UNIVERSITEIT·STELLENBOSCH·UNIVERSITY
jou kennisvennoot • your knowledge partner

Simulation Based Calibration of Turbo-Charger Boost Control

by

Schalk Willem Pienaar

Thesis presented in partial fulfilment of the requirements for the degree of Master of Science in Mechanical Engineering at the University of Stellenbosch

Department of Mechanical Engineering
Stellenbosch University
Private Bag X1, 7602 Matieland, South Africa

Supervisors:

Dr. A.B. Taylor   Prof. T. W. von Backström

November 2007

Declaration

I, the undersigned, hereby declare that the work contained in this thesis is my own original work and that I have not previously in its entirety or in part submitted it at any university for a degree.

Signature: ....
S.W. Pienaar

Date: ....

Abstract

Simulation Based Calibration of Turbo-Charger Boost Control

S.W. Pienaar
Department of Mechanical Engineering
Stellenbosch University
Private Bag X1, 7602 Matieland, South Africa

Thesis: MScEng (Mech)
November 2007

Electronic engine control systems utilise tables, or maps, of data to determine the set-points of the various actuators on the engine and to calculate the values of variables that cannot be directly measured. To ensure accurate control of the engine processes the values in these maps have to be accurately calibrated for the particular engine being controlled.

Due to the complex interaction of the various systems in the engine it is becoming more and more difficult for human calibration engineers to be able to take all the effects of changes to a particular parameter into account. This problem is made worse by increasingly strict emissions regulations and performance demands from the customer. The process of calibrating the maps in an Electronic Control Unit (ECU) is also very resource intensive since it involves taking a test engine installed in an engine test cell to every operating point on the various maps and adjusting the map values until the desired response is achieved.

The aim of this project was to develop a solution to this problem in the form of a simulation based calibration system. The proposed system would use an accurate model of the engine to simulate the effect of various map values on the engine response. This data would then be used to find the map values that would enable the engine to deliver a desired torque curve. In the case where it is not practical to use engine simulation the system would be able to process a database created by testing an actual engine. This testing could also be automated.

To achieve this aim the AutoCal program was created. This program can manage a commercial engine simulation code to create a database of the effect of various calibration values on the engine response. It can then evaluate the created database subject to user defined operating constraints and find calibration values that will deliver a desired torque curve. It can also be used to evaluate and process databases created by engine testing.

To provide the data required for the development and testing of the AutoCal program, a naturally-aspirated engine was turbo-charged and tested at various operating points. The resulting data was used to calibrate and validate a model of the engine created and simulated with the WAVE software package from Ricardo.

The project was focused on finding calibration values for the maps used to control the turbo-charger wastegate and ignition timing of the test engine. Work was limited to the full load operating region and fixed Air/Fuel Ratio (AFR) values were used.

The project showed that simulation based calibration can be used to calibrate control system maps once an accurate model of the engine being controlled has been created. Very useful insight was gained into the process of building, testing and modelling a turbo-charged internal-combustion engine and calibrating modern electronic engine control systems. The end result is a useful engineering tool with the following functions:

- Automatically simulating the effect of various control inputs on engine performance.
- Determining the correct calibration settings to deliver the desired performance subject to user-definable constraints. This can be done using results from simulation or physical engine testing in the case of simulation tools not being available.
- Providing the data required when calibrating the engine model used during simulation by processing and displaying the outputs of the simulation program compared to test data.
- Plotting any test or simulation results in a format configured by the user.

Using this tool facilitates a more structured and less resource intensive approach to engine control system calibration.

Uittreksel

Simulasie-gebaseerde Kalibrasie van Turbo-Aanjaer Beheer

("Simulation Based Calibration of Turbo-Charger Boost Control")

S.W. Pienaar

Departement Meganiese Ingenieurswese
Universiteit van Stellenbosch
Privaatsak X1, 7602 Matieland, Suid Afrika

Tesis: MScIng (Meg)

November 2007

Elektroniese enjinbeheereenhede maak gebruik van kalibrasiewaardes, vervat in verskeie datatabelle, om die stelpunte van die onderskeie aktueerders op die enjin te bepaal. Die kalibrasiewaardes word ook gebruik om waardes wat nie direk meetbaar is nie, te bereken. Hierdie tabelle moet akkuraat gekalibreer word vir elke spesifieke enjinmodel om te verseker dat al die prosesse wat in en om die enjin plaasvind, akkuraat beheer word.

Vanweë die komplekse interaksie tussen die verskeie stelsels in die moderne enjin, word dit al hoe moeiliker vir kalibrasie ingenieurs om al die gevolge, wat ’n verstelling aan ’n spesifieke parameter teweegbring, in ag te neem. Hierdie probleem word vererger deur toenemend streng regulasies met betrekking tot uitlaatgasse en verhoogde verrigtingsverwagtinge van kliënte. Die kalibrasieproses waardeur ’n enjinbeheereenheid moet gaan, is arbeids- en hulpbronintensief, aangesien die enjin wat gekalibreer word in ’n laboratorium op ’n toetsbank opgestel moet word vir uitgebreide toetswerk. Gedurende hierdie toetswerk, word die enjin na elke moontlike bedryfstoestand geneem en die kalibrasiewaardes van die beheereenheid gestel, totdat die stelsel na wense funksioneer.

Die doel van hierdie projek was om ’n oplossing, in die vorm van ’n simulasiegebaseerde kalibrasiestelsel, vir bogenoemde probleme te vind. Die voorgestelde stelsel gebruik ’n akkurate model van die enjin om die effek van verskeie kalibrasiewaardes op die enjin se werksverrigting te simuleer. Hierdie data word vervolgens gebruik om die versameling kalibrasiewaardes wat ’n gewensde draaimoment-kurwe teweegbring, te bepaal. Waar die gebruik van ’n simulasieprogram onprakties is, kan die kalibrasiestelsel ook gebruik maak van die data afkomstig van toetswerk op ’n fisiese enjin. Hierdie toetswerk kan moontlik ge-outomatiseer word.

Ten einde die doelwit van die projek te bereik, is die AutoCal-program geskep. AutoCal skep ’n databasis van die effek van verskeie kalibrasiewaardes, deur ’n kommersiële enjinsimulasie-pakket te bestuur. Hierdie databasis word geëvalueer, onderhewig aan sekere bedryfsbeperkings wat deur die gebruiker verskaf word. Gedurende die evaluasie vind AutoCal die kalibrasiewaardes wat ’n gewensde draaimoment-kurwe sal lewer. Dit kan ook ’n databasis wat deur toetswerk geskep is, evalueer.

Ter ondersteuning van die ontwikkeling van AutoCal, is ’n binnebrandenjin met ’n turbo-aanjaer toegerus en getoets. Die toetsdata is gebruik om ’n model van die enjin te kalibreer. Om die model te skep, is gebruik gemaak van Ricardo se WAVE-modelleringspakket.

Die spesifieke doelwit van die projek was om kalibrasiewaardes vir die tabelle, wat gebruik word om die turbo-aanjaer en elektroniese ontsteking te beheer, te vind. ’n Vaste lug- tot brandstofverhouding is gebruik en die studie is beperk tot die vollas-bedryfsgebied.

Die projek het getoon dat ’n simulasie-gebaseerde kalibrasiestelsel gebruik kan word om enjin-beheereenhede te kalibreer, sodra ’n akkurate enjinmodel geskep is. Praktiese kennis met betrekking tot die bou, toets en modellering van ’n turbo-aangejaagde binnebrandenjin, sowel as die kalibrasie van elektroniese beheereenhede, is verkry. Die eindresultaat van die projek is ’n bruikbare gereedskapstuk, met die volgende funksies:

- Die outomatiese simulasie van die effek van verskeie beheerinsette op enjinwerksverrigting.
- Die bepaling van die korrekte kalibrasiewaardes om ’n gewensde draaimoment-kurwe te lewer, onderhewig aan beperkings soos gedefineer deur die gebruiker. Die program kan simulasie- of toetsdata as inset hiervoor gebruik.
- Die voorsiening van die data wat gedurende die model-kalibrasieproses benodig word, deur die uitsette van die simulasiemodel met toetsdata te vergelyk en te vertoon.
- Die vertoon van enige beskikbare toets- of simulasiedata in die databasis.

Die AutoCal program en die geasosieerde prosesse fasiliteer ’n meer geordende kalibrasieproses wat ook minder hulpbronintensief is.

Acknowledgements

I would like to thank my supervisor, Dr. Andrew Taylor, for the abundant knowledge, about engineering and life in general, that he has shared so freely over the years and also for believing in me. I am also very grateful towards my co-supervisor, Prof. Theo von Backström, for his patience and positive encouragement.

Stellenbosch Automotive Engineering (CAE) and Volkswagen of South Africa (VWSA) contributed very generously with equipment, test facilities and general technical support. Lastly I want to thank the NRF and THRIP for their financial support of this project.

SOLI DEO GLORIA

Dedications

Aan Martinette, my wonderlike vrou, sonder wie se geduld, hulp en liefde, hierdie werk nooit sou plaasgevind het nie.

Contents

List of Figures

List of Tables

Acronyms

AFR	Air/Fuel Ratio
BDC	Bottom Dead Centre
BMEP	break mean effective pressure
BPE	barometric pressure estimator
BSFC	Brake Specific Fuel Consumption
CA	Crank Angle
CAD	Computer Assisted Design
CAE	Stellenbosch Automotive Engineering
CFD	Computational Fluid Dynamics
csv	Comma Separated Value
ECU	Electronic Control Unit
EGO	Exhaust Gas Oxygen Sensor
EGR	Exhaust Gas Recirculation
ETA	Engine Test Automation
EVC	Exhaust Valve Closing
FMEP	friction mean effective pressure
GUI	Graphical User Interface
HEGO	Heated Exhaust Gas Oxygen Sensor
IEG	Induction-to-Exhaust-Gas
IMEP	indicated mean effective pressure

IVO	Inlet Valve Open
MBT	Maximum Brake Torque
MVM	Mean-Value Models
NA	Naturally Aspirated
NO_x	Nitrogen Oxides
PID	Proportional-integral-derivative
PLC	Programmable Logic Controller
PMAX	maximum in-cylinder pressure
PWM	Pulse Width Modulated
rpm	revolutions per minute
RPM	Revolutions per Minute
RON	Research Octane Number
SAE	Society of Automotive Engineers
SAM 2000	Siemens Application and Measurement
SBC	Simulation Based Calibration
SFC	Specific Fuel Consumption
TDC	Top Dead Centre
TMAP	Temperature and Manifold Absolute Pressure
VGT	Variable Geometry Turbine
VVA	Variable Valve Actuation
VW	Volkswagen
VWSA	Volkswagen of South Africa
WOT	Wide Open Throttle

Chapter 1

Introduction

Modern model based engine control systems work on the principles outlined in the sequence below:

- An ECU reads the outputs of various sensors on the engine and the vehicle.
- It then calculates additional variables with the aid of characteristic models of the engine systems.
- This information is used in conjunction with a collection of look-up tables (maps) calibrated for each specific engine type, to determine which signals to send to a selection of actuators on the engine.
- These actuators then control the engine to provide the response requested by the driver, while also satisfying other criteria such as emissions compliance.

According to Bergström (2003) there are two main ways of building the characteristic models of the various systems; physical modelling and identification. Physical modelling means that the system being modelled is divided into subsystems, with known behaviour. For technical systems, this in general means that the laws of physics are used to describe the subsystems.

Identification means that an observation from the system is used to adjust the model properties to the system properties. The data gathered during this identification process is also stored in various look-up tables referred to by the ECU. This principle is often used as a complement to the first one.

Both the look-up tables originating from the identification process and those used to determine actuator signals, are usually created through a process of exhaustive test bench and in-vehicle testing. Due to the complexity of modern control systems which have to cope with ever increasing demands in the areas of emissions and drivability in particular, the process of finding the calibration values stored in these look-up tables is very labour and equipment intensive and in some instances almost becomes an art.

Recently it has become apparent that it is less and less feasible to obtain the optimum values for all the points in the look-up tables in an ECU with traditional methods. This has led to the relatively new field of Simulation Based Calibration (SBC). SBC involves the modelling of the engine system with an accurate simulation package before utilising co-simulation with one or more other software packages to run the simulation through all feasible operating conditions, thereby creating a matrix of possible calibration values. This data is then entered into an optimisation program together with certain constraints and the condition to be optimised for, such as maximum performance or lowest emissions. The optimisation program then outputs the optimum values for the look-up table being calibrated. This process offers the major advantage of being able to simultaneously consider various interrelated processes in the operation of the engine, which would be difficult and time consuming for a human calibration engineer. It also offers noticeable savings in time and money, by avoiding long hours spent on a dynamometer or in a vehicle.

The goal of this project was to develop a simulation based calibration process and use it to find base calibration values for the tables controlling turbo-charger boost pressure and spark timing, on a turbo-charged spark ignition engine.

1.1 Research Questions

The questions posed before starting this project were:

1. Is it possible to use engine simulation software to assist the calibration engineer in the process of calibrating a map-based engine control system?
2. Can a software tool be created to analyse the data from engine testing and simulation and find the ECU calibration values to deliver a desired torque output, subject to certain operating constraints?

To answer these questions the following objectives were set:

1. Create and validate a simulation model of a turbo-charged internal-combustion spark-ignition engine, using a commercially available engine simulation package.
2. Develop a technique for using a software program to find the most suited calibration values of the turbo-charger boost and ignition maps. Specifically for a desired, full-load torque curve, subject to pre-determined constraints such

as air fuel ratio, maximum intake air temperature, maximum exhaust temperature and maximum turbo-charger speed, from a database of simulated values.

3. Simulate the performance of the engine with the calibration values from the software program and compare the results with the desired torque curve.

1.2 Methods

Below is a basic outline of how the three objectives defined above were achieved.

1. A naturally aspirated test engine was fitted with a turbo-charger and installed in an engine test cell. The engine was tested under various operating conditions in order to gather the data required for validating the engine simulation model. The process of preparing and testing the engine is covered in detail in Appendix A. An existing Ricardo WAVE model of the naturally aspirated test engine was adapted to represent the test engine after being turbocharged. This model was calibrated using test data from the turbo-charged engine. The modelling and calibration process is covered in Appendix B.
2. A software program called AutoCal was created to automatically run WAVE simulations and process the resulting data. A series of simulations was run, each with different combinations of the possible calibration values for the turbo-charger boost and ignition timing maps being calibrated. An interpolation process was then applied to the database of simulation values and used to find the calibration values best suited to achieving a desired torque curve, while meeting the set constraints. The development of the AutoCal program is discussed in Chapter 3.
3. The AutoCal program was also programmed with the facility to simulate the effect of a chosen set of calibration values on the engine simulation model and compare the results with the desired torque curve. This was used to confirm the accuracy of the above mentioned interpolation process. This application and evaluation process is covered in Chapter 4.

Chapter 2

Literature Survey

2.1 Theory of Engine Control

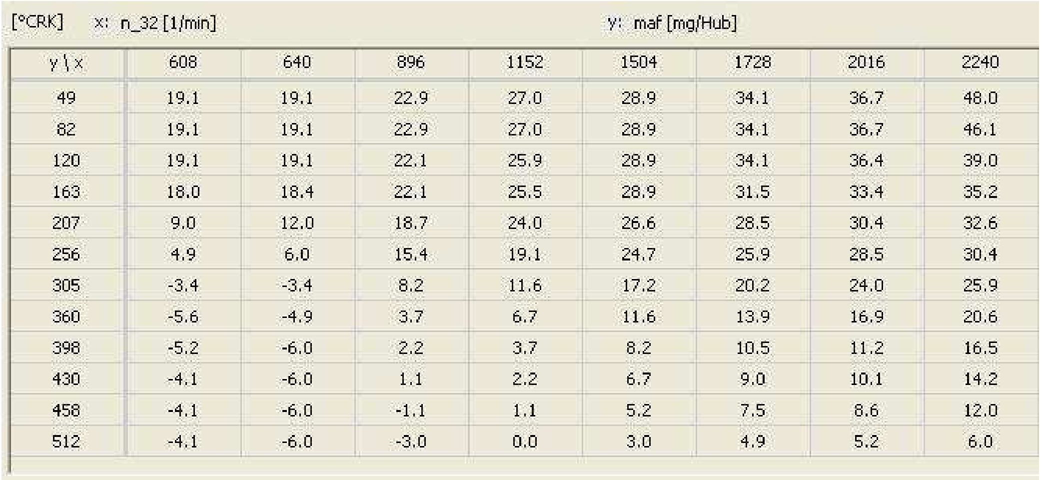
A modern engine control system consists of the following main components (Steinmeyer, 1995):

- Sensors to determine the operating state of the engine.
- A processor to interpret the signals from the sensors, calculate additional parameters that are not measured by using various models and determine the required actions to reach the desired operating state.
- Controllers to implement the required actions.
- And lastly the actuators used to bring about the control actions. The desired operating state is determined from various inputs including the driver and the various safety and emissions control systems, if implemented.

This section starts with an explanation of the two different methodologies used in engine control systems, namely the original map based systems and the more modern model based control systems which are an increasingly necessary requirement for meeting the high prevailing and future control demands. The process being controlled is then described along with the need for accurate control and the sensors and actuators used. Lastly the two areas this project focused on namely turbo-charger boost control and ignition control, are described in more detail.

2.1.1 Map Based Engine Control Systems

Map based control systems utilise tables of values to determine the settings for the various actuators being controlled. These tables usually have reference points defined along the upper row and left most column, an example of which is given in Figure 2.1.



[°CRK] x: n_32 [1/min] y: maf [mg/Hub]

y \ x	608	640	896	1152	1504	1728	2016	2240
49	19.1	19.1	22.9	27.0	28.9	34.1	36.7	48.0
82	19.1	19.1	22.9	27.0	28.9	34.1	36.7	46.1
120	19.1	19.1	22.1	25.9	28.9	34.1	36.4	39.0
163	18.0	18.4	22.1	25.5	28.9	31.5	33.4	35.2
207	9.0	12.0	18.7	24.0	26.6	28.5	30.4	32.6
256	4.9	6.0	15.4	19.1	24.7	25.9	28.5	30.4
305	-3.4	-3.4	8.2	11.6	17.2	20.2	24.0	25.9
360	-5.6	-4.9	3.7	6.7	11.6	13.9	16.9	20.6
398	-5.2	-6.0	2.2	3.7	8.2	10.5	11.2	16.5
430	-4.1	-6.0	1.1	2.2	6.7	9.0	10.1	14.2
458	-4.1	-6.0	-1.1	1.1	5.2	7.5	8.6	12.0
512	-4.1	-6.0	-3.0	0.0	3.0	4.9	5.2	6.0

Figure 2.1: Example of the interface to an ignition map as described in the text.

The signals from the sensors either correspond directly to these reference values, for example temperatures and speed, or are used in calculations to determine the reference values, such as air mass flow calculated from the air temperature, pressure and engine speed. If the intersection of the reference values do not correspond directly to a value in the table, the system will either interpolate between the surrounding values, use the value closest to the intersecting point, or use one value until the referencing variable reaches the next value on the referencing axis. Which strategy to use depends on the accuracy of control needed for the process being controlled by that specific table as well as the processing power available on the ECU.

The referenced values from the tables are then used, usually after performing more calculations with them, by a controller to drive the actuators. To control the engine during steady state conditions, maps are normally used together with feedback control to take care of disturbances. Transients are handled by separate controllers which operate in parallel with a disturbance rejection controller (Andersson *et al.*, 1999). Separate controllers have the problem of strong interactions between them since they are both controlling the same event. One way to reduce these interactions is by using a model based control structure as explained in the next subsection.

The values in these tables are determined during an exhaustive calibration process during which the engine is run on a dynamometer and taken to each operating point of the various tables used by the control system. Each applicable table value is then iteratively adjusted until the desired engine behaviour is achieved. Changes in one map can affect other maps, resulting in little independent freedom of change. With this multiple interaction, fine-tuning of the control system calibration becomes an art (Rask and Snellnau, 2004), (Challen and Stobart, 1998).

Additional adjustments are made during in-vehicle testing to account for the different operating conditions in the vehicle as compared to the dynamometer. These include differences in amongst others the engine cooling system, transient conditions and the mechanical and electronic noises that the control system has to cope with.

2.1.2 Model Based Engine Control Systems

Models to Improve Controller Performance

With the increasing demands in terms of accuracy and functionality placed on modern control systems, it is often required to have information about a variable which cannot be measured directly (Stroh *et al.*, 2001). Or cannot be measured accurately enough with current production-type sensors. This is either due to a suitable sensor not being feasible due to the cost of implementation, or the fact that the variable simply cannot be measured directly.

The flow of air into the cylinders calculated by the speed-density principle is an example of this. The mass flow of air is calculated by considering the displacement volume of the engine, the engine speed, the throttle position, the pressure and density of the air in the manifold and the volumetric efficiency of the engine which is mapped and stored in a table.

Andersson *et al.* (1999) explain that measured intake manifold pressure is often filtered to reduce the noise from engine pumping and standing waves. A drawback of filtering the pressure signal is the time delay caused by the filter during transients, which results in air-fuel-ratio control errors. Estimators using models of parts of the intake system are therefore often proposed since they can filter the signal and predict manifold pressure during transients.

(Olin and Maloney, 1999) presents the development of a BPE algorithm that is based on a physical description of flow through the engine intake system. The BPE algorithm described has as its basis a static relationship between barometric pressure and the inputs shown in Figure 2.2.

Another example is the exhaust gas temperature, which is also difficult to measure reliably and cost effectively on a production engine. It is however essential for its value to be determined for protection of the catalyst and other exhaust system components under high loads and when running leaner mixtures for fuel economy

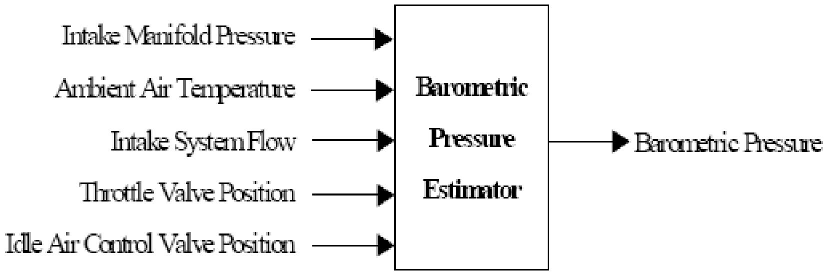


Figure 2.2: Input-output diagram of BPE.(Olin and Maloney, 1999)

and emissions control. This temperature can be estimated by using a model which has the following basic inputs:

- The engine and vehicle speed
- AFR setpoint
- Ignition timing
- Intake air temperature
- Intake air mass flow
- Engine coolant temperature.

Thus a controller using the model based technique relies not only on the physical information obtained from the sensors but produces a comparative picture of the system behaviour, using this to anticipate the likely future. In this way a number of advantages can be provided, at some cost of computation but probably nowadays at a saving in development time. The operation and use of such designs is sometimes not well understood. One example of this type of controller design is sometimes referred to as predictive control, leading to the assumption that there is some absolute prediction involved. In fact the controller makes use of the model to predict the likely behaviour of the system, so that it can achieve an improved controller performance. (Challen and Stobart, 1998)

The formulation of the mathematical model of the powertrain, or as much of it as is necessary, is becoming easier as tools become available for engineers to carry out the analysis and validation of the models. Internal-combustion engines are especially difficult to represent as a model, since numerous relationships are very non-linear. Historically this had led to simplistic representations but experience is now developing a series of approaches that can provide acceptable results. (Challen and Stobart, 1998)

To account for the non-linear nature of the processes being modelled, the models used in current engine controllers still rely on tables of values, often multiplicatives and additives, which are used in the calculation of the desired variable from the various inputs. These variables still have to be determined by extensive testing of the engine, or through SBC which will reduce the amount of actual testing and time required to arrive at an optimised solution.

Torque Model Based Systems

The primary goal of an engine control system is to deliver the torque requested by the driver. In order to achieve this the cylinder charge or amount of air entering the cylinder, the spark timing and the fuel-air-ratio have to be precisely controlled to deliver the required torque in the most efficient manner while protecting the various components from overheating or excess forces and keeping the vehicle emissions within set limits. To complicate matters further, the requested torque also has to be adjusted according to requests from other systems in the vehicle such as the air conditioning unit, traction control system and automatic transmission controller. Torque output also has to be managed in such a way as to prevent inappropriate excitation of the drivetrian, sometimes referred to as drivetrain shunt (McKay *et al.*, 2000),(Gerhardt *et al.*, 1998).

Traditional control systems which handle the subsystems separately require complex prioritisation algorithms to handle different demands from the subsystems at the same time, as illustrated in Figure 2.3. If a change is made to one of the subsystems, or a new system is added, recalibration of these algorithms is required (Heintz *et al.*, 2001).

With the introduction of electronic throttle units it has become possible to coordinate all the subsystems by using a torque based control structure. This means that the engine torque is used as the major interface between the engine control unit and further functionalities inside the vehicle control system. The torque control module of the control unit converts the signal from the accelerator pedal sensor into a driver torque demand and adds or subtracts any other torque requests. This information, together with an estimation of the current engine charge is used by a central actuator demand coordinator to calculate the setpoints of the three groups of torque affecting engine actuators,required to achieve the desired torque output. These three groups are:

- Throttle and wastegate
- Ignition
- Fuel injection.

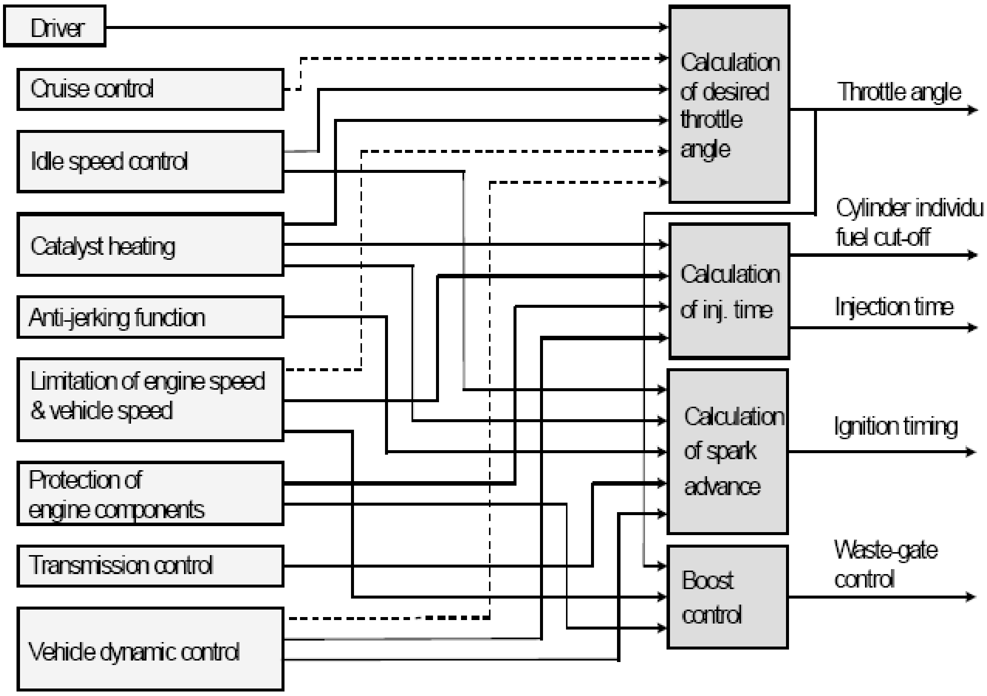


Figure 2.3: Influences on engine torque in a non-torque-based control structure.(Gerhardt *et al.*, 1998)

The setpoints are expressed in terms of efficiencies, where the efficiency is equal to the torque attained at the current actuator settings divided by the torque achievable at optimal actuator settings. The efficiencies are supplied to the modules controlling the actuators which convert it to setpoint values for the actuators (Gerhardt *et al.*, 1998). Figure 2.4 outlines this arrangement.

The estimated engine output torque, which is calculated with a physics-based model, is then matched to this value by adjusting the various actuators. This is performed by the modules controlling the torque affecting engine actuators, namely the throttle and turbo-charger control devices, the ignition timing and the injection timing.

The following section taken from Heintz *et al.* (2001) explains in general how the actuator demands are coordinated in current engine control systems. The air mass flow in the engine, the spark angle and the air-fuel-ratio are the variables that determine the indicated engine torque. The relation between the nominal torque which can be achieved for a particular engine charge and the desired torque can be described as

$$TQ_{des} = TQ_{nom} \times \eta_{spark} \times \eta_{\lambda} \tag{2.1.1}$$

where

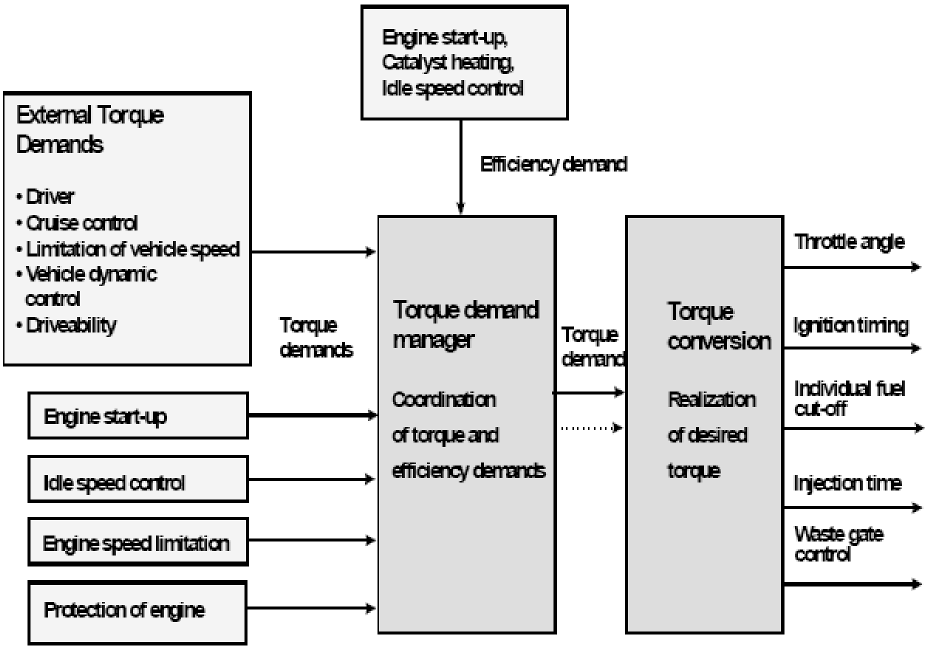


Figure 2.4: Influences on engine torque in a torque-based control structure.(Gerhardt *et al.*, 1998)

TQ_{des} desired torque;
TQ_{nom} nominal torque at optimal setpoints;
η_{spark} spark efficiency;
η_{lambda} air-fuel-ratio efficiency;

The actuator efficiencies can be described by theoretical models. To reduce the calculation time for the algorithms, tables parameterised from engine data recorded on the testbed can be used.

The torque estimation can be applied to coordinate the settings for the engine actuators. Generally it is desirable to satisfy the torque demand completely by adjusting the engine charge to normal conditions for the air-fuel-ratio and the spark timing. To compensate for deviations between the desired torque and the nominal torque the efficiencies for spark and air-fuel-ratio can be modified.

With respect to the coordination of the actuators, two different qualities of torque requests can be distinguished: torque changes that will be needed only for a short time and long term torque demands.

For long term torque demands the engine charge will be adapted in a way that makes the engine run with optimised spark angle and air-fuel-ratio. The setpoint for relative charge is characterised by a map depending on engine speed and desired torque.

Requests for rapid torque changes cannot generally be satisfied by demanding a change of charge. As the throttle needs some time for actuation and the dwell time of the air in the intake manifold causes some further delay (Heywood, 1988), the effect on torque would be too slow in many cases. Therefore, especially if a torque reduction is required, the spark angle or even the fuel-air-ratio have to be changed in order to attain the demanded torque. Once the throttle has been opened to the required setpoint and the airflow into the cylinders has reached the level required to deliver the requested torque the spark angle and fuel-air-ratio will be returned to values closer to their respective optimums.

Figure 2.5 shows the applied approach for determining the desired actuator efficiencies. At the beginning of the calculation it is assumed that the adaption of the torque will be done completely by changing the spark angle. So the quotient of desired and nominal torque can be regarded as a spark efficiency request. The efficiency request is limited by a lower and upper boundary. These boundaries have been introduced to avoid knocking and to keep the exhaust temperature within a permitted range. The boundaries are obtained as output of a spark efficiency table that is supplied with the earliest and the latest possible spark angle. Those spark angle limits are provided by maps depending on engine speed and relative charge. If the requested spark efficiency passes one of the boundaries, adjusting the spark angle within the scope of the boundaries is not sufficient to satisfy the change in torque demand. In this case the lambda efficiency has to be changed from the optimal value. Having calculated the efficiencies the setpoints for ignition and for the air-fuel-ratio can be calculated using inverted efficiency tables. When the actual charge approaches the requested charge the efficiencies can be taken back to nominal values.

Short-term torque changes can only be obtained by varying the spark angle. To ensure that there is enough scope for spark angle variation in different operation modes, such as idle mode, the spark efficiency can be reduced by enhancing the engine charge. If there is the need for a fast increase in torque this can be attained by advancing the spark angle.

Models for Fault Diagnosis

Models are also used for diagnosing faults and according to Struss *et al.* (2000), for engine management control units currently about one half of the software is dedicated to diagnosis and this share is still growing. A typical model based diagnosis system transforms the sensor signals that are available to the standard on-board ECU to a qualitative level and exploits them for detecting and localising faults based on a model of the engine control system.

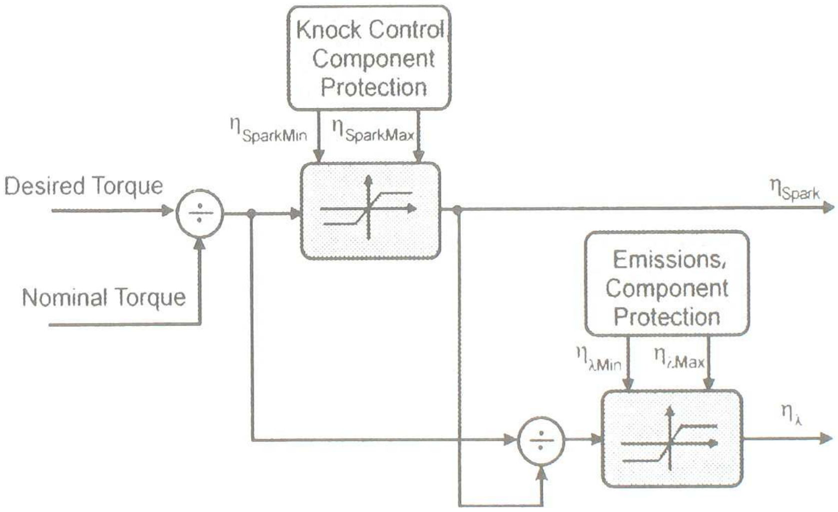


Figure 2.5: Relationship between torque and efficiencies.(Heintz *et al.*, 2001)

This information is used to trigger alarms to warn the driver in case of a failure and to generate fault codes that can be further used during servicing to track down a failure (Ribbens, 2003). For failures which are considered critical, on-board diagnosis also aims at selecting appropriate recovery actions. The built-in recovery actions that will be performed depend on the assumed failure and the expected failure effects and range from minor performance reductions to full engine stop (Bidian *et al.*, 2003). They attempt to take the vehicle back into safe operational conditions (so-called limp-home modes), which allow the driver to reach the closest service centre, albeit at reduced performance levels.

(Andersson, 2001) presents two methods using models to estimate the exhaust pressure of a turbo-charged engine. Knowledge of the exhaust manifold pressure on a turbocharged SI-engine with wastegate is useful for diagnosis of the wastegate, the turbine and the exhaust system. The wastegate controls the power to the turbine and prevents engine turbine destruction by reducing the pressure in the exhaust manifold. Therefore it is crucial for turbine safety to monitor the wastegate. One method to diagnose wastegate operation is to use the exhaust manifold pressure. This is not normally measured due to the high temperatures in the exhaust system and the extra cost of an additional sensor.

2.2 Control of Internal-Combustion Engines

2.2.1 The Process Being Controlled

The following is a basic description of the path that air follows through an engine. The sections following describe the points in this process which have specific applicability to this project in more detail.

Air enters the engine through an air filter. In most production vehicles it then passes through a clean air duct incorporating one or more resonance chambers. These chambers act to silence the induction noise caused by the air being sucked and blown past the rapidly opening intake valves as well as the sound of the air flowing through the turbo-charger compressor. From here the air passes through more ducting into the compressor where it gets compressed using power extracted from the exhaust gas by the turbine. From the compressor it can either first go through an intercooler or just straight through the throttle valve and into the intake plenum.

Utilising an intercooler allows some of the increased internal energy of the compressed air to be passed to a cooling medium, such as the airstream outside the moving vehicle. According to the ideal-gas equation of state, $PV = mRT$, this will allow an increase in the density in the intake manifold, compared to the situation without an intercooler. More oxygen can therefore be inducted into the cylinder during each stroke, thus increasing the amount of fuel that can be efficiently combusted and the power that can be produced by the engine. The lowered intake air temperature with an intercooler also reduces the likelihood of knock.

The throttle valve is used to control the amount of air entering the engine and thus the torque produced. It consists of a round or slightly elliptical plate which sits in a round section of the intake ducting and pivots around a shaft at its centreline. The rotation of this shaft and thus the opening angle of the throttle plate is controlled by the driver through either a cable running directly from the throttle pedal, or through an electrical signal from a position sensor mounted on the throttle pedal which is interpreted by the ECU and used to drive a motor connected to the shaft.

When the intake valve opens as the piston starts moving downwards on the intake stroke, the lower pressure in the cylinder causes the air to flow from the intake plenum through the runners and into the cylinder. The plenum is a volume between the throttle and the intake manifold runners which distributes the throttled air to the runners, while the runners are the intake manifold branches connecting the intake ports of the individual cylinders to the plenum.

The diameter and length of the intake runners strongly influence the volumetric efficiency of an engine. When the inlet valve opens during the intake stroke it causes an expansion wave to be propagated back into the intake runner. On reaching the open end of the runner at the plenum, these expansion waves can be reflected. Thereby causing positive pressure waves to be propagated towards the open inlet valve. If the intake runner is configured correctly the positive pressure wave will cause a locally higher pressure at the open inlet valve (Heywood, 1988), (Ferguson and Kirkpatrick, 2001). This will result in increased charge density in the cylinder and correspondingly higher volumetric efficiency.

The frequency of these waves is directly related to the engine speed. Therefore the increase in volumetric efficiency due to a certain runner configuration will have a maximum value at a specific engine speed. A runner configured for a chosen engine speed are referred to as "tuned". Tuned exhaust runners will create a locally lower pressure when the exhaust valve is open, increasing the exhaust gas outflow from the cylinder.

Manifold tuning is very relevant to naturally aspirated engines due to its direct influence on the shape of the torque curve and therefore the driving characteristics of the vehicle. On turbo-charged engines it is used to ensure proper responsiveness when not operating under boost. Intake manifold runners on turbo-charged engines are therefore relatively long in order to ensure sufficient torque at low engine speeds.

Electro-mechanical fuel injectors are usually mounted in the intake runners, just upstream of the intake valve. They inject fuel into the airstream as the air is flowing into the cylinder. Injecting the fuel into the rapidly moving air helps to mix the air and fuel properly which is essential for good combustion.

After the piston reaches Bottom Dead Centre (BDC) the intake valve closes and the cylinder starts moving upwards on the compression stroke. At a certain crank angle before the cylinder reaches Top Dead Centre (TDC) a spark is fired by a spark plug inside the combustion chamber, which ignites the compressed charge of air and fuel. A turbulent flame developes and propagates across the combustion chamber, adding heat to the mixture and extinguishes at the combustion chamber wall while the piston moves downwards. The combustion continues as the piston moves downwards on the expansion stroke after TDC. Work is done on the piston by the expanding gases (Bergström, 2003).

An optimum spark timing exists which, for a given mass of fuel and air in the cylinder, gives maximum torque. This timing is known as the Maximum Brake Torque (MBT) timing. More advanced or retarded timing will result in lower torque output (Heywood, 1988). Timing that is too advanced will cause the increase in pressure due to combustion to work against the piston on the compression stroke. If the timing is retarded too much it will decrease the peak pressure on the expansion stroke and thereby the work done on the piston by the expanding gases.

Between 40 to 60° before the piston reaches BDC again the exhaust valve starts to open to allow the pressure difference between the cylinder and the exhaust manifold to discharge the burned cylinder gasses. After BDC the remaining gases are forced out by the upwards moving piston. These two processes are termed *blowdown* and *displacement* respectively (Heywood, 1988). In automotive engines it is typical for the exhaust valve to stay open during the initial portion of the intake valve opening period. The crankshaft angle over which both valves are open is called the valve overlap. It serves the dual purpose of using the inertia of the gasses in the exhaust and intake systems to assist with the intake process during Wide Open Throttle (WOT) conditions, thus improving volumetric efficiency and allowing small amounts of exhaust gas to be recirculated at part throttle conditions. Recirculating the exhaust gases has the effect of lowering the peak combustion temperature which in turn lowers the amount of Nitrogen Oxides (NO_x) in the exhaust gas since NO_x formation increases with combustion temperature.

Controlled Exhaust Gas Recirculation (EGR) is achieved in modern engines by utilising an ECU controlled valve to allow precisely controlled amounts of exhaust gas to flow from the exhaust manifold to the intake manifold. Up to a certain percentage of EGR, the increase in hydrocarbon emissions and Brake Specific Fuel Consumption (BSFC) is significantly smaller than the decrease in NO_x and is thus advantageous to emissions control (Ribbens, 2003).

The gases then flow through the exhaust manifold and into the turbo-charger turbine where a portion of the energy remaining in the gases is converted to shaft torque to power the compressor. The remaining energy consists of the blow-down energy from the continued expansion of gasses after the opening of the exhaust valve at the end of the expansion stroke and the work done by the piston in displacing the remaining gasses on the exhaust stroke (Watson and Janota, 1984).

After exiting the turbine, the gas flows through a catalytic converter which uses a catalytic reaction to further reduce the NO_x in the exhaust stream and reduce the harmful emissions of the combustion process. It then passes through one or more silencers and then out into the atmosphere.

2.2.2 The Need For Accurate Control

Emissions

- The main driver behind the development of electronic engine management systems was the emissions regulations that came into play in the 1970's. Until then all cars used carburettors to mix the air and fuel before it entered the cylinders. However carburettors cannot control the air fuel mixture accurately enough, which necessitated the development of electronically controlled injectors to inject the fuel in precise amounts. To meet even higher emissions standards in the following decades it became necessary to fit three-way catalytic converters to the exhaust in order to further reduce the levels of NO_x, HC and CO in the exhaust.

These catalytic converters operate most efficiently with a stoichiometric air-fuel-ratio, which represents the ratio with just enough oxygen for conversion of all the fuel into completely oxidized products (Heywood, 1988). Consistently achieving this ratio places a high demand for accuracy on the fuel control system. Closed-loop fueling, utilising feedback from an oxygen sensor in the exhaust system before the catalytic converter, is employed to ensure this demand is met.

To ignite the mixture of fuel and air a system consisting of a coil, breaker point, distributor and spark plugs was used. This system passed a low voltage current through the primary winding of the ignition coil which created a magnetic field between the primary and secondary windings of the coil. At the crank angle where the spark was required in the cylinder, the current through the primary winding was interrupted by a mechanical switch called a breaker point which was actuated by a cam connected to the engine camshaft. This interruption caused a rapid decay of the magnetic field which induces a high voltage pulse in the secondary winding. This pulse was then distributed to the spark plug of the next cylinder to fire through the distributor by means of a rotor which was mechanically driven at half the crankshaft speed. The resulting spark then initiated combustion in the cylinder which was at the end of the compression stroke. The ignition timing was determined by the position of the breaker point cam in relation to the engine crankshaft and was adjusted for different engine load conditions by a vacuum mechanism connected to the intake manifold. The basic timing had to be set by rotating the housing of the assembly by hand.

The closer the ignition timing is to the TDC position the less time the gas has to burn before the exhaust valve opens and thus the higher the temperature of the gas exiting the cylinder will be. This also has an effect on the composition of the exhaust gas. For this reason accurate ignition timing was also required which led to the development of distributor-less electronic ignition systems with transistors and multiple coils replacing the breaker point and distributor, that offers improved adjustability and accuracy over the systems used until then.

Performance and Component Protection

- The amount of work performed during the expansion stroke, for a certain fixed cylinder and compression ratio, is primarily affected by two variables. These are the air-fuel-ratio of the charge in the cylinder and the ignition timing. To ensure safe, reliable and efficient engine performance these variables have to be closely controlled.

As stated above, the best air-fuel-ratio for catalyst efficiency and fuel economy is stoichiometric or approximately 14.7 parts air for every part fuel, depending on the composition of the fuel. This is called lambda 1 since lambda is calculated as the actual air-fuel-ratio divided by the stoichiometric ratio. The air-fuel-ratio for highest power output is slightly richer (more fuel) at a lambda value of approximately 0.9 to 0.92. This richer ratio ensures that all the inducted air is used to extract the maximum energy from the available fuel. Not all of the fuel will be burned which means that this ratio is not optimal for low emissions or good fuel economy.

To ensure that as much of the combustion energy as possible is used to perform useful work, the ignition timing must be adjusted as close to the MBT point described in Section 2.2.1 as possible. There are two reasons for not setting the timing to the MBT point. Firstly for the purpose of emissions control and secondly to avoid an abnormal combustion condition called knock, which can be described as follows:

Normally, under homogenous charge conditions, the spark will fire and a flame front will proceed from the spark plug into the unburned gas like a spherical wave with a ragged surface because of turbulence (Ferguson and Kirkpatrick, 2001). When the spark is advanced too far, or the intake pressure is too high, the temperature of the unburned gas ahead of the flame front will exceed the autoignition point. The remaining unburned gas will then start to burn rapidly from several autoignition sites, causing rapid fluctuations in pressure. This is called knock and can sometimes be heard as an audible pinging sound outside the engine. Knock can be aggravated or even caused by glowing points in the combustion chamber such as sharp corners or even the spark plug electrode. These can heat the cylinder charge and cause it to ignite prematurely, having the same effect as advancing the spark timing too far. The susceptibility of an engine to knock is dependant on the operating conditions, the type of fuel used and the design of the engine. The compression ratio has a strong influence since the higher the compression ratio, the higher the temperature of the mixture at the end of the compression stroke. The compression ratio of a spark ignition engine is therefore knock-limited.

The rapid pressure fluctuations associated with knock breaks down the cylinder thermal boundary layers, exposing the components to higher combustion temperatures which results in surface erosion and failure (Ferguson and Kirkpatrick, 2001). The high rate of pressure increase can also damage the piston rings and the ring lands in the pistons. It is therefore essential to accurately control the ignition timing of the engine to get the maximum power and efficiency without damaging the engine. Most modern engine management systems utilise a knock sensor to determine when the engine is knocking and then retard the timing temporarily until the knocking subsides. Systems without knock sensors have to be calibrated more conservatively and therefore have a performance disadvantage when compared to systems with knock sensors, which can run at more advanced timing when the engine is not knocking.

2.2.3 Engine Variables Measured for the Control Process

Temperatures

The following temperatures are measured by the engine control system:

- Coolant and engine oil temperature are measured to determine when the engine has reached its correct operating temperature. The spark timing and fuel injection duration are calibrated in terms of coolant and intake manifold air temperature to allow for the different combustion characteristics of the fuel air mixture at varying temperatures and the different lubrication characteristics of the oil.

- Intake air temperature is measured to determine the density of the air entering the engine. Ignition and fuelling are also adjusted in terms of intake air temperature to account for the varying combustion characteristics at different charge temperatures. In a turbo-charged engine the temperature after the compressor, or intercooler if fitted, will be measured due to the change in temperature after the compressor.

Pressures

- Air intake pressure is measured and used together with the air intake temperature and engine speed to calculate the air mass flow into the engine by the speed-density method explained below. Again, if the engine is turbo-charged the post-compressor pressure is measured.

- Oil pressure is measured and used to ensure that the engine contains sufficient oil at all times. If the oil level drops below the pickup of the oil pump due to oil usage or damage to the sump or oil galleries the oil pressure will drop and the oil pressure signal will notify the driver and/or the ECU to stop the engine before serious damage occurs.

Speed and Position

- Crankshaft angular speed and position are measured to determine the engine speed as well as the positions of the pistons in the cylinders.
- Camshaft angular position is used together with the TDC marker on the crankshaft to determine the position of each cylinder in relation to TDC.
- Vehicle speed is normally calculated from the speed of the gearbox output shaft. This signal is used by the ECU for various functions including; to indicate the speed to the driver through the speedometer, as an indication of the airflow over the exhaust when calculating the exhaust temperature,and to calculate the current gear ratio. The gear ratio information is used as a reference to ensure good drivability as well as component protection.

Combustion Elements

- Airflow - There are two methods for determining the mass of air flowing into the engine. One is by using a mass flow sensor which has a film or wire element placed in the intake air stream. This element is wired to be one of the resistors of a Wheatstone bridge and heated to a constant temperature above the inlet air. The inlet air cools down the element which changes its resistance and causes an imbalance in the bridge circuit, thereby causing an input voltage to the amplifier. This voltage is then converted to air mass flow using a calibration curve. The other method is to calculate the air mass flow from the engine speed and measured intake air pressure and temperature. This is called the speed-density method and has been mentioned previously. It consists of multiplying the rate at which volume is displaced by the pistons at the current engine speed, by the volumetric efficiency of the engine at the same operating point, see Equation 2.2.1.

  $$\dot{V} = \left(\frac{N}{60}\right)\left(\frac{V_d}{2}\right)\eta_{vol} \tag{2.2.1}$$

  where

  $\dot{V}$ is the volume flowrate;
  V_d is the engine displacement;
  N is the engine speed;
  η_{vol} is the volumetric efficiency read from a table of values.

This volume flow is then multiplied by the inlet density of the air to determine the mass air flow rate.The inlet density is calculated by using the ideal gas law $PV = mRT$ as shown in Equation 2.2.2.

$$\rho_{ai} = \frac{p_i}{RT_i} \tag{2.2.2}$$

where

ρ_{ai} is inlet air density;

p_i is the inlet air pressure;

T_i is the inlet air temperature.

Volumetric efficiency, η_{vol}, is a measure of the effectiveness of the engine to induct fresh air and is defined as the ratio of actual volume flow rate of air entering the cylinders, $\frac{\dot{m}_{ac}}{\rho_{ai}}$ and the rate at which volume is displaced by the piston, $\frac{V_d N}{2\times 60}$ (Bergström, 2003). The factor 2 in the denominator arises from the fact that a four stroke engine only inducts fresh air in each cylinder every second revolution. Using the above, the expression for the volumetric efficiency becomes:

$$\eta_{vol} = \frac{2\dot{m}_{ac}}{\rho_{ai} V_d N/60} \tag{2.2.3}$$

- Air-fuel-ratio. The air-fuel-ratio of the engine is determined by an Exhaust Gas Oxygen Sensor (EGO) sensor mounted in the exhaust manifold or down-pipe. The sensors essentially consist of an assembly of zirconium dioxide or titanium dioxide sandwiched between two platinum electrodes. One side of this assembly is exposed to the exhaust gas and the other to the atmosphere. The side exposed to the atmosphere is exposed to a much higher concentration of oxygen ions than the exhaust side. This difference in the amount of oxygen ions between the two sides and the fact that oxygen ions are negatively charged due to having two excess electrons, causes the atmospheric side to become electrically more negative than the exhaust side. An electric field therefore exists across the zirconium dioxide or titanium dioxide sandwich material and a voltage results. The polarity of this voltage is positive on the exhaust side and negative on the atmospheric side. The magnitude of the voltage depends on the concentration of oxygen in the exhaust gas (Ribbens, 2003).

  Two types of EGOsensors exist, namely the linear and switching types. Most vehicles are fitted with switching type EGO sensors that output a low voltage for lean mixtures and a higher voltage for rich mixtures. The ECU uses this information to control the air-fuel-mixture and maintain it at stoichiometric. The point where the output switches from high to low differs slightly from the low to high switch-point and the difference between the two output values increases with temperature. These effects have to be compensated for by the ECU in order to achieve accurate control of the air-fuel ratio. The sensor needs to be heated to above 300°C for the difference in outputs to be large enough for control purposes. In the past this has been compensated for by using open-loop fuel control until the sensor has reached its operating temperature but this arrangement is not able to meet the stricter emissions regulations that came into play during the $1990's$. To overcome this problem, the sensors are fitted with a section of resistance material and are heated to the correct operating temperature with power from the car battery. These sensors are called Heated Exhaust Gas Oxygen Sensor (HEGO) sensors. Linear EGO sensors are more expensive and are used mainly during the ECU calibration process when information about the exact air-fuel-ratio is required, although an increasing number of new cars are being fitted with linear sensors to allow better control of the air-fuel-ratio and thus better emissions control.

The control system also uses a number of indicators to determine the operating state of the vehicle, these include the following:

- Brakes engaged
- Wide open throttle
- Closed throttle
- Air conditioner compressor clutch engaged

2.2.4 Actuators Used for the Control Process

- Fuel injectors - These are essentially solenoid-operated valves situated between a pressure regulated supply of fuel and either the intake manifold or the combustion chamber of the engine, depending on whether the engine utilises indirect or direct injection respectively. With no current flowing though the solenoid, the valve is held closed by a spring. When the current is switched the valve opens and fuel sprays into either the manifold or the combustion chamber. For a given regulated pressure and injector nozzle geometry, the fuel flow rate through the injector is constant and therefore the amount of fuel is proportional to the time that the valve is held open by the current flowing through the solenoid coil. To control the amount of fuel entering the engine, the current is pulsed on and off by the ECU.

- Spark plugs - The spark plug is screwed into the cylinder head so that its tip protrudes into the combustion chamber. There are two electrodes on the tip, one connected to the engine ground and the other to the distributor through a spark plug lead. When the high voltage pulse from the coil reaches the spark plug through the distributor, a spark is created between the two electrodes which then ignites the cylinder charge. This pulse voltage reaches levels as high as $20kV$ to $40kV$. The gap between the electrodes is important to ensure the correct spark properties and is specified for each engine. Figure 2.6 shows the typical configuration of a spark plug.

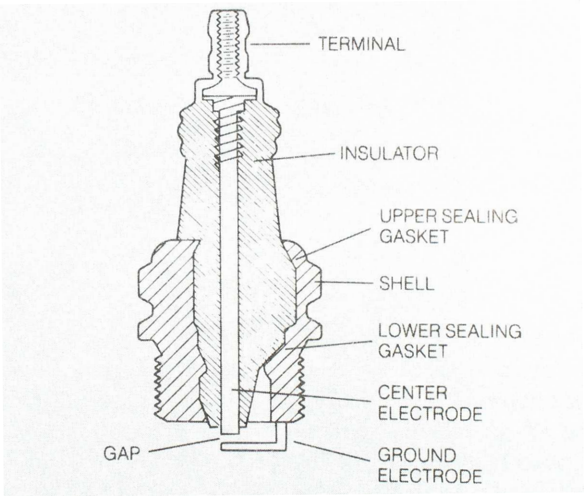


Figure 2.6: Spark plug configuration.(Ribbens, 2003)

- Turbo-charger Controller - There are currently two main types of turbo-charger in common use. The one type utilises a wastegate to control the amount of energy available to drive the compressor and the other makes use of variable guide vanes.

  A wastegate is a valve situated upstream of the turbo-charger turbine, which can allow some of the exhaust gas to bypass the turbine in order to reduce the energy available to the turbine and therefore to the compressor. This valve is called an internal wastegate when it is housed in the turbine housing, as is most commonly the case, and an external wastegate when it is a separate component mounted on a section of exhaust pipe teeing off from the manifold upstream of the turbine housing.

Since not all the the exhaust gas has to pass through the turbine and no more air than the requirement of the engine passes through the compressor, a smaller turbo-charger can be used than would be required if passing all the air through the turbine at full load and high speed. A smaller turbo-charger is better able to provide the boost required at low speeds and offers reduced lag due to its lower inertia (Watson and Janota, 1984).

Both types of wastegate have actuators with diaphragms which are connected to the compressor outlet on the one side and open to the atmosphere on the other. Once the post compressor pressure reaches a pre-set level the actuator starts opening the wastegate.

To control the wastegate actuator a solenoid valve is placed in the line connecting the compressor outlet and the actuator diaphragm. This valve can vent some of the air to the atmosphere and so decrease the pressure acting on the diaphragm. By increasing the duty cycle of the pulse-width-modulated signal powering the solenoid, the wastegate can be held closed when it would normally have opened if it were directly connected (Barr, 2001).

This configuration has the safety feature that if the electronic signal controlling the normally closed solenoid is disrupted, the compressor outlet pressure will drop to the lowest possible level and prevent possible damage to the engine. The gas going through an internal wastegate joins the rest of the exhaust gas just after the turbine inside of the turbine housing. In the case of an external unit the wastegate exhaust is normally plumbed back into the main exhaust after the turbo-charger, or simply vented to the atmosphere in certain motorsport applications.

Turbo-chargers with variable guide vanes have vanes inside the turbine inlet nozzle which can be adjusted between various degrees of convergence. When adjusted for higher convergence the blades effectively form a small nozzle over which there is a large pressure drop from the exhaust inlet to the turbine blades. The higher the pressure drop, the higher the amount of kinetic energy available to spin the turbine rotor. This enables the turbo-charger to speed up faster at low engine speeds enabling good low speed engine performance. The angle at which the gas enters the rotor is also closer to the optimum which increases the efficiency of the turbine (Sayers, 1990). Opening up the vanes will adjust the exhaust gas energy utilisation to suit medium to high engine speeds, while preventing the turbo-charger from over-speeding (Watson and Janota, 1984).

- Throttle - The throttle plate is a round flap situated at the entry of the intake plenum which controls the amount of air entering the engine. The throttle has until recently been connected directly to the throttle pedal but in order to achieve greater control of drivability, traction and engine emissions, manufacturers are increasingly moving towards electronically controlled throttles. These systems utilise a sensor on the throttle pedal to determine the driver demand for torque which the ECU then utilises to determine the best throttle opening. This allows the ECU to consider many factors, including the available traction and the possibility of the vehicle jerking due to a rapid increase in torque, to deliver a smoother and safer ride. An electronic actuator fitted to the throttle body then adjusts the throttle opening according to commands from the ECU.

2.2.5 Electronic Ignition Control

Figure 2.7 outlines the functioning of the ignition section of a modern torque-based control system with electronic throttle and knock control. It illustrates how the engine speed and mass air flow signals are used as references to find the basic ignition angle for the current operating point. From there on a series of correction factors and requests from other control subsystems are used to calculate the actual ignition angle that reaches the spark plugs.

The values in the basic ignition map at the top of the figure would represent values as close to MBT as possible. One of the goals of the AutoCal program developed during this project is to determine suitable values for this map. When the engine being controlled is running at steady state conditions representing the conditions under which mapping was performed, these map values would be used as the actual ignition values at the spark plugs. The other inputs on the path between this map and the actual timing sent to the spark plugs serve to compensate for deviations from this steady state.

The main functions of the other inputs are to ensure smooth starting and running, good drivability, low emissions and protection from damaging conditions such as knock and overheating. A large number of additional maps, as well as models are utilised to determine the changes to the basic ignition angle requested by these inputs (Heintz *et al.*, 2001).

Both short and long term changes can be made. A rapid increase in spark advance to increase torque and minimise a fluctuation in engine speed as the airconditioning is switched on, represents a short term change. Longer term changes are represented by addaption values which are calculated to prevent a consistently occuring unwanted situation from persisting. An example of this is when a lower octane fuel is used and the knock detection system requests a permanent reduction in the basic timing to prevent knock. This reduction will be removed if a change is made to the fuel octane level (Ribbens, 2003).

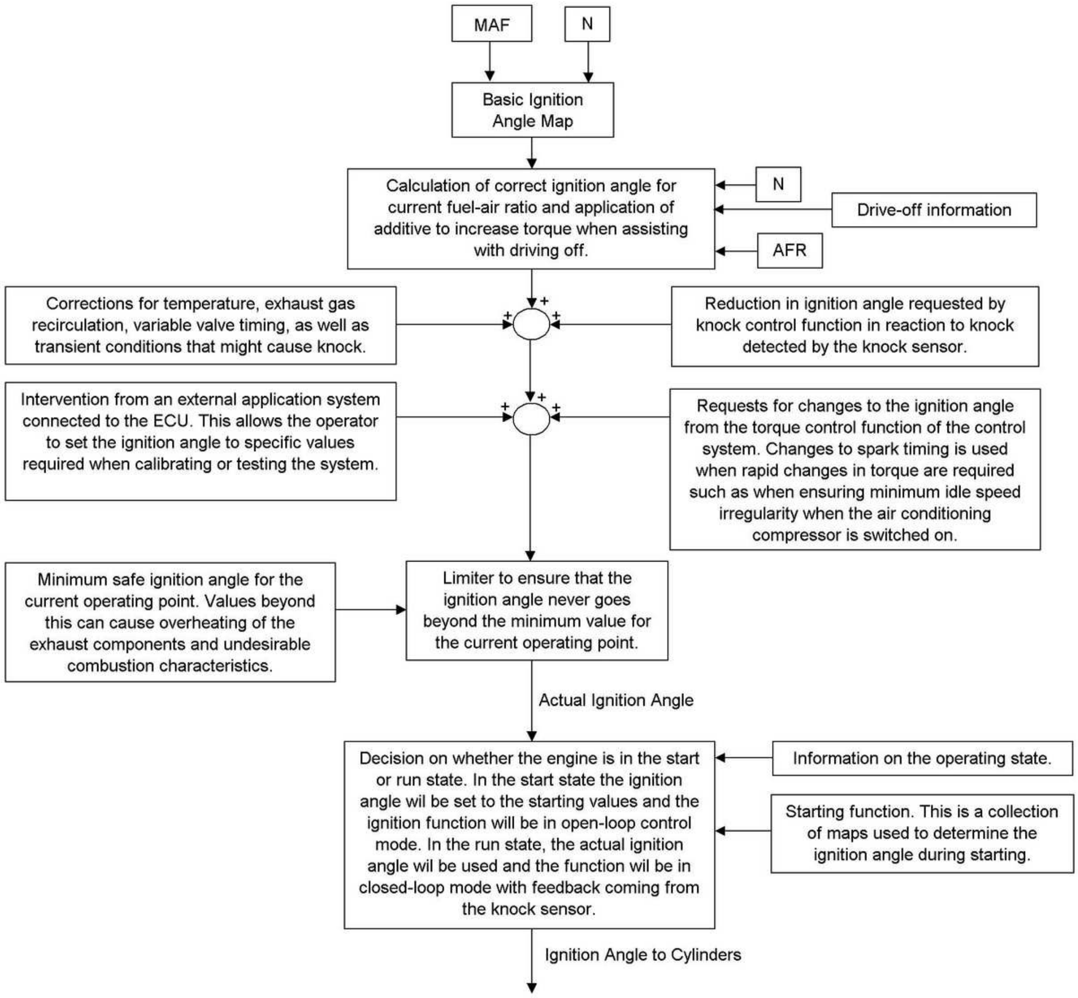


Figure 2.7: Simplified flow diagram of the ignition function of a modern torque-based engine control system.

2.3 Simulation Based Calibration

Traditional calibration methods use a combination of engine dynamometer and vehicle testing but pressure to reduce powertrain development cost and time is driving development of more advanced calibration techniques. In addition, modern engines feature new technology, such as variable valve actuation, that is necessary to improve fuel economy, performance and emissions. This introduces a greater level of system complexity and greatly increases test requirements to achieve successful calibrations.

(Rask and Snellnau, 2004) addressed these problems by developing new simulation tools and procedures to rapidly generate optimised calibration maps. The objective of their work was to reduce calibration effort while fully realizing the potential benefit from advanced engine technology. The developed procedure utilizes GT Power engine simulation software and engine models validated through limited dynamometer testing.

A front end to GT Power was written to automatically call GT Power executables and produce the calibration dataset. Several methods were used to accelerate the simulation process. Calibrations were optimised using an additional software tool that includes a weighted optimization scheme. User-defined constraints were applied during optimisation for cam phaser position.

(Osborne, 1999) applied an engine cycle simulation code (WAVE) in co-simulation with a control and dynamics modelling package (Matlab Simulink) to provide total vehicle system modelling capability. WAVE was used to model and simulate the engine while Simulink was used for the drive-line model. The ECU was also modelled in Simulink as a representation of the appropriate control strategy.

These models were incorporated in a analytical calibration tool with the aim of the reducing the amount of dynamometer-based and vehicle calibration. The analytical ECU calibration process was successfully applied and has delivered real-world performance benefits in vehicle tests.

Chapter 3

Simulation Based Calibration System - AutoCal

3.1 Introduction

In order to answer the research questions posed at the start of this project, it was required to develop a software tool that could achieve the objectives of the project. These objectives are repeated below:

1. To create and validate a model of a turbo-charged internal-combustion spark-ignition engine using a suitable simulation program.
2. To develop a technique for using a software program to find the most suited calibration values of the turbo-charger boost and ignition maps for a desired torque curve, subject to pre-determined constraints such as air fuel ratio, maximum intake air temperature, maximum exhaust temperature and minimum turbo-charger efficiency, from a database of simulated values.
3. To simulate the performance of the engine with the calibration values from the software program and compare the results with the desired torque curve.

This chapter describes the AutoCal program that was created to achieve these objectives. AutoCal automates the process of running multiple simulations and evaluating the resulting data to find the combinations of input variables that will deliver a desired output. It also offers convenient facilities for manually analysing the simulation results and examining the effect of different combinations of input variables than those suggested by the program.

The program works on the principle of creating a database of simulated data by running a user-defined full-factorial experiment and then post processing the database. This once-off creation of a database reflecting the effect of a large variety of calibration values allows rapid matching of different torque curves by avoiding the process of having to run a simulation to match the torque values every time a different torque curve needs to be calibrated, as would be the case with a program that searches for the correct calibration values while simulating. With the AutoCal configuration the simulation of various calibration values can be done during non-working hours such as overnight and the results then examined inter-actively in a time-efficient manner. Once a set of calibration values has been chosen, only those values have to be simulated again to confirm their accuracy. This also allows changing of constraints and optimisation criteria without having to re-simulate.

3.2 The Design Problem

The entire project was treated as a design problem and approached in the familiar way of identifying the clients and their needs, developing specifications and concepts and developing the chosen concepts into the final product. This section presents the scope defined for the AutoCal program from the project objectives. The next section discusses the functional development of the AutoCal program as well as the concepts chosen to fulfill the functions.

3.2.1 Scope

1. It was decided early on in the project that Ricardo WAVE would be used for engine simulation and Matlab for data processing and as a user interface. This was due to the fact that it was the only software offering these capabilities available to the student.
2. In accordance with the second objective of the project all work was limited to the full-load operating region.
3. Minimum engine control inputs to simulation model:
    - Wastegate position
    - Ignition advance angle
4. Minimum outputs to be simulated:
    - Torque
    - Power
    - SFC
    - Exhaust gas temperature
    - Compressor efficiency

- Compressor Shaft Speed
- Knock intensity

3.3 Chosen Concepts

In order to properly understand the problem and to develop a list of tasks to perform, a functional analysis was performed. Eight main functions were identified and broken down further into subfunctions. The identified tasks consisted of developing concepts to fulfil each of these subfunctions and then implementing these concepts to create the AutoCal program.

This section presents the functional flow diagrams as well as the chosen concepts. The main functions are outlined first and then the chosen concepts are discussed together with the groups of subfunctions.

3.3.1 Main Functions

The following eight main functions were defined for the AutoCal program:

1. Define the simulation setup.
2. Define the desired torque curve and operating constraints.
3. Perform simulations to find calibration values.
4. Select calibration values to achieve the desired torque curve.
5. Compare the results with the desired torque curve.
6. Allow user adjustment of the calibration values.
7. Confirm the results of the calibration selection process.
8. Store results and exit.

The functional flow diagram for these functions is shown in Figure 3.1.

3.3.2 Sub Functions

Each of the main functions described above will now be examined in more detail. The concepts chosen to fulfill the functions as well as the implementation of these concepts will also be discussed.

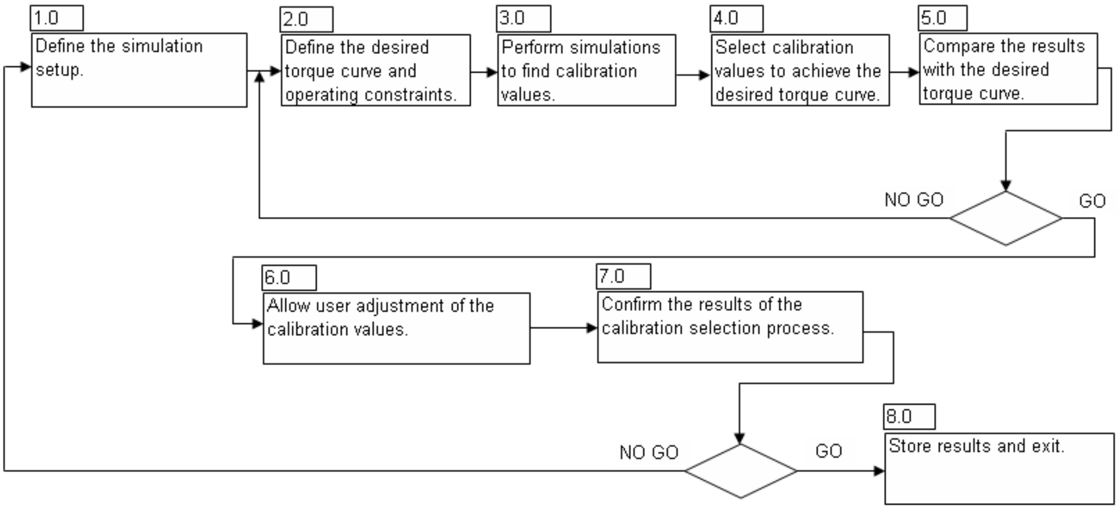


Figure 3.1: The eight main functions of the AutoCal program.

Function 1 **- Define the simulation setup.**

A major requirement of answering the research questions was to have accurate simulated engine data. To create this data a model of the test engine had to be programmed and validated. To facilitate the simulation of the effect of different input variable values the user must be able to define these values. Figure 3.2 shows the various subfunctions defined under Function 1 to achieve these requirements.

The concept chosen to achieve these functions was a combination of Ricardo WAVE to model the engine and Matlab to handle the user interface and manage the resulting database of simulation results. The use of WAVE and Matlab was decided on at the start of the project as stated above.

The GUI facility in Matlab was used to create the interface shown in Figure 3.3. This GUI and its associated Matlab code fulfills all the subfunctions of Function 1 excepting Subfunctions 1.1.1, 1.1.3 and 1.1.4. These subfunctions are handled by the engine modeler using Ricardo WAVE as well as another Matlab application created for evaluating and calibrating the WAVE models. This application is presented in Appendix B.4. The GUI also fulfils some subfunctions from the other main functions. These will be explained under their respective main functions.

The treatment of Subfunctions 1.1.1, 1.1.3 and 1.1.4 for this project is discussed in detail in Appendices A and B. A test engine was assembled and installed in an engine test facility. It was then tested under various operating conditions in order to gather data for calibrating the WAVE engine model used during the rest of the project. As discussed in Appendix B, the calibration process was only completed until the main outputs were accurate to within 10% to 20% of the test data. This was mainly due to time constraints and uncertainty about the accuracy of the available valve flow coefficients and turbo-charger flow maps. This level of accuracy was was deemed sufficient for the development and evaluation of the AutoCal program.

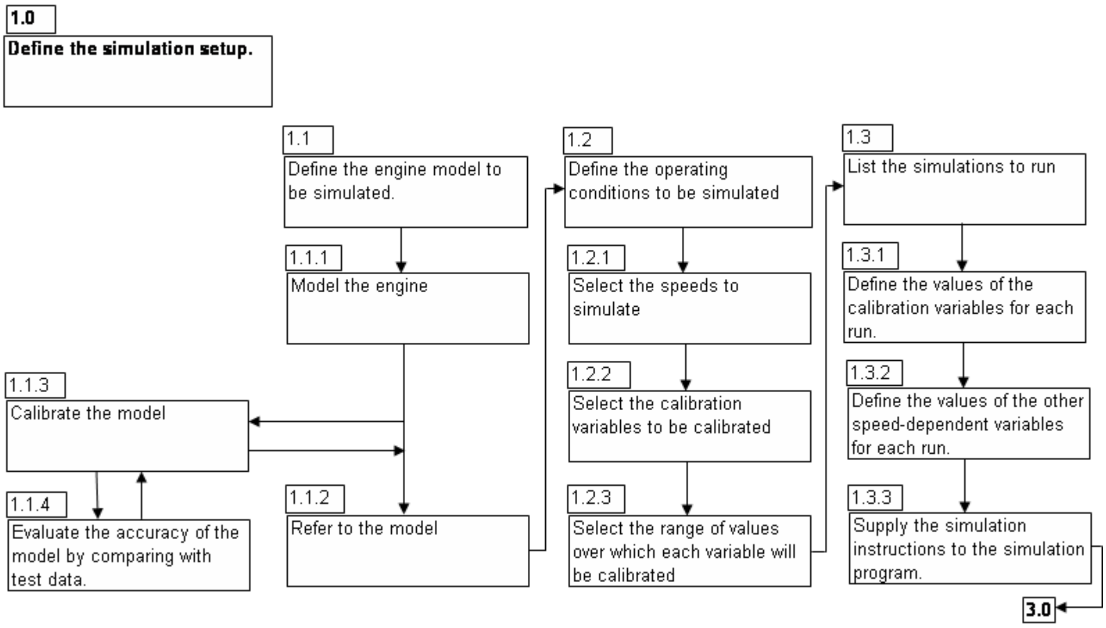
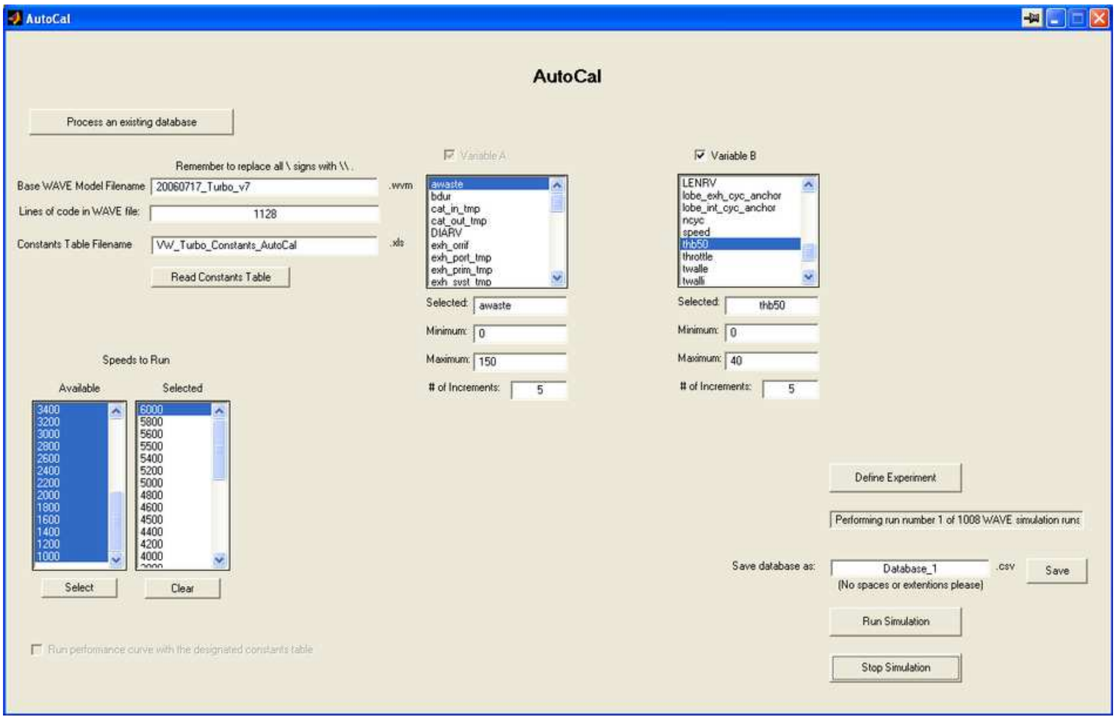


Figure 3.2: Related subfunctions of Function 1 - Define the simulation setup.

Figure 3.4 is a flow diagram depicting the flow of actions required when using the GUI in Figure 3.3 to create a database of simulation results. From this screen the simulations are configured and started. It also offers the option of proceeding directly to the data processing function if a previously created database is to be used for finding calibration values. A database consisting of actual test data can also be processed using this option, in the case of accurate simulation not being possible.

The user provides the names of the WAVE model and the constants table to be used during simulation. The constants table contains the speed dependent input variables for the WAVE model. For this project the calibration variables for the control of the ignition timing and turbo-charger boost pressure were included in the constants table.

After defining the model and inputs the user can choose between creating a database by simulating a series of different calibration variables, or simulating a performance curve with the inputs defined in the constants table. The performance curve facility is used when performing Function 7. It also acts as a convenient facility for simulating the performance curve of any WAVE engine model.



Base WAVE Model Filename 20060717_Turbo_v7 .wvm

Lines of code in WAVE file: 1128

Constants Table Filename VW_Turbo_Constants_AutoCal .xls

☑ Variable A

Selected: awaste

Minimum: 0

Maximum: 150

# of Increments: 5

☑ Variable B

Selected: thb50

Minimum: 0

Maximum: 40

# of Increments: 5

Performing run number 1 of 1008 WAVE simulation runs

Save database as: Database_1 .csv

☐ Run performance curve with the designated constants table

Figure 3.3: The AutoCal user interface used when setting up the simulations for creating the database.

To create a database of calibration values the user first selects the speeds to be simulated. The speeds are chosen from the list of speed points defined in the chosen constants table. This list is displayed in a listbox in the GUI and a second listbox is used to display the speeds that have been selected. This avoids the possibility of speeds being selected that are not defined in the constants table.

The user then has to select one or two calibration variables as well as their values to be simulated. The variables are chosen from the list of inputs in the designated constants table. The user can select the range over which each variable must be simulated, as well as the number of increments across the range. Choosing a higher number of increments increases the number of simulations to be run but also increases the accuracy of the interpolation that will be used to find the values corresponding to the desired torque curve. This interpolation process is described in more detail in the description of Function 4. A suitable compromise depending on time available and desired accuracy will therefore have to be decided on. The user inputs are stored in the Matlab handles structure which makes them available to other parts of the AutoCal code.

Once the variables are defined, the list of simulations to be run is defined by AutoCal. The list is created by defining a full-factorial experiment which lists every possible combination of the fixed variables in the constants table and the variables

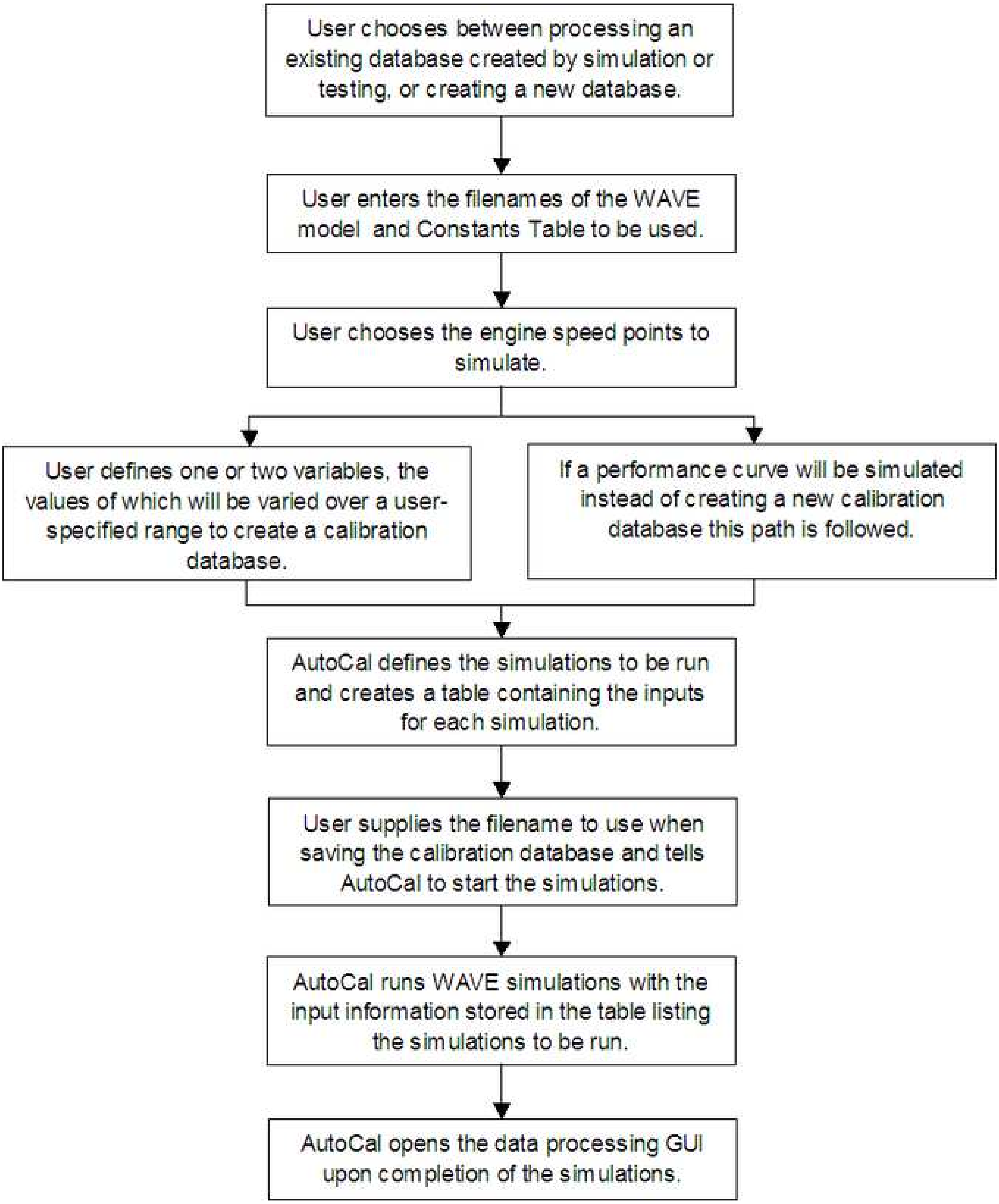


Figure 3.4: The flow of actions followed to create a calibration database.

chosen by the user. This is done for each of the selected speed points. This code is initiated by clicking on the **Define Experiment** button. The list of simulations to be performed is stored in a file called Runs.csv.

After defining the experiment the number of simulation runs that will be performed will be displayed. The user must then supply a name for the database to be created and initiate the process by clicking the **Run Simulation** button. While the simulations are running the progress is displayed to give an indication of the time remaining up to completion. On completion of the listed runs the data will be stored in the Comma Separated Value (csv) file requested by the user and the processing interface opened.

The **Stop Simulation** button was programmed to give the user the option of stopping the simulation at any time, upon which the results of simulations performed up to that point will be stored in the csv file requested by the user. The data processing interface will then be automatically opened. The csv file would have to be opened in Microsoft Excel and manually manipulated to ensure that it only contains data from completed speed points. This might mean that rows of data at the bottom of the file from a speed point that was stopped before completion will have to be deleted. The file can then be used for processing the speed points that were completed. Data from other speed points that were simulated during a separate session can also be pasted on to create a larger database. This procedure can also be used to replace the data for a speed point that was re-simulated, or to add speed points into a file to increase the resolution of a performance curve.

Function 2 - Define the desired torque curve and operating constraints.

Answering the second research question posed in the introduction requires the definition of the desired torque curve as well as the operating constraints. The second of the main functions was broken down into the subfunctions shown in Figure 3.5 in order to clarify the required tasks to achieve this. To fulfill these and other functions the GUI shown in Figure 3.6 was created. The GUI fulfills Function 2 by receiving inputs consisting of the name of the database to be processed, the name of the desired performance curve, which operating constraints to apply and the values of these constraints. It then stores this data in the Matlab handles structure for use by the code fulfilling the other functions.

The constraints listed below were used to develop and demonstrate Functions 2.3 and 2.4. These constraints, especially those relating to the turbo-charger, have been simplified to ensure a clear demonstration. This list can be expanded to include other operating variables, as well as more complex methods of determining the constraining values. An example of this would be a calculation of the compressor

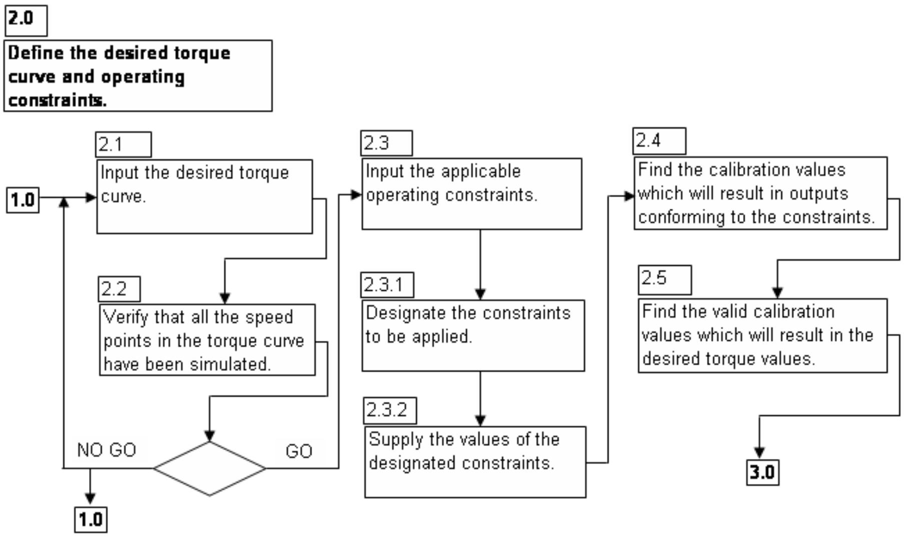
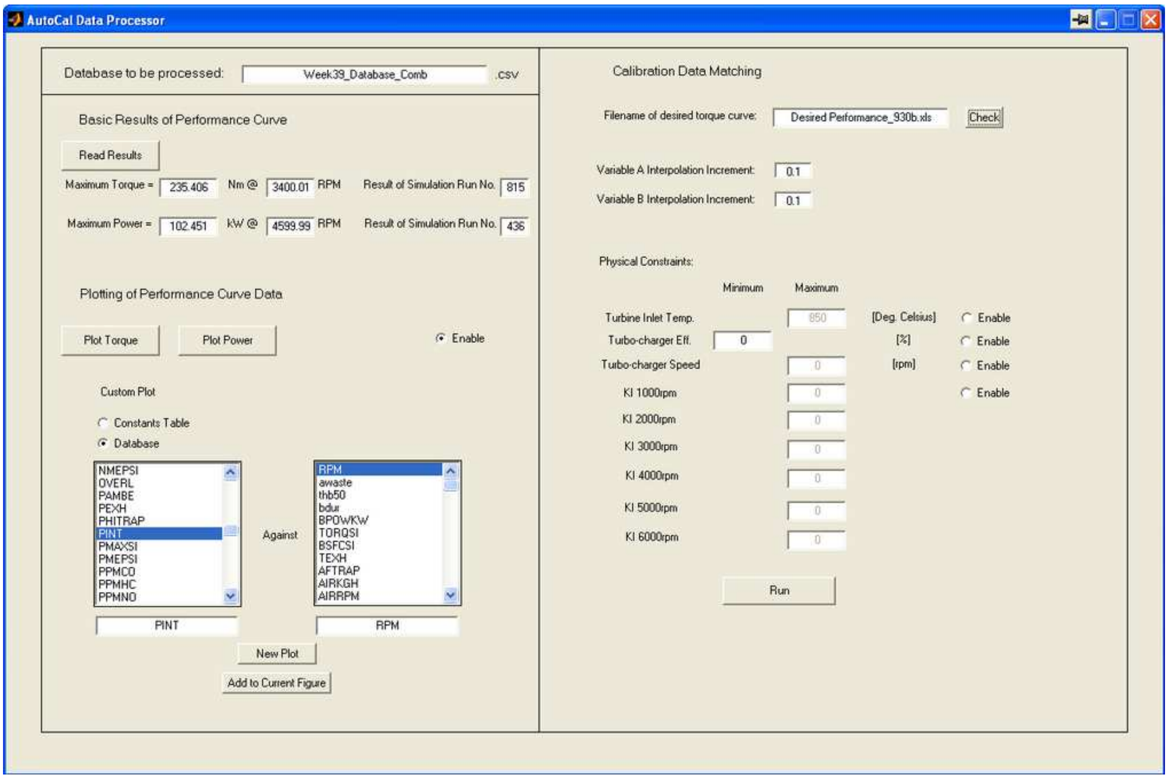


Figure 3.5: Related subfunctions of Function 2 - Define the desired torque curve and operating constraints.



AutoCal Data Processor

Database to be processed: Week39_Database_Comb .csv

Basic Results of Performance Curve

Maximum Torque = 235.406 Nm @ 3400.01 RPM Result of Simulation Run No. 815

Maximum Power = 102.451 kW @ 4599.99 RPM Result of Simulation Run No. 436

Plotting of Performance Curve Data

● Enable

Custom Plot

○ Constants Table

● Database

PINT Against RPM

Calibration Data Matching

Filename of desired torque curve: Desired Performance_930b.xls

Variable A Interpolation Increment: 0.1

Variable B Interpolation Increment: 0.1

Physical Constraints:

	Minimum	Maximum		
Turbine Inlet Temp.		850	[Deg. Celsius]	○ Enable
Turbo-charger Eff.	0		[%]	○ Enable
Turbo-charger Speed		0	[rpm]	○ Enable
KI 1000rpm		0		○ Enable
KI 2000rpm		0		
KI 3000rpm		0		
KI 4000rpm		0		
KI 5000rpm		0		
KI 6000rpm		0		

Figure 3.6: The interface used when processing the data from a calibration database or evaluating a simulated performance curve.

operating point in relation to the choke and surge lines of the compressor used on the engine. The constraining values would then be the boundary of a pre-defined area of the compressor map that provides adequate surge and choke margins. This boundary would then have to be defined i.t.o. compressor map variables, instead of the simplified approach discussed below.

Exhaust Gas Temperature. This temperature has to be limited to prevent damage to components in the exhaust path. The turbine inlet temperature was used.

Turbo-charger Efficiency. By staying above a certain efficiency level, choke can be avoided to a degree. The rise in intake temperature can also be kept within acceptable limits, which is important since higher intake manifold temperature decreases the mass of oxygen entering the engine and increases the likelihood of knock.

Turbo-charger Shaft Speed. In order to avoid damage to the turbo-charger components due to excessive centrifugal force the turbo-charger should be prevented from over-speeding.

Knock Intensity. This must be kept below a certain level to avoid damage to the engine as explained in Section 2.2.2. Allowable knock intensity varies with engine speed, therefore inputs are required for speeds from 1000 rpm to 6000 rpm in 1000 rpm increments. The code interpolates for speeds between these values. Since the simulated knock intensity value could not be validated against experimental results, due to the impracticality and expense of fitting indicating equipment to the test engine, knock intensity values were chosen from the range of simulated results. The object of evaluating the AutoCal program was not to find exact results but rather to show that the AutoCal program reacts correctly to the various constraints. In the case of knock intensity, it was to retard the ignition as well as lowering the boost pressure by opening the wastegate.

Function 3 - Perform simulations to find calibration values.

Figure 3.7 shows the subfunctions of Function 3. The figure also illustrates the sequence followed in performing the simulations to create the database. The code to perform these functions is contained in a Matlab m-file which is referenced to by the m-file of the main AutoCal GUI, Figure 3.3, when the **Run Simulations** button is clicked.

The first step is to receive the details of the model and the simulations to be run from the Matlab handles structure and the Runs.csv file created when performing

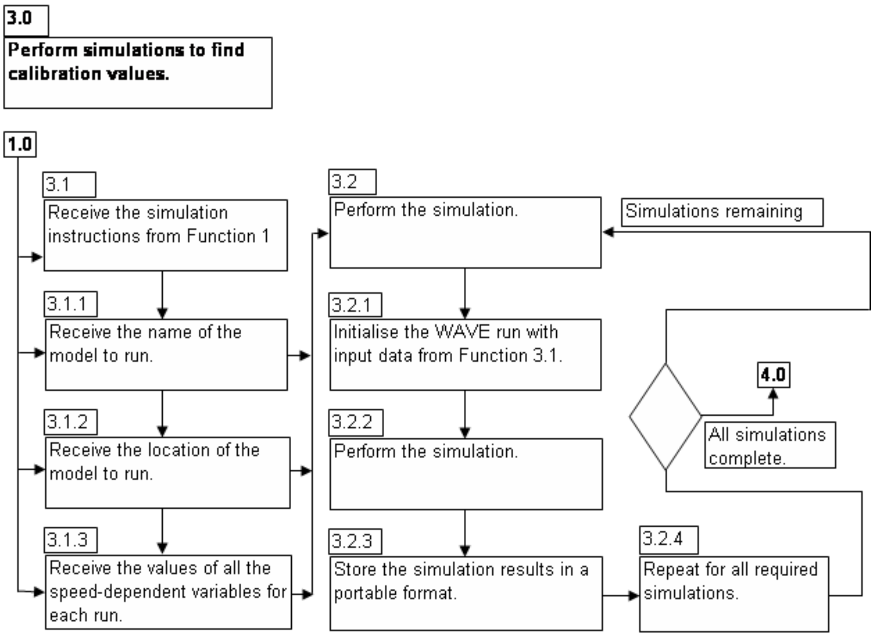


Figure 3.7: Related subfunctions of Function 3 - Perform simulations to find calibration values.

Function 1. The handles structure is a mechanism used by Matlab to store and retrieve the data created by, or required by, the various GUI components, as well as for referring to the various GUI components. For example, a name entered into a text box in one GUI, would be available to a routine initiated by a push button in the same GUI, or another GUI if the handles structure was passed to this other GUI.

After receiving the required data, the text-based WAVE model file is opened and the speed-dependant constants changed to the values of the next simulation to be run, as read from the Runs.csv file. Only the list of the speed-dependant constants at the start of the WAVE code is changed. The remaining code which describes the configuration of the engine model is left unchanged. The changed WAVE code describing the model with the settings for the next run is then saved as AutoCalRun.wvm and the simulation started from a DOS command prompt using the following Matlab instruction:

```
dos('wave AutoCalRun.wvm -s')
```

Once the simulation run is complete the results are written to a user-defined summary file by WAVE. This summary file lists the requested outputs in the separate columns in space-delimited text format with the name of each variable in the first row of each column. More detail on the various input and output files used by WAVE can be found in Appendix B.4.1. The AutoCal code then reads this summary file and stores the simulation results in the Matlab workspace.

The process is repeated for all the simulations listed in the Runs.csv file with the results of each simulation being added to that of the previous runs. Once all the runs have been completed the combined simulation results are stored in a csv file under the name supplied by the user. The data processing GUI shown in Figure 3.6 is then opened.

Function 4 - Select calibration values to achieve the desired torque curve.

The concept chosen to fulfill Function 4 and the related subfunctions is a database processing program written in Matlab and controlled by the GUI in Figure 3.6. As seen in Figure 3.8, the first function to be performed by this program is reading the database stored by Function 3, as well as the desired torque curve supplied by the user. Both sets of data are stored in the Matlab workspace to make it available to the code performing the other functions.

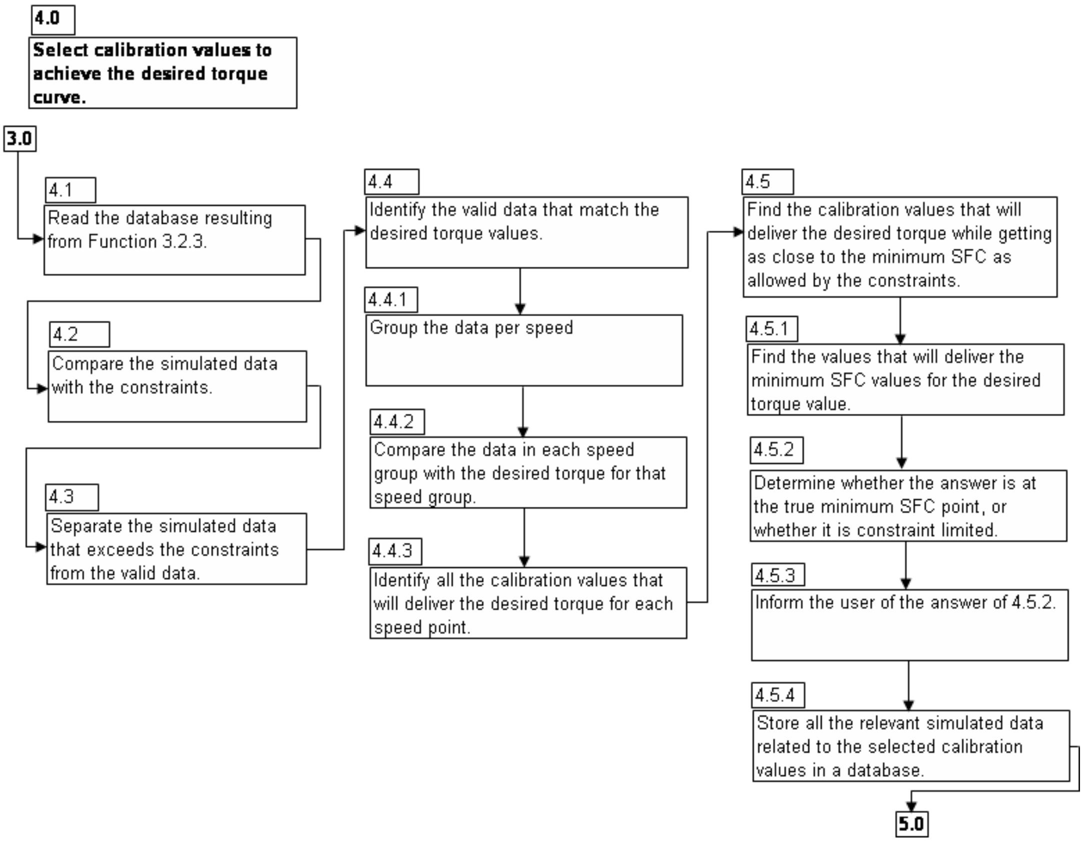


Figure 3.8: Related subfunctions of Function 4 - Select calibration values to achieve desired torque curve.

The program then groups the database of simulated results according to the simulated speeds and compares the list of speed groups to the speed points in the desired torque curve. If a speed point in the desired torque curve is not found in the list of speed groups a visual warning is issued. The buttons on the GUI which are used to initiate the remaining data processing steps are disabled until the speed points in the desired torque curve and database to be used are found to match. Options to correct this problem include adjusting the desired torque curve file or choosing a different database.

If the speed point check has been passed, the section of the GUI that allows input of constraints and the initiation of the calibration matching process is enabled. The process of defining the constraints is discussed above under Function 2. Once the user is satisfied that the constraint inputs have been entered correctly the **Run** button is used to initiate the process of finding the required calibration values.

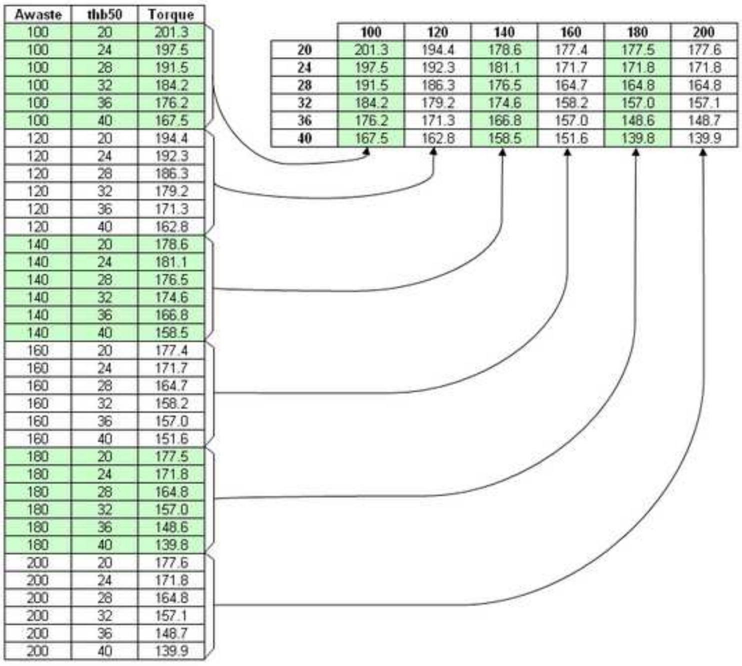
The following is a point-by-point description of how the subfunctions of Function 4 are fulfilled by the chosen concept once initiated by the **Run** button. Note that Subfunctions 2.4 and 2.5 are included in this discussion while Subfunctions 4.5.1 to 4.5.3 are handled and discussed together with Function 6.

A loop containing the following tasks corresponding to the subfunctions of Function4 is performed for each speed point in the database:

- The column-based data for the current speed point is read from the workspace and converted to a table format with the first calibration variable in the first row, the second on the first column and the data being processed in the body of the table. This process is illustrated in Figure 3.9. The data to be processed includes torque, SFC and the variables used when defining the operating constraints. The rest of the data available in the database is not processed during this stage.

- In order to increase the resolution of the available data a user-definable mesh is created between the minimum and maximum values of the variables being matched. Interpolation is used to find the values of the variables being searched at all the points of the mesh. The interpolation is performed by the Matlab interp2, two-dimensional interpolation function using the bicubic interpolation option. This method fits a bicubic surface through existing data points. The value of an interpolated point is a combination of the values of the sixteen closest points. The resulting interpolated data and its derivative is continuous (Mathworks, 2000). Figure 3.10 illustrates the result of this interpolation performed on a section of simulated data.

  New tables are then created with the interpolated values for each of the variables being searched. Increasing the number of increments that were chosen between the minimum and maximum of the calibration variables when the



Awaste	thb50	Torque
100	20	201.3
100	24	197.5
100	28	191.5
100	32	184.2
100	36	176.2
100	40	167.5
120	20	194.4
120	24	192.3
120	28	186.3
120	32	179.2
120	36	171.3
120	40	162.8
140	20	178.6
140	24	181.1
140	28	176.5
140	32	174.6
140	36	166.8
140	40	158.5
160	20	177.4
160	24	171.7
160	28	164.7
160	32	158.2
160	36	157.0
160	40	151.6
180	20	177.5
180	24	171.8
180	28	164.8
180	32	157.0
180	36	148.6
180	40	139.8
200	20	177.6
200	24	171.8
200	28	164.8
200	32	157.1
200	36	148.7
200	40	139.9

	100	120	140	160	180	200
20	201.3	194.4	178.6	177.4	177.5	177.6
24	197.5	192.3	181.1	171.7	171.8	171.8
28	191.5	186.3	176.5	164.7	164.8	164.8
32	184.2	179.2	174.6	158.2	157.0	157.1
36	176.2	171.3	166.8	157.0	148.6	148.7
40	167.5	162.8	158.5	151.6	139.8	139.9

Figure 3.9: Illustration of how data in column-based format is converted to table-based format for further processing.

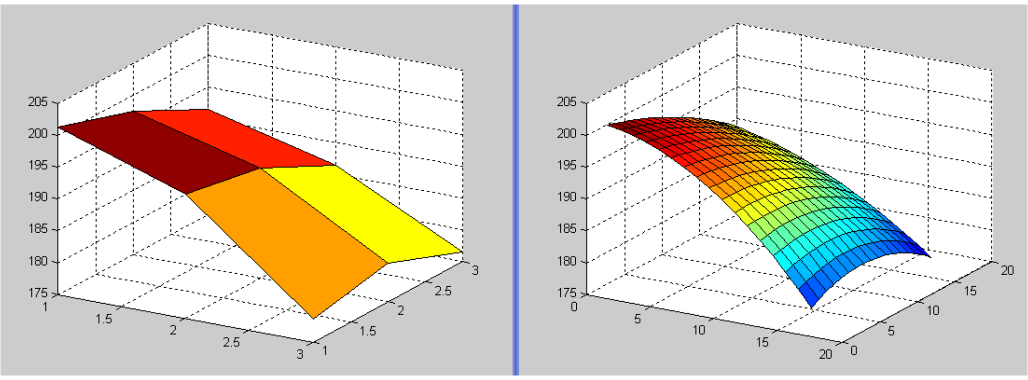


Figure 3.10: Results of interpolation. On the left is a surface created from a section of simulated data. On the right is the same data after fitting a bicubic surface through the available data points.

database was created would increase the accuracy of the interpolation but also significantly increase the number of simulations to be run. It is therefore necessary to find a compromise between increasing interpolation accuracy and computation time. The process of checking the accuracy of the interpolated results is discussed in Section 4.5.

- The variables identified as operating constraints are then evaluated to identify any values that fall outside of the indicated bounds. Each value in the interpolated tables of the constraining variables is checked against the defined constraints stored in the Matlab handles structure. If it falls outside the bounds, its coordinates are used to set the corresponding value in the table containing the interpolated torque values, to zero. This eliminates the possibility of an invalid torque value being chosen when the simulated data is matched to the desired torque, as explained in the next point. If an infeasible constraint is detected, the program will inform the user and stop the calibration process. The user will then have to redefine the constraint and restart the calibration process. A possible example is the maximum exhaust temperature being set below the minimum value found in the simulation database.

- After converting the data and eliminating invalid points, the desired torque value for the current speed point is read from the supplied Excel file. The program then starts by checking whether the desired value falls within the range defined by the minimum and maximum simulated torque values for the current speed point as found in the database of simulation results. If it falls outside this range the minimum or maximum simulated value will be used as the matched value for that point, depending on whether the desired value is higher or lower than the simulated range.

  If the desired value falls inside the range the program searches for any exact matches for this value from the interpolated torque data. If an exact match is not found, a tolerance is applied to the desired value and all the values within this tolerance are identified. If this still does not produce a match the tolerance will again be increased and the process repeated until a match is found. When a tolerance has been applied a pop-up window informs the user for which speed point it was applied and what the size of the tolerance was.

  This process of applying tolerances to the desired torque values is used primarily when the exact or close match to the desired torque has been invalidated in the previous step. This will happen if one of the critical operating parameters associated with the simulated value corresponding to the desired torque exceeds any of the physical constraints defined by the user. Even if no constraints are exceeded, a tolerance equal to the increments used when interpolating the original simulation data might have to be applied. This will occur if the exact desired torque value lies between two points in the table of interpolated torque values.

- Whether a tolerance is applied during the previous step or not, it is still possible that the table of interpolated torque values can contain more than one match for the desired torque value. Applying a tolerance will increase the likelihood of this occurring. The next step therefore is to apply the criteria for choosing between multiple suitable values if more than one was identified. To apply the minimum SFC criteria chosen for this project the SFC values corresponding to the coordinates of the matched torque value are listed and the lowest value identified. The coordinates of this point are then used to identify the chosen torque value. This torque value is then the matching value for the current speed point. It is saved in the Matlab workspace and in a csv file, together with the current speed point, the matching SFC value and the matching values of the two calibration variables. The matching results for each speed point will be added to this file as they are determined.

As can be seen from the subfunctions in Figure 3.8 it was decided early on that calibration values delivering the minimum SFC point would be used as the most suited combination. The objective of base calibration of ignition timing is to find the MBT point for each constant speed and load operating point. For each of the operating points being calibrated the AutoCal program therefore has to evaluate different combinations of the calibration values delivering the desired torque for that specific point. It then has to find the combination of calibration values that results in an MBT value equal to the desired torque.

At constant engine speed, mixture composition and intake air flow rate, the timing that gives the MBT also gives the minimum SFC (Heywood, 1988). The minimum SFC value was therefore used to find the required combination of turbo-charger boost pressure and ignition timing.

Apart from the obvious fuel consumption benefits, this combination will also have the lowest possible turbo-charger boost pressure that can deliver the desired torque. This adds the benefit of decreased turbo-charger outlet temperatures which decreases the likelihood of knock. It also reduces the heat load on the inter-cooler and the wear on engine components such as the turbo-charger bearings.

The fact that the turbo-charger might be forced to operate outside of its feasible efficiency zone by choosing the lowest possible turbo-charger boost level to achieve the desired torque must be avoided. This is addressed by setting up the turbo-charger efficiency constraints in such a way that calibration values resulting in operation outside the feasible range are not used in the matching process.

Once the matching process has been completed for each of the speed points in the desired torque curve the GUI designed to fulfill Function 5 is opened.

Function 5 **- Compare the results with the desired torque curve.**

Figure 3.12 shows the GUI designed to fulfill the majority of the subfunctions under Function 5 (Figure 3.11) by displaying the results of the matching process together with the desired torque curve. This allows the user to see whether a match was found for each of the speed points, as well as how close each match is to the desired value. Subfunctions 5.2.1 and 5.2.2 are fulfilled by another GUI and associated code together with the subfunctions of Function 6 and are therefore discussed under Function 6.

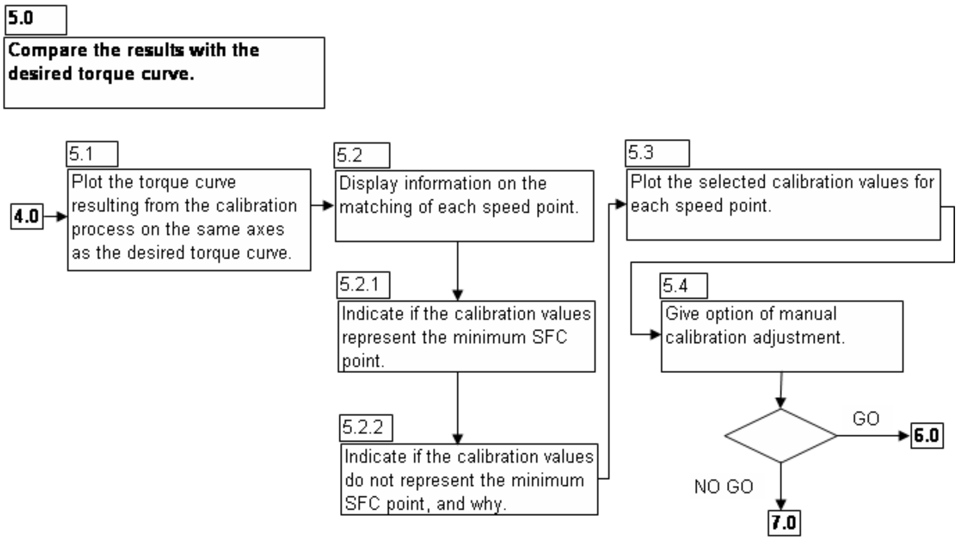


Figure 3.11: Related subfunctions of Function 5 - Compare the results with the desired torque curve.

Figure 3.12 shows and example of the desired torque curve exceeding the available data in the simulation database. This can be seen at the 2000 rpm and 3200 rpm points and is indicated by a red cross and associated comment in the legend.

The data used to create this comparative plot is read from the matching results file described under Function 4 above as well as the desired torque file supplied by the user. The calibration values determined during the matching process are displayed on command of the user when the **Plot Variable A Calibration** and **Plot Variable B Calibration** buttons are clicked.

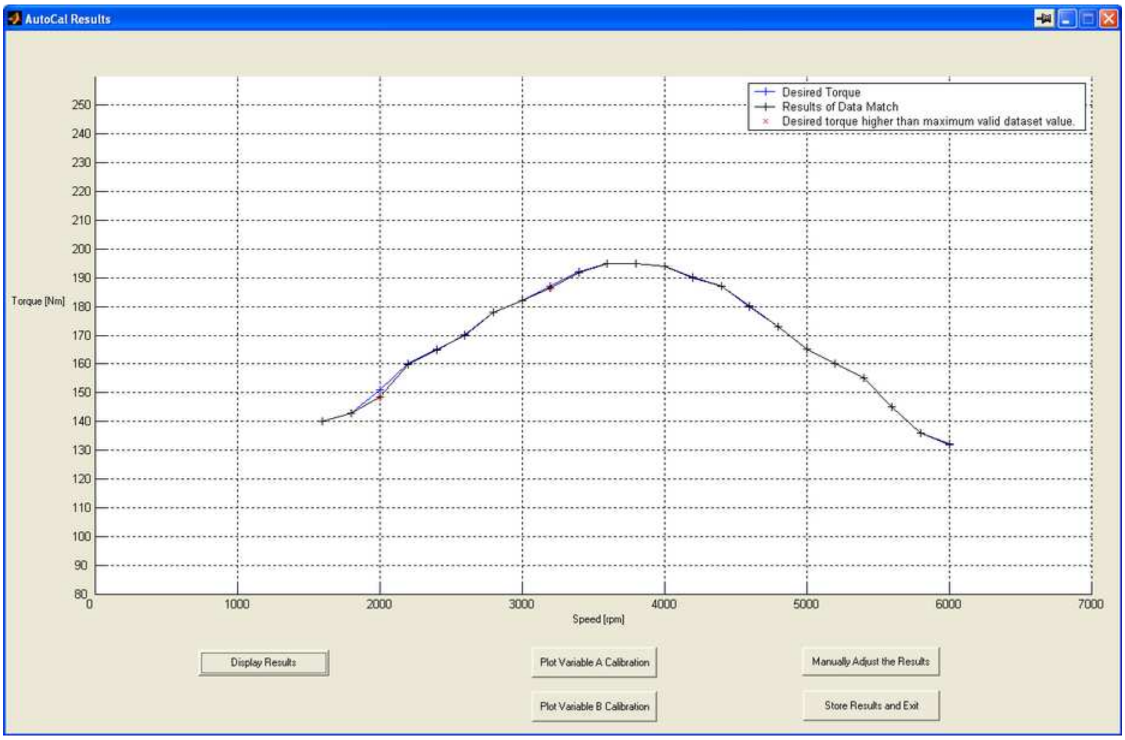


Figure 3.12: The interface used to evaluate the results of the matching process.

To perform Subfunction 5.4 two buttons were programmed. In the case of the user not being satisfied by the calibration results the **Manually Adjust the Results** button opens the GUI used to fulfill Function 6. If the user is satisfied code activated by the **Store Results and Exit** button will save the matching results to the WAVE-Results-match.csv file and close the AutoCal program.

Function 6 **- Allow user adjustment of the calibration values.**

This section discusses how AutoCal performs the subfunctions under Function 6 shown in Figure 3.13. Also included are Subfunctions 4.5.1 to 4.5.3.

The **Manually Adjust Results** button on the primary results display shown in Figure 3.12 opens the interface seen in the upper quarter of Figure 3.14. Clicking on the **Initiate** button on this interface activates code that reads the WAVE-Results-match.csv file and displays the speed points in this file in the speed selection listbox. The code also plots the four graphs shown underneath the interface in the figure. These graphs display the torque and specific fuel consumption resulting from the matching process as well as the corresponding calibration values. Also shown in the graphs is the range of calibration values remaining after the constraints have been applied, see Figure 4.9.

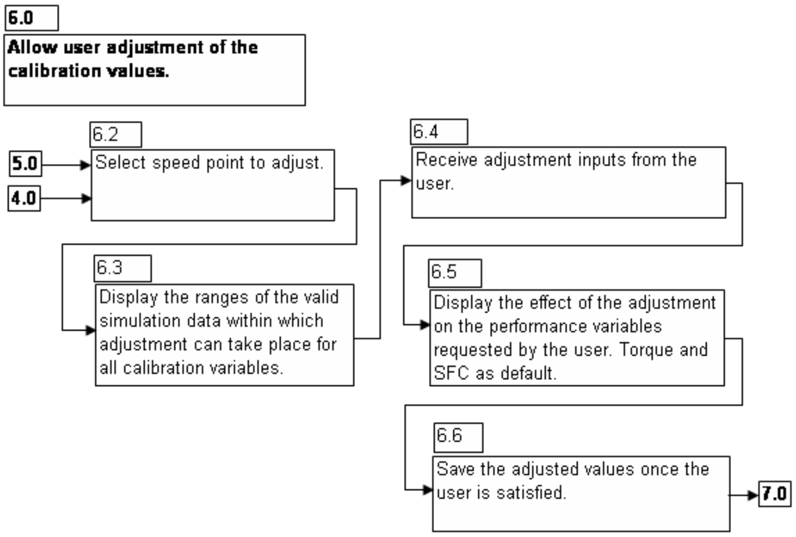
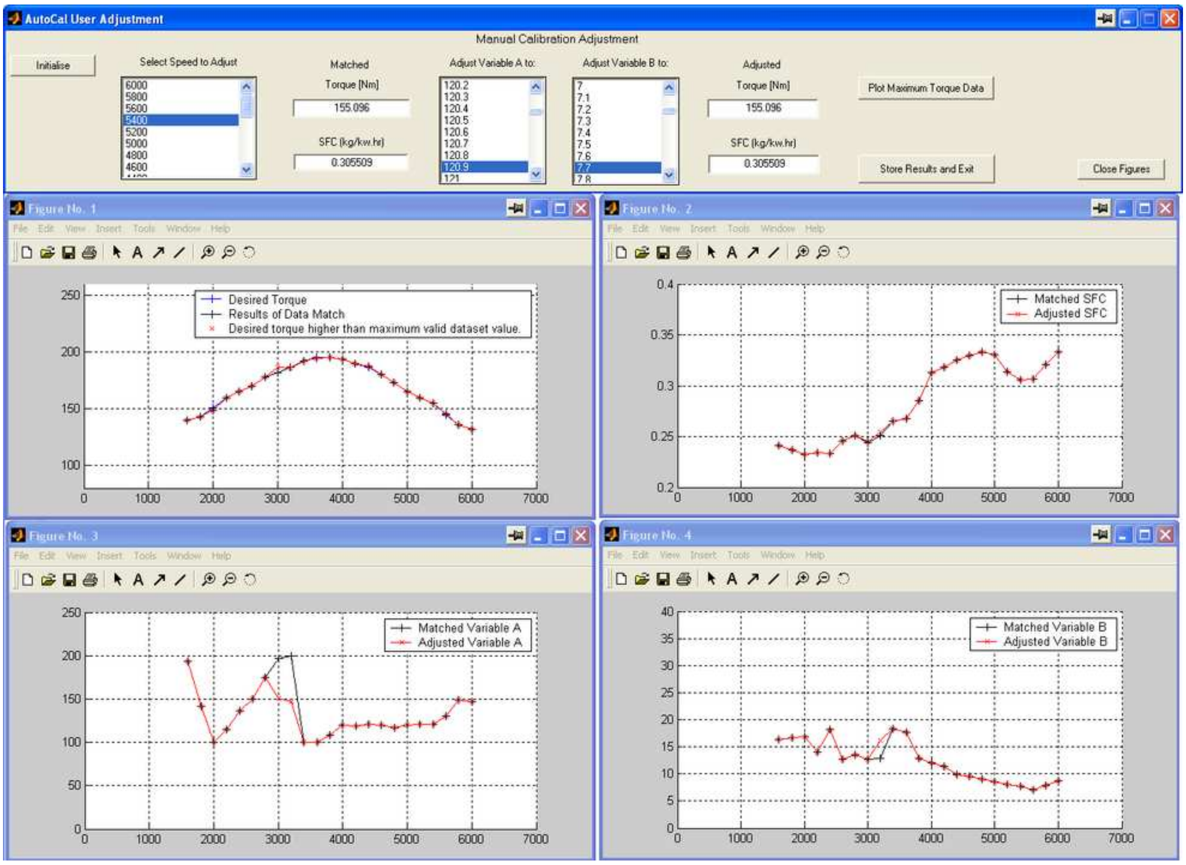


Figure 3.13: Related subfunctions of Function 6 - Allow user adjustment of the calibration values.



Matched Torque [Nm] 155.096

SFC (kg/kw.hr) 0.305509

Adjusted Torque [Nm] 155.096

SFC (kg/kw.hr) 0.305509

Figure 3.14: The interface that allows the user to further examine the results of the matching process as well as make manual changes to the calibration if required.

Upon selection of a speed point from the available list the following actions will be performed:

1. The interpolated torque table for the selected speed is opened.
2. The torque and SFC values resulting from the matching process are displayed.
3. The constraint limited points in the interpolated torque table is identified and the corresponding calibration values changed to 0.
4. The processed calibration values are displayed in the two calibration value listboxes with the listbox value set to the matched calibration values.

Selecting a different value in either one of the calibration value listboxes will initiate the following sequence of actions:

1. The new calibration value is saved in a duplicate of the WAVE-Results-match table called Adjust-Results together with the corresponding torque and SFC values. The rest of the values in this new table will be identical to the values in the WAVE-Results-match table.
2. The values in the Adjust-Results table will be plotted on the four graphs together with the data from the WAVE-Results-match table. This will show the effect of the the calibration adjustment.
3. The data in the interpolated torque table corresponding to the new calibration value will be examined to find the values of the other calibration variable that are constraint-limited, as explained below.
4. The new list of values for the other calibration variable will be displayed in the applicable list box.

Re-evaluating the constraint situation is required every time a calibration value is changed. This is due to the fact that a change in one value changes the operating condition and the effect that the other calibration variable has on the constraining outputs. For example: If the turbo-charger boost pressure is increased by decreasing the wastegate opening area, the likelihood of knock will increase. The number of degrees that the timing can be advanced relative to TDC will therefore be reduced compared to a lower turbo-charger boost level.

The lists of calibration values in the two calibration variable list boxes also serve as an indication of whether a particular point is truly at minimum SFC or not. Not being a minimum SFC can have two causes. The first being not simulating a large enough range of calibration variables and the second being the possibility that the constraints prevented the minimum SFC point from being reached.

Determining whether either one of these conditions exist is accomplished by examining the calibration values in the graphs and the list boxes. If the calibration variables resulting from the matching process are on the limits of the simulated range, it means that values outside this range can possibly provide an even better match for the desired torque value. If any of the values in the calibration list boxes are equal to zero it indicates that constraints are affecting the results. If the calibration variable selected by the matching process lies on the edge of the zero values it might mean that the true minimum SFC point could not be reached. Whether this is true can be determined by adjusting the calibration values while observing the changes in the torque. Once the user is satisfied with the calibration matching results the **Store Results and Exit** button will save the current data and close the AutoCal program.

Function 7 - Confirm the results of the calibration selection process.

Since interpolation was used during the calibration process, the validity of the final calibration result has to be confirmed. Using fewer increments between the minimum and maximum calibration values during database creation will decrease the information available - increasing the requirement for confirmation.

This function is performed by compiling a new constants table using the calibration matching results and simulating a performance curve using this data. Interfaces used for this process were programmed to fulfill a selection of the other functions required for AutoCal.

Once the new constants table has been compiled using Microsoft Excel the performance curve simulation is set up using the GUI in Figure 3.3. Performing the simulation follows a similar sequence to the one described under Function 3. Once the simulation is complete AutoCal will automatically open the GUI also used to manage most of Function 4. The section seen on the left of Figure 3.6 was programmed primarily to fulfill Function 7. It allows convenient examination of simulation data and in particular processing of performance curve simulations.

As can be seen in the figure the name of the database to be processed is defined using an text-edit box. A quick check is then performed on the data by reading and displaying the maximum torque and power values found in the defined database. An invalid database will result in an error at this stage and the database will have to be redefined. The code performing this function also makes the data available to to other components in this GUI.

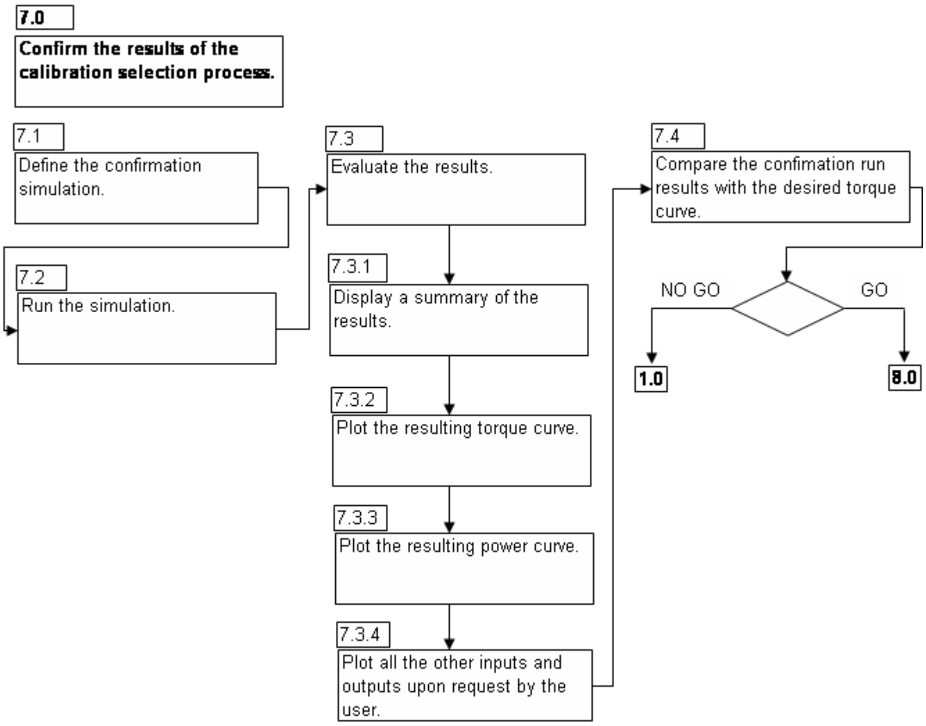


Figure 3.15: Related subfunctions of Function 7 - Confirm the results of the calibration selection process.

Once the database has been validated the torque and power curves are displayed on activation of the relevant buttons. This information can then be used to confirm whether the calibration values determined during the matching process can deliver the desired torque curve.

A custom plotting facility was also programmed into this GUI. Any of the inputs or outputs of the simulation can be plotted against speed or any other input or output. This function was created to enable convenient evaluation of the simulation results without using another software package. The values of constraining variables can for instance be examined to gain insight into the matching results and assist with deciding on corrective action, if any is required.

Function 8 **- Store results and exit.**

The functions shown in Figure 3.16 are performed by code activated by the Save Results and Exit buttons. These buttons were placed on the interfaces shown in Figures 3.14 and 3.12. The code saves the results in csv files as well as the Matlab .mat format. These files can then be used for further processing of the results. The calibration results can also be converted to the format required by the engine control unit for physical engine testing.

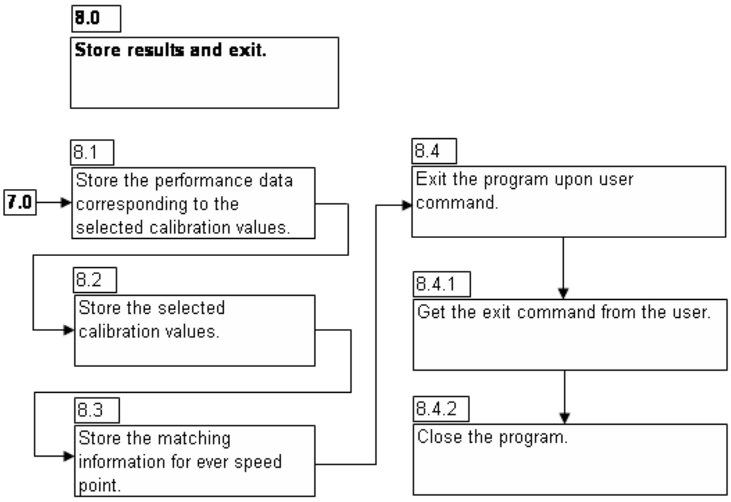


Figure 3.16: Related subfunctions of Function 8 - Store results and exit.

Chapter 4

Application and Evaluation of AutoCal

This chapter documents the process of evaluating the AutoCal program. A complete simulation based calibration process was completed in order to evaluate whether all the AutoCal functions are performing according to the specification.

4.1 Creating the Database

Evaluation of Functions 1 **and** 3

AutoCal was set up to create the database by firstly inserting the name of the WAVE model to be simulated as well constants table to be used. Following this the desired speeds, as well as the ranges and number of increments of the wastegate area (*awaste*) and 50% burn point (*thb50*) variables were defined. The final steps were to insert the name to be used for the database and starting the simulation process.

The chosen combination of variable increments and speed points resulted in 1512 simulation runs, which took approximately eighteen hours to complete. On completion, the database processing GUI shown in Figure 3.12 automatically opened to allow processing of the simulation database and determination of the calibration values.

4.2 Desired Torque Curve

Evaluation of Functions 4 **and** 5**. Partial evaluation of Function** 2

Figures 4.1 and 4.2 show the range of torque and power values available in the simulation database. These figures were created by using the torque plotting function of the GUI shown in Figure 3.12. After examination of this data, the performance curves shown in Figure 4.3 were defined through manual input, in order to demonstrate the ability of AutoCal to find calibration values for any torque curve, falling within the bounds shown in Figure 4.1 and conforming to the constraints defined by the user. This evaluated Functions 2 and 4 and confirmed that they were performing as specified.

The following specifications were used when defining the desired performance curves used for demonstration: Maximum power of 95 kW at 5600 rpm, maximum torque of 202 Nm and torque backup provided by keeping the torque curve level, or sloping downwards from around 2800 rpm onwards. These values were selected because they fit inside the defined range and represented a significant improvement over the naturally aspirated torque curve shown in Figure A.11. The torque and power values were also defined high enough to trigger the operating constraints and thereby test the ability of the AutoCal program to find calibration values which will avoid damage inducing engine operation.

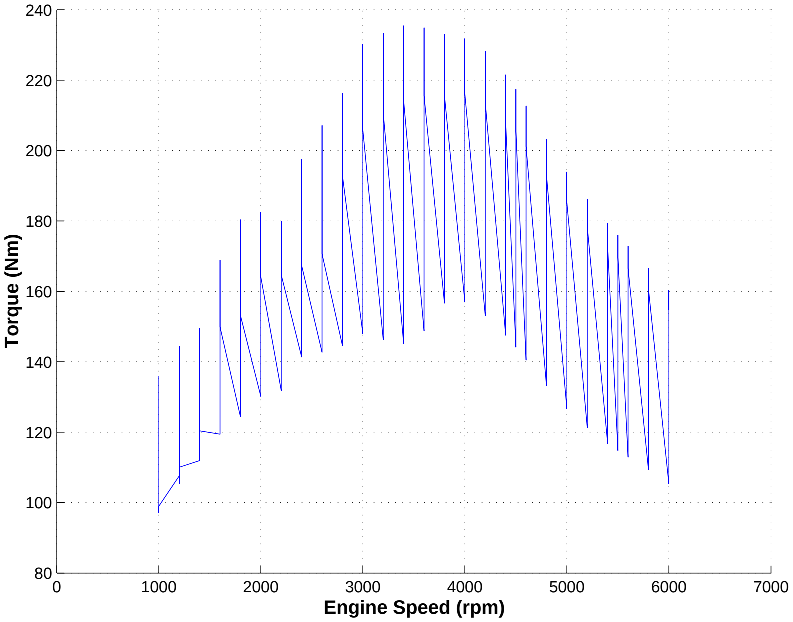


Figure 4.1: Range of torque values found in the database of simulation results.

Figures 4.4 and 4.5 show the result of the matching process for the case with no constraints defined. Figure 4.13 shows the results of the manual adjustments made to smooth out the ignition curve without serious compromises to the SFC.

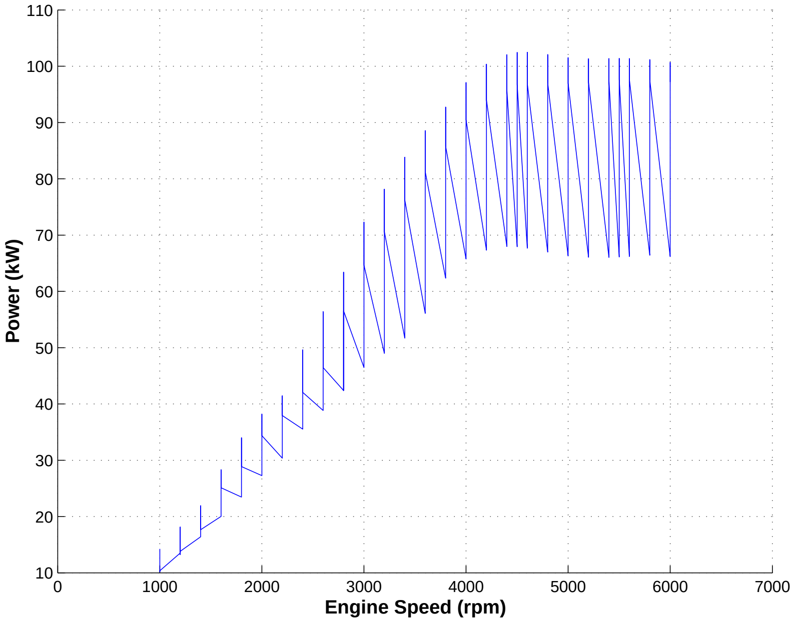


Figure 4.2: Range of power values found in the database of simulation results.

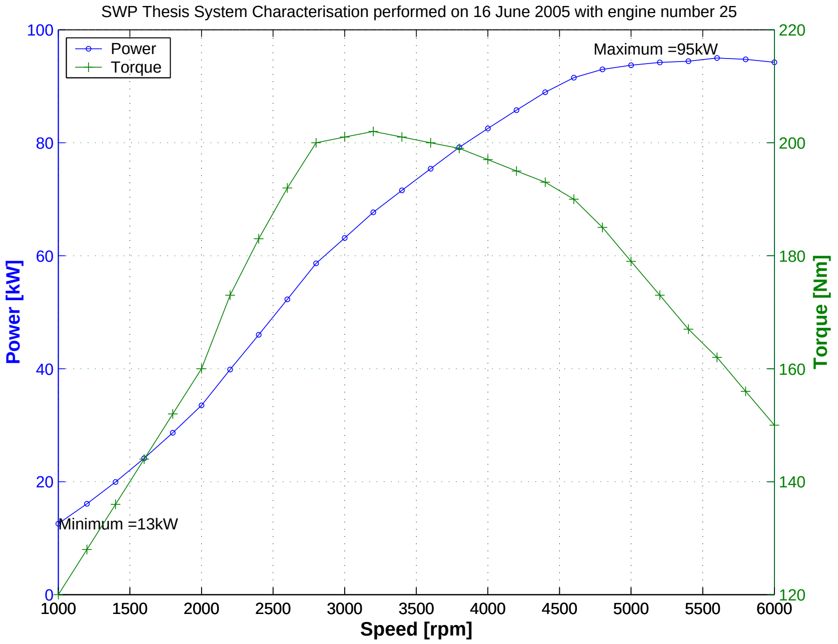


Figure 4.3: Desired performance curves used to evaluate the AutoCal program.

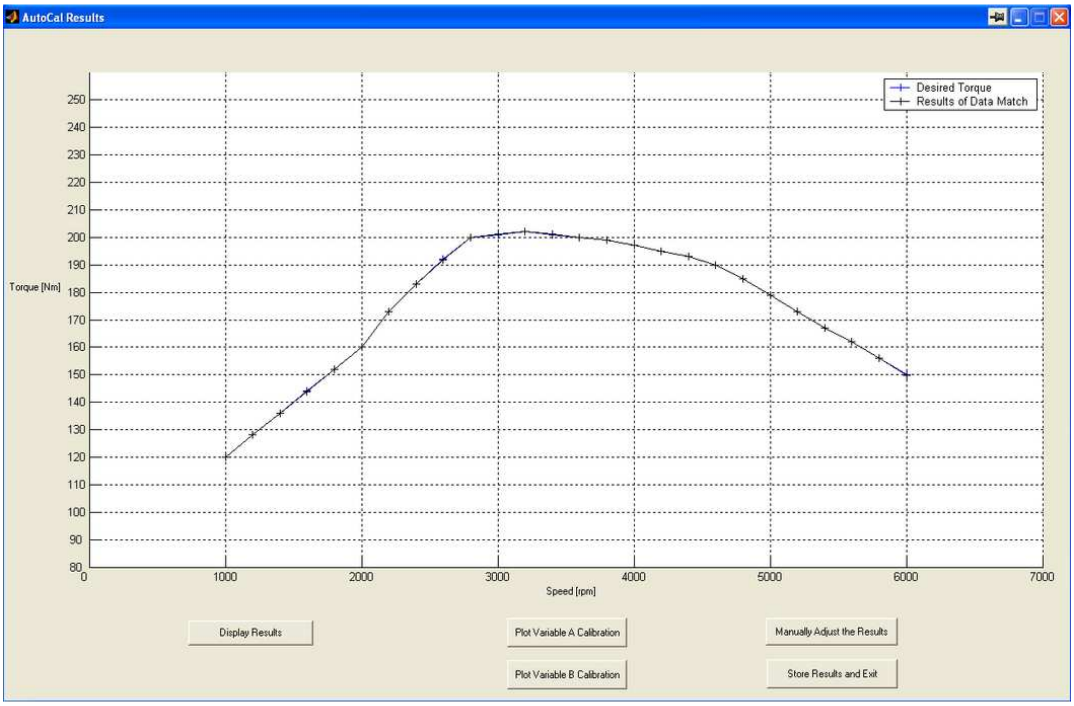
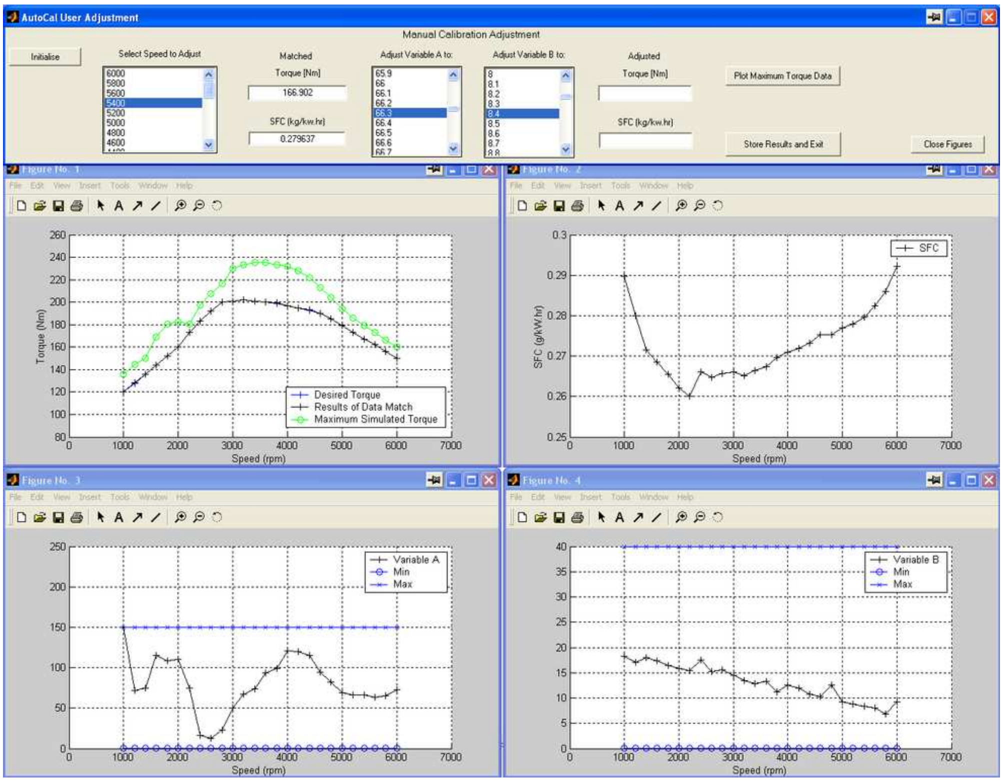


Figure 4.4: Torque match with no constraints enabled.



Matched Torque [Nm]: 166.902

SFC (kg/kw.hr): 0.279637

Figure 4.5: User-adjustment GUI displaying the calibration matching results with no constraints enabled. Also shown is the maximum simulated torque with no constraints.

4.3 Operating Constraints

Further evaluation of Function 2

4.3.1 Turbine Inlet Temperature

Figure 4.6 shows the range of the simulated turbine inlet temperature values found in the simulation database. The various windows in Figure 4.7 illustrates the effect that setting the maximum turbine inlet temperature to 1140°C, has on the calibration matching results. Comparing the torque results in this figure with Figure 4.4 shows that AutoCal was able to find slightly different calibration values that did not exceed the temperature constraint but would still result in the desired torque output. The two figures at the bottom of Figure 4.7 show that in this case only the ignition timing values at the high speed points were affected by the constraint.

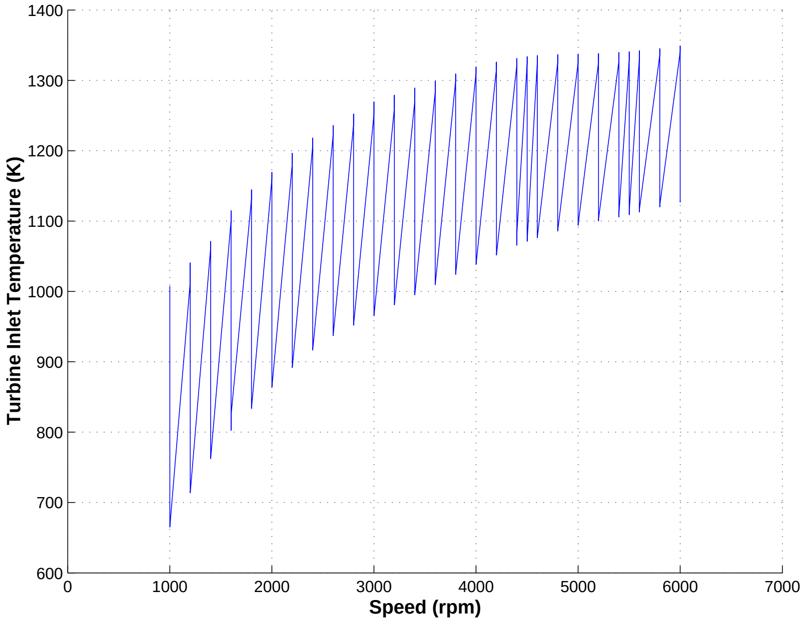
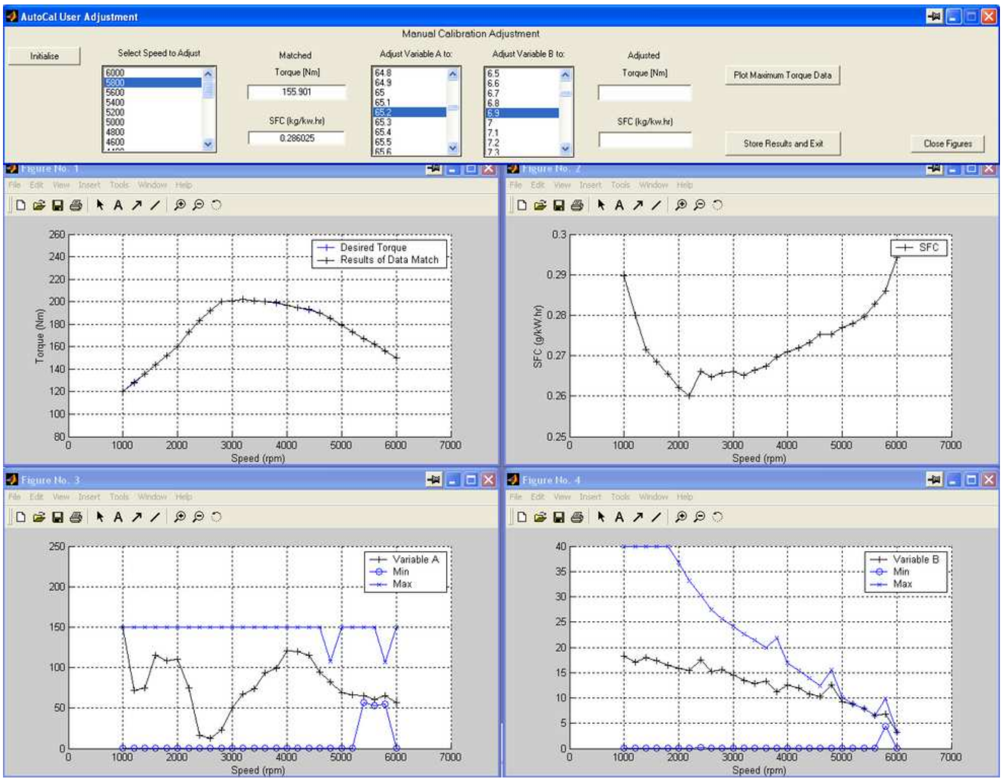


Figure 4.6: Range of turbine inlet temperature values found in the database of simulation results.

4.3.2 Turbo-charger Shaft Speed

Figures 4.8 and 4.9 show the results of the matching process with the maximum turbo-charger speed set to 130 000 rpm as the only constraint. Comparing Figures 4.4 and 4.8 it can be seen that limiting the maximum turbo-charger shaft speed decreases the torque available at high engine speeds. The range of turbo-charger shaft speed in the simulation database can be seen in Figure 4.10. The variable



Select Speed to Adjust: 5800

Matched Torque [Nm]: 155.901

SFC (kg/kw.hr): 0.286025

Adjust Variable A to: 65.2

Adjust Variable B to: 6.9

Figure 4.7: User-adjustment GUI displaying the calibration matching results with maximum turbine inlet temperature set to 1140°C.

listboxes in the GUI in Figure 4.9 shows how the wastegate area (*awaste*) and 50% burn point (*thb50*) have been limited by AutoCal during the calibration process. Comparing the plots in the lower left-hand corners of Figures 4.5 and 4.9, illustrate how the calibration of the wastegate area is affected by the turbo-charger speed constraint.

4.3.3 Knock Intensity

Figure 4.11 shows the effect of activating the knock intensity constraint on the calibration values. Both the wastegate area (Variable A) and the ignition timing (Variable B) are affected, so that the maximum turbo-charger boost pressure and ignition advance angle will be limited. Figure 4.12 shows the range of knock intensity in the simulation results.

When all the different constraints are defined together, the separate effects discussed above will be combined during the process of eliminating data exceeding the constraints. This gives AutoCal the ability of monitoring multiple constraints when determining calibration values.

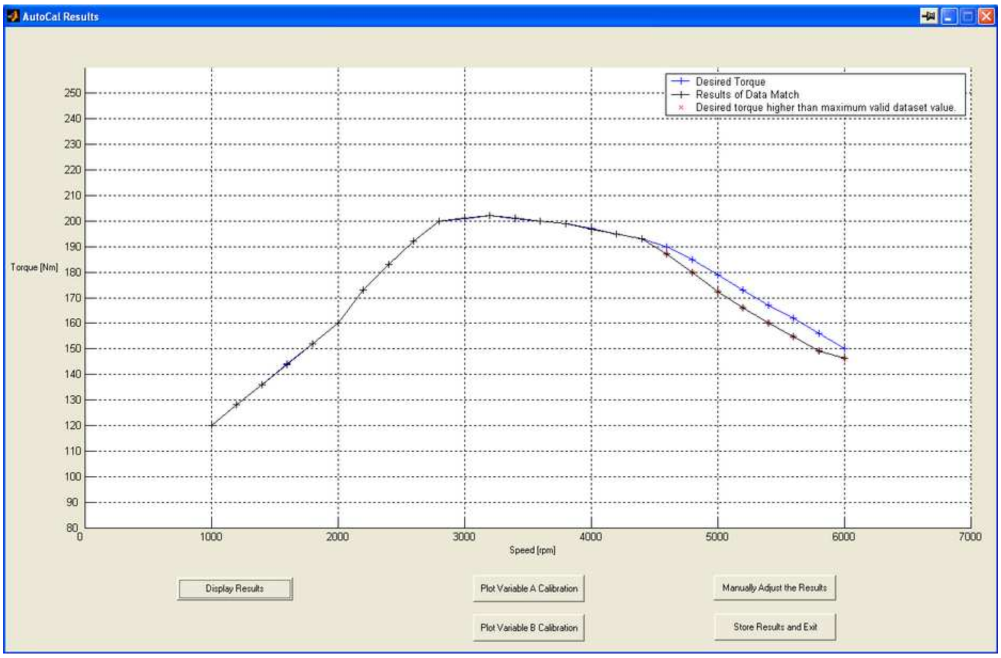
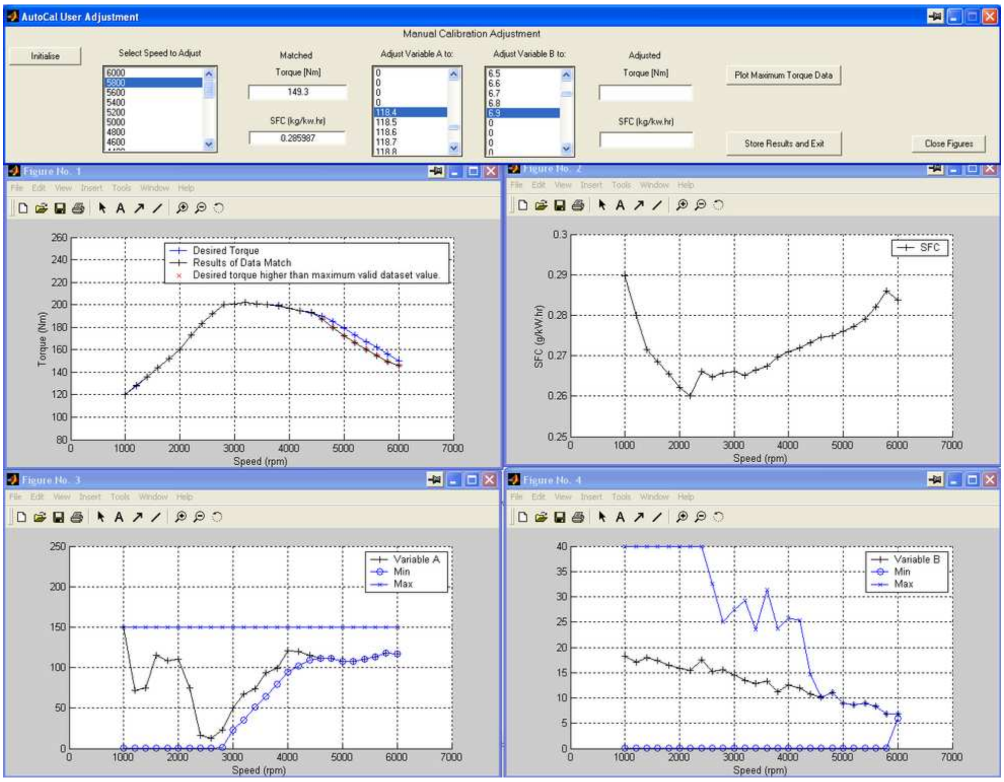


Figure 4.8: Matching results with maximum turbo-charger shaft speed set to 130 000 rpm.


AutoCal User Adjustment

Manual Calibration Adjustment

Select Speed to Adjust: 5800

Matched Torque [Nm]: 149.3

SFC (kg/kw.hr): 0.285987

Adjust Variable A to: 118.4

Adjust Variable B to: 6.9

Figure 4.9: User-adjustment GUI displaying the calibration matching results with maximum turbo-charger shaft speed set to 130 000 rpm.

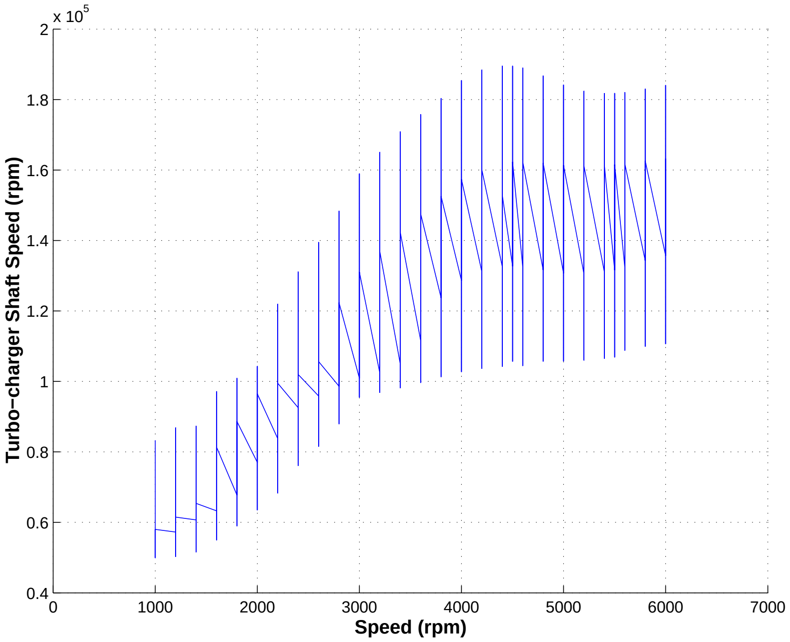
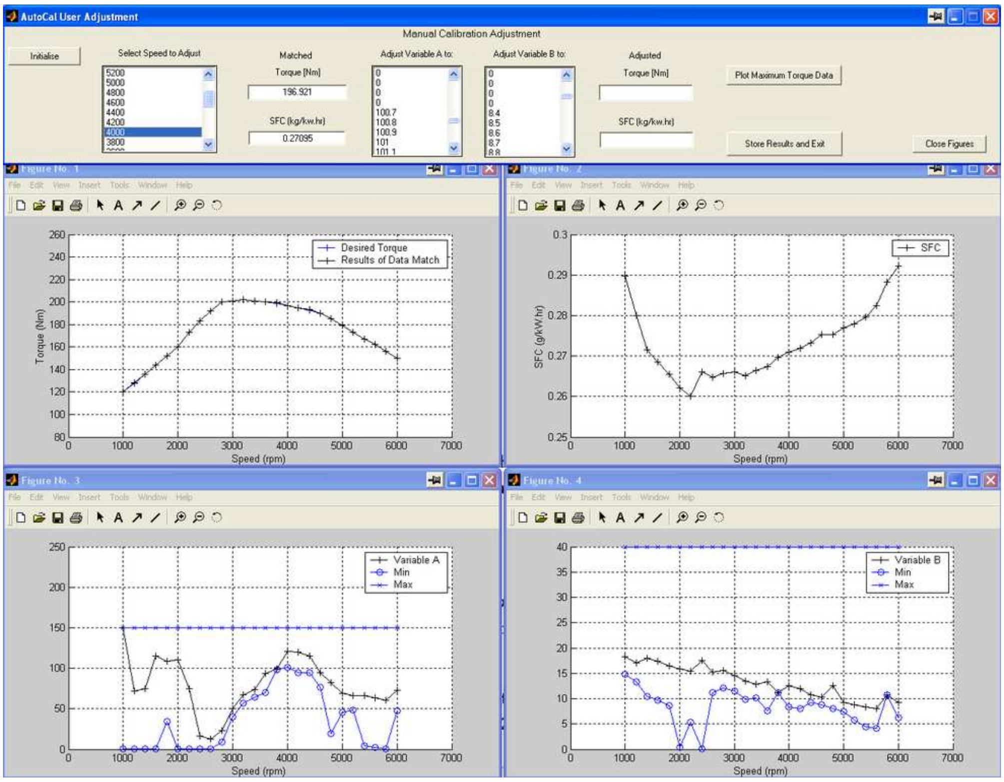


Figure 4.10: Range of turbo-charger shaft speed values found in the database of simulation results.



Matched Torque [Nm]: 196.921

SFC [kg/kw.hr]: 0.27095

Figure 4.11: User-adjustment GUI displaying the calibration matching results with the knock intensity constraint activated.

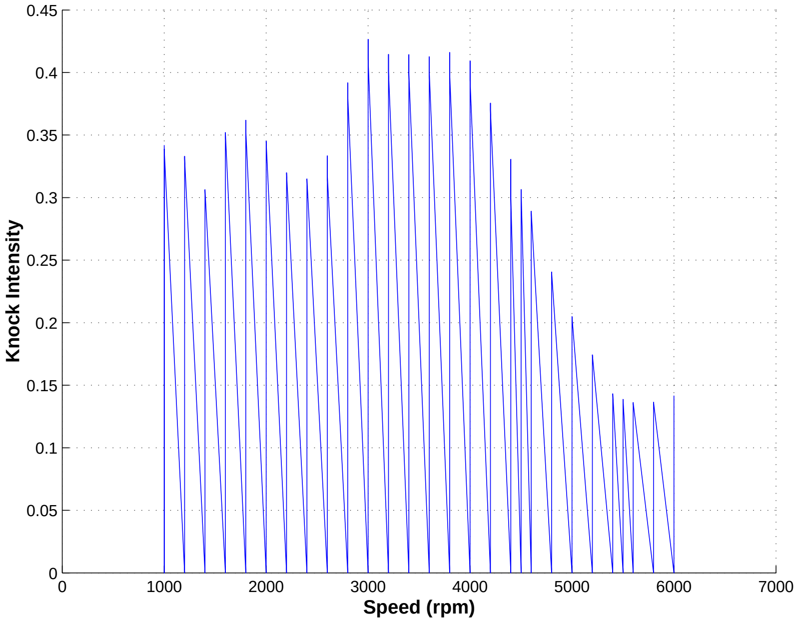


Figure 4.12: Range of knock intensity values found in the database of simulation results.

4.4 Manual Adjustment

Evaluation of Function 6

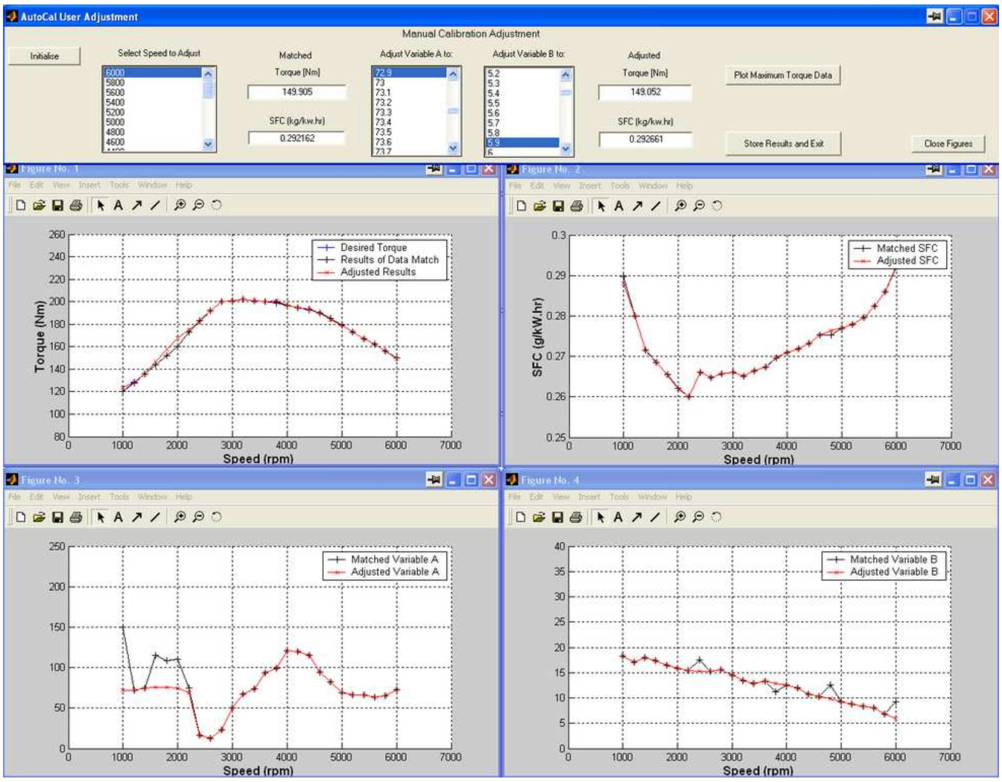
Figures 4.13 to 4.17 illustrate the manual adjustment process described in Section 3.3.2. This process was used to demonstrate how the user can manually adjust the calibration matching results to achieve smoother calibration curves, by slightly compromising the minimum SFC and desired torque matches.

4.5 Validating the Results

Evaluation of Function 7

The results of the automatic calibration and manual adjustment were used to create a new constants table in order to validate the accuracy of the results. The required performance curve simulation was set up using the main AutoCal GUI (Figure 3.3) and resulted in the data displayed in Figure 4.18. Figure 4.19 shows the validation results of the calibration values determined using the manual adjustment function.

The results in the regions below 2400 rpm and above 4000 rpm were sufficiently accurate, with the manual adjustments made to the torque curve clearly reflected in the validation data. In between these two regions the data did not show good agreement.



Select Speed to Adjust: 6000

Matched Torque [Nm]: 149.905

Matched SFC (kg/kw.hr): 0.292162

Adjust Variable A to: 72.9

Adjust Variable B to: 5.9

Adjusted Torque [Nm]: 149.052

Adjusted SFC (kg/kw.hr): 0.292661

Figure 4.13: User-adjustment GUI displaying the calibration matching results after manual adjustment.

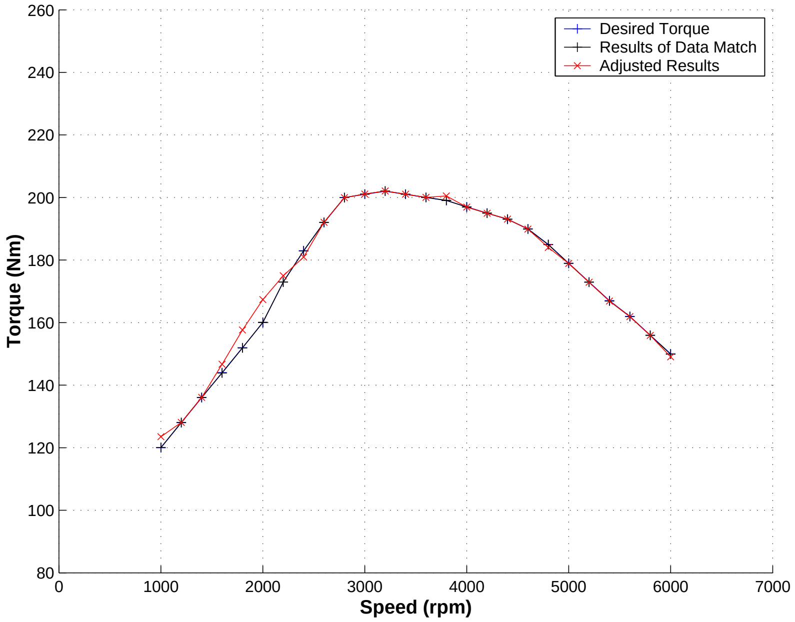


Figure 4.14: Torque after manual adjustment.

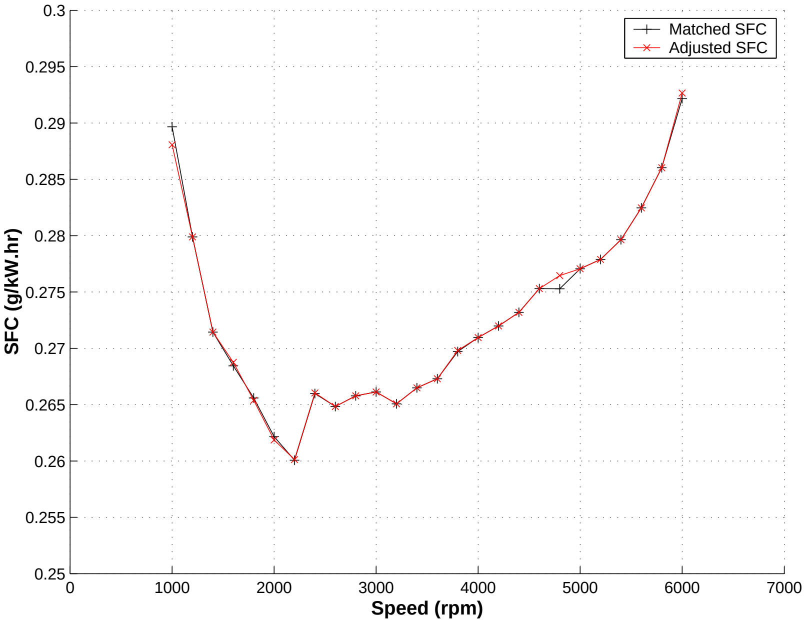


Figure 4.15: SFC after manual adjustment.

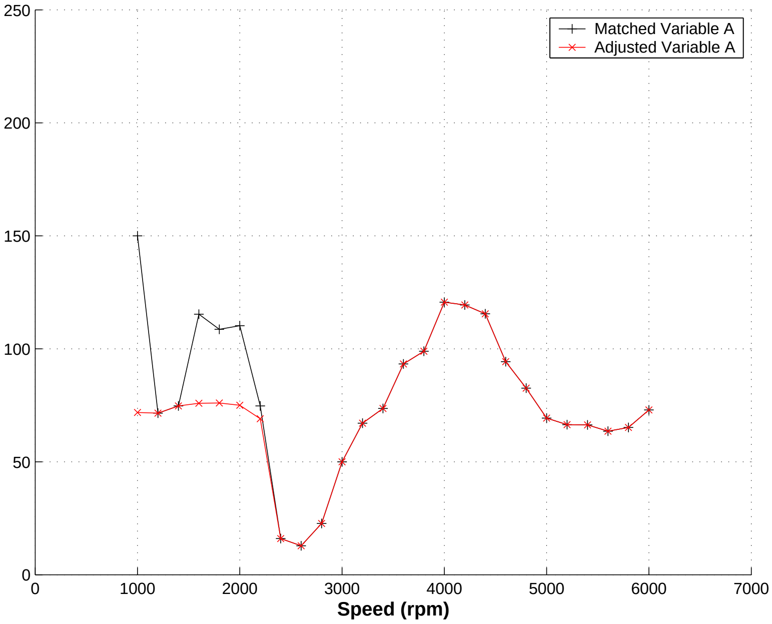


Figure 4.16: Variable A (wastegate area) after manual adjustment.

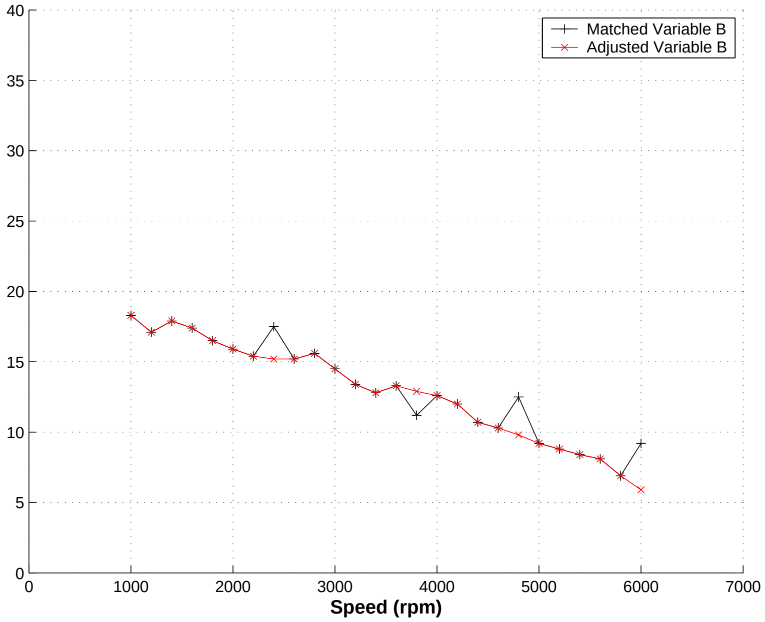


Figure 4.17: Variable B (*thb50*) after manual adjustment.

In order to improve the accuracy between 2400 rpm and 4000 rpm the amount of increments defined between the minimum and maximum wastegate area (Variable A) was increased from 6 to 11. The new simulation data was inserted into the database used for the first match and the calibration matching process repeated.

Figures 4.20 and 4.21 show the torque and SFC results of the matching process performed with the new database. Also shown are the results of the manual adjustment (Section 4.4) performed with the new simulation results. Figure 4.22 compares the values of Variable A and Variable B resulting from the new database with those of the initial database.

As can be seen from Figures 4.23 and 4.24, the increased simulation increments resulted in a significant improvement in accuracy, except at the 2600 rpm point. The inaccuracy at 2600 rpm can possibly be due to the fact that, relative to the other points, this particular point is more sensitive to a change in *awaste*, as seen in Figure 4.25. Even more increments are therefore required at this speed point to ensure accurate results. Furthermore, Figure 4.26 shows that even with the increased amount of increments, the major range of torque values fall between only two *awaste* points. The torque values in this region are therefore quite sensitive to a change in *awaste*, or to an inaccurately interpolated *awaste* value. Increasing the increments improved the interpolation which resulted in better validation results.

The amount of simulation increments required to ensure accurate results will depend on the properties of the specific engine model used for the calibration process. It is therefore important to determine the correct compromise between computational time and accuracy for every different model being processed and ideally also for the different speed points. A convenient feature of the AutoCal program is that data from different speed points, simulated with various numbers of increments, can be combined into one database and processed together. This can reduce computational time by allowing fewer simulation points in less sensitive regions.

By examining an initial set of low resolution results it will be possible to determine where increased resolution is required. From Figure 4.26 it can be seen that the required torque range could be simulated by using a smaller range of wastegate area values but with the same amount of increments. This will increase the accuracy without an increase in computational time.

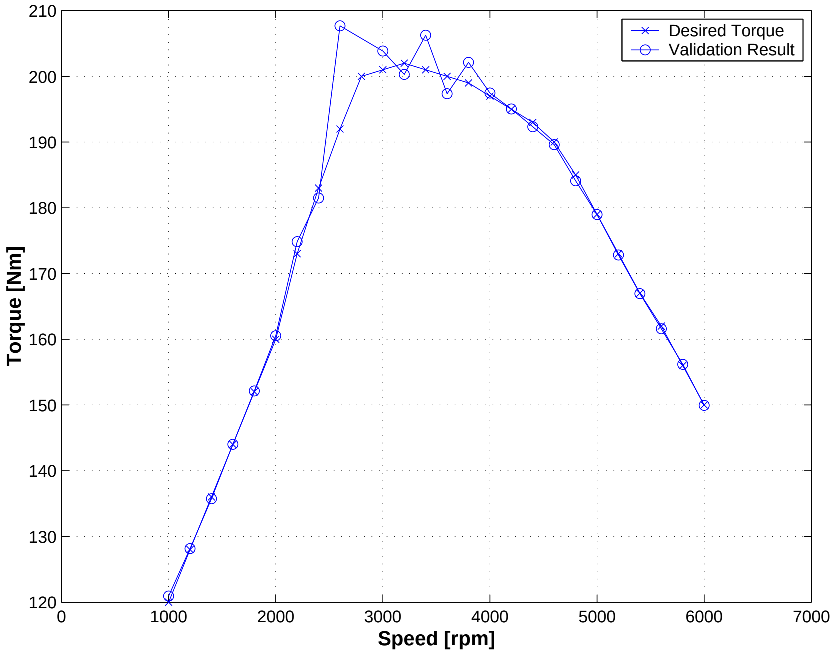


Figure 4.18: Initial comparison between desired torque and the result of the performance curve performed to validate the calibration matching process.

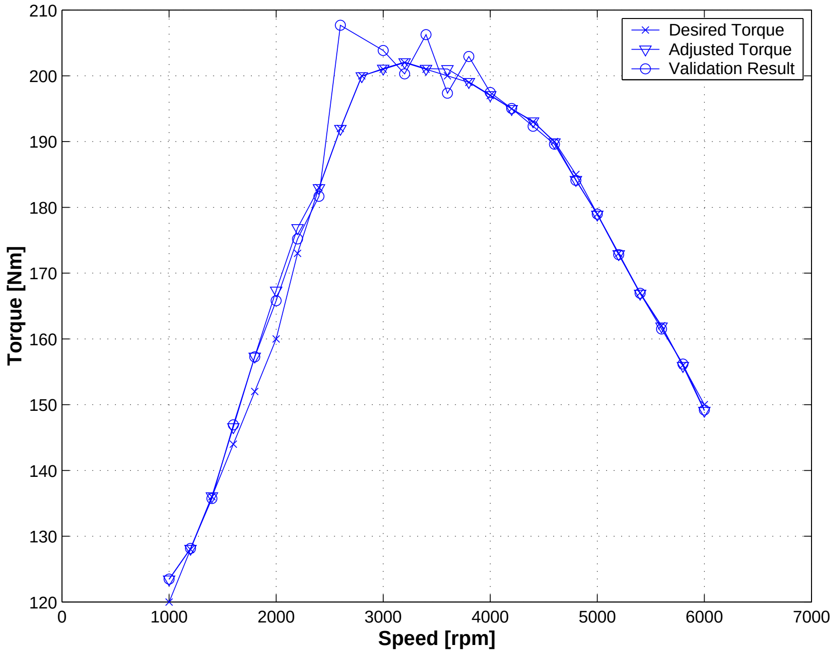


Figure 4.19: Initial comparison between desired torque and the result of the performance curve performed to validate the manual adjustment process.

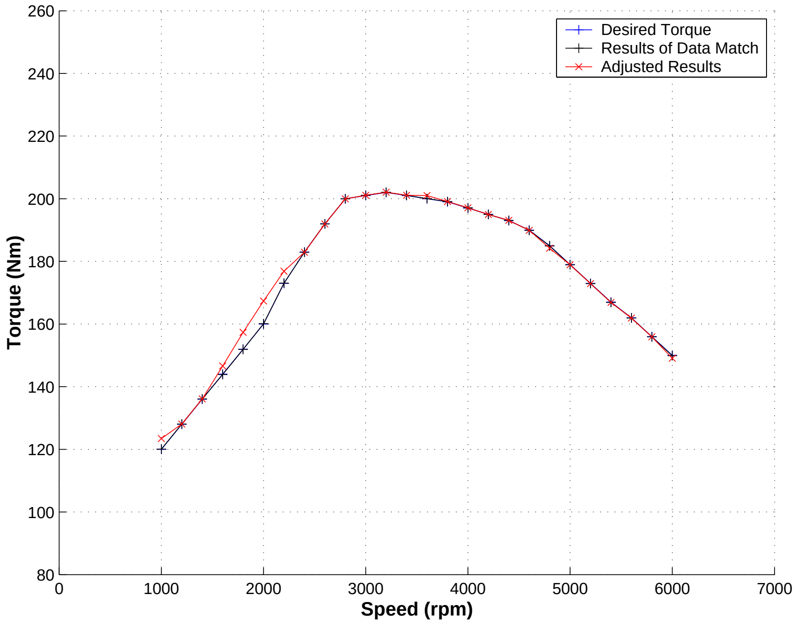


Figure 4.20: Torque after manual adjustment.

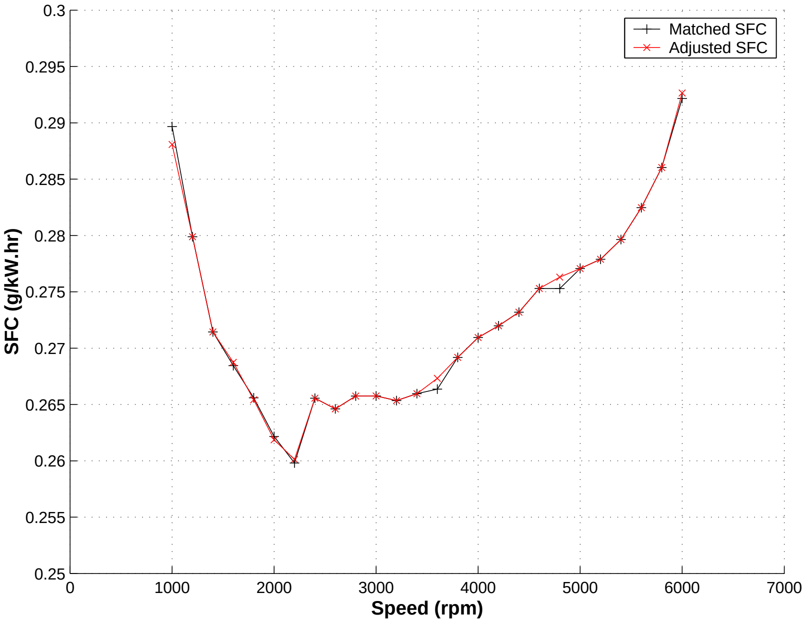


Figure 4.21: SFC after manual adjustment.

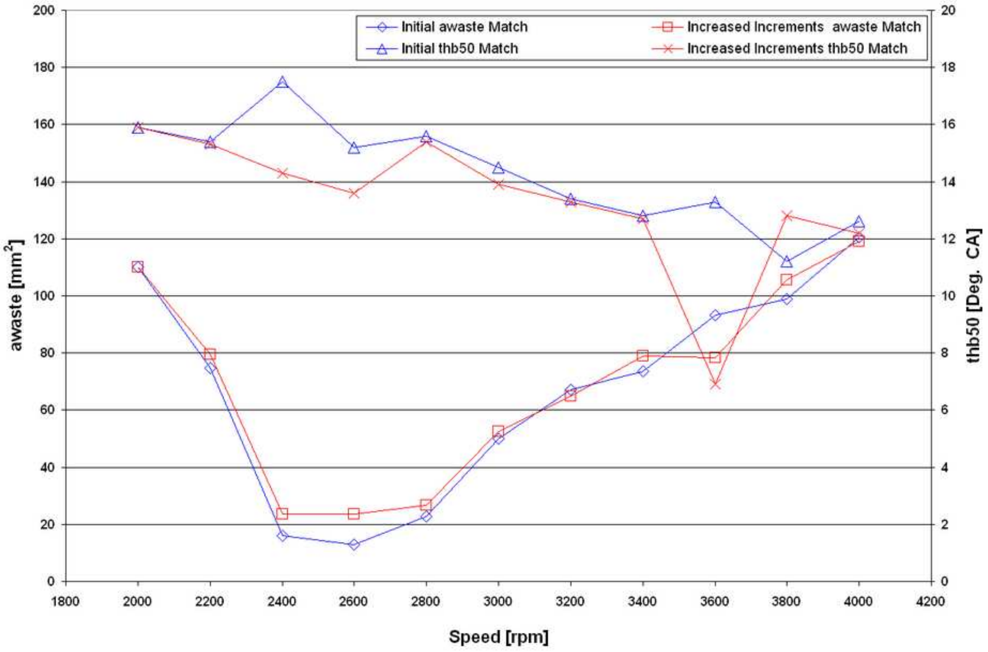


Figure 4.22: Calibration matching results from the first database compared to the results from the database with increased increments for Variable A (*awaste*).

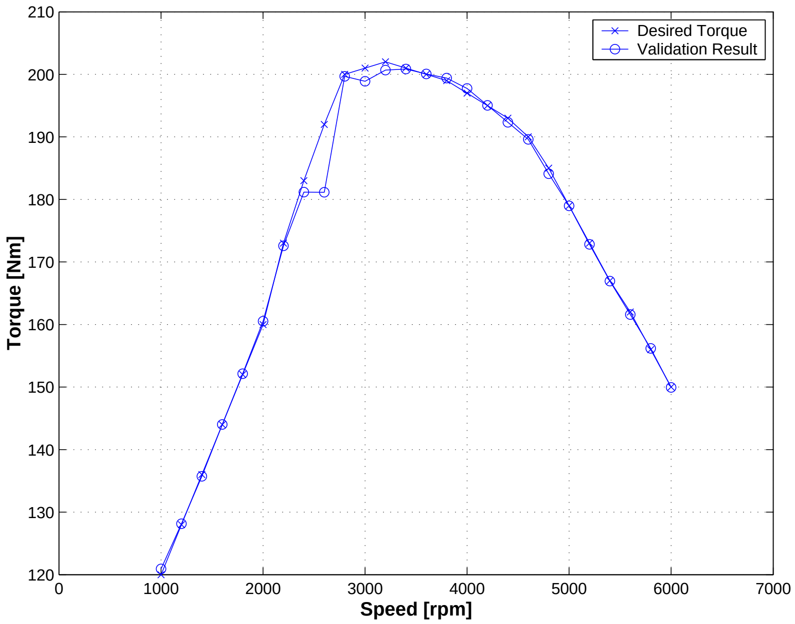


Figure 4.23: Improved comparison between desired torque and the result of the performance curve performed to validate the calibration matching process with more simulation points between 2200 rpm and 4000 rpm.

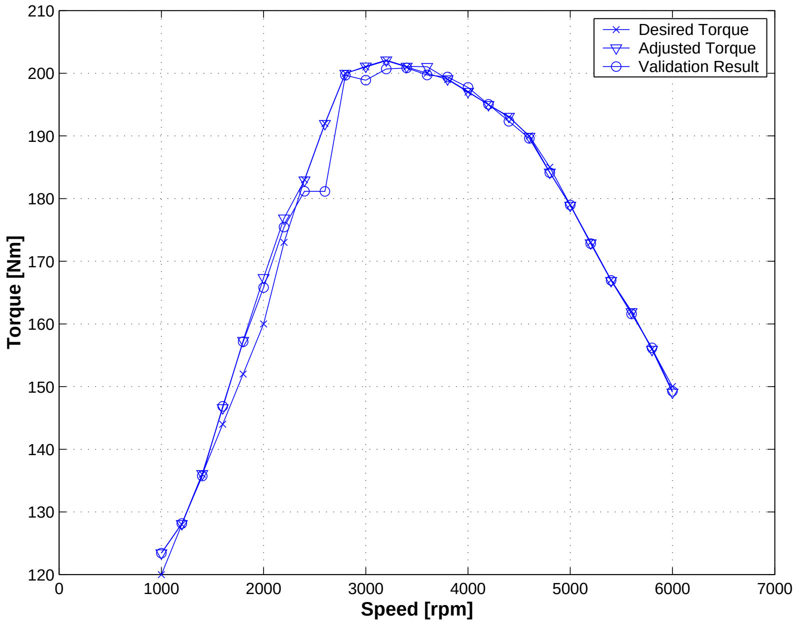


Figure 4.24: Improved comparison between desired torque and the result of the performance curve performed to validate the manual adjustment process with more simulation points between 2200 rpm and 4000 rpm.

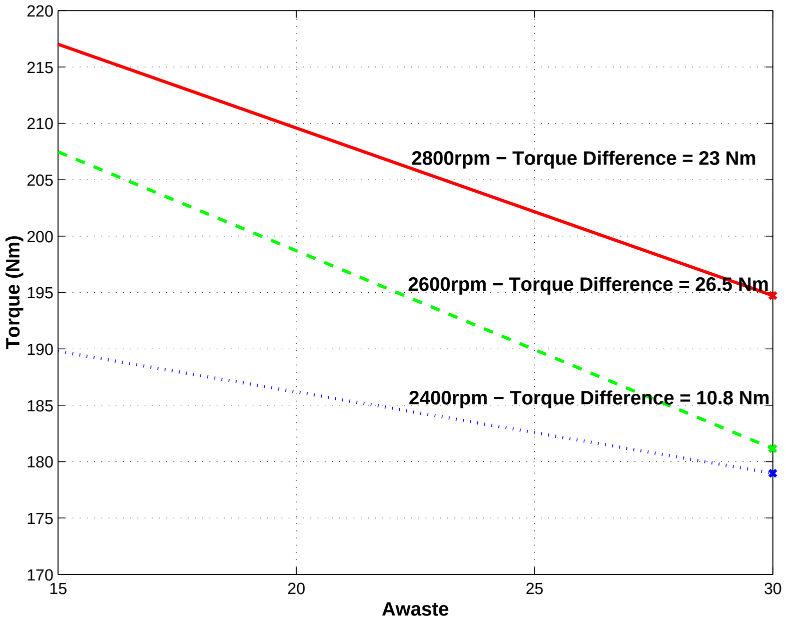


Figure 4.25: Illustration of the increased sensitivity to a change in wastegate area at 2600 rpm compared to 2400 rpm and 2800 rpm.

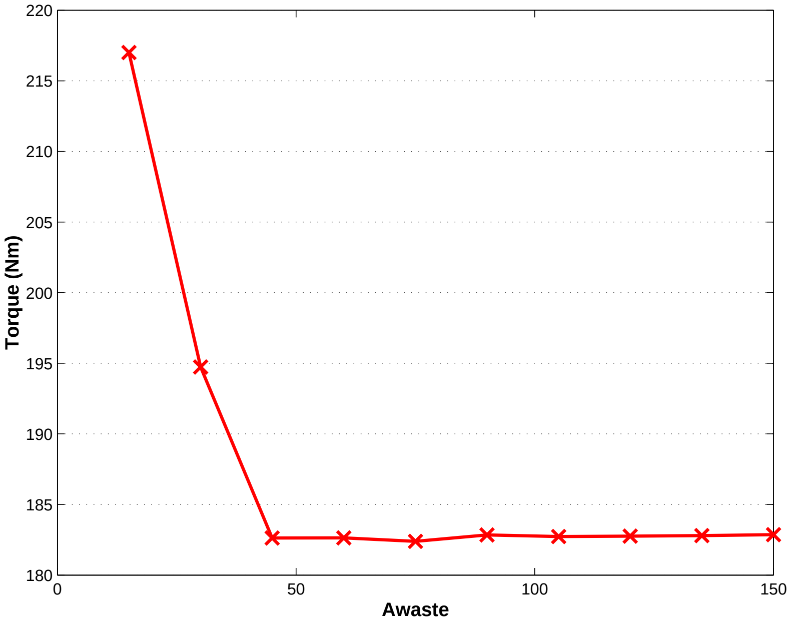


Figure 4.26: Example of the effect of wastegate area on simulated torque at 2800 rpm and constant spark advance, showing the potential resolution problem at small wastegate areas.

4.6 Storing the Results

Evaluation of Function 8

The results of the calibration matching process, as well any manual adjustments made by the user, were automatically stored by AutoCal as they were calculated. This data was saved in both csv and Matlab .mat format. The csv files were used to create the inputs for the validation runs that resulted in the data shown in Figures 4.18, 4.19, 4.23 and 4.24. The **Store Results and Exit** button will also store the results available in the Matlab Workspace and close the AutoCal program.

4.7 Evaluation Conclusion

This chapter has shown that the AutoCal program satisfactorily performs all the functions defined during the design process. It has therefore achieved the main goal of the project, namely the development of a simulation based calibration process, for turbo-charger boost pressure and spark timing, on a turbo-charged spark ignition engine. In so doing, it has also answered the following research questions posed at the start of the project:

1. Is it possible to use engine simulation software to assist the calibration engineer in the process of calibrating a map-based engine control system?
2. Can a software tool be created to analyse the data from engine testing and simulation and find the ECU calibration values to deliver a desired torque output, subject to certain operating constraints?

Chapter 5

Conclusion

Due to the high demands placed on the modern internal combustion engine control systems, especially in terms of low emissions and high efficiency, these systems are becoming increasingly complex. Together with this, the amount of time and resources required to calibrate these control systems is also increasing, as discussed in Section 2.1. In order to address this issue the following questions were asked:

1. Is it possible to use engine simulation software to assist the calibration engineer in the process of calibrating a map-based engine control system?
2. Can a software tool be created to analyse the data from engine testing and simulation and find the ECU calibration values to deliver a desired torque output, subject to certain operating constraints?

To answer these questions the following objectives were set:

1. Create and validate a simulation model of a turbo-charged internal-combustion spark-ignition engine, using a commercially available engine simulation package.
2. Develop a technique for using a software program to find the most suited calibration values of the turbo-charger boost and ignition maps. Specifically for a desired, full-load torque curve, subject to pre-determined constraints such as air fuel ratio, maximum intake air temperature, maximum exhaust temperature and maximum turbo-charger speed, from a database of simulated values.
3. Simulate the performance of the engine with the calibration values from the software program and compare the results with the desired torque curve.

A fourth objective was required in order to facilitate the first of the three listed above. This was the design, assembly and testing of a turbo-charger conversion for a naturally aspirated internal-combustion engine.

The approach to each of these four objectives are described in detail in the preceding chapters and the various appendices. The following sections summarises the tasks performed and explains the extent to which each objective was achieved. The contributions made are discussed and some suggestions are made for possible further work.

5.1 Project Summary

The tasks performed and objectives achieved are discussed below. This section also reflects the general sequence in which the various objectives were completed.

5.1.1 Literature Study

The first task of the project was to conduct a study of the relevant literature. This was completed and the knowledge gained was applied to planning and performing the tasks required to meet the project objectives.

5.1.2 Engine Testing

The turbo-charging conversion of a naturally aspirated engine was performed partly to create an engine to be used as a test base for developing AutoCal and partly to gain knowledge in the field of turbo-charging internal combustion engines. As shown in Appendix A, both of these objectives were achieved and it is felt that much more knowledge was gained than would have been the case if an already turbo-charged engine, or a naturally-aspirated engine had been used.

Figure A.11 shows that the turbo-charging conversion has the potential to increased the torque output of the engine by between 24% and 71% over the evaluated speed range. These results were achieved while gathering data for calibrating the WAVE engine model and the calibration values used were therefore not optimised. Proper calibration of the engine control system will result in better torque curve characteristics, especially with respect to smoothness and the speed at which the maximum torque is delivered. If this calibration process is to be performed, the first step would be to determine the values for the various base maps using AutoCal.

The hardware components used for the turbo-charging conversion were sourced and assembled with limited resources, especially time and money. Giving more attention to these components and replacing the turbo-charger with a unit properly matched to this engine will also result in significant improvements of the performance.

5.1.3 Engine Modelling

The process of modelling the test engine, as detailed in Appendix B, consisted of adapting an existing model of the naturally aspirated engine and calibrating the resulting turbo-charged model. This involved modelling the additional components used for the turbo-charging, as well as the changes made to the engine and adding them to the existing model. These steps were completed and the model was used to develop and test the AutoCal program.

A software program and the related processes were also developed to process test data and use it to run simulations with the same input values as those used during the testing. The results of these simulations were then used in various formats to evaluate the accuracy of the WAVE model. The model was then calibrated using theory of internal-combustion engines and gas flow, together with the procedures recommended by the suppliers of the modelling software.

The calibration process was only followed until the main outputs were accurate to within 10 % to 20 % of the test data. Due mainly to time constraints and uncertainty about the accuracy of the available valve flow coefficients and turbocharger flow maps, as explained in Appendix B.4.2. This was deemed sufficiently accurate for development and evaluation of the AutoCal program. Other authors, including (Osborne, 1999) and (Rask and Snellnau, 2004), have shown that with proper resources it is possible to achieve simulations accurate enough to deliver calibration data that can be used directly on an actual engine with good results.

This part of the project provided advanced insight into the events taking place inside an internal combustion engine. It also re-enforced existing knowledge on the interaction of engine components with each other as well as with the gasses and liquids passing through the engine. Creating the turbo-charged model required careful study of the process of modelling turbo machines and especially how to interpret the characteristic curves of compressors and turbines. Finally the process of working through the recommended steps for calibrating a WAVE model demonstrated the discipline and care required to create a model that produces useful results.

5.1.4 Simulation Based Calibration System - AutoCal

To achieve the second objective, the AutoCal program was created using Matlab. In Section 4 this program was shown to be able to receive instructions from a user and use these instructions to define and perform a series of simulations. The results of these simulations were combined in a database reflecting the effect on the outputs of the simulated engine of various combinations of turbo-charger wastegate opening area and ignition angle settings. This set of results is called the calibration database.

The AutoCal program was then used to evaluate the calibration database and find the wastegate opening area and ignition angle settings that would result in a torque output matching a desired torque curve supplied by the user. To increase the accuracy of this process the database was expanded using interpolation. If multiple combinations of wastegate area and ignition angle matching a specific torque value were found, the combination with the lowest specific fuel consumption would be selected. The evaluation process included filtering out any values exceeding the pre-defined maximum limits of various engine operating parameters.

Simulation offers the advantages of being more cost and time effective compared to testing, once an engine model has been created and validated. But in some cases simulation might not be the most cost effective solution considering the cost of the software, the time spent to create the model and the fact that a test engine will have to be available to create a usefully accurate model. The cost of the testing equipment used to record the combustion data of the test engine, required for a truly accurate model is also prohibitive. These considerations could possibly make it necessary to create the calibration database from test data. The test cell control system can be configured to interface with the ECU and programmed to run the engine at the various combinations of inputs to create the database. Care must be taken to avoid running the engine at possibly damaging operating points, which is a concern that does not exist with simulation. To provide for having to use test data AutoCal was programmed to be able to process data created from engine testing as well as simulation.

After finding the calibration settings matching the desired torque curve, AutoCal displays the desired and matched torque curves on the same axes together with explanations if any of the speed points could not be matched closely enough. The GUI used for this display is also used to display the calibration settings used to achieve the final match.

The user is then given the option of manually adjusting the results of the matching process. This option is used when the minimum SFC combinations selected by AutoCal do not result in a smooth calibration curve, or when the user wants to evaluate the effect of different calibration settings than those automatically selected by AutoCal. If a combination of ignition angle and wastegate area results in a slightly lower SFC value than a different ignition value combined with the same wastegate area, AutoCal will select the lower SFC combination. This type of situation can sometimes result in jaggedness in the curve of a particular calibration variable plotted against speed. The user can then manually smooth this out without serious compromise to the SFC. Section 4.4 illustrates this process.

In order to achieve the third objective the results of the matching process, which consisted largely of interpolated data, were used as inputs for a performance curve simulation. The results of this validation process, performed automatically by AutoCal, showed good agreement to the original desired torque curve. The accuracy was improved by increasing the amount of increments used when defining the wastegate opening area values for the simulations. This illustrated the fact that a compromise has to be reached between accuracy and simulation time when creating the database. (Rask and Snellnau, 2004) also concluded that, due to the highly non-linear behavior of engine operating characteristics, large databases may be necessary to achieve accurate calibration maps.

This project concentrated on the calibration of the base maps of the turbo-charger and ignition control systems but the AutoCal program is able to create a calibration database reflecting the effect of any other calibration settings included in the engine model. The process can also be extended to include operating regions other than the full load region concentrated on during this project.

5.2 Contribution of Thesis

The thesis mainly contributed to the understanding of modern electronic engine control systems as well as the process of calibrating them using a non-conventional, resource conserving, technique. The methodology demonstrated that the AutoCal program, when developed further, could become an indispensable part of the process to calibrate engine control systems using fewer resources, in less time, while meeting stricter performance requirements.

AutoCal can save time by running simulations during non-working hours and storing the simulation results of various different calibration variables in a database. This allows the evaluation of different calibration settings without time consuming re-simulation. It can therefore also increase the value gained from a single modelling software licence by freeing up the licence for other simulations while the database is being processed.

It was also shown that SBC offers the benefit of being able to run multiple simulations once the engine model has been created and validated. This allows the evaluation of the effect of changes to various parameters and inputs without the costs and time delays associated with testing. In the case of a proper engine development project, information required to create an accurate model such as the valve flow coefficients will be available. An indicating system will also be used to gather combustion data which will greatly assist the process of validating the model. Once the model has been validated, the engine will in theory only have to run again to evaluate the calibration created by the SBC system. Another major advantage, also stated by (Osborne, 1999), is the ability to analyse many of the inter-dependant engine system parameters that are difficult to measure during actual testing.

The AutoCal program can also be used to simulate the performance of the engine at much finer speed increments than would be practical during actual testing. This data can then be used to determine the most sensitive speed points in the operating region and allow the calibration engineer to configure the maps in the control system to have finer speed increments around these points, thereby ensuring accurate control.

Additionally the thesis also provided valuable knowledge and experience in the field of engine modelling and simulation. Being able to create accurate engine simulations is becoming more and more important with the increasing need to develop engines and engine components in as little time as possible, while using fewer resources.

In conclusion, it can be stated that the completed work provides positive answers to both the research questions.

5.3 Further Work

This project concentrated only on the steady state, full load operating region of the engine in order to demonstrate the concept of simulation based calibration. The first step forward would therefore be to adapt the AutoCal program to allow calibration of the relevant maps over the entire operating range of the engine.

To improve the results and limit the amount of manual user adjustment required, a smoothing function can be incorporated in the calibration matching process. This would give priority to a smooth calibration curve over finding the calibration values yielding the absolute minimum SFC value.

This project was focused on the calibration of the steady state performance of the engine. A dynamic load model could also be incorporated in the simulation and strategies developed for calibrating transient behavior. This would consist of fine tuning the base maps to provide a better starting point for use during transient conditions as well as developing and calibrating transient control strategies.

Bibliography

Andersson, P. (2001). Comparison of two exhaust manifold pressure estimation methods. (Cited on page 13.)

Andersson, P., Eriksson, L. and Nielsen, L. (1999). Modeling and architecture examples of model based engine control. (Cited on pages 6 and 7.)

Barr, M. (2001 September). Pulse width modulation. *Embedded Systems Programming*, pp. 103–104. (Cited on page 24.)

Bergström, A. (2003). *Torque Modeling and Control of a Variable Compression Engine*. Master's thesis, Linköpings Univeristet. (Cited on pages 1, 15, and 21.)

Bester, W.W. (2005). *Application of Turbochargers in Spark Ignition Passenger Vehicles*. Master's thesis, University of Stellenbosch. (Cited on page 103.)

Bidian, P., Tatar, M., Cascio, F., Theseider Dupré, D., Sachenbacher, M., Weber, R. and Carlén, C. (2003). Powertrain diagnostics a model based approach. (Cited on page 13.)

Blair, G. and Drouin, F.M.M. (1996). Relationship between discharge coefficients and accuracy of engine simulation. *SAE 962527*. (Cited on page 120.)

Challen, B. and Stobart, R. (1998). Some more diesel engine sensors. *SAE 980167*. (Cited on pages 7 and 8.)

Ferguson, C. and Kirkpatrick, A. (2001). *Internal Combustion Engines - Applied Thermosciences*. 2nd edn. John Wiley & Sons, Inc. (Cited on pages 15, 18, and 93.)

Gerhardt, J., Hönninger, H. and Bischof, H. (1998). A new approach to functional and software structure for engine management systems - bosch me7. *SAE 980801*. (Cited on pages xv, 9, 10, and 11.)

Heintz, N., Mews, M., Stier, G., Beaumont, A. and Noble, A. (2001). An approach to torque-based engine management systems. *SAE 2001-01-0269*. (Cited on pages xv, 9, 10, 13, and 25.)

Heywood, J.B. (1988). *Internal Combustion Engine Fundamentals*. McGraw-Hill. (Cited on pages 12, 15, 16, 17, and 44.)

Jung, M., Ford, R.G., Glover, K., Collings, N., Christen, U. and Watts, M.J. (2002). Parameterization and transient validation of a variable geometry turbocharger for mean-value modeling at low and medium speed-load points. *SAE 2002-01-2729*. (Cited on page 120.)

Mathworks (2000 November). *Using Matlab*. The Mathworks. (Cited on page 41.)

McKay, D., Nichols, G. and Schreurs, B. (2000). Delphi electronic throttle control systems for model year 2000; driver features, system security, and oem benefits. etc for the mass market. *SAE 2000-01-0556*. (Cited on page 9.)

Olin, P.M. and Maloney, P.J. (1999). Barometric pressure estimator for production engine diagnostics. *SAE 1999-01-0206*. (Cited on pages xv, 7, and 8.)

Osborne, R. (1999). Concurrent wave/matlab simulink simulation applied to hsdi diesel ecu calibration. (Cited on pages 27, 73, and 76.)

Perfect-Power (2003). *SMT6 Developers Manual*. (Not cited.)

Rask, E. and Snellnau, M. (2004). Simulation-based engine calibration: Tools, techniques, and applications. *SAE 2004-01-1264*. (Cited on pages 7, 27, 73, and 75.)

Ribbens, W.B. (2003). *Understanding Automotive Electronics*. Sixth edn. Newnes. (Cited on pages xv, 13, 16, 21, 23, and 26.)

Ricardo (2005). *WAVE v7 Basic and Engine User Manuals*. (Cited on pages 109, 110, and 117.)

Sayers, A.T. (1990). *Hydraulic and Compressible Flow Turbomachines*. (Cited on page 24.)

Steinmeyer, D. (1995). Development of a configurable electronic engine control computer. *SAE 950010*. (Cited on page 5.)

Stroh, J., Franchek, M.A. and Kerns, J.M. (2001). Eliminating maps from engine fueling control algorithms. *SAE 2001-01-0259*. (Cited on page 7.)

Struss, P., Sachenbacher, M. and Carlén, C. (2000). Insights from building a prototype for model-based on-board diagnosis of automotive systems. (Cited on page 12.)

Taim, K. (2004). *Personnal interview with Kalev Taim - Engine modeller at CAE.* (Cited on page 120.)

Watson, N. and Janota, M. (1984). *Turbocharging the Internal Combustion Engine*. Macmillan. (Cited on pages 16 and 24.)

Appendices

Appendix A

Engine Testing

A.1 Engine and Component Sourcing

The 1,6 L 4-cylinder naturally-aspirated spark-ignition engine used during the project was made available by VWSA, together with an air-to-water intercooler. A turbo-charger from a previous turbo-charging project was used and the rest of the materials and components were either bought-in, or taken from the stores at CAE. A complete functional analysis was performed for the test engine to determine the list of required components and define the tasks involved from engine build, through to testing.

A.2 Engine Build

Description of Engine Build

The test engine had been used extensively for testing at CAE but its history before this project was not known in detail. As preparation for this project it was stripped and all the components cleaned and examined for wear, before being rebuilt. The pistons were found to have approximately half a millimeter of carbon deposit on their upper surface as well as some deposits in the ring grooves and on the piston top lands. Since new pistons and rings were not available it was decided to clean the pistons and rings and reuse them. To provide additional piston cooling to compensate for the extra heat of combustion with turbo-charging the engine block was replaced with a new block from an engine with a slightly different specification. This block was fitted with jets that spray oil onto the underside of the pistons while the engine is running in order to conduct more heat away from the pistons. The only other change to the engine was to fit a thicker cylinder head gasket to lower the compression ratio of the engine slightly to compensate for the increased intake pressure due to turbo-charging. The standard cylinder head gaskets are made up of three layers of coated metal riveted together. The centre, thicker layer is flat and coated with a black substance that bonds to the other two layers when heated by the engine, thus forming the seal between the layers. The two outer layers have pressed-in grooves which flatten when the gasket is clamped between the engine block and the cylinder head, creating a seal between the gasket and the two surfaces above and below it. The thickness of the cylinder head gasket was increased by disassembling two new gaskets and combining the middle sections of the two with the outer sections of only one. This increased the thickness from 1,2 mm to 2 mm and lowered the compression ratio from 10,05 : 1 to 9,22 : 1. An oil cooler was fitted between the oil filter and the oil filter bracket to provide additional cooling of the engine oil. The following components were also replaced to ensure that the engine performed reliably for the duration of the project:

- Spark plugs
- Oil filter
- Front crankshaft seal
- Stretch bolts
- Critical gaskets
- Crankshaft bearings.

A.2.1 External Engine Modifications

Building a turbo-charged engine was not one of the main goals of this project. The reason for adding a turbo-charger and separate control system to a naturally aspirated engine was twofold. The first reason was to clearly separate the control of the turbo-charger boost from the ignition timing, in order to demonstrate the AutoCal calibration software.

The second reason was that a basic, separate turbo-charger control system like the one used in this project is a convenient tool for demonstrating map based control systems and their calibration. It offers a clear view of the plant, sensors, actuators and control computer without being cluttered by interaction with other parameters being controlled by the same computer, as is the case with the rest of the control system used on the project engine.

By taking the above into account when planning the engine modifications, the approach taken was to use as many standard and available components and keep the resources spent on the engine to a minimum. This approach led to a combination of components that were not exactly matched but still fulfilled all the requirements for the test engine.

Exhaust Gas Path

Instead of having a new exhaust manifold made up for the project, the standard manifold was inverted so that its outlet pointed upwards. An adapter was manufactured to place the turbo-charger at a convenient height to allow the lubricating oil to drain back to the sump. This adapter also channelled the exhaust gas from the twin outlets of the standard manifold into the single inlet of the turbine, see Figure A.1. A downpipe was also manufactured to receive the exhaust gas after it had passed through the turbine and channel it through an exhaust silencer into the exhaust stack on the roof above the test cell. Figure A.2 shows this arrangement fitted to the engine in the test cell.

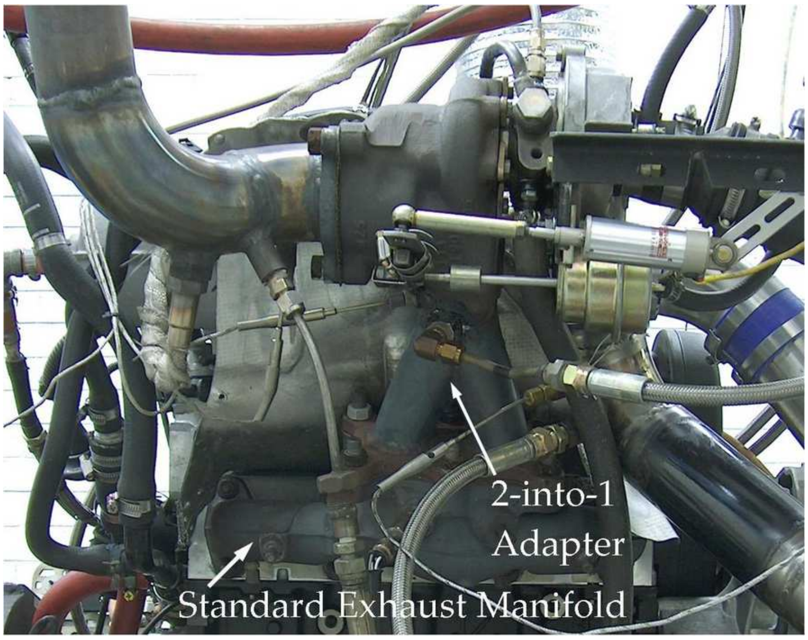


Figure A.1: The side of the test engine showing the standard exhaust manifold in inverted position and the 2-into-1 adapter.

Turbo-charger Oil Supply

In order to lubricate the bearing which supports the shaft connecting the turbine and compressor, the turbo-charger needs a constant supply of clean oil from the engine. A T-piece was installed between the oil pressure sensor and its normal location on the oil filter bracket and a small diameter braided hose was run from there to the oil inlet on the turbo-charger. This location was chosen for the following reasons:

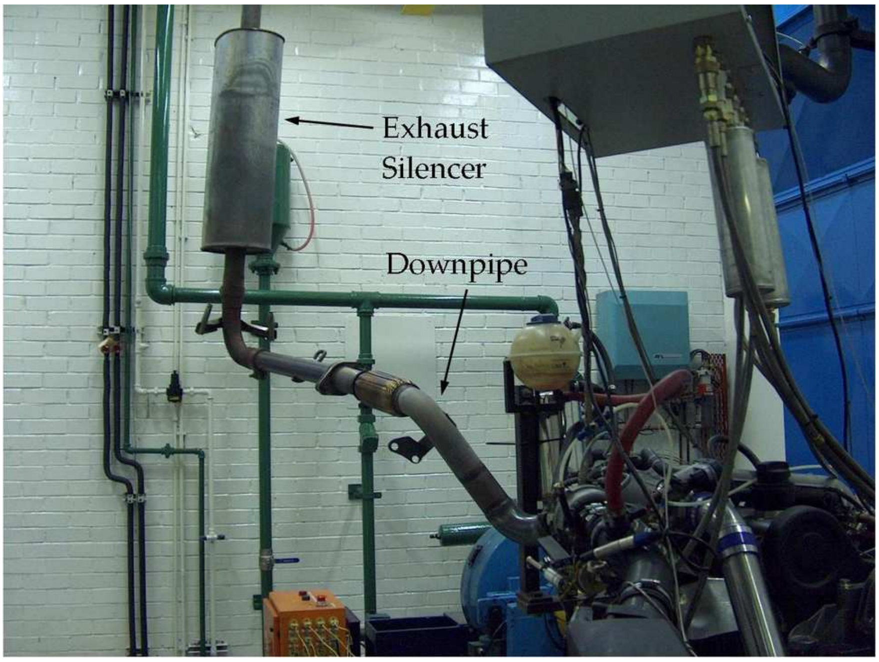


Figure A.2: The exhaust system from the turbo-charger to the outlet in the roof of the test cell.

- This location is close to the oil pump outlet and thus provides oil to the turbo-charger bearings at high pressure.
- It is situated just after the filter so the oil should contain the minimum amount of foreign particles.
- It required no machining.
- It ensures that there will be a warning when the oil pressure drops since it is right next to the oil pressure sensor.

To drain the oil a mild steel pipe with an inside diameter of 10mm was welded into the sump at an angle that would allow the oil to flow freely into the sump without damming up (Figure A.3). If the oil was to dam up it could be forced through the seals of the compressor and turbine, foul the intake system and be burned in the engine and exhaust. A flange was manufactured to fit onto the oil-drain hole on the underside of the turbo-charger bearing housing. Heat resistant rubber hose was fitted between this flange and the pipe welded into the sump.

Bearing Coolant Supply

In addition to the oil lubrication, this model of turbo-charger has the feature of cooling the bearings with a separate water jacket around the bearing section. The plumbing of the engine cooling system was adapted to allow the circulation of

Figure A.3: Oil drain adapter welded into sump with drain pipe from turbo-charger attached.

engine coolant through this water jacket. The coolant pipes are shown connected to the turbo-charger in Figure A.4.

Air Intake System

The air intake system is defined here as all the components through which the air has to pass before entering the turbo-charger compressor. This consisted of a laminar air flow meter, the airbox with filter from the standard engine and various sections of clean-air ducting. Figure A.5 shows how the components were connected to the engine.

Compressed Air Path

To handle the increased temperature and pressure of the air after the compressor the intake air ducting was manufactured from stainless steel pipe with an outer diameter of 63,5 mm. The various straight sections and bends were joined together with high temperature and high pressure resistant silicone rubber pipe. The ducting directs the compressed air from the compressor outlet to the water-to-air inter-cooler and from the inter-cooler to the throttle body on the intake manifold. The engine, turbo-charger and the related piping can be seen in Figure A.6.

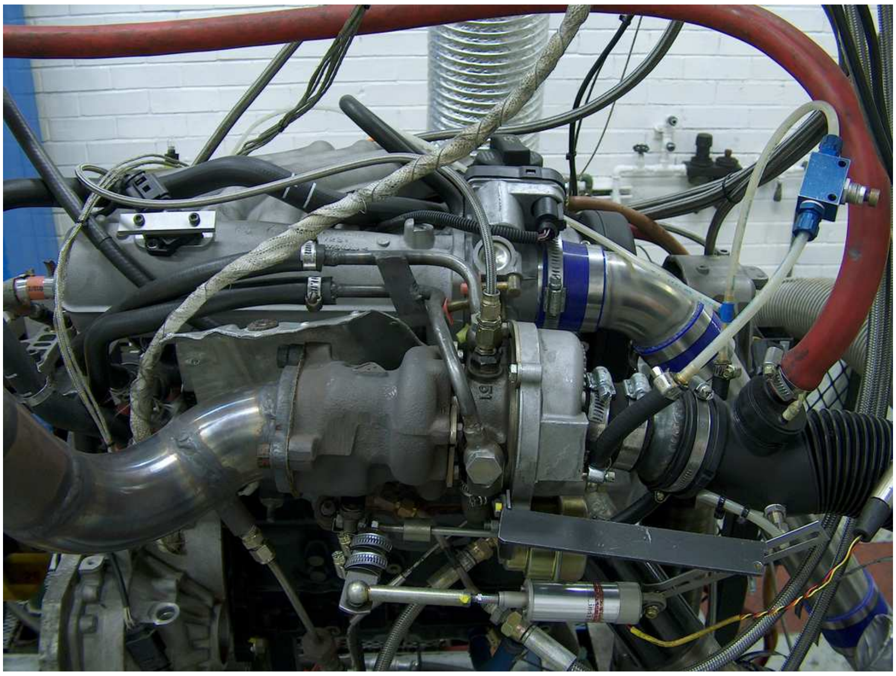


Figure A.4: Coolant and oil supply pipes bolted to turbo-charger bearing housing. The oil supply pipe is the braided hose entering at the centre of the housing.

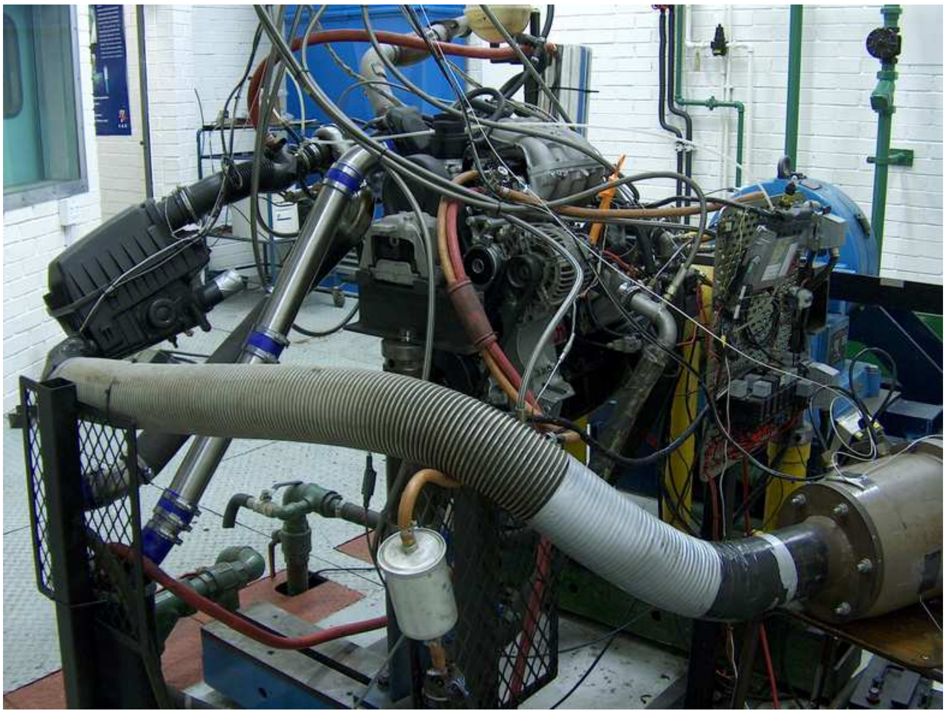


Figure A.5: Air intake components. The laminar air flow meter can be seen on the bottom right-hand side of the picture and the airbox with filter the left-hand side.

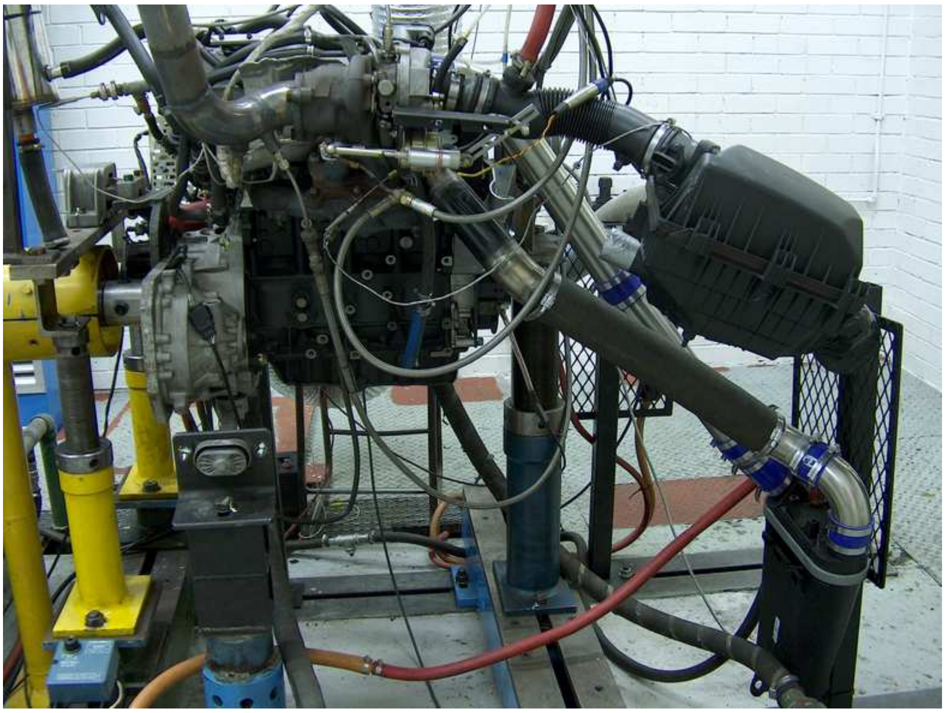


Figure A.6: An overall view of the test engine with the air-to-water inter-cooler in the bottom right-hand corner.

Inter-cooler Water Supply

The inter-cooler used in this project is a heat exchanger with compressed air on the warm side and water on the cool side. To supply the water, a line was tapped into the cooling water supply for the dynamometer and connected to the water inlet of the inter-cooler. The inter-cooler water outlet was routed through an electronically controlled valve and into the return of the dynamometer cooling water supply. The electronically controlled valve was used to control the temperature of the air entering the engine by adjusting the water flow through the inter-cooler.

Oil Separator

During normal engine operation a small amount of combustion gas passes by the piston rings and into the crank-case. This gas is commonly referred to as blow-by and must be vented from the crank-case to minimise contamination of the engine oil and prevent a build-up of pressure. Due to strict emissions regulations the blow-by gas may not be vented straight into the atmosphere. The normal practice is to route all blow-by gas into the intake system so as to be burned before passing through the catalytic converter.

On a turbo-charged engine it is required to enter the intake system before the turbo-charger to ensure a pressure drop between the crank-case and the intake air stream. This blow-by gas contains oil and other contaminants from the crank-case and to prevent this from fouling the turbo-charger and inter-cooler a oil separator was employed. This consisted of a stainless steel container filled with steel-wool with a gas inlet near the bottom and an outlet at the top. The blow-by gas was routed through this container and upon coming into contact with the steel wool, it would cool down and any oil would condense and flow down to the bottom of the container and into a small reservoir. After passing through the oil separator the blow-by gas entered the intake air stream just before the compressor inlet as in the case of production vehicles.

A.2.2 Changes to Standard Engine Control System

The standard engine control system had to be altered slightly to handle the increase in intake manifold pressure with turbo-charging. On a naturally aspirated engine the pressure in the manifold will never exceed atmospheric pressure. Therefore the Temperature and Manifold Absolute Pressure (TMAP) sensor and its related calibration maps in the ECU are only calibrated up to atmospheric pressure.

The standard TMAP sensor on the test engine was replaced with unit having a measuring range from 20 kPa to 300 kPa. The maps that convert the voltage signal from the TMAP sensor to a pressure value in the ECU were adjusted with the Siemens Application and Measurement (SAM 2000) system to reflect the characteristics of the high pressure sensor.

To enable correct operation, all the maps in the ECU software using the inlet pressure as a reference would have to be modified to reflect the higher maximum intake pressure. This involves changing the reference values on the intake pressure axes of the various maps, as well as populating the maps with new calibration values once these have been determined, either manually or by an automatic calibration program such as AutoCal.

During testing the fuel injection quantity and ignition timing was set manually using a facility provided by the SAM 2000 software and the special ECU used. This procedure, discussed further in Sections A.3.3 and A.3.4, was used to collect the data required for validating the model used by the AutoCal program when determining the values required to populate the maps.

A.2.3 Turbo-charger Boost Control System

The turbo-charger control system on the test engine consisted of the following components:

- Perfect Power SMT-6 map-based control computer.
- A 12 V solenoid valve of the type that is closed when not energised and open when energised.
- An adjustable needle valve.
- An assortment of pipes and fittings to connect the components.

The diagram in Figure A.7 illustrates how these components were integrated with the test engine.

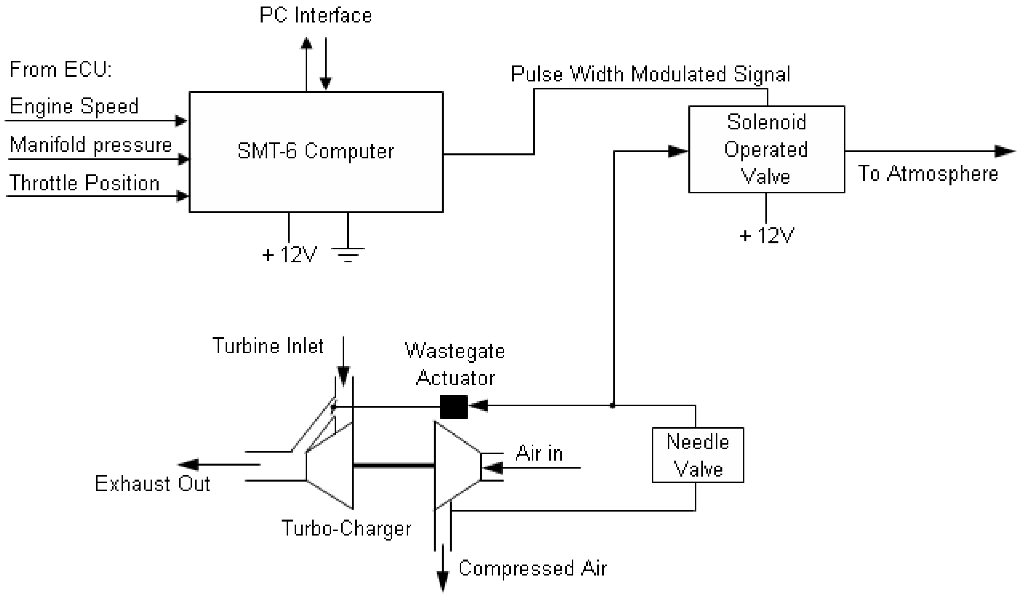


Figure A.7: Diagram of the system used to control turbo-charger boost pressure.

The turbo-charger boost control system functions by controlling the pressure acting on the diaphragm of the wastegate actuator. This then affects the position of the wastegate and the amount of exhaust energy available to act on the turbine. In a system without electronic control an air line would be connected directly between the compressor outlet and the wastegate actuator. Once the compressed air reaches a certain pre-set pressure, the diaphragm would start moving and forcing the wastegate open. This will then allow exhaust gas to by-pass the turbine and decrease the amount of energy available to drive the compressor. The compressor outlet pressure will therefore be limited.

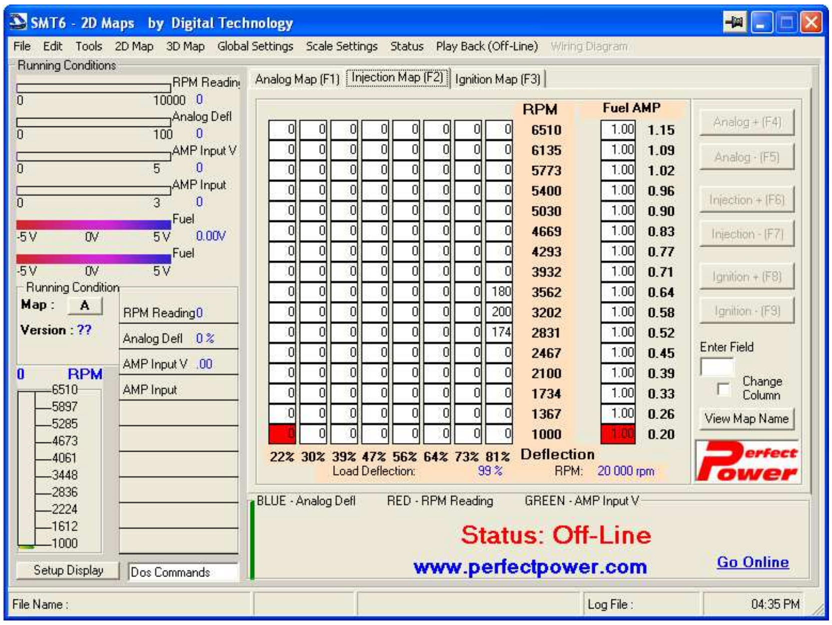
As can be seen in the diagram in Figure A.7, the solenoid actuated valve of the electronically controlled system can allow some of the air in the line between the compressor outlet and wastegate actuator to escape. This has the effect of lowering the pressure acting on the wastegate and delaying the opening of the wastegate until the compressor outlet has reached a higher pressure than when the solenoid valve is closed, as defined by the control system.

The compressor outlet pressure can therefore be controlled by adjusting the duty cycle of the signal sent to the solenoid valve. The higher the duty cycle of the pulse width modulated signal, the lower the pressure on the diaphragm for the same compressor outlet pressure. The needle valve limits the amount of air flowing into the control circuit which allows the solenoid valve to control the pressure by venting only small amounts of air. It also prevents unnecessary waste of compressed air. For this project an adjustable needle valve was used to allow fine tuning of the control system behaviour. On production vehicles a fixed restriction orifice is placed in the pipe connected to the compressor outlet.

For this project the Pulse Width Modulated (PWM) signal output of the Perfect Power SMT-6 computer was used to drive the solenoid valve. The unit is programmed with adjustable maps which can be configured through the accompanying SMT6 Win software (Figure A.8). It uses a speed signal from the engine ECU, or a dedicated pickup, as one of the references for the maps. The other main map reference is the throttle position which is also sourced from the engine ECU and used as an indication of engine load. The absolute manifold pressure can be used as a reference for an additional multiplicative factor on top of the map value referenced by speed and throttle position.

To adjust the duty cycle of the PWM signal, or the length of time that the solenoid valve stays open every time it is opened, the user has to adjust the map value for the current operating point. The SMT-6 unit will then use this number to adjust the duty cycle or width (in time) of the pulse used to open the solenoid valve. The turbo-charger control can thus be manually "mapped" by taking the engine to every operating point on the map and adjusting the value until the desired compressor outlet pressure is achieved.

This is a resource intensive task and the AutoCal program was created to perform this process automatically with simulated data. In order to generate data for calibration of the simulation model used by AutoCal this process was performed manually for a selection of points, as described in Section A.3.4 below.



RPM Reading 0
Analog Defl 0
AMP Input V 0
AMP Input 0
Fuel 0.00V
Fuel

Running Condition
Map : A
Version : ??
0 RPM

RPM Reading 0
Analog Defl 0 %
AMP Input V .00
AMP Input

								RPM	Fuel AMP	
0	0	0	0	0	0	0	0	6510	1.00	1.15
0	0	0	0	0	0	0	0	6135	1.00	1.09
0	0	0	0	0	0	0	0	5773	1.00	1.02
0	0	0	0	0	0	0	0	5400	1.00	0.96
0	0	0	0	0	0	0	0	5030	1.00	0.90
0	0	0	0	0	0	0	0	4669	1.00	0.83
0	0	0	0	0	0	0	0	4293	1.00	0.77
0	0	0	0	0	0	0	0	3932	1.00	0.71
0	0	0	0	0	0	0	180	3562	1.00	0.64
0	0	0	0	0	0	0	200	3202	1.00	0.58
0	0	0	0	0	0	0	174	2831	1.00	0.52
0	0	0	0	0	0	0	0	2467	1.00	0.45
0	0	0	0	0	0	0	0	2100	1.00	0.39
0	0	0	0	0	0	0	0	1734	1.00	0.33
0	0	0	0	0	0	0	0	1367	1.00	0.26
0	0	0	0	0	0	0	0	1000	1.00	0.20
22%	30%	39%	47%	56%	64%	73%	81%	Deflection		

Load Deflection: 99 %
RPM: 20 000 rpm

Status: Off-Line
www.perfectpower.com

File Name :
Log File :
04:35 PM

Figure A.8: Screenshot of the SMT6 Win program used to configure the turbo-charger control computer.

A.3 Testing

In addition to the SMT6 Win interface described in Section A.2.3 above, two other software packages and associated hardware were used during the project. This section describes the test equipment and set-up as well as the management of the engine and test cell during testing. It then goes on to cover the naturally aspirated and forced induction testing and the processing of the data.

A.3.1 Test Cell Equipment

Dynamometer

The engine was connected to a Froude EC38 eddy-current dynamometer to apply load to the engine and absorb the power from the engine. The dynamometer can be used to control either the engine speed or the torque. If the engine controls are adjusted to increase the engine power output in speed control mode, the dynamometer load is increased to keep the speed constant. In torque control mode, a constant torque (break mean effective pressure (BMEP)) will be applied by the dynamometer and the engine speed will increase until the friction mean effective pressure (FMEP) has increased by an amount equal to the increase in the net indicated mean effective pressure (IMEP) (Ferguson and Kirkpatrick, 2001). In other words the engine speed will stabilise where the IMEP minus the FMEP is equal to the BMEP.

Engine and Dynamometer Cooling System

The cooling system in the test cell is part of a larger cooling system servicing all the test cells in the facility. This main cooling system consists of a reservoir of cooling water which is circulated by pumps through the test cells that are in use. Two cooling towers are used to cool the water in the reservoir once a certain temperature threshold is exceeded. Inside the test cell the cooling water is circulated through the cooling channels of the dynamometer and through the cool side of the heat exchanger for the separate engine coolant circuit. This heat exchanger fulfils the function of the radiator in a motor vehicle but with its significantly higher cooling capacity it enables accurate control of the engine coolant temperature to a range of specified temperatures. This capability is used during special tests when various operating conditions need to be simulated.

Test Cell Ventilation System

Each test cell in the facility is serviced by central extraction and supply systems. The extraction system prevents the buildup of exhaust gasses in the test cell and helps to cool the cell by extracting the warm air. The cooling system also has fans to supply fresh air from outside the facility into the test cells.

Intake Air Cooling System

In order to simulate various operating conditions in the test cell it is necessary to be able to control the engine air intake temperature and humidity. To achieve this the intake air is passed through an air conditioning unit. This unit has a cooling and a heating section. The cooling section consists of a air to water heat exchanger which is supplied with chilled water from a central chiller plant. The flow of chilled water can be controlled by the test cell control system. The heating section comprises a duct with a electric heating element which is also controlled by the test cell control system. By using different combinations of heating and cooling the temperature and humidity of the air can be controlled independently to each be at a defined value. The temperature and humidity of the intake air is determined by a thermocouple and humidity sensor placed at the engine air intake.

Test Cell Control System

The test cell control system consists of the following items:

- Sensors and actuators in and around the engine and test cell.
- A computer running the Engine Test Automation (ETA) software which sets the actuators using the information from the sensors and user inputs. ETA software also records the measured data. It is covered in more detail below.
- A Programmable Logic Controller (PLC) acting as interface between the computer and the sensors and actuators.

The function of these components is explained in the following section.

A.3.2 Engine Set-Up

Figure A.9 shows the test engine installed in the test cell. From left to right in the picture is the heat exchanger for the engine cooling system, the dynamometer, the shaft guard around the shaft connecting the engine and dynamometer and the engine. Also visible are the exhaust system and the housing unit for the pressure sensors and the connectors for the other sensors and actuators used by the test cell control system.

Mounting the Engine

It can be seen in Figure A.9 that the engine itself is mounted on a combination of sliding cross-bars, screw-jacks and custom made brackets. The engine is connected to this combination with the brackets that the engine would be mounted on if it were installed in a vehicle engine bay. The engine brackets provides the damping required to isolate the engine vibrations from the rest of the setup and ensure that the engine vibrations are representative of a vehicle installation.

Connecting Engine to Dynamometer

To connect the engine output to the dynamometer a custom made drive shaft, similar to the units used on front-wheel-drive motor vehicles, is used . The shaft consists of a solid bar with two constant velocity joints at the ends. This arrangement allows for slight misalignment between the engine output and the dynamometer input. For safety reasons a shaft guard is placed around the drive shaft.

Figure A.9: The test engine installed in the test cell.

Connection to the Cooling System

For this test the engine cooling system was connected as it would be in a vehicle, excepting the following differences:

- The normal air-to-water heat exchanger is replaced by the water-to-water unit seen on the left in Figure A.9
- The pipes between the engine and the heat exchanger are significantly longer than in a vehicle
- The thermostat normally fitted to the engine cooling system to control the temperature to a certain set-point was removed. Its function was replaced by the test cell control system using a combination of thermocouples to sense the temperature and a bypass mixing valve to control the temperature.

The standard water pump installed in the engine was used to circulate the engine coolant through the engine and heat exchanger. On some tests an additional external pump is used to enable rapid cooling of the engine.

Fuel Supply

Fuel is pumped to the test cell from a central fuel store. Upon entering the test cell it flows into a mass-based flow measuring unit and then on to the engine. The fuel supply system after the flow measuring unit is also very similar to the system used in a vehicle and consists of a filter, high pressure fuel pump, pressure regulator, return line and vapour disposal system.

Exhaust System

The engine exhaust pipe is connected to the test cell exhaust system which exits through the test cell roof. After exiting through the test cell roof, the exhaust gas passes through another silencer before exiting the main roof of the facility.

Sensors

The sensor inputs for the PLC is situated in the housing unit overhanging the engine in Figure A.9. There are a number of inputs for thermocouples and pressure sensors, as well as a number of additional analog and digital inputs. The number and combination of the various inputs depends on the particular PLC configuration.

Thermocouples are inserted into the fluid or gas streams to be measured and plugged into the junction box. The pressure sensors are mounted in the junction box and connected to their respective measuring points using suitable pipes. The aluminium cans seen in the figure provide damping volumes which mechanically filter out the pulsations inherent to gas flow into and out of reciprocating internal combustion engines. This allows a more accurate measurement of the average pressure in the intake and exhaust systems.

Actuators

The test cell control system controls various actuators in order to control the engine operating conditions during a test. The signal sent to these actuators can be grouped into digital and analog signals.

The digital signals are used to switch the power supply to the various electrical devices in the test cell on and off. These include the fuel and water pumps, the cooling fans, the engine starter motor and the intake air heater. It is also used to switch the engine ignition on or off. Analog output signals are used to control actuators such as flow control valves and the throttle actuator.

Calibration and Set-up Checklist

To ensure accurate test results and safe operation of the engine, the sensors and actuators have to be accurately calibrated before any testing can take place. It is also very important that all the mounting bolts, pipe clamps and other fittings have been fastened properly to avoid anything coming loose while the engine is running. As an example, a water pipe that comes loose during testing can cause serious damage to the expensive electronic devices in the test cell.

To manage this an extensive test set-up, calibration and inspection procedure is followed. This procedure documents all the set-up details and instructions and requires the sign-off of all the different actions before testing can take place. It also ensures that the correct components are used in the set-up and that the test cell control system is configured correctly for every particular test.

A.3.3 Test Management

The engine testing was managed from the test cell control room using three software packages installed on two computers. These three packages are discussed below.

ETA

ETA is a engine test cell management program developed by CAE. Its most basic function is to enable an operator to 'drive' an engine installed in a test cell. It interfaces with the PLC software to read the various sensors in the test cell and control the actuators. Its displays user requested information on screen and can be configured to record any of the measured parameters as well as calculated values. Its GUI further allows the user to switch devices in the test cell on and off, as well as adjust the set-points of the various controlled parameters.

The majority of engine tests are a combination of various operating states that have to be performed in a certain sequence and under certain conditions. A typical test contains a large amount of different test points and one sequence of an endurance test can take up to four hours or even longer to complete. This makes it impractical to perform manually. ETA enables the automation of this process by allowing the user to compile a test program which contains details of the various engine operating states (load, speed etc.), as well as the associated operating conditions. Automating a test ensures good repeatability and allows efficient use of manpower since one test technician can supervise more than one test at a time.

For this project ETA was used to drive the engine to the various operating points where data was recorded for the validation of the WAVE model. At these points the recording facility was utilised to record the values of the required sensors. The channels to be recorded as well as the sampling rate and duration of recording can be set in ETA. The data is stored in a database on the test computer and can be exported from there into a comma separated value file.

ECU Calibration and Analysis Software

The ECU used during this project was a special unit that allows access to the calibration values stored on it through additional circuits. A special cable and data card is used to connect it to a computer with the SAM 2000 calibration and analysis software installed.

With this software it is possible to view all the calibration tables and constants stored in the memory of the ECU. These values can then be adjusted to achieve a desired engine response. The control strategies on the ECU can also be overridden by the operator and the signal to the actuators can be set to a desired value. This facility was used during this project to set the ignition timing when recording the validation data.

During the process of calibrating an ECU for use on a new engine or to incorporate changes to an existing engine, this software would be used to adjust all the calibration values until the desired engine behaviour has been achieved. The data would then be used to create a dataset that will be programmed onto the standard production ECUs to be fitted to that particular engine in production.

The software also allows the user access to the values of all the sensors used by the ECU as well as any calculated values used in the control strategies. This access is given in the form of various display options including a simple value display as well as plotting facilities. These values can also be recorded at adjustable sampling rates and stored on the computer for later analysis.

The diagnostic functions of the ECU are also accessible with this software. This facility is used during fault finding and to clear any errors logged on the ECU after correcting the cause of the problem.

During the validation testing the software was used to adjust ignition timing and AFR as well as to set the electronically controlled throttle to the fully open position. It was also used to monitor knock by displaying the processed signal from the knock sensor. Various ECU parameters were recorded at each of the test points and analysed for use in the validation process.

SMT6 Win

This program controls the functions of the SMT6 computer used to control the wastegate. The SMT6 computer uses map based control and the maps containing the required actuator settings for each operating point can be adjusted with the SMT6 Win program.

During the validation testing the SMT6 controller was configured to drive the solenoid valve controlling the wastegate actuator. According to the SMT6 Developers Manual, the unit closes the circuit shown in Figure A.10 at a frequency of 38 Hz with a duty cycle determined by using Equation A.3.1 below.

$$DutyCycle = \left(\frac{38\,\mathrm{Hz} \times MAP \times 0{,}1\,\mathrm{ms} \times FAMP}{10} \right) \tag{A.3.1}$$

where

MAP	The map value at the current operating point, found by referencing the map using engine speed and throttle position;
FAMP	A correction factor based of the current manifold pressure

For example, if the current value of the map value was 120, the duty cycle according to Equation A.3.1 would be 45.6% assuming that the FAMP correction factor was set to 1. The solenoid valve would therefore be open 45.6% of the time.

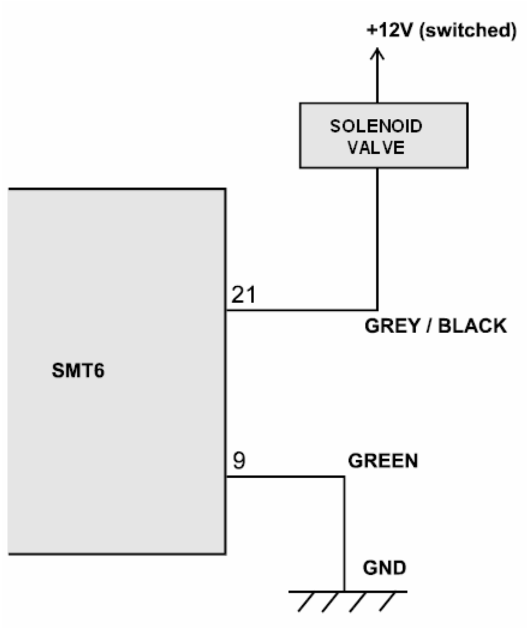


Figure A.10: Wiring configuration of SMT6 to control a solenoid valve.

A.3.4 Forced Induction Testing

The purpose of testing the turbo-charged engine was to gather data about the behaviour of the system (engine, turbo-charger, boost controller) under different boost and ignition timing settings across the wide open throttle operating range of the engine. This data was required to calibrate and validate the computer models of the system so that accurate simulation based calibration of the control system could be performed.

The amount of testing required to collect sufficient data for model calibration and validation is significantly lower than what would be required if the control system was to be calibrated manually. Data was recorded at three different boost levels. At each boost level between one and three different ignition values were tested depending on operating constraints such as exhaust temperature and knock. This project focused on wide open throttle performance but the techniques developed can be adapted for use with part load work, thus enabling significant savings of cost, time and other resources, by avoiding the need for extensive testing over the entire operating range of the engine.

The turbo-charged performance testing was performed as follows:

- Low boost - compressor outlet was connected directly to the wastegate actuator to ensure that the wastegate would open at the lowest pressure possible, which would result in the lowest boost developing. The throttle was opened fully and engine speed ramped up in 200 rpm intervals. At each speed point a course timing swing was performed. The swing started from a setting that resulted in the maximum allowable exhaust temperature and was advanced in steps of approximately 3° Crank Angle (CA) until two or three different settings were recorded, or up to the onset of knock. At each point, engine data was recorded for 30 seconds with both ETA and SAM.
- High Boost - The same procedure as for low boost was followed, except that the boost control system (described in Section A.2.3) was set to enable the highest possible boost level permitted by exhaust temperature and a maximum torque level of 200 Nm.
- Intermediate Boost - The same procedure was again followed, only this time the boost pressure was set to a level between the high and low boost levels.

The resulting collection of data consisted of up to nine points of data for every speed point tested. Some of the speed points had less than nine data points due to testing not being possible at all nine points. This was due to either of the limits for maximum knock, torque, or exhaust temperature being reached while performing the timing swing. Some data points were also eliminated due to measurement errors.

Figure A.11 shows the maximum performance measured during the testing process compared to the performance of the engine before fitting the turbo-charger. This illustrates the effect of turbo-charging.

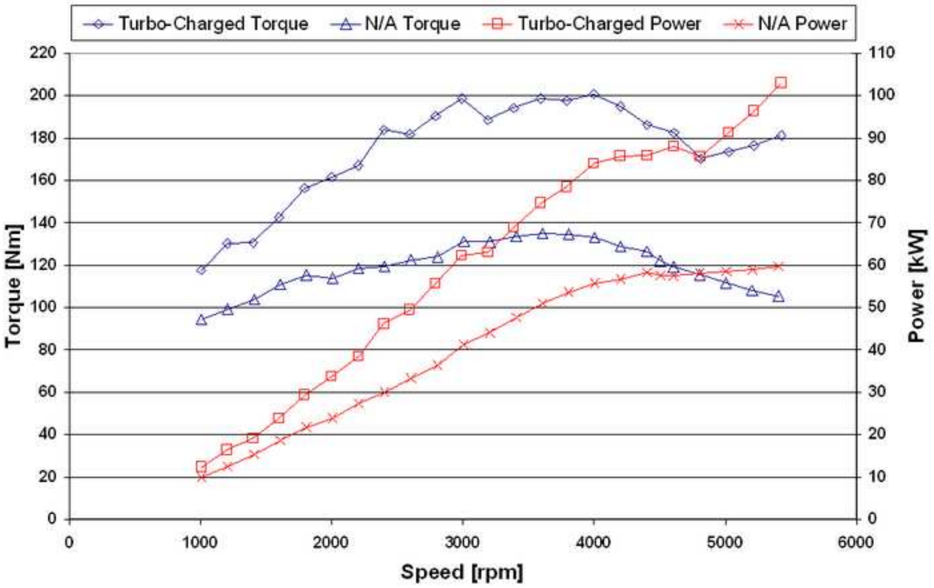


Figure A.11: The maximum torque and power achieved during turbo-charged testing compared to the torque and power of the engine before fitting the turbo-charger.

Data Processing

The process of calibrating the WAVE model as described in Section B.4 required test data for the following variables:

- Volumetric Efficiency
- Intake Manifold Pressure
- Exhaust Back Pressure
- Exhaust Temperature
- Torque
- Specific Fuel Consumption

The data was processed according to the same principle used by the AutoCal program when processing simulation results. All the data belonging to a particular speed point is handled as a group separately from the other speed points. These groups of measured data can then be conveniently processed further to extract specific information required when calibrating and validating the WAVE model.

An example of this is Figure A.11 which was created by sorting all the columns in the different speed groups to reflect ascending torque values. The upper most row of each speed group was then plotted. This information can then be compared to the results of WAVE simulations using the same boost control, ignition angle and AFR settings. A comparison such as this, over the entire tested speed range of the engine can be used to identify speed-specific problems with the WAVE model.

Once the problem speed points have been identified the data comparison process can be concentrated around the specific speed point by creating graphs of the various data points in each speed point (Figures B.9 to B.16). Once again these can be compared to results of WAVE simulations using the same settings as were used on the test engine when testing the specific speed point. The process of comparing results and drawing conclusions is covered in detail in Appendix B.4.

In the case of the calibration process being extended to other than full-load regions the data for a particular speed would be grouped according to throttle position as well as speed. This will allow comparison to data from simulations using the same throttle area, as well as the other variables mentioned.

A.3.5 Characterisation of the Boost control system

In order to calibrate the SMT6 computer it was necessary to know the relationship between the SMT6 map setting and the wastegate area at various operating conditions. This is due to the fact that the WAVE model uses wastegate area as an input into the turbo-charger model and thus the output of the simulation based calibration system will be the wastegate area values that will deliver the desired torque curve. These area values would then have to be converted to values for the SMT6 calibration map.

During a previous project using the same turbo-charger (Bester, 2005), it was determined that the relationship between wastegate area and actuator rod displacement is given by Equation A.3.2. This equation was determined using a combination of measurements of the geometry of the wastegate components and the actuator rod displacement versus pressure on the diaphragm. It was also found that the area has a maximum value of 490 mm^2, which is the area of the wastegate orifice. The wastegate area reaches this value when the actuator rod has travelled 75% of its total reach. This occurs when the pressure acting on the wastegate actuator diaphragm has reached 55 kPa.

$$WastegateArea = 2 \times \pi \times Radius \times ArmRatio \times Displacement \qquad \text{(A.3.2)}$$

where

Radius	The radius of the wastegate orifice;
Arm Ratio	The ratio of the distance from the actuator rod attachment point to the wastegate swingarm pivot point, to the distance from the wastegate to the same pivot point;
Displacement	The wastegate actuator rod displacement.

To characterise the boost control system on the test engine it was decide to measure the displacement of the wastegate actuator rod versus the SMT6 map value. To measure the actuator rod displacement a linear transducer was mounted on the turbo-charger and connected to the wastegate actuator rod, as shown in Figure A.12. The output of this transducer was recorded as a percentage with 0 % equalling no movement and 100 % equalling the fully extended position. The displacement in the fully extended position was measured at 18,3 mm.

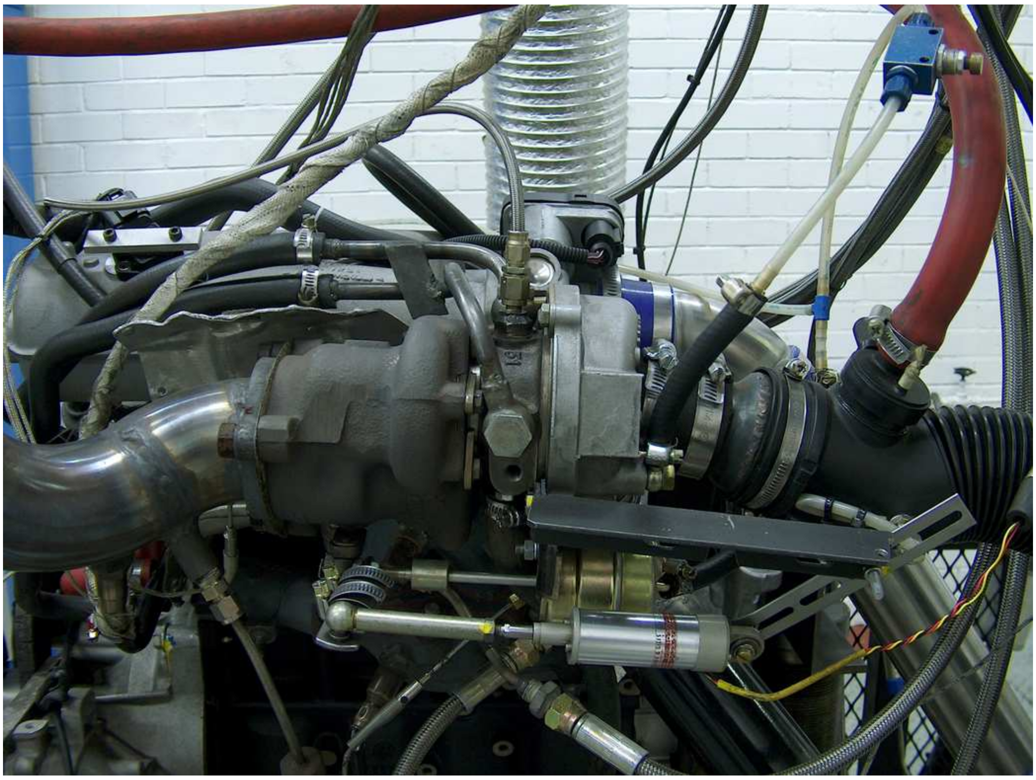


Figure A.12: The linear transducer used to measure the movement of the wastegate actuator rod mounted on the turbo-charger.

At a safe ignition timing value, the boost pressure was increased from minimum to either 0,5 bar or the maximum value achievable at that point, by adjusting the SMT6 map value. The SMT6 setting at which the wastegate started moving was noted and from then on the SMT6 map value was increased by 10 integer values at a time and data sampled for 5 seconds at each point at 10 Hz. This test was repeated at 2600 rpm, 3000 rpm and 3600 rpm.

From the recorded data it was noted that there is a linear relationship between the SMT6 map value and the wastegate actuator rod displacement but that the values differed for every speed point. To find an general equation that could describe this relationship the following procedure was followed:

1. A linear trendline was found for each of the three groups of data.
2. The factor and offset of each of the equations was plotted against their respective speed points and two polynomial equations in terms of engine speed found that fit through all three points in the two separate graphs.
3. The polynomial equations were then used as the factor and the offset of a new speed-dependent linear equation describing the relationship between SMT6 map setting and actuator rod displacement (Equation A.3.3).

$$\begin{aligned} SMT6 = \big(-4.57E-6 \times Speed^2 + 3.25E-2 \times Speed - 70.1\big) \times Displacement \\ + \big(-1.97E-5 \times Speed^2 + 2.35E-1 \times Speed - 139\big) \end{aligned} \tag{A.3.3}$$

The results of this process are plotted together in Figure A.13. It can be seen that the equation provides an acceptable match for the three speed points tested. Note that some of the points at the 3000 rpm were retested at a different ignition timing value, resulting in the slight change in actuator displacement.

The relationship between wastegate opening area and the SMT6 map value of a particular operating point can be found by using the following steps:

1. Find the displacement value of the opening area by solving Equation A.3.2 for displacement.
2. Convert the displacement distance to percentage by dividing by the total displacement of 18,3 mm and multiplying by 100%.
3. Convert the displacement percentage to an SMT6 map value with Equation A.3.3.

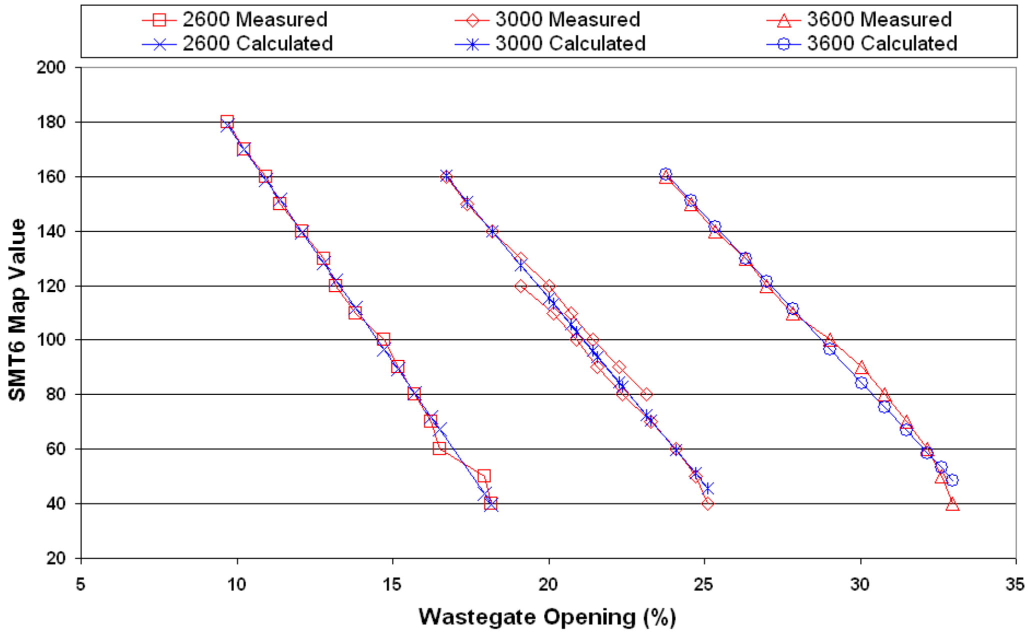


Figure A.13: Results from the boost control system characterisation process.

A.3.6 Summary of Engine Instrumentation

Monitored variables and sensors used:

- Fuel consumption - mass-based fuel totaliser.
- Mass air flow - calculated by ECU using speed-density method as well as measured using orifice plate air-flow meter.
- Intake air humidity - electronic humidity sensor.
- Mean value of the actual ignition angle - value read from ECU.
- Mean value of the ignition angle adjustment due to detected knock - value read from ECU.
- Lambda set point - value read from ECU.
- Measured lambda value after turbine - linear lambda probe and lambda scanner.
- Engine Speed - inductive pickup and toothed wheel mounted on dynamometer shaft.
- Torque - load cell connected to arm of dynamometer.
- Power - calculated by ETA using engine speed and torque values.
- Throttle position - value read from ECU.

- Wastegate opening percentage - linear transducer.
- Intake air temperature at air filter - thermocouple.
- Pre-intercooler inlet air pressure and temperature - thermocouple and pressure transducer with damping volume in-line between the transducer and the inlet air pipe.
- Inlet manifold pressure (post-intercooler inlet air pressure) and temperature - thermocouple and pressure transducer with damping volume in-line between the transducer and the inlet air pipe.
- Pre-turbine exhaust gas pressure and temperature - thermocouple and pressure transducer with in-line damping volume.
- Post-turbine exhaust gas pressure and temperature - thermocouple and pressure transducer with in-line damping volume.
- Barometric pressure - pressure transducer mounted in test cell.
- Fuel pressure and temperature - pressure transducer and thermocouple.
- Engine coolant inlet pressure and temperature - pressure transducer and thermocouple.
- Engine coolant outlet pressure and temperature - pressure transducer and thermocouple.
- Oil temperature in sump and at oil filter - thermocouple.

Controlled variables and actuators used:

- Intake air temperature - test cell air conditioning unit.
- Throttle position - electronic throttle, controlled through the ECU.
- Engine speed and load - controlled by the eddy-current dynamometer.
- Ignition angle - controlled through the ECU.
- Fuel-air-ratio - controlled through the ECU.
- Wastegate position - Solenoid valve controlled by the SMT6 computer as discussed in Section A.3.3.
- Engine coolant temperature - electronically controlled valve.
- Intercooler water temperature - electronically controlled valve.

Appendix B

Engine Simulation

B.1 Software

The reference manual of the Ricardo WAVE engine simulation package (Ricardo, 2005), describes the WAVE program used to model the test engine as follows: "WAVE is a computer-aided engineering code developed by Ricardo to analyse the dynamics of pressure waves, mass flows and energy losses in ducts, plenums and the manifolds of various systems and machines. WAVE provides a fully integrated treatment of time-dependent fluid dynamics and thermodynamics by means of a one-dimensional formulation. This incorporates the general treatment of working fluids including air, air-hydrocarbon mixtures, combustion products, liquid fuels and Freon gases.

In addition, WAVE provides a completely coupled interface to many industry-standard Computational Fluid Dynamics (CFD) codes. This allows various system components with complex geometry or physical phenomena to be simulated as a full three-dimensional model.

Finally, WAVE provides a completely coupled interface to user-defined external models and numerous other commercial codes, which can be employed to further describe the physics of a system component or to establish controls for the WAVE system."

WAVE models general and complex compressible-flow fluid networks in terms of a set of building blocks, which include constant-area or tapered pipes/ducts, junctions of multiple ducts, orifices and termination points such as infinite plenums (ambient spaces) and anechoic boundaries.

WAVE also includes a library of special machinery components such as engine cylinders, piston compressors, turbo-chargers/super-charger compressors and turbines and pumps. There components can be attached to the pipe networks to serve as sources or absorbers of pulsating flows. These features make WAVE an excellent tool for simulating the Internal Combustion Engine as well as other complex compressible-fluid flow networks.

B.2 Building the Engine Models

WAVE is a tool used for quickly calculating fundamental physical and mathematical models. The results of these calculations are only as good as the inputs used. It is therefore the responsibility of the user to accurately enter the relevant engine information using the graphic user interface, WaveBuild supplied with WAVE. The process of entering this information will be covered in three sections, namely Gathering Data, Preparing Data and Constructing the Model as explained in the WAVE user manuals (Ricardo, 2005).

Gathering Data

A wide variety of geometric and other engine data as well as operating parameters are required by the model to calculate accurate results. This data is gathered from various sources including physical measurements of the engine components, Computer Assisted Design (CAD) drawings and data, as well as test data from the engine. Appendix C contains a list of the geometry and engine data required for the model.

A very important part of building a WAVE model is to have accurate data for the following:

1. The flow coefficients through the intake and exhaust valves.
2. The intake and exhaust valve lift profiles.
3. The turbo-charger or super-charger flow and efficiency characteristics for a forced induction engine.

If this data cannot be sourced from the manufacturer of the engine or turbocharger it will have to be measured or simulated with a CFD package from accurate geometry of the components. Both these options are very resource consuming, especially if being done for the first time. In the case of turbo-chargers it also requires very special and expensive measurement and testing equipment which might be out of reach for most organisations.

Preparing Data

Before entering any of the duct data into WAVE, it is necessary to organise how the geometry of the real system is going to be split up and modelled. The best way is as follows:

1. Make a sketch of the real system, including all of the bends, corrugated sections (flexible joints), perforates (silencer components) and volumes such as air cleaners and mufflers. This sketch will be valuable as a reference tool when entering the data into WAVE and laying out the network in a manner that is clear and representative.
2. Split the pipes into sub-lengths at convenient points such as bends, corrugated sections, changes of cross sectional area and flanges or real system joints.
3. Split air cleaner and muffler volumes into smaller sub-volumes that can be represented by one of the WAVE style volumes, usually the Y-Junction elements. Isolate other items such as inter-coolers, catalysts and perforated sections as these are more difficult to model.

Once the drawing has been split into its subsections, all the necessary geometric details such as diameters, lengths, volumes and surface area should be calculated. Following this procedure facilitates the completion of the remaining stages of the model building process in an efficient manner.

Building the Model

WaveBuild is the preprocessor used to build the geometric models and provide all the input data for physical models required to perform a WAVE analysis. WaveBuild consists of a canvas onto which a number of basic building blocks representing the geometry of a model are placed (ducts, volumes, engine cylinders, etc.). These building blocks can then be edited so that the geometry accurately represents the real system. Pull down menus and panels also allow the user to define the physical models to be used (combustion, emissions formation, injection, etc.). The entire WAVE model can be constructed through WaveBuild. Figure B.1 shows the symbolic representation of the turbo-charged engine model in the WaveBuild interface.

Values of parameters that have to be varied across the operating range of the engine are defined in a table called the Constants Table. These parameters include amongst others, starting temperatures, fuel/air ratio and burn duration. The correct adjustment of the values in this table, to accurately represent the engine and the conditions being simulated, is a very important part of the WAVE model calibration process.

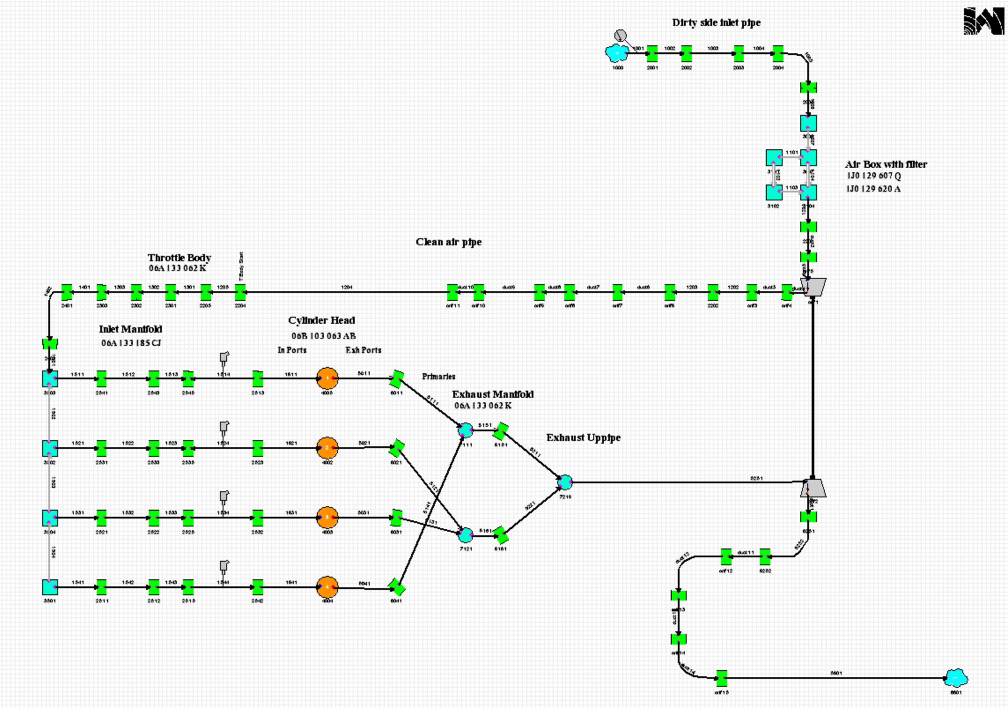


Figure B.1: The model of the turbo-charged test engine in WaveBuild.

The values of the various input parameters such as throttle position, wastegate position and ignition timing are also stored in the Constants Table. Section 3.3.2 discusses how the Constants Table is used by AutoCal to adjust the simulation inputs during the process of creating the simulation database.

Project Engine Model

The base WAVE model of the Naturally Aspirated (NA) 1595 cc four cylinder Volkswagen (VW) engine used in this project was developed during a previous engine development project at CAE. This model was examined in detail to ensure that the modelled geometry matched the test engine before being turbo-charged. The various additional components and modifications described in Appendix A.2 were then measured and defined before being added to the NA model to create the turbo-charged model. Table B.1 contains the main specifications of the test engine. The process of turbo-charging and testing the engine is discussed in Appendix A. The fuel specified for the model was Octane (C_8H_{18}) with a lower heat of combustion of 44,43 MJ/kg. The Research Octane Number (RON) defined in the setup of the knock model was 95.

Table B.1: Specifications of test engine.

Cubic capacity:	1595 cc
Number of cylinders:	4
Strokes per cycle:	4
Firing order:	1324
Valvetrain:	Single over head cam, two valves per cylinder
Bore:	81,1 mm
Stroke:	77,4 mm
TDC combustion chamber volume:	31,49 cc
Compression ratio:	$9,23:1$
Connecting rod length:	152 mm
Wrist pin offset:	0 mm
ECU:	Siemens Simos 7.4 with electronic throttle control
Fuel used:	95 RON Petrol

B.3 Model Calibration and Validation

This section discusses the procedures for model calibration provided in the online help facility of the WAVE program.

Volumetric Efficiency Calibration

The power developed by an internal-combustion engine is directly dependent on the amount of oxygen available to react with fuel during the combustion process. An important aim of any engine development program is therefore to maximise the flow of air through the engine. The volumetric efficiency is a good indication of the success of this process. It therefore represents the foundation upon which the other outputs of the engine are built and is calibrated first.

Typical factors that can negatively affect the volumetric efficiency include:

- Intake valve restriction
- The timing of the intake valve closing
- Trapped residual gasses
- Restrictions in the induction system
- Exhaust system back-pressure.

Once the WAVE model has been defined in WaveBuild a test simulation needs to be performed to gather data for the calibration process. The particulars of this process are described below in Section B.4. The online help facility of the WAVE program provides the following checklist for volumetric efficiency calibration:

- Compare volumetric efficiency vs. revolutions per minute (rpm), against measured data.
- Compare plenum pressure vs. rpm and vs. mass flow against measured data.
- Compare exhaust back pressure vs. rpm and mass flow, against measured data.

If these comparisons reveal a good match between WAVE results and measured data, no further work is needed on the calibration of volumetric efficiency response and attention can be given to calibrating the torque output. If on the other hand, the volumetric efficiency results are not satisfactory, then attention must be given to the influencing factors. These are separated into two groups depending on the findings of the comparisons:

- Low-to-medium rpm calibration.
- Medium-to-high rpm calibration.

Detailed instructions for calibrating the volumetric efficiency in these two speed-regions are provided in the online help facility of the WAVE program. A list of the important steps can be found in Appendix C.

Torque Calibration

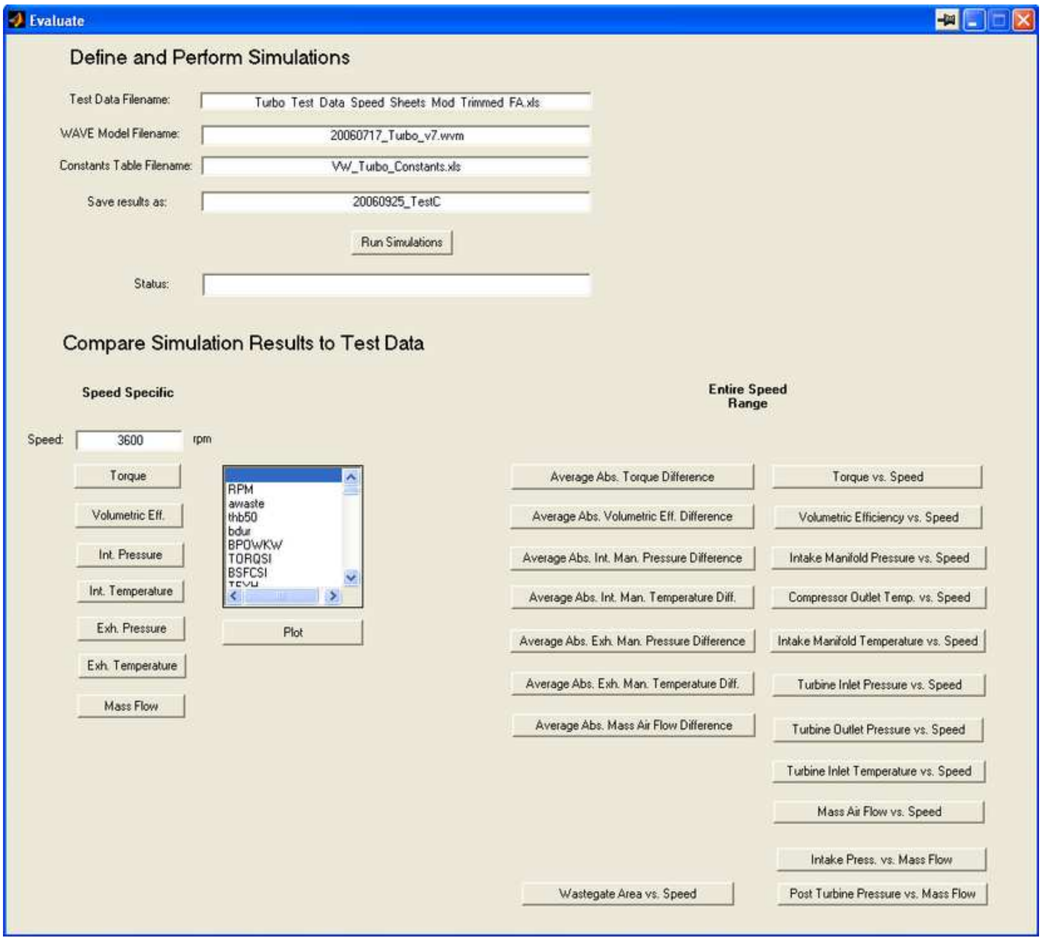
Once it has been confirmed that the WAVE model has been correctly built according to the provided guidelines and that the modelled volumetric efficiency matches the measured data from the test engine, the process of calibrating the torque results can begin. Once again the detailed procedure is presented in the WAVE user manual and some guidelines and tips from the WAVE manual can be found in Appendix C.

B.4 Calibrating the Turbo-charged Engine Model

The engine model used for evaluating the performance of the AutoCal program had to be calibrated in order to provide meaningful results. This section describes how the simulation process was managed and how the calibration procedures described above were utilised.

B.4.1 Setting up and running the simulations

In order to compare the WAVE results to the measured data from the test engine, a number of simulations had to be performed. These simulations had to represent the test engine, as well as the conditions under which the measured data was recorded. A Matlab program was created to set up and run the WAVE simulations. The GUI in Figure B.2 was programmed to facilitate the process of setting up and running the required simulations. As can be seen in the figure the required inputs are similar to those used for AutoCal (Figure 3.3) and consist of the name of the Excel spreadsheet containing the processed test data, the name of the WAVE model being calibrated and the name of the associated Constants Table. A filename for the simulation results is also required.



Test Data Filename: Turbo Test Data Speed Sheets Mod Trimmed FA.xls
WAVE Model Filename: 20060717_Turbo_v7.wvm
Constants Table Filename: VW_Turbo_Constants.xls
Save results as: 20060925_TestC
Status:
Speed: 3600 rpm

Figure B.2: The interface used during the model calibration process for setting up and running simulations, as well as comparing the results with test data.

The program code reads the values of the relevant variables such as AFR, ignition timing and wastegate setting from the test data. These values are then combined with the WAVE calibration values stored in the Constants Table to create new Constants Table entries for each test run being simulated.

The simulation runs were grouped according to the various speed increments in the performance curve, as was done with the measured test data. The simulations performed to create data for the calibration process thus consisted of up to 9 different runs for every speed increment, depending on the number of measured data points available for each particular speed increment. The reason for the variation in available measured data points is that at some of the speed increments it was not possible to test all 9 different variations of boost and timing settings, as described in Appendix A.3.4. The 2600 rpm speed point is used as an example.

Table B.2 lists the different combinations of ignition timing and wastegate opening percentage evaluated at this speed, as well as the torque output measured for each combination. It can be seen that the testing procedure covered a relatively large range of possible torque outputs.

Table B.2: Measured torque at 2600 rpm for various settings of the wastegate and ignition timing.

	Torque	Ign. Angle	Wastegate
	[Nm]	[$°CA$ BTDC]	[%]
Point 1	137.79	1.10	18.48
Point 2	143.92	3.00	18.05
Point 3	152.51	6.00	17.22
Point 4	174.80	6.00	11.11
Point 5	180.63	9.00	10.52
Point 6	185.10	12.00	10.10
Point 7	184.46	9.00	7.28

The following variables from the test data were used when defining the simulations:

- Engine speed
- Throttle angle
- Air/fuel ratio (converted to fuel/air ratio for WAVE)
- Ignition angle (converted to 50% burn point)
- Wastegate opening percentage (converted to opening area)
- Intake air temperature (before the compressor).

In some cases the format of the test data had to be converted to the input format required by WAVE. This process is discussed below for the two control variables this project focused on, namely ignition timing and wastegate position.

Ignition Timing Input

Instead of using the actual ignition advance angle as an input to determine start of combustion, WAVE uses the angle at which 50% of the combustion process has taken place relative to TDC. This is called the 50% burn point and in WAVE it is referred to as *thb50*. The ignition angle used during testing therefore had to be converted to this *thb50* value before being used as an input to the simulations. This was done by reversing the equations used in WAVE to determine the ignition angle from a *thb50* value, as described below.

The correlative combustion model for premixed charge spark ignited engine models in WAVE is based on a Wiebe function relationship widely used to describe the rate of mass burned in thermodynamic calculations. This relationship allows the independent input of function shape parameters and of burn duration. It is known to represent quite well the experimentally observed trends of combustion heat release (Ricardo, 2005). When using the Wiebe correlation, the cumulative mass fraction burned as a function of crank angle is given by the following equation.

$$W = 1 - \exp\left(-AWI\left(\frac{\Delta\Theta}{BDUR}\right)^{(WEXP+1)}\right) \tag{B.4.1}$$

where

W	is the cumalative mass fraction burned;
$\Delta\Theta$	is the crank degrees past the start of combustion;
BDUR	is the user-entered $10\% - 90\%$ burn duration in crank degrees;
WEXP	is the user-entered Wiebe exponent;
AWI	is an internally calculated parameter to allow BDUR to cover the range of $10\% - 90\%$.

Using the user-entered values WAVE calculates the start of combustion, *thign*, as follows:

$$WEXP = WEXP + 1 \tag{B.4.2}$$

$$wi = \frac{1}{WEXP} \tag{B.4.3}$$

$$y50 = 0.6931^{wi} \tag{B.4.4}$$

$$yb = \left(2.3026^{wi} - 0.1054^{wi}\right) \tag{B.4.5}$$

$$thign = thb50 - BDUR * \frac{y50}{yb} \tag{B.4.6}$$

It then follows that:

$$thb50 = -thign + BDUR * \frac{y50}{yb} \tag{B.4.7}$$

Wastegate Input

The measured wastegate opening percentage had to be converted to opening area for use in the WAVE model. To achieve this required the steps outlined in Appendix A.3.5 were followed.

Running of the Simulations

The WAVE package consists of a number of products with different functions. The two products mainly utilised during this project were the WaveBuild modelling interface and the WAVE code itself. The WAVE code is an entirely non-interactive program that reads input data and writes output data. As inputs WAVE requires the .wvm file that is created by WaveBuild, as well as the accessory files defining such items as the turbine (.trb) and the compressor (.cmp). The output files all have the same prefix as the .wvm file used as input. The following output files are created by default:

- .out file - This file contains all of the information processed from the .wvm file, the simulation run-time output and some basic results.
- .sum file - This file contains cycle-averaged values from the WAVE analysis.
- .wvd file - This file contains numerous different sets of post-processing data.

Additional output files can be requested by the user through WaveBuild. For this project a summary file containing the converged results of all the simulations run during a particular session was requested. The contents of this summary file is defined by referencing to a text file containing the names of all the simulated variables to be recorded in one space-delimited row at the top of the file. On running WAVE a new summary file is created with this row of variable names at the top. After each simulation run has converged, WAVE writes the value of each variable to the summary file in a column underneath the respective variable name. The extension used for the summary file is .stb.

WAVE runs from either a DOS command prompt, Linux/UNIX shell prompt, or is launched into either of the afore mentioned via the WaveBuild interface. The Matlab code was programmed to launch WAVE into the DOS command prompt. The procedure followed when running each of the simulations required to create the database can be summarised as follows:

1. Read the user input information which includes the name of the WAVE model and constants table to be used, as well as the speeds to run and the details of the variables to adjust.
2. Define the simulation to be run and write the .wvm WAVE input file.
3. Launch WAVE in DOS. The command used for this was:

   ```
   dos('wave Evaluate.wvm -s')
   ```

4. Wait for the simulation to converge, then read the Evaluate.stb summary file and store the data in the database.

B.4.2 Comparing the simulation results with the test data

Following the procedure prescribed in the WAVE user manual, the average values of volumetric efficiency, plenum pressure and exhaust back pressure, was compared to test data. The lower half of the GUI in Figure B.2 was programmed to facilitate this process. Examples of these results are presented in Figures B.3, B.4, B.5, B.6, B.7 and B.8 respectively. These plots facilitate the identification of the speed regions requiring the most calibration attention. It also helps to identify which strategy to use as defined in Section C.2.

After using these graphs to identify the speed points needing attention, a closer inspection can be performed by looking at only the data for those specific speed points. Data for the 3600 rpm speed point is presented as an example of a relatively good match in Figures B.9, B.10, B.11 and B.12. Figures B.13, B.14, B.15 and B.16 show the data for the 3200 rpm speed point which requires more calibration work.

The speed-specific comparisons (bar graphs) gives insight into the accuracy of the conversion of the two critical inputs, ignition angle and wastegate opening area for use in WAVE. It also makes it possible to identify problems with the turbo-charger model due to data from various turbo-charger operating conditions being available.

Having more than one data point for each speed increment enables calibration and validation of the WAVE model over a wide range of operating points around the WOT region. It is especially important for the calibration of the ignition and wastegate settings which are essential for the success of the simulation based calibration process.

The results represented in Figure B.17 and the other figures in this section indicate the accuracy of the model used to evaluate the AutoCal program. This level of accuracy was deemed sufficient for the purpose of evaluating AuoCal. For actual calibration work a higher level of accuracy will however be required. Reasons for not calibrating the WAVE model further during this project include the following:

Valve Flow Coefficients These values are used by the simulation code to determine the gas flow through the intake and exhaust valves and therefore need to be defined very accurately. The process of determining these values include measurements and mathematical post-processing which is not trivial. (Blair and Drouin, 1996) shows that the use of traditional, or ideal, discharge coefficients during engine simulation can underpredict engine torque by some 10 %, in comparison to using the actual flow coefficients. The WAVE model used for this project was a development of a naturally aspirated WAVE model of the same engine used during a previous commercial project. Since the valve flow coefficients were inherited together with this model, their accuracy , as well as whether they were actual or ideal, could not be confirmed with absolute certainty. According to (Taim, 2004) the valve flow coefficients were received from Volkswagen for use in the above mentioned commercial project. During preparation for that project, the naturally aspirated torque curve of a similar engine to the one used for this project was simulated using these coefficients. The resulting data was compared to test engine data and the valve flow coefficients slightly adapted to improve the simulation results. To confirm the accuracy of the coefficients further would have required access to an accurate flow bench, which was unfortunately not available.

Turbo-charger Maps The maps describing the flow and efficiency properties of the compressor and turbine were also inherited from a previous project and were in a different format than that which is required by WAVE. They thus had to be converted to the required format which could possibly have led to some inaccuracy. Another concern is that under low airflow conditions, the heat exchange via the turbo-charger housing affects the temperature-based measurement of the efficiencies (Jung *et al.*, 2002)). The maps supplied by manufactures therefore normally exclude the low speed regions, requiring the WAVE program to rely on extrapolation of the maps when simulating these conditions. Due to these factors the turbo-charger in the WAVE model could not be assumed to be 100% representative of the unit used on the test engine.

Wastegate Model During the evaluation it was noted that better results can be achieved by adjusting the values of the constants used when converting the measured wastegate opening percentage to wastegate area. Figure B.18 compares the wastegate areas used for determining the results presented here, to the areas calculated using the measurement results. This disagreement can be due to inaccurate measurement of these constants as well as the opening percentage, or an oversimplified approach to modelling the opening area vs. opening percentage. It shows that it is vitally important to accurately model these characteristics of engines to be calibrated using SBC.

On consideration of the above mentioned facts, it can be seen that there was not much value in spending further effort on calibration of the WAVE model during this project. It does however, highlight the importance of having accurate data for modelling and validation of the engine and control system being calibrated.

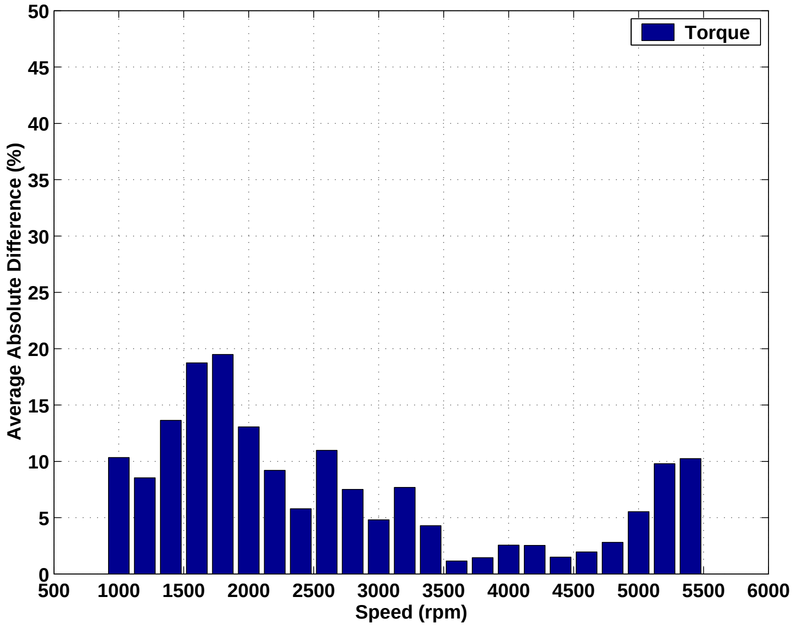


Figure B.3: Percentage difference between average torque of simulated and test data over the entire speed range.

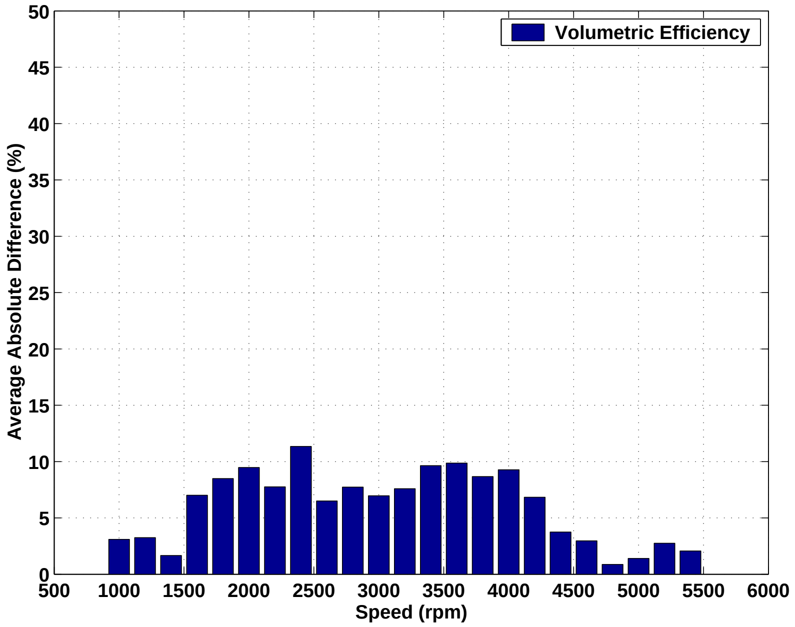


Figure B.4: Percentage difference between average volumetric efficiency of simulated and test data over the entire speed range.

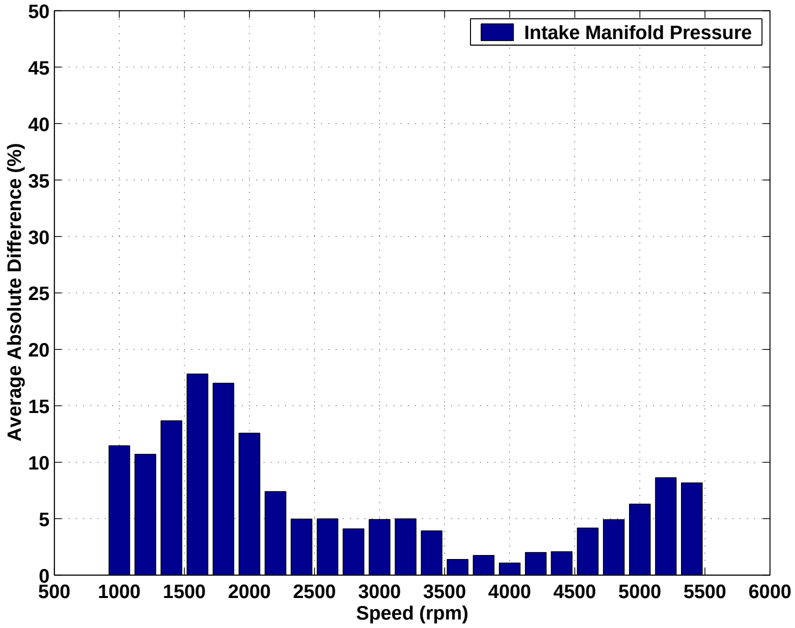


Figure B.5: Percentage difference between average intake manifold pressure of simulated and test data over the entire speed range.

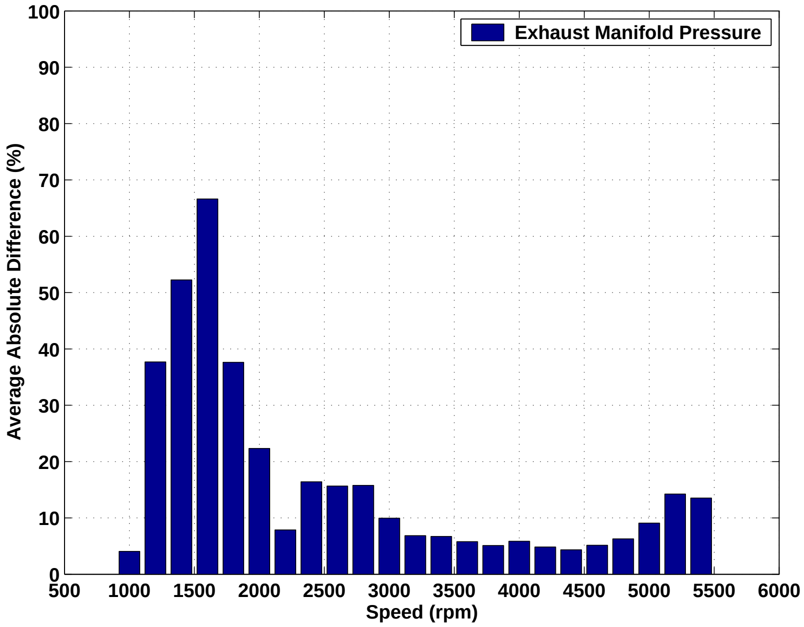


Figure B.6: Percentage difference between average exhaust back pressure of simulated and test data over the entire speed range.

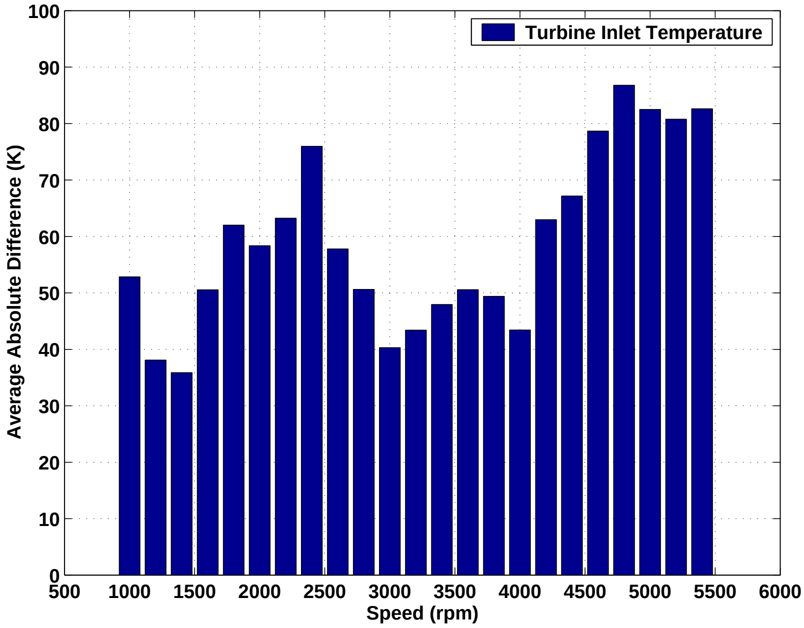


Figure B.7: Difference between average turbine inlet temperature of simulated and test data over the entire speed range.

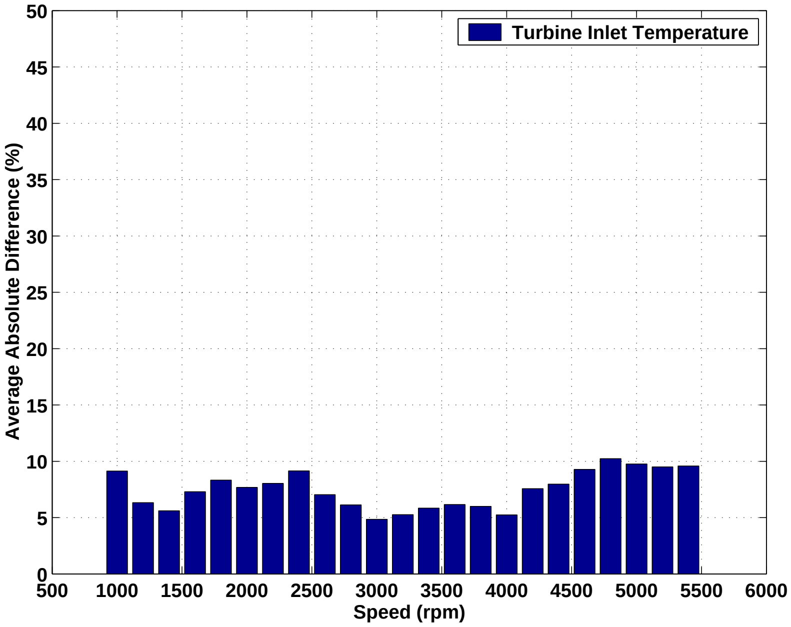


Figure B.8: Percentage difference between average turbine inlet temperature of simulated and test data over the entire speed range.

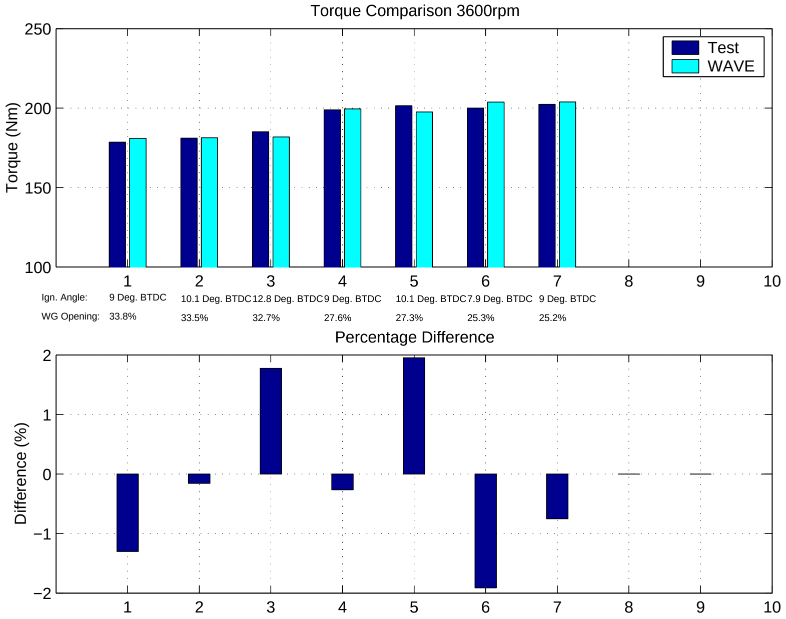


Figure B.9: The bar graphs used to compare the WAVE simulation torque results with measured data from the test engine at the 3600 rpm speed point.

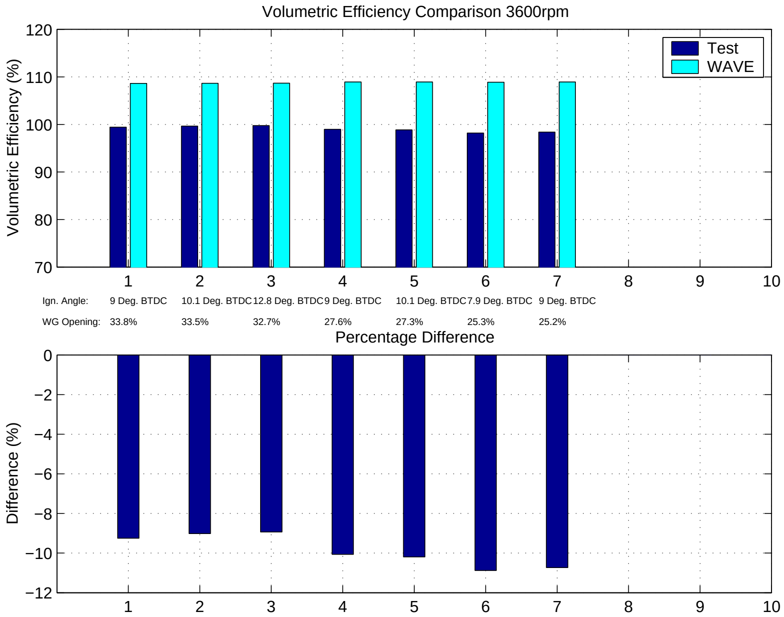


Figure B.10: The bar graphs used to compare the WAVE simulation volumetric efficiency results with measured data from the test engine at the 3600 rpm speed point.

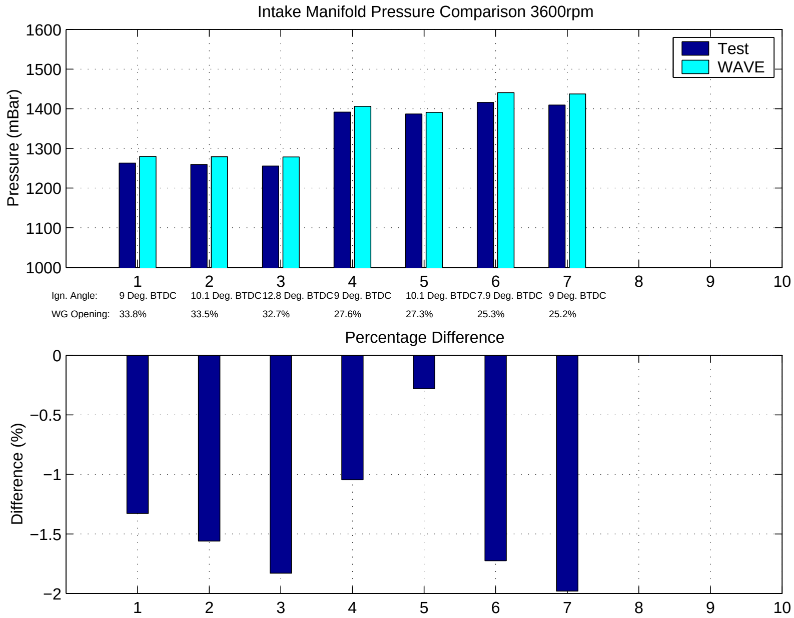


Figure B.11: The bar graphs used to compare the WAVE simulation intake manifold pressure results with measured data from the test engine at the 3600 rpm speed point.

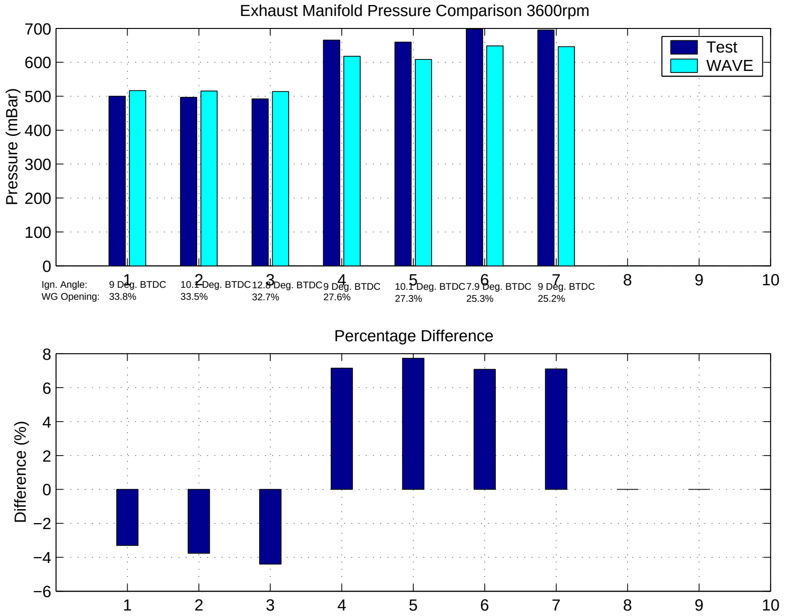


Figure B.12: The bar graphs used to compare the WAVE simulation exhaust manifold pressure results with measured data from the test engine at the 3600 rpm speed point.

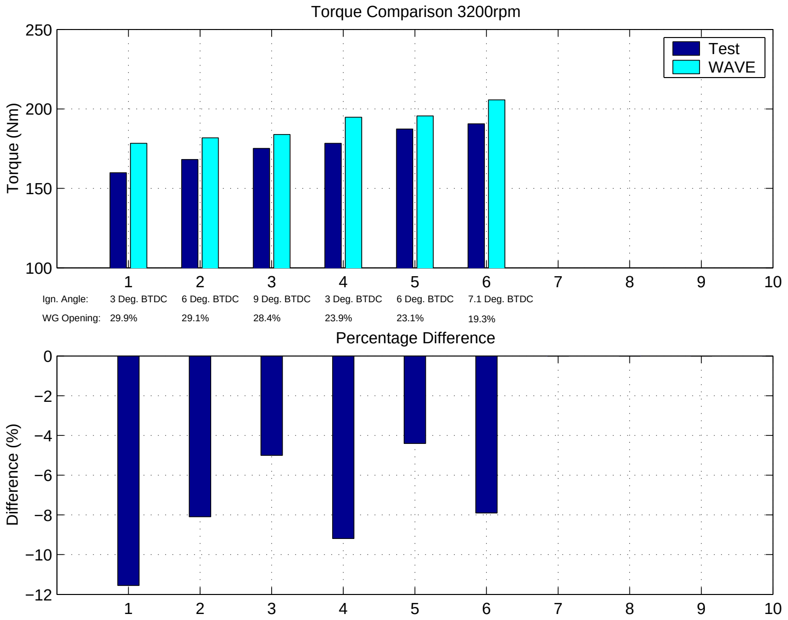


Figure B.13: The bar graphs used to compare the WAVE simulation torque results with measured data from the test engine at the 3200 rpm speed point.

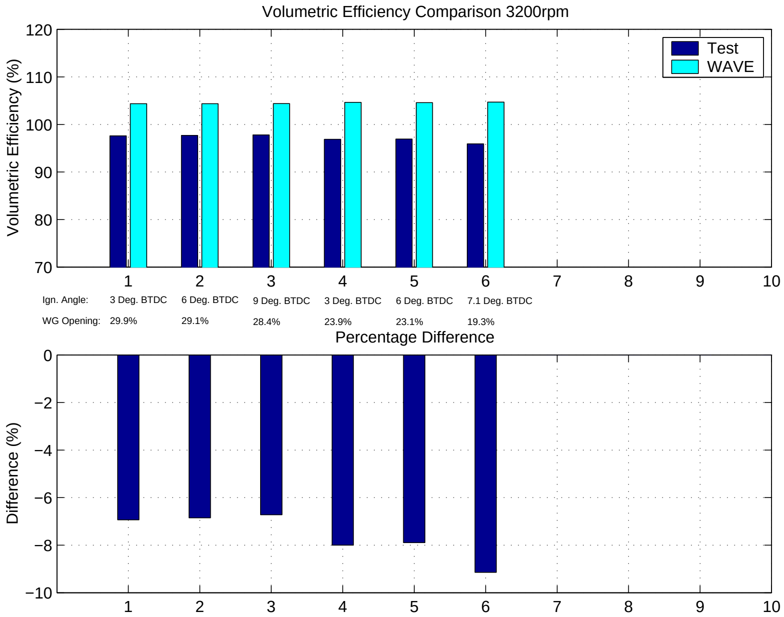


Figure B.14: The bar graphs used to compare the WAVE simulation volumetric efficiency results with measured data from the test engine at the 3200 rpm speed point.

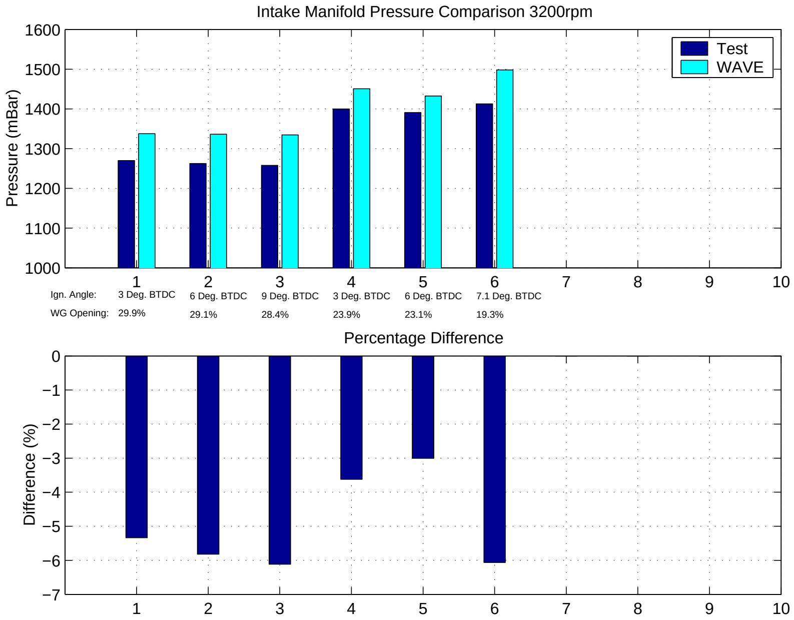


Figure B.15: The bar graphs used to compare the WAVE simulation intake manifold pressure results with measured data from the test engine at the 3200 rpm speed point.

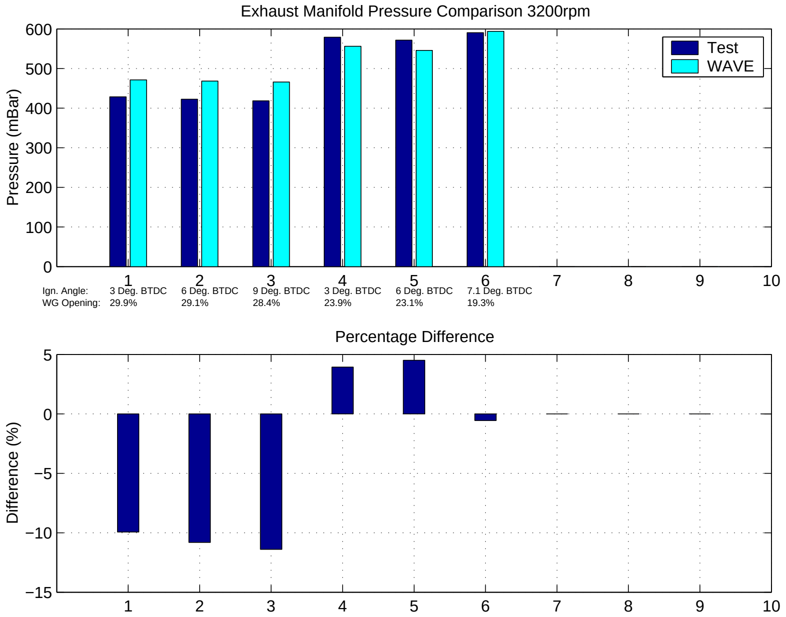


Figure B.16: The bar graphs used to compare the WAVE simulation exhaust manifold pressure results with measured data from the test engine at the 3200 rpm speed point.

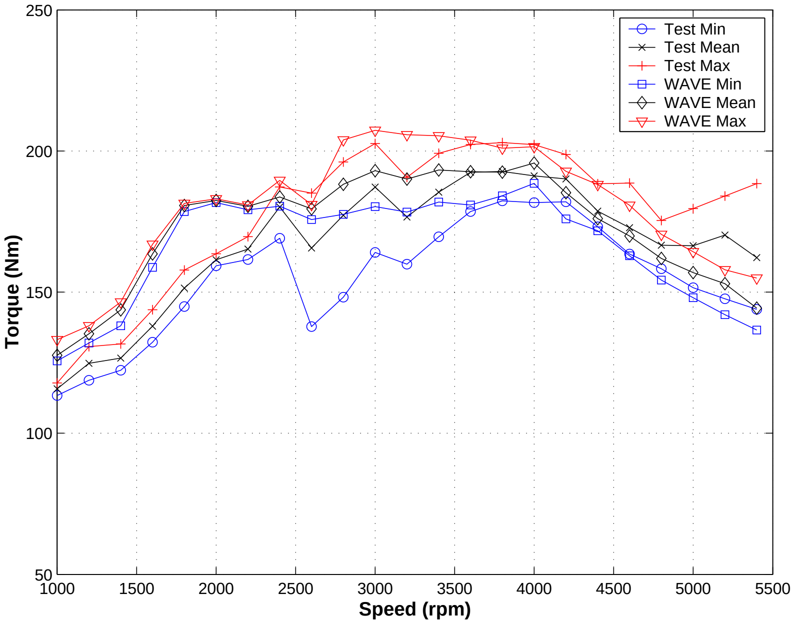


Figure B.17: Comparison of simulated and measured torque.

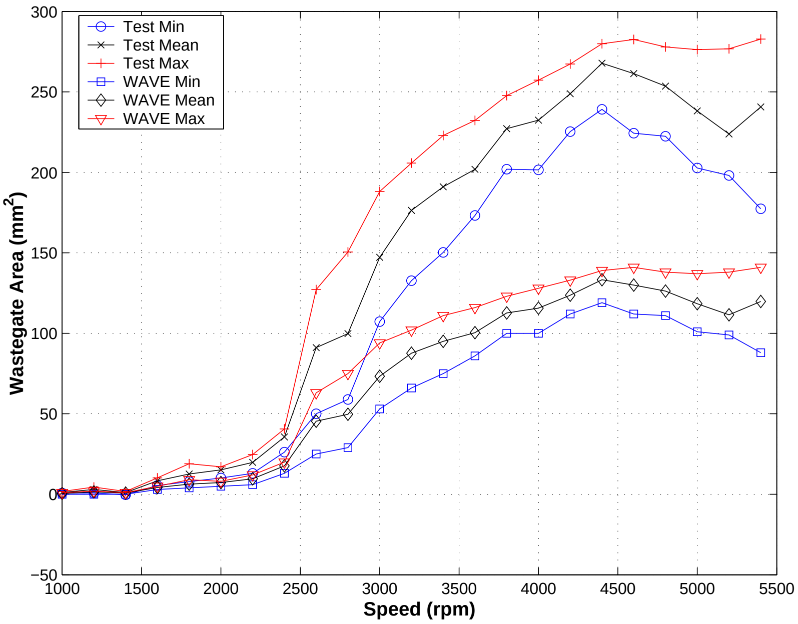


Figure B.18: Comparison of the wastegate area values calculated from measured data and the area values used when simulating the torque shown in Figure B.17.

Appendix C

Additional WAVE Model Information

C.1 List of data required for WAVE model

WAVE is a detailed multi-cylinder reciprocating engine simulation code. Its various sub-models require a number of input parameters related to combustion chamber geometry, valve flow, manifold configuration, etc. The data list below contains items that are either necessary or very helpful to successfully construct and validate a WAVE engine model.

Suggested units are provided where appropriate. Other units may be used but these should be indicated clearly when supplied. In order to validate the model with a high degree of precision, it is important to have as much engine test data as possible. Test data can be provided as ASCII text files (preferred) or as printouts from data acquisition systems.

1. Power Cylinder
    - Bore $[mm]$
    - Stroke $[mm]$
    - Connecting rod length, centre to centre $[mm]$
    - Piston pin offset (positive toward major thrust side) $[mm]$
    - TDC combustion chamber volume $[m^3]$
    - Compression ratio
    - Number of cylinders
    - Firing order
    - Firing interval °CA

  - Two or four stroke
  - Two-strokes: scavenging curve
  - Heat transfer area of combustion chamber: piston and head surfaces [expressed as multiple of bore area]
  - Clearance height between top of piston and top of cylinder [mm]

2. Intake and Exhaust Geometry

   - Intake piping and manifold geometry
   - Exhaust piping and manifold geometry
   - EGR circuit geometry

3. Valve/Port

   Data Duplicate this information for each intake and exhaust valve:

   Poppet Valves

   - Profile of lift vs. crank (or cam) angle - ASCII text file preferred
   - Valve/cam timing events
   - Dynamic valve data (e.g. valve event phase shift vs. engine rpm)
   - Tappet type (hydraulic/fixed)
   - Valve lash (hot) $[mm]$
   - Rocker arm ratio (if cam lift is prescribed)
   - Inner seat diameter (D) $[mm]$
   - Maximum valve lift $[mm]$
   - Valve flow data: flow coefficient(Forward and reverse) vs. L/D - ASCII text file preferred

     Piston Ported
   - Number of ports of the same type
   - Port geometry and precise location (drawing)
   - Profile of geometrical area (as an alternative to lift profile)
   - Port flow data: flow coefficient

     EGR Valves
   - Max lift to stop $[mm]$
   - Cross-sectional area at maximum lift $[cm^2]$
   - Profile of pressure loss vs. Flow

- Profile of flow coefficient vs. lift

4. Turbo-charger

- Compressor map showing operating points (map)
- Compressor reference temperature $[K]$
- Compressor reference pressure $[bar]$
- Compressor gas C_p $[J/kgK]$
- Specify total/total or total/static or static/static pressure ratio
- Compressor inlet diameter $[cm]$
- Compressor exit diameter $[cm]$
- Turbine map showing operating points (map)
- Turbine reference temperature $[K]$
- Turbine gas C_p $[J/kgK]$
- Turbine reference pressure $[bar]$
- Turbine speed (can be linked to compressor or geared to crankshaft) $[rpm]$
- Turbine inlet diameter $[cm]$
- Turbine exit diameter $[cm]$
- Mechanical efficiency [%]
- Moment of inertia of all rotating parts $[kgm^2]$
- Wastegate (if equipped)
- Variable geometry turbine (if equipped): provide maps at different settings.

C.2 Model Calibration

C.2.1 Volumetric Efficiency Calibration: Low-to-Medium RPM Calibration

If the volumetric efficiency comparison between WAVE results and measure data is not satisfactory over low-to-medium engine speeds the following WAVE settings should be investigated:

- Wall temperatures of the induction system, intake manifold and intake port. Typical values that should be used are provided in the manual.
- Heat transfer multipliers of the intake manifold and intake port. This must not be set to 0, typical values are provided in the manual.

The wall temperatures and heat transfer multipliers affect the volumetric efficiency by heating the intake charge, causing it to expand. It must therefore be accurately set according to the guidelines in the WAVE manual in order to represent the conditions in the test engine.

- Confirm that the intake valve timing and intake valve lash, or tappet clearance, are correct. The valve lash amount, if over-estimated, can have a significant affect on engine performance at very low rpm.

If going through these points do not result in a satisfactory match between the WAVE results and the measured data the modelling techniques used to model the intake-side components must be reconsidered.

C.2.2 Volumetric Efficiency Calibration: Medium-to-High RPM Calibration

If the volumetric efficiency comparison between WAVE results and measure data is not satisfactory over medium-to-high engine speeds the following WAVE settings should be investigated:

- Compare the Intake Manifold Pressure from WAVE against the measured data and consider the following two cases :

  1. If predicted intake manifold pressure is lower or higher than expected, then you might need to readjust the parameters or even rethink the modelling techniques used to model the components on the intake side upstream of the manifold (e.g. mass flow meter, throttle body, air cleaner and flow restrictions) as these components contribute pressure losses which may become significant at high flow rates.

2. If the intake manifold pressure is close to measured data but volumetric efficiency is lower or higher than expected, then pressure losses of the intake runners and/or runner entrance losses should be investigated.

- The discharge coefficient of the plenum Y-junction to intake runner connection can be set high $(0.9 - 1)$ to promote smooth flow from the plenum to the runners, or set lower (0.8) to increase pressure losses.

- The bend angle of the intake runners creates a pressure loss at high speed. If the losses are relatively small, the engine volumetric efficiency will be higher than expected. As such, ensure that the curvature aspects are entered correctly. If the Bend Angle field is not providing a sufficiently high/low C_p consider setting the Bend Angle to 0 and manually entering a Pressure Loss Coefficient.

- For engines with high tumble/swirl ratio, consider adjusting the Heat Transfer Multiplier When Intake Valves are Open in a procedure similar to the one described in the section for Low-to-Medium RPM Calibration volumetric efficiency problems. This affects the engine volumetric efficiency by heating the intake charge, causing it to expand. Increasing this multiplier will decrease the volumetric efficiency and vice-versa.

C.2.3 Torque Calibration

1. Torque Correction

    - Correct the measured torque results using Society of Automotive Engineers (SAE) standards for the range of engine operation.

2. Identification of causes of mismatch between measured and predicted Torque

    - One of the practical methods of identifying causes of mismatch between measured and predicted torque values (i.e. if cause is being poor breathing, wrong fuelling, etc.) is the calculation and comparison of the measured [volumetric efficiency / corr. Torque] ratio vs. engine speed against the same ratio calculated from predicted volumetric efficiency and torque in WAVE.

3. Fuel Issues

    - Torque depends on, among other factors, the fuel amount injected (i.e. AFR) and the heating value of the fuel (LHV). As such, one has to make sure that the fuel property input file being used has the correct LHV and the AFR ratio matches the specifications of the actual experiment where and when torque was measured.

4. Combustion Model
    - Typically, a correlational combustion model is used, applying a heat-release vs. time profile to the simulation. Make sure that the applied profile changes appropriately as a function of engine speed. Typically, in the SI Wiebe function the $10\% - 90\%$ Combustion Duration increases steadily as engine speed increases and the location of the 50% burn point is affected by spark timing advance, which increases with rpm but remains nearly constant as the engine speed changes.
5. Engine Knock
    - Compare maximum in-cylinder pressure (PMAX) with measured data if available. As a simple check for engine knock, make sure that PMAX values are not exceeding the typical value of 75 bar for naturally aspirated gasoline engines or 85 bar if the engine is turbocharged. Alternatively, there is always the option of activating the knock model to detect if the knock phenomenon occurs.
6. Friction Losses
    - If the engine model accurately predicts volumetric efficiency, indicated mean effective pressure IMEP and PMAX while the BMEP and brake torque are under- or over-predicted, it should be clear that the friction model needs to be adjusted (i.e. to lower or increase FMEP as needed).
7. Exhaust Pressure and Temperature
    - Compare exhaust gas temperatures against measured data. If these temperatures are not in good match, investigate the cylinder Woschni Heat Transfer Multipliers When Intake Valves are Closed, explained earlier and the related Heat Transfer Surface Temperatures (Cylinder, Head and Liner). If there is no performance improvement reconsider the exhaust component models and their individual and combined effect on gas temperatures. Notes regarding the use of the multiplier:
        - Increasing it removes energy from the in-cylinder gas, resulting in reduced IMEP, reduced exhaust temperature and increased heat loss to the cylinder walls

  - Use similar range limits as CENHTO: 1 for quiescent, up to 2 for high turbulence (swirl/tumble) conditions.

- Exhaust gas temperatures are especially important for turbocharged engines as such temperatures figure in the turbine model as the turbine inlet temperature and may affect engine performance considerably.

- Optionally, compare predicted exhaust pressure traces (using WAVE time plots) at a specific location where measured data are available. Incorrectly high exhaust pressure can increase pumping work losses, leading to reduced IMEP prediction.