k - 1) \\
+ \alpha_0(MAP(k), RPM(k), ECT)u(k) \\
+ \alpha_1(MAP(k - 1), RPM(k - 1), ECT)u(k - 1)
\]

In the above model, all parameters (which are functions of \(\beta_1, \alpha_0, \alpha_1,\)) are linear spline functions with respect to MAP and RPM\(^{12}\).

**Lesson Learned:** There is no obstacle to build such a model and it can be identified using experimental data; however, as a result of use in control, it was

\(^{12}\)The use of linear splines for automotive engine control is considered in the US patent application by K. P. Dudek, in “Application of Linear Splines to Internal Combustion Engine Control”.  

71
conjectured that $MAP(k)$, $MAP(k-1)$, $RPM(k)$ and $RPM(k-1)$ should be predicted values rather than the past and current measurement. Unfortunately, due to the unsolved issue of MAP/RPM prediction in the face of misfire and poor-start, the model parameters, especially $\alpha_0(MAP(k), RPM(k), ECT)$, may have unwanted variations, causing a large variation in $CINJ(1)$. Thus, such a nominal fuel dynamics model cannot yield robust control, and we decided to eliminate MAP and RPM terms in the above model equation.

**Design 5: Usage of Mass Conservation.** Recall that the overall fuel dynamics model is partitioned into two parts: 1) UFF model; 2) nominal fuel dynamics model. This modeling decision actually creates a problem in identification: these two components are coupled and cannot be identified by two separated processes, thereby requiring UFF and nominal fuel dynamics to be identified simultaneously. A simplified identification scheme for these two components is described in Figure 3.9. Due to problems inherent to nonlinear optimization

![Figure 3.9: Nested loops identification scheme for UFF and nominal fuel dynamics](image-url)
methods, optimality of parameter estimates using such a scheme is not well formulated. Therefore, an approximation is made such that

\[
\sum_{k=1}^{20} CINJ(k) = \sum_{k=1}^{20} MBFM(k)
\]

by tuning \( \gamma(ECT) \). This approximation actually indicates that, at cycle 20, no significant amount of fuel mass is left in the intake port and cylinder. The value of \( \gamma(ECT) \) can be easily found by a root finding scheme. As such, the UFF model can be identified in a separate process and the nominal fuel dynamics identification can be carried out alone, independent of the UFF parameter. Namely, decoupling is achieved by neglecting the deposited fuel mass inside intake port and cylinder.

**Lesson Learned:** Although the decoupling scheme supplies a clean solution, and appears to be a very good solution, it results in an over-gained response when used as control. That is, the parameter \( \gamma(ECT) \) is significantly smaller than it should be and the trend of the parameter estimates of the nominal fuel dynamics model (with respect to ECT) is incorrect. Figure 3.10 depicts such effects. Upon close inspection of this approximation method, we learned that the difference between the sum of CINJ and the sum of MBFM over 20 engine cycles indeed tells an accurate story about the nominal fuel dynamics, and it should not be neglected. This approximation method actually breaks the law of mass conservation. In contrast, deposited fuel mass actually supplies a constraint for the nominal fuel dynamics model identification. This is true for all dynamic systems, given the input trajectory and zero past input/output (such as in the engine start condition). The detailed usage of mass conservation
is revealed in Chapter 5, where we will see that a concept of accumulative mass conservation indeed yields a robust identification.

**Design 6: UFF model without independent UFF(1).** Fully investigated in [19], the fuel utilization phenomena appears to be unique at C1 and C2. However, in the development of Chapter 2, the lost fuel effects at C1 and C2 are treated identically. With such an approach, a sensitivity issue arises when inverting the UFF function as control, as depicted in Figure 3.11 (heavy dashed curve). In the figure, a small variation on commanded CINJ can induce a huge variation on RINJ. Under normal engine start conditions, the commanded CINJ trajectory is sufficiently smooth and generally decays monotonically, and the commanded RINJ trajectory is well behaved. However, under abnormal engine
start conditions such as misfire and poor-start, the predicted GPO may respond to abnormalities accordingly by correcting the previously predicted GPO. As such, the fuel command CINJ, computed by the nominal fuel dynamics compensator, may have some variation (sliding on the heavy solid line in Figure 3.11). Therefore, the final fuel command RINJ will have a large variation. Such a resulting compensation is unsatisfactory because the rescue action, in the face of misfire and poor-start, is inappropriate.

**Lesson Learned:** The sensitivity issue as discussed above appears only when anomalous engine starts occur, not in normal engine starts. However, the fuel dynamics control must be able to accommodate all kinds of abnormalities so as to ensure robustness. Our major learning here is that in order to circumvent this problem, the UFF function at C1 or C2 should be a fixed number, rather than
a nonlinear function. By doing so, the sensitivity issue is drastically reduced while an adequate lost fuel correction requirement is attained (heavy solid line in Figure 3.11).

**Summary of Lessons Learned**

In the above, six designs ideas and attempts were introduced and analyzed. It is worth pointing out that, among these six designs, four of them are directly tied to air prediction. Thus, it is obvious to draw a meaningful general conclusion:

\[
\text{inaccurate air prediction} \Rightarrow \text{poor fuel dynamics compensation}
\]

This discovery may seem overstated, however, it indeed tells the key point in fuel dynamics control during engine start and crank-to-run transition. Generally speaking, the whole control requirement in this research is open loop tracking control using predicted input (commanded MBFM); without a good handle on input prediction, there is no point in dwelling on the control problem.

In the rest of this chapter, details of each component of the final design predictive fuel dynamics control are given.

### 3.3 Three Step Ahead GPO Prediction For I-4 Engines

For PFI engines, a prominent control problem is to compute a fuel command prior to fresh air charge breathed into the cylinders. Because the air loop dynamics, which are mainly governed by the driver via the acceleration pedal, are relatively slow with respect to fuel delivery (directly controlled by the ECU), fuel calculation must follow the air flow, rather than the other way around. Hence, a viable and a best solution to address this causality issue is to deploy model predictive control. As such, cylinder
air rate is predicted prior to the intake event by assuming the throttle position is not varying during prediction and no large external load is present on the engine crankshaft. Based on an event triggered execution of the engine control algorithm, cylinder air rate prediction must be made at least three steps ahead for I-4 engines and six steps ahead for V-8 engines, by means of counting the number of engine events.

In order to facilitate the realization of this idea, three basic concepts come into play. First, in order to start as quickly as possible, it will be important to synch the engine at the beginning of cranking. For this purpose, a Known CAM Position Algorithm (KCPA) is designed and executed in the immediate previous engine shut down where the CAM position is captured. Secondly, ideally we wish to maximize the fuel residence time in order to enhance vaporization; therefore, fuel should be injected as early as possible. Lastly, and in conjunction with the preceding concept, we wish to take advantage of existing fuel injection hardware that intelligently processes multiple commands to a single cylinder. That is, the fuel processor should be smart enough to implement one command versus another (for example, a command which is larger than another received simultaneously should take precedence). With such a capability, successive fueling actions become increasingly more accurate. We will accomplish this idea by making predictions (up to) three events ahead and computing new predictions at each tick of the clock (thereby progressively lending accuracy to the total fuel delivery process).

If the fuel command, can be updated iteratively as the engine progresses toward the intake stroke, the fuel command can be adjusted so as to reflect the latest engine operating conditions. As a result, multi-step GPO prediction is essential in this
scheme. Based on the GPO models which are developed in Chapter 2, future GPOs are predicted using these models under corresponding transition rules by assuming that the throttle opening is not varying during prediction.

![Diagram of GPO prediction iterations](image)

Figure 3.12: Three steps ahead GPO prediction for I-4 engines

### 3.3.1 Crank GPO Predictor

The crank GPO predictor consists of 1\textsuperscript{st}, 2\textsuperscript{nd}, and 3\textsuperscript{rd} step ahead GPO predictions. The equations associated with the crank GPO predictor are summarized below:

\[
GPO(k + 1|k) = \alpha_{CRK}GPO(k|k) + (1 - \alpha_{CRK})GPO(k - 1|k)
\]
\[
GPO(k + 2|k) = \alpha_{CRK}GPO(k + 1|k) + (1 - \alpha_{CRK})GPO(k|k)
\]
\[
GPO(k + 3|k) = \alpha_{CRK}GPO(k + 2|k) + (1 - \alpha_{CRK})GPO(k + 1|k).
\]
The parameter, $\alpha_{CRK}$, where subscript “CRK” denotes “crank” condition, is a single fixed number for all engine start conditions. Because the crank GPO predictor only runs for a short period of time (e.g., on an I-4 engine, only the first three engine events are involved with the crank GPO predictor if KCPA is used), $\alpha_{CRK}$ is tuned manually. The notation $GPO(k+i|k)$, means, “GPO prediction at event $k+i$, given information at event $k$”.

To initialize the crank GPO predictor, the following initial condition setup is suggested:

\[
\begin{align*}
GPO(1|1) &= GPOM(1) \\
GPO(0|1) &= GPOM(0).
\end{align*}
\]

(3.2) (3.3)

In the above expressions, the GPO measurement at event $k$, $GPOM(k)$, is calculated using

\[
GPOM(k) = V_{E_{CRK}} \frac{MAP(k)}{IAT(k)}
\]

(3.4)

where $V_{E_{CRK}}$ is the Volumetric Efficiency (VE) at the cranking speed, calculated from the geometry of the piston and cylinder head, with known compression ratio (a single fixed number). $GPOM(0)$ simply means the GPO measurement at Key-On event. Because of the high signal-to-noise ratio in the crank GPO measurement, no state estimation is needed, therefore no measurement update is required.

13When the KCPA is used, the intake manifold pressure will not drop significantly prior to the first firing event, and it is reasonable to use this simple GPO predictor.
3.3.2 Crank-to-Run GPO Predictor

The crank-to-run GPO predictor consists of 1\textsuperscript{st}, 2\textsuperscript{nd}, and 3\textsuperscript{rd} step ahead GPO predictions and measurement update. The equations associated with the crank-to-run GPO predictor are summarized below:

\begin{align*}
GPO(k+1|k) &= \alpha_{CTR}GPO(k|k) \\
GPO(k+2|k) &= \alpha_{CTR}GPO(k+1|k) \\
GPO(k+3|k) &= \alpha_{CTR}GPO(k+2|k).
\end{align*}

The predictor coefficient $\alpha_{CTR}$, where subscript “CTR” denotes “crank-to-run” condition, is a linear spline function of TPS and engine RPM signals:

\[
\alpha_{CTR}(TPS(k), \text{RPM}(k)) = K_0 + \sum_{i=1}^{m}(K_i)TPS_i(k) + \sum_{j=1}^{n}(K_{m+j})\text{RPM}_j(k). \tag{3.5}
\]

Similarly, the notation $GPO(k+i|k)$, means, “GPO prediction at event $k+i$, given information at event $k$”.

A scheduled GPO filter for GPO measurement is implemented during crank-to-run transition; the two equations associated with the GPO filter are:

\begin{align*}
GPOF(k) &= (0.1)GPOF(k) + (0.9)GPOM(k) \tag{3.6} \\
GPOF(k) &= (0.9)GPOF(k) + (0.1)GPOM(k) \tag{3.7}
\end{align*}

where $GPOF$ and $GPOM$ denote the filter GPO and the measured GPO, respectively. Equation 3.6 is used for normal engine start, whereas Equation 3.7 is used for anomalous engine start such as misfire and poor-start. The measurement update equation for the crank-to-run predictor is

\[
GPO(k|k) = \alpha_{CTR}GPO(k-1|k) + L(GPOF(k) - GPO(k|k - 1)), \tag{3.8}
\]

80
where $L$ denotes the estimator gain. In the above expressions, the GPO measurement at event $k$, $GPOM(k)$, is calculated using

$$GPOM(k) = VE_{RUN} \frac{MAP(k)}{IAT(k)}$$  \hspace{1cm} (3.9)$$

where $VE_{RUN}$ is the volumetric efficiency at running conditions, obtained by a separate engine calibration process (assumed known).

### 3.3.3 Run GPO Predictor

When the engine crankshaft speed passes the RPM flare and the intake manifold pressure reaches the bottom of the MAP “well”, a mass balancing characteristic between GPC and GPO is nearly achieved. As discussed in Chapter 2, the Run mode can be distinguished around event 50. It is approximately at this point that we “hand-off” the air prediction task to whatever scheme is used for the Run regime. The equations associated with the run GPO predictor are omitted here, because we are not concerned with this aspect of the problem in this dissertation research.

### 3.3.4 Misfire, Poor-Start GPO Predictor

Whenever misfire or poor-start is detected in an engine start, the mode of the filtered GPO is scheduled accordingly. The GPO prediction is carried out according to the following rules:

$$GPO(k + 1|k) = \alpha_{m,p}GPO(k|k)$$

$$GPO(k + 2|k) = \alpha_{m,p}GPO(k + 1|k)$$

$$GPO(k + 3|k) = \alpha_{m,p}GPO(k + 2|k).$$

The parameter, $\alpha_{m,p}$, can be hand-tuned (calibrated).
3.3.5 Transition Rules

Defined by their occurrence, four modes during engine start and crank-to-run transition are depicted in Figure 3.13. The misfire/poor-start predictor is activated when its corresponding mode is detected.

![Engine Start Process Measured in Time](image)

Figure 3.13: Three phases in a typical engine start

For I-4 engines, the rules to define the transition between modes are summarized below:

- With known CAM position, Event 4 is the default event for the transition from the crank mode to the crank-to-run mode.

- At Event 4, if the change in RPM is less than a calibratable number (e.g., 200 RPM), the misfire GPO predictor is activated; the anomalous GPO filter and the misfire GPO predictor are used at this event.

- At Event 5, if engine speed is less than a calibratable number (e.g., 700 RPM), the poor-start GPO predictor is activated. At the same time, the anomalous GPO filter is activated. Otherwise, the normal GPO filter and the crank-to-run GPO predictor are activated.
• If the engine speed passes the calibratable RPM threshold (e.g., 1400 RPM), either from a poor-start recovery mode or normal start mode, the prediction scheme switches to the run GPO predictor.

The state-flow chart in Figure 3.14 illustrates the causal logical relationship of the rules; note that the algorithm is in State 4 only once, whereas it loops in State 5 thereafter. For engines with more than four cylinders, similar but modified rules can be applied.

Figure 3.14: Transition Rules

A timing chart for prediction is given in Figure 3.15. The key points in Figure 3.15 are to convey the information that

• The GPO prediction is gradually shifted from the crank GPO predictor to crank-to-run GPO predictor. After only a few cranking events, the crank-to-run GPO predictor is starting to kick in.
- The three GPO predictions are made simultaneously but used for different cylinders.

- As the intake event is approached, more accurate GPO prediction is achieved.

This figure only reveals part of the whole story of the predictive fuel dynamics control.

In Section 3.6 of this chapter, we will revisit this figure.

Figure 3.15: Timing chart of GPO prediction
3.4 Nominal Fuel Dynamics Compensator

In this dissertation research, the nominal fuel dynamics compensator is formulated as the inversion of the nominal fuel dynamics model. As explained in Chapter 2, parameters in

\[ y(k) = -\beta_1 y(k - 1) + \alpha_0 u(k) + \alpha_1 u(k - 1) \]

are influenced only by ECT during engine start and crank-to-run transition. In the above expression, \( y(k) \) and \( u(k) \) denote MBFM and CINJ at cycle “\( k \)”, respectively. By inverting the above equation, the nominal fuel dynamics compensator is obtained:

\[ u(1) = -\frac{\alpha_1}{\alpha_0} u(k - 1) + \frac{1}{\alpha_0} y(k) + \frac{\beta_1}{\alpha_0} y(k - 1), \quad (3.10) \]

where \( y(k) \) and \( u(k) \) become commanded MBFM and commanded CINJ (compensator output), respectively. When the nominal fuel dynamics compensator is initialized with zero initial conditions, the first cycle fueling command prior to lost fuel correction for a cylinder is

\[ u(1) = \frac{1}{\alpha_0} y(1). \quad (3.11) \]

It is worth noting that the above equations are used for fueling only one cylinder; the rest of cylinders are treated identically but with different current and past fuel commands and compensator outputs. In other words, the following equations are actually used in the nominal fuel dynamics control for I-4 engines:

\[
\begin{align*}
    u_{\text{cyl}#1}(k) &= -\frac{\alpha_1}{\alpha_0} u_{\text{cyl}#1}(k - 1) + \frac{1}{\alpha_0} y_{\text{cyl}#1}(k) + \frac{\beta_1}{\alpha_0} y_{\text{cyl}#1}(k - 1) \\
    u_{\text{cyl}#2}(k) &= -\frac{\alpha_1}{\alpha_0} u_{\text{cyl}#2}(k - 1) + \frac{1}{\alpha_0} y_{\text{cyl}#2}(k) + \frac{\beta_1}{\alpha_0} y_{\text{cyl}#2}(k - 1) \\
    u_{\text{cyl}#3}(k) &= -\frac{\alpha_1}{\alpha_0} u_{\text{cyl}#3}(k - 1) + \frac{1}{\alpha_0} y_{\text{cyl}#3}(k) + \frac{\beta_1}{\alpha_0} y_{\text{cyl}#3}(k - 1) \\
    u_{\text{cyl}#4}(k) &= -\frac{\alpha_1}{\alpha_0} u_{\text{cyl}#4}(k - 1) + \frac{1}{\alpha_0} y_{\text{cyl}#4}(k) + \frac{\beta_1}{\alpha_0} y_{\text{cyl}#4}(k - 1)
\end{align*}
\]
3.5 Inverse Utilized Fuel Fraction

The direct inversion of the UFF function without any modification is shown in Figure 3.16. In the forward direction, the UFF function is a one-to-one mapping. When inverted as control, however, the one-to-one mapping property is not retained simply due to the saturation effect of the forward UFF function. It is highly possible to have a situation that the commanded CINJ (output of the nominal fuel dynamics compensator) is so large such that there is no corresponding RINJ within a practical range (see Figure 3.16).

![Figure 3.16: Inverse UFF](image)

In addition to this incomplete mapping issue, a sensitivity issue is inherent to the inverse UFF function (see Figure 3.16): small swing on CINJ can result in a huge swing on RINJ. This sensitivity will be worse when the commanded CINJ is varying near the saturation limit. Thus, without a proper modification, the inverse
UFF function cannot be used in control directly. To respond to these two issues, the inverse UFF function is modified with a remedy shown in Figure 3.17.

![Clipped Inverse UFF](image)

**Figure 3.17: Clipped Inverse UFF**

Although this small fix is simple and intuitive, the sensitivity issue is solved effectively. In addition, this clipping limit on RINJ, when CINJ is sufficiently large, shown in Figure 3.16, implies a profound impact on control: whenever the fuel command (commanded CINJ) is requesting a value beyond 90% capability for a given ECT\(^{14}\), no more effort is made to achieve such an unreasonable physical fueling level by increasing RINJ; rather, the strategy is to maintain the effort which corresponds to 90% of its capability. This point makes sense for control implementation and calibration.

\(^{14}\)Under a certain ECT, the amount of vapor fuel mass achievable near the intake port and inside the cylinder has an upper limit, regardless of how much fuel is injected.
The forward UFF function is a two-input, one-output nonlinear function. In order to invert together with the clipping limit, numerical methods would be the best choice to approximate the inverse function. Figure 3.18 depicts this inversion requirement.

![Inverse UFF function with clipped output](image)

**Figure 3.18: Inverse UFF function with clipped output**

Based on the UFF formulation and a known function $UFF_{20}(ECT)$, the following function is actually inverted:

$$T = \frac{CINJ}{UFF_{20}(ECT)} = \left(1 - \frac{2}{\pi} \arctan \left( \frac{RINJ}{\gamma(ECT)} \right) \right) RINJ.$$ (3.12)

The implementation of this inversion scheme is shown in Figure 3.19. The parameter $\gamma(ECT)$ is a simple function of ECT and can be stored in a table. A numerical approximation of the inverse $T$ function is plotted in Figure 3.20. This two-input, one-output function (see Figure 3.20) can be efficiently implemented by a linear spline
Figure 3.20: The inverse $T$ function with clipped output

function. By doing so, two divides, two multiplies and one sum are required to complete the computation of the inverse $T$ function.

As discussed previously, the first cycle UFF function can be a single number for a given ECT. Therefore, the final fuel control command will be as simple as

$$RINJ(1) = \frac{CINJ(1)}{UFF(1)}.$$  

3.6 Complete Predictive Fuel Dynamics Control

Based on the GPO prediction scheme via three iterations, the fuel control command is computed three times sequentially in each fueling cycle for a given cylinder. As such, the event based GPO predictions are required to be systematically assigned to corresponding cylinders at each engine event. Figure 3.21 depicts a simple logic for GPO assignment for I-4 engines. As shown, at any single event, the three step
GPO predictions are made simultaneously. Moreover, in each fueling cycle for a given cylinder, three fueling commands are issued: the latest fuel command is computed using the most accurate, predicted GPO information.

3.6.1 Logic for GPO Assignment

To better convey this innovative fuel dynamics control scheme, a detailed GPO assignment procedure is explained below.

**Key-On:** At this event, the ECU is awakened by the driver action, such as by turning the ignition key. The absolute atmosphere pressure is measured (MAP sensor measurement) to compute $GPOM_0$(Key-On GPO measurement). The fuel command may be computed using the crank GPO predictor and delivered...
as prime fuel pulses. This has been noted as an option in Figure 3.21 to retain the legacy approach used in current calibration. The reason for priming only two cylinders is that the KCPA may be off by one engine event; priming these two cylinders (cylinder 3 and cylinder 4) will guarantee closed valve injection when mis-synch occurs\(^{15}\).

**Event 1:** At this event, mis-synch correction is carried out and three step ahead GPO predictions are made. As shown in Figure 3.21, the 2\(^{nd}\) step ahead GPO prediction is assigned to cylinder 3 and the 3\(^{rd}\) step ahead GPO prediction is assigned to cylinder 4. It is worth noting that all three GPO predictions are made by the crank GPO predictor at this engine event because cylinders 3 and 4 will complete the intake stroke under cranking speed. Because no cylinder will fire at the next engine event, only the 3\(^{rd}\) and 2\(^{nd}\) step head GPO predictors are activated.

**Event 2:** Because cylinder 2 will be the third cylinder to fire (based on the firing order), the intake stroke takes place during crank-to-run transition. Thus, the crank-to-run GPO predictor, shown in Red in Figure 3.21, is used to compute the 1\(^{st}\) step GPO prediction, whereas cylinder 3 and cylinder 4 keep progressing toward the intake stroke, and the 1\(^{st}\) and 2\(^{nd}\) step ahead GPO predictions, made by the crank GPO predictor, are assigned respectively.

**Event 3:** At this event, the 2\(^{nd}\) and 3\(^{rd}\) step ahead GPO predictions are made by the crank-to-run GPO predictor for cylinder 2 and cylinder 1, respectively; the

\(^{15}\)For an I-4 engine at any arbitrary parking position, only one cylinder is in the intake stroke (intake valve(s) is opened); therefore, priming the other three cylinders can guarantee closed valve injection if the cylinder which is in intake stroke is known.
1\textsuperscript{st} step ahead GPO prediction is still made by the crank GPO predictor for cylinder 4.

**Event 4:** About 60°-70° CA after Event 3, the first firing should occur and a sensible RPM increase can be detected at this event; otherwise misfire (including weak fire) is detected and corresponding action will be taken. At this point, either the crank-to-run predictor takes care of all three GPO predictions or the misfire GPO predictor takes over.

**Event 5:** At this event, if RPM is below a predefined threshold, poor-start is detected and poor-start GPO predictor takes care of all three GPO predictions; otherwise, the crank-to-run predictor takes over.

If engine RPM passes a predefined threshold for the RUN mode, the run GPO predictor will take over until the RUN algorithm takes over.

### 3.6.2 Fuel Dynamics Control Initial Condition Setup

To expand on the information embodied in Figure 3.21, one important fact of fuel dynamics control is worth emphasizing via the following question: what is the initial condition for the fuel dynamics control for next fueling cycle if a cylinder has completed the fueling requirement? For example, what is the initial condition for the fuel dynamics compensator of cylinder 3 at Event 5? Due to the fact that we employ three iterations of fuel dynamics control, three sets of initial conditions (three past CINJs) are created sequentially and all of them can be used for the initial condition setup for the next fueling cycle. Three options are supplied:

1. Use the largest CINJ among the three past values of CINJs.
2. Use the mean value of three past values of CINJs.

3. Use the true delivered CINJ (by the engine hardware) as the initial condition.

The comparative impact of the above three choices is left for further investigations.

3.7 Summary

This chapter begins with an unconventional review of research development in a form of lessons learned practice. By analyzing problems and failures encountered in the dissertation research, feasible solutions for each subproblem emerged gradually. This led to a clearer picture in the evolution of the final control design.

By means of the multi-step GPO prediction scheme, the cylinder air rate prediction is iterated at each engine event, facilitating individual cylinder AFR control. Fuel dynamics effects of three different cylinders are compensated simultaneously through three independent inverse fuel dynamics compensators, given three predicted GPOs. By commanding a desired exhaust EQR target, the predictive fuel dynamics control realizes an approximate unity gain response. Moreover, the misfire/poor-start detection and corresponding GPO predictor respond to abnormalities in engine start, and proper rescuing actions are undertaken by enriching the immediate next firing event, therefore rendering robustness in the face of anomalous engine starts.

Throughout this chapter, the structure and detailed description of each component of the predictive fuel dynamics control are elucidated. The next immediate task would be to obtain this control; namely, undertaken control calibration. However, in order to dive into this important issue, which is discussed in Chapter 5, the identification methodology suitable for control calibration is introduced first in Chapter 4.
CHAPTER 4

IDENTIFICATION METHODOLOGY FOR CALIBRATION

Generally speaking, the main methodology of control design in this dissertation research is model based control, cast in a predictive framework. As such, a key step, namely system identification, carried out well ahead of the control design process, dictates the whole control development. That is to say, in the cylinder air rate prediction and the fuel dynamics control developed herein, air and fuel loop models are essential. Hence, identification of these models becomes a central focus of this research. In particular, for the predictive open loop fuel dynamics compensation problem attacked in this dissertation research, accuracy of the identified models is even more critical. The saying, the control will not be better than the process model, reflects this important artifact of open-loop, feed forward control.

In today’s automotive industry, system identification together with a systematic procedure throughout data collection to final control implementation is called control calibration. This may sound a little different from what it has been called in the past, where engine control, such as engine start control, is typically not model based (control), but rather governed by rules or hand-tuning algorithms based on the best engineering practice. As such, there is no distinction between calibration and
control\textsuperscript{16}. "Putting a number in to make it work" would be a concise characterization of start control calibration in many current production algorithms. To obtain this type of control, significant amount of calibration effort may result, depending on calibrators' skills. In contrast, model based control is more appealing because of high efficiency in the process of control calibration, and becomes a main trend in current automotive powertrain control development.

This chapter servers as a bridge between Chapter 3 (Predictive Fuel Dynamics Control) and Chapter 5 (Calibration of Fuel Dynamics Control). As such, this chapter is somewhat philosophical in nature, with a primary focus to overview system identification methods, and the concept of approximation. Several common methodologies of identification suitable for this research problem are reviewed, followed by several questions raised in this dissertation research. By explaining these questions and clarifying ambiguity, a promising concept emerges and is introduced for linear, discrete time, Multiple Input, Single Input (MISO) systems. After a simple conceptual example, differences among various identification methods are displayed and the essences of subspace methods in ARMA model identification are discussed. Because the main focus in this research has been to solve a practical industrial problem, exploring new methodologies for system identification has been application driven, and only preliminary work (from the theoretical point of view) has been accomplished. Therefore, relevant proofs and statistical properties are not presented here, but await future development.

\textsuperscript{16}The quality of control is mostly justified by visual, acoustic, tactile feelings and eye-balling exercise of calibration personnel; the final decision of calibration usually is the final control, going into production.
4.1 Common Tools in Identification

Because of the problem characteristics dealt with in this research, classical system identification methods, mostly stemming from the least squares algorithm, take on a central focus, thereby excluding frequency domain methods. When narrowed down to linear model (linear in parameter) identification, candidates are even more limited [23]:

**Least Squares:** The fundamental idea behind the least squares algorithm is vector projection in solving the $Ax \approx b$ problem\(^{17}\). The least squares solution is obtained by projecting a vector $b$ onto the columns of $A$. When solving the least squares problem, matrix inversion is implicitly involved (QR decomposition). Therefore, the columns of $A$ ($m \times 1$ vectors) must be well conditioned; otherwise, the solution $x$ is very sensitive to the variation of data in $A$. Other than that, the least squares can easily and reliably produce a solution. It is worth noting that, by solving the least squares problem, partial information contained in vector $b$ is lost by the projection operation and the least squares solution $x$ corresponds to the coordinates of the *image* of the vector $b$ in $A$. As such, however, no information in $A$ is lost [21].

**Total Least Squares (TLS):** Classified as a variant of the least squares method [12], the total least squares method uses the singular value decomposition to find out a solution to an inconsistent system of linear equations [22]. Instead of solving $Ax \approx b$ problem, the total least squares method solves $[A|b]y \approx 0$ problem, by eliminating the smallest singular value so as to reconstruct a set of linearly

\(^{17}\)In our discussion, $A$ is a $m \times n$ matrix, $b$ is a $m \times 1$ vector and $x$ is a $1 \times n$ vector.
dependent vectors out of $[A|b]$. In general, all reconstructed vectors rotate from vectors in $[A|b]$ to create linearity; therefore information in $[A|b]$ is altered. The weakness of the total least squares method lies in the fact that, the smallest singular value must be sufficiently small such that it is really due to noise influence alone, rather than structural collinearity. Otherwise, the relatively large value of the smallest singular value may indicate incorrect model structure, poor signal to noise ratio of measurement data, or both. However, to distinguish the noise influence and structure collinearity is a very difficult task in most practical applications. This is actually the key limitation of TLS algorithm.

**Instrumental Variables Method:** Similar to solving the normal equation in a least squares problem, such as $x = (A^TA)^{-1}A^Tb$, the matrix inversion operation uses a different, uncorrelated matrix $C$ (the same dimension as $A$) such that $x = (C^TA)^{-1}C^Tb$. The choice for matrix $C$ is infinite if $C^TA$ is not singular, as is the solution $x$. In practical applications, the matrix $C$ is suggested to be obtained by the same experiment but with different input sequences (uncorrelated with the previous test) [5] [49]. However, the uncorrelated correspondence (matrix $C$) is still not unique and the variation in parameter estimates is not well understood. Besides, experimental requirements such as different excitation signal (uncorrelated input sequences) may be problematic for certain problems such as the one dealt with in this research.

**Maximum likelihood:** In this method, the main objective is to maximize the inherent probability distribution of the fitting errors by tuning the model parameters, based on certain assumptions on model structure and a probabilistic model. The
underlying assumption is that the probabilistic distribution of the error signal must be known \textit{a priori}. Viewed as a rigorous identification method for linear or quasi linear identification problems, extension to nonlinear system is still an open research problem. In addition, the maximum likelihood method carries a property inherently connected to sample space size: for systems with data richness issues, robustness of the parameter estimates is problematic.

\textbf{Parameter Set Estimation:} In certain control problems such as robust control, the main concern in system identification is often more about a set in which the true parameter could stay, rather than a point estimate [6]. When used in robust control design, the \textit{size} of the parameter set dictates the stability margin and performance of the closed loop system; of course, the \textit{center} of the set is frequently used in nominal controller design [7]. Due to the requirement for open loop compensation in this research problem, set estimation algorithms were not explored but remain promising for future work.

\textbf{Summary of Identification Methodologies}

Objectives and characteristics of different engineering problems dictate which of the above methodologies can be justified for implementation. In general, there is no single answer to the question as to which is the best among these methods. However, as observed in many practical applications, the least squares method with a prediction error structure is commonly used, often times simply because the results are sufficient, and therefore the methodology is unchallenged.
4.2 Question and Ambiguity

For open-loop control problems (such as in this work), where system identification of forward and inverse model structures is intimately tied to control performance, questions and ambiguities arise which are not typically seen in normal system identification exercises. In such cases where poor explanation of mismatch exists between the physical model and experimental data, the concept of system identification is weakened; in those cases, we would say that approximation prevails. As the underlying physical concept gradually fades away, physical causality relationships among data can be reduced as well. As a result, a clear distinction can be lost between the forward model and the inverse model; meanwhile, we may be perplexed by numerous choices.

**System Identification or Approximation?** An important aspect of system identification is referred to as structure/model identification. In addition to obtaining a set of parameters through a certain criteria (identification method), system identification also evaluates the quality and accuracy of the underlying model structure, simultaneously; if the underlying structure of the model is well imposed, system identification makes perfect sense. However, in many automotive applications, the true model structure is either unknown or too complicated for use in control; then the role of system identification is weakened. Under this situation, approximation plays a major role in the modeling process and is frequently used to describe the input/output relationship of engine dynamics, without emphasizing the origin of the underlying model structure. As such, the approximation leans, more or less, toward dynamical regression (such as the
ARMA model structure). In particular, the linear spline modeling technique, introduced in Chapter 2, is a perfect example of this concept. Although, in realization, the two concepts of identification and approximation differ only in motivation and underlying assumptions, the impact on control design is very different: control design based on a mathematical approximation, without dealing with its underlying physics or appropriate interpretation of input, output, and state variables, represents a “big leap” in model based control design, strongly conflicting with classical model based control using first principles.

In this dissertation research, a major objective is to capture key air loop dynamics and to overcome profound nonlinear fuel dynamics effects. Due to the inherently short time scale and limited control authority, a reasonably good approximation of the air loop dynamics would be a best choice. Without validating the approximation idea in model based control from a pure theoretical approach\textsuperscript{18}, to progress for now in this research we conclude that, if a mathematical approximation of the target dynamics is sufficiently accurate in such a way that the cause-effect relationship is well comprehended and can be easily ported into control design and implementation, there is no serious concern about the exact physical meanings of the model structure and coefficients. An expansion of this statement is an open problem and is left for future research.

**Ambiguity in ARMA Model Identification.** Given input/output data from a dynamical process obtained through experiments, a straight forward modeling method would be to deploy the ARMA model structure, finding a reliable relationship, among current, past input/output measurements. Namely,

\textsuperscript{18}Left for future investigation.
this deployment consists of picking a system order determined by two arbitrary numbers \( m, n \) such that an \( m \)-pole, \( n \)-zero model can be reasonably constructed. Also, the following approximate expression is usually used to inspect the input/output measurements, \( u \) and \( y \), which may be corrupted by noise or disturbance:

\[
y(k) \approx -\beta_1 y(k - 1) \ldots - \beta_m y(k - m) + \alpha_0 u(k) \ldots + \alpha_n u(k - n). \tag{4.1}
\]

As shown in (4.1), the primary goal of identification is to find a parameter vector \([\beta_1 \ldots \beta_m \alpha_0 \ldots \alpha_n]^T\) to minimize the prediction error \( \varepsilon(k) \) in

\[
y(k) = -\beta_1 y(k - 1) \ldots - \beta_m y(k - m) + \alpha_0 u(k) \ldots + \varepsilon(k). \tag{4.2}
\]

The immediate above equation has been referred to as the ARX model and \( \varepsilon(k) \) is viewed as an extra input, presumably possessing a property similar to white noise. The least squares method, by far, is a “default” choice in solving this type of problem. Because this model can be viewed as minimizing the error between true and simulated output, the model parameters, therefore, are viewed in the perspective of the forward direction, thereby resulting in a forward model. However, with a slight modification of the approximation in (4.1), a different approximate expression is produced below, which corresponds to the identification of the inverse model:

\[
u(k) \approx -\frac{\alpha_1}{\alpha_0} u(k - 1) \ldots - \frac{\alpha_n}{\alpha_0} u(k - n) + \frac{1}{\alpha_0} y(k) \ldots + \frac{\beta_m}{\alpha_0} y(k - m) \tag{4.3}
\]

In many open loop compensation problems, the inverse model is actually the controller. The least squares solution to the approximate expression (4.3) always exists, given enough measurements\(^{19}\). By analogy, any measurement can be put

\(^{19}\)Here, enough measurements essentially means the measurement data are rich enough and the system is persistently excited.
on the Left Hand Side (LHS) as well, such as

\[
y(k - i) \approx -\frac{1}{\beta_i} y(k) \cdots - \frac{\beta_m}{\beta_i} y(k - m) + \frac{\alpha_0}{\beta_i} u(k) \cdots
\]

\[
u(k - j) \approx -\frac{\alpha_0}{\alpha_j} u(k) \cdots - \frac{\alpha_n}{\alpha_j} u(k - n) + \frac{1}{\alpha_j} y(k) \cdots.
\] (4.4)

Among the finite number of possible permutations, solutions always exist, but the concept of the forward model or inverse model is lost. One sensible choice, that can obtain a nonzero solution of the parameter vector \([\beta_1 \cdots \beta_m \alpha_0 \cdots \alpha_n]^T\), is by means of the TLS method, according to

\[
y(k) \cdots + \beta_m y(k - m) - \alpha_0 u(k) \cdots - \alpha_n u(k - n) \approx 0.
\] (4.5)

As seen, “\(m + n + 3\)” solutions (in total) to the ARMA model identification problem are available. But the solutions to the structures represented in (4.1), (4.3), (4.4) and (4.5) are highly unlikely to be the same (factorization may be required to make comparison)!

To date, the open literature has not given, in a rigorous mathematical sense, a strong and valid reason to pick one solution against the others. Therefore, the question of which solution should be trusted becomes a nagging annoyance.

The end result is to identify the fuel dynamics controller directly in this research (inverse open loop control) without identifying the forward fuel dynamics model as a first step, and then following the path of model based control.

Because of the doubts that arose, and the ambiguities present in this forward/inverse modeling philosophy, we continued to pursue general solutions to these problems and started from another property commonly shared by the above methods: the angle view of data streams. In so doing, some interesting discoveries have arisen in a class
of methods loosely characterized as “subspace methods” for identification and approximation.

4.3 Angle View of Linear Dynamic System

Identification of an ARMA model from experimental data can be very difficult if the input and output measurements are corrupted with noise. Due to the linearity of this specific model structure used in identification, the noise from both the input and the output, presumably possessing different distributions, must be minimized so as to identify the model parameters meaningfully. In addition, because control design is mostly based on properties determined by pole/zero locations of the model, two relationships are studied together: 1) relation between current and past inputs, and 2) relation between current and past outputs.

Again, the standard ARMA model is given by

\[ y(k) + \beta_n y(k - n) = \alpha_0 u(k) + \alpha_m u(k - m), \quad (4.6) \]

where \( u \) is the system input, and \( y \) is the system output. Due to the measurement noise and disturbances, the consistency of the equation is distorted and we must identify the parameters of

\[ y(k) + \beta_n y(k - n) \approx \alpha_0 u(k) + \alpha_m u(k - m). \quad (4.7) \]

After collecting \( p \) measurements, we can arrange the measurement sequences in matrix form to arrive at

\[ Y(k) + \beta_n Y(k - n) \approx \alpha_0 (U(k) + \frac{\alpha_m}{\alpha_0} U(k - m)), \quad (4.8) \]

where \( Y(i) \in \mathbb{R}^{p \times 1} \) and \( i = 1 \cdots n; U(j) \in \mathbb{R}^{p \times 1} \) and \( j = 1 \cdots m \) (assuming \( \alpha_0 \neq 0 \) and \( p > \max(m, n) \)). If we collect the model parameters into vectors, the approximation
can be written as

\[
Y\beta \approx \alpha_0 U\alpha, \quad (4.9)
\]

where \( \beta = [1 \cdots \beta_n]^T, \alpha = [1 \cdots \alpha_m/\alpha_0]^T, \ Y = [Y(1) \cdots Y(n)] \) and \( U = [U(1) \cdots U(m)] \). \( \beta_i \) and \( \alpha_j \) are unknown and to be identified. For convenience, the LHS and the Right Hand Side (RHS) of (4.9) are defined as \( l = Y\beta, \ r = U\alpha \).

In an ideal situation, \( l = \alpha_0 r \). However, in a more general case, there always exists an angle between the vector \( l \) and the vector \( r \), or

\[
l \approx \alpha_0 r. \quad (4.10)
\]

Such an interesting concept is depicted in Figure 4.1. The angle \( (\theta_{\text{min}}) \) supplies a “best” measure of potential influence of the inputs on outputs. For example, if the angle is close to 90°, it can be said that an insufficient model structure is specified for the problem, given the measurement data. That is to say, the smaller the angle, the better the relationship is captured by the specified model structure. Therefore, in order to take full advantage of this unique measure, altering the model system

![Figure 4.1: Minimal angle between two subspaces](image-url)
order iteratively by changing the number of poles and zeros, while simultaneously observing the changes of the angle \( \theta_{\text{min}} \), could serve as a system order indication. It is worth emphasizing that this step actually detects this cause-effect relationship in a more rigorous mathematical sense. More importantly, when irrelevant information is added into the process, such as adding extra past input/output measurements or different inputs in the case of MISO model identification, the minimal angle is not affected. Unlike identifying a model using the ARX model structure, the reduced prediction error does not necessarily imply a correct choice of model order or structure, by means of adding extra terms of past input/output measurements.

At this point, it appears to us that the discovery and introduction of the angle concept (between subspaces) for the purpose of ARMA model identification/approximation is a contribution to the field of system identification, and carries interesting potential.

Continuing with the concept in Figure 4.1, to arrive at an approximate solution, we could project one vector onto the other. For instance, we can project \( \mathbf{l} \) onto \( \mathbf{r} \) to find \( \alpha_0 \). If the parameters, \( \alpha_i \) and \( \beta_j \), are known \textit{a priori}, this projection is quite easy. However, without knowing the parameters beforehand, an alternative choice is to minimize the angle between \( \mathbf{l} \) and \( \mathbf{r} \) by “guessing” the value of the parameter vector. This can be done, for example, by optimization iteration.

Motivated by this idea, the angle between the vectors \( \mathbf{l} \) and \( \mathbf{r} \) can be calculated from the expression

\[
\cos \theta = \frac{\mathbf{l}^T \mathbf{r}}{\|\mathbf{l}\|_2 \|\mathbf{r}\|_2} = \frac{\beta^T \mathbf{Y}^T \mathbf{U} \alpha}{\|\mathbf{Y} \beta\|_2 \|\mathbf{U} \alpha\|_2}, \tag{4.11}
\]

where \( \| \cdot \|_2 \) is the Euclidian vector 2-norm. We would like to find the minimum angle with respect to a specific parameter vector \( [\alpha \beta]^T \).
Based on the results developed in [9], we have derived the following theorem and Corollaries, proceeded by definitions, in which \( \text{range}(U) \) is defined as the span of the set of columns, \{\( U(1) \cdots U(n) \)\} (similarly for \( Y \)).

**Definition 1.** The input subspace \( \mathcal{U} \) is defined as \( \mathcal{U} = \text{range}(U) \) and the output subspace \( \mathcal{V} \) is defined as \( \mathcal{V} = \text{range}(Y) \).

**Definition 2.** The QR decomposition of \( Y \) and \( U \) are \( Q_Y R_Y \) and \( Q_U R_U \), respectively. \( Q_Y \) and \( Q_U \) are the economically sized decompositions of \( Y \) and \( U \), respectively.

**Definition 3.** \( S_n = Q_Y d_y \) where \( \|d_y\|_2 = 1 \) and \( d_y \in \mathbb{R}^n \); \( S_m = Q_U d_u \) where \( \|d_u\|_2 = 1 \) and \( d_u \in \mathbb{R}^m \). \( S_n \) and \( S_m \) are called the unit \( n \)-sphere and \( m \)-sphere in subspaces \( \mathcal{V} \) and \( \mathcal{U} \), respectively.

Based on the above three definitions, the following theorem can now be introduced.

**Theorem 1.** The solution to the minimization problem

\[
\cos(\theta_{\text{min}}) = \max_{\alpha \in \mathbb{R}^m} \frac{\beta^T Y^T U \alpha}{\|Y \beta\|_2 \|U \alpha\|_2} = \max_{d_y \in S^n, d_u \in S^m} d_y^T Q_Y^T Q_U d_u
\]

always exists and is given by

\[
\theta_{\text{min}} = \arccos \sigma_1,
\]

where \( \sigma_1 \) is the largest singular value of \( Q_Y^T Q_U \). The singular value decomposition of \( Q_Y^T Q_U \) is

\[
Q_Y^T Q_U = Y \Sigma U^T
\]

where \( \Sigma = \text{diag}(\sigma_1, \sigma_2, \cdots, \sigma_m) \) and \( 1 \geq \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{m-1} \geq \sigma_m \). Under a loose condition, such as that which arises in the system identification problem of a physical plant, the solution can be unique.
Proof. A conventional proof for this theorem would follow that of [9] and would be formulated as a generalized eigenvalue problem because \( \cos(\theta_{\min}) \) is indeed a ratio of two polynomials [9] with respect to \( \alpha \) and \( \beta \). However, here we sketch a geometric approach to the proof which is more appealing and easier to understand.

The maximizer

\[
\max_{\beta \in \mathbb{R}^p} \max_{\alpha \in \mathbb{R}^m} \frac{\beta^T Y^T U \alpha}{\|Y \beta\|_2 \|U \alpha\|_2}
\]

\( (4.15) \)

can be further simplified by normalizing \( \|Y \beta\|_2 \) and \( \|U \alpha\|_2 \) to be unit length. As such, we are not concerned about the denominator (because it is identical to “1”) and we can concentrate on the numerator alone. This can be viewed as a process in which \( \alpha \) and \( \beta \) are allowed to take arbitrary values such that two unit vectors aligned with \( \alpha \) and \( \beta \) can form a smallest angle. Then, the maximizer becomes

\[
\max_{d_y \in S_n} \max_{d_u \in S_m} d_y^T Q_Y^T U^T Q_U d_u.
\]

\( (4.16) \)

The maximum value of the above quantity is solely dependent on the largest singular value of \( Q_Y^T U^T Q_U \). As a result, \( \alpha \) and \( \beta \) must be aligned at two specific directions so as to form such a smallest angle. This information is stored in the first columns of \( Y \) and \( U \), respectively.

**Corollary 1.** If the intersection of two subspaces \( U \) and \( Y \) is not empty\(^{20}\), there exists exact solutions to the approximation (4.7).

\(^{20}\) In a real application, the condition of existence of the exact solution indicates that there exists a severe collinear problem in the data, or an incorrect model structure was chosen. For instance, one column in \( Y \) is an image of \( U \).
Corollary 2. Given a set of the ARMA models $S = (s_1, s_2, \cdots, s_t)^{21}$ and data matrix $(U, Y)$, the optimal model structure $s_{opt}$ is

$$s_{opt} = \min_{s_1, \cdots, s_t} (\theta).$$

(4.17)

The minimal angle supplies a quality measure of the model structure given fixed measurement data.

After a conceptual example to follow next, the implications of such an angle concept will be fully revealed.

4.4 Examples for Angle between Subspaces

For convenience, a 1-pole, 1-zero model such as

$$Y(k) + \beta_1 Y(k - 1) = \alpha_0 U(k) + \alpha_1 U(k - 1)$$

is used to examine the different identification schemes in linear model identification.

Again, $Y(i) \in \mathbb{R}^{p \times 1}$ and $i = 1, 2$; $U(j) \in \mathbb{R}^{p \times 1}$ and $j = 1, 2$.

In this example, the relation between four measurement vectors is depicted in Figure 4.2. To make the example interesting, we have chosen these such that a perfect linear condition is artificially created, or

$$Y(k) = \alpha_1 U(k - 1),$$

and $U(k)$ is not orthogonal to $U(k - 1), Y(k - 1)$ and $Y(k)$.

Three conventional methods, closely related to least squares, are used to examine the solutions obtainable from the above simple example:

21This means the models have different numbers of poles and zeros.
1. Solve \( Y(k) \approx -\beta_1 Y(k - 1) + \alpha_0 U(k) + \alpha_1 U(k - 1) \), by least squares:

By projecting \( Y(k) \) onto \( Y(k - 1) \), \( U(k) \) and \( U(k - 1) \), the images of \( Y(k) \) are not zero on \( Y(k - 1) \) and \( U(k) \). Namely, the perfect linear relationship is not retained. This example reveals a key point of the standard least squares method: projecting one vector onto others without discerning collinearity can lead to ambiguity in the approximation (this concept is well known in structure identification). This point can cause a significant overhead in obtaining a reasonably good model structure and corresponding estimates.

2. Solve \( -Y_2 - \beta_1 Y_1 + \alpha_0 U_2 + \alpha_1 U_1 \approx 0 \), by total least squares:

The TLS method, by way of a zero singular value, indicates a perfect linear relationship among vectors. Under this situation, however, TLS does not supply a unique answer, and extra steps are required to discover the structure of the linear relationship.

3. Solve \( Y(k) + \beta_1 Y(k - 1) \approx \alpha_0 U(k) + \alpha_1 U(k - 1) \), by subspace methods:

Other than computing the parameter estimate in solving the problem, subspace
methods evaluate the angle between LHS and RHS as a first step. Clearly, with this step the perfect linear relationship can be easily detected. The subspace methods tell that the model structure has to be

\[ Y(k) = \alpha_1 U(k - 1), \]

and that the remaining measurement vectors are not necessary in formulating the model, which is an attractive feature. In addition to detecting a linear relationship with respect to input and output subspaces, the collinear characteristic among input vectors or output vectors alone will not hinder the angle calculation. This is because collinearity in subspace methods, either within input subspace or output subspace, is equivalent to “no new information” to the model structure. In other words, subspace methods deal directly with the relationship between input and output. If only one subspace is involved in an identification problem, such as in autonomous system identification, subspace methods are identical to TLS methods. After an appropriate model structure is detected, one that is true to the data, a solution can be found by least squares.

### 4.5 Subspace Methods for Linear Model Identification

One may question the purpose of the above example, and claim that it is too specific and not general enough. We would agree, but point out an important implication from the above example: the angle (between LHS and RHS) directly addresses the system properties which are of primary concern in control design, particularly when considering forward and inverse models.

To identify the ARX model (Equation 4.2), in order to get a good match between simulated and measurement error, we could keep adding past inputs and outputs
until we are comfortable with the system order. However, this does not necessarily coincide with what we really want, which is to capture the key model dynamics. As more terms are added, one may find that the cost of adding more terms is to affect the stability robustness of the identified model. This is evident from the fact that the least squares solution does not guarantee stability of the identified model, even though the experimental data indicates stability.

However, when subspace methods are used, there is no burden when extra terms are added on both sides of the model equation. This is, when adding more terms on both sides, if the angle is not reduced significantly, no new information is introduced and those terms should be discarded (not added in). In other words, the data does not indicate the model order. The angle argument, therefore, supplies an interesting measure for model order justification.

On the other hand, when the current output measurement vector $Y(k)$ dominates the angle of the output subspace with respect to the input subspace, the forward model identified using the ARX model structure can realize a reasonably good approximation. The same statement is also valid for the inverse model identification. Roughly speaking, the forward and inverse model identification, by means of the ARX model structure, is to find “relative weights” of the projected vector of LHS on the rest of measurement vectors on RHS of the ARX model equation. When the experimental data is well conditioned and the system is well excited, such as input/output measurements collected from a well excited, unity gained dynamic system, the “weights” (or values of model coefficients) are dictated by the angles, simply because the length of the measurement vectors is approximately the same. This is the key reason why it is sometimes acceptable to identify the forward model for control design in open-loop
(and feed forward) problems. However, to identify an underlying physical process which would be approximated by

\[ Y(k) + 0.9Y(k-1) \approx 0.1U(k) + 0.9U(k-1), \]

deploying the inverse model structure

\[ U(k) = -9U(k-1) + 10Y(k) + 9Y(k-1) + \varepsilon(k) \]

would be inappropriate. This is equivalent to projecting a “less relevant” vector onto the others, thereby losing more relevant information. If the subspace methods are deployed as a first step, the forward model structure, or

\[ U(k-1) = -\frac{1}{9}U(k) + \frac{1}{9}Y(k) + Y(k-1) + \varepsilon(k), \]

would be suggested.

To summarize the above statements, and to shed light on the impact on forward and inverse model identification, it would be a wiser choice to put the most dominating measurement vector (in terms of the angle between input and output subspaces) on the LHS, when using the ARX model structure.

Similar to the SISO ARMA model, the MISO ARMA model can be identified as well, by subspace methods. In this case, the angle between subspaces means the angle between the output subspace and the total input subspaces (treating all input subspaces in a combined subspace).

### 4.6 Unsolved Issues

Although subspace methods shed new light on ARMA model system identification with the concept of the angle between input/output subspaces, the method thus
developed cannot handle constraints effectively. For instance, when a steady state constraint is imposed on the ARMA model, the input/output data are shuffled to reduce an extra parameter. As such, we cannot clearly distinguish input/output subspaces anymore. The capability of subspace methods is then drastically reduced. In addition, numerical issues arise when an angle is small, although the \( \cos \) is not sensitive at small angles. Research on this topic is still developing.

4.7 Summary

This chapter has severed as a segue between the fuel dynamics control, with air prediction, and the implementation aspects of calibration. System identification methods (indeed, approximation) are critical for model-based control, and subsequent reduction in calibration effort.

This chapter has also introduced some relevant philosophy for calibration and approximation, and along the way introduces an interesting criteria when scrutinizing the least squares algorithm and its variants. Because several promising features are embodied by subspace methods, more effort needs to be put into this worthwhile research topic. Although this new criteria does not answer the constrained least squares problem, a review of various identification methods has built a bridge to the next stage, calibration of predictive fuel control. By remembering the key weaknesses and limitations of each identification method, correct action can be taken and errant results can be detected efficiently.
CHAPTER 5

CALIBRATION OF FUEL DYNAMICS CONTROL

Throughout Chapter 1 to Chapter 4, control problems associated with fuel dynamics compensation during transit engine operating conditions are discussed; air loop and fuel loop models during engine crank-to-run transition are developed; a novel predictive fuel dynamics control is presented; system identification methods suitable for control calibration are reviewed together with introduction of a new class of identification methods using the concept of angle between subspaces.

In this chapter, a detailed calibration procedure for the predictive fuel dynamics control is described to conclude this innovative control for engine start and crank-to-run transition. In addition to supplying a step-by-step instruction in data collection, data processing and final control tuning, performance of the predictive fuel dynamics control is supplied to compare against the current control practice. By doing so, the advantage of the new start control algorithm is fully revealed.

The organization of this chapter is as follows: In the first section, calibration of four different GPO predictors and corresponding transition rules are described; in what follows, calibration of the UFF function and the nominal fuel dynamics model are discussed. In the third section, results of control simulation are shown; in the
last section of this chapter, a comprehensive summary of the complete fuel dynamics control is supplied.

5.1 Calibration of Cylinder Air Rate Predictors

Calibration of four GPO predictors at different operating modes is discussed in this section followed by the formulation of calibration strategy for transition rules. For convenience, the I-4 PFI engine with KCPA are used to demonstrate detailed calibration procedure. As a result, the presumed first firing event is always at event 4 unless otherwise specified.

5.1.1 Crank GPO Predictor

During engine start, the time of the crank phase is relatively short. Depending on whether the KCPA is used or not, the total number of engine events in crank phase for a typical engine start is between 3 and 13. Based on the crank GPO model,

\[ GPO(k + 1) = \alpha_{CRK} GPO(k) + (1 - \alpha_{CRK}) GPO(k - 1), \]

\( \alpha_{CRK} \) is the only parameter to be identified and at least three data points in each engine start are required to fulfill this task. Based on the total number of engine events prior to the first robust engine firing (\( \geq 3 \)), the least squares algorithm can be used to identify \( \alpha_{CRK} \) given cranking data with several engine starts. Because of low dimensionality of this identification problem (one dimension), solution of the subspace methods is the same as the least squares solution.

When used in the GPO predictor for the crank phase, \( \alpha_{CRK} \) also is the only parameter used in GPO prediction, since

\[ GPO(k + i|k) = \alpha_{CRK} GPO(k + i - 1|k) + (1 - \alpha_{CRK}) GPO(k + i - 2|k), \]
where \( i = 1, 2, 3 \) and denotes three step ahead GPO predictions. As mentioned previously, due to the short period of crank mode, state estimation is not necessary and, therefore, the measurement update is not implemented. As such, the estimated GPO is identical to the measured GPO such as

\[
GPO(k|k) = GPO_M(k),
\]

where \( GPO_M(k) \) refers to measured GPO at event \( k \). To initialize GPO prediction starting from event 1, the initial condition must be set up properly. An example of initial condition setup is supplied as follows:

\[
GPO(0|1) = V_{ECRK} \frac{MAP(0)}{IAT(0)} \\
GPO(1|1) = V_{ECRK} \frac{MAP(1)}{IAT(0)}.
\]  \hspace{1cm} (5.1)

Here \( MAP(0) \) and \( IAT(0) \) refer to “Key-On” MAP and IAT measurements.

In this dissertation research, experimental data were collected on a production I-4 engine without KCPA and, therefore, the engine usually takes about 11 to 12 engine events to start. Under this condition, MAP reaches steady state conditions, as does GPO. As a consequence, performance of the crank GPO predictor cannot be verified effectively against experimental data. In order to perform a quick sanity check, massaged data\(^{22}\) are generated to test the crank GPO predictor and results are shown in Figure 5.1.

As shown, though prediction error is obvious, the percent error is very small. For the purpose of the crank fuel dynamics control, this result is sufficiently accurate.

\(^{22}\)This is done by merging data of first few events with data of crank-to-run.
5.1.2 Crank-to-Run GPO Predictor

As introduced in Chapter 2, the crank-to-run GPO model takes on a linear splines modeling approach so as to capture TPS and RPM dependant GPO decay rate. The model equation is repeated here for convenience:

\[ GPO(k+1) = (K_0 + \sum_{i=1}^{m} (K_i) TPS_i(k)) + \sum_{j=1}^{n} (K_{m+j}) RPM_j(k)) \times GPO(k). \]

In the above model equation, there is no clear indication of input and output subspaces (autonomous system). Therefore, subspace methods are not applied. However, the least squares method can be used to solve

\[ GPO(k+1) - (K_0 + \sum_{i=1}^{m} (K_i) TPS_i(k)) - \sum_{j=1}^{n} (K_{m+j}) RPM_j(k)) \times GPO(k) \approx 0, \]

using measurement data, and is adopted here. How to robustly identify the linear spline model for autonomous systems is left for future investigation.
In addition to the flexibility characteristics offered by the linear splines modeling technique, an extra "knob" to fine tune the quality of model is the partition of scheduling variables. A partition example of MAP and RPM, with sufficient accuracy for I-4 engines, is given by

\[
MAP \in [20, 50] \cup (50, 80] \cup (80, 110] \\
RPM \in [200, 500] \cup (500, 800] \cup (800, 1500].
\] (5.2)

With such partitions, the crank-to-run GPO model becomes

\[
GPO(k+1) = (K_0 + \sum_{i=1}^{3} K_i TPS(k)_i + \sum_{j=1}^{3} K_3+j RPM(k)_j)GPO(k),
\]

in which seven parameters, \(K_0\) through \(K_6\), must be identified. As discussed above, the parameters \(K_i\) are found using the least squares algorithm.

When used as GPO predictor, the estimator gain \(L\) in

\[
GPO(k|k) = \alpha_{CTR} GPO(k|k-1) + L(GPOM(k) - GPO(k|k-1))
\]

is a fixed number and set to 0.9 all the time due to the well behaved, filtered GPOs.

### 5.1.3 Run GPO Predictor

When run mode is detected, the run GPO predictor is used. The run GPO model for this study is one used in production, with minor modification in choice of estimation gain.

### 5.1.4 GPO Filter

Before the detailed calibration of misfire/poor-start GPO predictor is introduced, it is worth introducing the GPO filter first because of its importance. In general, the
event measurements of intake manifold pressure (MAP) at the first firing events are not changing drastically. That is, at the first firing event, the MAP measurement is used as initial condition for the GPO filter. When the second (presumed) firing engine event arrives, the GPO filter is activated to replace the true GPO measurement until the prediction for the run mode takes over.

The GPO filter has two modes based on engine firing behavior: the normal GPO filter is

\[ GPOF(k) = (0.1)GPOF(k) + (0.9)GPOM(k). \]

and the performance is shown in Figure 5.2 (a). As shown, with the filter gain set to 0.9, there is almost no role played by this filter. On the other hand, when poor-start is detected, the GPO filter is switched to

\[ GPOF(k) = (0.9)GPOF(k) + (0.1)GPOM(k), \]

where the filter gain is set to 0.1 and the large GPO fluctuations shown in Figure 5.2 (b) are filtered out. Of course, other choices for the filter gains are possible, such as 0.8 and 0.2, depending on engine firing behaviors. However, the simulation suggests than 0.1 and 0.9 are two excellent candidates for the purpose of smoothing.

5.1.5 Misfire and Poor-Start GPO Predictors

From the standpoint of functionality, the misfire and poor-start GPO predictors are very similar. It is important to note, however, that the misfire GPO predictor is executed only once, whereas the GPO predictor may be executed many times depending on the future engine crankshaft speed development. In the rest of this subsection, these two GPO predictors are discussed separately.
Figure 5.2: Filtered GPO and true measured GPO

Misfire GPO Predictor: The misfire GPO predictor is governed by a misfire detector. At event 4, the first presumed engine firing is expected to occur. If not, the misfire detector sets a “misfire-flag” and initiates operation of the misfire GPO predictor. This can be justified by a RPM threshold, given changes of crankshaft speed: for I-4 engines, a good candidate for this threshold would be 400 RPM. A practical observation indicates that if misfire does occur at this specific engine event, there is only one possible reason to explain this scenario: air and fuel mixture is too lean to ignite inside the combustion chamber\textsuperscript{23}. The best action to perform under this circumstance is to enrich the fuel requirement to make an immediate rescue action for the upcoming firing event (event 5), by over-predicting GPO in all three-step-ahead GPO predictions. Rather than using the decay rate specified by $\alpha_{CTR}$, the GPO decay rate under misfire ($\alpha_m$) is set to a value very close to “1” (or even identical to “1”). The calibration

\textsuperscript{23}If the air-fuel mixture is on the rich side, the engine will start robustly anyway.
requirement at this point is to achieve, by hand-tuning of $\alpha_m$, a response of the first step ahead GPO prediction that is slightly greater than the true GPO measurement in the worst case of engine start (such as the one shown in Figure 5.2 (b)). By doing so, event 6 and event 7 may also get partially enriched by the second and third step ahead GPO predictions (see Figure 5.3); however, when the engine progresses to event 5, fuel commands for event 6 and event 7 will be corrected when new information is available. Most importantly, the upcoming engine event (event 5) is enriched. In other words, the new fuel control responds to abnormalities intelligently rather than leaving the engine to recover (if possible), as seen in many production algorithms today. Although the decay rate of the misfire predictor is scheduled, the estimator gain is maintained at
0.9 because the GPO measurement is not drastically changed. Meanwhile, the GPO filter is initialized.

**Poor-Start GPO Predictor:** Similarly, the poor-start GPO predictor is governed by a poor-start detector and poor-start recovery detector. A RPM threshold, such as 700, would be a good candidate for both the poor-start threshold and the poor-start recovery threshold. In fact, *poor-start* defines a specific “state” of engine operations. In addition to setting the poor-start flag, the GPO decay rate and estimator gain are scheduled accordingly. The decay rate $\alpha_p$ for a poor-start condition is hand-tuned close to “1”, but less than “1” such that the first step ahead GPO prediction is slightly greater than the true GPO measurement in the worst case of engine start, as shown in Figure 5.4. This requirement actually makes a rescue action for later firing events. The estimator gain is set to 0.9 all the time because the poor-start GPO filter is activated. As shown in Figure 5.4, the engine RPM is below the poor-start threshold (700) at event 5 and the state of *poor-start* is entered. At this event, the third step ahead GPO prediction made for event 8 is pulled up. Similar actions are undertaken by the first and second step GPO predictions. As such, fuel enrichment is achieved by over-predicted GPOs.

### 5.1.6 Calibration of Transition Rules

After going through the detailed calibration procedure of the misfire and poor-start predictors, some of the transition rules are already introduced, but not completely revealed yet. In this subsection, calibration of transition rules is described based on
causal occurrence at each engine event in a typical engine start, including key control actions.

**Key-On:** Measure ambient pressure via MAP sensor for Key-On GPO calculation (for Key-On primes).

**Event 1:** Mis-synch correction if needed. Make the 3\textsuperscript{rd} and 2\textsuperscript{nd} step ahead GPO predictions for first two firing events using only the crank GPO predictor.

**Event 2:** Make the 3\textsuperscript{rd} step ahead GPO prediction using the crank-to-run GPO predictor, while the 2\textsuperscript{nd} and 1\textsuperscript{st} step ahead GPO predictions are made by, still, the crank GPO predictor.
**Event 3:** Make the 3\textsuperscript{rd} and 2\textsuperscript{nd} step ahead GPO predictions using the crank-to-run GPO predictor, while the 1\textsuperscript{st} step ahead GPO prediction is made using the crank GPO predictor.

**Event 4:** Starting from this event, no crank GPO predictor is involved and all three GPO predictions are made by the crank-to-run GPO predictor or the misfire GPO predictor. Misfire detection is performed at this event. If misfire is detected, misfire GPO predictor is used; otherwise, the crank-to-run GPO prediction is used.

**Event 5:** Regardless of what has happened in previous engine events, only the crank-to-run GPO predictor and poor-start GPO predictor are used at this engine event. If poor-start is detected, the poor-start GPO predictor is used for all three step GPO predictions; otherwise the crank-to-run GPO predictor is used. In addition, the corresponding GPO filter is used as well.

**Event 6 and onward:** Unless there is no poor-start detected previously, and RPM is beyond 700, the poor-start GPO predictor supervises the GPO prediction. If poor-start is detected in previous engine events, but RPM is greater than 700, the poor-start recovery is detected and GPO prediction is switched to the crank-to-run GPO predictor. If the engine speed is consistently above 1200 RPM, the run GPO predictor takes care of all three step GPO predictions.

**Summary of Predictors and Transition Rules**

Cylinder air prediction during engine crank-to-run transition is a difficult task; however, four different GPO predictors, three detectors, and a set of transition rules take
good care of air prediction in an iterative fashion under all kinds of engine start
conditions, together with some intelligent handling on misfire/poor-start.

For the purposes of identifying the crank and crank-to-run GPO models, normal
good care of air prediction in an iterative fashion under all kinds of engine start
engine starts are required. However, such data requirement is automatically satisfied
engine starts are required. However, such data requirement is automatically satisfied
when we collect data for the purposes of fuel dynamics model identification, where
when we collect data for the purposes of fuel dynamics model identification, where
about 40 engine starts at various ECTs are performed. Therefore, there is no need
about 40 engine starts at various ECTs are performed. Therefore, there is no need
to perform start tests dedicated to GPO model identification. However, in order to
to perform start tests dedicated to GPO model identification. However, in order to
calibrate the misfire/poor-start GPO predictor and transition rules, several purposely
calibrate the misfire/poor-start GPO predictor and transition rules, several purposely
designed engine start tests, with misfire and poor-start, are required. This can be
designed engine start tests, with misfire and poor-start, are required. This can be
done by further retarding spark advance or disabling one or two cylinder spark signals.
done by further retarding spark advance or disabling one or two cylinder spark signals.

5.2 Calibration of Fuel Dynamics Compensator

An example of data requirement for calibrating the fuel dynamics compensator is
An example of data requirement for calibrating the fuel dynamics compensator is
given in Table 5.1. At least 27 good cold start tests are required. Taking the several
given in Table 5.1. At least 27 good cold start tests are required. Taking the several
special start tests for air loop model calibration into account, 40 start tests would be
special start tests for air loop model calibration into account, 40 start tests would be
sufficient to meet the data requirement for calibrating the entire control.
sufficient to meet the data requirement for calibrating the entire control.

<table>
<thead>
<tr>
<th>ECT</th>
<th>-25°C</th>
<th>-20°C</th>
<th>-15°C</th>
<th>-10°C</th>
<th>-5°C</th>
<th>0°C</th>
<th>10°C</th>
<th>25°C</th>
<th>45°C</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO. of tests</td>
<td>≥ 3</td>
<td>≥ 3</td>
<td>≥ 3</td>
<td>≥ 3</td>
<td>≥ 3</td>
<td>≥ 3</td>
<td>≥ 3</td>
<td>≥ 3</td>
<td>≥ 3</td>
</tr>
</tbody>
</table>

Table 5.1: Data requirement for fuel dynamics compensator calibration

Table 5.1: Data requirement for fuel dynamics compensator calibration

Calibration of the fuel dynamics compensator consists of three parts: calibration
Calibration of the fuel dynamics compensator consists of three parts: calibration
of the $UF_{20}$ function, the UFF model, and the nominal fuel dynamics model. As
of the $UF_{20}$ function, the UFF model, and the nominal fuel dynamics model. As
discussed previously, the last two parts must be identified simultaneously via a joint
discussed previously, the last two parts must be identified simultaneously via a joint
optimization. However, an important concept using mass conservation for the fuel
dynamics identification is discussed first.

5.2.1 Law of Mass Conservation

Mass conservation used in this research refers to the unity gained, asymptotically
stable characteristics of a dynamic process. If the initial condition (memory/storage)
of an asymptotically stable, unity gained dynamical system is identically zero\(^{24}\), then
the energy stored is simply the difference between the sum of the input energy and
the sum of the output energy, given a certain amount of time. Due to exhaust
EQR measurement problem at C1 C2 and data richness issue, deeper physical insight
and constraints are helpful to reduce sensitivity in the identification process so as
to improve robustness of parameter estimates. The resulting consequence is that
the state variable representation of one-pole, one-zero model is deployed to reveal
additional physical constraints:

\[
\begin{align*}
    m_{\text{dep}}(k) &= (1 - \tau)m_{\text{dep}}(k - 1) + (1 - X)u(k) \\
    m_{\text{cyl}}(k) &= \tau m_{\text{dep}}(k - 1) + Xu(k),
\end{align*}
\]

where \(m_{\text{dep}}(k)\) refers to deposited fuel mass\(^{25}\) (or fuel puddle mass) and \(u\) refers to
CINJ. The above state equation can be readily written as an ARMA model by

\[
y(k) - (1 - \tau)y(k - 1) = Xu(k) - (X - \tau)u(k - 1). \quad (5.3)
\]

The relationship among parameters used in both models is shown in Table 5.2.

Because of the mass conservation property of the above models, one important fact

\(^{24}\)Including zero past inputs and outputs.

\(^{25}\)This state variable cannot be directly measured in an engine system, but can be inferred from
other engine variables.
Table 5.2: Relationship among parameters

<table>
<thead>
<tr>
<th>ARMA model</th>
<th>τ-X model</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta_1$</td>
<td>$-(1 - \tau)$</td>
</tr>
<tr>
<td>$\alpha_0$</td>
<td>$X$</td>
</tr>
<tr>
<td>$\alpha_1$</td>
<td>$-(X - \tau)$</td>
</tr>
</tbody>
</table>

Clearly, this is the same statement as the one made at the beginning of this subsection, but shown in a different form. The term $m_{\text{dep}}(T)$ is simply the total deposited fuel mass (energy storage) of the whole system, observed at event $k$. In addition, when the input $u$ and output $m_{\text{cyl}}$ are steadily approaching each other within a sufficiently small neighborhood such that

$$m_{\text{cyl}}(16 \leq k \leq 20) \approx u(16 \leq k \leq 20), \quad (5.5)$$

the following is true:

1. $m_{\text{dep}}(k) \geq 0$.
2. $m_{\text{cyl}}(16 \leq k \leq 20) = \tau m_{\text{dep}}(16 \leq k \leq 20) + Xu(16 \leq k \leq 20)$.
3. By defining a new variable such as $R = \frac{1 - X}{\tau}$ (or $X = 1 - R\tau$), the following expression results

$$R = \frac{m_{\text{dep}}(20)}{m_{\text{cyl}}(20)} \approx \frac{1}{5} \sum_{k=16}^{20} m_{\text{cyl}}(k) \approx \frac{\sum_{k=1}^{20} u(k) - \sum_{k=1}^{20} m_{\text{cyl}}(k)}{\frac{1}{5} \sum_{k=16}^{20} m_{\text{cyl}}(k)}. \quad (5.6)$$
Note that $R$ is a measurement if $	ext{CINJ}$ (denoted as $u$ in the above equation) is known.

Using the relationship $X = 1 - R\tau$, one parameter is eliminated by replacing $X$ in the following equation:

$$y(k) - (1 - \tau)y(k - 1) = Xu(k) - (X - \tau)u(k - 1),$$

resulting in

$$u(k) - u(k - 1) - y(k) + y(k - 1) = \tau(y(k - 1) - u(k - 1) + R(u(k) - u(k - 1))).$$

Since the above expression only has one unknown parameter, the least squares algorithm can robustly identify the parameter $\tau$, given sparse data.

This special usage of mass conservation is used to robustly identify a model using an inherent relationship among model parameters given sparse and noisy data. More importantly, the benefit of this special usage is less sensitive to measurement noise. Forcing mass conservation for solution to this problem is a significant contribution to reducing parameter variation in the identification (calibration) process with sparse and noisy data.

### 5.2.2 Calibration of $\text{UFF}_{20}(ECT)$

The calibration procedure of the $\text{UFF}_{20}(ECT)$ curve would seem to be simple: simply carry out a cubic polynomial curve-fit given several $\text{UFF}_{20}$ measurements at each representative ECT. One such example is to collect data at the following representative value of ECTs: -25°C, -20°C, -15°C, -10°C, -5°C, 0°C, 10°C, 25°C and 45°C. However, there is one special treatment when using the result of the regressed polynomial: when the regressed $\text{UFF}_{20}(ECT)$ curve reaches unity as ECT is large enough, the output of the regressed function is saturated at “1”. An example, which is similar to the one used in Chapter 2, is demonstrated in Figure 5.5.
Figure 5.5: The regressed $UFF_{20}(ECT)$ function with saturation limit

5.2.3 Calibration of UFF and Nominal Fuel Dynamics Model

The effect of diminishing return, fuel delivered versus power generated from that fuel, occurs wherein the parameter $\gamma(ECT)$ varies as a function of ECT. This effect becomes increasingly pronounced for colder temperatures, until ECT drops below approximately $-20^\circ C$, at which point $\gamma(ECT)$ becomes constant. The only difference between the correction effects of the UFF function, for instance, at temperatures below $-20^\circ C$, results from the contribution of $UFF_{20}(ECT)$. In addition, when the $UFF_{20}(ECT)$ function approaches “1”, the diminishing return effect becomes negligible, and therefore the parameter $\gamma(ECT)$ does not vary for temperatures beyond that value of ECT. This nonlinear behavior of the UFF function is summarized in Figure 5.6.
Figure 5.6: Trend of $UFF_{20}(ECT)$ and $\gamma$ vs. ECT

The multi-step calibration process for $\gamma(ECT)$ and the nominal fuel dynamics model are obtained through a joint optimization routine implemented with the following iteration steps:

**Step 1:** Begin optimization from a reasonable initial value for $\gamma(ECT)$, at a given ECT. Examples of reasonable values for initial $\gamma(ECT)$ are shown in Table 5.3.

<table>
<thead>
<tr>
<th>$ECT$</th>
<th>-25°C</th>
<th>-20°C</th>
<th>-15°C</th>
<th>-10°C</th>
<th>-5°C</th>
<th>0°C</th>
<th>10°C</th>
<th>25°C</th>
<th>45°C</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\gamma(ECT)$</td>
<td>500</td>
<td>450</td>
<td>400</td>
<td>350</td>
<td>300</td>
<td>250</td>
<td>200</td>
<td>200</td>
<td>200</td>
</tr>
</tbody>
</table>

Table 5.3: Starting values of $\gamma(ECT)$ for optimization
**Step 2:** Apply

\[
CINJ = UFF_{20}(ECT) \left(1 - \frac{2}{\pi} \arctan \left( \frac{RINJ}{\gamma(ECT)} \right) \right) RINJ
\]

to calculate CINJ. Note that \( UFF_{20}(ECT) \) is obtained from each individual test rather than from the regressed \( UFF_{20}(ECT) \) function.

**Step 3:** Use the accumulative mass conservation to calculate the fuel deposit by the formula

\[
m_{dep}(T) = \sum_{k=1}^{T} u(k) + \sum_{k=1}^{T} m_{cyl}(k),
\]

where \( T \) is set to 20. Also, \( m_{cyl}(1) \) and \( m_{cyl}(2) \) take the MBFM measurements at the lean limit (0.7), given measured GPO\(^{26}\).

**Step 4:** Use the formula

\[
R \approx \frac{\sum_{k=1}^{20} u(k) - \sum_{k=1}^{20} m_{cyl}(k)}{\frac{1}{5} \sum_{k=16}^{20} m_{cyl}(k)}
\]

(5.7)

to calculate \( R \) for each start.

**Step 5:** Obtain an averaged ratio “\( R_{avg} \)” by the following formula:

\[
R_{avg} = \frac{1}{n} \sum_{i=1}^{n} R
\]

(5.8)

where \( n \geq 3 \) is the number of good start tests at a given ECT.

**Step 6:** In the ARMA model representation

\[
u(k) - u(k-1) - y(k) + y(k-1) = \tau (y(k-1) - u(k - 1) + R(u(k) - u(k-1))),
\]

replace parameter \( R \) with \( R_{avg} \).

\(^{26}\)Recall that \( m_{cyl}(1) \) corresponds to \( MBFM(1) \); \( MBFM(1) \) and \( MBFM(2) \) are not accurate due to poor exhaust EQR measurement.
Step 7: Employ a basic least squares algorithm to obtain the parameter $\tau$ based on the reduced ARMA model

$$u(k) - u(k-1) - y(k) + y(k-1) = \tau (y(k-1) - u(k-1) + R_{avg}(u(k) - u(k-1)))$$

Step 8: Calculate the parameter $X$ using the identified parameter $\tau$ in the formula

$$X = 1 - R_{avg}$$

and calculate parameters $\beta_1$, $\alpha_0$, and $\alpha_1$.

Step 9: Simulate the nominal fuel dynamics model in the forward direction with computed CINJ and zero initial condition on $y(k)$.

Step 10: Obtain the simulated MBFM from Cycle 1 through Cycle 20.

Step 11: Calculate the optimization cost criteria, given by the Mean Squared Error (MSE) between simulated MBFM and MBFM, starting from Cycle 3 through Cycle 20.

Step 12: Optimization terminates according to a pre-defined threshold on MSE.

Step 13: The procedure defined in Steps 1-12 is repeated for data corresponding to all values of ECT.

Based on experimental studies, the UFF correction requirement for RINJ at Cycle 1 for each cylinder is different from Cycle 2 and onward. The procedure is then to specify a free parameter, $UFF(1)$, at Cycle 1 in the UFF function and perform an optimization to identify the parameter. Note that $UFF(1)$ is only applied for RINJ correction at Cycle 1; accordingly, the parameter $UFF(1)$ will only be used in the fuel dynamics control at Cycle 1 as well.
The following two equations summarize the above adjustment in the UFF function formulation (with a slight abuse of notation):

\[ CINJ(k = 1) = UFF(k = 1)RINJ(k = 1) \]
\[ CINJ(k > 1) = UFF(ECT) \left( 1 - \frac{2}{\pi} \arctan \left( \frac{RINJ(k > 1)}{\gamma(ECT)} \right) \right) RINJ(k > 1). \]

For control implementation, the choice of which result to use (between these two off-line optimization schemes) is made based on the worst case engine start scenario. For example, for I-4 engines, the scheme with independent \( UFF(1) \) is preferred; for V-8 engines, because of larger inertia, the first scheme is preferred due to reduced RPM fluctuations in poor engine starts.

To summarize the 13-step procedure, a flowchart is shown in Figure 5.7. The result of the above procedure is a family of nominal fuel dynamics models. In this
research, a linear interpolation method is used to obtain a nominal fuel dynamics controller, scheduled according to values of ECT.

5.2.4 Calibration of Inverse UFF Function

Calibration of the inverse UFF function is comprised of two primary components: 1) calibration of the saturation limit; and 2) implementation of numerical inversion. These two portions are obtained by the following procedure:

1. Allow $RINJ(k)$ to increase such that $CINJ/UFF_{20}$ is close to the saturation limit at each given $\gamma(ECT)$, according to

   $$CINJ/UFF_{20} = \left(1 - \frac{2}{\pi} \arctan \left( \frac{RINJ(k)}{\gamma(ECT)} \right) \right) RINJ(k).$$

   An example of a $RINJ(k)$ value sufficient to reach the saturation limit is $RINJ(k) = (4)\gamma(ECT)$, in which case, in the steady state

   $$CINJ/UFF_{20} \approx 4 \left(1 - \frac{2}{\pi} \arctan (4)\right) \gamma(ECT) \approx 0.64\gamma(ECT).$$

2. Find a value of $RINJ(k)$ corresponding to 90% of $CINJ/UFF_{20}$ in the steady state. For convenience, the corresponding values of $RINJ(k)$ and $CINJ/UFF_{20}$ are denoted here as $CINJ/UFF_{20}^{90\%}$ and $RINJ^{90\%}$, respectively.

3. Create data pairs such that when $CINJ/UFF_{20} \geq CINJ/UFF_{20}^{90\%}$, $RINJ(k)$ is “clipped” at the value of $RINJ^{90\%}$.

4. Use the data pair thus obtained to construct the linear splines approximation function for different values of ECT.

Again, the inverse UFF function with 90% saturation limit is shown in Figure 5.8 and, $\gamma(ECT)$, $CINJ/UFF_{20}$ are two inputs of the inverse UFF function, respectively.
5.3 Performance of Control

Before we dive into the simulation results, the simulation setup is given first. For simplicity, the simulation setup is partitioned into four groups.

- The first group covers the thresholds used in the detectors, given in Table 5.4.

<table>
<thead>
<tr>
<th>Threshold</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Misfire</td>
<td>400 RPM</td>
</tr>
<tr>
<td>Poor-start</td>
<td>700 RPM</td>
</tr>
<tr>
<td>Poor-start recovery</td>
<td>700 RPM</td>
</tr>
<tr>
<td>Run mode</td>
<td>1300 RPM</td>
</tr>
</tbody>
</table>

Table 5.4: Detector thresholds
The second group covers the predictor decay and estimator gain setup, shown in Table 5.5.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha_m$</td>
<td>1</td>
</tr>
<tr>
<td>$\alpha_p$</td>
<td>0.9</td>
</tr>
</tbody>
</table>

Table 5.5: Estimator gain and decay rate

The third group covers the initial condition for the fuel dynamics compensator. That is, the initial condition of the past fuel commands (commanded CINJs) take the averaged value of three previous fuel commands.

The forth group covers the commanded desired exhaust EQR. In this comprehensive simulation study, the commanded desired exhaust EQR is set to “1” identically, throughout all ECTs.

Control under Normal Start Condition (Emission Ambient) Because emission tests are usually conducted at the emission ambient condition ($25^\circ C$), accuracy of GPO predictions are extremely important. A “good start” data set collected at $25^\circ C$ is used to test the performance of the GPO predictor and results are shown in Figure 5.9 (a). As shown, three step ahead GPO predictions are accurate. The fuel dynamics control simulation is shown in Figure 5.9 (b). The following observations and corresponding explanations are made:
Three step ahead GPO predictions vs. true GPO measurement at 25°C

(a) GPO prediction results

CRINJ & RINJ with EQR=1 at 25°C

(b) Fuel control results

Figure 5.9: Normal cold start simulation at emission ambient condition
1. Three GPO predictions perform accurately in this starting condition. Transitions through three modes, crank, crank-to-run and run, are sufficiently smooth. The GPO model indeed captures the dominant air loop dynamics.

2. Fueling levels in the first four fueling events are greater than the approach from which the testing data was taken (a production calibration). It is evident that this control results in rich crank fuel, rendering good startability.

3. Under a normal start condition, this control achieves a much smoother fueling trajectory than the production approach.

4. Near steady state, the commanded fuel is leaner than the production approach.

5. A lean start can be obtained by commanding the desired EQR to be less than “1”.

Control under Normal Cold Start (-20°C) To further comprehend the performance of this control under normal start conditions, but at lower temperatures, another start test is used to verify the control. The results of GPO prediction and fuel dynamics control against actual testing data are shown in Figure 5.10. Similar to normal cold start at emission ambient condition, the GPO prediction is quite accurate. Although there are some errors during transition from crank-to-run mode to run mode (around event 14), the prediction error can be tolerated. The basic understanding about this error during transition is because the crank-to-run GPO predictor does not sufficiently capture the correct decay rate. Namely, the simplified linear spline GPO decay rate in the crank-to-run
Three step ahead GPO predictions vs. true GPO measurement at −20°C

(a) GPO prediction results

(b) Fuel control results

Figure 5.10: Normal cold start simulation at -20°C
GPO model is inadequate, thereby passing the incorrect initial condition to run GPO predictor. This is the key reason for the relatively large prediction error during transition. This issue can be effectively solved by refining the partitions for MAP and RPM, or by deploying a full linear spline model to capture the decay rate in each partition. However, improvement on this issue is left for future development. It is worth noting that the testing data is obtained by purposely enriching the crank fuel beyond the production calibration. The good thing is that this control achieves persistently leaner commanded fuel than the production approach.

**Control under Misfire/Poor-start** In order to generate misfire/poor-start, a cold start with substantially lean crank fuel is performed. This not only results in misfire at event 4, but also three consecutive poor-start events. To reveal the detailed control action, three consecutive engine events are studied (refer to Figure 5.11):

**Event 3** The 3\textsuperscript{rd} step ahead GPO prediction is made for event 6 (diamond curve) by assuming the engine will fire as expected.

**Event 4** Because this is no sensible RPM rise at event 4, misfire is detected. The 3\textsuperscript{rd} step ahead GPO prediction for event 7 is the same as the 2\textsuperscript{nd} and 1\textsuperscript{st} step ahead GPO predictions (similar to a holding action). The corresponding fuel control realizes quick enrichment for the next fueling cycle.

**Event 5** Poor-start is detected at this event. All three step GPO predictions are just a simple decay from their estimated GPOs.
Figure 5.11: Cold start simulation with misfire and poor-start
In addition to the above observation, one important fact is also depicted in Figure 5.11 (b): fast lean-off action is undertaken in the second cycle fueling event, starting from event 11. This is a beneficial fuel dynamics compensation action; such an emission reduction feature is not attained in typical production algorithms.

**Control under Poor-start Alone** Although the poor-start simulation was described in the immediate previous subsection with a detailed explanation of control actions in Figure 5.11, it is also interesting to observe the GPO prediction and fuel control under poor-start condition alone.

As shown in Figure 5.12, misfire does not occur at event 4; however, in the next two consecutive engine events (events 5 and 6), the engine speed is not brought beyond the poor-start threshold (700 RPM). Consequently, the poor-start GPO predictor does three control actions starting from events 5 and 6:

1. The 3rd step ahead GPO predictions for the future events 8 and 9 (curve with diamond).
2. The 2nd step ahead GPO predictions for the future events 7 and 8 (curve with star).
3. The 1st step ahead GPO predictions for the future events 6 and 7 (curve with dot).

In those events, the decay rate $\alpha_p$ in the poor-start mode is set to 0.9. This decay rate can be clearly seen in Figure 5.12 (a). At event 7, the poor-start
recovery is detected (engine speed is greater than 700 RPM), the normal crank-
to-run GPO predictor kicks in to take over all three step GPO predictions. As
a result, the predicted GPOs promptly track the true GPO measurement.

Control under the Worst Start Condition  To ensure robustness of the predic-
tive fuel dynamics control in any engine start condition, the resulting control
action, in the face of the worst case scenario, must be scrutinized to make sure
that the no errant control action is undertaken. For this purpose, a worst cold
start test, shown in Figure 5.13, is used to evaluate the control. Three important
observations are worth discussing:

- Rather than responding to the true GPO fluctuation, the predicted GPOs
  are well behaved by means of the GPO filter and poor-start GPO predictor.
- The second cycle fueling events are enriched. More importantly, at event
  6, a make-up fuel pulse results for event 9, as a consequence of the 3rd step
  ahead GPO predictor.
- Fast lean-off is achieved in the 3rd, 4th and 5th fueling cycles. Such in-
dividual cylinder control, with sequential fuel correction, is generally not
  achieved in typical production algorithms.
Figure 5.12: Cold start simulation with poor-start only
Figure 5.13: Cold start simulation under the worst condition (partial engine firing)
5.4 Summary

In this chapter, a comprehensive description of overall calibration process of the predictive fuel dynamics control for I-4 PFI engines is made. By supplying detailed calibration instructions, engine calibrators would have no difficulty following the calibration procedure. By doing so, only 40-50 engine starts are required to complete the calibration task given any engine platform.

The resulting control is simulated against real engine data. The predictive fuel dynamics control not only achieves desired responses based on experience, but also embeds intelligence in the form of misfire/poor-start amelioration. In addition to high efficiency of the control calibration process, a characteristic distinctly embodied by this control, that is, “data in, calibration/control out,” is a concept which is not familiar in most academic circles. While the concept of model-based control is familiar to control engineers, and has been a standard for many years, the idea of using such tools to produce efficient calibration is novel to researchers outside the industrial applications realm.
CHAPTER 6

CONCLUSION

6.1 Summary of Dissertation

At the beginning of this dissertation research, the engine start problem during the first few seconds of engine operation is revealed and discussed with respect to identification and control aspects. Solution to the crank-to-run transition problem, while reducing calibration effort become the major goals. The scope of the dissertation research was therefore clearly defined: with only limited information available in the open literature, use model-based control ideas to solve this practical control problem.

Based on the nature of problem tackled in this research, gain scheduling became an important focal point, and the linear spline modeling technique was brought into this research, as an extension to ideas brought to us by Dr. Kenneth P. Dudek of General Motors Powertrain Advanced Engineering. Benefiting from this powerful modeling idea, a new methodology referred to here as Linear Parameter Varying Linear Spline system (LPV-LSP) was derived. In the dissertation research, the development of LPV-LSP system, as described in Chapter 2, is believed to be a contribution in
and of itself in field of model based engine control. After an example using LPV-LSP system in gain scheduled control, detailed engine models for the crank-to-run transition problem was introduced.

Prior to introducing the final solution to the problem, an unconventional review of research development in a form of lessons learned practice was conducted at the beginning of Chapter 3. The description of the learning development, though summarized and condensed, leads to the final control architecture which is the final predictive fuel dynamics control for engine start and crank-to-run fuel control. By analyzing problems and failures encountered in the dissertation research, feasible solutions for each subproblem emerged gradually. This led to a clearer picture in the evolution of the final control design. In what follows, the structure and detailed description of each component of the predictive fuel dynamics control were elucidated. By means of the multi-step GPO prediction scheme, the cylinder air rate prediction is iterated at each engine event, facilitating individual cylinder AFR control. Fuel dynamics effects of three different cylinders are compensated simultaneously through three independent inverse fuel dynamics compensators, given three predicted GPOs. Though the LPV-LSP model is only applied to the GPO model throughout air and fuel loop dynamics modeling, its impact on the GPO prediction is profound. By commanding a desired exhaust EQR target, the predictive fuel dynamics control realizes an approximate unity gain response. Moreover, the misfire/poor-start detection and corresponding GPO predictor respond to abnormalities in engine start, and proper rescuing actions are undertaken by enriching the immediate next firing event, therefore rendering robustness in the face of anomalous engine starts.
In Chapter 4, identification methodology for calibration is discussed. This chapter severed as a segue between the fuel dynamics control, with air prediction (Chapter 3), and the implementation aspects of calibration (Chapter 5). In addition to emphasizing the fact that system identification methods (indeed, approximation) are critical for model-based control, and subsequent reduction in calibration effort, this chapter has also introduced some relevant philosophy for calibration and approximation, and along the way introducing an interesting criteria when scrutinizing the least squares algorithm and its variants. Because several promising features are embodied by subspace methods, more effort needs to be put into this worthwhile research topic. Although this new criteria does not answer the constrained least squares problem, a review of various identification methods has built a bridge to the next stage, calibration of predictive fuel control. By remembering the key weaknesses and limitations of each identification method, correct action can be taken and errant results can be detected efficiently.

A comprehensive description of overall calibration process of the predictive fuel dynamics control for I-4 PFI engines is given in Chapter 5. By supplying detailed calibration instructions, engine calibrators would have no difficulty following the calibration procedure. In this process, only 40-50 engine starts are required to complete the calibration task given any engine platform, a very efficient process. The resulting control is simulated against real engine data. The predictive fuel dynamics control not only achieves desired responses based on experience, but also embeds intelligence in the form of misfire/poor-start amelioration. In addition to the high efficiency of the control calibration process, a characteristic new to the academic community is distinctly embodied by this novel control, that is, “data in, calibration/control out.”
6.2 Contributions

The major contributions of this work are simply stated: (i) model-based fuel control for engine start and crank-to-run transition, with misfire/poor-start detection and amelioration; and, (ii) a highly efficient calibration process. This dissertation research also makes significant contributions in the fields of model based engine control, gain scheduled modeling, identification and control design, and system identification. In summary, the major contributions of this dissertation research can be categorized as follows:

A Novel Modeling Technique: First of all, the linear spline modeling technique combines a simple but powerful approximation idea into system modeling approaches. Because of its inherent linear structure, the linear spline model is, in particular, suitable for gain scheduled modeling, identification and control design. Secondly, an extension of the (quasi) linear parameter varying model structure using the key idea of linear splines (LPV-LSP) yields an efficient description of a wide class of nonlinear systems, paving the way to further exploration in the theory of gain scheduling.

A State-of-the-Art Engine Start Control. Generally speaking, this dissertation research delivers a start-of-the-art engine start control with an efficient calibration process. In addition, the following advanced, unique features arose:

- Individual cylinder control: By means of individual cylinder air rate prediction iterated at each engine event, with four different GPO models, individual cylinder fuel dynamics control is possible via three correction actions.
• Misfire/poor-start amelioration. Through the misfire/poor-start detector and corresponding GPO predictor, the predictive fuel dynamics control incorporates engine firing behaviors intelligently and realizes appropriate rescue/correction actions to save an engine start in the face of misfire and poor-start, via over-predicted GPOs. This adds robustness to the model based predictive control.

• Flexible calibrator choices. In the control implementation stage, the predictive fuel dynamics control developed herein supplies three flexible choices to engine calibrators:
  
  1. The predictive fuel control offers a near unity gained response. It is straight-forward to achieve the exhaust EQR as lean as possible so as to reduce the engine-out HC emission with correspondingly lean EQR commands.

  2. The commanded EQR can be a single number or a trajectory, implying freedom in choosing the desired exhaust EQR target. If meeting HC emissions requirements is a problem, several candidates for manual tuning during calibration process are possible.

  3. Unlike many production approaches, the robust engine start can be precisely controlled via first cycle fueling events, without enriching the later fueling cycles. In addition to the individual cylinder fuel dynamics compensation, cast in a predictive control framework, plug fouling can be avoided during enrichment.
**Angle between Subspaces:** Primarily serving as a system model order detection tool, the concept of angle between input/output subspaces explains the discrepancy encountered in the forward and inverse model identification. Although the exploration of this concept is by no means thorough, the linkage between and to various concepts in system identification is established, which has not been clearly addressed in the open literature.

It is fair to say that all targeted goals have been met in this dissertation research and some parts have exceeded expectation. However, as in any successful research project, many unanswered questions remain, and several research directions have emerged. It is our hope that the results herein will, therefore, spawn more relevant work in their research areas.
APPENDIX A

NOMENCLATURE

ARMA : Auto regressive moving average
ARX  : Auto regressive with extra input
AFR  : Air fuel ratio
ATDC : After top dead center
BDC  : Bottom dead center
BTDC : Before top dead center
C1, C2 : Cycle #1 and Cycle #2
CA   : Crank angle
CAM  : Camshaft
CINJ : Corrected injected fuel mass
CWT  : Cylinder wall temperature
ECT  : Engine coolant temperature
ECU  : Engine control unit
EOI  : End of injection
EQR  : Equivalent ratio
ESA  : Engine start algorithm
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPC</td>
<td>In-cylinder fresh air charge measured at throttle</td>
</tr>
<tr>
<td>GPO</td>
<td>In-cylinder fresh air charge</td>
</tr>
<tr>
<td>GPOF</td>
<td>Filtered in-cylinder fresh air charge</td>
</tr>
<tr>
<td>GPOM</td>
<td>Measured in-cylinder fresh air charge</td>
</tr>
<tr>
<td>HC</td>
<td>Hydrocarbon</td>
</tr>
<tr>
<td>I-4</td>
<td>Inline four cylinders (engine)</td>
</tr>
<tr>
<td>IC</td>
<td>Internal combustion</td>
</tr>
<tr>
<td>IAT</td>
<td>Intake air temperature</td>
</tr>
<tr>
<td>IVC</td>
<td>Intake valve close</td>
</tr>
<tr>
<td>IVT</td>
<td>Intake valve temperature</td>
</tr>
<tr>
<td>IMEP</td>
<td>Indicated mean effective pressure</td>
</tr>
<tr>
<td>KCPA</td>
<td>Know CAM position algorithm</td>
</tr>
<tr>
<td>LHS</td>
<td>Left-hand side</td>
</tr>
<tr>
<td>LS</td>
<td>Least squares</td>
</tr>
<tr>
<td>LPV</td>
<td>Linear parameter varying</td>
</tr>
<tr>
<td>LPV-LSP</td>
<td>Linear parameter varying linear splines</td>
</tr>
<tr>
<td>LQR</td>
<td>Linear quadratic regulator</td>
</tr>
<tr>
<td>LTI</td>
<td>Linear time invariant</td>
</tr>
<tr>
<td>MAP</td>
<td>Intake manifold pressure</td>
</tr>
<tr>
<td>MBFM</td>
<td>Measured burned fuel mass</td>
</tr>
<tr>
<td>OBD</td>
<td>On-board diagnostics</td>
</tr>
<tr>
<td>PFI</td>
<td>Port fuel injected</td>
</tr>
<tr>
<td>PSA</td>
<td>Post start algorithm</td>
</tr>
<tr>
<td>RINJ</td>
<td>Raw injected fuel mass</td>
</tr>
<tr>
<td>RPM</td>
<td>Crankshaft speed</td>
</tr>
<tr>
<td>RHS</td>
<td>Right-hand side</td>
</tr>
<tr>
<td>Acronym</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>SI</td>
<td>Spark ignited</td>
</tr>
<tr>
<td>SULEV</td>
<td>Super low emission vehicle</td>
</tr>
<tr>
<td>TDC</td>
<td>Top dead center</td>
</tr>
<tr>
<td>TLS</td>
<td>Total least squares</td>
</tr>
<tr>
<td>TPS</td>
<td>Throttle position sensor signal</td>
</tr>
<tr>
<td>TWC</td>
<td>Three-way catalytic converter</td>
</tr>
<tr>
<td>UFF</td>
<td>Utilized fuel fraction</td>
</tr>
<tr>
<td>UFF20</td>
<td>Utilized fuel fraction measured at cycle 20</td>
</tr>
<tr>
<td>ULEV</td>
<td>Ultra low emission vehicle</td>
</tr>
<tr>
<td>VE</td>
<td>Volumetric efficiency</td>
</tr>
<tr>
<td>VVA</td>
<td>Variable valve actuation</td>
</tr>
<tr>
<td>VVT</td>
<td>Variable valve timing</td>
</tr>
<tr>
<td>WAFR</td>
<td>Wide range air-fuel ratio</td>
</tr>
<tr>
<td>ZLEV</td>
<td>Zero level emission vehicle</td>
</tr>
</tbody>
</table>
BIBLIOGRAPHY


[34] G. Rizzoni. “ME781 class notes”. The Ohio State University, Mechanical Engineering Department, Sep. 2004.


