UNIVERSITY OF
LIVERPOOL

Optimal dynamic calibration methods for powertrain controllers

Author
Kamil Ostrowski

June 22, 2015

Supervisors
Dr A.T. Shenton
Prof. H. Ouyang

Abstract

Emission legislation for passenger cars has become more stringent and the increasing demand for reduced fuel consumption has resulted in the introduction of complex new engine and after-treatment technologies involving significantly more control parameters. Vehicle manufacturers employ a time consuming engine parameter calibration process to optimise vehicle performance through the development of engine management system control maps. The traditional static calibration methods require an exponential increase in calibration time with additional calibration parameters and control objectives. To address this issue, this thesis develops and investigates a novel Inverse Optimal Behaviour Based Dynamic Calibration methodology and its application to diesel engines.

This multi-stage methodology is based on dynamic black-box modelling and dynamic system optimisation. Firstly the engine behaviour is characterized by black-box models, based on data obtained in a rapid data collection process, for accurate dynamic representation of a subject engine. Then constrained dynamic optimisation is employed to find the optimal input-output behaviour. Finally the optimal input-output behaviour is used to identify feedforward dynamic controllers. The current study applies the methodology to an industrial state-of-the-art WAVERT model of a 1.5 litre Turbo EU6.1 Diesel engine acting as a virtual engine. The approach directly yields a feedforward controller in a nonlinear polynomial structure which can either be directly implemented in the engine-management system or converted to a dynamic or static look-up table format. The results indicate that the methodology is superior to the conventional static calibration approach in both computing efficiency and control performance.

A low-cost Transient Testing Platform is presented in this work to carry out transient data collection experiments on a steady-state dynamometer with application to non-linear engine and emissions modelling using State Space Neural Networks. This modelling technique is shown to be superior to the polynomial

models and achieves similar performance to non-linear autoregressive with exogenous input neural (NARMAX) network models. Numerical Dynamic Programming is investigated in a simplified engine calibration problem for a virtual engine to potentially improve the dynamic calibration optimisation stage.

In a second study the novel dynamic calibration methodology is applied to the airpath control of a 3.0L Jaguar Land Rover (JLR) turbocharged Diesel engine utilizing a direct optimisation approach and State Space Neural Network models. A complete experimental application of the methodology is demonstrated in a vehicle where the vehicle-implemented calibration is obtained in a one-shot process solely from data obtained from the fast dynamic dynamometer testing. The results obtained demonstrate the potential of this methodology for the rapid development of efficient dynamic feedforward controllers based on limited data from the engine test bed.

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

AC	Alternating Current
AFR	Air-Fuel Ratio
APRBS	Amplitude Modulated Pseudo-Random Binary Sequence
ARMA	Autoregressive Moving Average
ARMAX	Autoregressive Moving Average with Exogenous Input
ARX	Autoregressive with Exogenous Input
AT	Averaging Time
BTDC	Before Top Dead Center
CI	Compression Ignition
CO	Carbon Oxide
CO_2	Carbon Dioxide
COM	Component Object Model
CT	Control Time
DEM	Discrete Event Model
DI	Direct Injection
DOC	Diesel Oxidation Catalyst
DoE	Design of Experiments
ECU	Electronic Control Unit
EGR	Exhaust Gas Recirculation
EMS	Engine Management Software
EPA	Environmental Protection Agency
EU	European Union
EUDC	Extra Urban Driving Cycle
FTP	Federal Test Procedure
GDI	Gasoline Direct Injection
HC	Hydrocarbon
HCCI	Homogeneous Charge Compression Ignition
IDI	Indirect Injection
IMC	Internal Model Control
IMEP	Indicated Mean Effective Pressure
INJ	Fuel Injection Mass

IV	Instrumental Variable
JLR	Jaguar Land Rover
LI	Laser Ignition
LOLIMOT	Local Linear Model Tree
MFB50	Mass Fraction Burned of 50%
MIMO	Multiple Input Multiple Output
MISO	Multiple Input Single Output
MBC	Model Based Calibration
MBT	Minimal spark advance for Best Torque
MLS	Maximum Likelihood
MPC	Model Predictive Control
MVM	Mean Value Model
NARX	Non-linear Autoregressive with Exogenous Input
NEDC	New European Driving Cycle
NLP	Non-linear Programming
NN	Neural Networks
NN-NARX	Neural Network Non-linear Autoregressive with Exogenous Input
NOx	Nitrogen Oxides
NRMSE	Normalized Root Mean Square Error
ODE	Ordinary Differential Equation
OE	Output Error
OLS	Ordinary Least Square
PEMS	Portable Emission Measurement System
PE	Prediction Error
PI	Proportional-Integral
PID	Proportional-Integral-Derivative
PM	Particulate matter
PN	Particulate number
PRBS	Pseudo-Random Binary Sequence
RDE	Real Driving Emission
RPM	Revolution per minute
SI	Spark Ignition
SCR	Selective Catalytic Reduction
SNOPT	Sparse Non-linear Optimizer

SOI	Start of Injection
SSNN	State Space Neural Network
ST	Stabilisation Time
SQP	Sequential Quadratic Programming
TC	Turbocharger
TTC	Transient Testing Platform
UDC	Urban Driving Cycle
UEGO	Universal Exhaust Gas Oxygen
US	United States
VGT	Variable Geometry Turbocharger
WLTP	World harmonized Light vehicle Test Procedure

Notation

x_{op}	Operation point
X_P	Calibration parameter
X_{op}	Operating range
Φ	Cost function for calibration
$y(t)$	System output
$u(t)$	System input
n_k	Discrete number of delays between input and output
q^{-1}	Time shift operator
$G(q^{-1})$	Transfer function of the system
$H(q^{-1})$	Transfer function of the noise affecting the system
$\epsilon(t)$	System noise
$A(q^{-1})$	Polynomial denominator of the system transfer function
na	Order of the polynomial $A(q^{-1})$
a_j	Parameter of polynomial $A(q^{-1})$
$B(q^{-1})$	Polynomial numerator of the system transfer function
nb	Order of the polynomial $B(q^{-1})$
b_j	Parameter of polynomial $B(q^{-1})$
$C(q^{-1})$	Polynomial numerator of the noise transfer function
nc	Order of the polynomial $C(q^{-1})$
c_j	Parameter of polynomial $C(q^{-1})$
$D(q^{-1})$	Polynomial denominator of the noise transfer function
nd	Order of the polynomial $D(q^{-1})$
d_j	Parameter of polynomial $D(q^{-1})$
$F(q^{-1})$	Polynomial denominator of the system transfer function
nf	Order of the polynomial $F(q^{-1})$
f_j	Parameter of polynomial $F(q^{-1})$
e	Error between measured and predicted output
$\widehat{y}(t)$	Predicted output
Y	Measurement vector of output

X	Matrix of regressors
θ	Vector of system parameters
N	Number of samples
R^2	Correlation coefficient
L_1	Manhattan distance norm
T_d	Demanded torque
T_{op}	Optimal torque
T_o	Original torque
T_b	Brake Torque
G_1	Transfer function of engine torque and engine emissions
G_2	Transfer function of road load
F	Feedforward controller
$\overline{e}$	Ratio between mean brake torque and mean INJ
$x(t)$	System state
x_0	Initial state
t_f	Final time
t_0	Starting time
$J(u(t))$	Cost function
$G(x(t_f))$	Final cost function
$H(x(t)), u(t), t)$	Intermediate cost function
$\lambda(t)$	Lagrange multiplier
u_k	Discrete input
x_k	Discrete state
u_{opt}	Optimal input trajectory
U	Vector of control parameters for optimisation
Y_u	Vector of control and state parameters for optimisation
T	Target set of states
U_k	Set of admissible inputs
X_k	Set of admissible states
V	Cost-to-go function
π	Optimal control policy
$\mu(k)$	Optimal control input
$g_k(x(k), u(k))$	Discrete cost function
w_i	Weights in optimisation problem
P	State weighting constant

Q	Control weighting matrix
R, S, L, M	Weighting matrices
ϕ	Non-linear smooth function
C_i	Scaling synoptic weights matrix
W_i	Internal synoptic weights matrix
$c(x(i), u(i))$	Constraint function

Chapter 1

Introduction

1.1 Thesis statement

The work reported in this thesis was set out to investigate a novel technique for engine calibration called concept of Inverse Optimal Behaviour based Dynamic Calibration by application this methodology to diesel engine calibration for reduced fuel consumption and to meet emission constraints. In the light of coming challenges for engine control and calibration in regard to new engine technologies that increase the number of control actuators and required testing and validating over more dynamic driving cycles there is a strong need for a calibration methodology that requires short testing time on the test bed, but without sacrificing either engine performance or emissions. The dynamic calibration methodology integrates the control processes with the calibration activity potentially resulting in saving testing time on the test bed due to the limited test data required also with the potential to improve fuel economy, satisfy emissions constraints and achieve smooth torque tracking.
The study sought to answer the following questions:

1. Is the Dynamic Calibration a viable methodology for producing a controller for smooth engine torque tracking with limited testing time while not exceeding the emission limits?

2. Does dynamic calibration have any fuel consumption benefit over conventional static maps?

3. What is the most efficient way of modelling the dynamic engine and emission behaviour in terms of a reduced number of parameters?

4. What is the best optimisation approach for optimising of the engine dynamics and emissions behaviour?

5. Is it possible to apply this methodology only to the Air-path control calibration problem?

6. Is this technology ready to implement into production level engine control units?

1.2 Motivation

The development of a new internal combustion engine [5] for a vehicle is driven nowadays by multiple conflicting factors especially for passengers cars. From the customer perspective an engine should be characterized by low fuel consumption and good engine performance. Increasing consumer environmental awareness adds low emission levels to this list, but this factor is driven mainly by emission regulation, as every new car can not exceed very stringent emission norms. In future one can expect that emission legislation will be more and more stringent. To meet such requirements manufacturers equip new power units with new technologies that keeps engine performance at a satisfactory level, but at the same time meets the emission regulations. This is the main reason why new cars have recently been fitted with an extended package of after-treatment systems. From one side it is beneficial as new technologies lead to engines which together with the peripheral systems meet all the demanding legislated requirements. On the other hand the engine becomes very complicated from a control system point of view which make it hard to calibrate because of the increasing number of parameters that must be calibrated. Additionally new driving cycles are to be introduced in the near future which are more transient, randomised and cover much bigger of operating envelope.

Table 1.1: Emission levels for passengers' cars with gasoline engines

Standard	**Date**	**CO** [g/km]	**HC** [g/km]	**NOx** [g/km]	**NOx+HC** [g/km]	**PM** [g/km]	**PN** -
EURO I	07/92	2.72	-	-	0.97	-	-
EURO II	01/96	2.20	-	-	0.5	-	-
EURO III	01/00	2.30	0.20	0.15	-	-	-
EURO IV	01/05	1	0.1	0.08	-	-	-
EURO V	09/09	1	0.1	0.06	-	0.005	-
EURO VI	09/14	1	0.1	0.06	-	0.005	6×10^{11}

1.2.1 Emissions

According to existing regulations every new car that leaves the factory must fulfil the emission standards and those standards differ according to the country it is sold to.
In Europe there is a European Union emission standard called the EURO standard which was introduced for the first time in 1992 for passengers' cars. This caused the development of new control strategies [6] and from 2001 diagnostic systems (1996 in USA) together with new after-treatment systems. The standards were introduced for both gasoline and diesel engines and because at that time the differences between the two engine types were large, the standards were significantly different for those two engine technologies. As currently both engine technologies are converging because of the use of high pressure fuel pumps and direct injection systems, the emission norms are converging accordingly. The current existing standard in Europe for passengers' cars is EURO VI which become valid from September 2014.

The EURO standard presented in Table 1.2 and 1.1 shows the trend of reducing the emission limits every couple of years. It can be observed that recent standards additionally include particulate matter (PM) and particulate number (PN) together with nitric oxide and nitrogen dioxide (NOx), carbon oxide (CO) and hydro carbon (HC) emissions.

Table 1.2: Emission levels for passengers' cars with diesel engines. For EURO II until 1999.09.30 there were different standards for Direct Injection (DI) and Indirect Injection (IDI) diesel engines. After that date DI engines must meet the IDI limits)

Standard	**Date**	**CO** [g/km]	**HC** [g/km]	**NOx** [g/km]	**NOx+HC** [g/km]	**PM** [g/km]	**PN** -
EURO I	07/92	2.72	-	-	0.97	0.14	-
EURO II IDI	01/96	1	-	-	0.7	0.08	-
EURO II DI	01/96	1	-	-	0.9	0.10	-
EURO III	01/00	0.64	-	0.50	0.56	0.05	-
EURO IV	01/05	0.5	-	0.25	0.30	0.025	-
EURO Va	09/09	0.5	-	0.18	0.23	0.005	-
EURO Vb	09/09	0.5	-	0.18	0.23	0.005	6×10^{11}
EURO VI	08/14	0.5	-	0.08	0.23	0.005	6×10^{11}

1.2.2 Driving cycles

In addition to limits becoming harder to be met, a new trend of developing new driving cycles can be seen. The reason for this is the regulation bodies want to make the legislative driving cycles as close as possible to typical driving profiles. The current driving cycles in Europe are based on very conservative driving behaviour which in practice is not very realistic. On engine speed-load grid it covers only the low-load low-speed region, when actually more typical driving is based across the whole operation envelope, including middle and high-load high-speed as well. Automotive manufacturers have been using this fact for their benefit to meet the legislated test targets by taking care of emissions in low-load low-speed conditions and focusing less on emissions in the rest of the grid, which not visited during the legislated test by not using EGR outside the low-load low-speed region to save cost.

New driving cycles are now to be introduced to cut this custom by forcing manufacturers to take care of emissions in the whole operating envelope. This will lead to increasing calibration time and calibration costs however.

New European Driving Cycle

In Europe the currently existing driving cycle is the New European Driving Cycle (NEDC). It consists of two different sections: Urban Driving Cycles (UDC) as in Figure 1.1 and Extra-Urban Driving Cycles (EUDC) as in Figure 1.2. The NEDC is modal which means that it involves protracted periods at constant speeds.

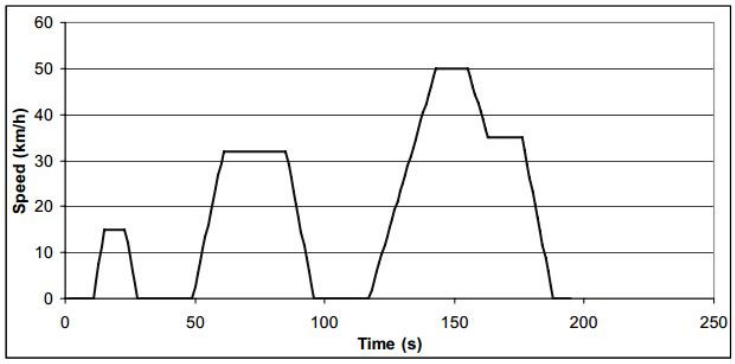

Figure 1.1: Urban Driving Cycle[1]

This driving cycle was derived theoretically, so that the acceleration trajectories is represented as a straight monotonic line and there are periods of steady state driving. This driving cycle is in contrast to US driving cycles which were derived from real driving measurements.

World harmonized light vehicle test procedure (WLTP)

The Worldwide harmonized Light vehicles test procedure is currently under development by specialists from EU, India and Japan under guidelines from the UNECE World Forum for Harmonization of Vehicle Regulations [7]. However no country has yet confirmed a final date for introducing this test cycle. The test is closer to real live driving profiles then NEDC.

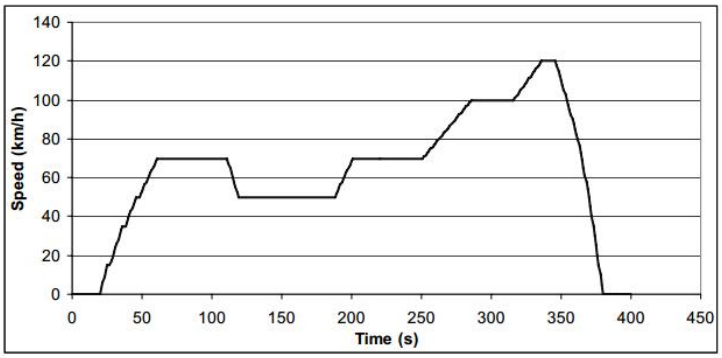

Figure 1.2: EUDC driving cycle[1]

FTP-75 and US06 in the United States of America

The city driving cycles known as Federal Test Procedure 75 is more close to real life driving then the current European cycle. Despite this fact a new driving cycle was needed because according to the US Environment Protection Agency (EPA) the engines run cleanly during the mandatory driving cycles but in normal use they generate more pollutants. As a result in 2007 the EPA has introduced a supplemental Federal Test Procedure - US06 to address the shortcomings with the FTP-75 by adding more aggressive driver behaviour with aggressive high speed and rapid speed fluctuations.

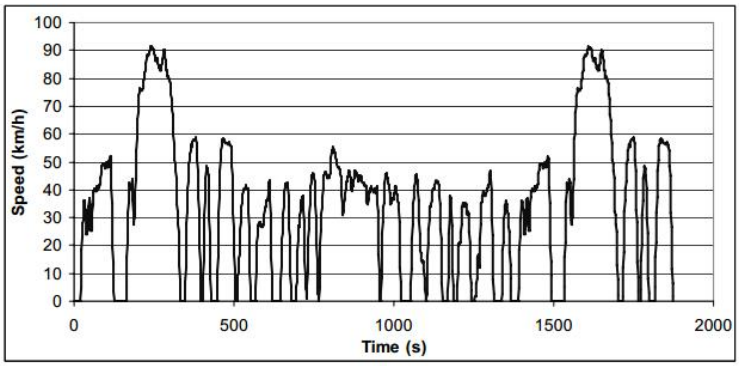

Figure 1.3: FTP75 driving cycle [1]

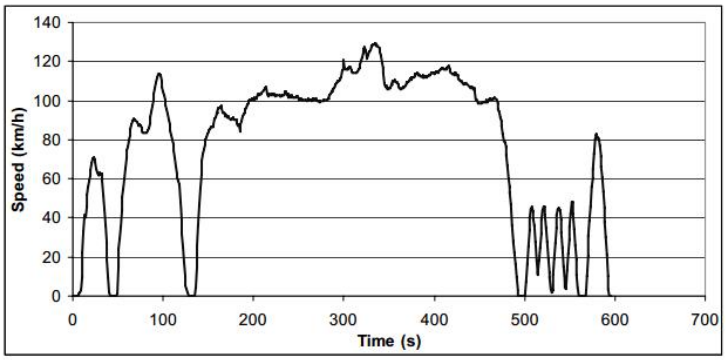

Figure 1.4: US06 driving cycle [1]

1.2.3 Real world driving

The newest European regulation that will be introduced at some point in the near future although this is still under discussion for light-weight vehicles is the Real Driving Emission (RDE) regulation. The concept behind it is quite simple: the cars are going to be commissioned on the road in real world condition instead of in a laboratory under predefined climatic conditions and specific driving cycles. This requires new emission equipment to be fitted in vehicle measuring the emission actually coming out of exhaust pipe in real road conditions. This equipment is called the Portable Emission Measurement System (PEMS) and is required to be capable of measuring all relevant engine emissions.

1.3 New engine technologies

The internal combustion engine is under constant development resulting in new technologies being introduced such as turbocharging and Exhaust Gas Recirculation to meet tightening emission limits and to improve engine performance.

1.3.1 Turbocharging

Recently turbocharging has become present in almost every new diesel engine and new charging technology such as sequential turbocharging, parallel turbocharging, twin-scroll turbochargers and electric charging have become more popular [8].

A turbocharger is a mechanical device combining a turbine and a compressor connected together by a shaft. The turbocharger enlarges the power of an engine by forcing more air going into the combustion chamber and thereby allowing more fuel to be injected to the combustion chamber resulting in increased engine performance. The energy for compressing the intake air by the compressor comes from the exhaust gas which accelerates the turbine. A turbocharger influences the dynamic behaviour of combustion engines significantly because of the inertia of the moving parts and time required to speed up the turbocharger.

1.3.2 Exhaust Gas Recirculation

Exhaust Gas Recirculation (EGR) technology was introduced to reduce NOx emissions in gasoline and diesel engines [9]. EGR works by redirecting an exhaust gas from the exhaust manifold to the intake manifold and then back to the combustion chamber. This technique is called external EGR in contrast to internal EGR where the exhaust gases are captured in the combustion chamber by modulating the exhaust valve timing.
This way concentration of nitrogen in the air-charge is reduced and also the exhaust gases absorb combustion heat resulting in a reduction of peak temperatures which reduces engine-out NOx emissions.

1.4 Engine controls

1.4.1 Look-up-tables

As a result of new technologies engine control systems have became very complex [10]. Surprisingly very little development of the control systems has been made by manufacturers and still the most common control structure for internal combustion engines is the static look-up-table structure which is very much more common than other control structures [11]. Look-up-tables are usually two dimensional based on an engine speed and engine torque grid. The look-up-table is a very simple concept which is of course always a desirable feature but in mod-

ern engines the number of look-up-tables needed has became quite significant. This makes calibration a very time consuming process. The look-up-tables are calibrated until the overall engine performance, emission and fuel economy meet predefined targets. Some additional controls are implemented to keep the engine in a safe condition but also to meet emission constraints. As an example a limiter for smoke can typically be implemented as a look-up table. The look-up-tables perform well in steady state conditions but in transient operation can lead to poor performance.

1.5 Objectives

The main objectives of the work in this thesis is to develop and evaluate a dynamic calibration methodology to produce comparable performance of the resulting dynamic controller compared to that from the look-up-tables broadly used in engine management system, whilst saving testing time on the test bed due to the limited test data required and with the potential to improve fuel economy, whilst satisfying emissions constraints and achieving smooth torque tracking.

1.6 Contributions

The contributions of this thesis are listed below:

- A novel approach to model-based dynamic engine calibration is proposed to obtain optimal settings of fuel consumption subject to constraints on torque and emissions and is applied experimentally on a diesel turbocharged direct injection 1.5L virtual engine. Global dynamic Neural Networks models are developed and utilized for model-based optimization. Dynamic polynomial models obtained by an inverse identification of the optimal input-output behaviour are used as feedforward dynamic calibration controllers which produce effective and smooth control signals.

- A novel automatic Transient Testing Platform is presented for testing on an engine test bed which originally was designed for only steady state mea-

surements. The structure of the platform and its implementation details are described.

- A novel application of State Space Neural Networks are applied to the experimental modelling of engine emissions and the engine performance of a 5.0L naturally aspirated JLR engine. An heuristic algorithm for structure selection is developed to overcome over-fitting and the local minima problem.

- Numeric Dynamic Programming was investigated as a possible alternative to existing optimisation algorithms in the dynamic calibration methodology.

- An experimental study is presented of the novel engine dynamic control calibration methodology applied to the air-path of a 3.0L Jaguar Land Rover turbocharged diesel engine. Implementation is made of a direct optimisation approach into dynamic calibration based on State Space Neural Network models. A complete application of the methodology is demonstrated in a vehicle where the vehicle-implemented calibration is obtained in a one-shot process solely from data obtained from novel dynamic dynamometer testing, without recourse to in-vehicle tuning.

1.7 Outline of the thesis

This thesis is structured as follows: Chap. 2 describes literature review related to engine control, engine calibration and modelling. In Chap. 3 a preliminary study is introduced that investigates experimental application of the dynamic calibration to a virtual engine. The calibration problem includes fuel consumption and emissions constraints. An application of Numerical Dynamic Programming for simplified calibration task is presented in Chap. 4. The Transient Testing platform is described in detail in Chap. 5 with a novel application of State Space Neural Network for the engine and emissions modelling. An experimental study is presented in Chap. 7 of the novel engine dynamic control calibration methodology applied to the air-path of a 3.0L Jaguar Land Rover turbocharged diesel engine

and in-vehicle validation of the resulting feedforward dynamic controllers. Chap. 8 is dedicated to the conclusions.

Chapter 2

Literature Survey

2.1 Introduction

The subject of engine calibration has been widely researched in recent years. This has became an important subject because modern internal combustion engines require accurate and multidimensional calibration in order to meet increasingly stricter emission regulation and CO_2 limits. Increasing requirements for engine performance, fuel economy and emissions lead to more and more subsystems and control functions being implemented in modern internal combustion engines. This has lead to a rapid increase in the number of control parameters and consequently a dramatic increase in the time and cost of engine calibration. Figure 2.1 shows an overview of typical calibration tasks that has to be carried out for the petrol engine. The only difference for diesel engine is that there is no lambda control.
The fundamental task in the process is the base engine calibration and emission calibration, where optimal engine settings to run the engine smoothly, minimising the fuel consumption and meeting the emission requirements are determined.
As current engine strategies are largely torque-based, there is special calibration needed for the internal torque model. The rest of the static calibrations provide safe and clean engine operation.
The vehicle and transmission calibrations take place after all engine related calibrations are finished. Those calibrations cover various comfort and dynamic functions for the transient states of the vehicle and the transmission.

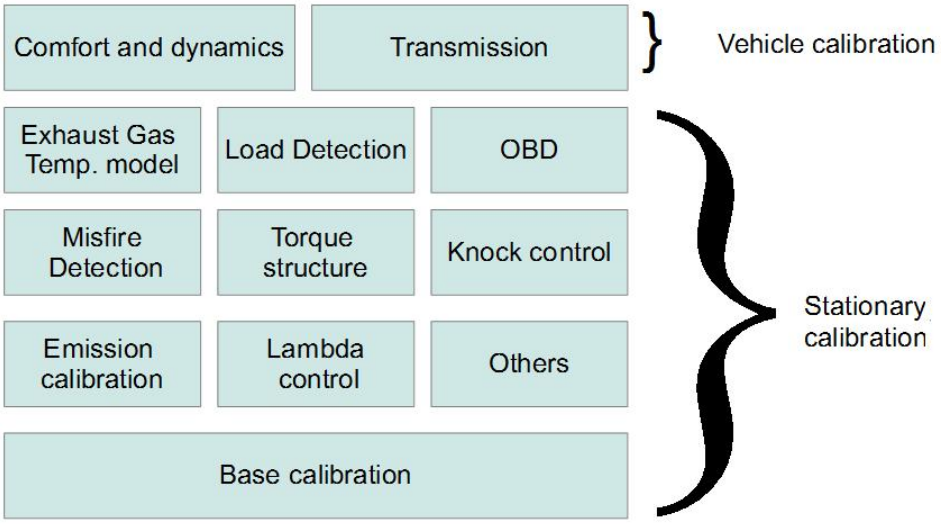

Figure 2.1: Numbers of different calibration tasks [2]

General practice within the automotive industry is to perform the engine calibration for steady state behaviour. This may be thought surprising as an engine operates very rarely on steady state conditions with constant speed and constant torque. Having said that, in an electric control unit (ECU) many functions are still realized in the form of static functions represented as 2D maps and the on-line optimisation of transients is not possible. Nevertheless, using dynamic measurements would still save a significant amount of time spent on a test bed compared to using steady state measurements. A second potential benefit of dynamic modelling is that the transient behaviour of the engine can be captured which could have applications to engine emissions which also show dynamic behaviour [12] [13] and so a calibration methodology which accounted for this sort of behaviour should lead to more realistic calibrations.

A wide scope of techniques for engine modelling has been investigated in research because the power-train system is a complex system which includes non-linear and linear elements [14].

2.2 Engine calibration

2.2.1 Steady state calibration

A calibration is called *steady state* when data collection on the test bed is carried out in a steady state manner. This means that discrete tests remain on a specified operating point generally of engine speed and torque. The simplified static calibration task is presented in Figure 2.2.

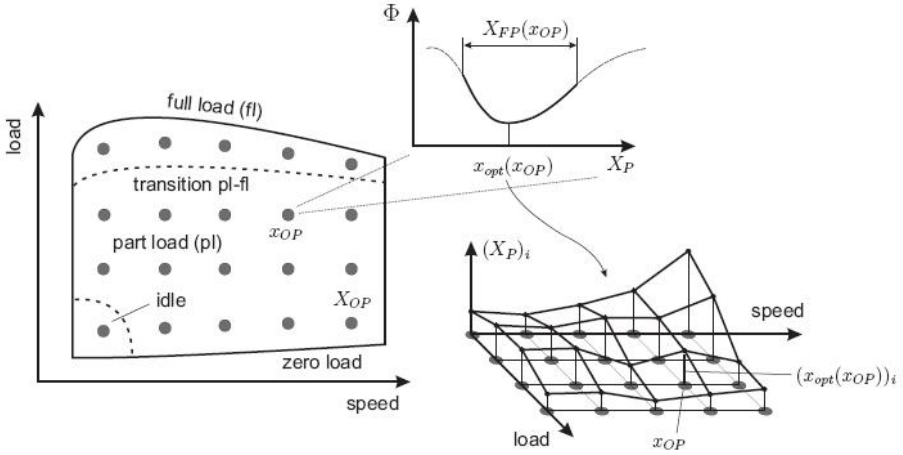

Figure 2.2: Example of simplified calibration task [2]

Here firstly set of operating points is designed to cover the operating range X_{op}. At each operating point x_{op} optimisation is carried out to find the optimal setting for the X_P parameter minimizing certain cost function Φ as follows:

$$\min_{X_P} \Phi(x_{op}, X_P) \tag{2.1}$$

And finally when for each operating point x_{op} minimization is completed, a 2D map is created to store the calibration in the ECU.
To reduce the test time static calibration procedures which choose a reduced numbers of test points based on Design of Experiments (DoE) optimisation methods have been extensively developed and exploited in recent years.
In one such method [15] a statistical machine learning tool was presented for diesel engine air path calibration. The methodology consists of a quasi-random

Sobol space-filling technique for test design, in which a radial basis function network was employed for engine modelling and numerical optimisation used to find optimized calibration maps.
In an alternative method an efficient automated steady-state DoE approach for GDI engines was developed for highly non-linear systems with irregularly shaped operating regions. This was achieved by combining new DoE techniques with automatic and adaptive identification of the region of interest in the high dimensional parameter space [16].
An improved static calibration process was presented in [17] which combines DoE optimisation with a fuzzy rule-based calibration. The calibration rules were described in accordance with operating parameters and physical engine phenomena, and developed a fuzzy logic enabling automatic calibration using a search based on these rules.

2.2.2 Transient calibration

Transient calibration is currently a manual process depending on the experience of the calibration engineer who is conducting the calibration process. The process is performed to meet regulation rather than optimise the system. Novel methodologies for transient calibration have been investigated over the last few years to integrate control and calibration processes and to obtain good control in transient conditions. In [18] neural network models were utilized for model-based control for fuel consumption and emission online optimisation.
In [19] a quasi-static engine model with a static emission model were used for actuator trajectory optimisation. Although this work does not result in a final engine calibration, the methodology shows a potential for transient calibration.
In [20] a dynamic calibration approach was applied for a virtual engine without emissions outputs with promising results.

2.3 Steady state data collection

Historically before electronic control units were introduced, the engine calibration was conducted manually on the test bench led by an experienced calibrator

adjusting mechanical parts to find optimal settings. In recent years, after the introduction of the ECU and more actuators, because of the fact that an internal combustion engine is an under-sensed system, where key controlled variables are not measured (such as torque and emissions) once the engine is placed into vehicle, some form of modelling process is usually part of any calibration methodology.

There are two basic approaches for engine modelling for calibration: static and dynamic. Measurements are generally performed in steady state conditions because many calibration tasks are conducted in steady state conditions. Figure 2.3

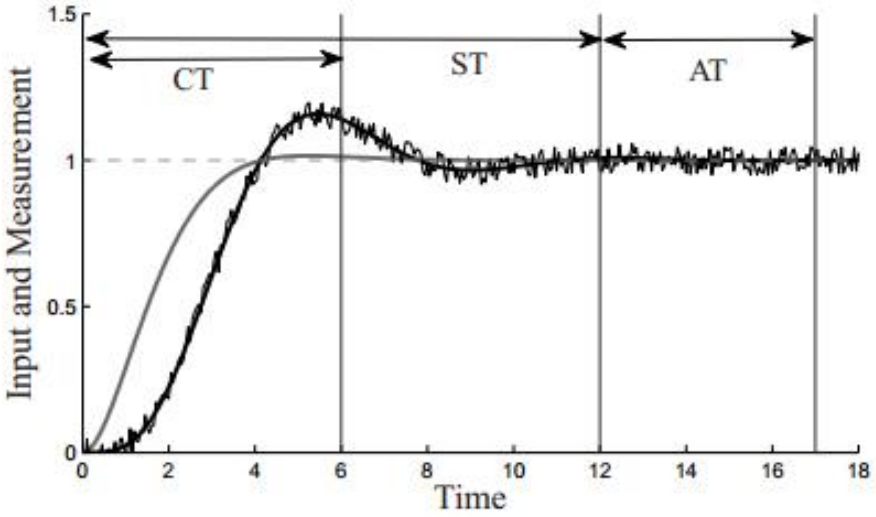

Figure 2.3: Steady state measurements phases: Control (CT), Stabilisation (ST) and Averaging time (AT) [2]

presents the steady state measurement procedure. Firstly the specified parameter is set to the desired value via some form of automatic control in the control time (CT). The adjustment always has an influence on measured variables, especially for variables with long time constants such as temperatures. Therefore there is a period of time called the stabilization time (ST) when all measured values are stabilized. And then finally during averaging time the mean values are captured. This is the reason why the overall measurement time take so long and depends on the time constants of several minutes, typically something from 2 to 5 minutes.

2.4 Engine modelling

For reasons mentioned above any calibration methodology employs some form of the system modelling. Modelling techniques can be divided into three main groups depending on the prior knowledge of the system:

- white-box modelling
- grey-box modelling
- black-box modelling

2.4.1 White-box modelling

The white-box model is sometimes called a physical or phenomenological model. This model is obtained by analysing the system or the process using some basic physical principles or laws such Newton's laws of motion or thermodynamic laws. The dynamics of such system should be understand clearly and physical parameters should be known with a degree of certainty, eg. resistances, diameters and mass. Although there are some advantages to this type of modelling [21], for power-train systems it is difficult to obtain an accurate physical model suitable for control. This is due to the fact that in internal combustion engines there are a large number of kinematic, thermodynamic and fluid dynamic processes that occur simultaneously.
For control systems design applications there two modelling approaches which are most popular in establishing engine models. Although both belong to this same family of modelling the two approaches are significantly different.

The first one is discrete event modelling (DEM). The DEM is based on discrete engine cycle events and is used for models which work on crank-based periods, so it represents the engine in a very detailed manner. The DEM is used for combustion modelling, air flow fluctuations, generating pressure traces across engine strokes etc. This type of model has been implemented in commercial available tools to support engine control development. Such models provide insights into the system which help researchers to decide on the most appropriate approach

to design control laws. Another advantage is that once a satisfactory model has been developed it can be applied to similar systems by means of changing basic parameters of the model. Two of the most popular commercial tools utilizing DEM which are available on the market for engine modelling are Wave by Ricardo [22] and GT-Power by Gamma Technologies Inc [23]. Typically because of the high computational cost of this type of model it is not feasible to run transient simulations in the real-time, however WAVE has on option to trim the model to obtained the real time capability. A detailed DEM for a spark ignited (SI) engine was developed in [24].

The second approach to physic based modelling is mean value modelling (MVM) which obtains not so detailed model as DEM. The typical output of such model is mean torque without consideration of the discrete characteristics of torque generation. This type of model has a limited capability for representing engine behaviour but on the other hand has a significantly reduced computational demand. In [25] [26] complete MVM models for SI engines have been presented. In [27] [28] MVM models have been presented with particular application to powertrain.

2.4.2 Grey-box modelling

Grey box modelling is an approach combing white-box modelling and black-box modelling. The white box approach is applied for those processes or phenomenons for which the physics behind them are better understood and are easier to model using physical laws. On the other hand for parts of the system for which there is no deep knowledge about the system an empirical modelling technique is used instead. In [29] [30] grey box modelling has been presented for homogeneous charge compression ignition engines and for application in control design.
A combination of equation-based and NN data-driven models was used to obtain dynamic models and verified the results from model-based calibration over transient tests and regulatory drive cycles in [31].

2.4.3 Black-box modelling

Black-box models are often referred to as empirical models as in general they are obtained during data-driven system identification experiments. The black-box modelling approach is to build a mathematical model of a subject system from only observed behaviour of the physical system. This type of modelling does not require a deep physical prior knowledge from researchers about the subject system, and the information about the physical process of the system is completely contained in the input-output data. The system identification is developed by collecting appropriate input signals and the observation of the responses often referred to as outputs. The empirical model describes the relationship between the input signals and the output signals of the system. The system identification is then a process of determining parameters of the model of the subject system and has been described in detail in [32] [33] and with special attention to non-linear identification in [34].
The engine modelling community has recently taken much interest in black-box modelling. In [35] LOLIMOT structured models were used to describe the dynamics of engine torque generation. Comparative study on engine torque modelling using different neural networks was presented in [36]. In [37] dynamic models as Volterra series were identified and then a base calibration was carried out with satisfactory results based on those models. The work presented in [38] has investigated input-output modelling, where linear ARX (autoregressive with exogenous input) models were employed to characterise air-fuel ratio dynamics.

2.5 System identification

The system identification process is an iterative process with multiple iterative loops. After the first run of the process the researcher has to make the decision whether the obtained model meets the requirements by verifying the model output against the real model output. If it does not meet requirements, the researcher has to make the decision about what needs to be changed to improve the results. The general procedure of the system identification process is shown in Figure 2.4.

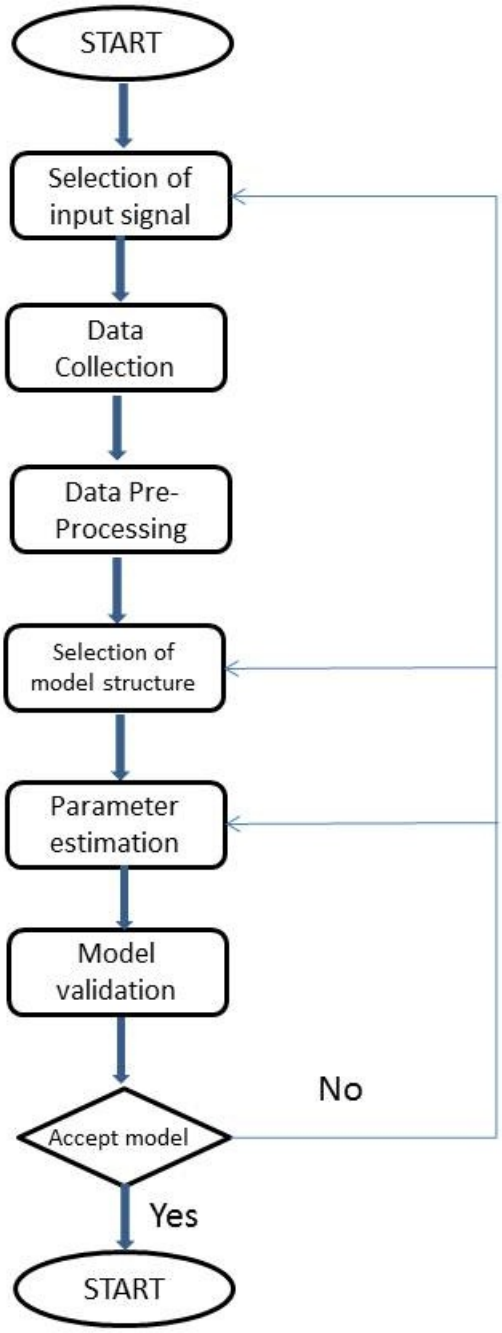

Figure 2.4: A general procedure of system identification

Typically the system identification experiment is carried out in open loop conditions. This gives the benefit of the input signals being not correlated to the output signals which helps with finding the input-output relationship. However in some situations the experiment may have to be carried out in closed loop conditions. Usually this is due to safety considerations to prevent the subjected system going into unsafe conditions. This introduces the issue of input signal being related with output signal though the controller. In [39] [40] the problem is described in detail and a number of possible solutions of closed loop identification are presented. One of them is the indirect method where firstly the transfer function of closed loop system is obtained and later the open loop transfer function of the plant is calculated using the controller equations.

2.5.1 Test signal design

The subject system has to be effectively excited by the test signal to capture the dynamical behaviour of the system in the operating region of interest. The choice of test input signal is critical to the final results of the system identification experiments. The test signal has to be chosen carefully because for black-box modelling pre-knowledge about the system is usually limited. Therefore a rich test signal should have the right frequency spectrum and amplitude to excite the desired frequency range and sensitivity to the magnitude of the system.
Optimal test signals are frequently close to white noise which is not practically possible to generate because it is an unconstrained signal and also could required too fast actuator switching.

For linear system identification, a popular test signal is the pseudo-random binary sequence (PRBS) which is a signal built with only two values: low and high. This type of signal has good properties as it can excite a wide range of frequencies. This signal approximates white noise [41].
For non-linear system identification, the chirp signal is broadly used which is a sinusoidal-like signal with time variant frequency. In [42] the chirp signals were applied to engine emission modelling with a novel scaling safety feature. This signal has the advantage of avoiding rapid changes if the very high frequency

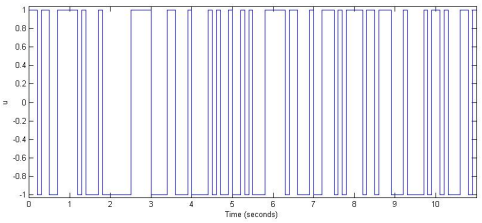

Figure 2.5: Example of PRBS signal

range is outside of interest.

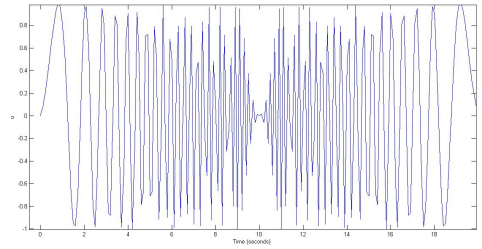

Figure 2.6: Example of chirp signal

Another signal used for non-linear identification is the Amplitude Modulated Pseudo-Random Binary Sequence which is a periodic deterministic signal with properties similar to the white noise [43]. The APRBS is often used in system identification on the test bed because it covers a wide amplitude range to capture the non-linearities [44].

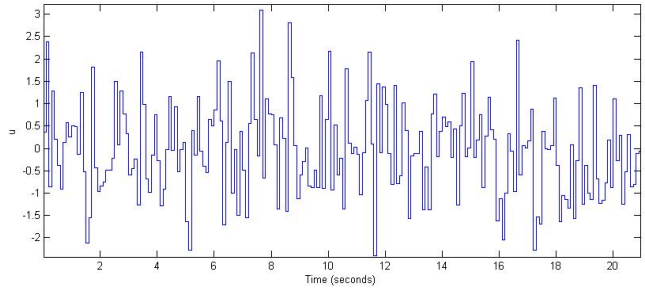

Figure 2.7: Example of APRBS signal

Special care must be taken for multiple input single output (MISO) and multiple input multiple output (MIMO) systems to consider different power density for

different channels. If for example stronger input occurs at this same time as a weak one, the way to identify the weak relationship would be challenging [45]. Some iterative methodologies of designing optimal input design using optimization can be found in [20].

2.5.2 Data Collection

Once the system is efficiently excited then the data collection process takes place. In this process output values containing the system dynamics are captured. Special care must be taken as the sampling frequency is very important. When a slow sampling rate is chosen then high frequency behaviour will be impossible to identify. According to the Nyquist-Shannon sampling theorem [46], a signal can be correctly identified only if the maximum frequency is less then half the sampling rate. Generally the sampling frequency is chosen at around 10 times the system bandwidth. However, a higher sampling rate can be selected without a risk of not capturing the important range of frequencies, as later in the post-processing activities the data can be down sampled.

2.5.3 Selection of model

Depending on the properties of the subject system either a linear or a non-linear model can be applied to describe the behaviour of the system. There are many different types of dynamic black-box models that researchers have been applying over last 50 years. In the past due to lack of computational capability only analytical tools were available for control system synthesis, therefore linear models were more preferable. The limitation of the linear model is that it is not capable of representing non-linear behaviour of the many significantly non-linear real systems such as power-train systems where non-linear models are best used. However the discrete-time linear model has been popular in research in last few decades due to its simple polynomial form and its well established way of estimating parameters.

The most popular linear models are models belonging to the autoregressive models family which means that the output depends on previous inputs and outputs. The family of autoregressive models consist of Auto Regressive Moving Average

(ARMA) models, Auto Regressive eXogenous (ARX) models, Auto Regressive Moving Average with eXogenous input (ARMAX) models, Output Error (OE) models and Box Jenkins models.
A general equation of discrete-time linear polynomial models is as follows:

$$y(t) = G(q^{-1})u(t - n_k) + H(q^{-1})\epsilon(t) \tag{2.2}$$

where q^{-1} is the time shift operator, y(t) is the discrete output, u(t) is the discrete input, $G(q^{-1})$ represents the discrete transfer function of the system, $H(q^{-1})$ represents the discrete transfer function of the noise affecting the system, ϵ (t) represents the system noise (which is commonly assumed to be white noise) and n_k is the discrete number of sample times of the delay between input of the system and output.
Different transfer functions can lead up to 32 different classes of models [32] and so only those commonly regarded as the most important classes are described in this chapter.

ARMA

A basic deterministic model is the Auto-Regressive Moving-Average (ARMA) model that has following structure:

$$y(t) = \frac{B(q^{-1})}{A(q^{-1})}u(t - n_k) \tag{2.3}$$

where

$$A(q^{-1}) = 1 + a_1q^{-1} + a_2q^{-2} + ... + a_{na}q^{-na} \tag{2.4}$$
$$B(q^{-1}) = b_1q^{-1} + b_2q^{-2} + ... + b_{nb}q^{-nb+1} \tag{2.5}$$

where $A(q^{-1})$ is the polynomial denominator of the system transfer function of order na and $B(q^{-1})$ is the numerator of the system transfer function of order nb.
The ARMA model does not contain any noise input to the system. This means that the output value can be determined in an entirely deterministic manner.

Sometimes it is more convenient to represent this system in regressive form:

$$y(t) = x(t)\theta \tag{2.6}$$

where $\theta = [a_1, a_2, ..., a_m, b_1, b_2, ..., b_n]$ is the vector of parameters and $x(t) = [-y(t-1), -y(t-2), ..., -y(t-na), u(t-1), u(t-2), ..., u(t-na-n_k)]$ is the vector of regressors.

ARX

The ARX model is a stochastic model derived from the equation 2.2 when the transfer function of noise is represented as $H(q^{-1}) = \frac{1}{A(q^{-1})}$. The Autoregressive part is represented by $A(q^{-1})y(t)$ and the exogenous input by $B(q^{-1})u(t)$. Then the ARX structure is as follows:

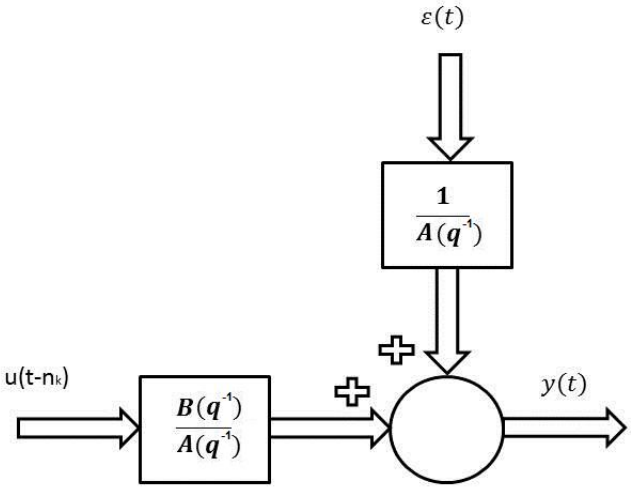

Figure 2.8: Structure of ARX model

$$y(t) = \frac{B(q^{-1})}{A(q^{-1})}u(t-n_k) + \frac{1}{A(q^{-1})}\epsilon(t) \tag{2.7}$$

where

$$A(q^{-1}) = 1 + a_1q^{-1} + a_2q^{-2} + ... + a_{na}q^{-na} \tag{2.8}$$

$$B(q^{-1}) = b_1q^{-1} + b_2q^{-2} + ... + b_{nb}q^{-nb+1} \tag{2.9}$$

ARMAX

The Autoregressive moving-average with exogenous input (ARMAX) model can be seen as an extended version of the ARX model. It includes independent the polynomial $C(q^{-1})$ to describe the more complex noise behaviour. The noise can then represent the plant uncertainty or input disturbances. The ARMAX structure is as follows :

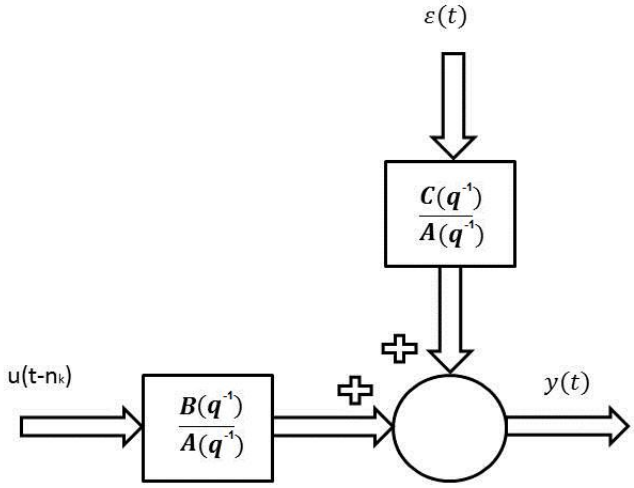

Figure 2.9: Structure of ARMAX model

$$y(t) = \frac{B(q^{-1})}{A(q^{-1})}u(t - n_k) + \frac{C(q^{-1})}{A(q^{-1})}\epsilon(t) \tag{2.10}$$

where

$$A(q^{-1}) = 1 + a_1 q^{-1} + a_2 q^{-2} + ... + a_{na} q^{-na} \tag{2.11}$$

$$B(q^{-1}) = b_1 q^{-1} + b_2 q^{-2} + ... + b_{nb} q^{-nb+1} \tag{2.12}$$

$$C(q^{-1}) = c_1 q^{-1} + c_2 q^{-2} + ... + c_{nc} q^{-nc+1} \tag{2.13}$$

Box-Jenkins

The Box-Jenkins model has the most complicated structure which gives a lot of flexibility for plant and noise modeling. In contrast to the ARX and ARMAX structures where the transfer function of the plant and noise share the same denominator, in the Box-Jenkins structure they are independent. The structure

of the Box-Jenkins model is as follows:

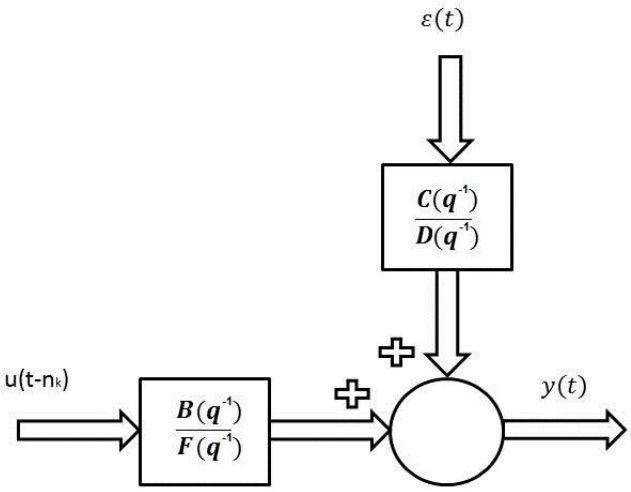

Figure 2.10: Structure of Box-Jenkins model

$$y(t) = \frac{B(q^{-1})}{A(q^{-1})}u(t - n_k) + \frac{C(q^{-1})}{D(q^{-1})}\epsilon(t) \tag{2.14}$$

where

$$B(q^{-1}) = b_1 q^{-1} + b_2 q^{-2} + ... + b_{nb} q^{-nb+1} \tag{2.15}$$

$$C(q^{-1}) = c_1 q^{-1} + c_2 q^{-2} + ... + c_{nc} q^{-nc+1} \tag{2.16}$$

$$D(q^{-1}) = d_1 q^{-1} + d_2 q^{-2} + ... + d_{nd} q^{-nd+1} \tag{2.17}$$

$$F(q^{-1}) = f_1 q^{-1} + f_2 q^{-2} + ... + f_{nf} q^{-nf+1} \tag{2.18}$$

Output Error model

The Output Error model is a simplified version of the Box-Jenkins model, where the transfer function of the noise is equal to 1. This means that the noise is directly added to the output without going through the dynamics of the plant and often refers to a pure error in the measurement of output.

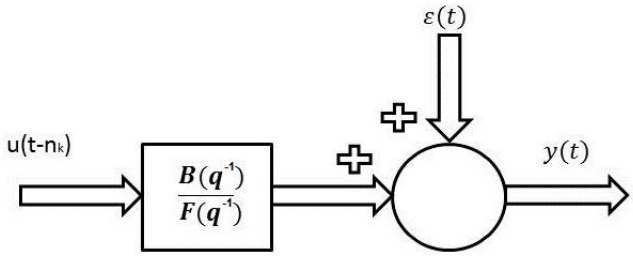

Figure 2.11: Structure of Output Error model

The structure of Output Error is as follows:

$$y(t) = \frac{B(q^{-1})}{A(q^{-1})}u(t - n_k) + \epsilon(t) \tag{2.19}$$

where

$$B(q^{-1}) = b_1 q^{-1} + b_2 q^{-2} + ... + b_{nb} q^{-nb+1} \tag{2.20}$$

$$F(q^{-1}) = f_1 q^{-1} + f_2 q^{-2} + ... + f_{nf} q^{-nf+1} \tag{2.21}$$

2.5.4 Parameter Estimation

Let us consider a typical linear dynamic model with ARX structure as follows:

$$y(t) = \frac{B(q^{-1})}{A(q^{-1})}u(t - n_k) + \frac{1}{A(q^{-1})}\epsilon(t) \tag{2.22}$$

Then let us multiply the equation 2.22 by $A(q^{-1})$ obtaining :

$$A(q^{-1})y(t) = B(q^{-1})u(t - n_k) + \epsilon(t) \tag{2.23}$$

This can be represented by the linear difference equations:

$$y(t) = -a_1 y(t-1) - a_2 y(t-2) - ... - a_{na} y(t - na) \tag{2.24}$$

$$+b_1 u(t-1) + b_2 u(t-2) + ... + b_{na} u(t - nb - n_k) + \epsilon(t) \tag{2.25}$$

This can also be represented in simplified form as follows:

$$y(t) = x(t)\theta + \epsilon(t) \tag{2.26}$$

where $\theta = [a_1, a_2, ..., a_m, b_1, b_2, ..., b_n]$ is the vector of parameters and $x(t) = [-y(t-1), -y(t-2), ..., -y(t-na), u(t-1), u(t-2), ..., u(t-na-n_k)]$ is the vector of regressors.

The equation 2.26 describes the linear system as it was obtained from the linear difference equation 2.24. Both the parameters and regressors are linear. However for systems with non-linear regressors this approach can also be used, as long as the parameters have a linear relationship to the output.
For such system many parametric methods have been developed. The Ordinary Least Square (OLS) method and the maximum likelihood (ML) method are one of the most popular[32] of these.

Ordinary Least Square Method

The objective of this method is to minimize the sum of squared errors between the measured output and the predicted output [47]. The error is then as follows :

$$e(t) = y(t) - \widehat{y}(t) \tag{2.27}$$

where e is the error, $y(t)$ is the measured output and $\widehat{y}(t)$ is the predicted output.

Let us consider the equation 2.26 and assume that N number of samples have been captured. Now this equation can be rewritten in the matrix form:

$$Y = X\theta + \epsilon(t) \tag{2.28}$$

where Y denotes a $N \times 1$ measurement vector of outputs, $Y = [y(1), y(2), ..., y(N)]$, θ as before denotes an $(na + nb) \times 1$ true system parameter vector.

The estimated output can be then described as follows:

$$\widehat{Y} = X\widehat{\theta} \tag{2.29}$$

where $\widehat{Y}$ denotes a $N \times 1$ estimated vector of the output $\widehat{Y} = [\widehat{y}(1), \widehat{y}(2), ..., \widehat{y}(N)]$, and $\widehat{\theta}$ as denotes a $na + nb \times 1$ estimated system parameters vector.

The Ordinary Least Square method is then the method of solving the optimisation problem:

$$\min_{\widehat{\theta}} \frac{1}{2}(Y - \widehat{Y})(Y - \widehat{Y}) = \tag{2.30}$$

$$\min_{\widehat{\theta}} \frac{1}{2}(Y - X\widehat{\theta})(Y - X\widehat{\theta}) = \tag{2.31}$$

$$\min_{\widehat{\theta}} \frac{1}{2}(Y^T Y - \widehat{\theta}^T X^T Y - Y^T X\widehat{\theta} + \widehat{\theta}^T X^T X\widehat{\theta}) \tag{2.32}$$

To find the solution for such a problem the first derivative of the cost function must first be obtained and equated to 0 as follows:

$$\frac{1}{2}\frac{\partial(Y^T Y - \widehat{\theta}^T X^T Y - Y^T X\widehat{\theta} + \widehat{\theta}^T X^T X\widehat{\theta})}{\partial\widehat{\theta}} = -X^T Y + X^T X\widehat{\theta} = 0 \tag{2.33}$$

$$X^T X\widehat{\theta} = X^T Y \tag{2.34}$$

$$\widehat{\theta} = (X^T X)^{-1} X^T Y \tag{2.35}$$

The resulting OLS estimator is an unbiased estimator for the ARX model when the system noise is uncorrelated. An unbiased estimator has the property that $E[\widehat{\theta}] = \theta$. For other types of models where the system noise is correlated, variants of the least square method have been developed. Amongst these, there are the bias correction method [48], the prediction error (PE) method [49] and the instrumental variable (IV) method [33].

So far only offline identification method have been mentioned, but there are

also recursive versions of the least square methods [50]. Recursive least square methods are online identification methods which mean that the parameters are identified whilst the plant is running. Such recursive methods are often combined with some adaptive control, where the control law can adapt to changes in the model parameters. In [51] details of recursive methods can be found with applications to online estimation of vehicle mass and gradient.

2.5.5 Validation

The validation in system identification is a process of checking or measuring the quality of the modelling. Once the parameters of the model have been obtained, the simulated output has to be compared to the measured data. It is important to prepare separate data set for validation, which is different to the data set which is used for identification. This data is often referred to as unseen data.

One way (and usually it is the first way) of assessing the quality of the modelling is a visual inspection where simply the predicted output is plotted against the measured output. The quantitative way of assessing the quality of any modelling can be achieved by the introduction of some statistical assessment of fit obtained by mathematical calculation. A multiple correlation coefficient R^2 is one such of measure and it is defined as follows:

$$R^2(Y,\hat{Y}) = 1 - \frac{\left\|\hat{Y} - Y\right\|^2}{\left\|Y - \overline{Y}\right\|^2} \tag{2.36}$$

where $\overline{Y}$ is the mean of the output Y.
The most common measure used is obtained by normalizing the root mean square error (NRMSE) which is often called 'the best fit' :

$$NRMSE(Y,\hat{Y}) = 1 - \sqrt{\frac{\left\|\hat{Y} - Y\right\|^2}{\left\|Y - \overline{Y}\right\|^2}} \tag{2.37}$$

The best fit values vary between -INF (bad fit) to 100% (ie perfect fit which then means the model can recover 100% of the real system).

2.6 State-of-art engine control

Control-oriented engine models were presented in [19] where quasi-static emissions models were integrated with mean value engine models. Two possible applications of such model were presented: in first case the models is used as a virtual sensor of engine out emissions to estimate Particulate Matter (PM) and Nitrogen Oxides (NOx) including validation results for steady-state and transient conditions. In the second application the model was used for optimal control of various actuators during transient manoeuvres and this showed potential in the reduction of the pollutant emissions from the diesel engine.A multi-variable emissions-feedback controller was designed in [52] for the NOx and PM emissions, where the air/fuel ratio (AFR) was used as an indicator for the PM emissions, and where control inputs were exhaust gas recirculation (EGR) and start of injection (SOI). Firstly the control synthesis was tested on only the AFR control. The multi-variable controller manipulating two inputs thus consisted of two independent Internal Model Control (IMC) control-loops based on linear approximation of the engine.

Many researchers have investigated novel methods for dynamic calibration. A novel local linear radial basic function network (LOLIMOT) applied to engine modelling was presented in [53] including integration with an online optimisation algorithm to find optimal engine inputs. In [54] a mean value model was used to run offline optimisation over dynamic manoeuvres and the trajectories were then used to train neural network controllers. A good introduction to dynamic model-based calibration for diesel engine can be found [55].

New sensing technologies introduce new opportunities for development of new controls system that were not available in past. One of such technology is the close-loop control of ignition timing presented in [56] using the Ionization Current Feedback for gasoline engines. From the ionization current number of combustion parameters can be calculated such as Mass Burn Fraction 50% and Peak Pressure

Position. The feedback algorithm regulates ignition timing to keep engine running at its Minimal spark advance for Best Torque (MBT) timing, avoiding knock limits and reducing the cycle-by-cycle variation. A proportional-integral (PI) controller was used to correct the MBT spark timing.

The close-loop combustion control is another approaching control algorithm for diesel engines [57]. A pressure sensing glow plug was the enabling technology for this control algorithm. A pressure sensing glow plug is a glow plug that integrates with pressure sensor enabling capability of measurig in-cylinder pressure measurement. Utilising the in-cylinder pressures signal from this sensor it is possible to keep engine running in optimal combustion point of MFB50 and correct the engine torque for climatic conditions, engine-to-engine variation and engine wear. In [58] in-cylinder pressure signal was used for the cylinder balancing and multi-pulse fuelling control system. The in-cylinder pressure signal was also investigated to estimate the NOx emissions [59]. In [60] [61] are presented presented algorithms based on Neural Networks modelling for reconstructing the in-cylinder pressure.

New actuating technologies also also initiate development of control strategies. A laser ignition (LI) engine presented in [62] is shown to be an alternative to traditional spark coil-based ignition replacing spark plug with set of optical lenses focusing the beam into one or multiple points to ignition the combustion process. This technology results in additional potential benefits of using such optical window to monitor the combustion event in the real-time.

Following the trend of downsizing of the engine to reduce the fuel consumption, a new trend can be observed in the automotive industry focusing more on the electrification of the vehicle fleet. One form of electrification is an electrically assisted turbo-charger (TC) for a modern diesel engine. In [63] control system based on sliding modes was developed to assess the befits of using the electrically assisted turbo-charger for a diesel engine equipped with Variable Geometry Turbocharging (VGT) and Exhaust Gas Recirculation (EGR) achieving precise regulation of the fresh air fraction in the intake charge during transient conditions on the FTP-75 driving cycle. A typical hybrid powertrain combines an internal

combustion engine and an electric motor which allows for significant reduction in fuel consumption. Dynamic programming is one of popular algorithm to find optimal usage of the electric mode sometimes refereed to as power management to improve the fuel economy [64] [65].

Not only new technologies are developed in the automotive industry and the academia but also new combustion modes. Recently very popular subject in the research is a Homogeneous Charge Compression Ignition (HCCI) engine as efficient and low-pollutant engine technology which results in much colder combustion which then can lead to misfires. The HCCI engine is a combination of a spark ignition (SI) engine and a compression ignition (CI) engine. It utilizes a premixed charge of air and fuel similarly to the SI engine but similarly to CI engine fuel-air mixture is very dilute and ignited by compression The work presented in [66] is focused on the analysis of combustion stability of the HCCI engine under various conditions and on the comparison of the static and dynamic feedforward controllers for rapid and large changes in fuelling level. In practice the in-cylinder pressure sensors mentioned before can be used for the engine calibration for optimal fuel consumption using online method such as Extremum seeking [67]. This method was also applied for tuning the PID and feedforward combustion-timing controllers for MBF50 is a manipulated input.

Nowadays more engines use the number of so-called operating modes such as normal mode, regeneration modes, starting mode and purge mode etc. From the calibration and controls perspective the problem with multiple modes is with switching between those modes that needs to take place without losing the engine performance and driveablility. The work [68] presents a systematic approaches for the control of switched systems with application to a diesel engine where optimal inputs are found for the torque and lambda tracking during switching between two operating modes.

The model-predictive control (MPC) has been widely implemented in the process industry over last decades. In one such application of MPC control system are proposed in [69] for air-fuel ratio control for a SI engine with Kalman filter

used to estimate current engine states. Another approach for the enhancement of the air-fuel control for SI engines can be achieved by active catalyst control strategy based on the use of two linear exhaust gas oxygen (UEGO) sensors both upstream and downstream of the catalyst. It is shown that using two sensor the bias present in upstream sensor measurement can be estimated. It turned out that actually the best location of the second sensor is not after the catalyst but in the middle of it.

The MPC can also be applied to the control of EGR and VGT control but it was shown in [70] that choice of outputs in this problem has a significant impact on overall results. It was stated that MPC controllers has superior control capability compared to PID controllers resulting in reducing EGR-error and lowering pumping losses. Another study focused on the minimization of pumping losses is presented in [71]. In [72] a black box model of an engine air-path was used for predictive control showing improvements in dynamic responses of the mass air flow and the intake manifold pressure but interestingly without any improvements on the emissions.

The introduction of new after-treatment systems has driven development of new control strategies to optimise the tailpipe emissions. One of the most popular in heavy-duty and light-duty vehicles after-treatment technology for reduction of NOx emissions is the Selective catalytic reduction (SCR) technology which uses injections of urea to the exhaust pipe. The urea then converts into ammonia and then reacts in the SCR converter with the engine-out NOx reducing the overall tilepipe emissions.
The associated problem with the SCR converter is the slippage of NH_3 as a result of the overdose of the urea. Comparision of NOx-based control using a NOx sensor and the NH_3 based-control using NH_3 sensor is presented in [73].

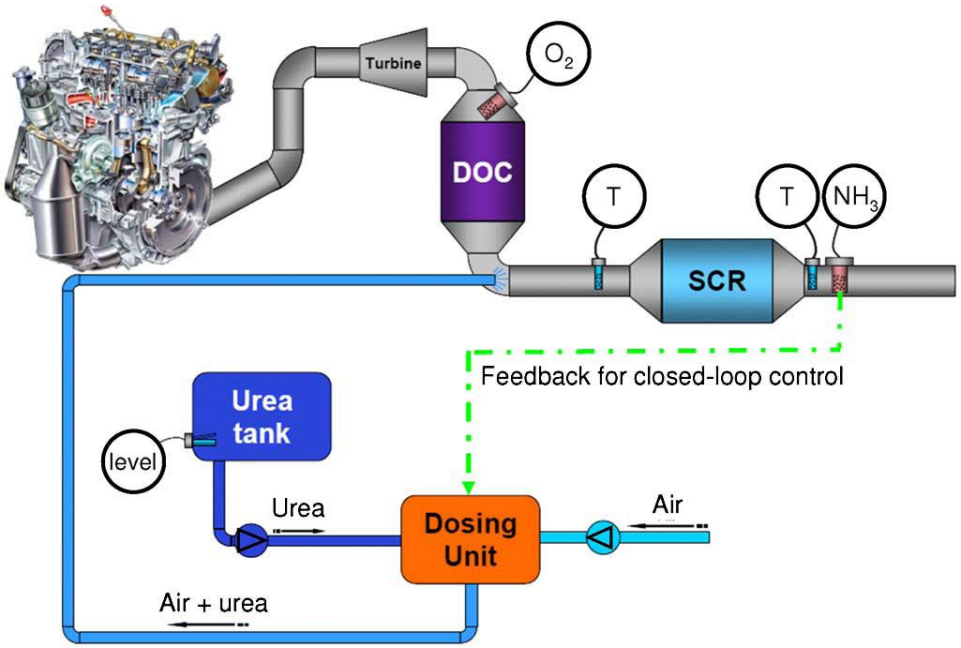

Figure 2.12: Automotive urea SCR system layout [3]

Chapter 3

Inverse Optimal Behaviour based Dynamic Calibration for Fueling in the Diesel Engine

3.1 Introduction

This chapter introduces a novel dynamic calibration methodology for multivariable control and its application to the diesel engine. Although the methodology can be applied to both spark-ignition and compression-ignition engine this work is conducted on a turbocharged compression-ignition unit only. This methodology is based on multi-stage black box modelling and offline optimization of control inputs. The objective of this methodology is to design an optimal dynamic feedforward engine controller which improves the fuel economy whilst delivering smooth torque and maintaining specified emissions limits such as on Nitrogen Oxides (NOx) and Particulate Matter (PM). The proposed dynamic calibration is a process that employs system identification techniques with dynamic Design of Experiments (DoE) and can exploit global constrained numerical optimisation codes. This calibration methodology in contrast to conventional steady-state calibration takes into account transient responses of an engine and emissions.
For purposes of this work an industrial state-of-art Wave RT model of 1.5 litre Turbo EU6.1 turbocharged DI engine integrated with emission models obtained

from data from a real test-bed is employed. Finally the performance of the dynamic controller are compared to current steady-state model-based calibration methods.

3.2 Experimental set-up

3.2.1 Virtual model

The dynamic calibration and control method development and validation is conducted on a WAVE 1.5 litre Turbo EU6.1 engine model which has been developed and experimentally validated by the Ford Motor Company to act as a virtual engine. The determinism and repeatability of such a virtual model is advantageous in assessing the effectiveness of the new control systems and calibration methodologies. The Ricardo Wave software is a state-of-art platform for 1D gas dynamics simulations and is equipped with a Wiebe combustion model. Because of their high accuracy capability such models are broadly used in academia and industry. The Wave-RT extension provides the capability of speeding up the Wave model to real-time speed capability. In this work the provided Wave model was converted into an an RT-WAVE model to gain fast running computation in Matlab/Simulink to act as a virtual engine during the investigation.

3.2.2 Conventional loop-up-tables

For validation of this method against a conventional calibration methodology look-up-tables were provided by Ford that are capable of tracking the desired torque with optimised fuel consumption whilst satisfying the limits on feed gas Nitrogen Oxides (NOx) emissions and Particulate Matter (PM) emissions.
These maps are compatible with the production standard used in automotive industry in EMS software, which have the dependent variables of engine speed and inner torque. It is worth mentioning here that recently the most popular control strategy for modern engines is the so-called *torque based* method where everything in the strategy is related to inner torque. The maps then require then accurate information about actual generated torque inside a cylinder. This

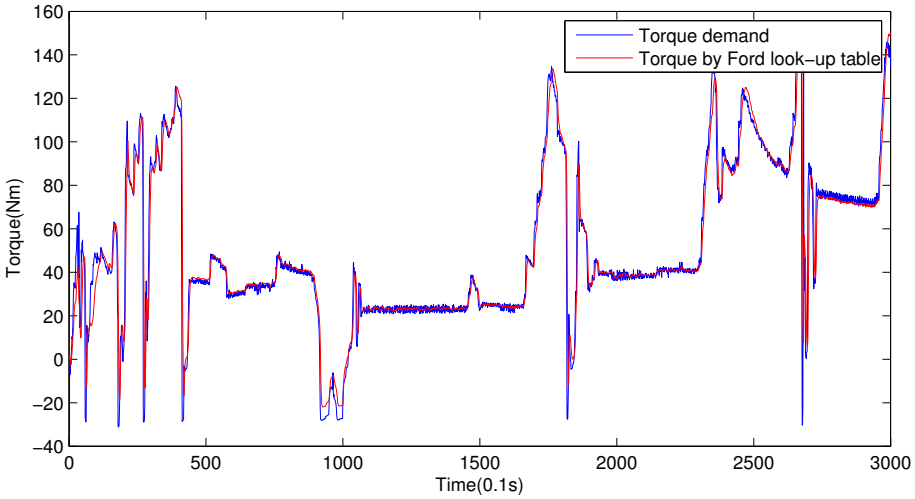

Figure 3.1: Control performance of look-up-tables

is done by online torque estimation that is implemented in the ECU. The big disadvantage of this approach is additional testing time on the engine test bed to calibrate it.

The control performance of the look-up-tables are evaluated over the EUDC drive cycle is presented in Figure 3.1. The resulting mean values of NOx and PM in feed gas are 126.73 ppm and 2.15 g/h. Those values are used later in the methodology as limits NOx and PM emissions.

3.2.3 Emission models

The virtual engine was equipped with emissions models to represent engine-out emissions in contrast to tailpipe emissions. Tailpipe emissions are very different from engine-out emissions and are strongly influenced by the after-treatment system and this is not a subject of this work. The emissions production is modelled as a static mapping of the dynamically produced in-cylinder states. This data-driven mapping is developed based on experimental data from emission analysers from the subject engine.

The data for developing this model were collected in 29 speed-load operating points. Polynomial sub-models of order two have been fitted using the Model

Based Calibration (MBC) toolbox with point-to-point models. A DOE methodology has been employed to generate calibration space for each of the 29 points for the different calibration inputs. The calibration inputs for those points are: Exhaust Gas Recirculation (EGR) rate, intake manifold pressure, pilot injection mass, pilot timing, main injection timing (SOI) and fuel rail pressure. For the sake of simplicity, the fuel rail pressure during the whole process of dynamic calibration was controlled by the map which was empirically tuned by Ford Motor Company and all pilot inputs were set as constant. Additionally engine speed and torque were not direct inputs, but were only used to interpolate sub-models between operating points.
The interpolation between sub-models is obtained by employing a *3 nearest neighbours algorithm* and *barycentric norm*.
The 3 nearest neighbours algorithm selects the 3 most adjacent points, between which the interpolated point is located, from the 29 available as shown in Figure 3.2. It is developed by employing an L_1 distance function known also as a Manhattan distance as follows:

$$d_1 = |x_1 - x_2| + |y_1 - y_2| \tag{3.1}$$

where d_1 is a Manhattan distance, x_1 and y_1 are X-Y coordinates of a first point, x_2 and y_2 are X-Y coordinates of a second point.
The d_1 distances between the 29 points and the interpolated point are calculated and the 3 with the shortest distance are then taken as the 3 adjacent points. When the interpolated point is outside the operating envelope the two closest points are selected and interpolation is made between those points.

After the 3 most adjacent points are selected the barycentric norm is applied to calculate 3 weights that are proportional to the distance from the interpolation point to the adjacent points. The final interpolated output is computed as the sum of three outputs from the 3 sub-models associated with the 3 nearest neighbour points, each multiplied by barycentric weight as illustrated on Figure 3.3, where example of weights for couple of points are presented in brackets. The

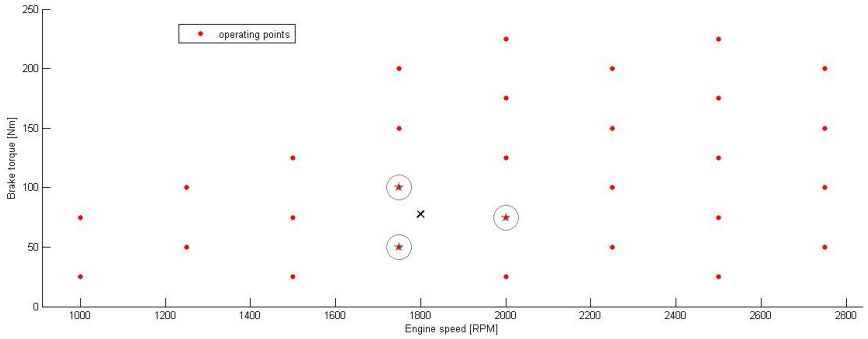

Figure 3.2: The Three Nearest Neighbours algorithm

barycentric weights are calculated as in equations 3.2.

$$W_1 = \frac{(y_2 - y_3)(x - x_3) + (x_3 - x_2)(y - y_3)}{(y_2 - y_3)(x_1 - x_3) + (x_3 - x_2)(y_1 - y_3)}$$
$$W_2 = \frac{(y_3 - y_1)(x - x_3) + (x_1 - x_3)(y - y_3)}{(y_2 - y_3)(x_1 - x_3) + (x_3 - x_2)(y_1 - y_3)} \tag{3.2}$$

$$\mathrm{W}_3 = 1 - W_1 - W_2$$

where x_n,y_n are X-Y coordinates of the n-th adjacent sub-model; x,y are X-Y coordinates of the interpolated point and W_n is the barycentric weight for the n-th sub-model. The emission model was coded in MATLAB as a S-function for implementation for the virtual model. The S-function was then imported into SIMULINK and the emission model was combined with the virtual model. Because for each sub-model was developed based on a individual set of calibration values, each sub-model has a individual input range space. Accordingly a dynamic saturation had to be applied to the sub-model inputs to ensure that the values fed into individual sub-model are within range for each sub-model.

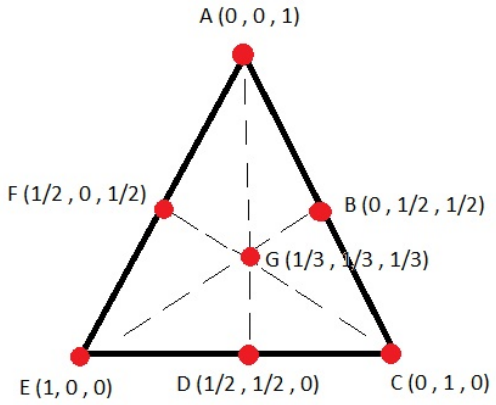

Figure 3.3: Illustration of barycentric weights for 3 points. For example at the point A the weights are as follows: $w_1 = 0$ $w_2 = 0$ and $w_3 = 1$

.

3.3 Dynamic calibration

The objective of the dynamic calibration methodology presented in this chapter is to obtain a feedforward controller that minimizes the fuel consumption of the subject engine with torque tracking of driver torque demand T_d and meeting NOx and PM limits. The Figure 3.4 shows the feedforward controller F together with identified mean value engine model G_1 of engine torque and engine emissions.

3.4 Identification

The first stage of dynamic calibration is a system identification process of obtaining behavioural data-driven description of the subject engine and engine-out emissions. For this work a black box modelling technique was used to minimise prior knowledge required about the system and to keep this method as general as possible for wide range of possible applications. This work presents only one application of this method but this method can by extended to more complicated calibration tasks using more inputs.

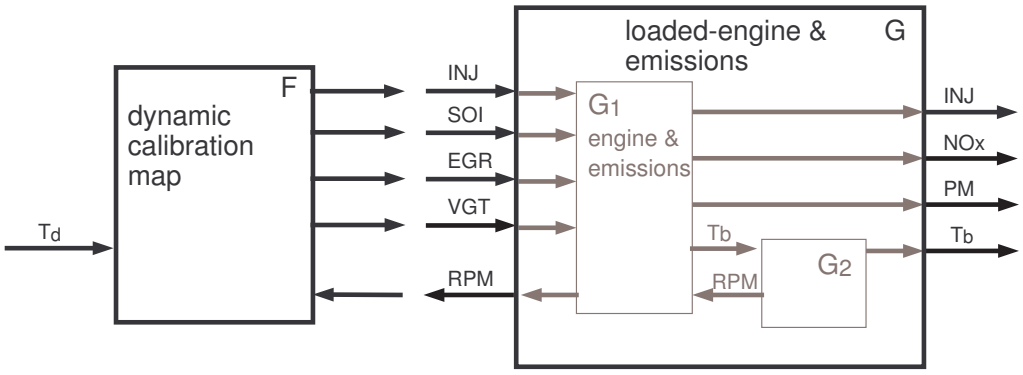

Figure 3.4: Feedforward controller and mean value engine model

The mathematical model employed to describe the engine behaviour needs to be able to represent the dynamics and non-linearities of the engine. To achieve that a rich data set must be obtained from the data collection process. The choice of rich excitation signals has a strong influence on the final results of the system identification. The input sequence should be sufficiently rich not only to excite the key frequencies of the system but also the non-linearities.
For this work APRBS signal was used to excite the following inputs:

- Start of Injection (later referred to as SOI)
- Exhaust Gas Recirculation position valve (later referred to as EGR)
- Variable Geometry Turbocharger position valve (later referred to as VGT)

Furthermore in an experimental application with a new engine, the safe upper and lower limits of the input channels would have to be determined by iterative experimental process or any sort of boundary search algorithm. However the limits for each input in this work were determined from prior experience with the subject engine with the existing calibration operated over the EUDC drive cycle, and were determined as:

- -0.7 rad <= SOI <= 0 rad
- 0 m <= EGR<= 0.02m
- 40% <= VGT <= 100%

It was found that the above constraints were more strict than the boundaries found applicable in the final control maps so the results of the final constrained optimisation was to some extent conservative. Improved results in future work might be expected with more relaxed constraints.
Because fuel injection mass (INJ) has the major influence on the torque output, a PI controller was used to track the EUDC profile of engine torque and to keep the experiment in safe conditions. The engine speed input was chosen also as the speed profile of the EUDC drive cycle.
Identification signals employed in this work can be seen in Figure 3.5. The sampling time was selected 0.1s and a 3000-point data sequence was recorded for identification.

3.4.1 Model Structure

In this work a type of dynamic recurrent Neural Network (NN), that realises as a Non-linear AutorRegressive with eXogenous inputs (NARX) structure, was employed to model the dynamic behaviour of the engine with emissions as both have a dynamic and non-linear nature [74]. As all three neural networks models are used for simulation of the system during each iteration of the optimisation stage, the NN structure is selected to have parallel structure which obtains the current output from previous inputs and previously simulated outputs. With such structuring of the obtained black box models there is no need for further experimental testing in the model-based optimisation and controller design process, which additionally reduces test-bed time.

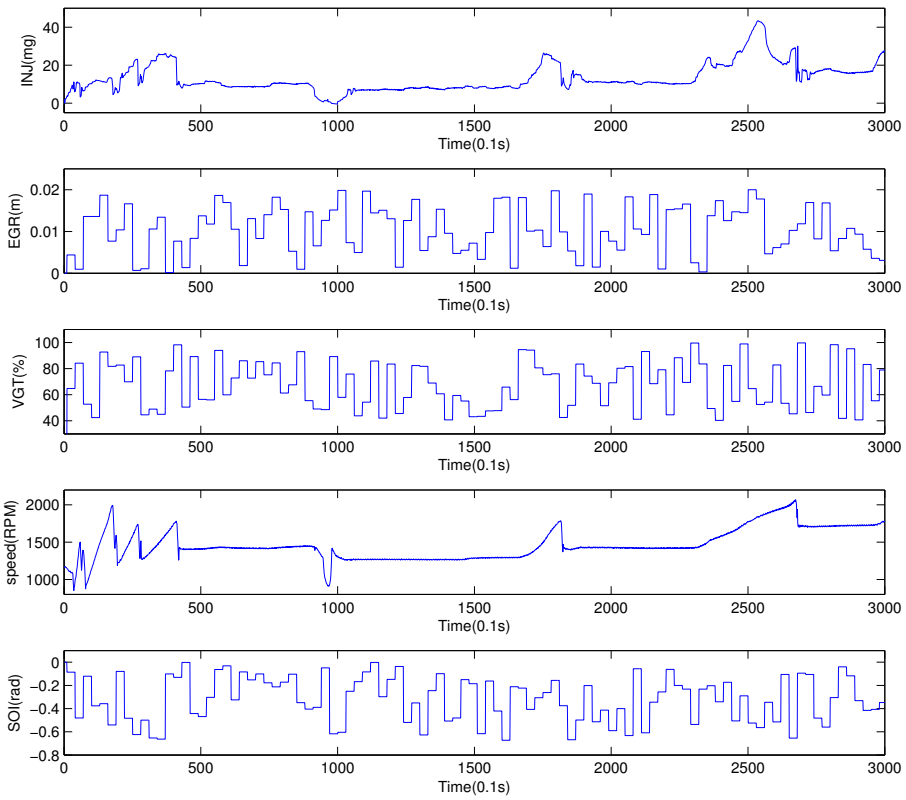

Figure 3.5: Identification signals

The model structure can be described as follows:
$Tb(n) = f(INJ(n-1), ..., INJ(n-5), SOI(n-1), ..., SOI(n-5),$
$EGR(n-1), ..., EGR(n-5), VGT(n-1), ..., VGT(n-5), RPM(n-1), ..., RPM(n-5), Tb(n-1))$

$NOx(n) = f(INJ(n-1), ..., INJ(n-5), SOI(n-1), ..., SOI(n-5),$
$EGR(n-1), ..., EGR(n-5), VGT(n-1), ..., VGT(n-5), RPM(n-1), ..., RPM(n-5), NOx(n-1))$

$PM(n) = f(INJ(n-1), ..., INJ(n-5), SOI(n-1), ..., SOI(n-5),$
$EGR(n-1), ..., EGR(n-5), VGT(n-1), ..., VGT(n-5), RPM(n-1), ..., RPM(n-5), PM(n-1))$

The neural networks is structured as:

- one hidden layer
- one output layer

The number of neurons at hidden and output layer is 10 and 1 respectively. The input is a vector containing 5 inputs, the output is a scalar.The default structure was used but it was found effective. The model contains 5 time delays for the input channel and one time delay for the output channel so the relation in general can be represented as follow:

$$y(t) = f(u(t-1), ..., u(t-5), y(t-1)) \tag{3.3}$$

Figure 3.6 presents the architecture of NARX model in the standard format of the Matlab NN toolbox. The hidden layer has 10 neurons with a tansigmoid activation function and the output layer has 1 neuron with a linear activation function. The Levenberg-Marquardt backpropagation method is employed as the training algorithm.

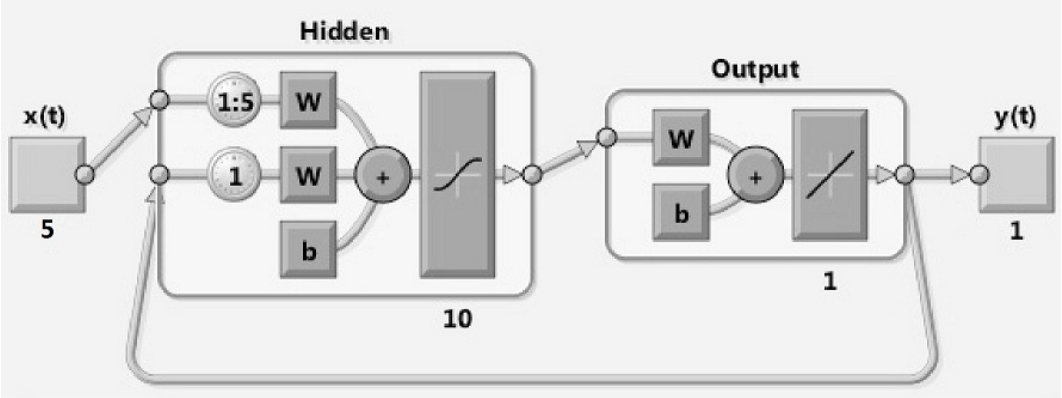

Figure 3.6: Structure of dynamic Neural Network model

Table 3.1: System identification results

Model	R^2
T_b	98.41%
NOx	97.44%
PM	87.75%

3.4.2 Identification results

A multiple correction coefficient R^2 is employed to measure output fitness as given

$$R^2(Y, \hat{Y}) = 1 - \frac{\left\|\hat{Y} - Y\right\|^2}{\left\|Y - \overline{Y}\right\|^2} \tag{3.4}$$

where $\overline{Y}$ is the mean of the output Y.

The results of system identification are presented in Table 3.1. The results show that the identified models of torque and NOx are very accurate and can represent the subject engine and emissions accurately during the optimisation. The PM model gives less accuracy when compared to the torque and NOx models, however using this model acceptable optimisation results are obtained. Figures 3.7, 3.8 and 3.9 show the comparison between measured and simulated data using the NARX structure model for all three identification experiments .

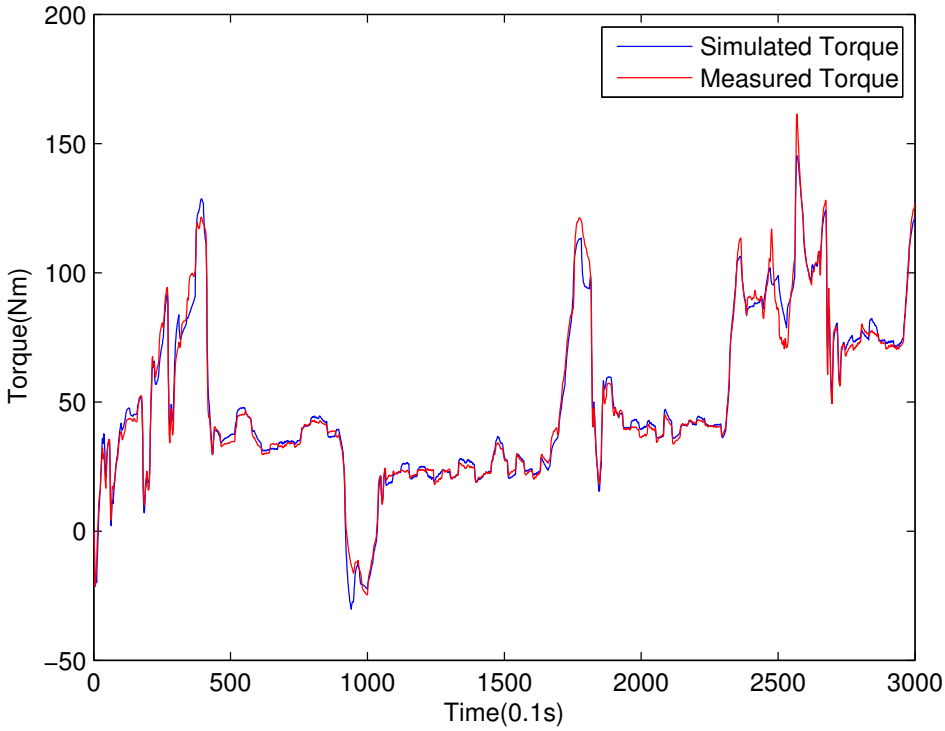

Figure 3.7: Comparison of torque measured and simulated data

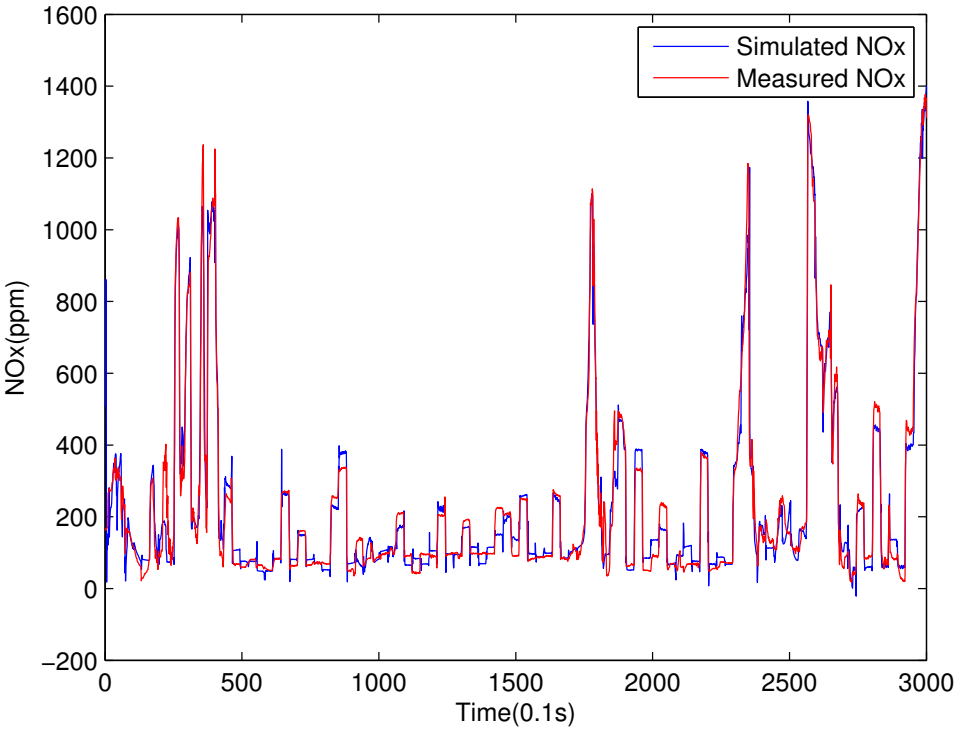

Figure 3.8: Comparison of NOx measured and simulated data

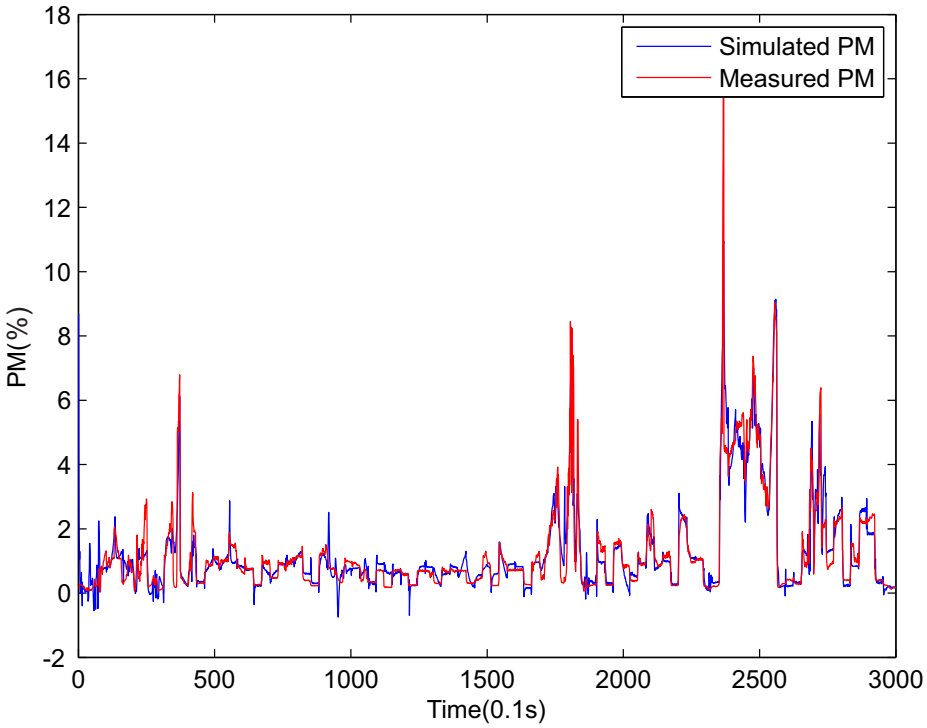

Figure 3.9: Comparison of PM measured and simulated data

3.5 Model-based optimisation

The second stage of the dynamic calibration is a numerical optimisation. In this stage the torque and emission models from the previous system identification stage are utilized to find an open-loop optimal trajectory to minimise the fuel consumption while satisfying the constrains on torque and emissions. In this work the Interior Point algorithm from the Matlab Optimisation Toolbox 2010b [75] is used as the constrained non-linear optimisation routine. The optimisation generates optimal input-output trajectories that satisfy the constraints.

3.5.1 Optimisation implementation

The objective of minimal fuel consumption while satisfying emission limits can by reflected in the numerical optimized behaviour by employing various designs of objective function and constraints. The optimised inputs for the optimisation are INJ, EGR, VGT and SOI at each time instance.
In this work two approaches to objective function design are presented.
In the first case the objective function is determined as the sum of INJ over the EUDC driving cycle and to obtain minimum fuel consumption and is given by:

$$\min_{INJ,EGR,VGT,SOI} \sum_{t=1}^{N} INJ(t) \tag{3.5}$$

subject to:
Equality constraints:

$$T_b(t) = T_{desired}(t) \tag{3.6}$$

$$t = 1, 2, 3..., n \tag{3.7}$$

Inequality constraints:

$$\overline{NOx} \leq 126.73ppm \tag{3.8}$$

$$\overline{PM} \leq 2.15g/h \tag{3.9}$$

The equality constraint is determined by the torque profile from the EUDC drive cycle for the subject engine as in Figure 3.1 in order to track a representative demanded torque and to realize the required engine speed.
The inequality constrains on NOx and PM are selected according to the mean values of the emissions over the data sequence because of the legislative regulation on the total amount of emissions.
At each time instance the inputs of INJ, EGR , VGT and SOI are changed to obtain a minimized fuel consumption while satisfying emission limits. So at each time instance there are 4 manipulated variables and so for 3000 data points which gives a total of $4 \times 3000 = 12000$ variables.
The second approach is based on a different objective function design which is configured to maximize the mean torque over EUDC which decreases computational burden significantly. The objective function is thus given as:

$$\max_{EGR,VGT,SOI} \sum_{t=1}^{N} T_b(t) \tag{3.10}$$

subject to:
Inequality constraints:

$$\overline{NOx} \leq 126.73ppm \tag{3.11}$$

$$\overline{PM} \leq 2.15g/h \tag{3.12}$$

At each time instance the inputs of EGR, VGT and SOI are changed to find the maximum of the objective function, while torque tracking is guaranteed by fixed INJ because of very strong relationship between torque and INJ. In this case now number of variables is reduced to $3 \times 3000 = 9000$.
The resulting optimisation obtains optimal input-output data, that for the given fuel trajectory maximizes mean torque, instead of focusing on fuel optimisation directly.
The variables of the entire data sequence are ideally manipulated in a one stage optimisation to achieve full global optimality for given optimisation problem.

However this leads to very high demand in memory resources and leads to a long computational time which makes this process computationally inefficient.
To solve this problem to further reduce the computational work in each iteration, a novel segment approach is proposed. For this approach the whole optimisation problem of 3000 data points is split into 15 sub-problems called segments, each of 200 data points. Optimisation is performed sequentially on each separate segment. This way each segment contains much fewer variables and constraints and so it results in a significantly reduced computational burden at the expense of true global optimisation.

3.5.2 Optimisation results

Optimisation was run in the Matlab environment using the Interior Point algorithm with 40000 numerical iterations. The results from the optimisation are presented in Figure 3.10. It took 5188s (= 1 hour 27 minutes) to compute the results on standard 1.6 GHz PC. The obtained optimal inputs are non-smooth and noisy as on some part of the data set the input values change rapidly due to the setting of the optimisation. The reason is that each time instance is treated as an individual set of variables and the algorithm adjust those variables in order to meet all the constraints while maximising the objective function. Since all variables are considered independently this is why the final optimal inputs are not smooth. However the next step of the dynamic calibration process is where a feedforward controller is developed and its structure is predetermined to produce smoothed output signals.
To verify the performance of the optimal solution found by the optimisation, optimal inputs were applied back to the virtual engine as open-looped trajectories. The optimized mean torque with this same fuel consumption has $\overline{T_{op}} = 51.43Nm$ and it is compared to the original mean values of torque $\overline{T_0} = 50.07Nm$. The results show improvements in the ratio:

$$\overline{T_{op}}/\overline{T_0} = 1.012 \tag{3.13}$$

The resulting optimal emission are $\overline{NOx_{op}} = 117.98ppm$ and $\overline{PM_{op}} = 1.19g/h$

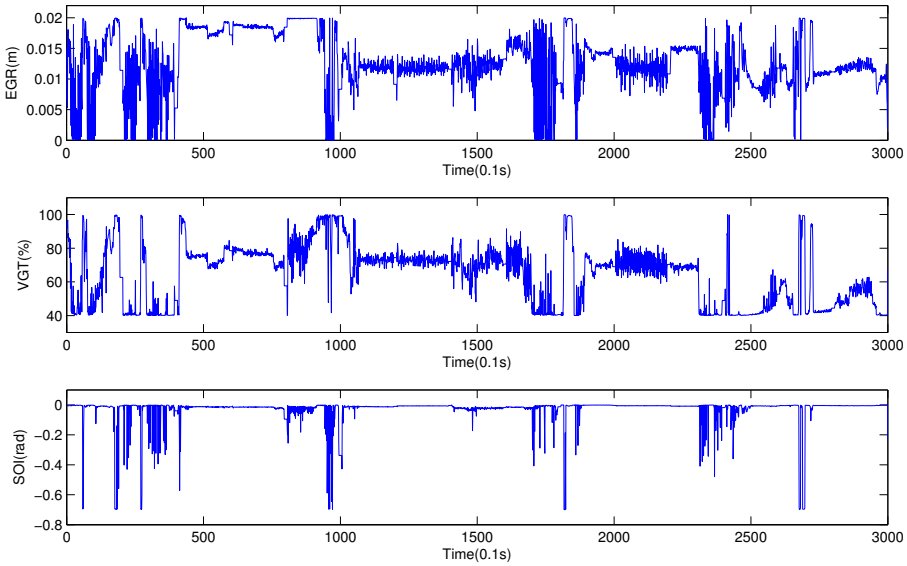

Figure 3.10: Optimal input signals

which satisfy the emission constraints set for the optimisation. The effectiveness of the optimal inputs in the virtual engine system test is thus proved.

3.6 Inverse identified dynamic controller design

In this stage the optimal inputs derived from the optimisation stage are used to obtain an optimal feedforward controller by identification from the inverted optimal data set. The inverse data set was determined as follows:

- torque output to act as an input
- INJ, EGR VGT and SOI inputs to act as outputs
- engine speed input to act as an input

In experimental applications of the method the input-output data would have to be time-shifted as in [76] to remove time delays from the optimal data set before the inputs and outputs can be swapped. In this application where signal measurement delays were not modelled in both the dynamic and the conventional

Table 3.2: Inverse identification parameters

	EGR	**SOI**	**VGT**	**INJ**
θ_1	-0.023	-0.26	-0.10	2.96
θ_2	-2.24x10^{-5}	2.42x10^{-4}	-0.0025	0.098
θ_3	3.81x10^{-5}	2.64x10^{-4}	3.77x10^{-4}	-0.0028
θ_4	0.30	-3.01	0.89	0.51
θ_5	-7.26x10^{-7}	-6.62x10^{-6}	2.20x10^{-5}	-6.79x10^{-6}
θ_6	8.91x10^{-8}	3.42x10^{-7}	-2.25x10^{-6}	-2.90x10^{-5}
θ_7	-0.0065	-0.0099	0.0027	2.75x10^{-4}
θ_8	-1.39x10^{-8}	-7.47x10^{-8}	-6.53x10^{-9}	-2.25x10^{-4}
θ_9	3.77x10^{-4}	0.0025	2.25x10^{-4}	1.07x10^{-4}

calibration applications such time-shifting was not found necessary.
Four inverse MISO models are thus obtained to reproduce the optimal inputs trajectories for: INJ, EGR, VGT and SOI with desired torque and engine speed RPM as inputs so as to obtain the feedforward controller F of Figure 3.4.

In this study a polynomial structure was employed to implement the controller. The polynomial structure only has a limited capability to model non-linear behaviour but is easier to implement then neural networks. A second order Volttera structure including output regressors together with input regressors and their products was predetermined as follows:

$$\begin{aligned} y(t) &= \theta_1 + \theta_2 u_1(t-1) + \theta_3 u_2(t-1) + \theta_4 y(t-1) + \theta_5 u_1(t-1)^2 \\ &\quad + \theta_6 u_1(t-1)u_2(t-1) + \theta_7 u_1(t-1) + \theta_8 u_2(t-1)^2 + \theta_9 u_2(t-1)y(t-1) \end{aligned} \tag{3.14}$$

where u_1 denotes the desired torque and u_2 denotes the engine speed. The parameters obtained in this stage are presented in Table 3.2.

Comparison of simulated and optimal inputs are shown in Figure 3.11. The corresponding R^2 that was obtained to measure quality of control input repro-

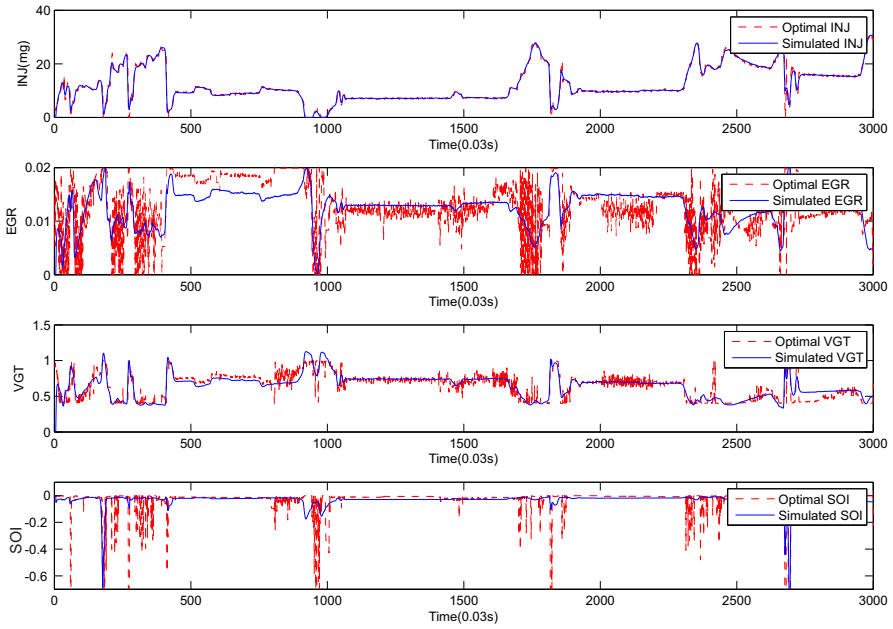

Figure 3.11: Optimal and simulated input signals

ductions for the model described above are given in Table 3.3. The better that the R^2 parameter is the better the inputs match the optimal inputs from model-based optimisation. Ideally R_2=100% would lead to perfect reproduction of the optimal inputs. For the EGR and SOI inputs the figures shows that the resulting representation is worse than for the INJ and VGT signals.
On the other hand this leads to a smoother input sequence which is also desirable as a fast changing input sequence has a significant impact on faster actuator wear. The final controller can never perfectly match the optimal signals although some improvements could be achieved by applying some form of model structure selection algorithm or a using different model type.

3.7 Controller performance validation

Experimental validation was carried out by applying the inverse identified controller to the virtual engine. During the experiment torque and emission were

Table 3.3: Identified controller results

R^2_{INJ}	97.95%
R^2_{VGT}	75.22%
R^2_{EGR}	40.82%
R^2_{SOI}	32.81%

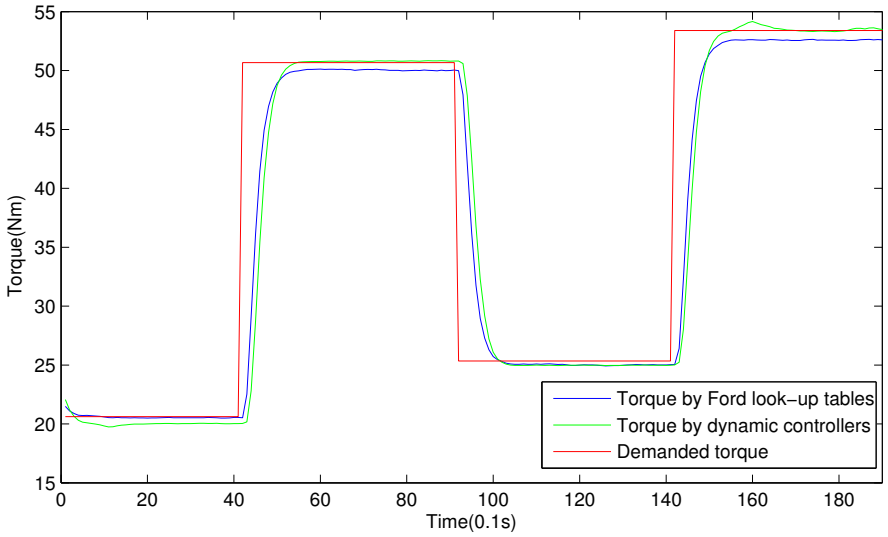

Figure 3.12: Validation results for random torque steps

captured and compared to the results of the conventional look-up table controller.

3.7.1 Random torque demand

The first test used a random torque demand with a 35 sec interval to validate the performance of tracking the torque. The Figure 3.12 presents a comparison of outputs obtained for the conventional look-up table controller and the dynamic controller.

The dynamic controller is very quick in response to torque changes, but also the controller tracks the torque closely to the demanded value.
One way of measuring the fuel economy is to measure how much torque was

Table 3.4: Fuel economy and emissions on random validation data

	$\overline{e}(Nm/mg)$	$\overline{NOx}(ppm)$	$\overline{PM}(g/h)$
Look-up table	3.89	73.83	1.13
Dynamic controller	3.93	63.12	1.04

generated relative to the amount of fuel used to generate this torque. The corresponding mathematical formula for this measure is :

$$\overline{e} = \frac{\overline{T_b}}{\overline{INJ}} \tag{3.15}$$

The fuel economy obtained using this measure and emission from the random torque demand validation are given in Table 3.4. The results show that the dynamic controller improves fuel economy by 1% in comparison to the conventional look-up table controller but also reduces NOx and PM emissions.

3.7.2 EUDC validation

A second test was carried out to validate the dynamic controller performance in practical application. The test was carried out on the EUDC drive cycle. Figure 3.13 presents a comparison of outputs obtained from the conventional look-up table controller and the dynamic controller.

The results are given in Table 3.5 and it can be observed that both controllers give quick and accurate control.

It can be seen that there is a 2% improvement in fuel economy for the dynamic control compared to the conventional controller. The emissions for the dynamic controller do not exceed those generated by the static look-up table controller.

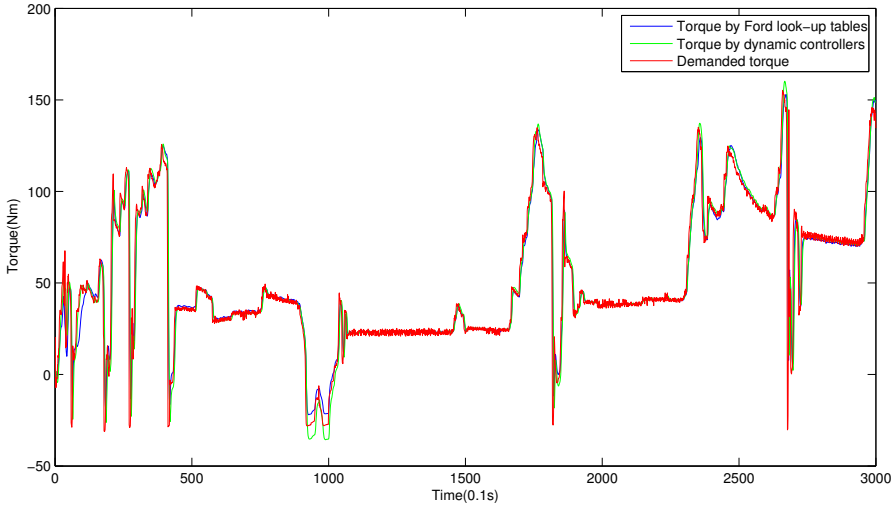

Figure 3.13: Validation results for EUDC driving cycle

Table 3.5: Fuel economy and emissions on EUDC validation data

	$\overline{e}(Nm/mg)$	$\overline{NOx}(ppm)$	$\overline{PM}(g/h)$
Look-up table	4.17	126.73	2.15
Dynamic controller	4.25	121.11	1.18

3.8 Conclusion

The novel dynamic calibration methodology presented in this chapter consists of a dynamic global modelling approach employing a system identification process, constrained numerical model-based optimization to produce optimal input-output trajectories and inverse behaviour identification to identify a dynamic controller to reproduce this optimal behaviour. The dynamic calibration methodology was evaluated on a virtual 1.5L Diesel engine and it was shown that this methodology is effective, resulting in improved fuel economy over a static calibration, satisfying emissions constraints, and giving smooth torque tracking with significant saving in testing time on the engine test bed due to limited test data required.

Chapter 4

Optimisation of engine states and inputs trajectory for engine calibration using Dynamic Programming

4.1 Introduction

In this chapter a dynamic programming method is evaluated as a possible optimisation approach to find optimal trajectories of inputs for the optimisation stage of a dynamic calibration. The main advantage of dynamic programming over other optimisation methods is that global optimality of the found solution is guaranteed, regardless of the type of problem.
The investigation in this chapter was carried out on this same virtual WAVE RT engine model as in chapter 3. The work included identification of a Volterra model of the engine torque and the implementation of numerical dynamic programming for torque tracking and fuel consumption minimization.

4.2 Optimal control

One subject of optimal control is to determine the input of a dynamic system that minimizes or maximizes a predefined cost function sometimes called a performance function or performance index, while satisfying any constraints on the behaviour of the system with fixed time and partially constrained final state. A lot of interest in the literature is focused on optimisation of continuous time problems and such problems can be described as follows:

$$\begin{aligned} \min_{x} \quad & J(u(t)) \\ \text{s.t.} \quad & \dot{x}(t) = f(x(t)), u(t), t) \\ & x(0) = x_0 \\ & x(t_f) \in [x_{f,min}, x_{f,max}] \end{aligned} \tag{4.1}$$

where

$$J(u(t)) = G(x(t_f)) + \int_{t_0}^{t_f} H(x(t)), u(t), t)dt \tag{4.2}$$

is the cost functional.

The objective of the optimal control solution as defined is to find the input u_{opt} that minimizes the cost function during the time internal t_0 to t_f.

Only some simple problems for linear dynamic systems with quadratic cost function can be solved analytically and even then simplicity of the problem solution is not that easy to obtain [77]. Such solution includes many matrix operations (inversions of matrix, etc.) and also it includes solving Riccati-equations which present difficult mathematical problems.
Because of the complexity of most applications and the non-linear, time-variant character of the dynamic system and constrained problem such as in engine calibration, optimal control problems must usually be solved numerically.
The work of Bellman [78] from the 1950s initiated many research activities in optimal control and nowadays the range of available methods and the corresponding variety of published applications has increased tremendously making optimal con-

trol a discipline that is relevant to many branches of engineering [79]. The need for solving optimal control problems numerically has given rise to a wide range of numerical methods that have been developed specifically for optimal control problems. These methods can be divided into three major groups[80]:

- indirect methods - converting the original control problem to a boundary-value problem
- direct methods - converting the original control problem into a non-linear programming problem
- dynamic programming

4.2.1 Indirect methods

The optimisation is a minimization of cost function J, subject to constraints on the system dynamics and the initial and final conditions. In the indirect approach to the optimisation problem time-varying Lagrange multipliers $\lambda(t)$ for continuous time systems or λ_i for discrete time systems are introduced. The solution to the boundary value problem must be solved numerically but firstly the differential equations involving the Lagrange multipliers must be obtained and these are often strongly non-linear.
The algebraic expressions linking cost to the system states are rather complex. Since the scope of this project included the utilization of strongly non-linear equations for description of the system dynamics for the engine and emissions including the use of complex Neural Network models with multiple time delays, analytically differentiating these terms with respect to state and control variables was not feasible.
However, such differentiations are required to express the differential equations involving the Lagrange multipliers so consequently this approach was beyond the scope of this thesis which focuses on the other two.

4.2.2 Direct methods

In contrast to indirect methods direct methods are applicable to the complex non-linear dynamics required in the engine calibration process. New solution for discrete optimal control problem using discretization of the Lagrange-d'Alembert principle and advantages of such approach are presented in [81].
In general direct methods can be divided into two groups:

- Sequential methods
- Simultaneous methods

The most intuitive direct method is the single shooting approach which belongs to the class of sequential methods. For this approach u(t) is discretized in time using an appropriate discretization method to obtain the vector of control parameters for time discretized on N_t intervals as follows:

$$t_0 < t_1 < t_2 ... < t_N = t_f \tag{4.3}$$

For each individual interval there is a respective control parameter as follows:

$$\text{U}=[u_0, u_1, ..., u_N] \tag{4.4}$$

The optimisation algorithm uses explicit numerical integration to satisfy the differential equations. Then the cost function J is calculated for each iteration and for the next iteration control parameter vector is upgraded according to the optimisation algorithm [82]. A block diagram of this method is presented in Figure 4.1 and example of one iteration of the algorithm is shown in Figure 4.2. A similar approach is used in chapter 3, but necessary changes were introduced for discrete time system.
Arguably the most powerful methods for solving general optimal control problems are direct collocation methods which fall into the simultaneous methods group. For continuous time the optimal control problem is transformed into a finite dimensional problem by an appropriate discretization of controls and states which then can be solved by a standard SQP-method [83]. For the sake of simplicity the number of discretization points of inputs and state are taken same.

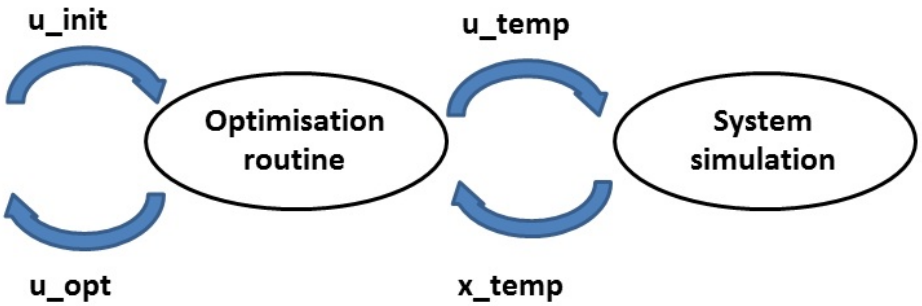

Figure 4.1: Single shooting approach

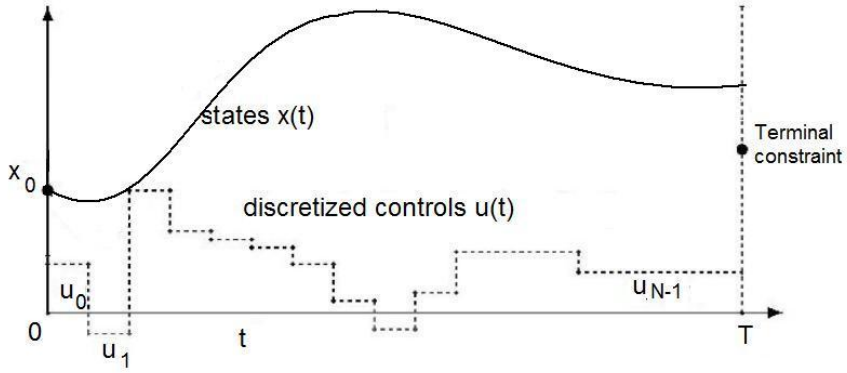

Figure 4.2: Illustration of single shooting [4]

The main difference between single shooting and direct collocation is in the problem formulation of the optimisation parameters. Instead of using only control parameters U the optimisation parameter vector also includes state parameters as follows:

$$Y_u = [u_0, u_1, .., u_f, x_0, x_1 ... x_N] \tag{4.5}$$

In addition some form of discretization of the system takes place. In equation 4.6 piecewise approximation of continuous is presented but there exists more advance collocation method [84]. The original optimal control problem now is formulated as follows:

$$
\begin{aligned}
\min_{Y_u} \quad & J \\
\text{s.t.} \quad & x_1 - [f(x_0, u_0) * dt + x_0] = 0 \\
& x_2 - [f(x_1, u_1) * dt + x_1] = 0 \\
& . \\
& . \\
& . \\
& x_N - [f(x_{N-1}, u_{N-1}) * dt + x_{N-1}] = 0
\end{aligned}
\tag{4.6}
$$

Using direct local collocation is now possible with large sparse non-linear programming with thousands to tens of thousands of variables and a similar number of constraints. This means that direct local collocation makes it feasible to solve optimal control problem efficiently using appropriate NLP solvers such SPRNLP, SNOPT and KINTRO [85].

The direct multiple shooting method combines elements of both single shooting and direct collocation. It is simultaneous approach similar to direct collocation where controls and system states are discretized in time. It differs from the direct collocation method since some ODE solver is used to simulate the system dynamics between discretization points instead of using piecewise approximation as shown in Figure 4.3.

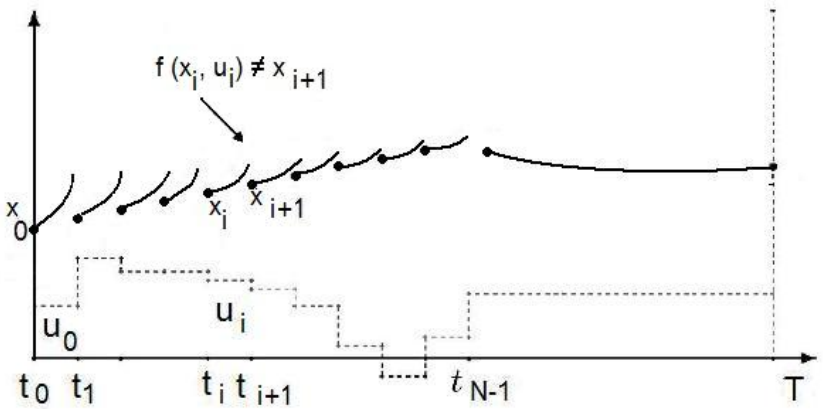

Figure 4.3: Illustration of multiple shooting [4]

Discrete direct methods

Although direct methods were developed for solving optimisation control problems for continuous systems similar approaches can be applied to discrete time systems.
In chapter 3 the optimisation problem was formulated in a way similar to that of the single shooting method described above. The optimisation parameters include only the control parameters. Also system dynamics are represented in input-output representation which does not allow the creation of a suitable set of constrains for NLP solvers.
In chapter 5 the optimisation problem was formulated in a way similar to that used in direct collocation or multiple shooting. The optimisation parameters includes control and state parameters, and due to a state-space representation used it was possible to employ an NLP solver to find optimal solution.

4.2.3 Dynamic programming

Dynamic programming is a very powerful numerical method for solving optimal control problems [78] [86] [87]. Dynamic programming was derived from Bellman's Principle of Optimality based on the idea that the optimal solution of sub-problems can be used to build the optimal solution of a larger problem. This algorithm is often used in solving shortest path problems. In case of optimal control problems to find the optimal solution firstly requires that the whole time interval $[t_0, t_f]$ is divided into shorter time intervals which are solved over consecutively in a backward direction. From the Principal of Optimality if an optimal control solution for the short time interval is found, it must be part of the optimal solution for the whole interval. The resulting optimal trajectory built with the shorter optimal trajectories is the global solution within the search space. The equation of Bellman's Principle of Optimality is as follows [88]:

$$V(x(k), k) = \min_{u(k)}[J(u(k), x(k) + V(k+1, f(x(k), u(k), k)] \tag{4.7}$$

for each $k = 1, 2, ..., N$.

4.2.4 Numerical Dynamic programming

Dynamic programming can be applied to either continuous-time or discrete-time systems with continuous or discrete time inputs. Since the focus of this project is data-driven models obtained from sampled data, only the implementation of dynamic programming for discrete-time systems is included in this work.
For discrete-time systems with n states and m inputs the general optimal control problem is summarized us follows: find an admissible control sequence $u(k)$, for $k = 0, 1, ..., N$ such that the cost functional J is minimized and the all constraints are satisfied. That is

$$\begin{aligned} \min_{u(k)\in U_k} \quad & J(u(k)) \\ \text{s.t.} \quad & x(k+1) = f(x(k), u(k), k) \\ & x(0) = x_0 \\ & x(N) \in T \subseteq \mathbb{R}^n \\ & u(k) \in U_k \subseteq \mathbb{R}^m \end{aligned} \tag{4.8}$$

where

$$J(u_k)) = g_N(x(N)) + \sum_{k=0}^{N-1} g_k(x(k)), u(k)) \tag{4.9}$$

is the cost functional. The cost functional contains a final cost term $g_N(x_N)$ associated with the final state and the state cost $g_k(x(k), u(k))$ which is the cost of applying the control signal u_k at discrete time k to the dynamic system given in equation 4.8. Note that the functions g_k and f are allowed to be time-variant, hence there is the index k. The state variables are constrained to the time variant set X_k and the input signals are constrained by the time-variant set U_k. The final value is partially constrained to be the target set T and the initial condition is given by x_0.

Since dynamic programming is discrete in nature the time, state space and control space need to be discretized.

Therefore the state space is discretized at time k to the finite set:

$$\mathrm{X}_k = \{x^1(k), x^2(k), ...x^q(k)\} \tag{4.10}$$

where $x^j(k)$ denotes the state in the discretized state-time space at the time k and j-th state index of the state grid q. Similarly, the control space is represented by the finite set as follows:

$$\mathrm{U}_k = \{u^1(k), u^2(k), ...u^p(k)\} \tag{4.11}$$

where $u^j(k)$ denotes the state in the discretized control-time space at the time k and j-th state index of the state grid p. Based on the principle of optimality, dynamic programming evaluates the optimal cost-to-go function $V_k(x^i(k))$ at every node in the discretized time-space by proceeding backwards in time. The algorithm works as follows:

1) Initialisation of cost-to-go function

$$V_N(x^i(k)) = \begin{cases} g_N(x^i(N)), & for x^i(N) \in T \\ infinity, & else. \end{cases} \tag{4.12}$$

2) Backward iteration for $k = N - 1$ to 0, $\forall x^i(k) \in X_k$

$$V_k(x^i(k)) = \min_{u(k) \in u_k} g_k(x^i(k), u(k)) + V_{k+1}(f_k(x^i(k), u(k))) \tag{4.13}$$

The optimal control is given by the argument that minimizes the right-hand side of equation 4.13.
In order to evaluate the cost-to-go at point $x^i(k)$, the algorithm simulates the system by applying all possible controls from the control candidates as in Figure 4.4. Thereby, the system is driven into $f_k(x^i(k), u(k))$ where $u(k) \in U_k$. Since these points do not generally coincide with the state grid, in this work linear interpolation is used when points do not coincide with the grid [89].

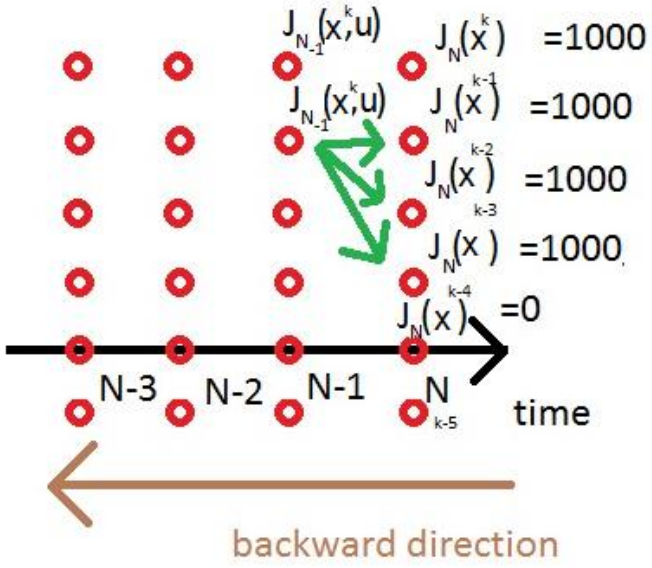

Figure 4.4: Evaluation of optimal cost-to-go

From the results of the dynamic programming optimal inputs on every node at each time step is known, resulting in on optimal control policy

$$\pi = \{\mu(0), \mu(1), ..., \mu(N-1)\} \tag{4.14}$$

Therefore, after the dynamic programming which runs backwards in time to obtain optimal inputs trajectories, the forward simulation of the dynamic system starting from x_0 using the model of the system needs to take place.

4.3 Application of Numerical Dynamic Programming for Dynamic Calibration

4.3.1 Identification

As work in this chapter was to investigate the potential of numerical dynamic programming as an optimisation tool for dynamic calibration, a simple non-linear polynomial model of Volterra series structure was employed to model the output y as engine torque with 5 inputs: u_1 is fuel injection mass (INJ), u_2 is variable

geometry turbocharger valve position (VGT), u_3 is exhaust gas recirculation position valve (EGR), u_4 is start of injection (SOI) and u_5 is engine speed. The structure of polynomial model was that in equation 4.15.

$$\begin{aligned} y(k+1) &= \theta_0 + \theta_1 u_1(k) + \theta_2 u_2(k) + \theta_3 u_3(k) + \theta_4 u_4(k) + \theta_5 u_5(k) + \theta_6 u_1(k)^2 \\ &\quad + \theta_7 u_2(k)^2 + \theta_8 u_3(k)^2 + \theta_9 u_4^2(k) + \theta_{10} u_5(k)^2(k) \\ &\quad + \theta_{11} u_1(k) u_2(k) + \theta_{12} u_1(k) u_3(k) + \theta_{13} u_1(k) u_4(k) \\ &\quad + \theta_{14} u_1(k) u_5(k) + \theta_{15} u_2(k) u_3(k) + \theta_{16} u_2(k) u_4(k) \\ &\quad + \theta_{17} u_2(k) u_5(k) + \theta_{18} u_3(k) u_4(k) + \theta_{19} u_3(k) u_4(k) \\ &\quad + \theta_{20} u_4(k) u_5(k) + \theta_{21} y(k) \end{aligned} \tag{4.15}$$

This structure of this model is similar to the deterministic ARMA structure but with only non-linear regressors terms and so the least-squares method was employed to determine the parameters.
An APRBS signal was used to excite the system and the signals as in Figure 4.5 and the data was captured with a 10Hz sampling rate. The data was normalized for convenience of the input and state discretization but in applications when a neural network model is used, the normalisation speeds up the training process and also helps to avoid local minimum [90]. The normalization was done as follows:

- Torque: Min -29.11Nm - Max 169.15Nm
- INJ: Min 0 mg/stroke - Max 32.74 mg/stroke
- SOI: Min -0.7 Max 0 rad
- EGR: Min 0 m Max 0.02m
- VGT: Min 40% Max 100%
- engine speed : Min 825 RPM Max 2068 RPM

The normalization converted minimum values from the above list to -1 and maximum values to 1.

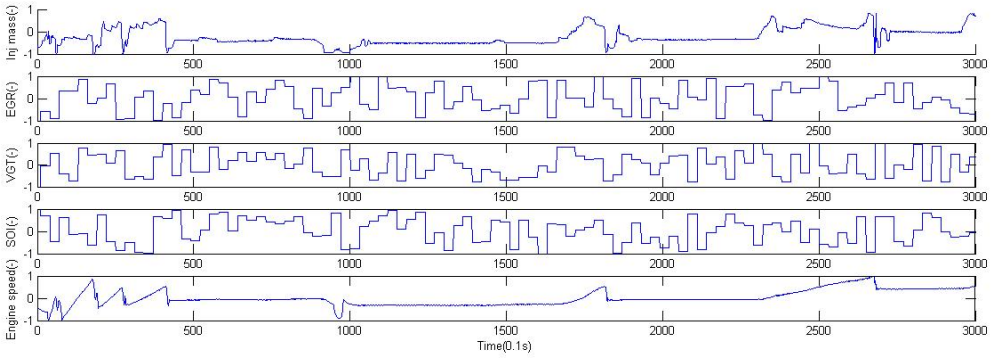

Figure 4.5: Identification signals

The least-squares method was used to obtain the parameters of the engine torque model and because the structure was predetermined no further structure selection was carried out.
The results obtained for the torque modelling are presented in Figure 4.6 which compares the simulated torque and the modelled torque. The quality of the modelling was verified based on the *best fit* norm described in equation 2.37 and achieved 77.97% which is quite low value. However the focus in this chapter was not on developing accurate models with high simulation capability but on the evaluation of numerical dynamic programming for dynamic calibration.

4.3.2 Converting input-output models into state space models

For Numerical Dynamic Programming not every mathematical representation can be used because the model has to be expressed in terms of states because this algorithm runs over each grid point for each individual state and so only models that can be represented in state space are appropriate. For this reason only steady state models or models that can be converted into state space format are appropriate for this algorithm. In the next chapter special State Space Neural Networks are developed that can model the engine accurately and are realizable in state space.

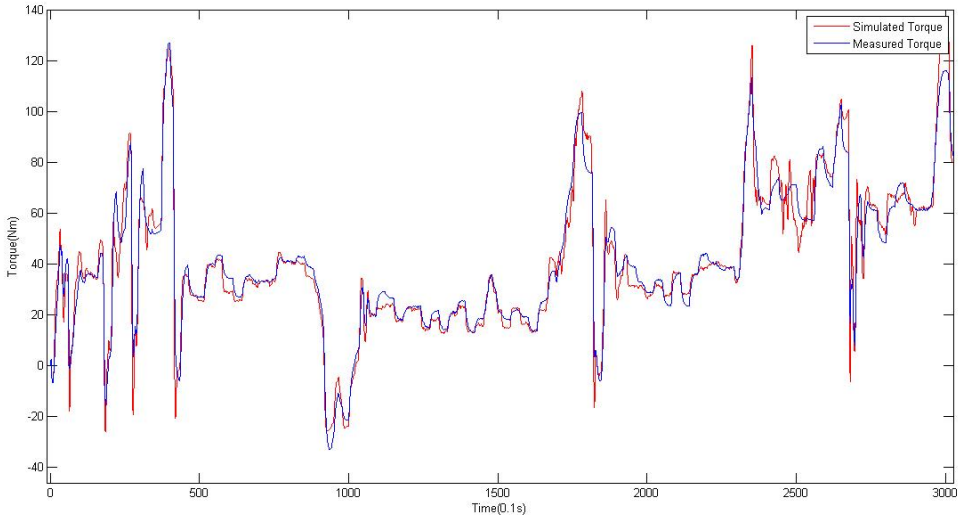

Figure 4.6: Comparison of torque measured and simulated data

Because dynamic programming requires a state space representation of the system dynamics, in this work the input-output model developed in the system identification stage was converted into a state space model. There are systematic approaches for such conversion [91] [92], however this conversion is straightforward due to the very simple model structure containing only one time delay for inputs and output. An input-output model can not be converted to a steady state representation when it includes products of regressors with different time delays, but in this model this is not a case.
The resulting state space model with this model with one state is as follows:

$$\begin{aligned} x(k+1) &= \theta_0 + \theta_1 u_1(k) + \theta_2 u_2(k) + \theta_3 u_3(k) + \theta_4 u_4(k) + \theta_5 u_5(k) + \theta_6 u_1(k)^2 \\ &\quad + \theta_7 u_2(k)^2 + \theta_8 u_3(t)^2 + \theta_9 u_4^2(t) + \theta_{10} u_5(k)^2(k) \\ &\quad + \theta_{11} u_1(k) u_2(k) + \theta_{12} u_1(k) u_3(k) + \theta_{13} u_1(k) u_4(k) \\ &\quad + \theta_{14} u_1(k) u_5(k) + \theta_{15} u_2(k) u_3(k) + \theta_{16} u_2(k) u_4(k) \\ &\quad + \theta_{17} u_2(k) u_5(k) + \theta_{18} u_3(k) u_4(k) + \theta_{19} u_3(k) u_4(k) \\ &\quad + \theta_{20} u_4(k) u_5(k) + \theta_{21} x(k) \end{aligned} \tag{4.16}$$

4.3.3 Numerical Dynamic Programming implementation for input optimisation

The optimisation problem is to find optimal inputs over given driving cycle while tracking demanded torque and minimizing fuel consumption. The length of this numerical experiment is $N = 3000$ stages.
The discretization was predetermined as follows:

- 101 points grid for state x
- 41 grid points for input u_1
- 11 grid points for input u_2
- 11 grid points for input u_3
- 11 grid points for input u_4
- no grid for u_5, because engine speed needs to follow the EUDC engine speed profile.

The higher number of points in the state's and u_1 grid were determined to obtain good torque tracking close to the demanded torque, and to overcome the drawbacks of interpolation. For the other inputs a high resolution was also required but this was limited to make this experiment feasible in terms of computational time.
The cost function was determined to achieve good torque tracking and to minimize fuel consumption. For this reason an LQ (linear quadratic) type of cost function was predetermined, however for this algorithm any non-linear cost function could be employed. The cost function was thus:

$$g_k(x(k), u(k)) = \sum_{k=1}^{N-1} (T_d(k) - x(k))P(T_d(k) - x(k))^T + u(k)Qu(k)^T \quad (4.17)$$

where T_d is the desired torque for torque tracking, with $P = 1$ and

$$Q = \begin{bmatrix} 0.02 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

The cost for the final stage g_N in equation 4.14 was determined as follows:

- if the final state equals the final target the cost is set to 0
- the cost is set to 1000 otherwise.

4.3.4 Results

The time elapsed to complete the numerical dynamic programming implemented as above on a standard 1.6 GHz PC was 88695s (21 hours 38min 15s) . Forward simulation of the dynamic system was run using the optimal control policy π to obtain the optimal inputs trajectories .

Figure 4.7 shows the optimal inputs trajectories obtained from forward simulation and shows that although the global optimum could have been found, it is hard to justify this fact, because the modelling capability of the polynomial model is quite low and so some behaviour of the system might not have been captured.

Verification

The verification of the open-loop trajectories was run for the identified engine and the results are presented on Figure 4.8. It can be seen that the measured torque does not follow exactly the desired output but follows it approximately. More runs of the algorithms would have take place to determine better P and Q parameters for the cost function, but due to the very long computational time, the simplistic engine model and the exponential growth of computational time,the approach was considered too costly and was diverted to direct optimisation approach showing very promising results.

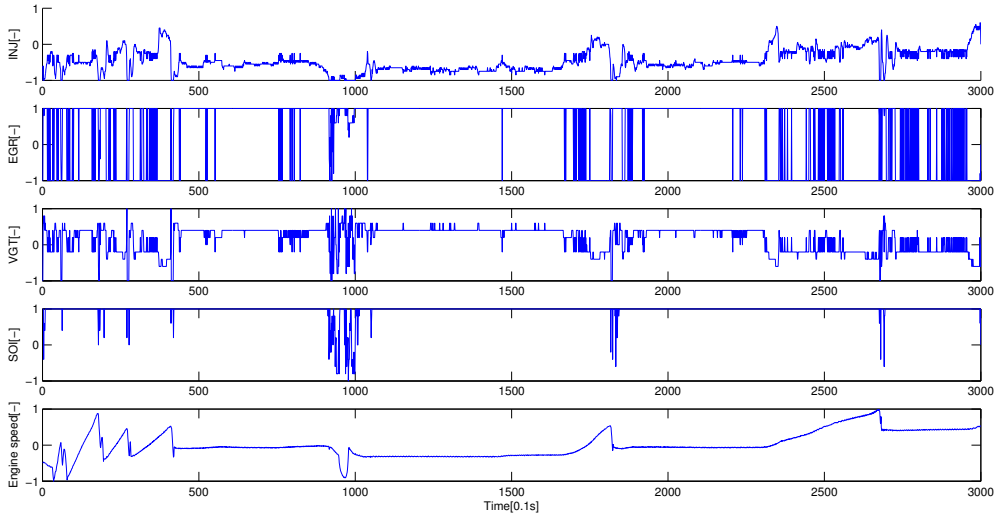

Figure 4.7: Optimal inputs

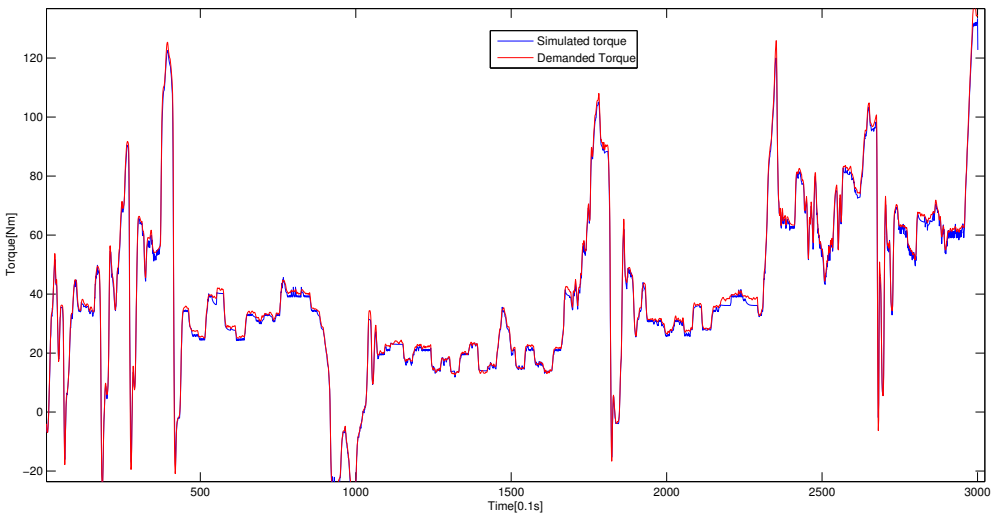

Figure 4.8: Verification of the open-loop trajectories

4.4 Conclusion

The implementation of a numerical dynamic programming approach to finding optimal input trajectories was presented in this chapter including the development of a simple polynomial model to describe the engine torque. The cost function was designed to achieve good torque tracking and minimize the fuel consumption. The results presented show that although optimal trajectories could be found using numerical dynamic programming and could be used for simplified calibration tasks this could only be achieved with very simple models, which are not suitable for describing the engine behaviour. To solve that problem more advanced model techniques would have to be used with the numerical dynamic programming but with the exponential growth of computational time such applications are not actually feasible.

Chapter 5

Behavioural data-driven engine and emission modelling using non-linear models

5.1 Introduction

This chapter presents the results of a novel application of State Space Neural Networks (SSNN) applied to the experimental modelling of engine particulate matter emissions (PM) and engine performance expressed as in-cylinder mean effective-pressure (IMEP) of a 5.0L naturally aspirated Jaguar Land Rover engine. The on-line data collection took place at the University of Bradford Powertrain laboratory on transient dynamometer facilities using a Labview data-logging and control environment which is described in this work. The novel SSNN model described in the chapter consists of three smooth differentiable functions: tansig, purlin and logsig, which are shown to make the model structure suitable for prediction of the non-linear engine behaviour. The simulating capabilities are compared to NARX polynomial models that have been used in the past for dynamic calibration methodologies where models with good simulating capabilities are required. An heuristic algorithm for structure selection was applied to find the best model structure that minimises mean squared error of prediction error.

According to detailed analysis of state-space neural networks described in [93] such models need fewer arguments than input-output models and can perform at least as well as Extended Kalman-Predictors without requiring the knowledge of a model of the process dynamics.
The motivation is to use such models for dynamic calibration with an improved optimisation stage using a direct optimisation algorithm to significantly speed up the dynamic calibration process. As such an optimisation requires models in state-space form, so the SSNN is a natural choice for such an implementation and potentially the methodology could be extended in future to state-estimation to improve control performance.
Traditionally engine testing is undertaken whilst running steady-state. For dynamic calibration transient testing to obtain data for the identification of dynamic data-driven models of engine performance and emissions, the engine operating point must be changed dynamically according to prepared test sequences. These test sequences are typically generated as random signals in order to excite the transient engine characteristics. Other important features required for dynamic testing include fast data sampling and sensors with small time constants and small time delays. Usually the biggest problem when using steady state equipment for transient testing is that the emissions measurement equipment is designed for steady-state measurement where slow sensors are sufficient to capture required data. As a result direct identification using such slow sensors results in their slow responses being added to the engine dynamics. Nevertheless post-processing can be performed to compensate to some extent for such unwanted sensor dynamics using time-shifting, inverse models and non-causal filtering.

5.2 Experimental setup

5.2.1 Existing equipment at the University of Bradford

Testing was undertaken in the Powertrain Laboratory at the University of Bradford using an Elin Transient AC dynamometer controlled using AVL PUMA. To provide the excitation signal to PUMA, National Instruments hardware and software (LabView) were used. ETAS INCA software was used to control the

engine control unit (ECU). This system allows full access to all calibration and measurement variables.

The test bed was equipped with Horiba MEXA 9000 and AVL Condensation Particle Counter to measure gaseous emissions: NOx, CO_2, CO, THC and lambda (air-fuel equivalence ratio) and particulate count. For in-cylinder measurements an AVL INDICOM system is used. Six in-cylinder pressure sensors are installed and are connected to AVL INDISMART Gigabit hardware to process the data output. The output data is sent to INDICOM software installed on a PC. Several variables are measured and calculated by the INDICOM including, 50 deg burn-rate, IMEP, knock levels, Peak Pressure Position etc.
The engine is presented in Figure 5.1.

Figure 5.1: Engine in the test cell

One of the limitations on the test equipment that had a significant influence on the range of the investigation was the power of the dynamometer. This was rated

at a much lower power than the engine, peaking at 120 kW compared with 287 kW on the engine.

5.2.2 Synchronisation and data logging

Data was captured and saved in four different locations: INCA, Labview, Indicom, and PUMA. Data from both emissions analysers were collected in Labview software. The data communication was based on exchanging AT Commands from Labview to and from the emission equipment through serial communication RS232. Some of the temperature measurements were also recorded in Labview. The data from in-cylinder pressure sensors was collected and logged through the Indicom software that runs on the same PC. All measurements from the ECU were saved using the INCA software. All data from PUMA was saved on a separate PC. Apart from the manual trigger that gave information that a steady state condition was achieved, to distinguish between steady-state test points there was no other synchronisation signal in the original steady-state test bed set-up. Although this could entail very time-consuming post-processing this was still acceptable for steady state measurement applications. For the reported transient measurement application all signals were required to be synchronized, so the original test bed required modification to produce a new transient testing platform.

5.2.3 New Transient Testing Platform

For the current work the original test platform was modified in the form of a Transient Test Platform (TTP) to gain control of all of the components of the test equipment from one location and most importantly to synchronise and log the data in one location as in Figure 5.2. By controlling experimental processes from one location, not only every input signal is set at this same time instance, but most importantly, every output signal can be captured and recorded at this same time instance, which fulfils the requirements of synchronized data collection during the experimental transient testing. An additional benefit of this centralised control is that the control of the whole testing process can then become fully automatic. The operator needs only to define the calibration variables in the ECU that will

be altered and measured by the ECU and to specify the sequences of the control signals for the test before the experimental work is begun.

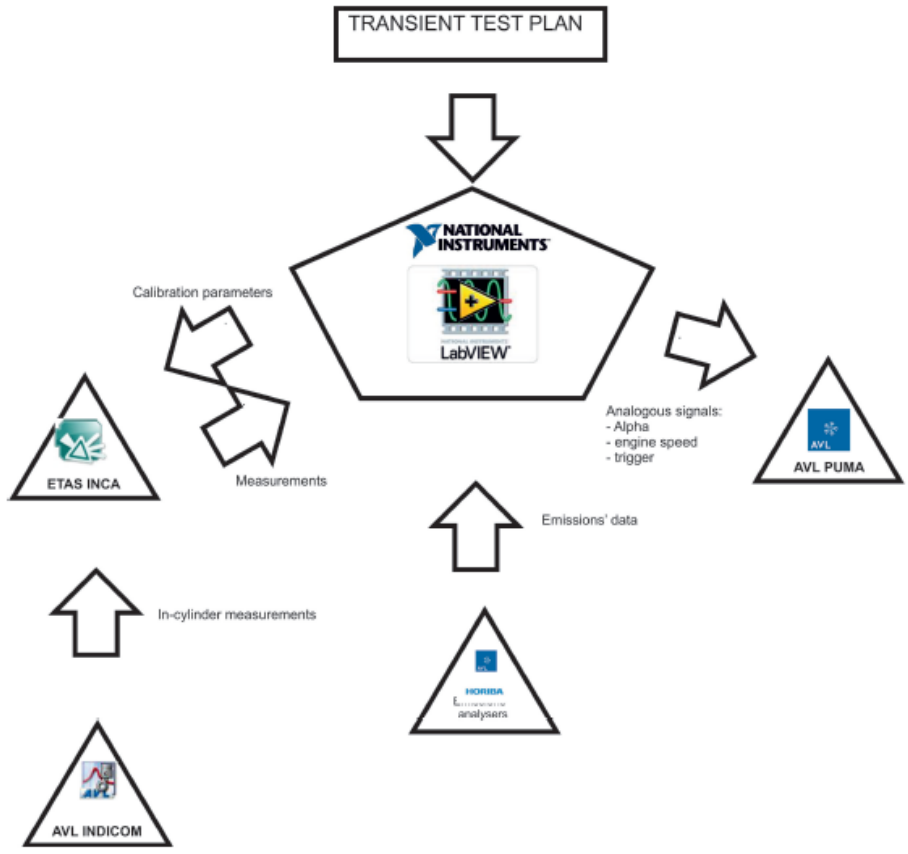

Figure 5.2: Block diagram of Transient Testing Platform (TTP)

The core of the revised experimental management system is custom software programmed in the National Instrument Labview graphical environment. The control platform was linked with the Puma system by using the additional hardware of the small sized low-cost NI6018 National Instrument module connected to the USB port to use analogue signals for setting the speed and torque operating points in PUMA and a digital signal trigger. The values of the demanded torque and speed inputs are scaled into a 0-5V range. The analogue outputs of the NI6018 module are connected with the analogue inputs from the PUMA cabinet. The PUMA software is configured to carry out the reading of the addi-

tional analogue signals and to scale the 0-5V analogue signal back to the range from 1000-2000RPM and $0 - 30\%$, for speed and load respectively. The PUMA was preconfigured to set the demanded values for operating points to post-scaled variables. Additionally to start the experiments the digital output from Labview was used to work as a trigger, which was set high when the start button was pressed and low when the experiment was terminated. This ensured that the synchronisation was maintained and is a safety feature in case of physical disconnection of the cable, since then the experiment will then be shut down.
The emissions equipment is controlled as in the original set-up from a separate Labview software system, however in the modified set-up measurement data from both emission analysers was redirected in the form of global software variables. The data in global variables is then available to other Labview software modules running on the same PC.
The most challenging modification to the original test-bed was the creation of an interface between INCA and the TTP. As both software systems are running on the same PC with a Microsoft Windows operating system, the communication between them was created based on an ActiveX - sometimes called Component Object Model (COM). This enables software components to communicate with each other utilizing Windows services. The communication between them was one-directional in terms of sending commands; the TTP works here as a master to INCA working as a slave. On the other hand, the working data flows in both directions, where the values of calibration variables are going from TTP to INCA, and the measurement data is going in the opposite direction together with the calibration parameters to be logged at this same time event.
The in-cylinder measured data is first processed in the INDICOM software to calculate numbers of combustion parameters and afterwards, the data is sent to the INCA software and then the data is sent to TTP using the ActiveX services. The knock information is used within the control strategy of the TTP, to decrease the chances of damaging the engine by detonation combination. The knock control strategy is simply a matter of halting the experiment whenever knock is detected, and then this test-point is removed from the time sequence and the experiment is run again.
The operator interface with TTP through front panel presented on Figure 5.3

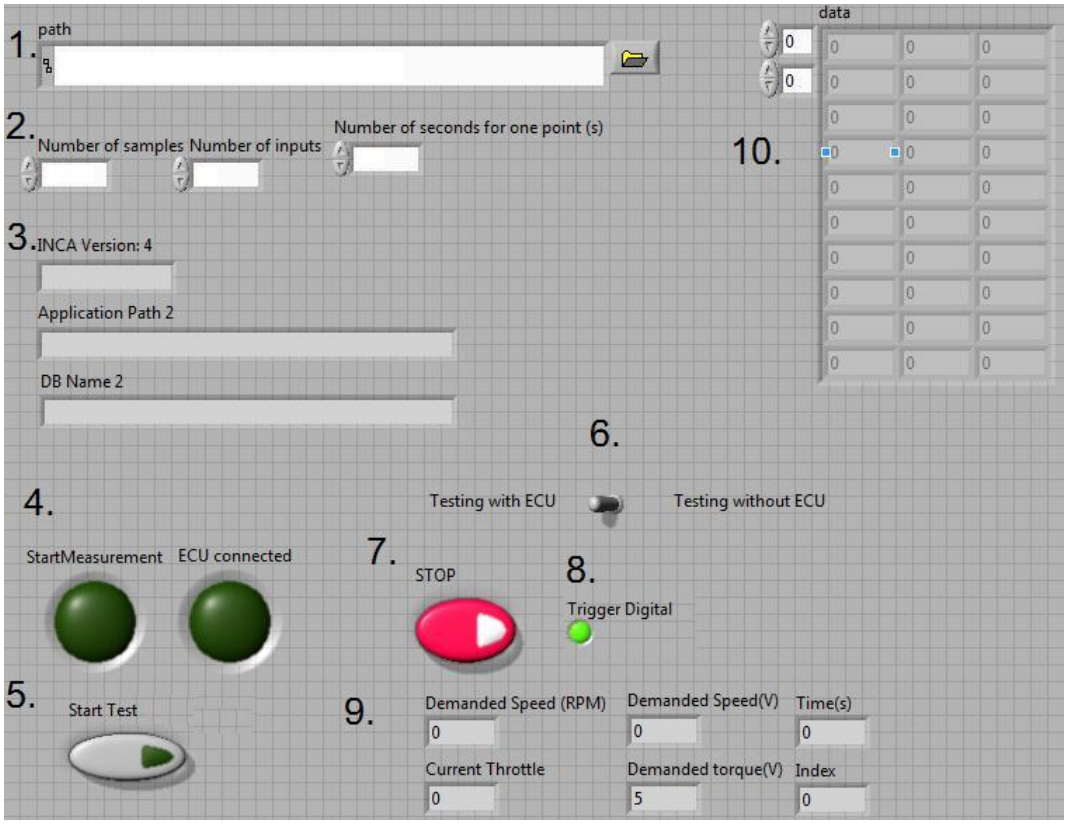

Figure 5.3: Front panel of Transient Testing Platform
1. Define location of excel sheet containing test plan
2. Configuration parameters of the test
3. Indicators of communication with INCA
4. Status LEDs
5. Start test button
6.Optional switch for testing with or without communication with ECU
7. Stop button to terminate testing - system going to the safe operation mode
8. LED to indicate status of trigger signal going to PUMA system
9. Current engine speed and torque demand
10. First 10 values from the test plan

5.3 Engine identification using State Space Neural Network

Since the current study to evaluate the capability of the State-Space Neural Networks for engine and emission modelling was preliminary, the model which was

investigated was restricted to the four inputs: u_1 is engine speed, u_2 is throttle position, u_3 is fuel mass per injection and u_4 is spark advance; and two outputs: IMEP and PN. The output y_1 of IMEP is chosen as a representative of an engine performance variable and y_2 is Particulate Number as representative of an emission variable. The results of SSNN are compared to those from both polynomial and neural NARX models.

5.3.1 Identification signals

APRBS signals were used as identification signals since both fast and slow dynamics are present in the engine and APRBS can excite a full range of frequencies and in contrast to a chrip signal, a range of amplitudes. The min-max values must be determined to design the identification signals. As stated before, for speed and load the range of 1000-2000RPM and $0 - 30\%$ is determined by the overall capability of the dynamometer. For spark advance the empirical limits from -5 deg BTDC to 40 deg BTDC are chosen. APRBS test sequences conforming to these limits were implemented for application in the TTP.

5.3.2 Polynomial model identification

The polynomial model structure for the comparative study was chosen as in [6] where dynamic calibration is developed on a virtual engine. The current work actually verifies the quality of this structure by using it to identify models based on data from the real engine. The second order polynomial model equation is thus taken as:

$$\begin{aligned} y(t+1) &= \theta_1 + \theta_2 u_2(t) + \theta_3 u_3(t) + \theta_4 u_4(t) + \theta_5 u_1(t)^2 + \theta_6 u_2(t)^2 \\ &\quad + \theta_7 u_3(t)^2 + \theta_8 u_4(t)^2 + \theta_9 u_1(t) u_2(t) + \theta_{10} u_1(t) u_3(t) \\ &\quad + \theta_{11} u_1(t) u_4(t) + \theta_{12} u_2(t) u_3(t) + \theta_{13} u_2(t) u_4(t) \\ &\quad + \theta_{14} u_3(t) u_4(t) + \theta_{15} y(t) + \theta_{16} \end{aligned} \tag{5.1}$$

where u_n is n-th input and θ_i is a linear parameter for the i-th term.
A minimised Least Squares fit method was used to obtain the parameters of the model from the engine experimental data. As the structure was predetermined no further structure selection was carried out. For simplicity the same structure was used for both IMEP Particulate Number.

5.3.3 Heuristic structure selection algorithm for neural networks

A neural network NARX model for the comparative study was chosen in the form:

$$\begin{aligned} y(t+n) &= \sum_{i=1}^{l} c_i \phi(w_{i_1} y(t) + ... + w_{i_n} y(t+n-1) \\ &\quad + w_{i_{n+1}} \boldsymbol{u}(t) + ... + w_{i_{2n+1}} \boldsymbol{u}(t+n-1)) \end{aligned} \tag{5.2}$$

where $\mathbf{u}$(t) is a vector of inputs at time t, and y(t) is a scalar output at time t , $\phi(.)$ is a smooth nonlinear function, l is the number of layers and c_i and w_i are synaptic weights. Such a general NARX structure does not guarantee a state space realisation because the non-linear function has every time step in its argument. Fig. 2 shows how a NN-NARX is represented in the MATLAB graphical NN format.

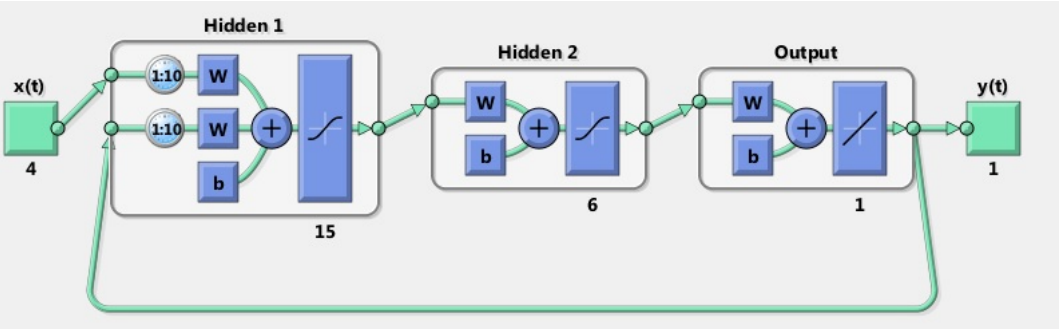

Figure 5.4: Example of NARX with 4 inputs 1 output neural network with 10 time steps for Inputs and outputs, two hidden layers 15 and 6

SSNNs are a subclass of generic NARX structure, but have the singular advantage

that their form guarantees realisation in state space form [4]. The general SSNN model is given by

$$y(t) = \sum_{i=1}^{n} C_i \phi(W_i \boldsymbol{z}(t-i)) \tag{5.3}$$

where $\boldsymbol{z}(t) = [y(t), \boldsymbol{u}(t)]$, C_i and W_i are $1 \times l$ and $l \times 2$ are dimensional matrices. Such a structure guarantees that the model is always realisable because it has all the different time instances separated. In this work, three smooth functions as on Figure 5.5 have been chosen: tansig, purlin and logsig.

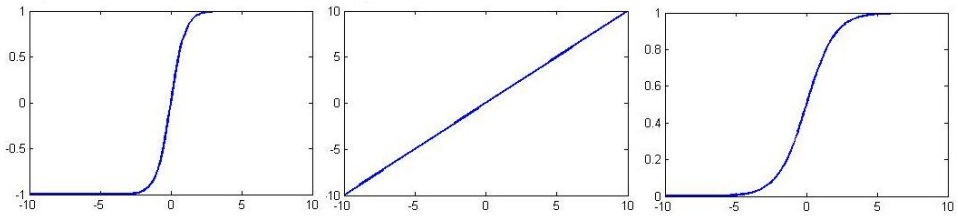

Figure 5.5: Smooth functions: tansig(left), purlin(center) and logsig(right)

The state space representation is then given by

$$\begin{aligned}
x_1(t+1) &= x_2 + C_1\phi(W_1[x_1(t), \boldsymbol{u}(t)]^T) \\
x_2(t+1) &= x_3 + C_2\phi(W_2[x_1(t), \boldsymbol{u}(t)]^T) \\
&\vdots \\
x_{n-1}(t+1) &= x_n + C_{n-1}\phi(W_n[x_1(t), \boldsymbol{u}(t)]^T) \\
x_n(t+1) &= C_n\phi(W_{n-1}[x_1(t), \boldsymbol{u}(t)]^T) \\
y(t) &= x_1(t)
\end{aligned} \tag{5.4}$$

Based on such a representation state-space oriented optimisation algorithms can be run such as: numerical dynamic programming, or general direct optimisation approaches.

The structure selection algorithm for both neural networks was achieved using

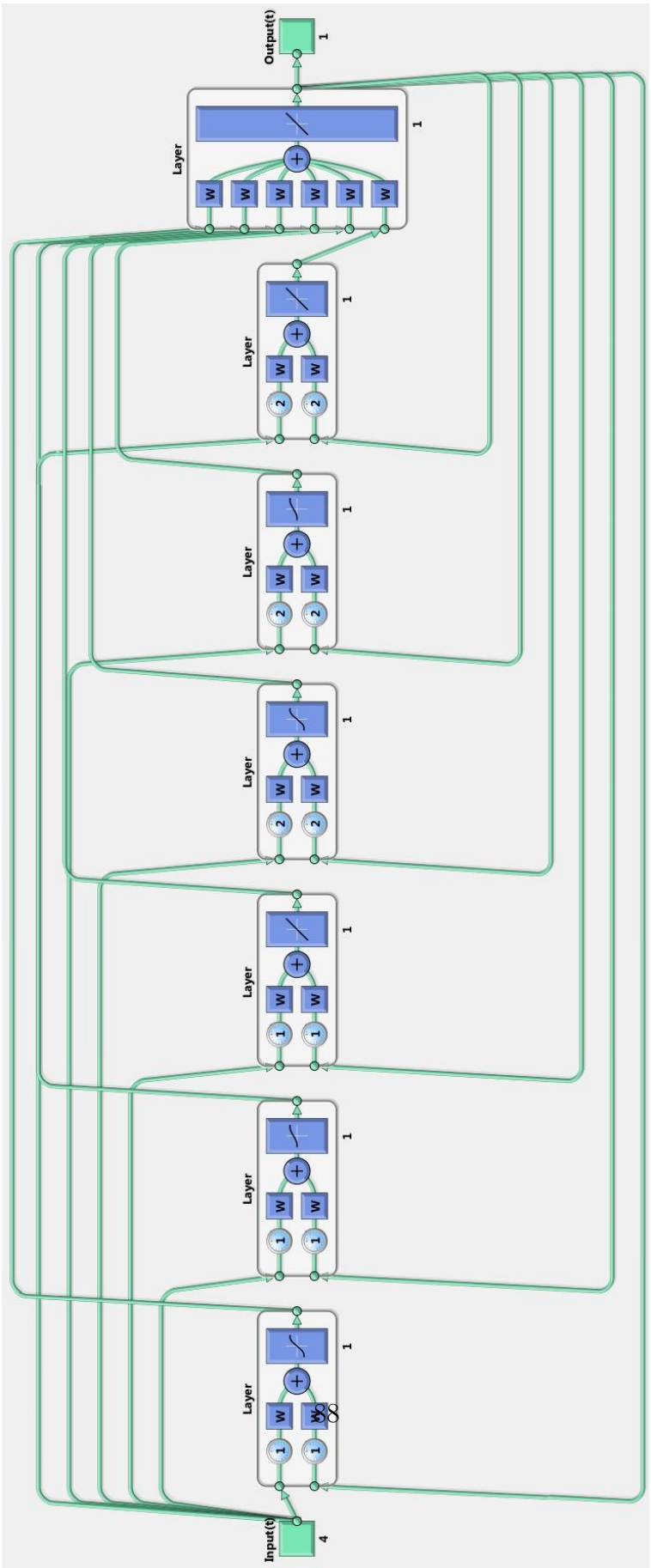

Figure 5.6: SSNN model in Matlab with 4 inputs, 1 output, two time steps, 3 smooth functions

an heuristic approach similar to that often used to solve typical issues with the training neural networks. The problems with such neural network modelling are overfitting and the sensitivity of the process to the initial starting point in the training algorithm [94]. To overcome those issues a number of different networks with a different number of neurons were trained for random numbers of epochs. For the NARX models the random parameters were:

- number of input time steps from 1 to 20
- number of output time steps from 1 to 20
- number of hidden layers 1 or 2
- number of neurons in hidden layer. If one hidden layer then 1 to 15, if two hidden layers then two random values were taken for the first and second hidden layer in the range 1 to 20 and 1 to 15 respectively, but the second not greater than the first hidden layer.
- number of training epochs from 1 to 300

The more runs of the algorithm then the higher the probability there is of finding a structure closer to optimal. Five-hundred different runs of the algorithm for each of the PN model and the IMEP model were accordingly taken where the Levenberg-Marquart [95] training algorithm was employed and the run with the best results chosen to represent the NARX neural network.

For the SSNN models a similar heuristic approach was adopted. As the general structure of the SSNN is less complicated (because all time instances are gathered), randomisation was restricted to only the following parameters:

- number of time steps from 2 to 5
- number of training epochs from 20 to 200

Again 500 runs of each model were investigated. It is important to remember that any two runs with these parameters with the same values are very likely to have different performance due to different random initial weights.

5.4 Results

The modelling task was taken to be that of obtaining a model with the best prediction capability for later use in the dynamic calibration application. Accordingly the performance of all models was tested based on criteria to measure prediction capability for unseen data. The simplest way of measuring prediction accuracy is to use Normalised Root Mean Square Error (NRMSE) typically called *best fit* and given by

$$NRMSE = [1 - \sqrt{\frac{\|x - x_{ref}\|^2}{\|x_{ref} - mean(x_{ref})\|^2}}]100\% \tag{5.5}$$

where x is the predicted output from a model and x_{ref} is the measured output. The NRMSE fit costs vary between -INF (bad fit) to 100% (perfect fit).
A total of 12000 samples were collected with a 10Hz sampling rate. The data was then divided into two parts of length 9000 and 3000. The first part was used for training purposes, the second part for verification.

The results obtained for the IMEP modelling are presented in Table 5.1 and show the capability of the models for unseen data. The Number of Parameters column gives how many parameters must be given to define such model. A higher number of parameters leads to a high computational effort to compute model output, so that a model with a lower number is preferred for offline optimisation computation.
The results show that for IMEP prediction the best results from out of the three model types are for the NN-NARX structure. Nevertheless the performance of SSNN is close to it. The polynomial model results are the worst but are still above 70% fit which is a very good result and this could be seen in Figure 5.7
The SSNN model shows best prediction capabilities for Particulate Number. The difference in the number of parameters between NN-NARX and SSNN is significant; more than 1000 parameters are needed for the NN-NARX to achieve comparable results. From the graph on Figure 5.8 and Table 5.2 it can be noticed that the polynomial model with simple structure is better for IMEP modelling

Table 5.1: Prediction comparison for IMEP modelling

Model type	**NRMSE**	**Number of parameters**	**Model structure**
Polynomial	70,96%	16	two time delays
NN-NARX	88,03%	647	9 inputs time delays; 17 outputs time delays; two hidden layer with 11 and 4 neurons and trained over 23 epochs
SSNN	85,97%	90	5 time delays trained over 113 epochs

Table 5.2: Prediction comparison for PN modelling

Model type	**NRMSE**	**Number of parameters**	**Model structure**
Polynomial	-7,29%	16	2 time delays
NN-NARX	22,41%	1298	11 inputs time delays; 19 outputs time delays; two hidden layer with 17 and 11 neurons and trained over 41 epochs;
SSNN	25,25%	90	5 time delays and trained over 147 epochs;

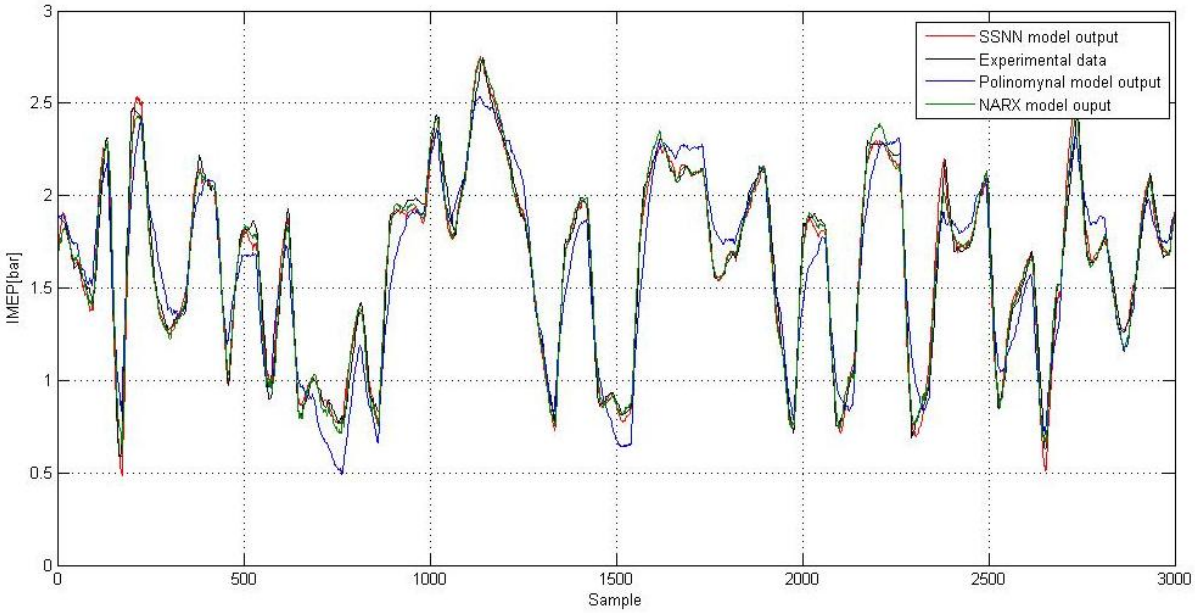

Figure 5.7: IMEP modelling verification

than for emission modelling.

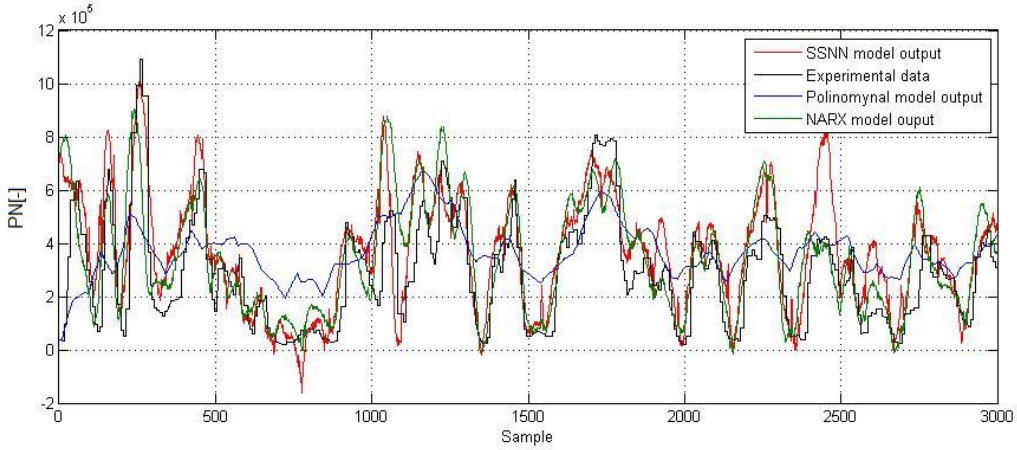

Figure 5.8: PN modelling verification results

5.5 Conclusion

The main contributions of this work are the presentation of the successful implementation of a low-cost Transient Testing Platform and the subsequent use of this platform for data transient data collection used in a comparative study of different model types for transient engine performance and emissions. Novel SSNN models for engine and emission modelling are found to have similar predictive capabilities to NN-NARX despite a significantly simplified structure requiring significantly fewer parameters. Both SSNN and NN-ARX models are shown to have superior simulation capability to polynomial models. The SSNN structure may be sufficiently simple to allow on-line implementation in forthcoming ECU modules.

Chapter 6

Experiment Based Dynamic Calibration of the Diesel Air-path

6.1 Introduction

This chapter presents the results of an experimental study on the application of a novel engine dynamic calibration methodology applied to control of the air-path of a Jaguar Land Rover (JLR) turbocharged diesel engine. A complete application of the methodology is demonstrated where final feedforward dynamic controller is implemented in a test vehicle after being obtained in a one-shot process using data solely obtained from a novel dynamic dynamometer testing process, without recourse to in-vehicle tuning. Although the dynamic calibration methodology can potentially be implemented for complete engine calibration including for emission constrained optimising of fuel consumption, the work of this thesis focused on control of only boost pressure and Exhaust Gas Recirculation (EGR) rate, with constraints on only NOx emissions and opacity without fuel consumption optimisation.

The approach combines State Space Neural Network (SSNN) modelling and causal dynamic optimisation to determine a feedforward Hammerstein-Wiener control map by an inverse identification from the synthesised optimal control behaviour. In the study, data collection for dynamic engine modelling took place on a state-of-the-art transient engine dynamometer facility equipped with emissions

analysers. The SSNN for the air-path dynamics was then established. A dynamically rich, optimal control behaviour for this model was thereby synthesised and the feedforward map accordingly identified. Verification of the controller performance by vehicle testing was then carried out at the Jaguar Land Rover Test Track, Gaydon UK, by implementing the controller map in-vehicle. The outcome showed that a well performing control calibration could be systematically obtained from limited dynamometer testing time.

6.2 Experimental setup

Data collection for system identification took place at the University of Birmingham where special reconfiguration of existing equipment was necessary to carry out the unique transient testing sequences. The equipment set-up included:

- AVL transient dynamometer with PUMA for test - cell control
- INCA for engine calibration and communication with ECU
- AVL emission analysers (AVL439 opacity meter and AVL i60)
- AVL INDICOM for in-cylinder measurements

To facilitate communication between PUMA and INCA an ASAM3 protocol was used for reading measurement variables and only software modification was required. The concept was to let PUMA change calibration variables in the ECU by sending commands to INCA over the ASAM3 protocol. This approach eliminates a need for any external hardware to obtain transient changes of calibration variables in ECU. Control over those parameters was by change of speed/load based set point maps to maintain consistency between measurement points. So at each point of a transient test the whole content of such a map is set to a single value according to the test schedule. PUMA was also programmed to collect all data from all emission analysers, in-cylinder sensors and ECU measurement variables with 10 Hz rate. This ensured that all the data collection was synchronised.

6.3 Dynamic calibration

6.3.1 System identification stage

System identification was carried out to obtain a mathematical description of engine emissions behaviour. Emission generation is a multi-input process that is dynamical [13], so models that have capabilities to describe system dynamics were chosen for the study. State Space Neural Networks (SSNNs) were employed to describe non-linear and dynamic behaviour of engine emissions, and also this model structure enables the use of state-of-art optimisation routines to speed up the optimisation process and has the potential for future exploitation of state-estimation methodologies in controller implementation.

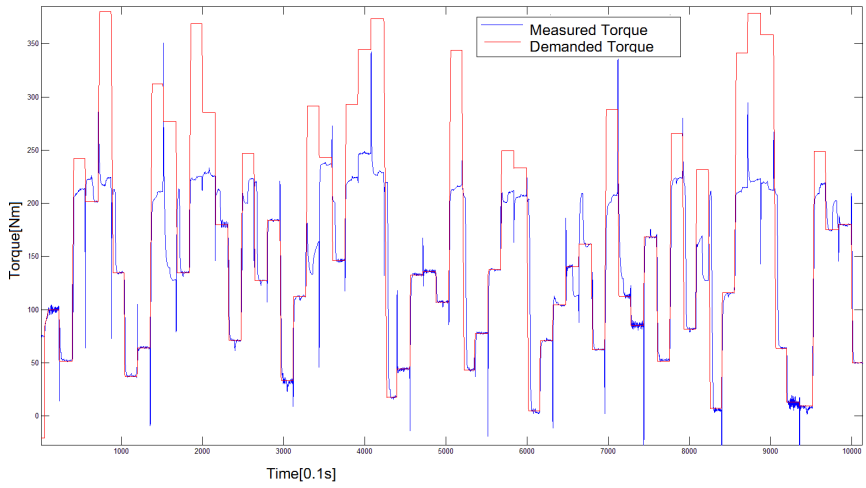

Figure 6.1: Torque response on the test bed when testing with EGR valve position and VGT valve position

Initially the experimental testing was set out to investigate development of low-level dynamic control of EGR valve position and VGT valve position as in chapter 3. Therefore the following set of inputs with corresponding ranges was utilized in the input signal design for identification experiments:

- Engine Speed: 850-2500 RPM
- Torque : 0-400 Nm

- VGT valve position: 0-100 %
- EGR valve position: 0-100 %

It turns out that this leads to a situation when demanded torque during the experiment could not be generated as shown in Figure 6.1. To overcome this problem better input signal design was required to be investigated including some form of boundary search. Therefore the scope of this work was focused on the development of a dynamic controller for higher-level inputs for Boost Pressure and EGR rate. With such inputs set the demanded torque during the data collection process met the demand as in Figure 6.2.

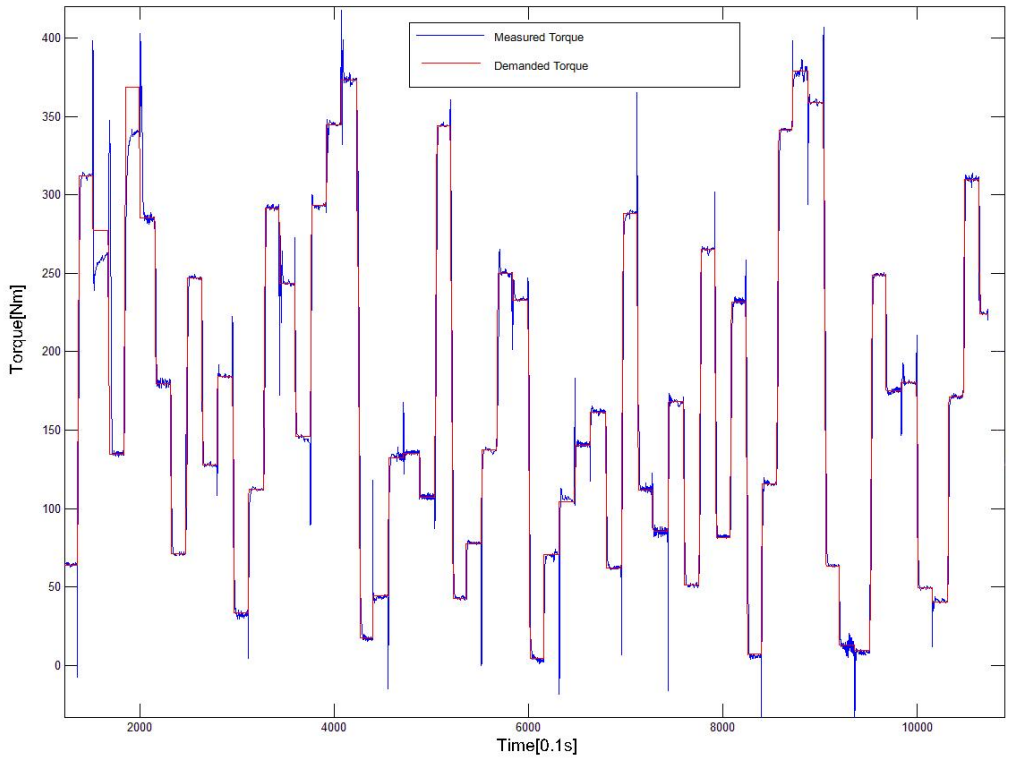

Figure 6.2: Torque response on the test bed when testing with Boost pressure setpoint and EGR rate setpoint as inputs

The input-output model structure of the identified system is shown as map G

in Figure 6.3. The inputs are u_1 injected fuel mass, u_2 engine speed, u_3 boost pressure set-point and u_4 EGR rate set-point. The outputs were y_1 NOx, and y_2 opacity. The engine speed was regarded as independent of injected fuel mass (and hence of brake torque Tb), because the engine speed torque relation was set by control of the engine dynamometer loading L as shown in Figure 6.3.

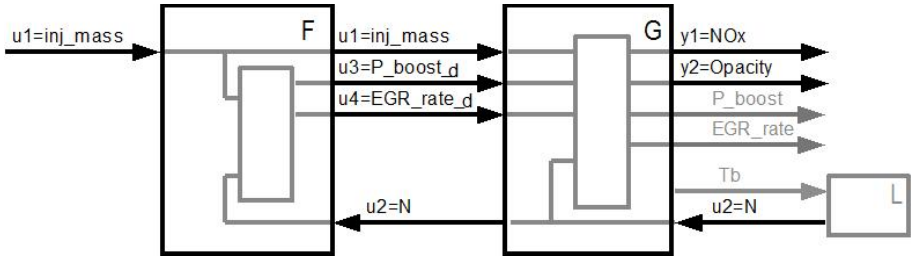

Figure 6.3: Feedforward controller F and identified system G

The measured boost pressure from pressure sensor and EGR rate output estimate from the ECU were available in the experimental system. APRBS signals were used as the input test signals for system identification to achieve rich excitation of the engine dynamics. A signal with a period of 4s was empirically chosen however optimisation of the choice of this value gives scope for future improvement. Physical limits were used in the design of the multi-input identification to make sure that the experiment was safe and a rich description of the model's behaviour for all channels is obtained. Ranges for each individual channel were set for both safe operation and to ensure that boost pressure and EGR set points were achievable and were accordingly chosen as:

- engine speed 800-2500RPM
- torque 0-400 Nm
- boost pressure limits based on value P_{cal} from a standard calibration map as follows:
  - min=max(1.050bar, P_{cal}-0.5 bar)
  - max=min(3bar, P_{cal}+0.5bar)
- EGR rate limits based on the EGR_{cal} value from a standard calibration map as follows:

- min=0
- max=min(55, EGR_{cal}+10%)

Input signal synthesis used four random signals generators, one for each input. Because the Min and Max maps mentioned above are engine speed-load dependent, random values of speed and torque are first generated. Later the two values from the range 0-1 from the other two random generators are produced and are then pre-scaled into min-max maps according to:

$$output = (max - min) \cdot (randomvalue) + min \tag{6.1}$$

As reported in [45] system identification is difficult when the system has different input sensitivities and time-scales for each input. For engine identification engine speed and torque are the inputs with the most dominant correlation with the outputs. Accordingly the input test sequence was constructed as:

- the first test point contains 4 random values
- the next test point holds speed, torque and boost pressure at the same level and changes only EGR rate
- the next test point sets all 4 values again
- the test point after this holds engine speed, torque and EGR rate at the same level as for the previous test point, but changes only boost pressure

  The signal generation algorithm was implemented in Simulink and is shown in Figure 6.4.

6.3.2 Time lags removal

As observed in [96] sensor dynamics become part of the system dynamics of the identified system. In our case the dynamics of the opacity meter are too fast to be an issue. On the other hand the NOx analyser has a significant transport lag due to its location which is 10 m away from the engine. A time

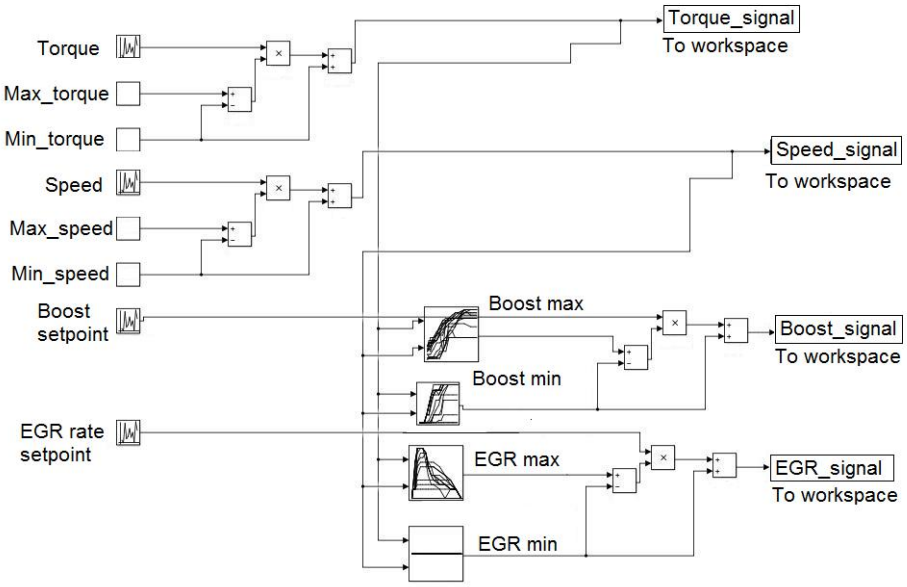

Figure 6.4: Generator of input signals for identification

lag of 3.8s was observed from step changes so this was accordingly removed from the data record of the NOx signal by shifting the signal backwards 38 samples (10 Hz sampling rate). A possible future approach would be to use exhaust mounted NOx and PM (particulate matter) sensors as required for EU6 and OBD application to allow improved removal of analyser delay.

6.3.3 State Space Neural Network models

The recurrent State Space Neural Network is a subclass of generic NARX structure, but it has the significant advantage that it guarantees state space realisability [97]. The input-output versions the SSNN models for each output channel are of the SISO NN form:

$$y(t) = \sum_{i=1}^{n} C_i \phi(W_i \boldsymbol{z}(t-1)) \tag{6.2}$$

where $\boldsymbol{z}(t) = [y(t), \boldsymbol{u}(t)^T]^T, \boldsymbol{u}(t) = [u_1(t), u_2(t), u_3(t), u_4(t)]^T$ is the vector of inputs and y(t) is one of the two scalar outputs $y_1(t)$ or $y_2(t)$, each at time t , l is the number of hidden neurons, ϕ_j is an $l \times 1$ vector function of $j = 1, ...l$ smooth non-linear scalar functions ϕ_j, n is the model order, and there are $i = 1, .., n-1$ matrices C_i and W_i of dimension $1 \times l$ and $l \times 5$ containing synaptic weights. Such a structure guarantees that the model is always realizable in the classical state space form because it has all the different time instances separated. In this work three smooth functions ($l = 1, 2, 3$) have been chosen: tansig, purelin and logsig. The state space representation for each y channel is then of the form:

$$
\begin{aligned}
x_1(t+1) &= x_2 + \sum_{j=1}^{3} C_{1,j}\phi_j(W_{j,1}[x_1(t), \boldsymbol{u}(t)]^T) \\
x_2(t+1) &= x_3 + \sum_{j=1}^{3} C_{1,j}\phi_j(W_{j,1}[x_1(t), \boldsymbol{u}(t)]^T) \\
&\vdots \\
x_{n-1}(t+1) &= x_n + \sum_{j=1}^{3} C_{1,j}\phi_j(W_{j,1}[x_1(t), \boldsymbol{u}(t)]^T) \\
x_n(t+1) &= C_n\phi(W_{n-1}[x_1(t), \boldsymbol{u}(t)]^T) \\
y(t) &= x_1(t)
\end{aligned}
\tag{6.3}
$$

6.3.4 Structure selection

The structure selection algorithm used for the SSNN models was based on an heuristic approach used in similar training applications of neural networks. The problems in the training here are overfitting and the high sensitivity of the final results to the initial starting point. To overcome these issues a number of different networks with different numbers of neurons were trained for different random numbers of epochs. As the general structure of SSNN is not very complicated because the same time instances

Table 6.1: Validation results of emissions models

Model	**Number of states**	**Number of epoches**	**NRMSE (best fit)**
Opacity	5	115	50,2%
NOx	5	125	81,4%

are grouped, only the following parameters needed to be chosen randomly in each iteration of the algorithm:

- number of time steps from 2 to 5
- number of training epochs from 20 to 200

6.3.5 Results of the identification

Data collected from the test bed was divided into two lengths: 70% of the data was used for training of neural networks and 30% for validation. The simulation capability was measured by the normalised root mean square error (NRMSE), ie best fit parameter as in equation 2.37. The results of the identification are presented in Table 6.1.

It can be noticed that the NOx model has better simulation capability than the opacity model. It is believed to be due to the existence of a high number of spikes in the opacity measured signal that can not be entirely described by such small set of inputs. Consideration of a larger or different set of inputs could potentially improve it. Currently it is necessary to manual tune out those spikes once installed in the vehicle which is an issue that must be addressed in future to fully exploit the potential of the approach.

6.4 Optimisation stage

The second stage of the dynamic calibration is an optimisation where optimal trajectories of inputs are determined with regard to a quadratic plus

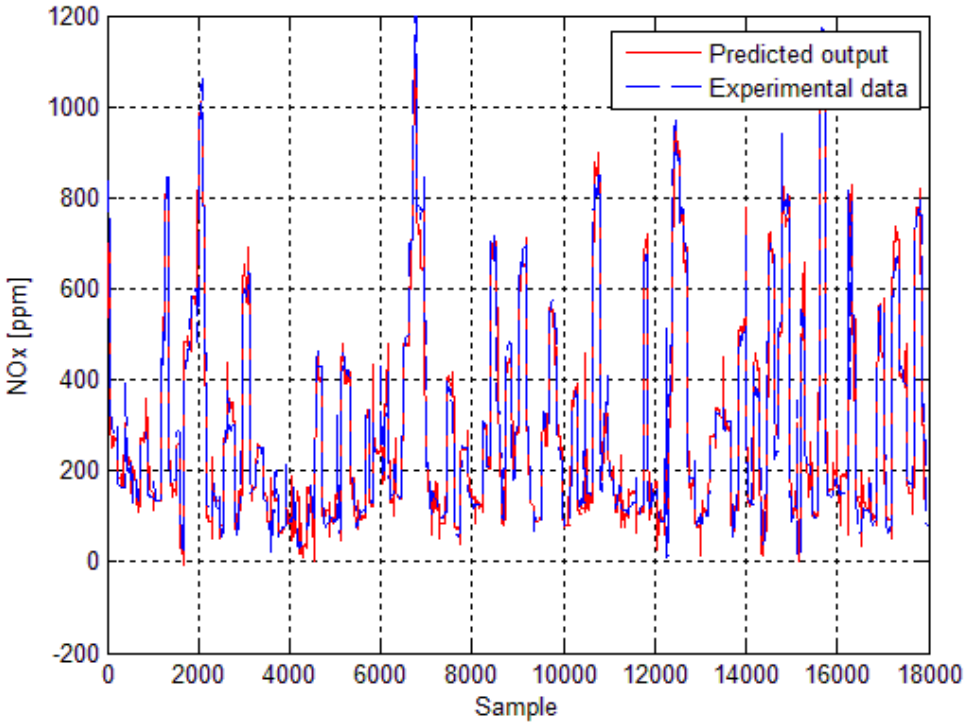

Figure 6.5: Comparison of NOx measured and simulated data

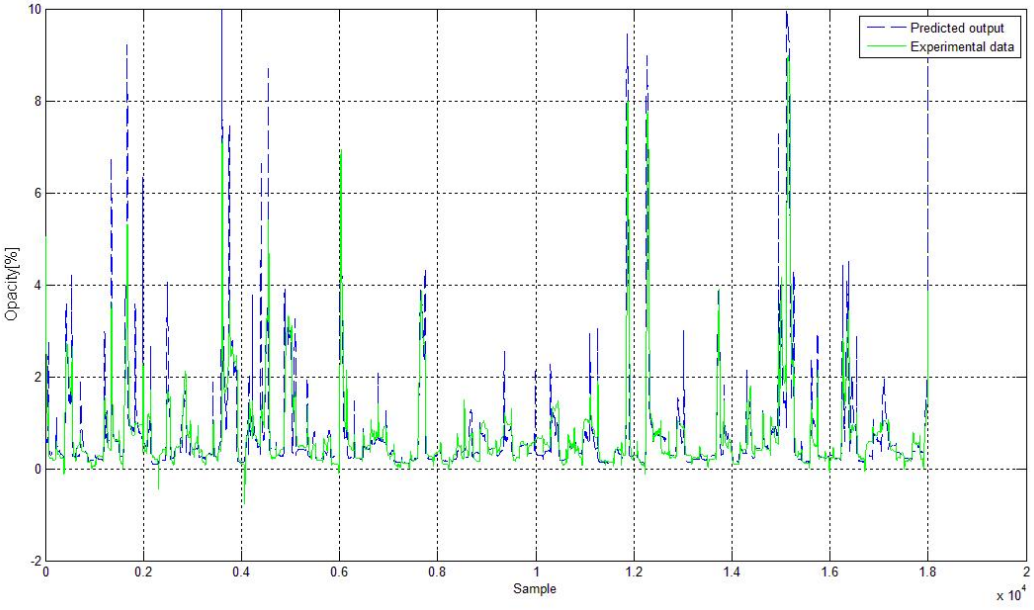

Figure 6.6: Comparison of Opacity measured and simulated data

linear cost function while satisfying all constraints. By modelling both the engine and emissions by SSNN models the state-space structure means that the corresponding optimisation is equivalent to a large scale but highly sparse non-linear (quadratic) programming problem (as formulated in the direct collocation method optimal continuous system control [80]). The resulting cost function is then of the form:

$$\min_{[\mathbf{u},\mathbf{x}]} T(x_N) + \sum_{i=1}^{N-1} \mathbf{x}(i)^T \mathbf{Q}(i)\mathbf{x}(i) + \overline{\mathbf{u}}(i)^T \mathbf{R}(i)\overline{\mathbf{u}}(i) + \mathbf{x}(i)^T \mathbf{S}(i)\overline{\mathbf{u}}(i) + \mathbf{L}(i)\mathbf{x}(i) + \mathbf{M}(i)\overline{\mathbf{u}}(i) \tag{6.4}$$

$$\begin{aligned} &\text{subject to} \\ \mathbf{x}(i+1) - f(\mathbf{x}(i), \mathbf{u}(i)) &= 0; \quad i = 1, ..., N-1 \\ \mathbf{x}(1) - \mathbf{x}_1 &= 0 \\ c(\mathbf{x}(i), \mathbf{u}(i)) &>= 0; \quad i = 1, ..., N-1 \end{aligned} \tag{6.5}$$

where N is the simulation time and $\mathbf{Q}$,$\mathbf{R}$,$\mathbf{S}$,$\mathbf{L}$ and $\mathbf{M}$ are corresponding weighting matrices.

In this scheme the optimisation is able to use knowledge of the state trajectory in the initializations which are required at each iteration, so as to show fast local convergence and can also treat unstable and non-minimum-phase systems [4]. As with Direct Collocation optimisation this optimisation can thus efficiently use Sequential Quadratic Nonlinear Programming solvers such SPRNLP, SNOPT and KINTRO. For this study the SNOPT algorithm was used which is based on a sequential quadratic programming routine run in the Matlab environment.

6.4.1 Equality constrains

A SSNN model with 5 states obtained in the previous stage was used to describe the system dynamics. An additional state was added to the opacity

model to decrease slightly the speed of response of this model because in initial optimisation runs, a set of too fast changing trajectories had been obtained. The optimisation was run for N=6000 sample points (10 min), The overall number of equality constraints was thus 6000x11 plus additional 11 for starting point.

6.4.2 Non-equality constrains

Since inputs and outputs of neural network have been normalized the inequality constraints were set to keep all inputs in the range [-1,1] which helps the optimisation speed up. Additional constraints were added for fuel injection mass and engine speed inputs so they followed a fixed trajectory during optimisation. To keep this calibration process independent from any driving cycle, randomised profiles of fuel injection and engine speed from real engine were used. Of course if needed driving cycle profiles from any given driving cycles could be used here. It may seem that it would be more natural to use equality constraints for fuel injection and engine speed trajectories constraints, but the SNOPT solver was found to be more efficient with these as inequality constraints.

6.4.3 Cost function

The objective of the present optimal control problem is to minimize outputs of NOx and particulate emissions as measured by opacity. In the work of Benz [19] the weights of NOx and PM in the cost function in the optimal control problem were chosen in proportion to their ratio in the EURO VI regulation. In this work, however because measured opacity is used as a substitute for PM this approach is not possible, so the weights of the cost function were obtained by an iterative process involving incrementing one weight and observing changes in the optimal outputs trajectories until satisfactory trajectories were obtained. The cost function additionally has weighted contributions from deviations in input to reduce sudden changes

of control inputs to give good driveability and to extend the life of the actuators. The cost function used was accordingly:

$$I = \sum_{i=1}^{3000} (w_{NOx} y_1(i) + w_{opacity} y_2(i) + w_{u3}(u_3(i) - u_3(i-1)) + w_{u4}(u_4(i) - u_4(i-1))) \tag{6.6}$$

where y_1 is the output from the NOx emissions model, w_{NOx} is the weight for NOx, y_2 is the output from the emission opacity model, $w_{opacity}$ is the weight for the opacity, u_3 is the Boost Pressure input, w_{u3} is the weight for the rate change of input u_3, u_4 is EGR input change rate, w_{u4} is the weight for the rate change of input u_4. The optimisation was run 5 times with different sets of weights as follows:

1. $w_{NOx} = 2$, $w_{opacity} = 1$, $w_{u3} = 0.25$, $w_{u4} = 0.25$.
2. $w_{NOx} = 1.5$, $w_{opacity} = 1$, $w_{u3} = 0.25$, $w_{u4} = 0.25$.
3. $w_{NOx} = 1$, $w_{opacity} = 1$, $w_{u3} = 0.2$, $w_{u4} = 0.2$.
4. $w_{NOx} = 0.75$, $w_{opacity} = 1$, $w_{u3} = 0.25$, $w_{u4} = 0.25$.
5. $w_{NOx} = 0.5$, $w_{opacity} = 1$, $w_{u3} = 0.1$, $w_{u4} = 0.1$.

Since in each engine for each operating point there is a minimum air flow mass to generate any given torque, such a relation can be used to implement a torque tracking component as well in an extended cost function in the future work.

6.4.4 Optimisation results

Each optimisation took about 4000s (1h 10min). Figure 6.7 shows the resulting NOx output from the emission model and Figure 6.8 shows the resulting output from the Opacity model. It is clear that with decreasing w_{NOx} more NOx is generated and simultaneously less opacity, which effect

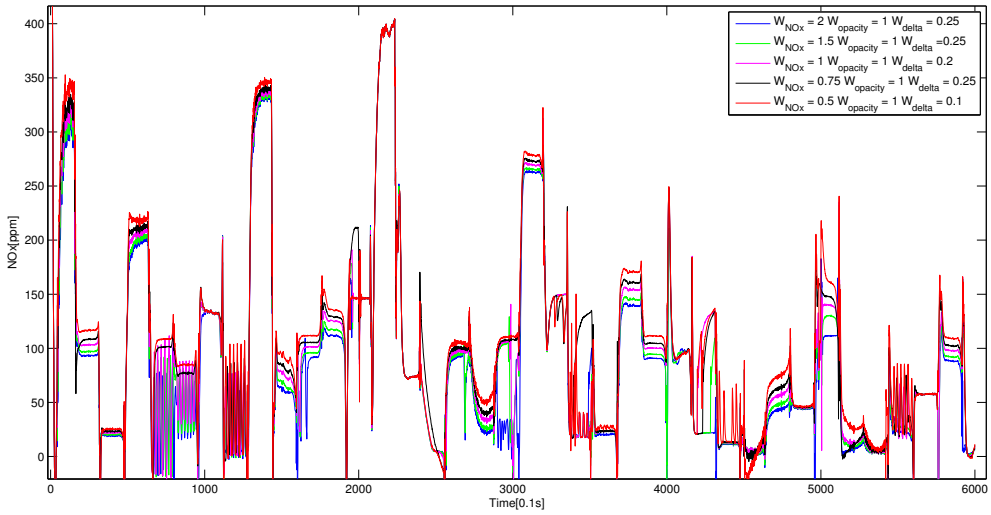

Figure 6.7: Comparison of NOx simulated for different optimisations

is presented in Figure 6.9 in terms of average values of NOx and Opacity. For the next stage the data resulting in the lowest opacity was chosen.

6.5 Inverse identification stage

The final stage of the dynamic calibration methodology is an inverse identification where final feedforward dynamic controllers are obtained. Inverse identification is the process of swapping some input sequences with some output sequences prior to identification to obtain either a partial or full inverse of the system so that the resulting identified model can then act as a feedforward control map [76]. The inverse identified approach was originally developed for non-optimal Neural Networks controls [98] [99], then subsequently extended to general non-optimal NARMAX controls [100] [101] and more recently to optimal feedforward controls [102].
In the present study, to obtain the dynamic optimal feedforward controllers for the inverse identification, the engine speed and fuel injection mass signals were retained as inputs while the corresponding optimal input trajectories for boost pressure and EGR rate determined in the previous stage were

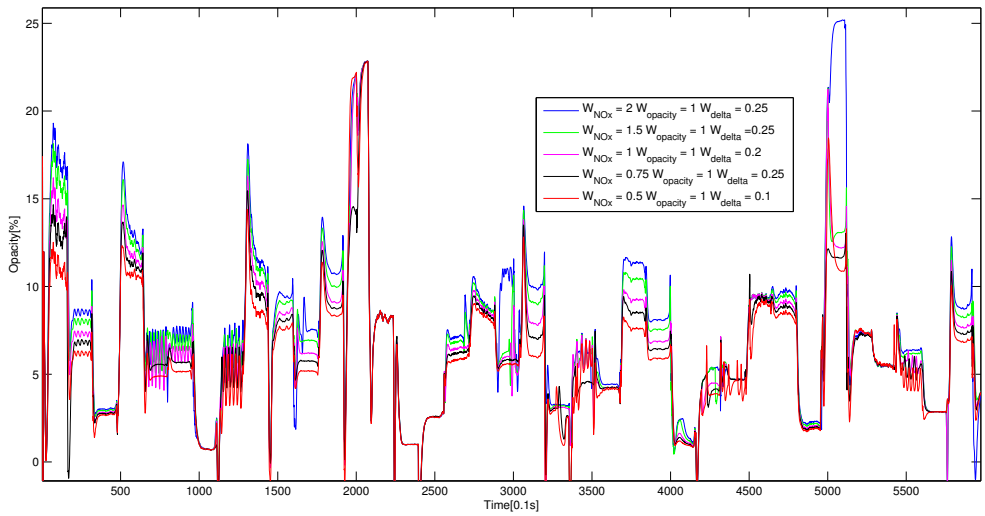

Figure 6.8: Comparison of NOx simulated for different optimisations

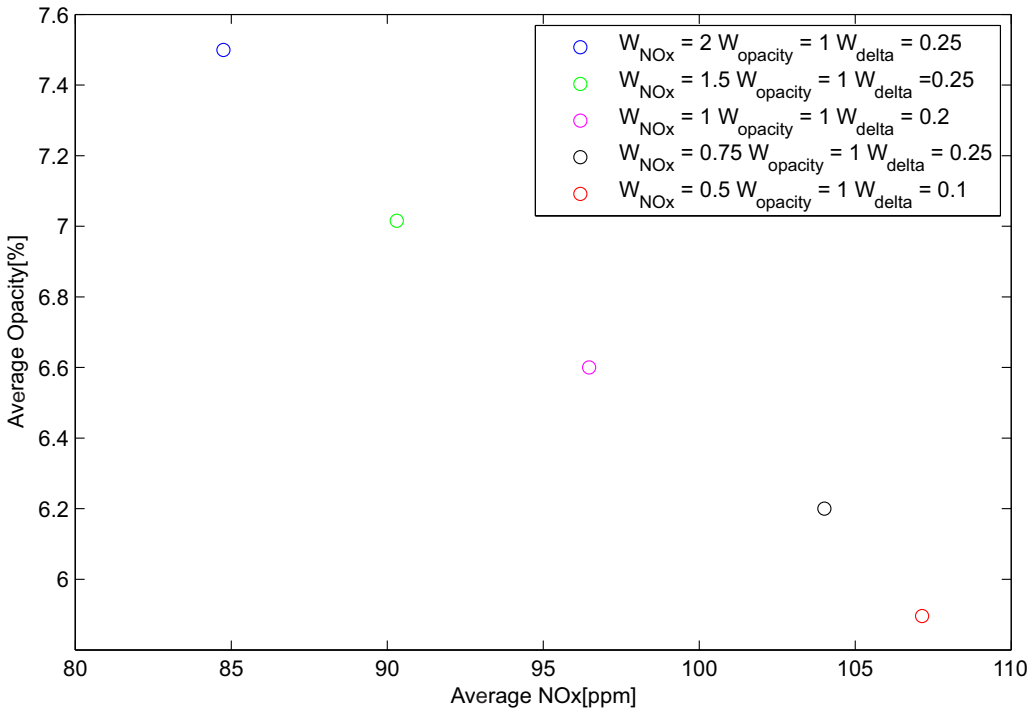

Figure 6.9: Average NOx and average Opacity

Table 6.2: Structure of Hamerstein-Wiener controllers

Hamerstein-Wiener structure	**Parameter**
Non-linearity of input 1	10 sigmoid
Non-linearity of input 1	10 sigmoid
Linear block	1 zero, 2 poles, 1 delay
Output non-linearity	10 piecewise

Table 6.3: Results from inverse identification

Model	**NRSE**
Boost pressure	74,4%
EGR rate	57,1%

treated as outputs. The input-output structure of the inverse identified controller is shown as the map F in Figure 6.3. A Hammerstein-Wiener structure was used for the two feedforward dynamic controllers because its special construction ensures good non-linearity estimation and it contains linear dynamics part which make it easy to implement on hardware as it can be converted to a few lines of code. For both controllers this same structure was used as in table 6.2.

The inverse identification results are justified using the best fit measure as used in the conventional identification of stage one. Results of the fit are presented in Table 6.3 and in Figure 6.10 and Figure 6.11.

The results shows that the Hammerstain-Wiener gives better simulation capability for Boost Pressure prediction than for EGR rate.

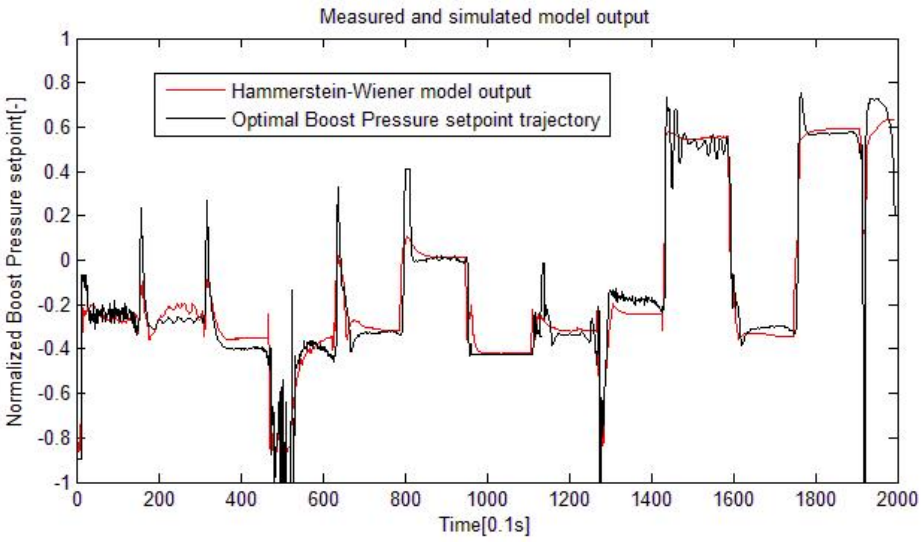

Figure 6.10: Fit of boost pressure by inverse identified controller

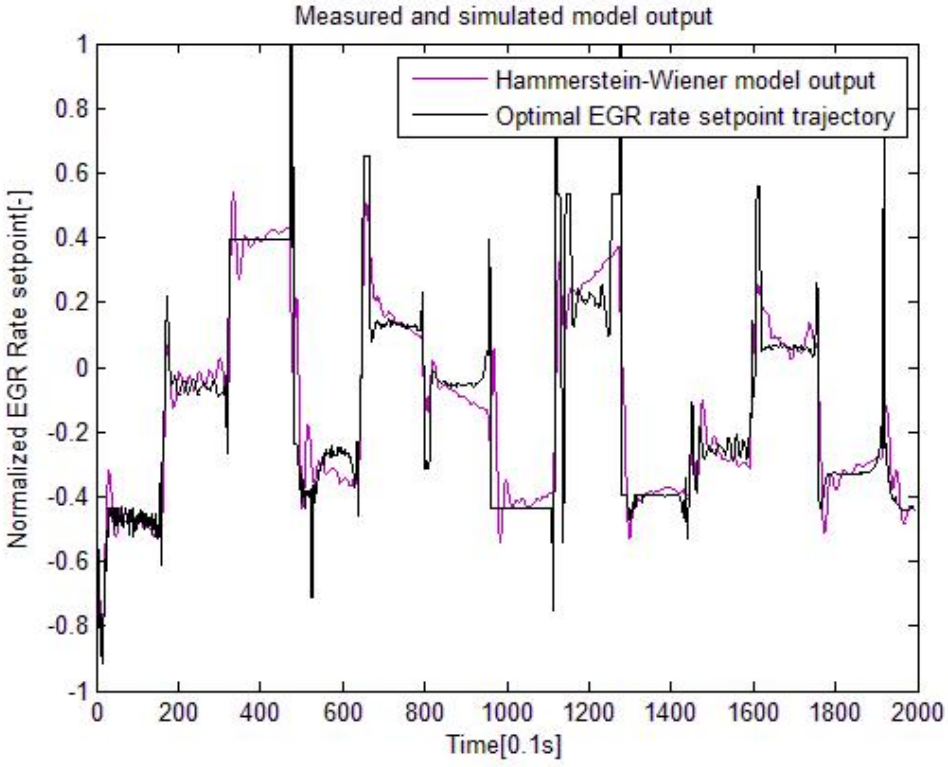

Figure 6.11: Fit of EGR rate by inverse identified controller

6.6 Validation

The dynamic controllers were validated on the Jaguar Land Rover Gaydon Test Track in a vehicle equipped with a Jaguar Land Rover turbocharged diesel engine. The controllers were implemented in the ECU target using ETAS EHOOKS [103] bypassing software. The technique allows for the implementation of controllers represented as Simulink models. Since no emission measuring equipment was available at the time of the test, no quantitative direct experimental validation of the emissions regulation was possible. Figures 6.12 and 6.13 show the output from both the optimised controller and from the standard ECU. The optimised controller output favoured higher EGR rates with lower boost pressure.

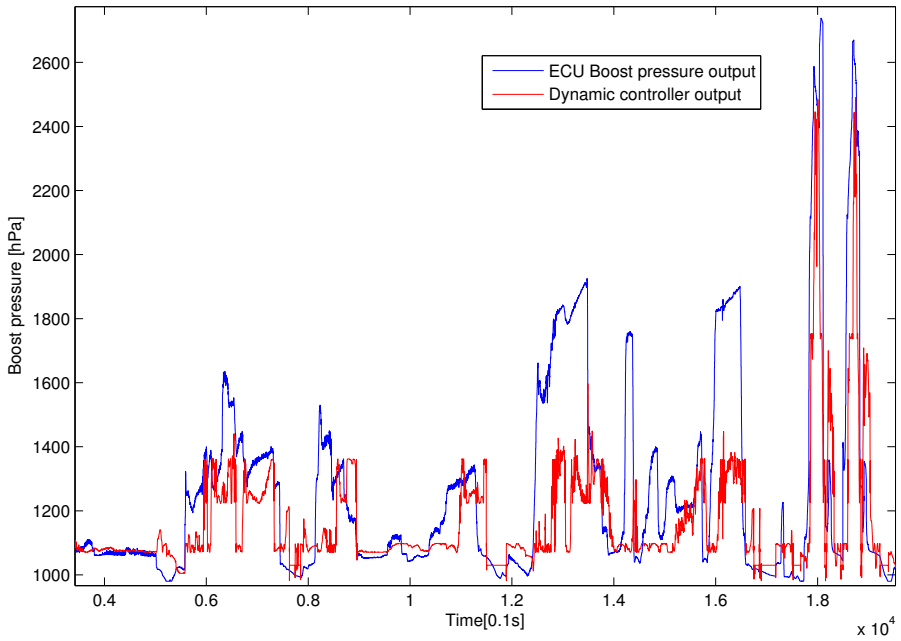

Figure 6.12: Comparison of Dynamic and ECU controllers for boost pressure

Figure 6.14 shows boost pressure set-points output from both dynamic optimised controller and from standard ECU and the measured boost pressure resulting from the dynamic controller. Figure 6.15 shows EGR rate set-

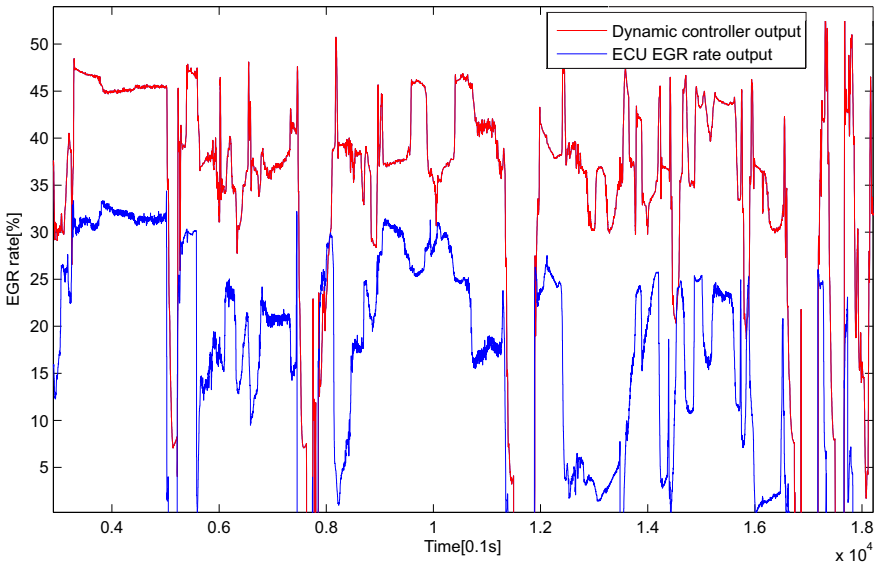

Figure 6.13: Comparison of Dynamic and ECU controllers for EGR rate

points output from both the dynamic optimised controller and from the standard ECU and the EGR resulting from the dynamic controller estimated by the ECU. Figure 6.16 shows comparison of simulated NOx for the dynamic controller and the standard ECU controller using the model developed in the identification stage. During the whole of the validation experiments the dynamic controller results in less NOx during steady state and dynamic events. Similarly in Figure 6.17 comparison of opacity is presented for the dynamic controller and the standard ECU. The dynamic controller results in removing some Opacity(smoke) spikes that appear with the standard ECU controller.

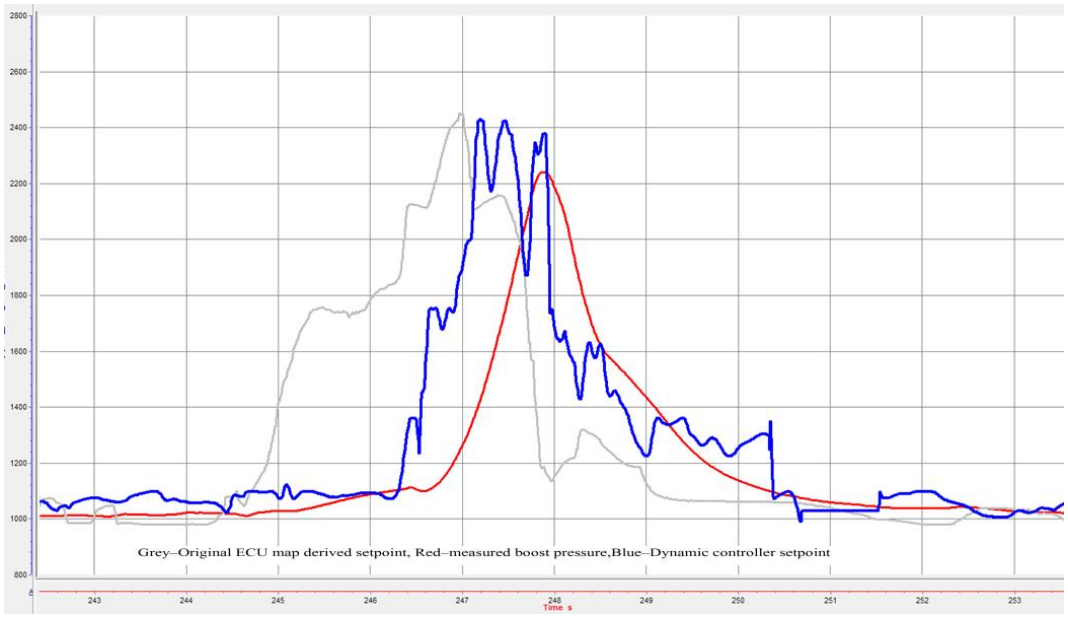

Figure 6.14: Dynamic and original ECU boost pressure setpoints and measured boost pressure

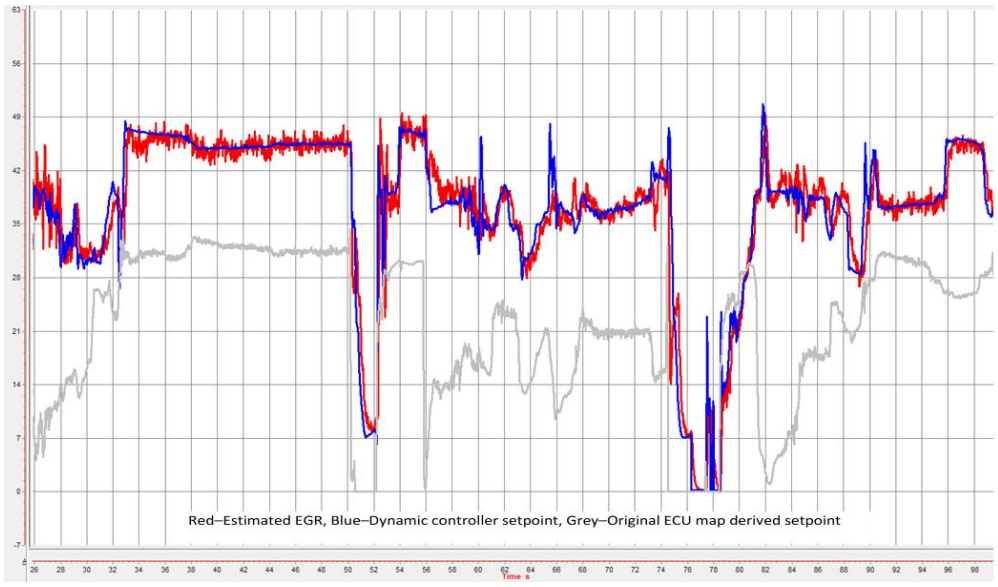

Figure 6.15: Dynamic control and standard calibration EGR setpoints and ECU estimated EGR

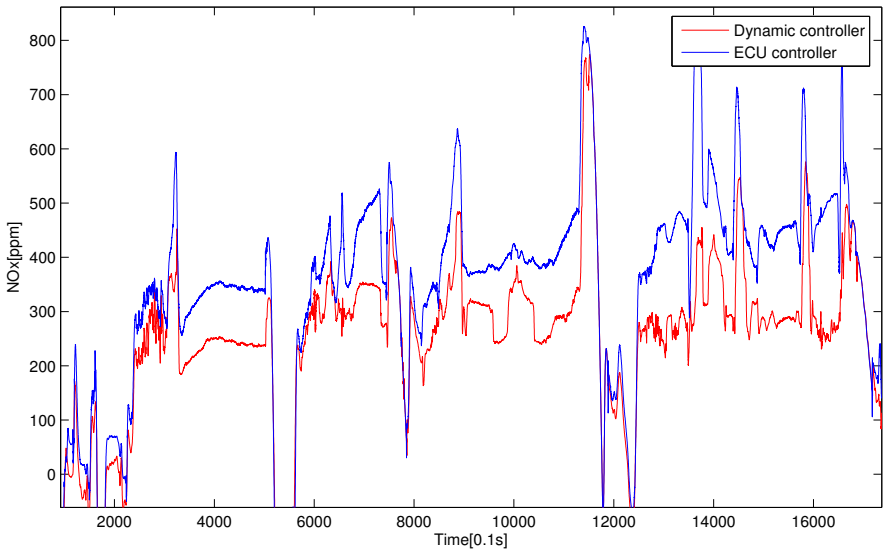

Figure 6.16: Comparison of simulated NOx for dynamic control and standard ECU calibration

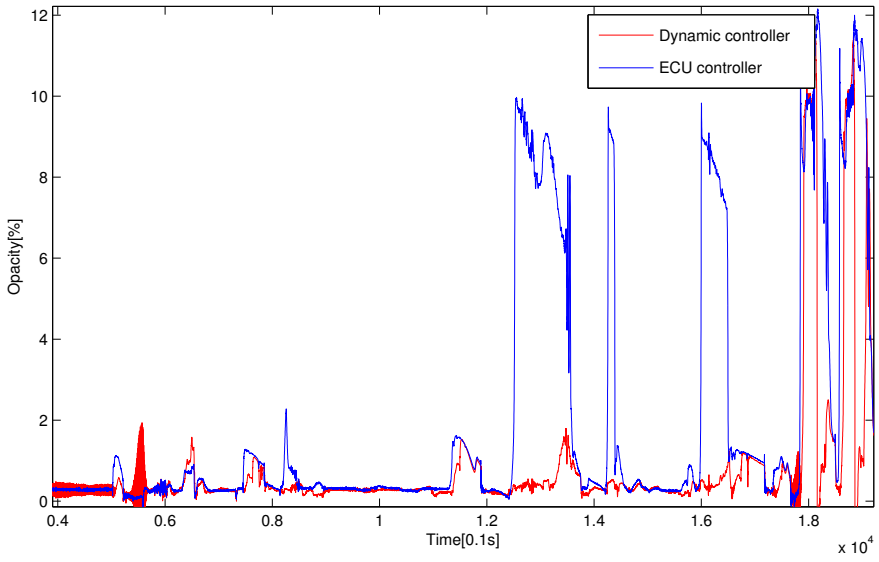

Figure 6.17: Comparison of simulated Opacity for dynamic control and standard ECU calibration

6.7 Results

A dynamic calibration SSNN methodology and an associated optimisation routine using the core NLP elements of the Quadratic Sequential Programming method was applied to obtaining control of the basic air-path of a Jaguar Land Rover turbocharged diesel engine. Validation results was demonstrated that the optimal controllers produced by the dynamic calibration approach can be developed with limited time on a test bed and applied in-vehicle without in-vehicle tuning, resulting in reducing NOx emissions and opacity.

Chapter 7

Conclusions and recommendation

This chapter presents conclusions from the work performed in this thesis and discuss potential future research areas in Inverse Optimal Behaviour based Dynamic Calibration.

7.1 Conclusions

The research reported in this thesis revealed the advantages of Inverse Optimal Behaviour based Dynamic Calibration over standard static calibrations by application of this methodology to two different types of calibration problems.

Firstly, a new dynamic calibration methodology was evaluated on a WAVE-RT virtual automotive Diesel engine to show its effectiveness for calibration including fuelling. The study answers the following research questions:*Is the Dynamic Calibration a viable methodology for producing a controller for smooth engine torque tracking with limited testing time while not exceeding the emission limits? Does dynamic calibration have any fuel consumption benefit over a conventional static maps?* The proposed methodology is based on system modelling using dynamic Neural Networks, a con-

strained numerical model-based optimization process to produce optimal input-output behaviour and finally inverse identification where the optimal controller is identified based the optimal input-output behaviour.

The study shows resulting 2% fuel economy improvements over a static calibration with smooth torque tracking whilst not exceeding emission limits for NOx and PM. In addition, only a limited amount of transient test data was required due to associated rapid data collection process which would result in a significant saving in testing time required for an experimental application compared with a steady state calibration.

Secondly, a comparative study of different model types for transient engine performance and emissions was evaluated the modelling capabilities of three model types:

- Polynomial model
- State Space Neural network model (SSNN)
- Neural Network based Non-linear Autoregressive with eXogenous input (NN-NARX)

to answer the research question: *What is the most efficient way of modelling the dynamic engine and emission behaviour in terms of number of parameters?*

SSNN models for engine and emission modelling were found to have similar simulation capabilities to NN-NARX despite a significantly simplified structure requiring significantly fewer parameters. A SSNN structured model that contains only five times as many parameters as a corresponding second order polynomial model is significantly better than the polynomial model in modelling unseen data and achieves comparable simulation capability to regular neural network NARX models which have a significantly higher number of parameters. For IMEP modelling the SSNN model with 90 parameters achieved 85,97% of *best fit* in a comparison to the NN-ARX model with 647 parameters achieving 88,97% and for Particular Number modelling the SSNN with 90 parameters achieved 25.25% and the NN-ARX

model with 1298 achieved only 22.14%. Both SSNN and NN-ARX models are shown to have superior simulation capability compared to polynomial models, but investigated NN models had more parameters than polynomial models so the comparison is only true for the chosen structures. The SSNN structure may be sufficiently simple to allow on-line implementation in forthcoming ECU modules. The SSNN is expected to inherit the usual advantages of state-space models in allowing the future exploitation of state-estimation methodologies or feedback linearisation in controller implementation.

Three different optimisation approaches were evaluated for use in the dynamic calibration to answer the question: *What is the best optimisation approach for optimising dynamic engine and emissions behaviour?* It was shown that numerical dynamic programming leads to very long computation for very simple calibration problems with simplified engine model and a very simple grid of possible states and controls. It would be necessary to extend it to more realistic models with more states, adding emission constraints or increasing grid density to achieve more realistic results but this would not be feasible in term of computational time.
Two direct optimisation approaches were implemented although for two different calibration problems, but the approach using the state space representation of the system dynamics was proven to be more efficient then the one using input-output models without splitting the optimisation problem into segments.

Thirdly, it was shown that the dynamic feed-forward maps are capable of replacing multiple 2-D maps resulting in a significant reduction in parameter number. The validation presented in this study constitutes answers for following questions: *Is it possible to apply this methodology only to Air-path only calibration problem? Is this technology ready to implement into production type of engine control units?* A systematic optimisation and identification method was presented to determine dynamic feed-forward maps for the Air-path calibration problem. The responses from the optimised controllers give a solution which although significantly different from the

original calibration gives a technologically valid alternative for the set of constraints used. The results presented in this thesis show the great potential of the methodology in the application of optimal control for an Air-path calibration problem. The combination of State Space Neural Network modelling techniques with direct optimisation was proven to be effective. The results from in-vehicle testing show that during transient engine operation this methodology can lead to elimination of some opacity peaks while reducing NOx at this same time. Reduction of opacity can lower the required frequency of regeneration of the Diesel Particulate Filter which can directly reduce fuel consumption. Additionally either more stringent emission standards can be reached or requirements for after-treatment system can potentially be lowered. Moreover this study shows that the dynamic feed-forward maps do not have excessive demands on the ECU and they can be implemented on a production type ECU.

7.2 Future work

There are several areas where the dynamic calibration methodology can be pursued further. First of all application of State Space Neural Network for torque modelling should be evaluated. Secondly torque constraints can be included in the new optimisation approach to evaluate the torque tracking capabilities of the new approach. Measurements of PM should be also taken into account instead of opacity as PM is the standard measurement variable which is present in the legislative standards. The in-vehicle validation should be extended with emission measurements to validate the true emission improvements.

Further investigation can be conducted into the application of dynamic calibration for actuators control of EGR valve position and VGT position by employing some boundary search algorithms to overcome issues reported in this thesis.

As State Space Neural Network models were used in this work with multiple input single output structure, extensions to multiple input multiple output

structure could be investigated for engine and emission modelling. This way the model would also contain relationship between the outputs resulting in improved simulation capability of the models.

Bibliography

[1] Tim J Barlow, S Latham, IS McCrae, and PG Boulter. *A reference book of driving cycles for use in the measurement of road vehicle emissions*. 2009.

[2] Benjamin Berger. *Modeling and Optimization for Stationary Base Engine Calibration*. PhD thesis, Universität München, 2012.

[3] D. Racine D. Cabush D. Wang, S. Yao and M. Shost. Ammonia sensor for scr reduction. Technical report, the DEER Conference Detroit MI, 2007.

[4] Moritz Diehl, Hans Georg Bock, Holger Diedam, and P-B Wieber. Fast direct multiple shooting algorithms for optimal robot control. In *Fast Motions in Biomechanics and Robotics*, pages 65–93. Springer, 2006.

[5] John B Heywood. *Internal combustion engine fundamentals*, volume 930. Mcgraw-hill New York, 1988.

[6] U Kiencke and L Nielsen. Automotive control systems for engine, driveline, and vehicle, 2000.

[7] Monica Tutuianu, Alessandro Marotta, Heinz Steven, Eva Ericsson, Takahiro Haniu, Noriyuki Ichikawa, and Hajime Ishii. Development of a world-wide worldwide harmonized light duty driving test cycle (wltc). Technical report, GRPE-68-03). Sponsoring Entity: UNECE, 2013.

[8] Hiroshi Uchida. Trend of turbocharging technologies. *R&D Review of Toyota CRDL*, 41(3):1–8, 2006.

[9] Ming Zheng, Graham T Reader, and J Gary Hawley. Diesel engine exhaust gas recirculation - a review on advanced and novel concepts. *Energy Conversion and Management*, 45(6):883–900, 2004.

[10] A Galip Ulsoy, Huei Peng, and Melih Çakmakci. *Automotive control systems*. Cambridge University Press, 2012.

[11] Lino Guzzella and Alois Amstutz. Control of diesel engines. *Control Systems, IEEE*, 18(5):53–71, 1998.

[12] A Brahma, D Upadhyay, A Serrani, and G Rizzoni. Modeling, identification and state estimation of diesel engine torque and nox dynamics in response to fuel quantity and timing excitations. In *American Control Conference, 2004. Proceedings of the 2004*, volume 3, pages 2166–2171. IEEE, 2004.

[13] Indranil Brahma. An investigation into the causes of transient particulate matter spikes in production diesel engines. In *ASME 2011 Internal Combustion Engine Division fall technical conference*, pages 765–783. American Society of Mechanical Engineers, 2011.

[14] C. Guhmann and J. M. Riedel. Comparison of identification methods for nonlinear dynamic systems. In *Proceedings of the 6th Conference Design of Experiments (DoE) in Engine Development*, 2011.

[15] Farraen Mohd Azmin, Richard K Stobart, John Rutledge, and Edward Winward. Using a statistical machine learning tool for diesel engine air path calibration. Technical report, SAE Technical Paper, 2014.

[16] Harald Stuhler, Thomas Kruse, Axel Stuber, Kurt Gschweitl, Walter Piock, Horst Pfluegl, and Peter Lick. Automated model-based gdi engine calibration adaptive online doe approach. Technical report, SAE Technical Paper, 2002.

[17] Yusuke Nozaki, Takao Fukuma, and Kazuo Tanaka. Development of a rule-based calibration method for diesel engines. Technical report, SAE Technical Paper, 2005.

[18] Chris Atkinson, LLC Atkinson, Marc Allain, and K Sisken. Demonstrating fuel consumption and emissions reductions with next generation model-based diesel engine control. In *Proceedings of Directions in Engine-Efficiency and Emissions Research (DEER) Conference Presentations*, 2011.

[19] Michael Benz. *Model-based optimal emission control of diesel engines*. PhD thesis, Diss., Eidgenössische Technische Hochschule ETH Zürich, Nr. 18796, 2010, 2010.

[20] Ke Fang. *Optimal test signal design and estimation for dynamic powertrain calibration and control*. PhD thesis, University of Liverpool, 2012.

[21] Rolf Isermann. *Engine modeling and control*. Springer, 2014.

[22] GianCarlo Pacitti, Steven Amphlett, Peter Miller, Robert Norris, and Anthony Truscott. Real-time crank-resolved engine simulation for testing new engine management systems. Technical report, SAE Technical Paper, 2008.

[23] GT-Power Userâ^{TM}s Manual. Gt-suiteâ„ version 6.1. *Gamma Technologies*, 2004.

[24] L.Guzzella C.Onder. Discrete-event models. In *Introduction to Modeling and Control of Internal Combustion Engine Systems*, pages 147–189. Springer Berlin Heidelberg, 2010.

[25] Minghui Kao and John J Moskwa. Turbocharged diesel engine modeling for nonlinear engine control and state estimation. *Journal of dynamic systems, measurement, and control*, 117(1):20–30, 1995.

[26] Elbert Hendricks and Spencer C Sorenson. Mean value modelling of spark ignition engines. Technical report, SAE Technical paper, 1990.

[27] Lars Eriksson. Mean value models for exhaust system temperatures. Technical report, SAE Technical Paper, 2002.

[28] Lars Eriksson and Lars Nielsen. *Modeling and control of engines and drivelines*. John Wiley & Sons, 2014.

[29] M. Bidarvatan V. Thakkar M.Shahbakhti B. Bahri A. Abdul Aziz. Grey-box modeling of {HCCI} engines. *Applied Thermal Engineering*, 70(1):397 – 409, 2014.

[30] Mehran Bidarvatan and Mahdi Shahbakhti. Gray-box modeling for performance control of an hcci engine with blended fuels. *Journal of Engineering for Gas Turbines and Power*, 136(10):101510, 2014.

[31] Chris Atkinson and Gregory Mott. Dynamic model-based calibration optimization: An introduction and application to diesel engines. Technical report, SAE Technical Paper, 2005.

[32] Lennart Ljung. *System identification*. Springer, 1998.

[33] Torsten Söderström and Petre Stoica. *System identification*. Prentice-Hall, Inc., 1988.

[34] Jonas Sjöberg, Qinghua Zhang, Lennart Ljung, Albert Benveniste, Bernard Delyon, Pierre-Yves Glorennec, Håkan Hjalmarsson, and Anatoli Juditsky. Nonlinear black-box modeling in system identification: a unified overview. *Automatica*, 31(12):1691–1724, 1995.

[35] Lee Zongyan. *Model Structure Selection in Powertrain Calibration and Control*. PhD thesis, University of Liverpool, 2012.

[36] Ding-Li Yu and Michael Beham. Comparative study on engine torque modelling using different neural networks. In *Advances in Neural Networks–ISNN 2005*, pages 865–870. Springer, 2005.

[37] Karsten Röpke, Mirko Knaak, Adrian Neßler, and Steffen Schaum. Rapid measurement. *MTZ worldwide*, 68(4):16–19, 2007.

[38] Tomáš Polóni, Tor Arne Johansen, and Boris Rohal'-Ilkiv. Modeling of air-fuel ratio dynamics of gasoline combustion engine with arx network. *Journal of Dynamic Systems, Measurement, and Control*, 130(6):061009, 2008.

[39] Paul MJ Van Den Hof and Ruud JP Schrama. Identification and controlâ"closed-loop issues. *Automatica*, 31(12):1751–1770, 1995.

[40] P Van Den Hof and Raymond A de Callafon. Multivariable closed-loop identification: from indirect identification to dual-youla

parametrization. In *Decision and Control, 1996., Proceedings of the 35th IEEE Conference on*, volume 2, pages 1397–1402. IEEE, 1996.

[41] Yucai Zhu. *Multivariable system identification for process control*. Elsevier, 2001.

[42] Nobuyuki Sakushima, Wolf Baumann, Karsten Röpke, and Mirko Knaak. Transient modeling of diesel engine emissions. 2013.

[43] Ai Hui Tan and Keith R Godfrey. The generation of binary and near-binary pseudo-random signals: An overview. In *Instrumentation and Measurement Technology Conference, 2001. IMTC 2001. Proceedings of the 18th IEEE*, volume 2, pages 766–771. IEEE, 2001.

[44] Michael Deflorian and Susanne Zaglauer. Design of experiments for nonlinear dynamic system identification. In *IFAC 18th World Congress, Milano 2011*, 2011.

[45] M. Berenthino. *On input design in system identification for control*. PhD thesis, Stockholm, 2006.

[46] Michael Unser. Sampling-50 years after shannon. *Proceedings of the IEEE*, 88(4):569–587, 2000.

[47] Karl Johan Åström and Peter Eykhoff. System identification-a survey. *Automatica*, 7(2):123–162, 1971.

[48] Petre Stoica and Torsten Söderström. Bias correction in least-squares identification. *International Journal of Control*, 35(3):449–457, 1982.

[49] Lennart Ljung. Prediction error estimation methods. *Circuits, Systems and Signal Processing*, 21(1):11–21, 2002.

[50] Lennart Ljung. Recursive identification algorithms. *Circuits, Systems and Signal Processing*, 21(1):57–68, 2002.

[51] Ardalan Vahidi, Anna Stefanopoulou, and Huei Peng. Recursive least squares with forgetting for online estimation of vehicle mass and road grade: theory and experiments. *Vehicle System Dynamics*, 43(1):31–55, 2005.

[52] Ezio Alfieri. *Emissions-controlled diesel engine*. PhD thesis, Diss., Eidgenössische Technische Hochschule ETH Zürich, Nr. 18214, 2009, 2009.

[53] M Hafner, M Schüler, O Nelles, and R Isermann. Fast neural networks for diesel engine control design. *Control Engineering Practice*, 8(11):1211–1221, 2000.

[54] Rabih Omran, Rafic Younes, and J-C Champoussin. Optimal control of a variable geometry turbocharged diesel engine using neural networks: Applications on the etc test cycle. *Control Systems Technology, IEEE Transactions on*, 17(2):380–393, 2009.

[55] Chris Atkinson, LLC Atkinson, Marc Allain, and Craig Savonen. Model-based transient calibration optimization for next generation diesel engines. In *11th Diesel Engine Emission Reduction (DEER) Conference*, 2005.

[56] Guoming G Zhu, Ibrahim Haskara, and Jim Winkelman. Closed-loop ignition timing control for si engines using ionization current feedback. *Control Systems Technology, IEEE Transactions on*, 15(3):416–427, 2007.

[57] Andreas F Kolbeck. Closed loop combustion control-enabler of future refined engine performance regarding power, efficiency, emissions & nvh under stringent governmental regulations. Technical report, SAE Technical Paper, 2011.

[58] Ibrahim Haskara and Yue-Yun Wang. Cylinder pressure-based combustion controls for advanced diesel combustion with multiple-pulse fuel injection. *Control Systems Technology, IEEE Transactions on*, 21(6):2143–2155, 2013.

[59] S. Formentin, M. Corno, H. Waschl, D. Alberer, and S.M. Savaresi. Nox estimation in diesel engines via in-cylinder pressure measurement. *Control Systems Technology, IEEE Transactions on*, 22(1):396–403, Jan 2014.

[60] Samir Saraswati and Satish Chand. Reconstruction of cylinder pressure for si engine using recurrent neural network. *Neural Computing and Applications*, 19(6):935–944, 2010.

[61] Xia Yong, Hao Guiyou, Shan Chunrong, Ni Zhibing, and Zhang Wu. Reconstruction of cylinder pressure of ic engine based on neural networks. In *Pervasive Computing Signal Processing and Applications (PCSPA), 2010 First International Conference on*, pages 924–927. IEEE, 2010.

[62] Geoff Dearden and Tom Shenton. Laser ignited engines: progress, challenges and prospects. *Optics express*, 21(106):A1113–A1125, 2013.

[63] Bradley Charles Glenn, Devesh Upadhyay, and Gregory N Washington. Control design of electrically assisted boosting systems for diesel powertrain applications. *Control Systems Technology, IEEE Transactions on*, 18(4):769–778, 2010.

[64] Christopher Vagg, S Akehurst, CJ Brace, and Lloyd Ash. Model-based optimal control of a hybrid electric vehicle using stochastic dynamic programming. In *6th Conference on Simulation and Testing for Automotive Electronics*. University of Bath, 2014.

[65] Olle Sundström, Lino Guzzella, and Patrik Soltic. Optimal hybridization in two parallel hybrid electric vehicles using dynamic programming. In *Proceedings of the 17th IFAC world congress*, volume 17, pages 4642–4647, 2008.

[66] Chia-Jui Chiang and Anna G Stefanopoulou. Stability analysis in homogeneous charge compression ignition (hcci) engines with high dilution. *Control Systems Technology, IEEE Transactions on*, 15(2):209–219, 2007.

[67] Nick J Killingsworth, Salvador M Aceves, Daniel L Flowers, Francisco Espinosa-Loza, and Miroslav Krstic. Hcci engine combustion-timing control: Optimizing gains and fuel consumption via extremum seeking. *Control Systems Technology, IEEE Transactions on*, 17(6):1350–1361, 2009.

[68] Michael Rinehart, Munther Dahleh, Dennis Reed, and Ilya Kolmanovsky. Suboptimal control of switched systems with an application to the disc engine. *Control Systems Technology, IEEE Transactions on*, 16(2):189–201, 2008.

[69] Kenneth R Muske, James C Peyton Jones, and EM Franceschi. Adaptive analytical model-based control for si engine air–fuel ratio. *Control Systems Technology, IEEE Transactions on*, 16(4):763–768, 2008.

[70] Johan Wahlstrom and Lars Eriksson. Output selection and its implications for mpc of egr and vgt in diesel engines. *Control Systems Technology, IEEE Transactions on*, 21(3):932–940, 2013.

[71] Johan Wahlstrom, Lars Eriksson, and Lars Nielsen. Egr-vgt control and tuning for pumping work minimization and emission control. *Control Systems Technology, IEEE Transactions on*, 18(4):993–1003, 2010.

[72] Peter Ortner and Luigi del Re. Predictive control of a diesel engine air path. *Control Systems Technology, IEEE Transactions on*, 15(3):449–456, 2007.

[73] Frank Willems and Robert Cloudt. Experimental demonstration of a new model-based scr control strategy for cleaner heavy-duty diesel engines. *IEEE Transactions on Control Systems Technology*, 19(5):1305–1313, 2011.

[74] Indranil Brahma and JN Chi. Development of a model-based transient calibration process for diesel engine electronic control module tables–part 1: data requirements, processing, and analysis. *International Journal of Engine Research*, page 1468087411424376, 2011.

[75] Matlab 2010b. *fmincon Interior-Point Algorithm with Analytic Hessian*. MathWork, 2010.

[76] P. Dickinson and A.T. Shenton. Dynamic calibration of fuelling in the pfi si engine. *Control Engineering Practice*, 17(1):26–38, 2009.

[77] Oliver Louis Robert Jacobs. Introduction to control theory. 1974.

[78] R. Bellman. Dynamic programming. 1957.

[79] O. Sundstom L.Guzzella. A generic dynamic programming matlab function. *18th IEEE International Conference on Control Applications*, pages 1625–1630, 2009.

[80] A.V. Rao. A survey of numerical methods for optimal control. *2009 AAS/AIAA Astrodynamics Specialist Conference*, 2009.

[81] Oliver Junge, Jerrold E Marsden, and Sina Ober-Blöbaum. Discrete mechanics and optimal control. In *Proceedings of the 16th IFAC World Congress*, volume 16, pages 00310–1, 2005.

[82] O. Stryk. Direct and indirect methods for trajectory optimisation. *Annals of Operations Research*, 37:508–210, 1992.

[83] O. Stryk. Numerical solution of optimal control problems by direct collocation. *Optimal Control - Caculus of Variations, Optimal Control Theory and Numrical Methods*, (111), 2005.

[84] Michael A Patterson and Anil Rao. Exploiting sparsity in direct collocation pseudospectral methods for solving optimal control problems. *Journal of Spacecraft and Rockets*, 49(2):354–377, 2012.

[85] Philip E Gill, Walter Murray, and Michael A Saunders. Snopt: An sqp algorithm for large-scale constrained optimization. *SIAM journal on optimization*, 12(4):979–1006, 2002.

[86] Dimitri P Bertsekas, Dimitri P Bertsekas, Dimitri P Bertsekas, and Dimitri P Bertsekas. *Dynamic programming and optimal control*, volume 1. Athena Scientific Belmont, MA, 1995.

[87] L. Guzzella P.Elbert S.Ebbesen. Implementation of dynamic programming for -dimensional optimal control problems with final state constraints. *IEEE Transactions on Control Systems Technology*, 21(3):924–931, 2013.

[88] Ioanid Rosu. The bellman principle of optimality. *Availiable at: http://faculty. chicagogsb. edu/ioanid. rosu/research/notes/bellman. pdf*, 2002.

[89] Steven M LaValle. *Planning algorithms*. Cambridge university press, 2006.

[90] J Sola and J Sevilla. Importance of input data normalization for the application of neural networks to complex industrial problems. *Nuclear Science, IEEE Transactions on*, 44(3):1464–1468, 1997.

[91] U Kotta and M Tonso. Realization of discrete-time nonlinear input-output equations: polynomial approach. In *Intelligent Control and Automation, 2008. WCICA 2008. 7th World Congress on*, pages 529–534. IEEE, 2008.

[92] Ülle Kotta, Fahmida N Chowdhury, and Sven Nõmm. On realizability of neural networks-based input–output models in the classical state-space form. *Automatica*, 42(7):1211–1216, 2006.

[93] Isabelle Rivals and Léon Personnaz. Black-box modeling with state-space neural networks. *Neural Adaptive Control Technology*, 15:237, 1996.

[94] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[95] Martin T Hagan and Mohammad B Menhaj. Training feedforward networks with the marquardt algorithm. *Neural Networks, IEEE Transactions on*, 5(6):989–993, 1994.

[96] Kowalczyk M. Schreiber A. and Isermann R. Method for dynamic on-line identfication with integrated determination of operation boundaries. In Ropke, editor, *6th. Conference Design of Experiments in engine development*. IAV, Expert-Verlag, 2011.

[97] Nomm S. Peltenkov E. and Kotta U. Neural network based anarax structure for identification and model based control. In *9th International Conference on Control,Automation, Robotics and Vision,*, pages 2284 – 2288. IEEE, Expert-Verlag, 2006.

[98] Kumpati S Narendra and Kannan Parthasarathy. Identification and control of dynamical systems using neural networks. *Neural Networks, IEEE Transactions on*, 1(1):4–27, 1990.

[99] MohameD Chtourou, Nabil Derbel, and Mohamed Ben Ali Kamoun. Control of a loaded induction machine using a feedforward neural network. *International journal of systems science*, 27(12):1287–1295, 1996.

[100] GC Luh and CY Wu. Inversion control of non-linear systems with an inverse narx model identified using genetic algorithms. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 214(4):259–271, 2000.

[101] A.P. Petridis and A.T. Shenton. Inverse-narma: a robust control method applied to si engine idle-speed regulation. *Control Engineering Practice*, 11(3):279–290, 2003.

[102] Chen M.Y. Shenton A.T., K.Fang and P.B.Dickinson. Engine dynamic calibration and control by inverse optimal behaviours. In *PMC Conference Bradford*, 2012.

[103] Vivek Jaikamal and Nigel Tracey. Ehooks–prototyping is rapid again. Technical report, SAE Technical Paper, 2009.

[104] Alexandra Grancharova and Tor Arne Johansen. *Explicit nonlinear model predictive control: theory and applications*, volume 429. Springer, 2012.

[105] Olle Sundstrom and Lino Guzzella. A generic dynamic programming matlab function. In *Control Applications,(CCA) & Intelligent Control,(ISIC), 2009 IEEE*, pages 1625–1630. IEEE, 2009.

[106] John May, Dirk Bosteels, and Cecile Favre. An assessment of emissions from light-duty vehicles using pems and chassis dynamometer testing. *SAE Int. J. Engines*, 7:1326–1335, 04 2014.