

Mean Value Modelling of HIPPO-2 Diesel Engine

Georgios Sideris

Diploma Thesis



School of Naval Architecture and Marine Engineering
National Technical University of Athens

Supervisor: Assistant Prof. George Papalamprou

Committee Member : Prof. N. Kyrtatos

Committee Member : Associate Prof. C. Papadopoulos

September 2019

Chapter 1

Abstract

A mean value model of the Caterpillar C9.3 ACERT Engine is developed parameterized and validated. The engine is a 9.3-liter diesel engine that has Exhaust Gas Recirculation (EGR) and a Wastegate. EGR is implemented using the built in Venturi flow meter system of the engine and for the calculation of the air to fuel ratio (Λ) the oxygen concentrations of both the intake and exhaust manifolds is considered (since EGR has oxygen). The turbocharger is of static geometry. The intended uses for the above model are system analysis, simulation and development of model-based controllers. The goal for this model is to accurately measure and compute the air flows both in and out of the manifolds (exhaust manifold, intake manifold as well as the engine itself), the turbocharger dynamics (airflows, efficiencies and turbocharger dynamics), the engine torque, the air-fuel ratio, the intake and exhaust temperatures etc. The total dynamic states of the complete model are two for the manifolds (pressure and oxygen concentration) and two for the turbocharger (speed and wastegate flow). The model needs to be fast and precise, needing short computational times while having great accuracy. A Simulink implementation of the model is included.

Initially, measured Data were acquired from the experimental facility of HIPPO2 in the Laboratory of Marine Engineering of the National Technical University of Athens using the engine's built in sensors and the facility setup. The Data were stored in the form of Time valued Data sets and were used for the initialization of the separate submodel parameters as well as for the complete engine model tuning. The data sets were both stationary and dynamic.

All the relative modelling principles and equations for each needed submodel are described and analyzed. Each submodel's implementation on Simulink is also shown and described.

The tuning of the complete model was done automatically through weighted least-squares optimization, which scopes in the minimization of a cost function that evaluates the errors in both dynamic and static submodels equally.

After the complete engine model tuning, the model is validated using measured Data, that were not used in its tuning beforehand and the error of the simulated data compared to the measured ones is calculated for each different Data set.

List of Figures

3.1	HIPPO2 Experimental Hybrid Setup	18
3.2	HIPPO2 Schematic Representation	19
3.3	Caterpillar C9.3	20
3.4	C9.3 Load Diagram	21
3.5	The ABB electric 3-phase motor	22
3.6	The ABB electric 3-phase brake	22
3.7	The HIPPO2 monitoring and control board	23
3.8	The general arrangement of Data Acquisition system of HIPPO2	24
4.1	Main system's input and output signals	28
4.2	Relevant Signals, a)variable of interest, b)very fast and c)very slow variables.	29
4.3	Cause and Effect Diagram of a Diesel Engine without EGR and WG	30
4.4	Receiver, inputs and outputs, [1]	31
4.5	Intake Manifold I/O Diagram	34
4.6	Exhaust Manifold I/O Diagram	34
4.7	Intake Manifold Simulink Model	34
4.8	Exhaust Manifold Simulink Model	34
4.9	Typical Aftertreatment System Layout of Diesel Engine	36
4.10	I/O Diagram of Volumetric Efficiency Model	37
4.11	Simulink implemented model for volumetric efficiency	37
4.12	I/O Diagram of Lambda Model	38
4.13	Simulink implemented model for Lambda	38
4.14	Venturi Orifice, similar to the one installed in the Engine	40
4.15	I/O Diagram of EGR flow	40
4.16	Simulink implemented model for EGR flow	41
4.17	Wastegated Turbocharger	43
4.18	I/O Diagram of Turbocharger Speed	44
4.19	Simulink implemented model for Turbocharger Speed	44
4.20	Garrett GT3782 Compressor Performance Map, [2]	45
4.21	Speed curves given in dimensionless parameters Ψ and Φ coordinates	47
4.22	Fitting Φ using equation 4.4.6.	48
4.23	Fitting a to the Mach number using a third order polynomial equation.	49
4.24	Fitting b to the Mach number using a third order polynomial equation.	49
4.25	Fitting c to the Mach number using a third order polynomial equation.	49
4.26	3D GT3782 Fitted Compressor Performance Map.	50
4.27	2D GT3782 Fitted Compressor Performance Map.	50
4.28	I/O Diagram of Compressor Mass Flow	51

4.29	Simulink implemented model for Compressor Mass Flow	51
4.30	Fourth order Polynomial Fitting for the Compressor Isentropic Efficiency	53
4.31	2D Isentropic Efficiency Model	53
4.32	I/O Diagram of Compressor Isentropic Efficiency	54
4.33	Simulink implemented model for Compressor Efficiency	54
4.34	Simulink implemented model for the Compressor Model	54
4.35	Garrett® GT3782™ Turbine Performance Map, [2]	55
4.36	Turbine Flow Map (Fitted)	56
4.37	I/O Diagram of Turbine Mass Flow	56
4.38	Simulink implemented model for Turbine Mass Flow	57
4.39	Simplified Turbine Efficiency map	57
4.40	I/O Diagram of Turbine Efficiency	58
4.41	Simulink implemented model for Turbine Efficiency	58
4.42	4-stroke Diesel thermodynamic cycle	59
4.43	I/O Diagram of Engine Torque	61
4.44	Simulink implemented model for Engine Torque	61
4.45	Seiliger cycle	62
4.46	I/O Diagram of Exhaust Gas Temperature	63
4.47	I/O Diagram of Exhaust Manifold Temperaure	65
4.48	Simulink implemented model for Exhaust Manifold Temperaure	65
5.1	Input Data from 3 Data Sets for the Initialization of the vol. ef. Model parameters	69
5.2	Volumetric Efficiency Fitting procedure	69
5.3	Volumetric Efficiency Measurements vs. Model	70
5.4	Relative Error between Model and Measurements	71
5.5	Relative Error between Model and Measurements (%)	71
5.6	Fitting Procedure for the Torque Model	74
5.7	Torque Model vs. Measured Data	75
5.8	Relative Error on the Torque Generation Model	76
5.9	Relative Error in Percentage on the Torque Generation Model	77
5.10	Pressure Ratio from Data	79
5.11	Fuel Rate from Data	79
5.12	Exhaust Mass Flow from Data	80
5.13	Intake Manifold Temperature from Data	80
5.14	Exhaust Gas Temperature for Different Iteration Numbers	81
5.15	Fitting Process for the second Model	82
5.16	Exhaust Temperature for the two different Models	83
5.17	Relative error between first and second model.	84
5.18	Relative error in % between first and second model.	85
5.19	Intake Manifold Pressure for initial guess	87
5.20	Exhaust Manifold Pressure for initial guess	88
5.21	Intake Manifold Pressure after the unknown parameter initialization	89
5.22	Exhaust Manifold Pressure after the unknown parameter initialization	89
5.23	Data Set A, Model vs. Measurements	94
5.24	Data Set B, Model vs. Measurements	95

5.25	Data Set C, Model vs. Measurements	96
5.26	Data Set D, Model vs. Measurements	97
5.27	Data Set E, Model vs. Measurements	98
5.28	Data Set F: Inputs to the Model	100
5.29	Data Set F, Model vs. Measurements	101
5.30	Data Set F, Flows	102
5.31	Data Set F, Turbocharger Modelled Outputs	103
5.32	Data Set F, Exhaust Temperatures	104
5.33	Data Set G: Inputs to the Model	105
5.34	Data Set G, Model vs. Measurements	106
5.35	Data Set G, Flows	107
5.36	Data Set G, Turbocharger Modelled Outputs	108
5.37	Data Set G, Exhaust Temperatures	109
5.38	Data Set H: Inputs to the Model	110
5.39	Data Set H, Model vs. Measurements	111
5.40	Data Set H, Flows	112
5.41	Data Set H, Turbocharger Modelled Outputs	113
5.42	Data Set H, Exhaust Temperatures	114
43	I/O arguments for the $W_{wastegate}$ estimation model	119
44	Simulink implementation for the $W_{wastegate}$ estimation model	119
45	Simulink implementation of the complete model	121

List of Tables

3.1	Caterpillar C9.3 Specifications.	20
3.2	Relevant Available Sensor Measurements from the ICE	25
4.1	Results of fitting for a,b,c.	48
5.1	Results of fitting for vol_1, vol_2, vol_3	70
5.2	Torque Model Resulting Parameters	75
5.3	Mean Values of Error on the Torque Generation Model	77
5.4	Exhaust Gas Temperature Second Model Parameters	83
5.5	Mean Values of Error on the Exhaust Gas Temperature Model	84
5.6	Manual Initial Parameter Setting	87
5.7	Automatic parameter tuning, after first cost function	90
5.8	Automatic parameter tuning, using the second cost function	92
5.9	The mean absolute relative error % between the dynamic tuning data and diesel engine model simulation where the model parameters are tuned according to the method in this section.	93
5.10	The mean absolute relative error % between the dynamic validation data and the diesel engine model simulation	115

Contents

1	Abstract	3
	List of Figures	5
	List of Tables	9
1.1	Acknowledgements	13
2	Introduction	15
2.0.1	Literature Review	16
3	Propulsion Plant and Data Acquisition	17
3.1	HIPPO-2 Experimental Test-bed	18
3.2	The Internal Combustion Engine: CATERPILLAR® C9.3	19
3.3	Electric Brake and Electric Motor	22
3.4	Data Acquirement	23
4	Modelling-Basic Principles	27
4.1	Cause and Effect Diagram	29
4.2	Air System	31
4.2.1	Manifolds	31
4.2.1.1	Manifold Pressures	31
4.2.1.2	Oxygen Fraction in the Receivers	33
4.2.2	Engine Air Mass Flow-Volumetric Efficiency Estimation	35
4.2.3	Oxygen to Fuel Ratio inside Cylinder (Lambda Calculation)	38
4.3	Exhaust Gas Recirculation-EGR Mass Flow approximation	39
4.4	Forced Induction-Turbocharger	42
4.4.1	Turbocharger Speed	43
4.4.2	Compressor Modelling	45
4.4.2.1	Modelling the Compressor Flow Rate	46
4.4.2.2	Modelling the Compressor Efficiency	52
4.4.3	Turbine Modelling	55
4.4.3.1	Modelling the Turbine Flow Rate	56
4.4.3.2	Modelling Turbine Efficiency	57
4.5	Mechanical System-Torque Generation	59
4.6	Cylinder-out Temperature	62
4.7	Heat Losses in the exhaust manifold pipes	64

5	Tuning and Validation	67
5.1	Initialization of the independent submodel parameters	68
5.1.1	Volumetric Efficiency Model	68
5.1.2	Torque Generation Model	73
5.1.3	Exhaust Manifold Temperature Model	78
5.1.3.1	Error Evaluation for the Exhaust Gas Temperature Model	84
5.2	Initialization of the rest of the parameters	86
5.3	Tuning the static parameters using the complete model	91
5.4	Tuning Completion	93
5.4.1	Result Diagrams	94
5.5	The importance of the initialization and the cost function	99
5.6	Validation	99
5.7	Conclusions	115
	Appendices	117
.1	APPENDIX A	119
.2	APPENDIX B	121
	Bibliography	123

1.1 Acknowledgements

This Thesis has been carried out at the Laboratory of Marine Engineering (LME for short) at the School of Naval Architecture and Marine Engineering of the National Technical University of Athens, under the supervision of Professor Georgios Papalamprou.

I would like to thank Professor Papalamprou for the opportunity he so gladly gave me to work on this project. In addition, I would also like to thank him for the helpful information, guidance and insight that I needed during the times I faced difficult problems that needed to be overcome.

I am also deeply grateful to the PhD student Nikolaos Planakis for handling the whole experimental setup of HIPPO2 and therefore providing me with crucial and essential real time and simulation data from the engine. Without the above this thesis would not have been developed.

Chapter 2

Introduction

The emission control legislations for internal combustion engines are becoming continuously stricter. Engine designers focus on creating cleaner and more efficient engines with minimal exhaust pollutants and lower fuel consumptions. New technologies that weren't that popular in the past, such as Exhaust Gas Recirculation, After-treatment units, Selective Catalytic Reduction are starting to become the standard in diesel engines, this growing complicity in the design of the modern Diesel engines has created the need for more complex control systems to be developed.

This growing complexity of engine control systems results in the extensive use of model-based control system designing and optimization. Different model complexities are created for different applications. The mean value modeling approach is a widely used method by many researchers [3], [4] because of, its capability of not only observing engine operating states such as pressure, temperature, mass flow, and torque but also capturing engine transient response. Moreover, its advantage of quick simulation speed that cannot be compared with other model approaches (i.e. cylinder by cylinder method) facilitates its application in control systems design.

In mean value models (MVEM), engines are assumed to work continuously and to ease simulations. MVEM's neglect the discrete cycle of the engine and assume that all processes and effects are spread out over the engine cycle. These processes such as the combustion in the cylinder or the air flow through a pipe, are simplified through the use of first principle models and empirical relationships. Consequently, the heavy use of physical relationships that rely on physical values, led to the great dependence of MVEM's accuracy on quantity of experimental data.

In this thesis, a MVEM of the turbocharged Caterpillar C9.3 ACERT Diesel engine is built, tuned and validated using limited Data out from the engine's sensors. The thesis will have the following structure. Firstly, the lab setup, experimental procedure and experimental data will be shown and analyzed so that the reader can have a clear view of how the basis of this model was created. After all the modelling of the engine would not be possible without this data. Secondly, a theoretical background and a mathematical approach will be shown for each submodel that in whole forms the final and complete mean value model of the engine. Thirdly, the approach and the procedure that was followed to actually solve the problem in Matlab will be shown, from parameter initialization to parameter tuning and problem overcoming. Finally, the resulted model will be compared with the real time data that was acquired in the lab in order to be validated in its accuracy.

The proposed model is mean value engine model that uses filling and emptying concepts for the manifolds. Numerous models have been published on modelling Diesel engines, with EGR or without, each with different selection of states and complexity. The proposed model here, has two states, for the control volumes (intake and exhaust manifold) namely, the pressures and the oxygen fractions, since the use of EGR is considered. Moreover, having also implemented the turbocharger system, another two states are added to the complete model, the turbocharger rotational speed, entirely calculated from the model operating scenarios and the wastegate controller which simulates the built in wastegate actuator of the engine. Due to the limited amount of Data from the engine's sensors, some of the submodels rely on empirical data (EGR).

2.0.1 Literature Review

Several mean value models have been developed over the years. For example, Winterbone (1977) developed a wholly dynamic model which represents the engine and the turbocharger (TC) gas flows by a set of 30 interconnected first order nonlinear differential equations. It employs the filling and emptying technique and being based mostly on physical principles, it gives an accurate description of the engine's properties. An important aspect of the mean value method is the use of physically based models. The fact that a given system is complex does not necessarily imply that there are no underlying physical principles which can give a simple overall picture of the engine operation [5]. Another MVEM with VGT and EGR was developed by [3] aiming to construct a model that describes the gas flow dynamics including the dynamics in the manifold pressures, turbocharger, EGR, and actuators with few states in order to obtain short simulation times. This approach to MVEM creation is different compared to similar studies mainly due to the fact that many Data that are needed for the tuning and the initialization of the model are not measured. This problem of limited data is faced by logical assumptions and physical measurements that can somewhat approximate the missing Data and allow for an accurate result.

Chapter 3

Propulsion Plant and Data Acquisition

In this chapter, the experimental hybrid powertrain facility *HIPPO2* where the Data that were required for the creation of the Engine model, the tuning and validation were acquired is presented. The facility is composed of three main components, *the Internal Combustion Diesel Engine, the Electric Brake (EB) and the electric motor/generator (EM)*. Each one will be thoroughly presented in its respective subchapter below.

The Data that were needed for the purpose of modelling the CATERPILLAR® model C9.3 ACERT™ diesel engine (shown in Fig. 3.3) were all acquired from the *HIPPO-2* Lab setup (shown in Fig.3.1,3.2). The Data were collected from engine test runs in the lab facility with specific control inputs. The Test-runs differ from one another in means that all different Test-runs have different input variables (i.e. engine speed, fuel rate, engine load etc.). More on the Data collection process is gonna be represented on the following section.

3.1 HIPPO-2 Experimental Test-bed

The *HIPPO-2 hybrid diesel-electric test setup* consists of an Internal Combustion Engine (ICE) in serial connection to an electric motor (EM) through an Electric Brake (EB). In this setup the rotational speed of the ICE is the same with the EM and the supplied torques from each one respectively add together to meet the total torque demand that is applied by the EB. In Figure 3.1 an actual Picture of the Lab is shown and in Figure 3.2 a schematic representation is included.

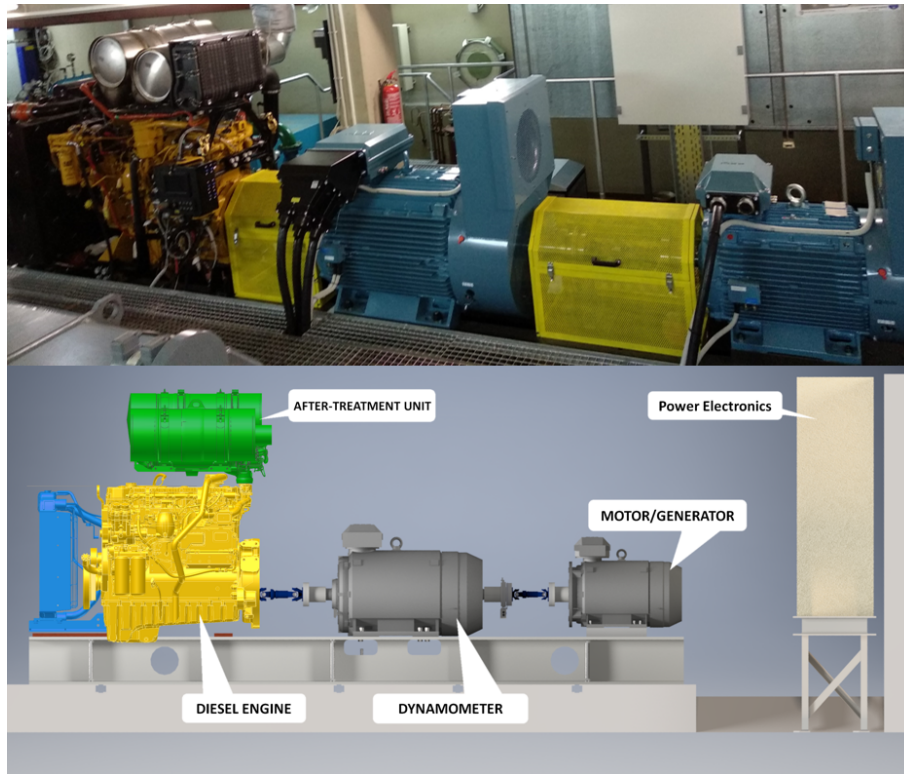


Figure 3.1: HIPPO2 Experimental Hybrid Setup

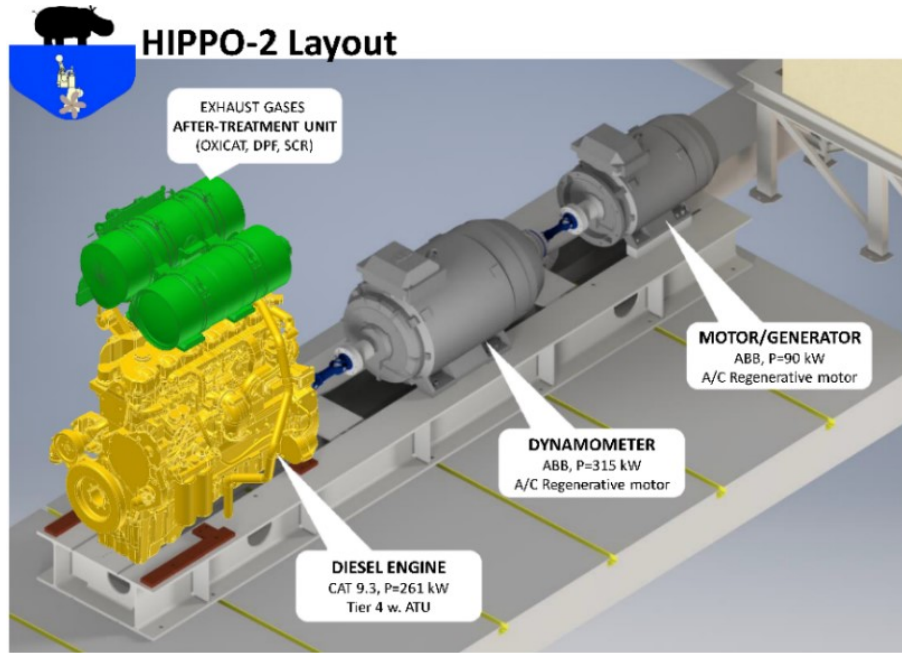


Figure 3.2: HIPPO2 Schematic Representation

3.2 The Internal Combustion Engine: CATERPILLAR® C9.3

The ICE of the Hybrid Setup is the turbocharged and aftercooled CATERPILLAR® 6-cylinder 9.3-liter 4-stroke industrial diesel engine, model C9.3 ACERT™, (shown in Fig. 3.3), producing 261 kW at 1800-2200 rpm and maximum torque 1597 Nm at 1400 rpm, as shown in its load diagram, Fig. 3.4 (Rating C). The engine speed is controlled from the electronic control unit (ECU) based on the speed reference set and the deviation of speed measurement. The ECU of the internal combustion engine controls the fuel-rate in the cylinders in closed loop control, using the lookup tables that were set in the factory.

The engine is equipped with the latest technology on emission control and is therefore designed to meet U.S. EPA Tier 4 Final and EU Tier VI emission standards. It is equipped with an Exhaust Gas Recirculation system (EGR) that is also controlled by the ECU. The engine is also equipped with an integrated after treatment unit which includes a Diesel Particulate Filter (DPF) and a Selective Catalyst Reducer (SCR). Therefore, harmful emissions like NO_x, soot and particulates are limited.



Figure 3.3: Caterpillar C9.3

The exact specifications of the Engine are all presented below in Table 3.1, found in [6]:

Table 3.1: Caterpillar C9.3 Specifications.

Parameter	Magnitude	Units
Power	261 @ 2200rpm	bkW
Torque	1597 @ 1400rpm	Nm
Cylinder N.	6	-
Bore	115	mm
Stroke	149	mm
Displacement	9.3	Liters
Compression Ratio	17.0:1	-
Cooling System Cap.	54	Liters
Lube System Cap.	30	Liters
Dry Weight	885	Kg
Length	1150	mm
Width	827	mm
Height	1123	mm

The Engine load diagram is presented below in Fig.3.4, following the "Rating C" curve:

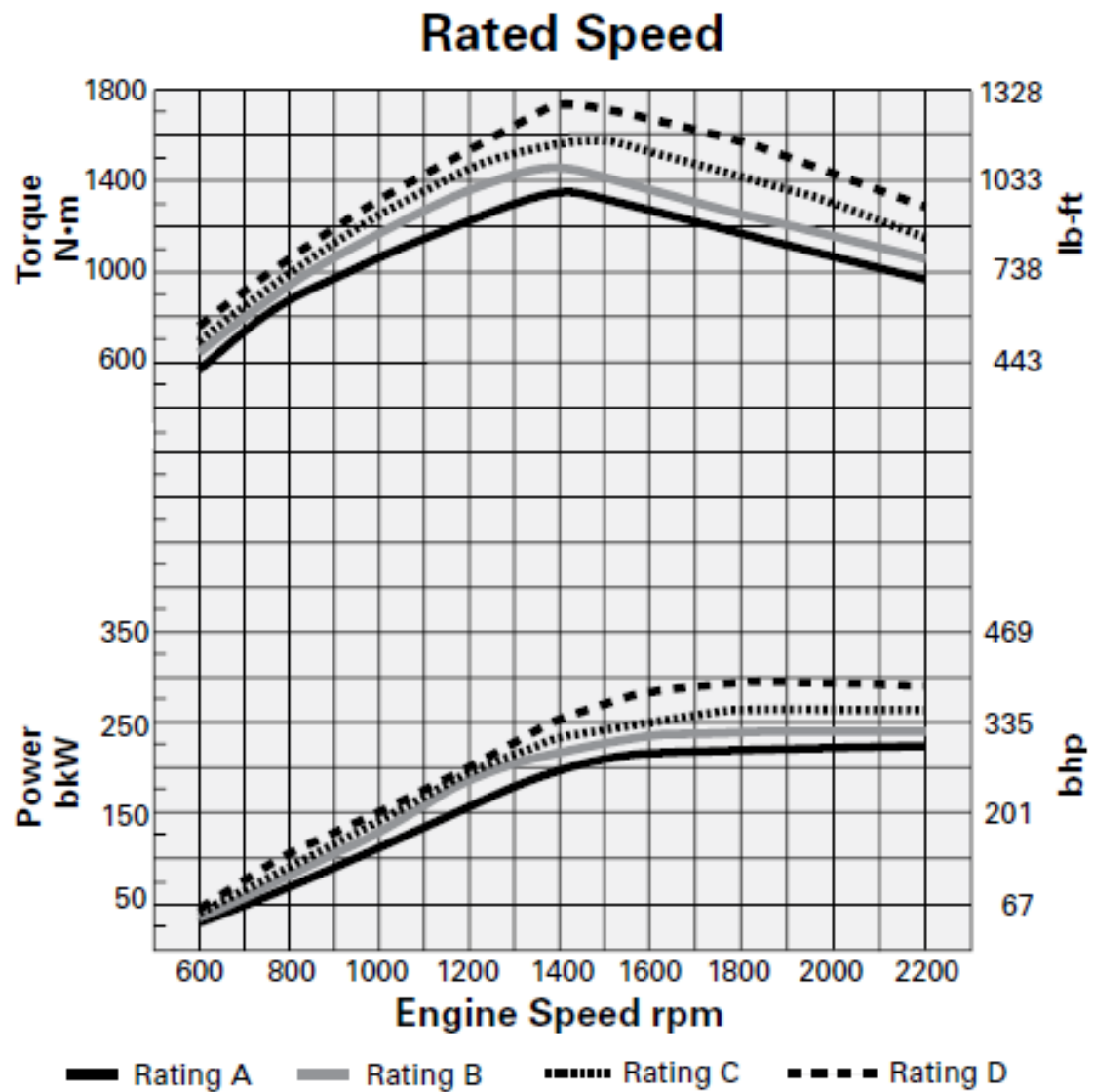


Figure 3.4: C9.3 Load Diagram

3.3 Electric Brake and Electric Motor

The EM is a standard AC induction 3-phase motor (seen in Fig.??), it's rated power is 90kW at 1483rpm, it's model type is M3BP 280SMB 4 IMB3/IM1001 and it's manufactured by ABB[®]. Obviously, the EM can double as an Electric Generator if Torque is supplied to it.

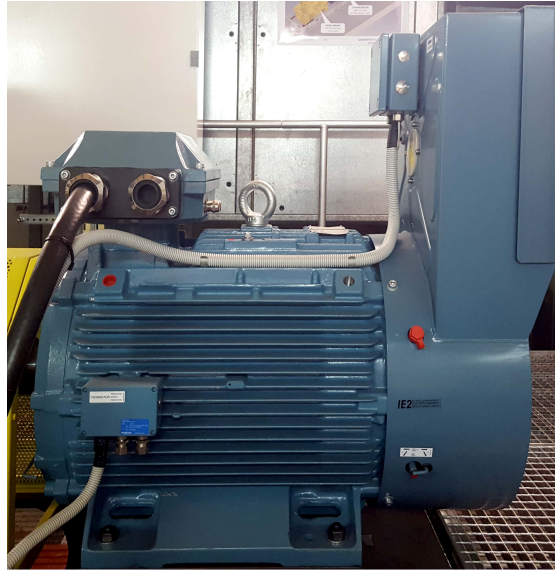


Figure 3.5: The ABB electric 3-phase motor

The EB is a standard AC induction e-phase motor, which is also manufactured by ABB[®]. It's model type is M3BP 355SMB 4 IMB3/IM1001 and it can withstand loads of up to 315kW operating at 1488rpm.

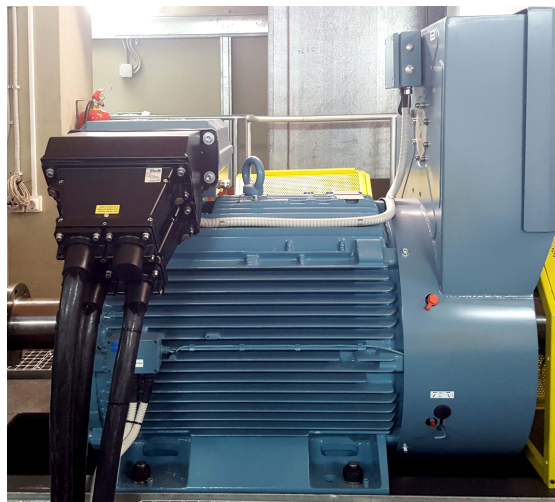


Figure 3.6: The ABB electric 3-phase brake

The 3 motors are connected in series through axels, thus the operating speed of HIPPO-2 is from 600 to 2200 rpm, with maximum load of 341kW (combined power of ICE and EM).

3.4 Data Acquisition

The Lab's data acquisition system consists of the following parts:

1. DSpace Microautobox II
2. CAN bus
3. RCAN modules
4. ABB Drivers
5. Signaling Cables for I/O
6. Ethernet Cable

The platform for Data Acquisition and control of the powertrain is based on the dSpace Micro Autobox II DS1401/1511 (shown in Fig.??) controller board, with rapid control prototyping capability, programmed under the MATLAB/ Simulink environment.



Figure 3.7: The HIPPO2 monitoring and control board

The simplicity of communication topography is depicted in Fig.3.8. dSpace Micro Autobox II is a real-time system for performing fast function prototyping in full-pass and bypass scenarios. It operates without user intervention, just like an ECU. It has the ability to connect with many sensors from the testbed through of its Analog, Digital and CAN ports. In fact, with dSMA II it is possible to read all the data from any sensor which is mounted on C9.3 diesel as well on ABB electric motor through CAN bus. The backbone of CAN bus is a twisted pair cable; in the particular case 3 CAN channels from dSpace are used. The first is CAN A from CAT engine's ECU and is connected to CAN 2 in dSMA II connector, the second is ABB CAN from RCAN modules which is connected to CAN 3 in dSMA II; the third CAN channel, CAN 4, has been installed for future use and is connected to AT's ECU. The CAT C9.3 engine together with the two ABB motors have almost 150 sensors and 700 signals that can be acquired and used for monitoring every parameter of the HIPPO-2 test bench. It is very important for the prototype test bench to have the capability to monitor a diverse set of parameters for basic engine, TC, AT as well as for those parameters related to the operation of electric drives. Taking into consideration all the above complexity of sensing, the most efficient way of connecting and controlling the motors is through CAN bus.

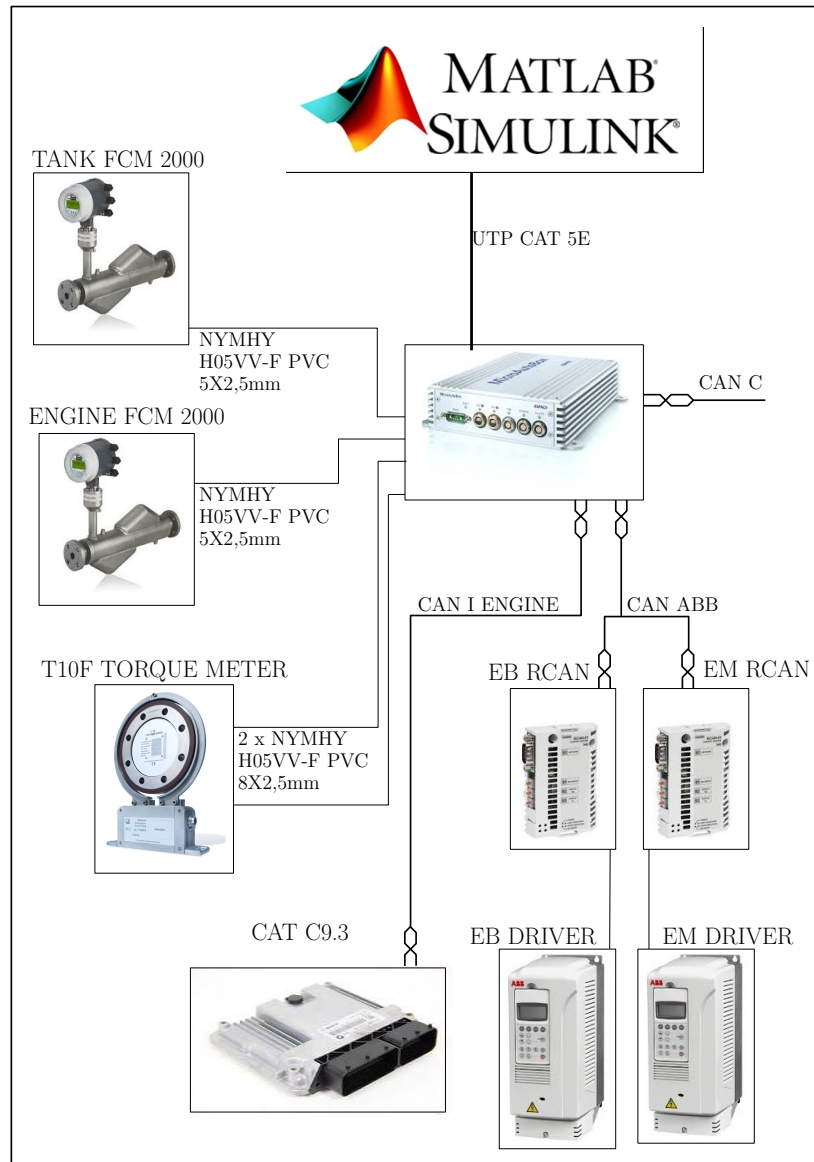


Figure 3.8: The general arrangement of Data Acquisition system of HIPPO2

The data were gathered from semi-randomized runs of the engine. No specific static or dynamic cycles were followed (i.e. European transient cycle). On the other hand, the Data were gathered in multiple different engine Runs on the Lab. The term “semi” random is used as in that each run was always with different input parameters to the engine than the others, for example no runs were on the same engine speeds (rpm) or on the same load from the EB. The parameters that can be and were indeed changed each time are countless and therefore having realistic runs with different parameters each time (the parameters that were set each time are close to the usual ones when the engine is actually on operation e.g. the rpm were set close to the realistic values that the engine will actually work on) is enough to fully model the engine and fit the unknown parameters in a way that describes the whole operation spectrum of the Engine.

Below there is a table of all the measured data that are of importance in the problem of modelling the diesel engine of the setup. Note that only these data were

able to be captured from the engine's built in sensors. The data that were missing, were mainly estimated through least square optimization algorithms and from model computations.

Table 3.2: Relevant Available Sensor Measurements from the ICE

Measurement	Unit
Load of the Engine	Percentage
ATU Exhaust Gas Mass Flow	Kg/h
Electric Brake Measured Torque	Nm
Engine air inlet Pressure	kPa
Engine intake manifold Temperature	
Engine Speed	Rpm
Fuel Rate	Kg/h
Lambda engine	-

Chapter 4

Modelling-Basic Principles

Modelling a turbocharged Diesel Engine is not an easy task. The engine is a complex, highly nonlinear dynamic system with at least one dynamic feedback loop, namely the turbocharger [7]. The turbine part of the turbocharger extracts the enthalpy of the exhaust gas and is used to speed up the turbocharger shaft. The rotational kinetic energy is then used by the compressor to increase the intake air mass flow against a pressure rise. Having an intercooler the temperature rise inside the compressor is mitigated to a point where the gas density increases leading to higher mass flows inside the combustion chamber. In turn, the increased mass flow provides additional enthalpy to the turbine so the process is repeated.

Having an engine that is equipped with a Waste Gate further complicates the modelling process as it introduces another feedback loop. This loop is connected with the turbocharger loop. The exhaust gas that is released bypasses the turbine of the turbocharger reducing the available enthalpy to be turned to kinetic energy.

Since diesel engines are so widely used in industry and torque requiring setups, several modelling approaches have been conducted [1, 3, 4, 8–10]. Most complete models that are proposed in the literature either operate without the use of EGR or Wastegate and are either simplified or formed with EGR and VGT. This approach shows the operation of a non VGT turbocharged Diesel Engine using EGR (which is not separately calculated, due to the lack of sensors, but rather the egr flow is contained in the flows that go both in and outside the engine) and WasteGate. The waste gate approximate flow is estimated using an independent integrator submodel that aims to simulate both the WG flow and dynamics.

Reciprocating engines differ in at least two ways from continuously operating thermal engines (i.e. Gas Turbines) [1]:

- The combustion process is highly transient (Diesel cycle, presenting large and rapid temperature variations).
- The thermodynamic conditions that capture the combustion process (i.e. IM pressure, exhaust gas temperature) are not constant.

The phenomena in the first class are very fast and hard to describe without the use of highly complex models. Therefore, they are not useful in the design of fast running, low computational demand engine models for control purposes. On the other hand, the second class phenomena are easier to capture due to the lower complexity and the simplification of the combustion phenomena as static effects.

They work on the assumption that, once all the starting conditions in the beginning of the thermodynamic cycle are fixed, the outcome of the combustion process will always be the same as long as the same initial conditions are imposed. Obviously, models that solely rely on the second class phenomena will not be able to account for special highly complex phenomena that are sometimes observed (i.e. Random combustion pressure deviations inside the cylinder). This text will only be focusing on the second class of phenomena.

As shown in Fig.4.1, the engine model has several inputs and several outputs. The inputs are signals i.e. quantities that can be set by the user. The outputs are also signals rather than physical quantities and they can be used for control, simulation, model based control or other purposes.

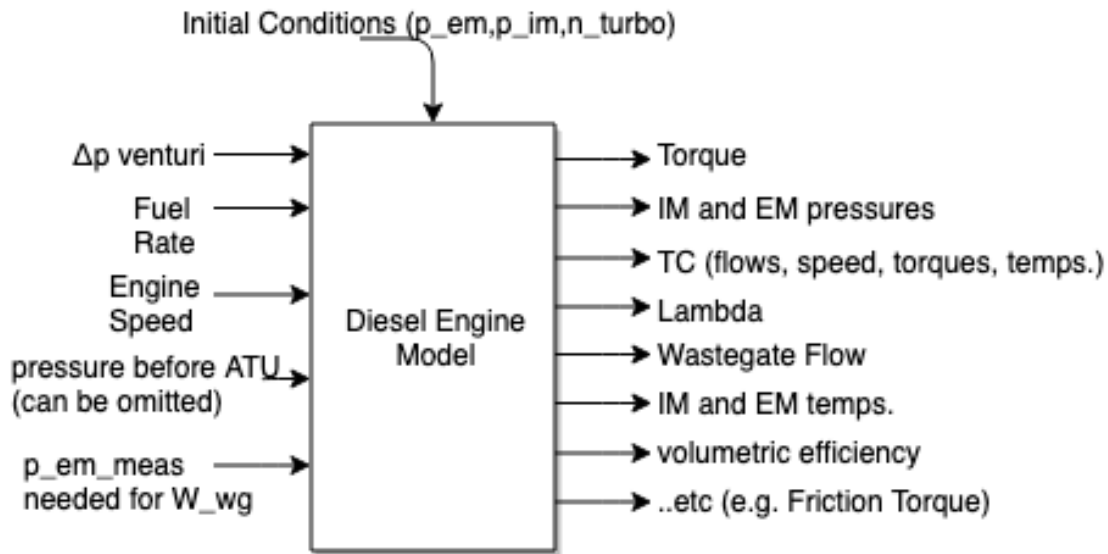


Figure 4.1: Main system's input and output signals

4.1 Cause and Effect Diagram

In this section, the internal structure of the MVM of a Diesel Engine will be shown. When modelling a physical system, there are two main classes of objects that must be taken into account:

- *reservoirs* (i.e. storages of kinetic energy, mass or information). There is an associated level variable for each reservoir that depends on the reservoir's content.
- *flows* (i.e. energy, mass etc.) flowing between the associated reservoirs (typically driven by differences in the reservoir levels).

A good MVM only contains the *relevant* reservoirs (otherwise slow and impractical models will be obtained). To define what is relevant and what is not, Fig.4.2 is representative of the answer. Signal a) is the signal of interest, signal b) is a signal which is way faster than a), therefore it must be modelled as a static variable which can possibly depend on the main signal a). On the other hand, signal c) is way slower than a) and must be considered as a constant. There is not a systematic rule on deciding which signal is of importance and what isn't however this is a result of the model needs and the model designer.

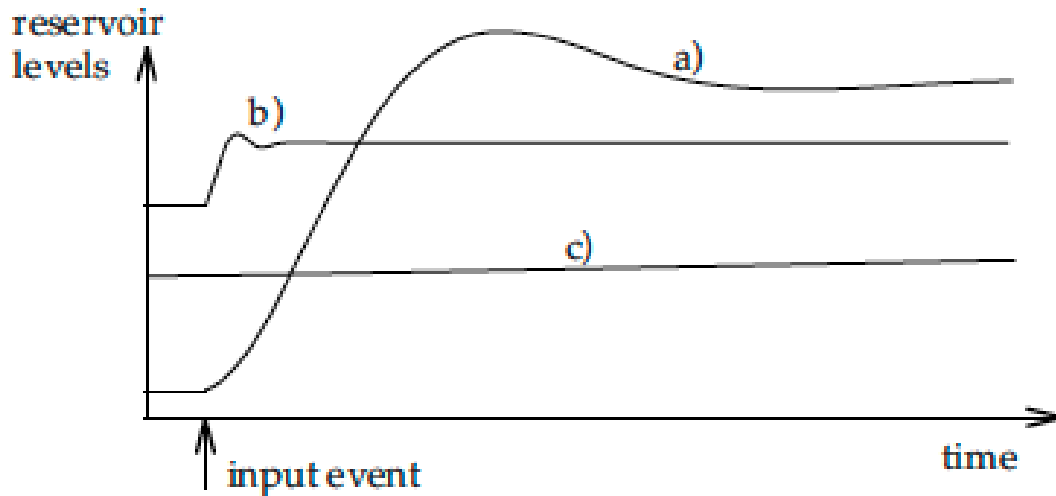


Figure 4.2: Relevant Signals, a)variable of interest, b)very fast and c)very slow variables.

The interconnection of the different models inside of the main MVM diesel model is apparent in the cause and effect diagram. In Fig.4.3 a cause and effect diagram of a diesel engine without EGR or WasteGate is shown [1]. The increased complexity from a SI engine is the existence of the turbocharger.

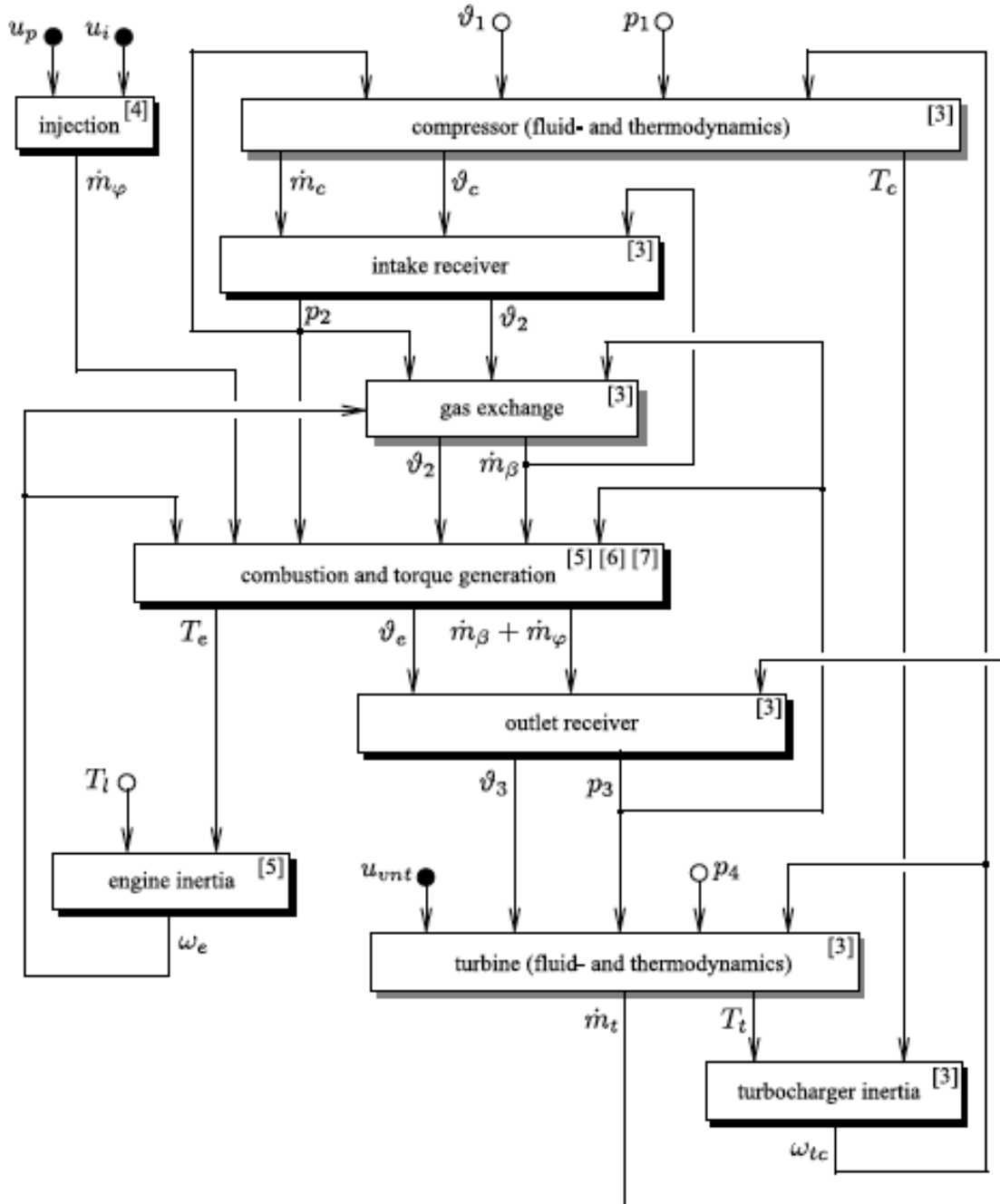


Figure 4.3: Cause and Effect Diagram of a Diesel Engine without EGR and WG

Obviously the existence of the above systems further increases the complexity.

4.2 Air System

4.2.1 Manifolds

The intake and exhaust manifolds are modelled as dynamic systems with two states each, and these are pressure and oxygen mass fraction.

4.2.1.1 Manifold Pressures

The basic building block in the air intake system as well as in the exhaust manifold system is the receiver. A receiver is basically a fixed volume that the thermodynamic states are considered to be the same throughout the said volume (i.e. temperature, pressure, composition etc.). A characteristic receiver is shown in Fig.4.4, [1].

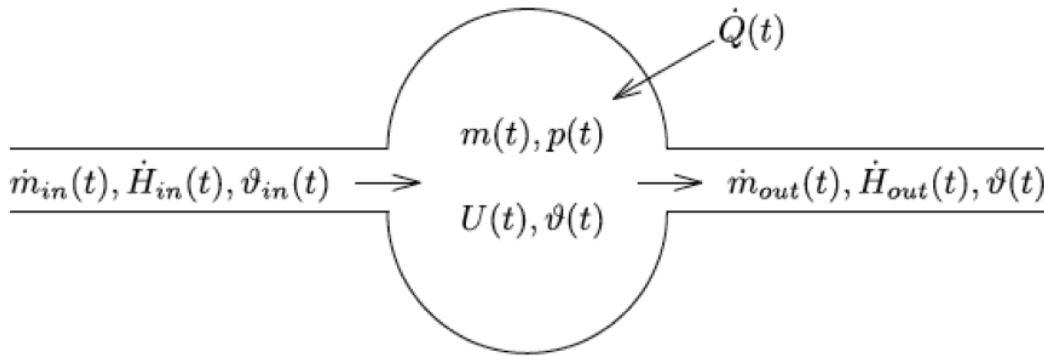


Figure 4.4: Receiver, inputs and outputs, [1]

The inputs and outputs are mass and energy flows. The receivers store mass and thermal energy in the form of pressure and temperature. If we accept, for reasons of simplicity, without sacrificing accuracy, that through the receiver's walls no phenomena of mass or heat transfer occur and also that no substantial changes in potential or kinetic energy in the flow occur then the receiver is fully described by the two differential equations below:

$$\frac{dm(t)}{dt} = \dot{m}_{in}(t) - \dot{m}_{out}(t) \quad (4.2.1)$$

$$\frac{dU(t)}{dt} = \dot{H}_{in}(t) - \dot{H}_{out}(t) + \dot{Q}(t) \quad (4.2.2)$$

Assuming that the fluids inside the receiver are ideal gases, the coupling between these two equations is given by the ideal gas law:

$$p(t)V = m(t)R\vartheta(t) \quad (4.2.3)$$

And by the caloric relations:

$$U(t) = c_v\vartheta(t)m(t) = \frac{1}{\kappa - 1}p(t)V \quad (4.2.4)$$

$$\dot{H}_{in}(t) = c_p\vartheta_{in}(t)\dot{m}_{in}(t) \quad (4.2.5)$$

$$\dot{H}_{out}(t) = c_p \vartheta(t) \dot{m}_{out}(t) \quad (4.2.6)$$

Note that the temperature of the gas coming out of a receiver is the same as the temperature of the gas inside of the receiver, as described above ($\vartheta_{out}(t) = \vartheta(t)$). Next we need to define some parameters, which are well known in thermodynamics:

- c_p , specific heat at constant pressure ($\frac{J}{kgK}$).
- c_v , specific heat at constant volume ($\frac{J}{kgK}$).
- κ , ratio of specific heats for ideal gases, $\kappa = \frac{c_p}{c_v}$, usually $\kappa = 1.4$ for atmospheric air.
- R gas constant (its quantity changes from gas to gas, R satisfies the equation $R = c_p - c_v$), units ($\frac{J}{kgK}$).

Continuing and solving on the above equations yields the following two differential equations which describe the level variables of temperature and pressure (which are the only measurable quantities in the receiver):

$$\frac{dp(t)}{dt} = \frac{R}{V} [\dot{m}_{in}(t) \vartheta_{in}(t) - \dot{m}_{out}(t) \vartheta(t)] \quad (4.2.7)$$

$$\frac{d\vartheta(t)}{dt} = \frac{\vartheta R}{pV c_v} [c_p \dot{m}_{in}(t) \vartheta_{in}(t) - c_p \dot{m}_{out}(t) \vartheta(t) - c_v (\dot{m}_{in}(t) - \dot{m}_{out}(t)) \vartheta(t)] \quad (4.2.8)$$

The adiabatic formulation 4.2.7, 4.2.8 that is described is one extreme simplification that accurately describes the receiver's behavior in cases when the gas stays for short periods of time in the receiver or when the Volume to Area ratio of the receiver is small (meaning that no heat exchange is happening, hence adiabatic). If the above is not true, the isothermal assumption is more accurate and hence a better approximation. The isothermal assumption assumes the exact opposite of the adiabatic, such a big heat exchange is happening that the gas temperature inside the receiver is practically the same as the temperature of the gas that is entering the receiver ($\vartheta(t) = \vartheta_{in}(t)$). If that is the case (which we assume that it is) ??,?? is simplified to:

$$\frac{dp(t)}{dt} = \frac{R\vartheta(t)}{V} [\dot{m}_{in}(t) - \dot{m}_{out}(t)] \quad (4.2.9)$$

Continuing on 4.2.9 and applying in to our current problem we get the following differential equations that describe pressure for the intake and exhaust manifold respectively (The time input to each function is not included, but it's obviously there):

$$\frac{dp_{im}}{dt} = \frac{R_a T_{im}}{V_{im}} [W_c + W_{ei} + W_{EGR}] \quad (4.2.10)$$

$$\frac{dp_{em}}{dt} = \frac{R_e T_{em}}{V_{em}} [W_{eo} - W_t - W_{EGR} - W_{wastegate}] \quad (4.2.11)$$

There are two sets of thermodynamic properties (marked with “a” and “e” respectively) that describe the air gas and exhaust gas. Air and exhaust gas have different gas constants “R” and different specific heat capacity ratios κ . For our kind of experimental setup (static diesel engine with water to air after cooler, meaning that it doesn’t use a common air to air intercooler) we make the assumption that the intake manifold air temperature is constant and equal to the after cooler passing water (323K).

The manifold volume parameters are clearly not easily identifiable and they are not known on first hand. On similar modelling problems, chances are that these parameters will also not be known. On this thesis these parameters were identified using least squares parameter identification on the complete model [11], [12]. This procedure runs on the complete model using the known data that were acquired from the lab setup and through the usage of least squares optimization is able to return the assumed values. What the user needs to do is set some physical boundaries on the program and know the scale of the wanted quantity, i.e. we know that the Volume number can’t be negative and we know that the scale of it is around 10^{-5} to 10^{-3} .

4.2.1.2 Oxygen Fraction in the Receivers

The Caterpillar C9.3 ACERT engine, makes use of an EGR system to reduce the NOx emissions in the exhaust gas. In order to accurately calculate the needed Lambda factor, the oxygen fractions in the receivers must be calculated, since EGR is not purely exhaust gas and contains oxygen as well. This effect is considered by modeling the oxygen concentrations X_{Oim} and X_{Oem} in the control volumes. These concentrations are defined in the same way as in reference [13]:

$$X_{Oim} = \frac{m_{Oim}}{mtot,im} \quad (4.2.12)$$

$$X_{Oem} = \frac{m_{Oem}}{mtot,em} \quad (4.2.13)$$

where m_{Oim} and m_{Oem} are the oxygen masses in the intake manifold and exhaust manifold respectively. The total masses in the intake and exhaust manifold are denounced as $mtot,im$ and $mtot,em$ respectively.

Differentiating 4.2.12 and 4.2.13 and using the law of mass conservation, we get:

$$\frac{d}{dt}X_{Oim} = \frac{R_a T_{im}}{p_{im} V_{im}} ((X_{Oem} - X_{Oim})W_{EGR} + (X_{Oc} - X_{Oim})W_c) \quad (4.2.14)$$

$$\frac{d}{dt}X_{Oem} = \frac{R_e T_{em}}{p_{em} V_{em}} (X_{Oe} - X_{Oem})W_{eo} \quad (4.2.15)$$

where X_{Oc} is the constant oxygen concentration passing through the compressor (assumed to be around 23.14%), $X_{Oc} = 23.14\%$ and X_{Oe} is the oxygen concentration in the exhaust gases coming from the engine cylinders, X_{Oe} is further described in the Volumetric Efficiency Modelling chapter.

Manifold Model I/O Diagram and Simulink Implementation

The I/O diagram of the intake manifold is shown below in Fig.4.5.

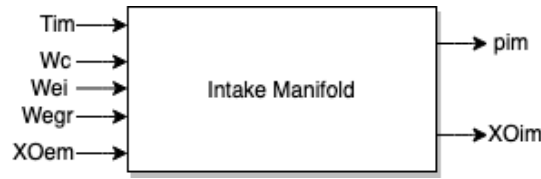


Figure 4.5: Intake Manifold I/O Diagram

Consequently the I/O diagram of the exhaust manifold is shown below in Fig.4.6.

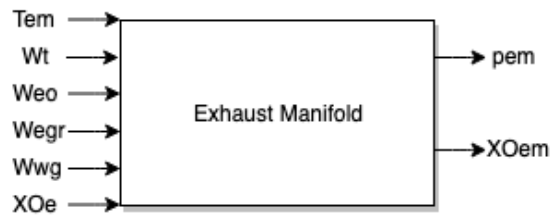


Figure 4.6: Exhaust Manifold I/O Diagram

The Simulink implementation is shown in Fig.4.7 and 4.8 respectively.

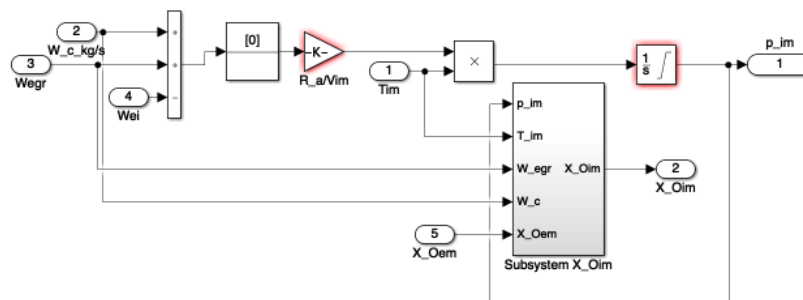


Figure 4.7: Intake Manifold Simulink Model

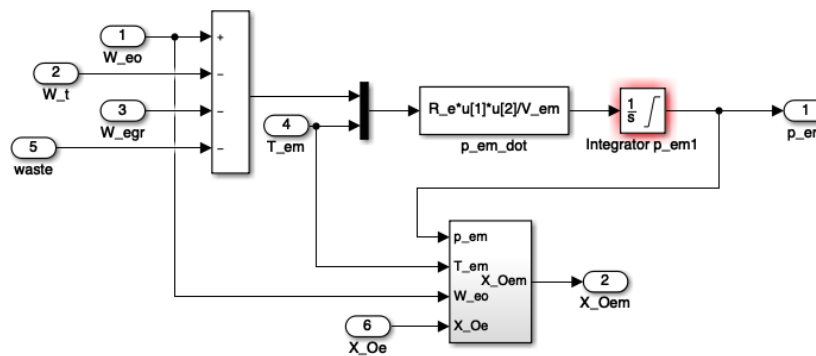


Figure 4.8: Exhaust Manifold Simulink Model

4.2.2 Engine Air Mass Flow-Volumetric Efficiency Estimation

The engine during operation can be approximated as a volumetric pump, meaning that it aspires air during its intake phase and emits exhaust gas during the exhaust phase. The air it aspires is proportional to its speed, just like a pump, meaning that the faster the engine works, the more air gets inside the engine. A typical differential equation that describes the above phenomenon is the following:

$$\dot{m}_{air,in}(t) = \rho_{in}(t)\dot{V}(t) = \rho_{in}(t)\eta_{volumetric}(p_{im}, \omega_e) \frac{V_d \omega_e(t)}{N} \frac{1}{2\pi} \quad (4.2.16)$$

Where the unknown beforehand variables as described as below:

- ρ_{in} the density of the intake gas (it is related to the intake air pressure and temperature through the ideal gas law, $p = \rho RT$);
- η_{vol} the volumetric efficiency, it describes how much the engine simulates an ideal volumetric pump, 1 = perfect pump, obviously the volumetric efficiency of the engine is always < 1 ;
- V_d the engine's displaced volume, for the CAT C9.3 $V_d = 9.3$ liters or $0.0093m^3$;
- N the number of revolutions per cycle, for a four stroke engine $N = 2$ and for a two stroke engine $N = 1$

Continuing on 4.2.16 using the ideal gas law, we get the following equation that simplifies the measurement of the engine intake air flow:

$$\dot{m}(t) = W_{ei} = \frac{\eta_{volumetric} p_{im} n_e V_d}{120 R_a T_{im}} \quad (4.2.17)$$

The fuel rate is known from the simulation data. Keeping in check with the mass conservation law, the mass of the burnt gas exiting the engine is:

$$W_{eo} = W_{ei} + W_f + W_{EGR} \quad (4.2.18)$$

Where W_f is the fuel rate that gets burned in the cylinder. Volumetric efficiency describes the ability of an engine to aspire air. Accurately predicting the volumetric efficiency is rather difficult. Many effects influence it i.e. EGR rate, ram effects in the intake manifolds, resonance in the manifold, cross coupling of different cylinders, intake manifold geometry etc. Having an accurate estimation of the volumetric efficiency is of paramount importance as it has an effect on manifold pressures, lambda estimation etc.

Volumetric efficiency η_{vol} , can be normally estimated through detailed CFD simulations, that can account for all the influencing factors (i.e. manifold geometry, ram effects etc.) but it is very consuming on time and processing power. For the purpose of a control oriented model, this approach is off limits. Another way of computing the volumetric efficiency is through the creation of a two dimensional look-up table/ map of mean effective pressure p_{em} and rotational engine speed n_e . Also, this approach involves running and mapping the engine throughout its operational limits, which again is very time consuming and inefficient. In addition, this

approach is less accurate for diesel engine engines, as the volumetric efficiency is not solely influenced by mean effective pressure p_{em} and engine speed n_e .

As it is already noted on Chapter 1, we were not able to gather data regarding the mass flows of air inside the engine. Meaning that W_{ei} is not a known parameter. Thankfully, the engine has sensors on the after-treatment system after the exhaust, right before the Diesel Particulate Filter entrance (point 2 at Fig.4.9). Therefore, we could acquire the mass flow of the exhaust gas through the filter and then consequently using 4.2.18, knowing the fuel rate, and the estimated egr rate (more on its estimation on the corresponding chapter), we could compute W_{ei} .

Note: the flow through the DPF, does not include the EGR rate, which continuously circles between the exhaust and intake manifold!

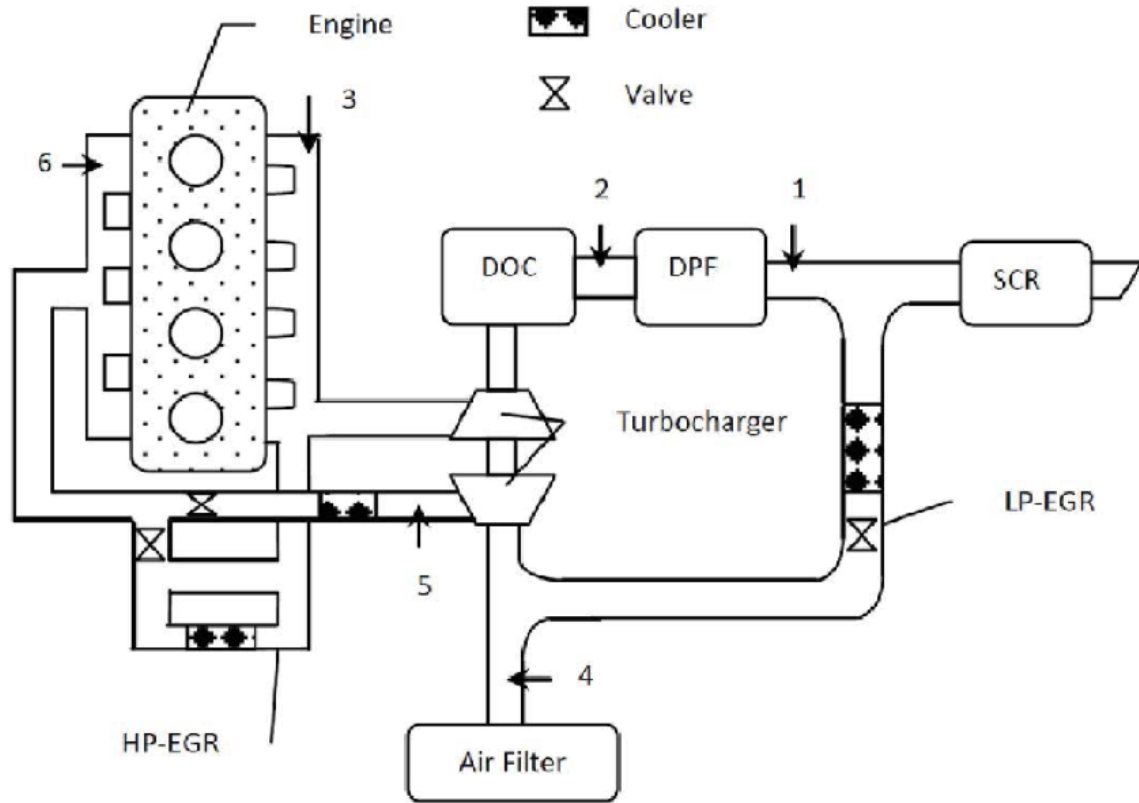


Figure 4.9: Typical Aftertreatment System Layout of Diesel Engine

$$W_{eo} = W_{DPF,in} + W_{EGR} \quad (4.2.19)$$

Subtracting the known fuel rate from the W_{eo} we get:

$$W_{eo} - W_f = W_{ei} \quad (4.2.20)$$

Solving 4.2.16 for $\eta_{volumetric}$ we get:

$$\eta_{volumetric} = \frac{W_{ei} 120 R_a T_{im}}{p_{im} n_e V_d} \quad (4.2.21)$$

The proposed model for the volumetric efficiency computation was developed in [3]. The inputs of the volumetric efficiency model are the square roots of both p_{im} and n_e . The proposed model is shown in 4.2.22.

$$\eta_{volumetric} = c_{vol1}\sqrt{p_{im}} + c_{vol2}\sqrt{n_e} + c_{vol3} \quad (4.2.22)$$

Where the constants $c_{vol1}, c_{vol2}, c_{vol3}$ are the model tuning parameters.

When the EGR rate was assumed, without knowing the EGR fraction from Data (more on the corresponding chapter), special attention was given in order to maintain computed volumetric efficiencies that agree with the corresponding Literature ?? (Scania V12 Engine with EGR and VGT, about 12liters in displacement). Typical volumetric efficiencies of modern turbocharged diesel engines with EGR were found to be around 0.8 to 1.2.

Model I/O Diagram and Simulink Implementation

The I/O diagram that describes the model proposed above that calculates the η_{vol} volumetric efficiency and the mass flow of the air entering the engine is shown in Fig.4.10.

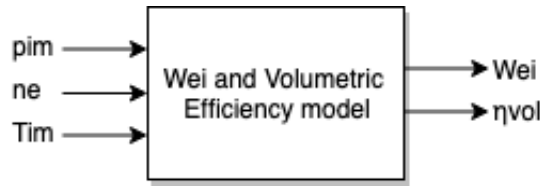


Figure 4.10: I/O Diagram of Volumetric Efficiency Model

The Simulink implemented model that describes the model proposed above that calculates the η_{vol} volumetric efficiency and the mass flow of the air entering the engine is shown in Fig.4.11.

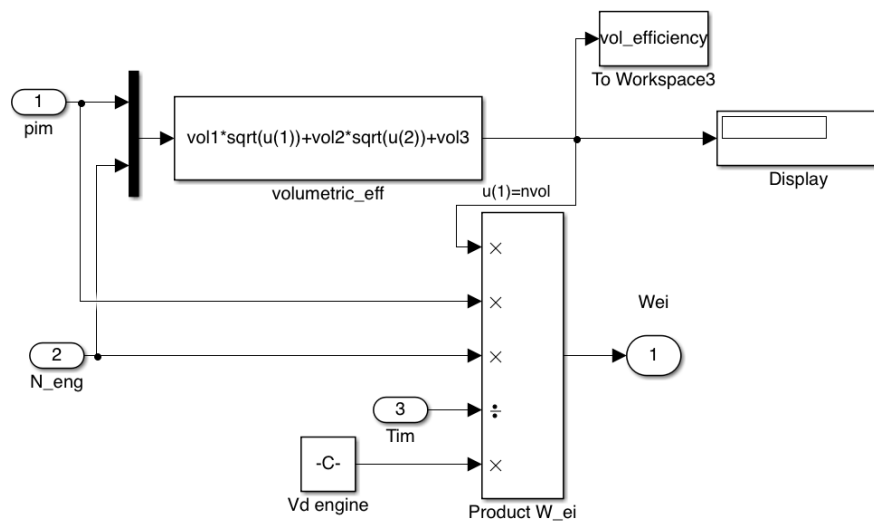


Figure 4.11: Simulink implemented model for volumetric efficiency

4.2.3 Oxygen to Fuel Ratio inside Cylinder (Lambda Calculation)

In order to accurately capture the engine's Lambda measurement with the presence of EGR, the EGR oxygen fraction must be accounted for. Measuring Lambda simply as the fraction of W_{ei} to W_f would be inaccurate due to the presence of oxygen in EGR. Therefore the oxygen to fuel ratio is introduced, which is simply the same as the usual Lambda but instead of air to fuel, it measures oxygen to Fuel. The oxygen to fuel ratio inside the cylinder is defined as:

$$\lambda_O = \frac{W_{ei}X_{Oim}}{W_f(O/F)_s} \quad (4.2.23)$$

where $(O/F)_s = A/F_{stoich}X_{Oc} = 14.57 * 0.2314$ for diesel fuel. During the combustion, the oxygen is burned in the presence of fuel. In diesel engines, λ_O must be above 1 to avoid smoke. The oxygen concentration out from the cylinder can be calculated as the unburned oxygen fraction:

$$X_{Oe} = \frac{W_{ei}X_{Oim} - W_f(O/F)_s}{W_{eo}} \quad (4.2.24)$$

Model I/O Diagram and Simulink Implementation The I/O diagram that describes the model proposed above that calculates the Λ_{O} is shown in Fig.4.12.

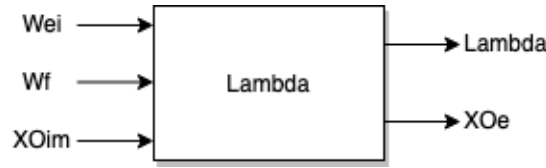


Figure 4.12: I/O Diagram of Lambda Model

The Simulink implemented model that describes the model proposed above that calculates the Λ_{O} is shown in Fig.4.13.

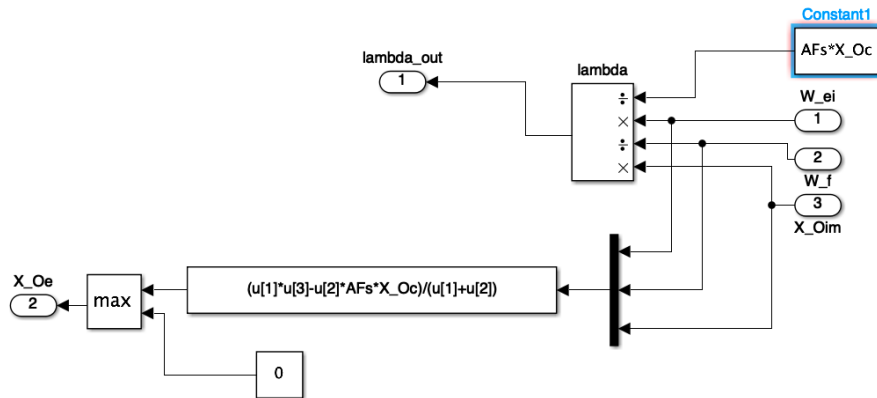


Figure 4.13: Simulink implemented model for Lambda

4.3 Exhaust Gas Recirculation-EGR Mass Flow approximation

For purposes of environmental protection, the engine's harmful emissions must be controlled. The Caterpillar C9.3 ACERT has numerous emission control systems, including EGR. EGR or Exhaust Gas Recirculation is either the direct or cooled recirculation of a portion of the engine's exhaust gas back to the cylinder through the intake manifold. With the use of EGR, the cylinder combustion temperatures drop significantly and so does the O₂ concentration (as Exhaust Gas takes the space that would otherwise be fresh air). Therefore as a result the production of NO_x particles is greatly reduced, considering that their chemical creation mechanism relies on the abundance of [O₂] and High Temperatures.

In order to model the EGR mass flow fraction, it is necessary that some assumptions are made:

1. The exhaust gas is collected immediately from the exhaust manifold through the EGR valve (no time delays or dynamic gas mixing);
2. The EGR is considered to be cooled, so that the intake manifold final temperature is cool and the isothermal assumption of 4.2.11 is true;
3. The exhaust gas and the fresh air entering the IM are perfect gases with the same $Ra = 287 \frac{J}{kgK}$;
4. The mixture is lean (diesel engine).

Usually mean value models of engines model the EGR flow from data acquired from engine sensors. In this instance no such measurements were available. The EGR mass flow was estimated using the engine EGR Venturi flow meter setup.

In the engine, the EGR flow goes through a Venturi Pipe 4.14 that has a pressure meter on it so that the ECU can know the amount of EGR that goes through the engine. The pressure sensor is measuring the difference in pressure from one pipe diameter to the other Δp . Having this data in real time and also by measuring the piping diameters in the engine itself we were able to measure and estimate the EGR flow at any given time.

The Venturi flow meters/ valves are well described by these equations that are derived from Bernoulli's law as long as the flow is sufficiently subsonic ($V < 0.3$ Mach):

$$p_a - p_b = \Delta p = \frac{1}{2} \rho V_b^2 - \frac{1}{2} \rho V_a^2 \quad (4.3.1)$$

From continuity, the throat velocity V_b can be substituted from 4.3.3 to give:

$$\Delta p = \frac{1}{2} \rho V_a^2 \left[\left(\frac{A_a}{A_b} \right)^2 - 1 \right] \quad (4.3.2)$$

Solving for the upstream velocity and multiplying by the cross-sectional area A_a gives the volumetric flow rate Q :

$$Q = \sqrt{2 \frac{\Delta p}{\rho}} \frac{A_a}{\sqrt{\left(\frac{A_a}{A_b} \right)^2 - 1}} \quad (4.3.3)$$

Finally multiplying the volumetric flow rate with the exhaust gas density that is calculated from the ideal gas law (assuming same Temperature as the intake Manifold, since the EGR is cooled), we get the EGR mass flow rate:

$$\rho_e = \frac{p_{exhaust}}{R_e T_{im}} \quad (4.3.4)$$

$$\dot{m}_{egr} = \rho_e Q \quad (4.3.5)$$

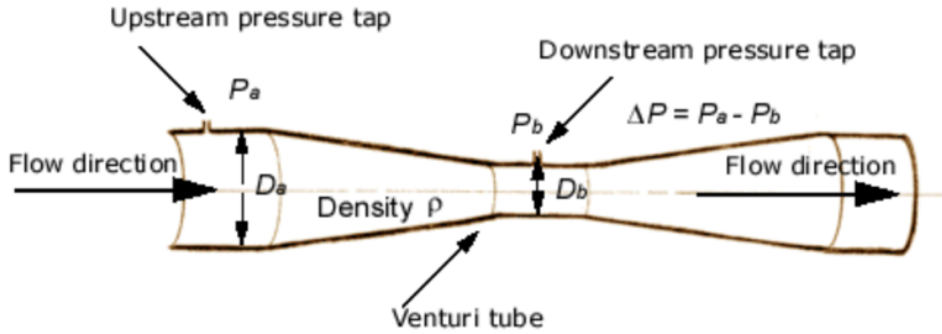


Figure 4.14: Venturi Orifice, similar to the one installed in the Engine

The outer pipe diameters were measured in the engine Lab using a caliper and were found to be: $D_a = 40mm$ and $D_b = 30mm$ respectively. Assuming an outer wall of $1cm$, we get the inner diameters of $D_a = 20mm$ and $D_b = 10mm$. Here lies the error of the EGR approximation, the inner orifice diameters are estimated.

Due to the lack of Data, this assumption is the only way to implement EGR on the engine model. However, for these diameter values, the engine runs on around 0.95 volumetric efficiency, which is logical for this kind of Diesel Engine, so this assumption can't be too inaccurate. The EGR estimated flow, mainly determines the volumetric efficiency of the Engine, meaning that for greater EGR flows, greater volumetric efficiencies are required since for known Intake and Exhaust Manifold pressures and known Turbocharger operation (which yields fairly known compressor and turbine flow values) if the EGR flow increases, the volumetric efficiency would have to counter the increase in order for the Pressure inside the manifolds to stay within the known bounds.

Model I/O Diagram and Simulink Implementation

The I/O diagram that describes the model proposed above that calculates the W_{egr} is shown in Fig.4.15.

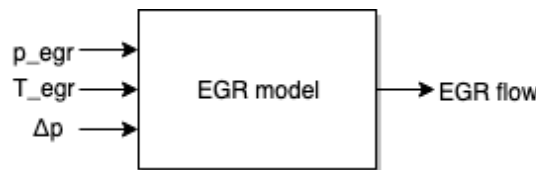


Figure 4.15: I/O Diagram of EGR flow

The Simulink implemented model that describes the model proposed above that calculates the W_{egr} is shown in Fig.4.16

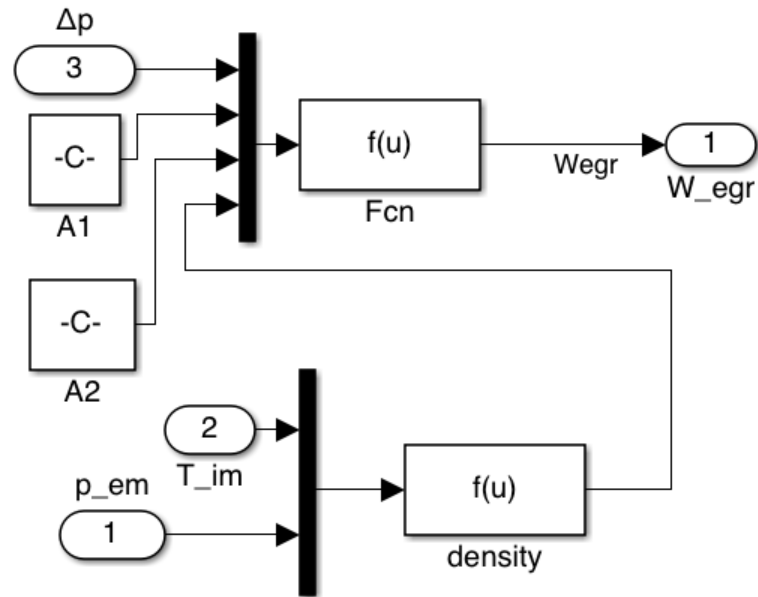


Figure 4.16: Simulink implemented model for EGR flow

4.4 Forced Induction-Turbocharger

Thus far the ability of an engine to pump air was discussed and calculated. In naturally aspirated engines, the air that enters the cylinders is limited from the engine's volumetric efficiency. Therefore, the amount of torque a Naturally Aspirated engine produces is tightly connected to its volumetric efficiency as well. Turbochargers are used to increase the maximum power that can be obtained from a given displacement engine. The work transfer to the piston per cycle, in each cylinder, which controls the power the engine can ultimately deliver, depends on the amount of fuel burned per cylinder per cycle. This depends on the amount of fresh air that is inducted each cycle. Increasing the air density prior to entry into the engine thus increases the maximum power that an engine of given displacement can deliver. Consequently modern engines make use of forced induction, specifically CAT C9.3 uses a fluid dynamic turbocharger, in order to produce more power and torque than a NA counterpart. Another great benefit of using forced induction in diesel engines is the great decrease of visible black smoke in the exhaust since black smoke is evidence of lack of air for the complete oxidization of the fuel that was injected in the cylinder. [14]

Turbochargers are fluid dynamic devices consisting of a compressor and a turbine in different housings that are connected on the same shaft. They spin on speeds of up to even 200,000 rpm (seen in smaller sized turbochargers). The turbocharger, a compressor-turbine combination, uses the energy available in the engine exhaust stream to achieve compression of the intake flow. The air flow passes through the compressor, intercooler and finally the intake manifold. Engine inlet pressures (or boost) of 150 kPa above atmospheric pressure are typical. The exhaust flow through the exhaust manifold drives the turbine which powers the compressor.

In order to prevent the built up intake manifold pressure from reaching excessively high numbers, the turbocharger speed must be somehow controlled and limited, especially on instances of high load and high throttle operation. That's where the waste gate comes to existence. Waste-gates operate as a controlled opening valve and can be either internal (installed inside the turbine housing) or external. They let some of the exhaust gas mass flow escape to the exhaust exit directly without going through the turbine of the turbocharger. That way, the exhaust gas that enters the turbine has decreased pressure and therefore the turbocharger rotational speed is kept in check. The waste-gate system along with the turbocharger (no intercooler) is fairly explained in 4.17:

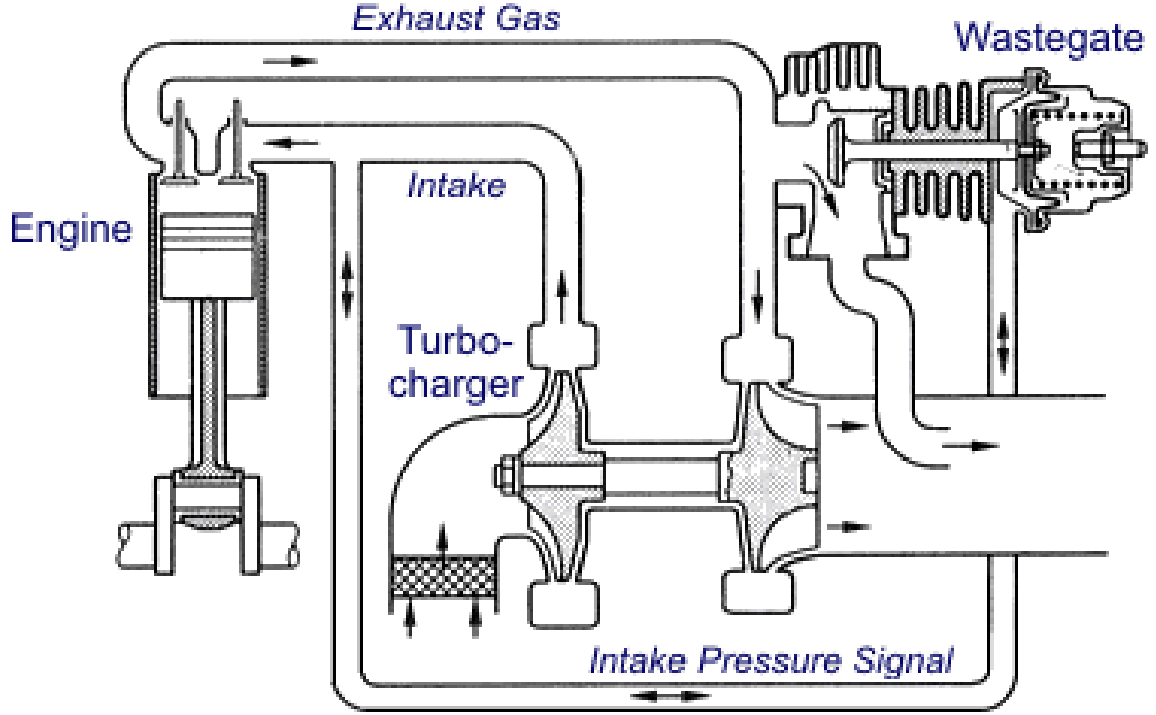


Figure 4.17: Wastegated Turbocharger

The turbocharger system model can be described by three separate submodels:

- The turbocharger mechanical inertia which is given by the difference in the Turbine (positive) Torque $M_{turbine}$ and the compressor absorbed (negative) Torque M_{comp} .
- The compressor static behavior, with pressure ratio $\frac{p_2}{p_1}$, turbocharger speed n_{tc} and ambient temperature T_{amb} as system inputs and Compressor mass flow Q_c , compressor isentropic efficiency η_c , air temperature after the compressor T_{cx} and absorbed torque M_{comp} as system outputs.
- The turbine static behavior, with pressure ratio $\frac{p_3}{p_4}$, turbocharger speed n_{tc} , exhaust gas temperature T_e as system inputs and turbine flow Q_t , turbine created torque $M_{turbine}$, turbine isentropic efficiency η_t and exhaust gas temperature after the turbine temperature T_{em} as system outputs.

4.4.1 Turbocharger Speed

The turbocharger speed is described from Newton's second law in 4.4.1:

$$\frac{d\omega_{turbocharger}}{dt} = \frac{M_t - M_c}{J_{turbocharger}} \quad (4.4.1)$$

Where,

- M is torque, t denoting turbine and c compressor.
- $J_{turbocharger}$ is the rotational inertia of the turbocharger rotating parts. Obviously, the turbo inertia is hard to measure without the use of special test benches

and tools. The inertia was calculated through least squares optimization on the complete engine model.

Model I/O Diagram and Simulink Implementation The I/O diagram that describes the model proposed above for the n_{turbo} , turbocharger rotational speed is shown in Fig.4.18. The Simulink implemented model that describes the model

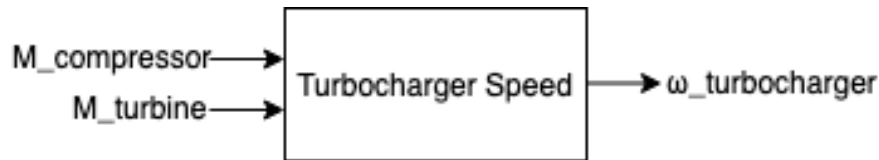


Figure 4.18: I/O Diagram of Turbocharger Speed

proposed above for the n_{turbo} , turbocharger rotational speed is shown in Fig.4.19.

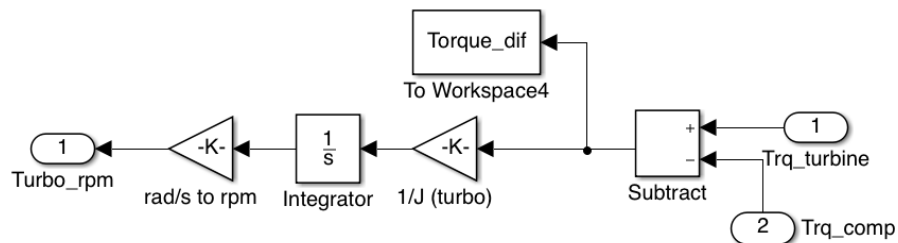


Figure 4.19: Simulink implemented model for Turbocharger Speed

4.4.2 Compressor Modelling

The operating characteristics of a centrifugal compressor are usually described by its unique *performance map*. This map, shows lines of constant compressor efficiency η_c and constant corrected speed $n_{turbocharger}$ on a plot of pressure ratio $\Pi_c = \frac{p_2}{p_1}$ against corrected mass flow W_c . The Caterpillar® C9.3™ uses a Garrett® GT3782™ compressor. The performance map of this compressor is shown in Fig.4.20.

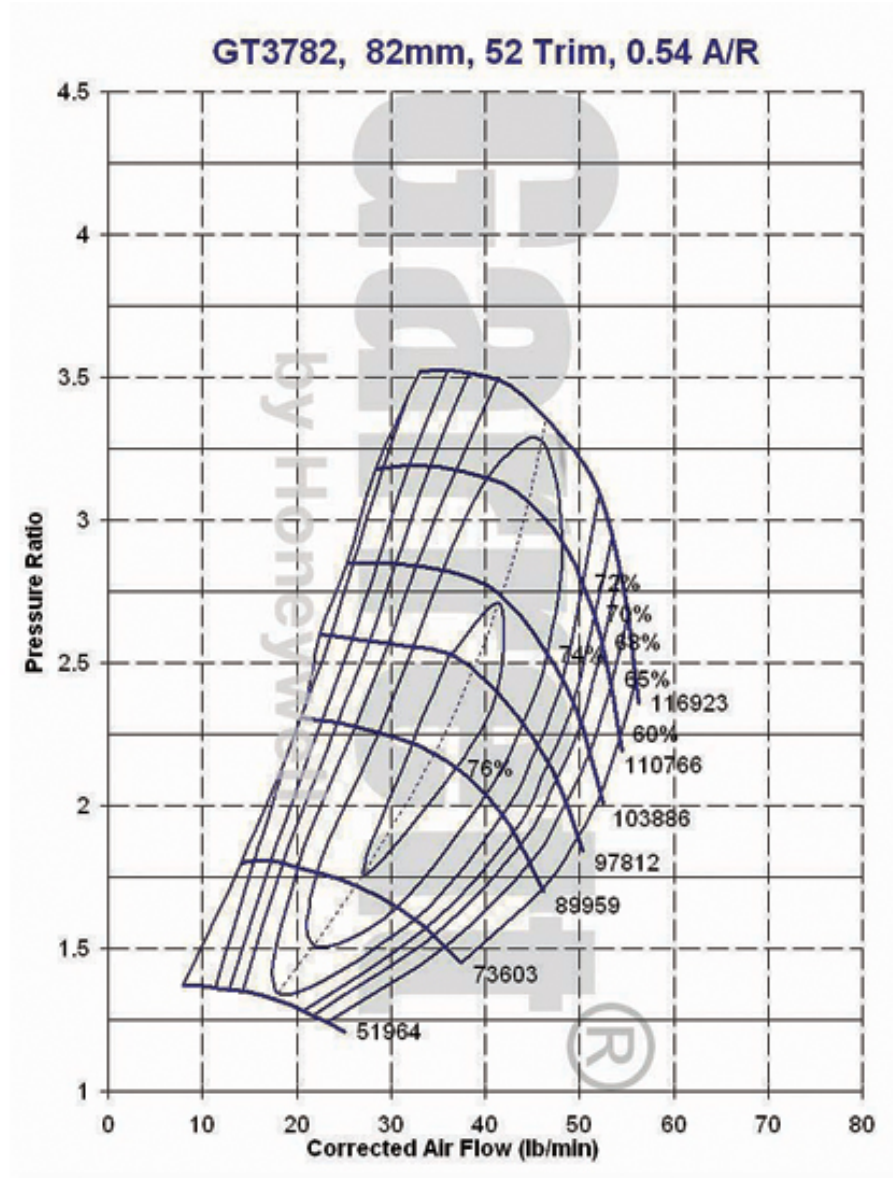


Figure 4.20: Garrett GT3782 Compressor Performance Map, [2]

Points inside the boundaries of the performance map indicate the points of stable operation of the compressor. The leftmost boundary curve that limits the performance map is called the *surge line*, which presents an unstable region. When the mass flow is reduced at a constant pressure ratio, local flow reversal eventually occurs in the boundary layer. Further reductions in mass flow cause the flow to reverse completely, causing a drop in pressure. This relieves the adverse pressure gradient.

The flow reestablishes itself, builds up again, and the process repeats. Compressors should not be operated in this unstable regime [14]. The rightmost curve is called the *choking line*. The velocity increases as the compressor mass flow increases and eventually becomes sonic in the limiting area of the compressor. The only safe way to increase the compressor mass flow, is through an increase in rotating speed. The compressor is also mechanically limited, meaning that it has a maximum designed rotational speed, operation beyond this speed (in this case, 116923 rpm) will result in mechanical damage and therefore failure of the compressor. Therefore the compressor is meant to operate inside the boundaries of its compressor map.

The compressor performance map is created in dedicated test benches by the manufacturer where the compressor is driven by burnt natural gas and is tested without the existence of an engine and all the necessary conditions are set.

In order to model this compressor for our engine model purposes, an exponential function was used for the compressor flow and a polynomial function was used for the efficiency. On first attempt a lookup table was used for the compressor in order to draw the wanted data. This approach yielded problems during the operation of the program and the resulting accuracy was not acceptable, therefore the second approach described above was used.

4.4.2.1 Modelling the Compressor Flow Rate

The model that calculates the compressor mass flow rate was developed at first from [9] Jensen and Kristensen in which the compressor characteristics are derived from the dimensionless pressure head Ψ , the normalized compressor flow rate Φ and the inlet Mach number M given as below:

$$\Psi = c_p T_{amb} \frac{\Pi_c^{\frac{\kappa-1}{\kappa}} - 1}{0.5 U_c^2} \quad (4.4.2)$$

Where,

$$U_c = \frac{\pi}{60} d_c n_c \quad (4.4.3)$$

$$\Phi = \frac{Q_c}{\frac{\pi}{4} d_c^2 U_c} \quad (4.4.4)$$

and

$$M = \frac{U_c}{\sqrt{\kappa R T_{amb}}} \quad (4.4.5)$$

Where the unknown parameters are described below:

- T_{amb} . the ambient temperature in K .
- d_c , the compressor diameter in m .
- n_c , the compressor rotational speed in $\frac{rotations}{min}$.
- U_c , the compressor blade tip speed in $\frac{m}{s}$.
- Q_c , the compressor mass flow in $\frac{lb}{min}$.
- Π_c , the compressor pressure ratio.

For each compressor speed line seen in the map (i.e. 51964 rpm) Ψ and Φ were calculated using actual map data. The result is seen below in Fig. 4.21:

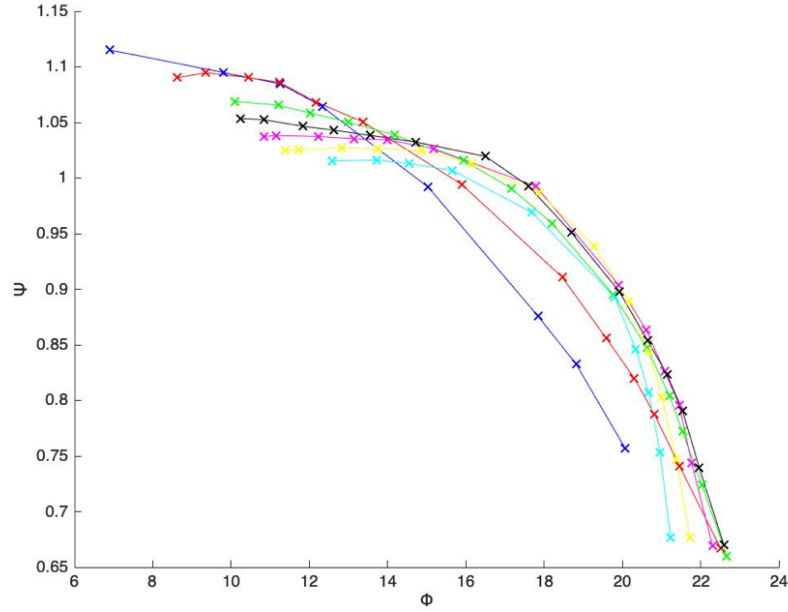


Figure 4.21: Speed curves given in dimensionless parameters Ψ and Φ coordinates

It is somewhat obvious from visual inspection that the dimensionless speed curves follow a negative exponential function of pressure head Ψ . The above curves that practically consist of the marked points were accurately fitted by the following exponential function [15]:

$$\Phi = a + (1 - e^{\Psi^{c+b}}) \quad (4.4.6)$$

Where the shape coefficients a , b and c are introduced for better fitment. The shape coefficients are fitted by a polynomial function of the Mach number M , so that they adapt to the different speeds and hence the different curve shapes.

- $a = a_m + M^m + a_{m-1}M^{m-1} + \dots + a_1M + a_0$
- $b = b_n + M^n + b_{n-1}M^{n-1} + \dots + b_1M + b_0$
- $c = c_l + M^l + c_{l-1}M^{l-1} + \dots + c_1M + c_0$

Fitting Φ to the model described in 4.4.6 yielded the below results for a,b,c (Table4.1). The fitting result for the above speed curves is shown in Fig4.22.

Table 4.1: Results of fitting for a,b,c.

Speed Curve (ascending)	a	b	c	Mach
1	22.22	1.06	4.838	0.6491
2	26.39	1.463	3.814	0.9195
3	24.8	1.219	6.054	1.1238
4	22.6	0.7824	13.79	1.2219
5	22.76	0.9527	13.83	1.2977
6	23.1	1.141	11.9	1.3837
7	23.32	0.9545	9.703	1.4606

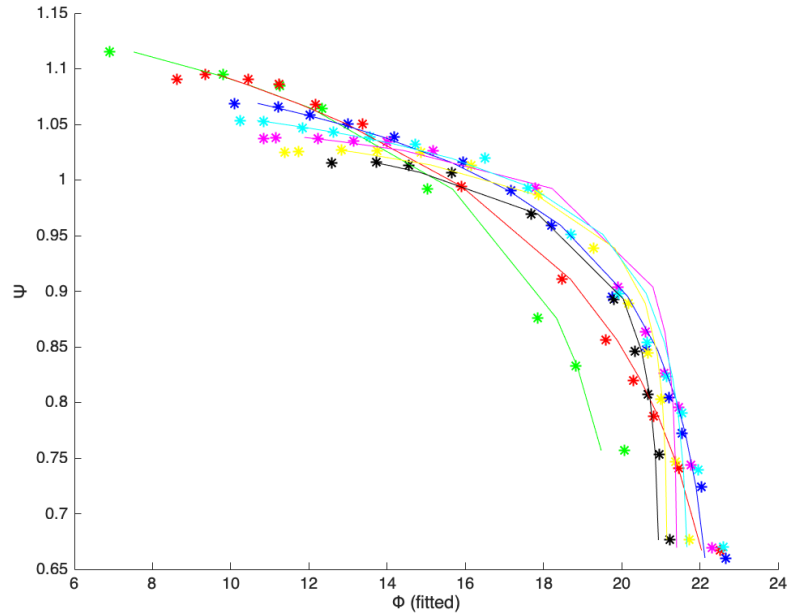


Figure 4.22: Fitting Φ using equation 4.4.6.

The result of fitting a, b, c for different Mach numbers with a third order polynomial as shown in the model above is seen in Fig.(4.23,4.24,4.25) respectively.

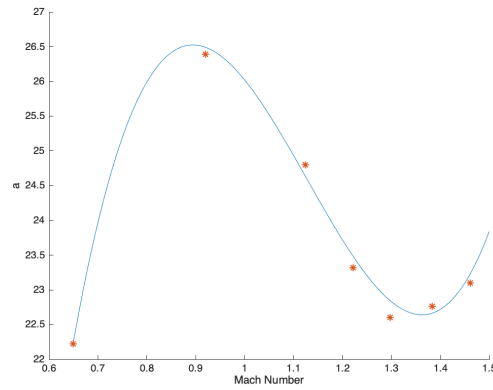


Figure 4.23: Fitting a to the Mach number using a third order polynomial equation.

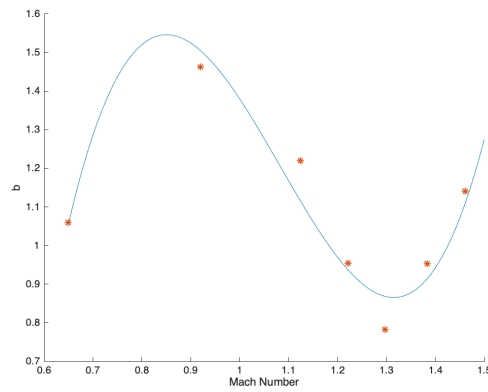


Figure 4.24: Fitting b to the Mach number using a third order polynomial equation.

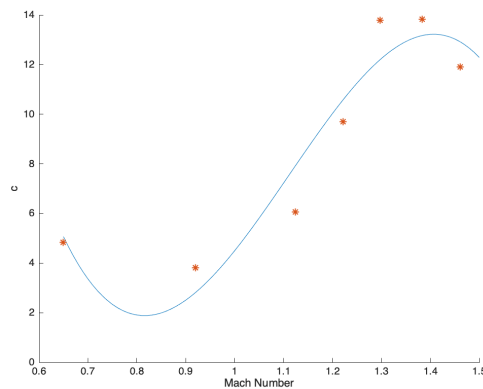


Figure 4.25: Fitting c to the Mach number using a third order polynomial equation.

Finally the compressor mass flow rate (W_c in $\frac{lb}{min}$) can be derived from 4.4.7.

$$Q_c = \Phi \frac{\pi}{4} d_c^2 U_c \quad (4.4.7)$$

Finally, the compressor performance map that describes the modelled flow-rate in connection with the turbocharger speed and Pressure ratio is derived:

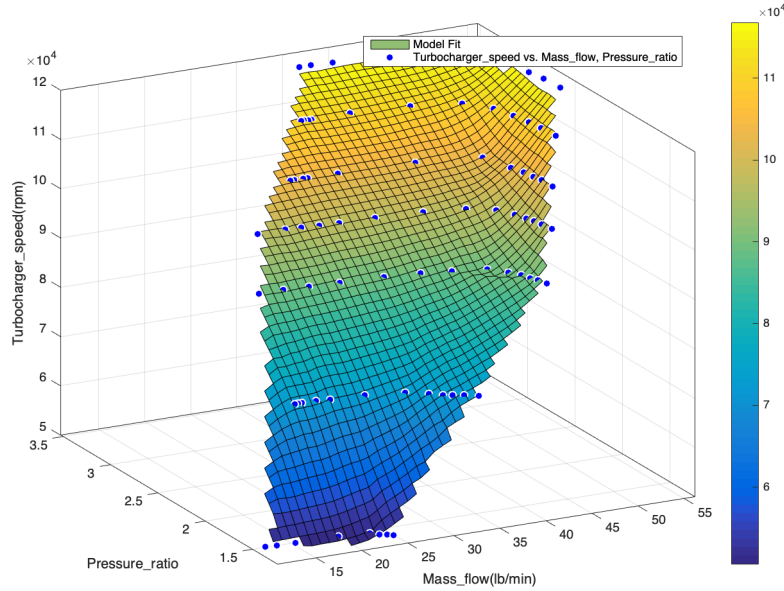


Figure 4.26: 3D GT3782 Fitted Compressor Performance Map.

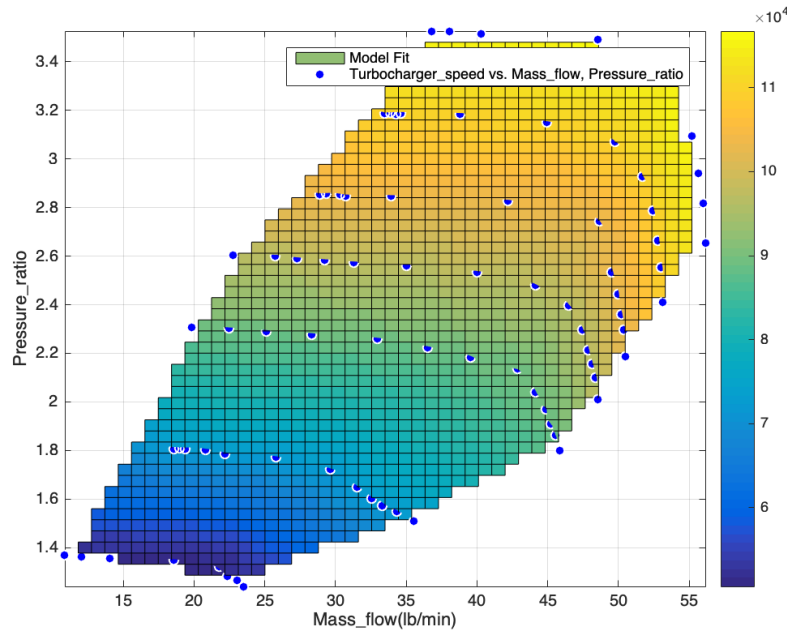


Figure 4.27: 2D GT3782 Fitted Compressor Performance Map.

Model I/O Diagram and Simulink Implementation

The I/O diagram that describes the model proposed above that calculates the W_c , Compressor mass flow is shown in Fig.4.28.

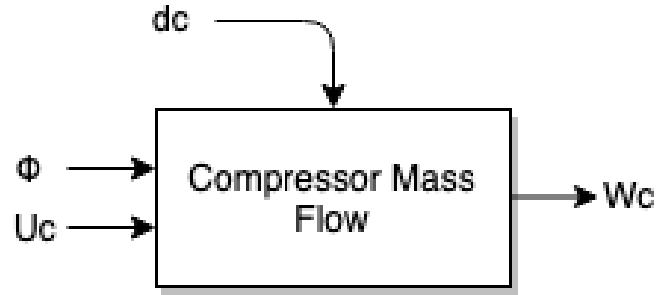


Figure 4.28: I/O Diagram of Compressor Mass Flow

The Simulink implemented model that describes the model proposed above that calculates the W_c , Compressor mass flow is shown in Fig.4.29.

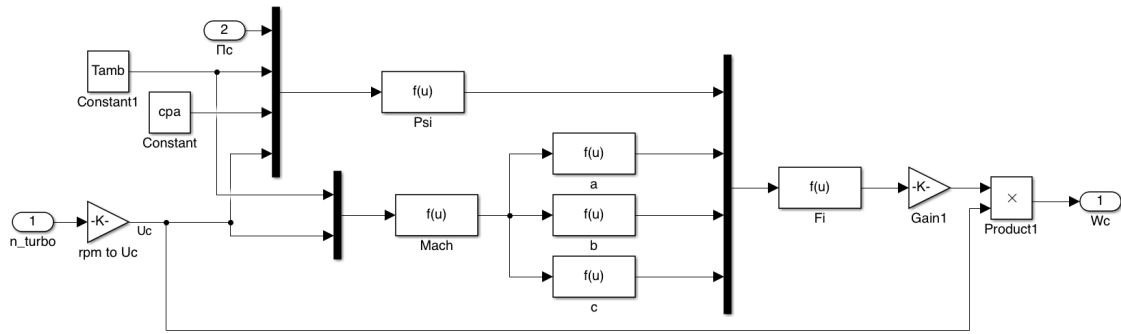


Figure 4.29: Simulink implemented model for Compressor Mass Flow

4.4.2.2 Modelling the Compressor Efficiency

The isentropic efficiency of the compressor is presented in the performance map in the form of ellipsoids as seen in Fig.4.20. The compressor efficiency is a parameter of great importance, as it impacts both the torque M_c the compressor absorbs and the temperature T_{cx} of the compressed air after the compressor. Usually automotive compressors tend to present the highest efficiency in an imaginary line that passes through the middle of the distance between the choke line and the surge line. A well designed compressor presents high isentropic efficiencies for a wide spectrum of rotational speeds.

The compressor efficiency is substantially easier to model compared to the mass flow rate that was analyzed above. The proposed model seen in [15] models the compressor's isentropic efficiency with two input variables. The proposed model is presented in 4.4.11:

$$\eta_{compressor} = f(Q_c, \Pi_c) \quad (4.4.8)$$

The model seen in [15] proposes that a polynomial fitting approach is attempted since the model is only based on two inputs. Indeed, a fourth order polynomial fit was chosen since it presents high accuracy. The polynomial modelling of the isentropic efficiency is seen in 4.4.9.

$$\begin{aligned} \eta_{compressor} = f(x, y) = & p_{00} + p_{10}x + p_{01}y + p_{20}x^2 + p_{11}xy + \\ & p_{02}y^2 + p_{30}y^3 + p_{21}x^2y + p_{12}xy^2 + p_{03}y^3 + p_{40}x^4 + p_{31}x^3 + \\ & p_{22}x^2y^2 + p_{13}xy^3 + p_{04}y^4 \end{aligned} \quad (4.4.9)$$

Where $x = Q_c$ and $y = \Pi_c$.

The fitting result is shown in Fig.4.30. The two dimensional map of the isentropic efficiency is shown in Fig.4.31.

Having the complete compressor modelled, we can calculate the temperature rise of the intake air inside the compressor T_{cx} and also the also the compressor absorbed torque M_c .

$$T_{cx} = T_{amb} \left(1 + \frac{1}{\eta_c} \left[\frac{p_{cx}}{p_{amb}}^{\frac{\gamma-1}{\gamma}} - 1 \right] \right) \quad (4.4.10)$$

$$M_c = \frac{W_c c_p T_{amb}}{\eta_c \omega_{tc}} \left[\frac{p_{cx}}{p_{amb}}^{\frac{\gamma-1}{\gamma}} - 1 \right] \quad (4.4.11)$$

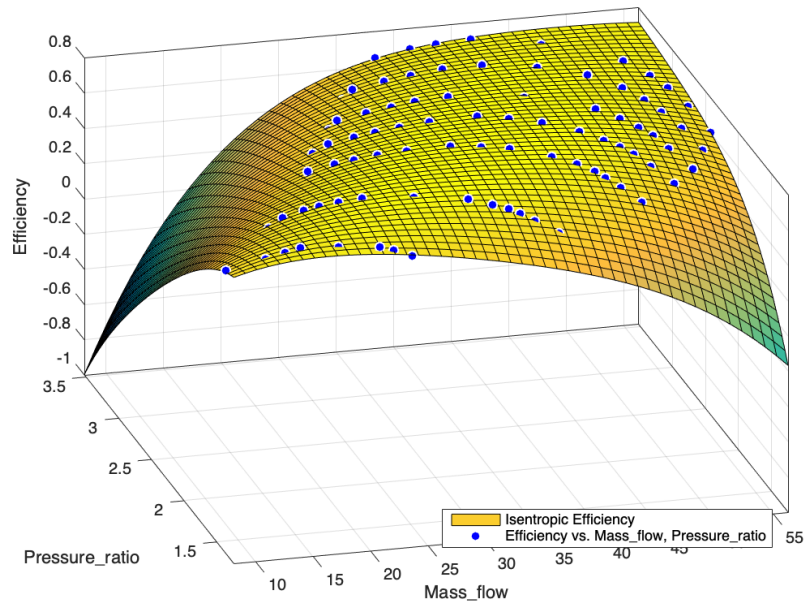


Figure 4.30: Fourth order Polynomial Fitting for the Compressor Isentropic Efficiency

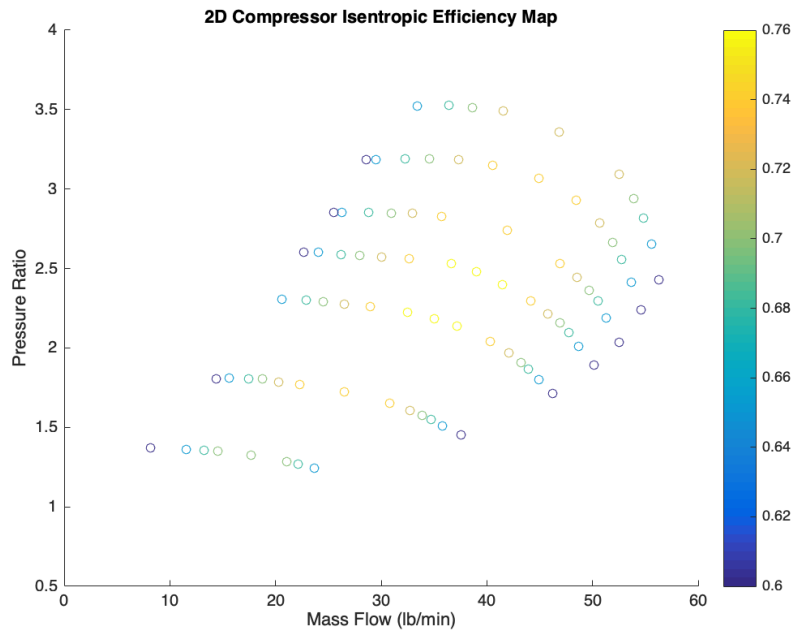


Figure 4.31: 2D Isentropic Efficiency Model

Model I/O Diagram and Simulink Implementation

The I/O diagram that describes the model proposed above that calculates the $\eta_{compressor}$, Compressor isentropic efficiency is shown in Fig.4.33.

The Simulink implemented model that describes the model proposed above that calculates the $\eta_{compressor}$, Compressor isentropic efficiency is shown in Fig.4.33.

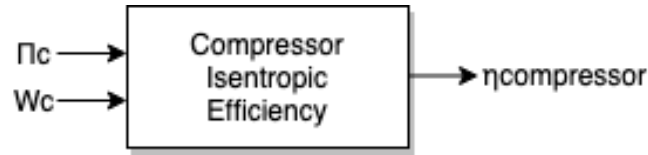


Figure 4.32: I/O Diagram of Compressor Isentropic Efficiency

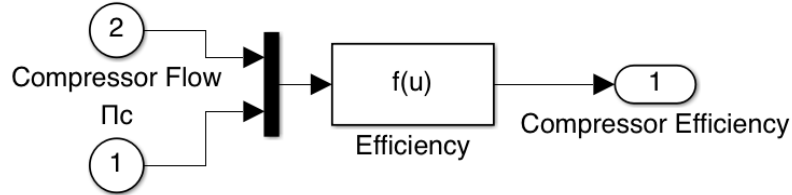


Figure 4.33: Simulink implemented model for Compressor Efficiency

The Simulink implemented model that describes the model for the complete Compressor is shown in Fig.4.34.

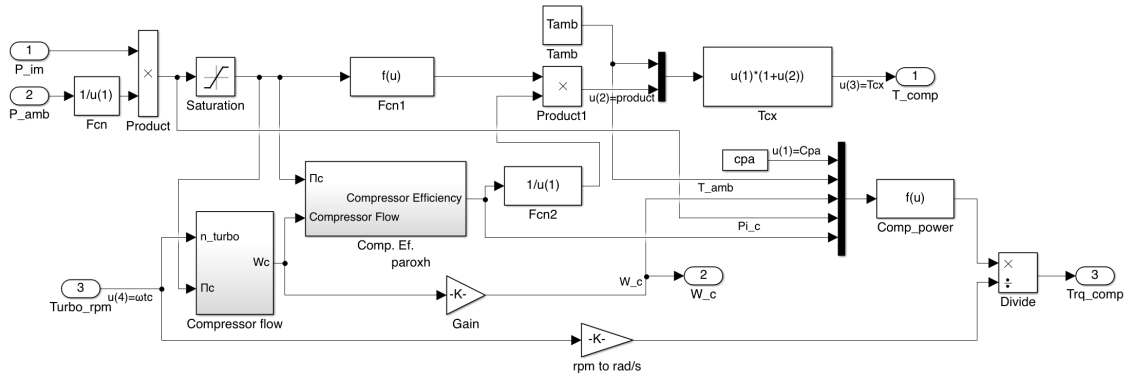


Figure 4.34: Simulink implemented model for the Compressor Model

4.4.3 Turbine Modelling

The turbocharger turbine is driven by the energy available in the engine exhaust. It consists of the blowdown work transfer produced by expanding the gas in the cylinder at exhaust valve opening to atmospheric pressure and for the four-stroke cycle engine) the work done by the piston displacing the gases remaining in the cylinder after blowdown. [14]

Reciprocating engines are generally unsteady pulsating flow devices meaning that the exhaust gas is in the form of high velocity pulses and not a steady flow. There are two different types of turbocharging configurations. The steady flow turbocharger and the pulsating flow turbocharger, those are in accordance to the exhaust gas receiver configuration. Steady flow turbines tend to have higher efficiencies but they do not take full advantage of the high kinetic energy of the exhaust gas. Mid size engines such as the one in question use pulsating flow turbochargers.

The Turbine model is similar to the compressor model in means that it is also consisted of a flow model and an efficiency model. Similarly to the compressor, the turbine part of the turbocharger is also described by the turbine map. The Turbine map of the GT3782TM turbocharger is shown in Fig.4.35.

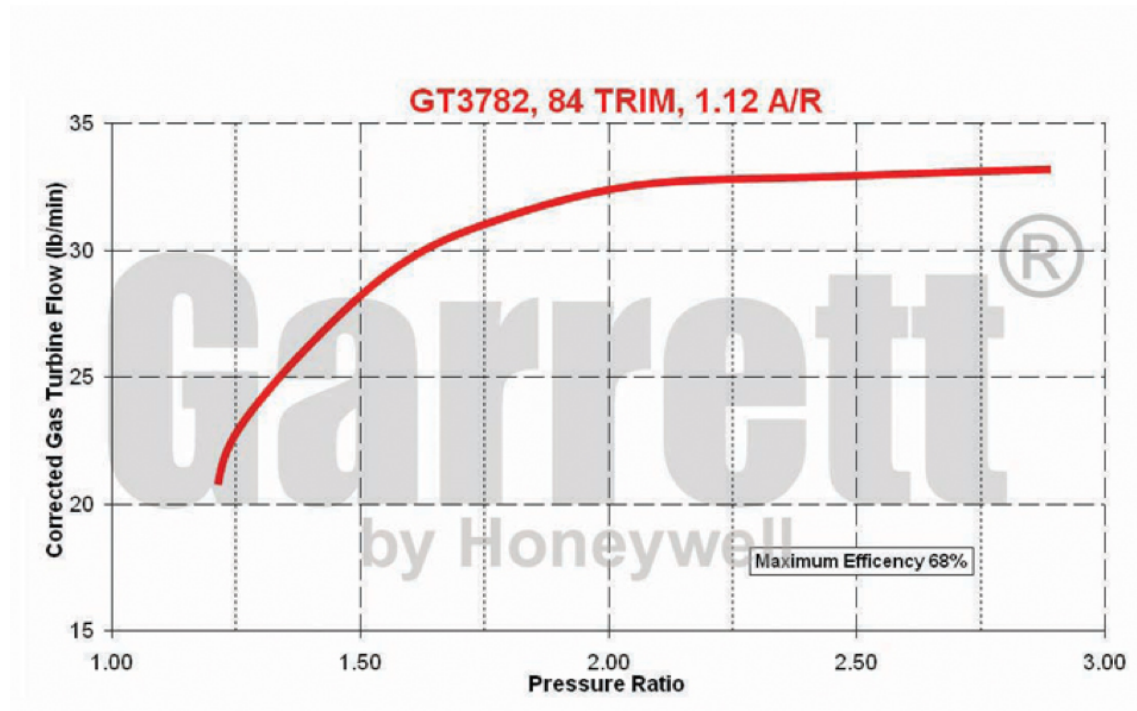


Figure 4.35: Garrett® GT3782TM Turbine Performance Map, [2]

The Turbine Performance Map shows the Gas Flow through the turbine W_t for a given Pressure Ratio Π_t . The maximum isentropic efficiency is also presented.

4.4.3.1 Modelling the Turbine Flow Rate

Modelling the turbine flow is way simpler than modelling the compressor flow (as described above), basically due to the simplicity of the Turbine Map itself. The map is imported in Matlab in the form of data points that are acquired the same way as the compressor map (using the Matlab addon GUI "Grabit"). These data points are then fitted with a power function which is adequate due to the simplicity of the curve itself. Therefore the turbine mass flow Q_t is calculated:

$$Q_t = a_t \Pi_t^{b_t} + c_t \quad (4.4.12)$$

The fitted model for the turbine mass flow is seen in Fig.4.36. **Model I/O**

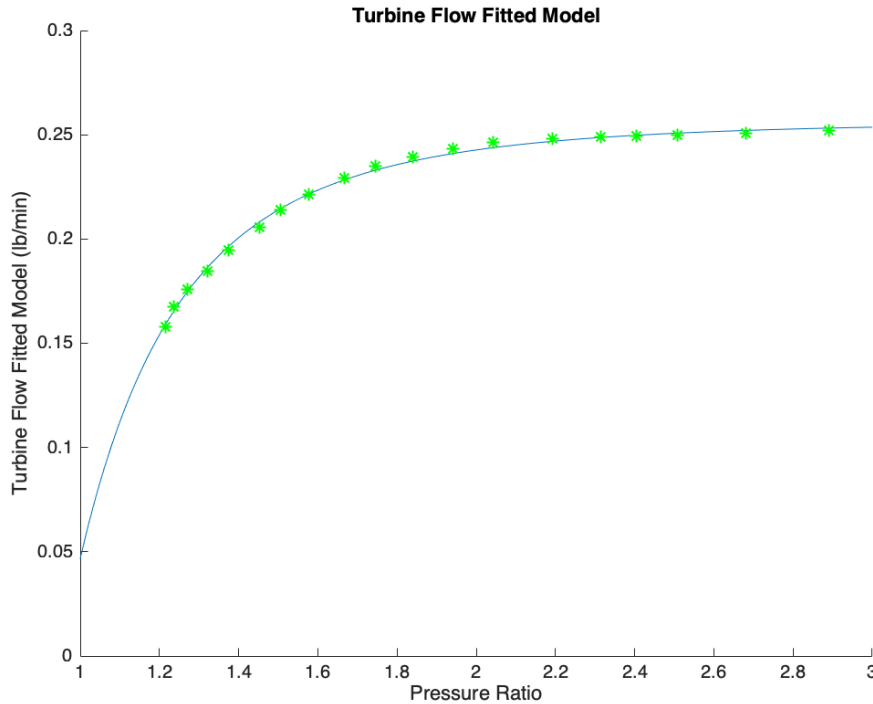


Figure 4.36: Turbine Flow Map (Fitted)

Diagram and Simulink Implementation

The I/O diagram that describes the model proposed above that calculates the $W_{turbine}$, Turbine Flow is shown in Fig.4.37.



Figure 4.37: I/O Diagram of Turbine Mass Flow

The Simulink implemented model that describes the model proposed above that calculates the $W_{turbine}$, Turbine Flow is shown in Fig.4.38.

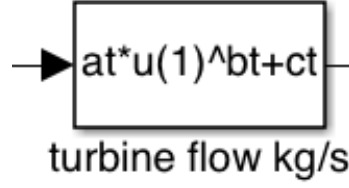


Figure 4.38: Simulink implemented model for Turbine Mass Flow

4.4.3.2 Modelling Turbine Efficiency

The tricky part in modelling the turbine's efficiency is that the Turbine flow map does not project the turbine isentropic efficiency, it only shows the maximum (on maximum efficiency operation) turbine efficiency ratio. The torque and turbine-out temperature can only be calculated knowing the isentropic efficiency. The isentropic efficiency of the turbine is mainly affected by the angle of incidence of the inflowing gas [16]. Instead of the turbocharger speed, the “turbine blade speed” ratio is used as the main independent variable for calculations:

$$\tilde{c}_u = \frac{r_t \omega_t}{c_{us}} \quad (4.4.13)$$

Where

$$c_{us} = \sqrt{2c_p T_3 [1 - \Pi_t^{\frac{\gamma-1}{\gamma}}]} \quad (4.4.14)$$

and r_t being the effective turbine radius.

A turbine efficiency map is drawn as a parabola. A usual Turbine efficiency map is shown below in Fig.4.39. In order to calculate the isentropic efficiency and adjust

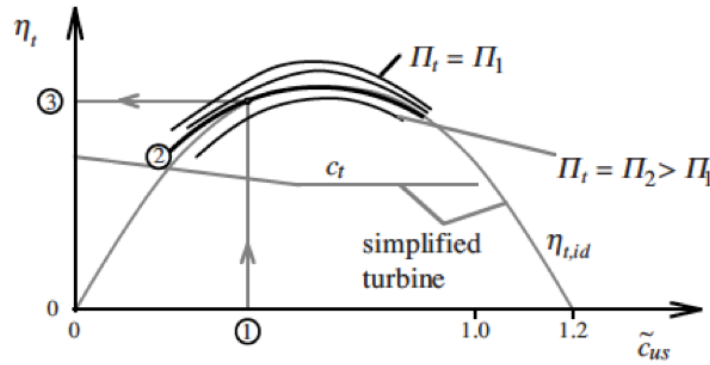


Figure 4.39: Simplified Turbine Efficiency map

the map to the actual turbine, $\eta_{t,max}$ and $C_{u,opt}$ have to be known. The turbine max efficiency $\eta_{t,max}$ is given on the manufacturer's flow map and for automotive applications it usually ranges from 0.65-0.75. The second parameter, $C_{u,opt}$ which in general describes the ideal value of c_{us} (where the parabola hits its peak) ranges from 0.55-0.65 for fast rotating ($> 150000rpm$) automotive applications and is lower for mid and slower rotating turbocharger systems. In this instance, the maximum efficiency was known but the second parameter was identified through least squares

optimization (parameter identification) on the whole model (more on the relevant chapter). Summarizing, the turbine isentropic efficiency is calculated with:

$$\eta_t(\tilde{c}_u) = \eta_{tmax} [2 \frac{\tilde{c}_u}{\tilde{c}_{uopt}} - (\frac{\tilde{c}_u}{\tilde{c}_{uopt}})^2] \quad (4.4.15)$$

Having the turbine efficiency modelled, the Turbine created torque can be easily calculated:

$$M_t = \frac{P_t}{\omega_t} = P_{ts} \eta_t = \frac{W_t c_{pe} T_3 \eta_t}{\omega_t} [1 - \Pi_t^{\frac{1-\gamma}{\gamma}}] \quad (4.4.16)$$

Model I/O Diagram and Simulink Implementation

The I/O diagram that describes the model proposed above that calculates the $\eta_{turbine}$, Turbine Flow is shown in Fig.4.40.

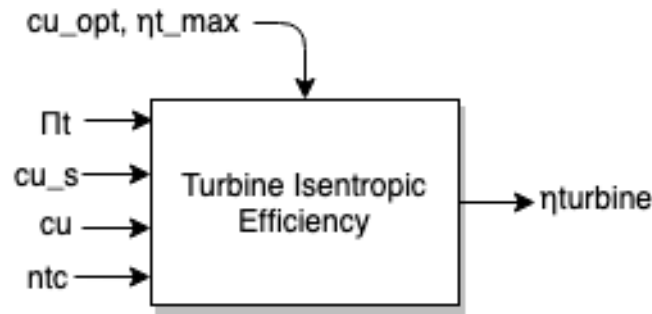


Figure 4.40: I/O Diagram of Turbine Efficiency

The Simulink implemented model that describes the model proposed above that calculates the $\eta_{turbine}$, Turbine Flow is shown in Fig.4.41.

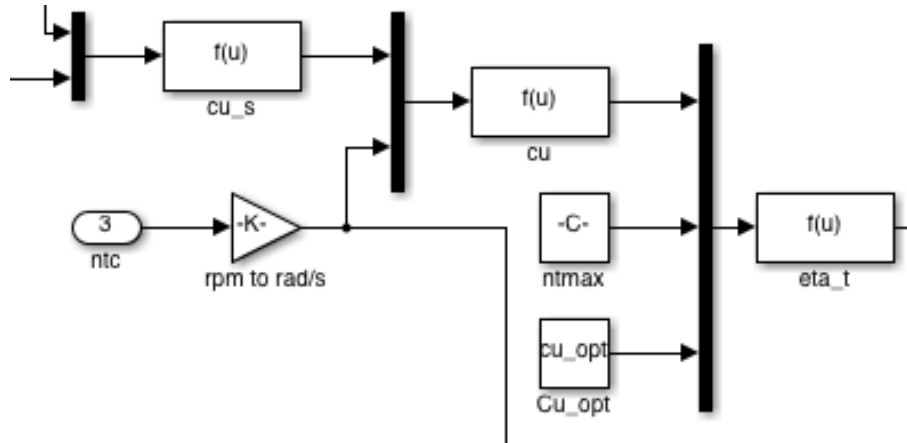


Figure 4.41: Simulink implemented model for Turbine Efficiency

4.5 Mechanical System-Torque Generation

The sole purpose of having an engine is to produce mechanical power i.e. Torque. The engine speed is a level variable, meaning that it cannot be set arbitrarily. The torque however being dependent on many cylinder mixture factors i.e. fuel rate, air rate etc. can be changed arbitrarily.

$$T_e = f(W_{ei}, \lambda, \omega_e, x_{egr}, \dots) \quad (4.5.1)$$

In order to accurately predict an engine's torque, detailed thermodynamic simulations are necessary. For the control purposes of a mean value model though, these simulations would take a lot of time to complete. Therefore, other ways to accurately predict the engine torque are investigated. Another possible way to predict the engine torque would be to create a lookup table. That means running the engine in all possible scenarios, measuring the torque and storing the data in the lookup table. This solution is also not applicable as it is also very time consuming. The way chosen to calculate the engine torque is the one that separates the factors that consist the torque and calculating each one separately. The added result should describe the final torque which is the actual one. The torque M_e produced by the engine is modelled using three different engine components, the gross indicated torque M_{ig} , the pumping torque M_p and the friction torque M_{fric} [14].

$$M_e = M_{ig} - M_p - M_{fric} \quad (4.5.2)$$

Indicated Torque:

Data concerning the gas inside the cylinder over the operating cycle of the engine can be used to calculate the work transfer from the gas to the piston. If the cylinder pressure and corresponding cylinder swept volume throughout the engine cycle are plotted on a p-V diagram, we will get the actual diesel 4-stroke cycle:

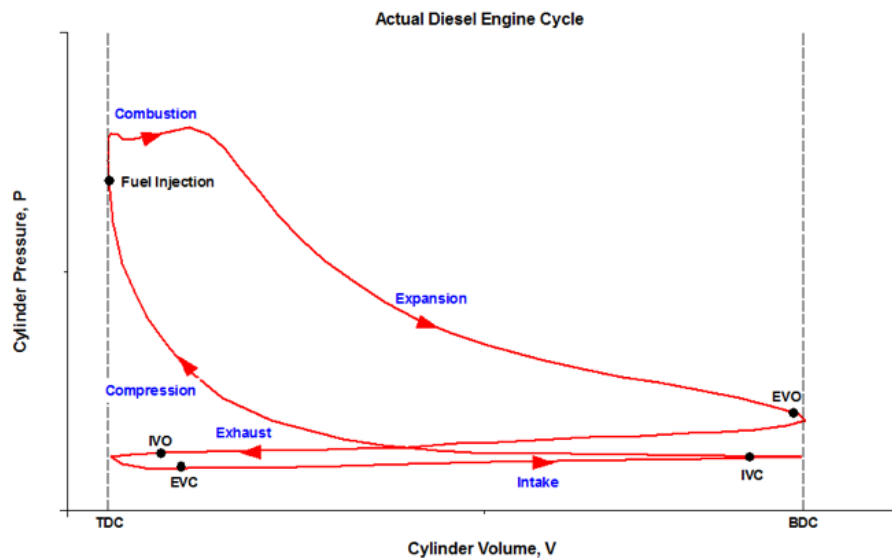


Figure 4.42: 4-stroke Diesel thermodynamic cycle

The indicated work per cycle, $W_{c,i}$ is obtained by integrating around the curve of the thermodynamic cycle the engine is operating.

$$W_{c,i} = \oint p dV \quad (4.5.3)$$

Therefore, the gross indicated work per cycle is the work that is delivered to the piston over the compression and expansion strokes only. The same applies to the gross indicated torque meaning that it is the ideal quantity, it represents the sum of the useful work/ torque that is applied to the shaft in order to overcome all the engine's losses. The actual work/ torque (net indicated work/ torque) has to factor the entirety of the four-stroke cycle, meaning that it has to be subtracted of all the engine's losses. M_{ig} is modelled as seen below:

$$M_{ig} = \frac{u_\delta 10^{-6} n_{cyl} q H V \eta_{ig}}{4\pi} \quad (4.5.4)$$

where $u_\delta = \frac{120 W_f}{10^{-6} n_e n_{cyl}}$. Assuming that the engine is always running at the optimal injection timing (crankshaft angle), the gross indicated efficiency is modelled as:

$$\eta_{ig} = \eta_{igsch} \left(1 - \frac{1}{r_c^{\gamma_{cyl}-1}} \right) \quad (4.5.5)$$

Where r_c is the compression ratio of the engine.

Pumping losses/Torque:

While the internal area of the upper enclosed diagram (without the horizontal bottom part) represents the gross indicated work, the area of the horizontal diagram represents the pumping work. Easily explained, if the intake gas pressure is higher than the exhaust gas pressure, the pumping work transfer will be from the gas to cylinder (meaning that the pumping torque would be positive!), in other cases that the inverse is true, the pumping work transfer is from the cylinder to the gas (meaning that the pumping work/ Torque is negative). That is what the pumping torque loss describes and that's why it factors the intake pressure p_{im} as well as the exhaust pressure p_{em} .

$$M_p = \frac{V_d}{4\pi} (p_{em} - p_{im}) \quad (4.5.6)$$

Friction losses/Torque:

We have seen the part of the gross indicated torque that is needed to expel the exhaust gases and draw in the fresh air charge. The other part of the gross torque that needs to be subtracted from the net torque is the torque that is needed to overcome basically the friction of the bearings, shaft, pistons and the power appliances (water pump, fuel pump etc.).

It is hard to determine the friction torque accurately, namely, a working approach to determine it is to tie the engine to a dynamometer and measure the torque needed to spin it (without firing the engine). The engine speed, throttle setting, oil and water temperatures, and ambient conditions are kept the same in the motored test

as under firing conditions. The possible resulting inaccuracy to this test lies to the fact that the air coming in and out of the engine has difference in pressure with the air coming in and out on a firing engine.

Therefore, being the more vague part of this equation M_{fric} will have to be fitted to the existing data and therefore help in the fitment of the whole torque model.

Friction torque is modelled as seen below:

$$M_f = \frac{V_d}{4\pi} 10^5 (c_{fric1} n_{e, ratio}^2 + c_{fric2} n_{e, ratio} + c_{fric3}) \quad (4.5.7)$$

Where $n_{e, ratio} = \frac{n_e}{1000}$.

Therefore in order to accurately model the engine torque, the tuning parameters that need to be set are η_{igsch} , c_{fric1} , c_{fric2} , c_{fric3} . The tuning parameters are determined by solving a linear least squares problem that minimizes $(M_e + m_p - M_{e, meas} - m_{p, meas})^2$ with the above tuning parameters as optimization variables. The first two factors are replaced by solving for them from 4.5.1. The measured parameters are replaced by $M_{e, meas} + \frac{V_d}{4\pi} (p_{em} - p_{im})$.

Model I/O Diagram and Simulink Implementation

The I/O diagram that describes the model proposed above that calculates the M_e , Engine Torque is shown in Fig.4.43.

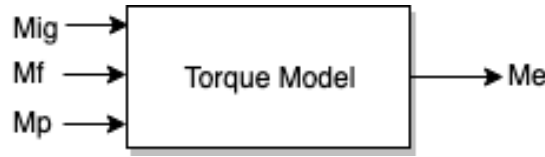


Figure 4.43: I/O Diagram of Engine Torque

The Simulink implemented model that describes the model proposed above that calculates the M_e , Engine Torque is shown in Fig.4.44.

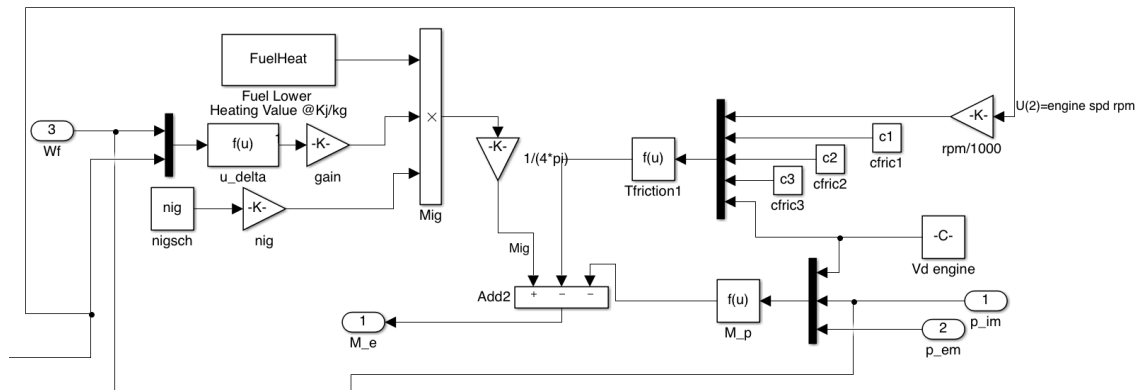


Figure 4.44: Simulink implemented model for Engine Torque

- $q_{in} = \frac{W_f q_{H.V.}}{W_{ei} + W_f} (1 - x_r)$, the cylinder temperature when the inlet valve closes after the intake stroke and mixing.
- $x_r = \frac{\Pi_e^{\frac{1}{\gamma_a}} x_p^{\frac{1}{\gamma_a}}}{r_c x_v}$, the residual gas fraction.
- $x_v = \frac{v_3}{v_2} = 1 + \frac{q_{in}(1-x_{cv})}{c_{pa}[(q_{in} \frac{x_{cv}}{c_{va}}) + T_1 r_c^{\gamma_a - 1}]}$, the volume ratio between point 3 and point 2 (after and before combustion) in the Seiliger cycle. Obviously this factor is of significant quantity for a Diesel Engine.

Model I/O Diagram

The I/O diagram that describes the model proposed above that calculates the T_e , Exhaust Gas Temperature is shown in Fig.4.46.

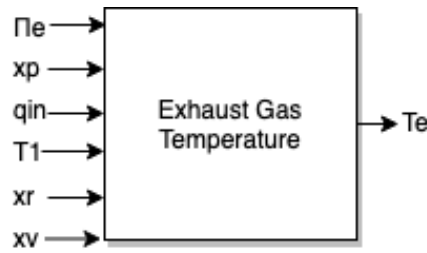


Figure 4.46: I/O Diagram of Exhaust Gas Temperature

4.7 Heat Losses in the exhaust manifold pipes

The model above only describes the cylinder- temperature, meaning the temperature of the burnt gas exiting the engine. In order to describe the exhaust manifold temperature, accounting for the heat losses that happen between the walls of the exhaust manifold pipes and the surrounding fresh air outside of them is essential. Therefore, a sub-model that accounts for these losses was added.

The temperature drop sub-model is built following the rules of Heat convection, originally developed in [18]. Also, the sub-model is similar to its structure with the intercooler model, meaning that it is also flow dependent. The equation is seen below:

$$T_{em} = T_{amb} + (T_e - T_{amb})e^{-\frac{h_{tot}\pi d_{pipe}l_{pipe}n_{pipe}}{w_{eo}c_{pe}}} \quad (4.7.1)$$

Where,

- h_{tot} , the heat convection coefficient.
- d_{pipe} , the pipe diameter.
- l_{pipe} , the pipe length.
- n_{pipe} , the pipe quantity.

The physical values of the exhaust manifold were measured from the engine itself in the lab using a measuring tape and a caliper. Tuning and computing the heat convection coefficient h_{tot} was accomplished by using least-squares optimization on the complete model and using dynamic data from the operation of the engine in the lab.

Model I/O Diagram and Simulink Implementation

The I/O diagram that describes the model proposed above that calculates the T_{em} , Exhaust Manifold Temperature is shown in Fig.4.47.

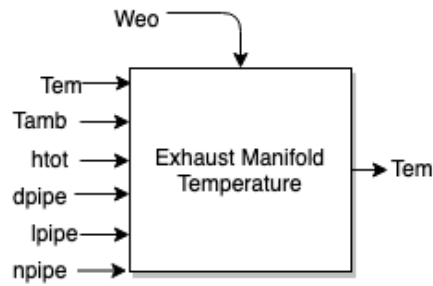


Figure 4.47: I/O Diagram of Exhaust Manifold Temperature

The Simulink implemented model that describes the model proposed above that calculates the T_{em} , Exhaust Manifold Temperature is shown in Fig.4.48.

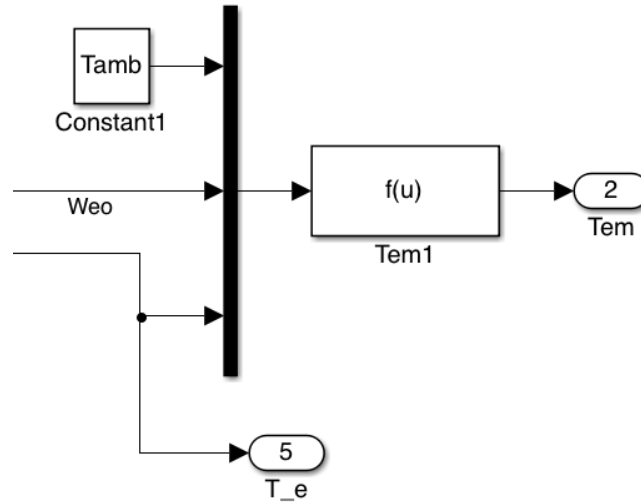


Figure 4.48: Simulink implemented model for Exhaust Manifold Temperature

Chapter 5

Tuning and Validation

One step in the development of a model that describes the system dynamics and non-linear effects of the engine is tuning and validation. First, all the parameters of the separate submodels are initialized using the models described in Chapter 2. Then the unknown parameters, that mainly affect the dynamic submodels of the complete engine model, are initialized and tuned using a cost function that minimizes the errors of the dynamic measurements and the dynamic models. Afterwards, the static parameters that were beforehand initialized, are automatically fine tuned using a separate cost function that minimizes both the static and the dynamic errors between the measurements in the Data Sets and the modelled values from the complete engine model. Having all the parameters, both dynamic and stationary, tuned, the model accuracy compared to the dynamic tuning data is evaluated and the corresponding errors for each separate Data Set are calculated. Additionally, the importance of the tuning process specifically is explained. Finally, the whole model is validated using the dynamic data that were acquired during the beginning of this project. It is important that the resulting model captures the dynamic responses and behaviors of the actual engine and additionally the right stationary operating points of each output.

It is important to show the error on the tuning of each submodel and to keep said error low. Relative errors are calculated and used to evaluate the tuning and validation of the model. Relative errors for stationary measurements between a measured variable $y_{meas,stat}$ and a modelled variable $y_{mod,stat}$ are calculated as:

$$Stationery_{relativeerror}(i) = \frac{(y_{meas,stat}(i) - y_{mod,stat}(i))}{\frac{1}{N} \sum_{i=1}^N y_{meas,stat}(i)} \quad (5.0.1)$$

Where i is an operating point. N is the number of operating points.

Relative errors for dynamic measurements between a measured variable $y_{meas,dyn}$ and a modelled variable $y_{mod,dyn}$ are calculated as:

$$Dynamic_{relativeerror}(i) = \frac{(y_{meas,dyn}(i) - y_{mod,dyn}(i))}{\frac{1}{N} \sum_{i=1}^N y_{meas,dyn}(i)} \quad (5.0.2)$$

5.1 Initialization of the independent submodel parameters

In this section, the independent submodel parameters are initialized using least squares optimization algorithms and Data acquired from the Lab setup.

5.1.1 Volumetric Efficiency Model

The volumetric efficiency is extremely crucial on the accuracy of the complete model as it affects the intake manifold pressure, the exhaust manifold pressure, the lambda estimation etc. Relatively small errors in its estimation result in great errors in the whole system. The tuning parameters are initialized by solving the least squares problem that minimizes $(W_{ei} - W_{ei,meas})^2$.

The initialization procedure makes use of equations 4.2.16 to 4.2.22. For the initialization of the parameters $c_{vol1}, c_{vol2}, c_{vol3}$, that model the volumetric efficiency, the majority of the Data (3 of the 5 Data Sets) were used. The data were used without any filtering. The input data that are needed to calculate the volumetric efficiency are:

- Fuel Rate $W_f = \frac{kg}{s}$.
- The DPF mass flow, $W_{DPF} = \frac{kg}{s}$ so that the W_{ei} can be computed.
- The EGR mass flow, $W_{EGR} = \frac{kg}{s}$ also for the W_{ei} computation.
- Engine Speed, $n_e = rpm$.
- Intake Manifold Pressure, $p_{im} = Pa$.
- Intake Manifold Temperature, $T_{im}=K$.

The 3 Data sets differ from each other substantially on the operation range. The measured data of importance were all added to corresponding matrices in order to be optimized with least squares algorithms. The purpose of this is to initialize the volumetric efficiency parameters on a wide spectrum of engine operation, so the model is accurate in it.

After the initialization process, the error between the measured and the modelled volumetric efficiencies will be evaluated and discussed. The results are very close to the actual values giving an accurate estimation of the volumetric efficiency of the Engine. It is very important to remind the reader of the great influence the EGR mass flow has on the volumetric efficiency.

Input Data: The data that were used for the initialization of the volumetric efficiency parameters, are shown below in Fig.5.1. The fitting process is shown below in Fig.5.2

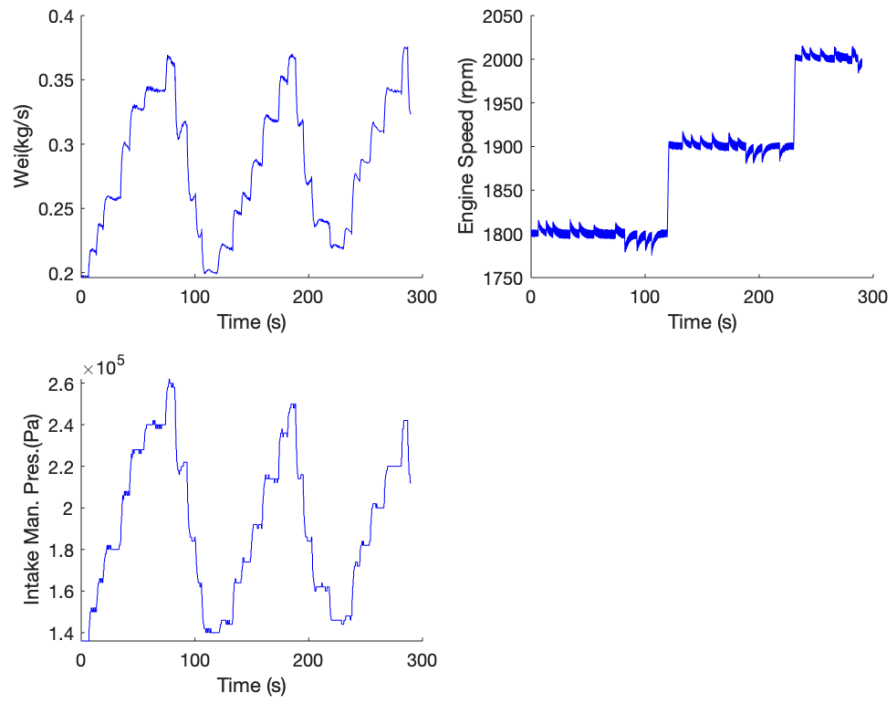


Figure 5.1: Input Data from 3 Data Sets for the Initialization of the vol. ef. Model parameters

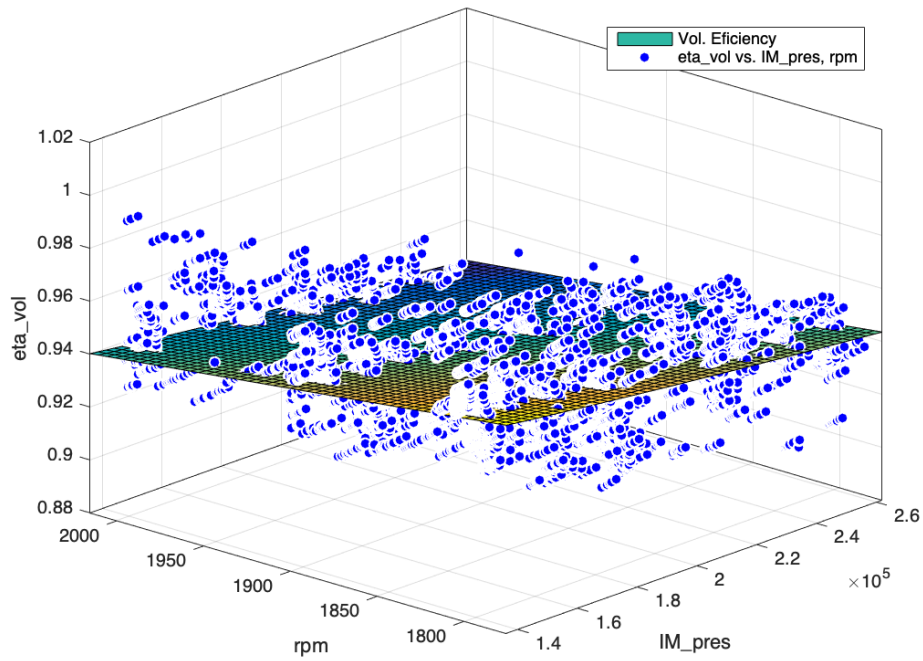


Figure 5.2: Volumetric Efficiency Fitting procedure

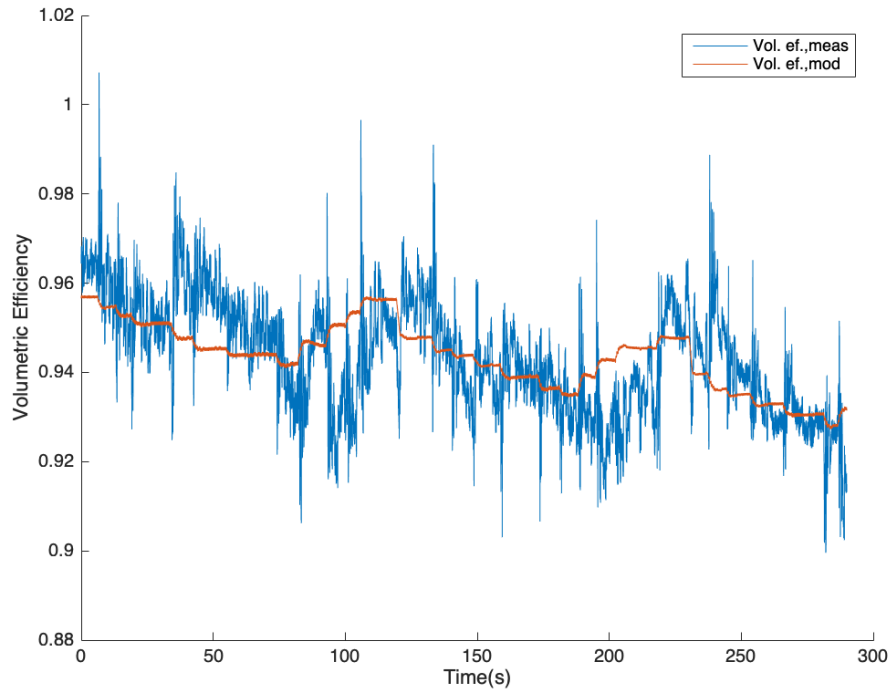
Results of Initialization process:

Figure 5.3: Volumetric Efficiency Measurements vs. Model

Table 5.1: Results of fitting for vol_1, vol_2, vol_3

Tuning Parameter	Result
vol_1	-0.0001077
vol_2	-0.006772
vol_3	1.284

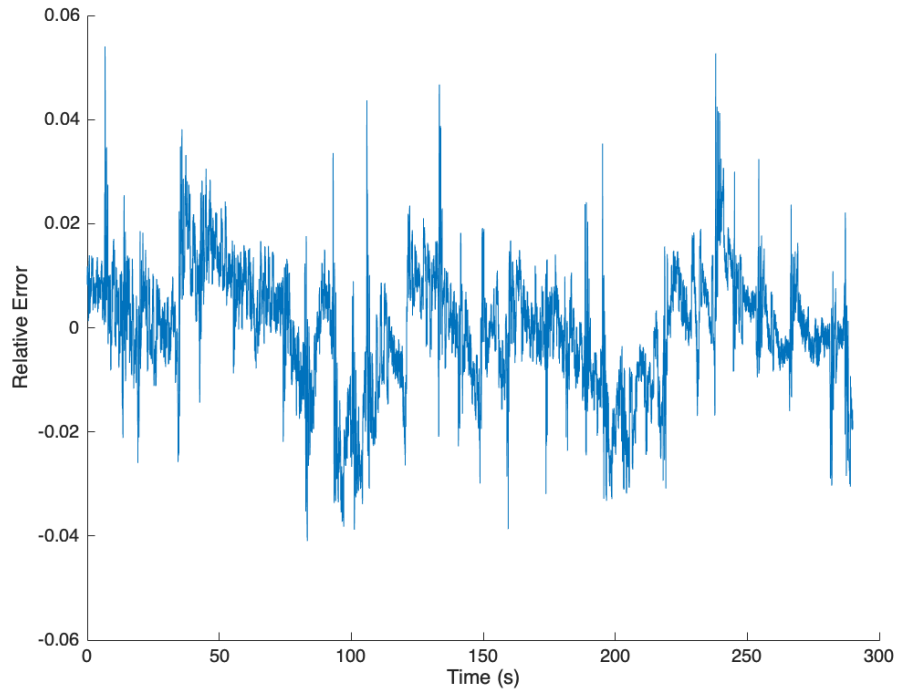
Errors in Model vs. Measurements:

Figure 5.4: Relative Error between Model and Measurements

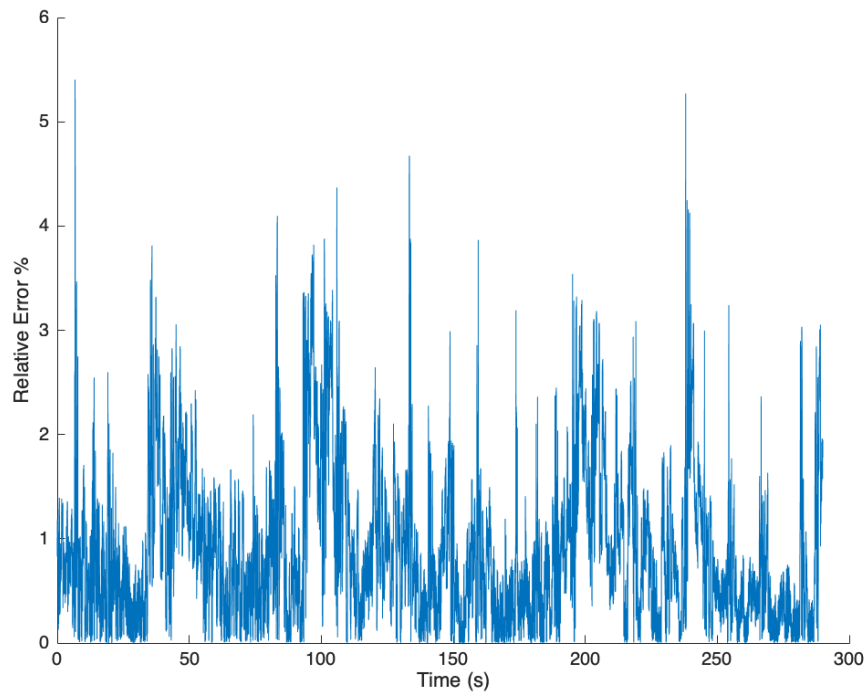


Figure 5.5: Relative Error between Model and Measurements (%)

Discussion: The fitment of the volumetric efficiency model parameters (vol_1, vol_2, vol_3) is successful achieving a mean relative error percentage of 0.84%. The instability seen in the measured data are due to sensor errors due to the very high frequency of sampling in the Lab (1000Hz). If the input data were to be filtered (especially the engine speed) this phenomena would not be present. However, this does not pose problems to the actual fitting, since the model captures the mean value of the volumetric efficiency very well.

Moving on to the results, from a physical sense point of view, it is shown that the volumetric efficiency tends to drop for higher engine speeds and higher p_{im} , intake manifold pressures. That is true, since as the speeds climb higher and higher, it is harder for the exhaust gases to escape and for the gas trading with clean air to happen efficiently. However, it should be noted that with higher engine speeds, the overall flows both in and out of the engine increase, overcoming this marginal drop in volumetric efficiency.

5.1.2 Torque Generation Model

The tuning parameters for the Torque Generation submodel are c_{fric1} , c_{fric2} , c_{fric3} and n_{igsch} . The first three describe the friction Torque losses and the fourth parameter is needed to compute the indicated torque M_{ig} .

To model and tune the torque generation submodel, equations 4.5.1 through 4.5.7 were used. Similarly to the volumetric efficiency tuning, dynamic data were used as input to the tuning program. Unlike the volumetric efficiency tuning however, only a single Data Set was used as the resulting parameter tuning was deemed very accurate for all circumstances. Meaning that the model can accurately describe and compute the Torque of the engine no matter the operation circumstances set on it.

The data that is needed to tune the torque model and estimate the needed parameters are described below:

- Exhaust Manifold Pressure $p_{em} = Pa$. Due to lack of this measurement (because the engine doesn't have a sensor for it) the EGR intake pressure sensor was used. Inspecting Fig.4.9, it is apparent that between the actual exhaust manifold and the EGR intake point there is a certain piping distance. To overcome this, the pressure Data that was acquired in the EGR intake sensor were increased by 10%, which was found to be the ideal increase number through testing.
- Intake Manifold Pressure $p_{im} = Pa$.
- Engine Speed $n_e = rpm$, needed in the form of $n_{e,ratio} = \frac{n_e}{1000}$.
- Fuel Rate $W_f = \frac{kg}{s}$, needed in the form of $u_\delta = \frac{120W_f}{10^{-6}n_en_{cyl}}$.
- The Actual Measured Engine Torque of the engine M_e , needed for tuning.

Due to the state of the Torque model equations, the process of parameter initiation is not straightforward. In order to simplify the process of the parameter tuning, the model was broken down accordingly:

$$M_e = M_{ig} - M_p - M_{fric} \quad (5.1.1)$$

Replacing each factor with it's corresponding value and model, we get:

$$M_e = \frac{u_\delta 10^{-6} n_{cyl} q H V \eta_{ig}}{4\pi} - \frac{V_d}{4\pi} (p_{em} - p_{im}) - \frac{V_d}{4\pi} 10^5 (c_{fric1} n_{e,ratio}^2 + c_{fric2} n_{e,ratio} + c_{fric3}) \quad (5.1.2)$$

This is the equation that computes the final torque of the Engine. The tuning parameters c_{fric1} , c_{fric2} , c_{fric3} and η_{igsch} are not known and need to be specified. They are determined by solving the least squares problem that minimizes $(M_e + M_p - M_{e,meas} - M_{p,meas})^2$ with the tuning parameters as the optimization parameters. Having the actual engine torque M_e at hand from the electric motor that acts as a brake in the engine setup and knowing the intake and exhaust manifold pressures, we can replace the measured factors in the above equation with their corresponding values as $M_{e,meas} + M_{p,meas} = M_{e,data} + V_d \frac{p_{em} - p_{im}}{4\pi}$.

Therefore, 5.1.2 becomes:

$$M_e + M_p = \frac{u_\delta 10^{-6} n_{cyl} q H V \eta_{ig}}{4\pi} - \frac{V_d}{4\pi} 10^5 (c_{fric1} n_{e, ratio}^2 + c_{fric2} n_{e, ratio} + c_{fric3}) \quad (5.1.3)$$

Equation 5.1.3, has many factors to be optimized as is. Therefore, to simplify the tuning process even further, the sum $M_e + M_p$ is set as M_{sim} to MATLAB. Additionally, as M_{ig} is known for the most part (i.e. Fuel rate is known, the number of cylinders is known, the fuel calorific value is known, having a static part and a part that needs optimization) the only part that needs to be tuned, is the unknown factor η_{igsch} :

$$\eta_{ig} = \eta_{igsch} \left(1 - \frac{1}{r_c^{\gamma_{cyl}-1}}\right) \quad (5.1.4)$$

Therefore, the known part of M_{ig} is set as and is calculated as a static entity in the tuning process:

$$M_{ig, no, \eta_{igsch}} = \frac{u_\delta 10^{-6} n_{cyl} q H V}{4\pi} \left(1 - \frac{1}{r_c^{\gamma_{cyl}-1}}\right) \quad (5.1.5)$$

By doing the above it is easier to implement 5.1.3 for tuning. Of course it is crucial that the boundaries of η_{igsch} are specified before the tuning, as it represents the engine's gross indicated efficiency (i.e the efficiency of the ideal thermodynamic cycle this specific Caterpillar C9.3 engine runs on), meaning that it can't be above 1 or below 0 ($0 < \eta_{igsch} < 1$).

The Tuning procedure is presented below in Fig.5.6.

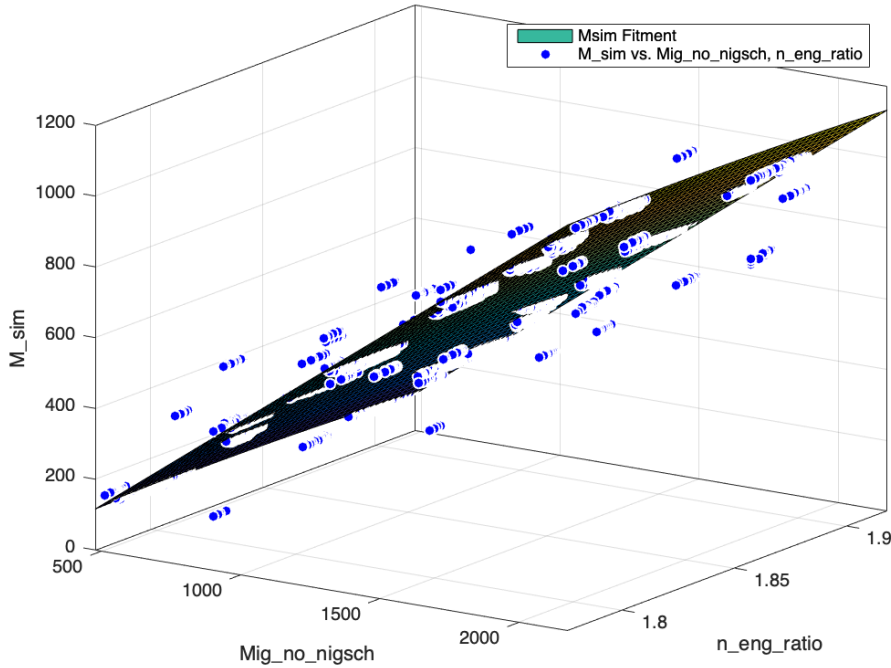


Figure 5.6: Fitting Procedure for the Torque Model

The resulting parameter values are shown in Table 5.2:

Table 5.2: Torque Model Resulting Parameters

Parameter	Value
η_{igsch}	0.6077
C_{fric1}	2.249
C_{fric2}	-6.885
C_{fric3}	7.416

After the completion of the tuning process of the Torque Generation Model and all the parameter initiation, an indicative figure of the model accuracy is shown below:

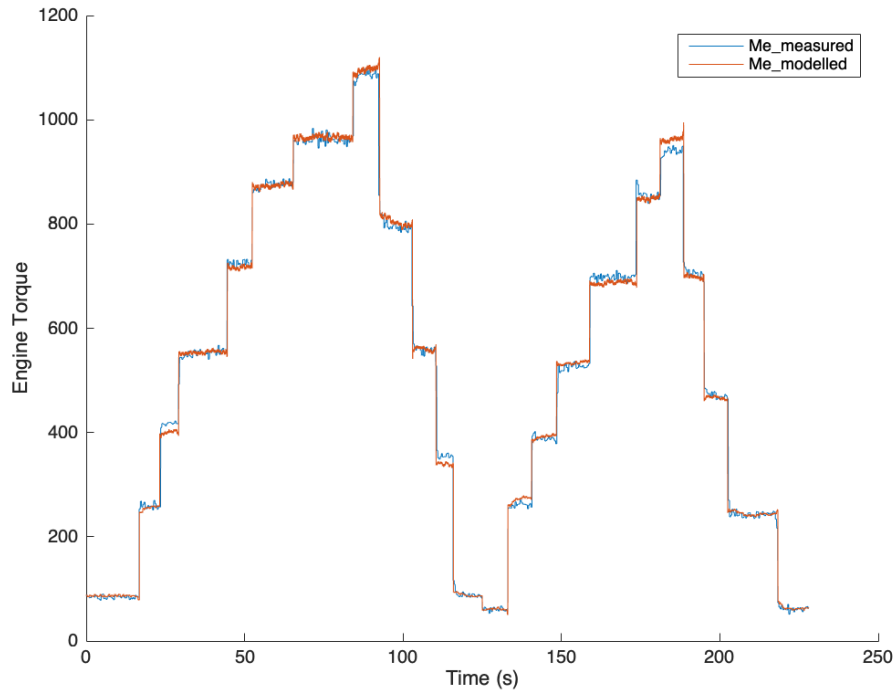


Figure 5.7: Torque Model vs. Measured Data

Error in the Evaluation for the Torque Generation Model

The model fitment is deemed successful and the error diagrams are indicative of this fact. In the error diagrams, in the time sections that transitions happen from a stationary torque measurement to another, "spikes" in the error scale emerge. This is due to the fact that the used Data set is dynamic and due to the fact that the transitions are extremely fast for the modelled torque to follow at the same speed. These "spikes" are not indicative of the goodness of the fit as they only happen on transitions. The model accurately captures the stationary values as seen in Fig.5.7.. Ideally we could use static engine operation data as input but with the way the tuning was done, we get to fit the parameters for multiple operation points. That way we make sure that the fitted parameters will be accurate for a wider spectrum of operation. Even with the presence of the "spikes" (which exist because of the nature of the Data Set itself) the mean errors are kept very low, as seen in Table 5.3.

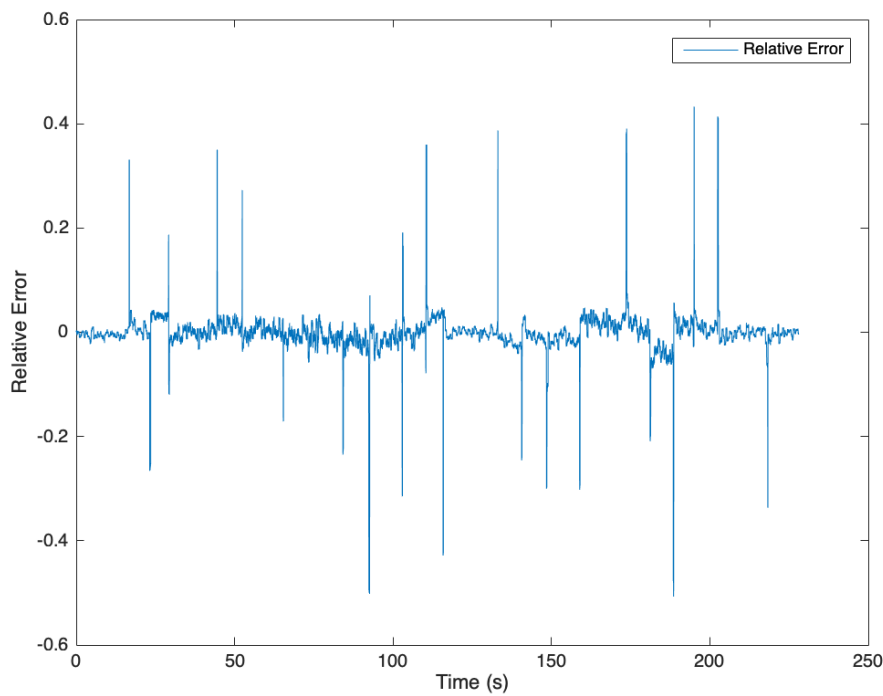


Figure 5.8: Relative Error on the Torque Generation Model

The relative error presented in percentage scale is shown below in Fig.5.9.

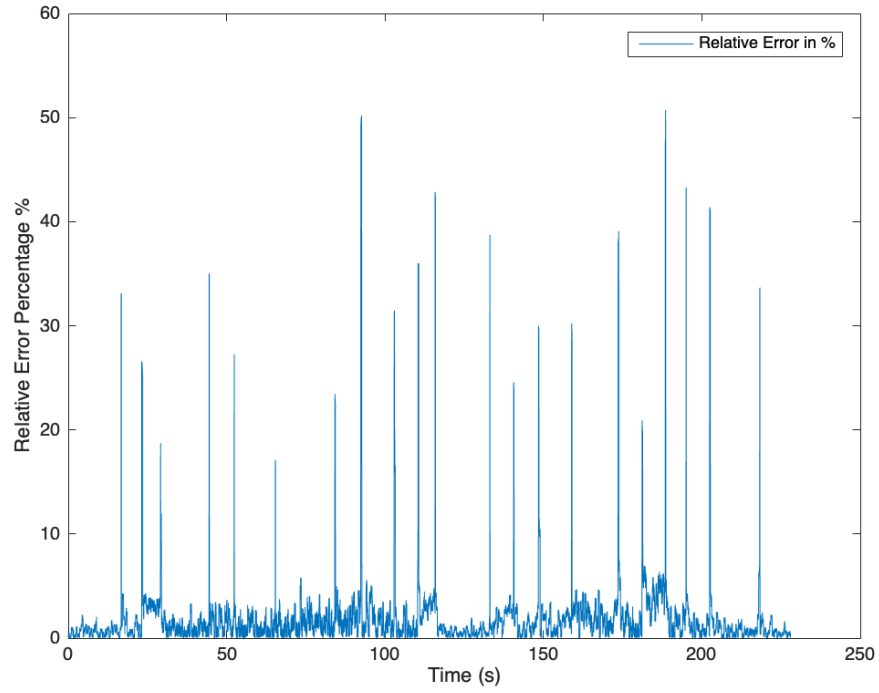


Figure 5.9: Relative Error in Percentage on the Torque Generation Model

Table 5.3: Mean Values of Error on the Torque Generation Model

Type of Error	Mean Value
Relative Error	-0.0016
% of Error	1.8864

5.1.3 Exhaust Manifold Temperature Model

Computing the Exhaust Manifold Temperature is considerably more complex than the other submodels because the equations that describe it are non-linear and depend on each other, therefore they are not separable. The cylinder-out temperature is calculated numerically using a fixed point iteration which starts with the initial values $x_{r,0} = 0$ and $T_{1,0} = 323K$. Then the equations that describe the exhaust gas temperature (which were shown in the respective chapter) are further transformed to:

- $q_{in,k+1} = \frac{W_f q_{HV}}{W_{ei} + W_f} (1 - x_{r,k})$.
- $x_{p,k+1} = \frac{p_3}{p_2} = 1 + \frac{q_{in,k+1} x_{cv}}{c_{va} T_{1,k} r_c^{\gamma_a - 1}}$
- $x_{v,k+1} = \frac{v_3}{v_2} = 1 + \frac{q_{in,k+1} (1 - x_{cv})}{c_{pa} [(q_{in,k+1} \frac{x_{cv}}{c_{va}}) + T_{1,k} r_c^{\gamma_a - 1}]}$
- $x_r = \frac{\Pi_e^{\frac{1}{\gamma_a}} x_{p,k+1}^{\frac{1}{\gamma_a}}}{r_c x_{v,k+1}}$
- $T_{e,k+1} = \eta_{sc} \Pi^{1 - \frac{1}{\gamma_a}} r_c^{1 - \gamma_a} x_{p,k+1}^{\frac{1}{\gamma_a} - 1} [q_{in,k+1} (\frac{1 - x_{cv}}{c_{pa}} + \frac{x_{cv}}{c_{va}}) + T_{1,k+1} r_c^{\gamma_a - 1}]$
- $T_{1,k+1} = x_{r,k+1} T_{e,k+1} + (1 - x_{r,k+1}) T_{im}$

The equations are applied in each iteration k. In each sample during the simulation, the initial values $x_{r,0}$ and $T_{1,0}$ are set to the solutions of x_r and T_1 from the previous sample. Running the simulation for the measured Data showed that it is sufficient to use two iterations in the iterative process. The accuracy of the model is very high for just one iteration but with two iterations the absolute relative error is even smaller as the results reach their final value (i.e three or more iterations yield the same results as two). The error difference between one and two iterations is about 1.2 percent. The error is small as the fixed-point iteration has initial values that are close to the solution.

For the exhaust temperature computation the data that were used were also dynamic. The input data of $\Pi_e, W_f, W_{eo}, T_{im}$ are all shown below:

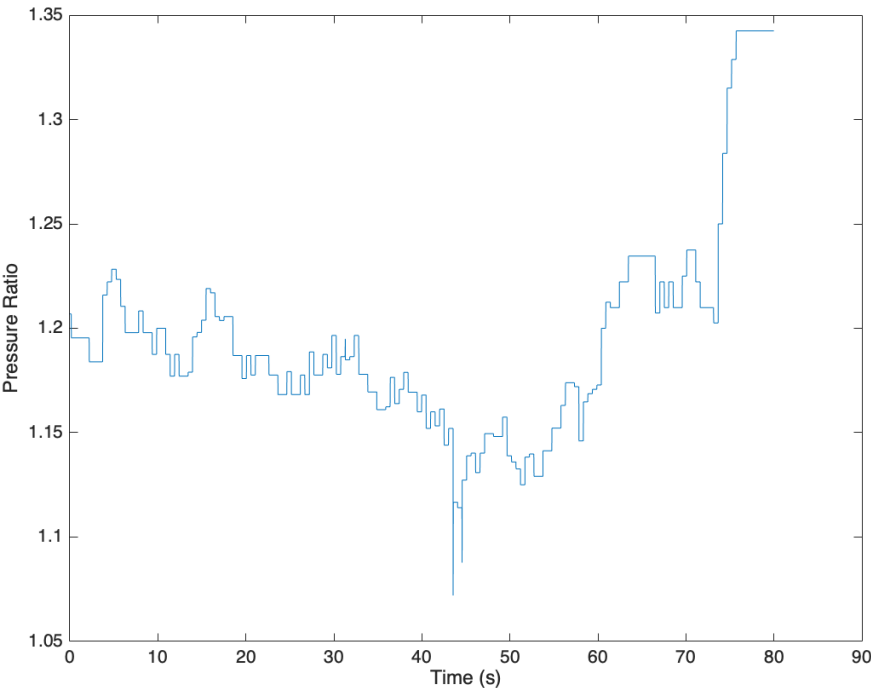


Figure 5.10: Pressure Ratio from Data

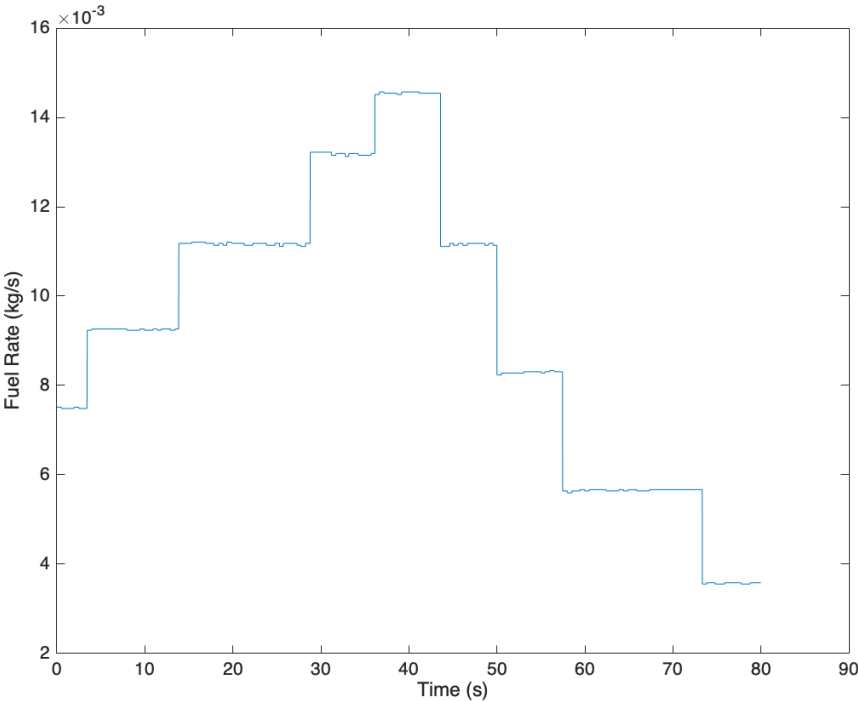


Figure 5.11: Fuel Rate from Data

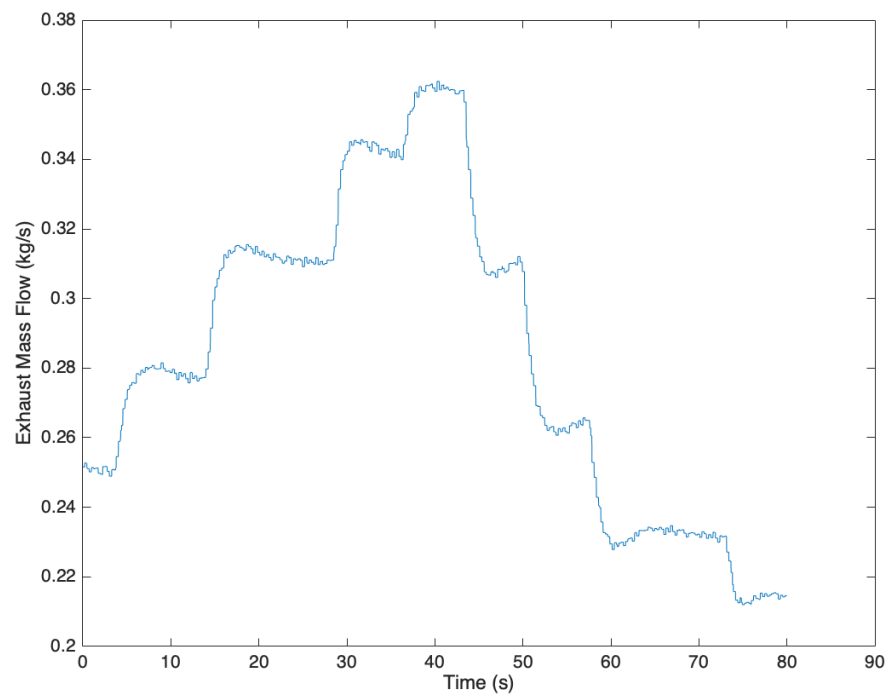


Figure 5.12: Exhaust Mass Flow from Data

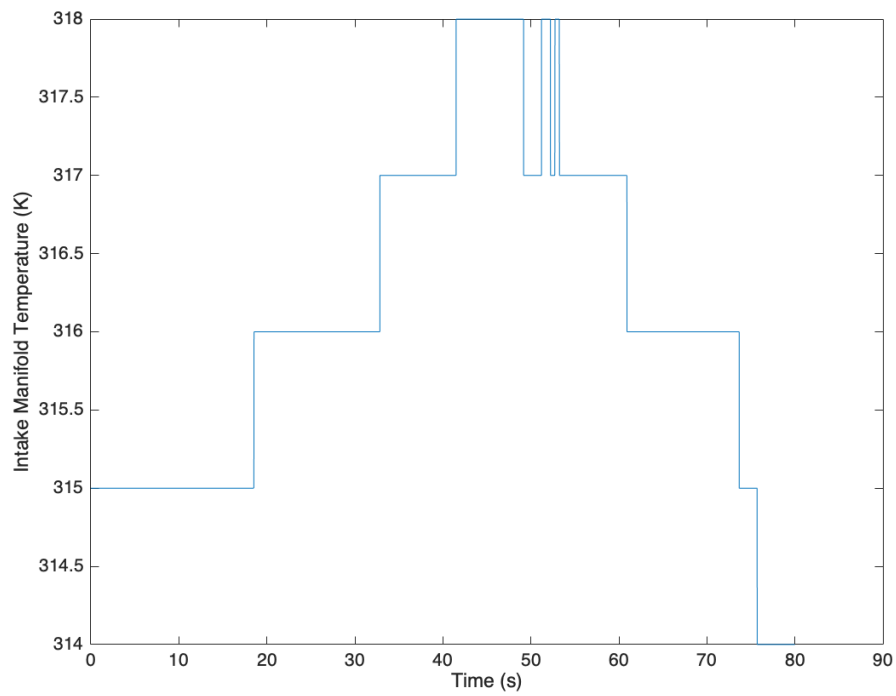


Figure 5.13: Intake Manifold Temperature from Data

The results of the iterative simulation are shown below. For purposes of demonstration the simulation was done for 3 iterations on each fixed point. The difference between one and two iterations is obvious (the error is about 1.2 percent) and the difference between two and three (or more) iterations is zero. The solution of two iterations is deemed to be enough for the computation of the Exhaust Gas temperature.

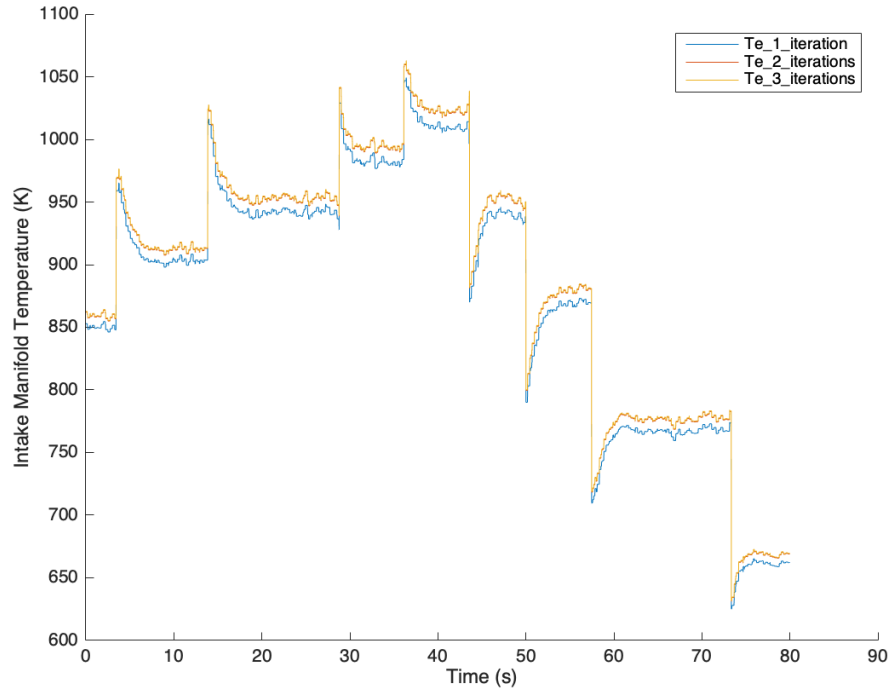


Figure 5.14: Exhaust Gas Temperature for Different Iteration Numbers

The model for computing the cylinder-out temperature while being very accurate and detailed, is quite complex to be applied to the Simulink model of the Engine. It's need for two iterations instead of one deems inapplicable on the Simulink interface. After the cylinder-out temperature was computed (which was not already stored in the acquired data from the engine's operation in the Lab) another model was applied on top of the modelled data in order to simplify the computation of the cylinder-out temperature in Simulink.

In diesel engines according to [1], the exhaust gas temperature is mainly determined by the air-fuel ratio λ of the cylinder mixture, such that for warmed-up conditions of operation (i.e engine oil and engine coolant are up to operation temperatures) the temperature immediately after the exhaust ports can be approximated by:

$$T_e(t) = T_{im}(t) + a_1 \lambda^{a_2}(t) + a_3 \quad (5.1.6)$$

The constants a_1, a_2, a_3 have the following orders of magnitude according to [1]:

- $a_1 \approx 2500$
- $a_2 \approx -2$
- $a_3 \approx 100$

At this point, it should be noted that this model doesn't hold for very rich conditions inside the cylinder ($\lambda > 1.8$). The tuning process is shown below:

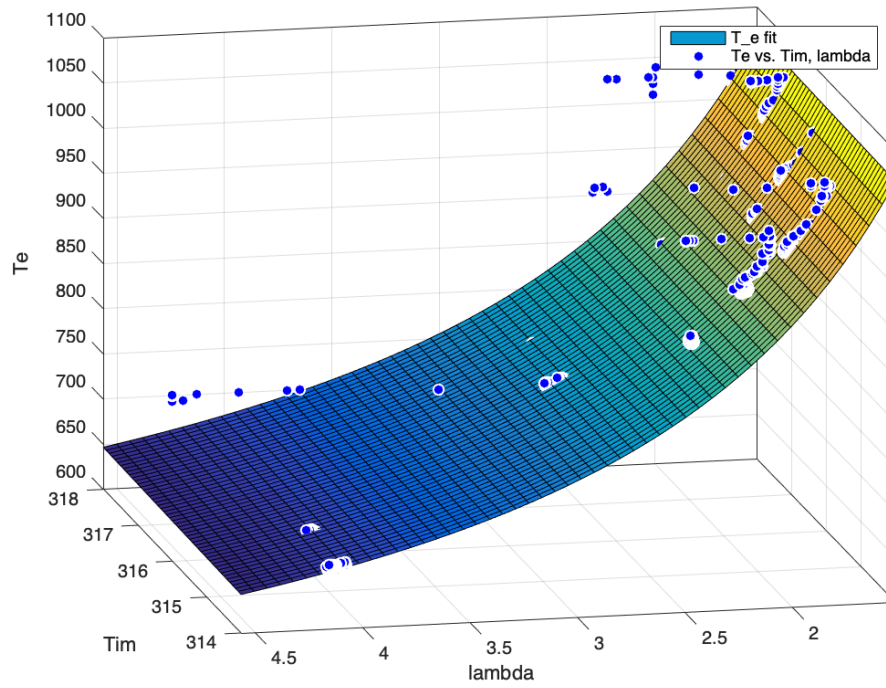


Figure 5.15: Fitting Process for the second Model

Below, the table that summarises the parameters is shown:

Table 5.4: Exhaust Gas Temperature Second Model Parameters

Parameter	Value
a_1	2044
a_2	-2.5272
a_3	-138

The result model is shown together with the computed cylinder-out temperature in Fig.5.16 that was computed with the complex model analyzed above. The differences are minimal but the gains in the whole model's computation speeds and simplicity are considerable.

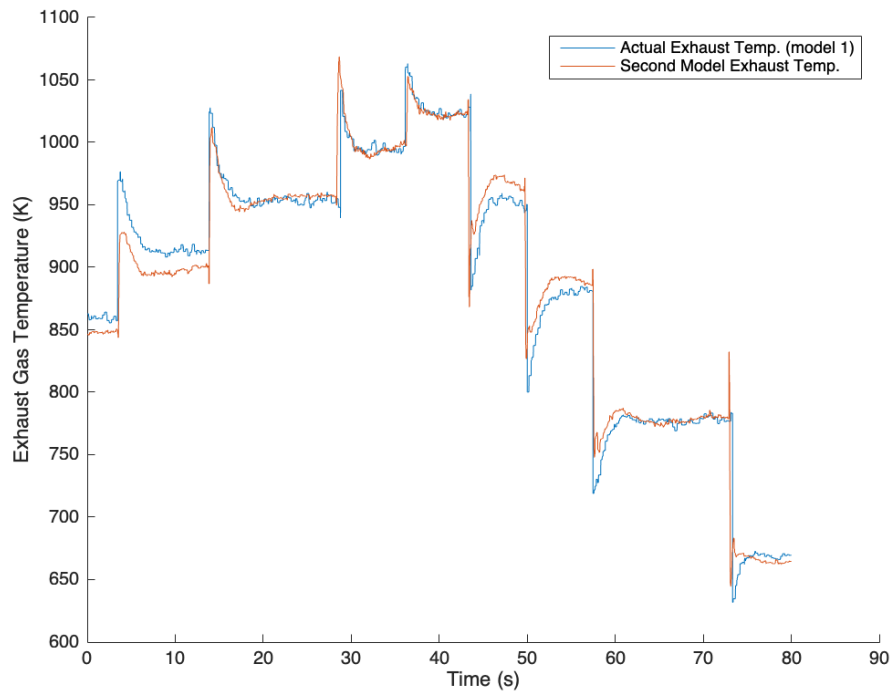


Figure 5.16: Exhaust Temperature for the two different Models

5.1.3.1 Error Evaluation for the Exhaust Gas Temperature Model

The errors that the second model presents have to be evaluated before it is deemed applicable for the engine model. The errors are indeed low and the second model is therefore the optimal choice for calculating the Exhaust Gas Temperature. Both the relative error and the relative error in percentage will be presented below in Fig.5.17,5.18. Afterwards, the mean values will be presented as usual.

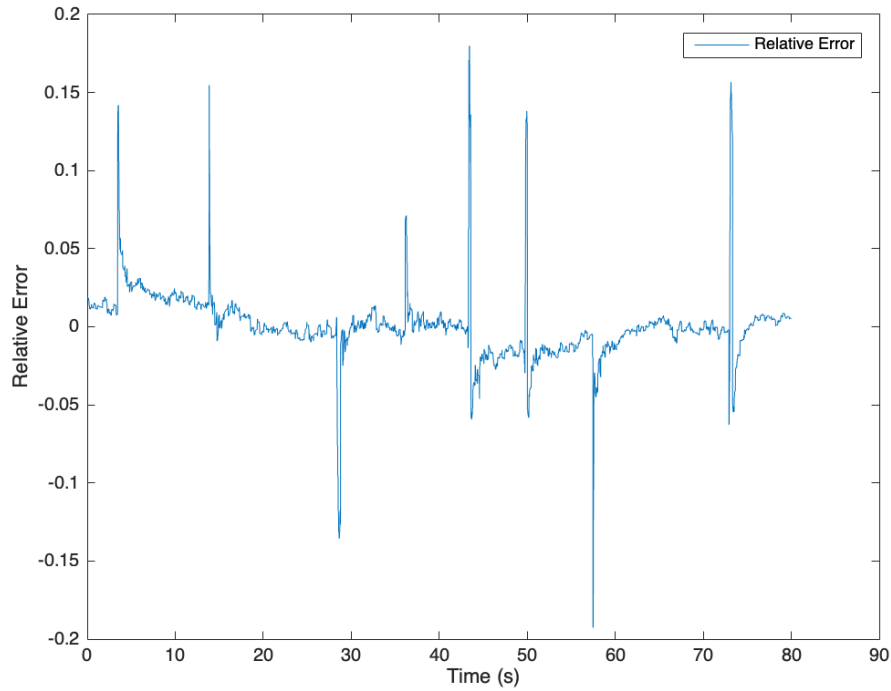


Figure 5.17: Relative error between first and second model.

Table 5.5: Mean Values of Error on the Exhaust Gas Temperature Model

Type of Error	Mean Value
Relative Error	1.17
% of Error	1.21

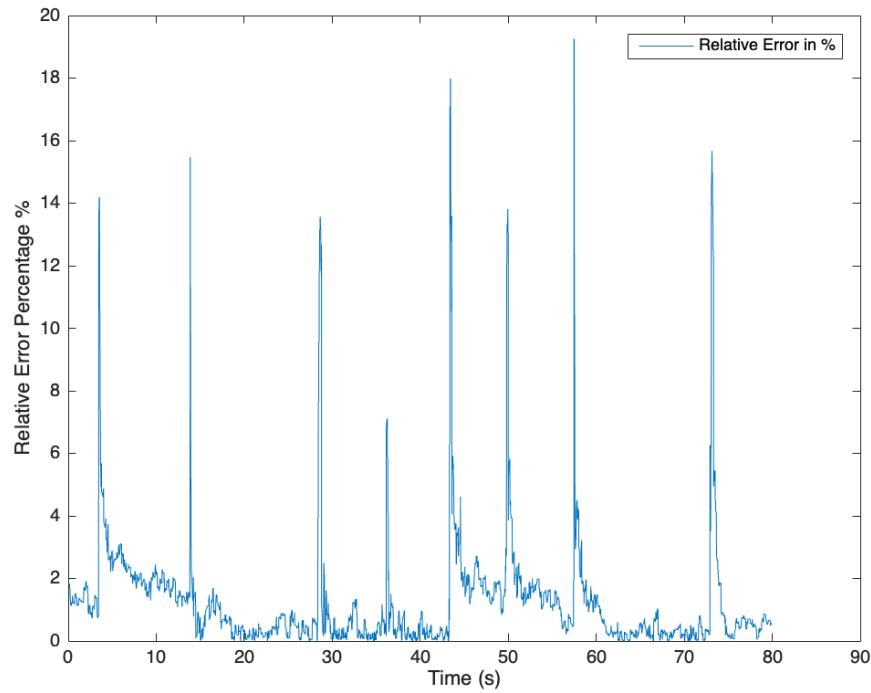


Figure 5.18: Relative error in % between first and second model.

The exhaust manifold heat loss is modelled while tuning the whole Engine model on dynamic data. The only parameter of the heat loss model that needs to be specified is the heat convection coefficient h_{tot} .

Specifying such a coefficient is hard and depends on many factors, such as the kind of material of the manifold, the geometry, the room temperature etc. Therefore, its tuning is done with the whole engine model on simulation, as there are no relevant stored data to be taken into account. Least-squares optimization was also used like the above models, the difference is that this parameter was specified while using least-squares optimization for the whole model not a specific equation that is built to minimize the difference between measured and modelled data for a specific submodel.

5.2 Initialization of the rest of the parameters

The separate submodels of the complete engine model, were all individually tuned, initiating the corresponding parameters respectively. However, there are some parameters that could not be individually tuned. Those parameters are mainly affecting the dynamic submodels and processes of the complete engine model. The parameters that are described and the corresponding subsystems are:

- V_{im} , having an effect in the Intake manifold subsystem pressure dynamics.
- V_{em} , having an effect in the Exhaust manifold subsystem pressure dynamics.
- J_{turbo} , the inertia of the turbocharger blades with their connecting axel, having an effect on the turbocharger speed dynamics.
- $c_{u,opt}$ as described on the corresponding chapter, this turbine efficiency factor is unknown and can only be evaluated with the complete model on simulation. It is mainly dependent on the turbocharger application. For smaller, faster turbochargers used in automobiles it tends to be around 0.4 to 0.5. The engine that is being modelled is of larger scale and therefore lower values are expected.
- h_{tot} , the exhaust manifold heat convection factor, needed to calculate the heat losses. This factor is extremely hard to accurately predict beforehand as it is affected by many separate factors.

For the model that is described in this thesis, the engine speed was decided to be set as an input and not an output parameter. Therefore, the complete engine model has three dynamic outputs, namely, p_{im} , p_{em} , $n_{turbocharger}$. The parameters that have an effect on these dynamics (described above) and the rest of the above described parameters were estimated by solving the least squares optimization problem described below:

$$\min V(\theta) \tag{5.2.1}$$

$$\text{s.t. } \theta_{min} \leq \theta \leq \theta_{max}$$

Where $V(\theta)$ is the cost function given by:

$$V(\theta) = \sum_{k=1}^K \sum_{j=1}^J \frac{1}{L_j} \sum_{l=1}^{L_j} (w_k^{meas,dyn}[l] - w_k^{mod,dyn}[l])^2 \tag{5.2.2}$$

Where θ represents the tuning parameter, whether it a manifold's volume or the turbocharger inertia and θ_{min} and θ_{max} are the lower and upper bounds for each parameter respectively. The cost function minimizes the errors between the dynamic measurements $w_k^{meas,dyn}$ and the dynamic simulations $w_k^{mod,dyn}$ of the complete engine model. Further, J is the number of Data sets in the dynamic measurements with $J = 4$, the total number of Data sets used. The constant L_j is the number of samples in data set j where both data and simulations are sampled with a frequency of 1000Hz. There are three dynamic outputs in the simulations and two in the measurements, since there is no sensor for the turbocharger rotational speed, namely:

- $w_1 = p_{im}$
- $w_2 = p_{em}$

The initialization of the unknown parameters is done by solving the problem described in 5.2. The initial guesses that are attributed to the parameters are shown below:

Table 5.6: Manual Initial Parameter Setting

Parameter	Value
V_{im}	0.03 m^3
V_{em}	0.025 m^3
J_{turbo}	$3 * 10^{-4} \text{ kgm}^2$
$c_{u,opt}$	0.32
h_{tot}	$99 \frac{\text{W}}{\text{m}^2 \text{K}}$

The results of these guesses are shown below in Figures 5.19, 5.20. It should be noted that the initial conditions for the dynamic outputs of the simulation model are set to be the same as the ones on the measured dynamic data.

Again, it is reminded that the cost function in this instance is only minimizing the error between the measured Intake Manifold Pressure and Exhaust Manifold Pressure and the corresponding modelled Data.

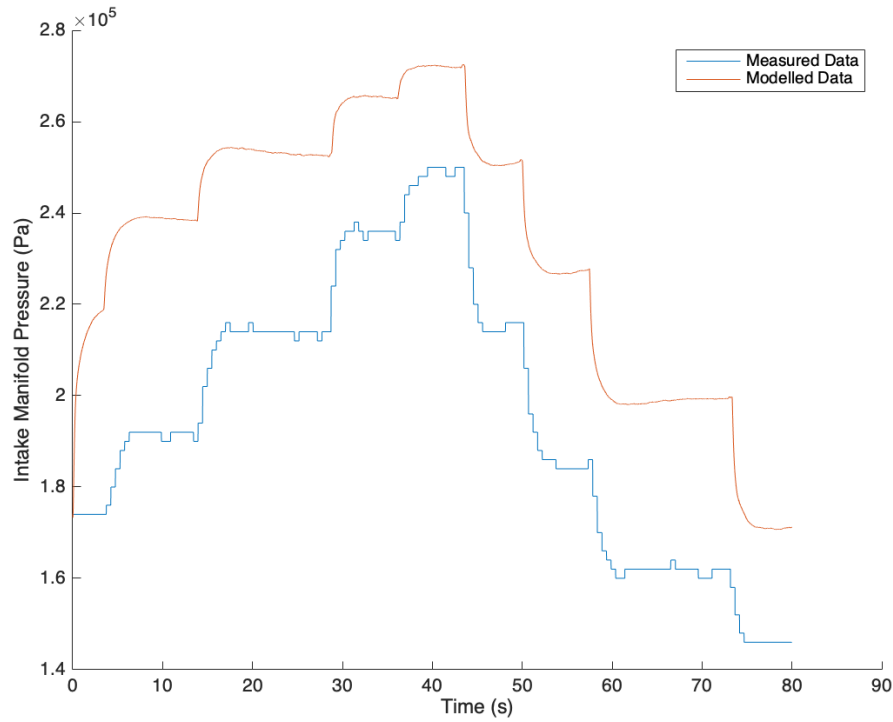


Figure 5.19: Intake Manifold Pressure for initial guess

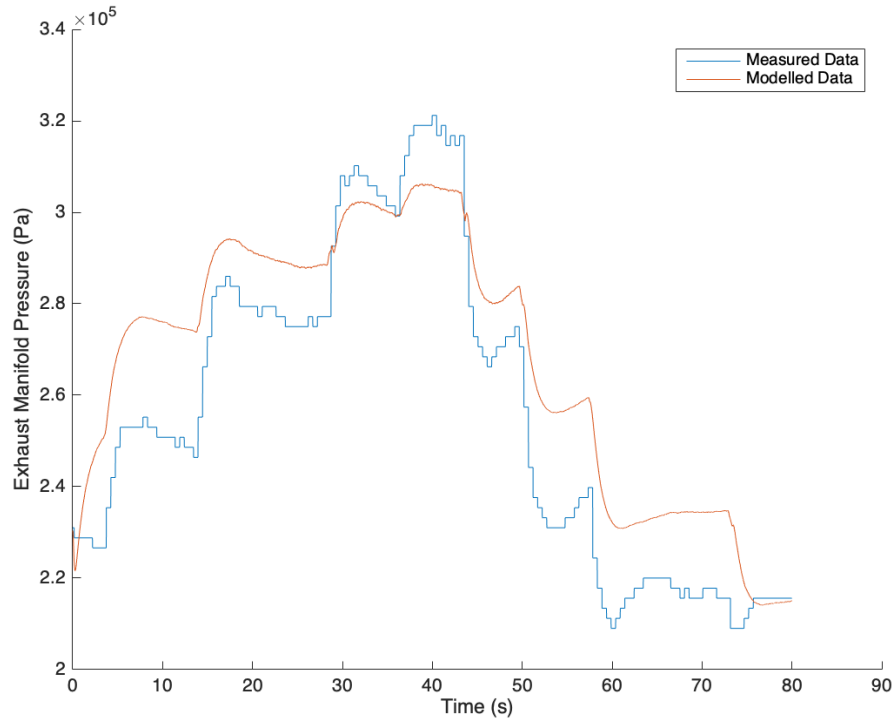


Figure 5.20: Exhaust Manifold Pressure for initial guess

These results are evident of how wrong the initial guesses are compared to their actual value. The iterative process of minimizing the cost function is triggered inside Matlab. The process is automatic and the parameters are all changed accordingly in each iteration until the best convergence possible between the set Measured and Modelled Data happens. The iterative process ends after 6 iterations in about 5 minutes. The results are far from ideal and the model is not yet tuned. However the unknown parameters have been initialized. The cost function goes from a beginning value of 2.1 to a final value of 1.8. The result diagrams may seem inaccurate in capturing the stationary measurements (which is normal, as the stationary parameters are not yet tuned) however they are vastly improved in the dynamic responses of both the intake and exhaust manifold pressures. The result convergence is shown in Fig.5.21, 5.22.

These results are only representing the fifth Data Set. The tuning process continues further on, where both the now initialized unknown parameters and the previously initialized separate submodel parameters are tuned on a new cost function.

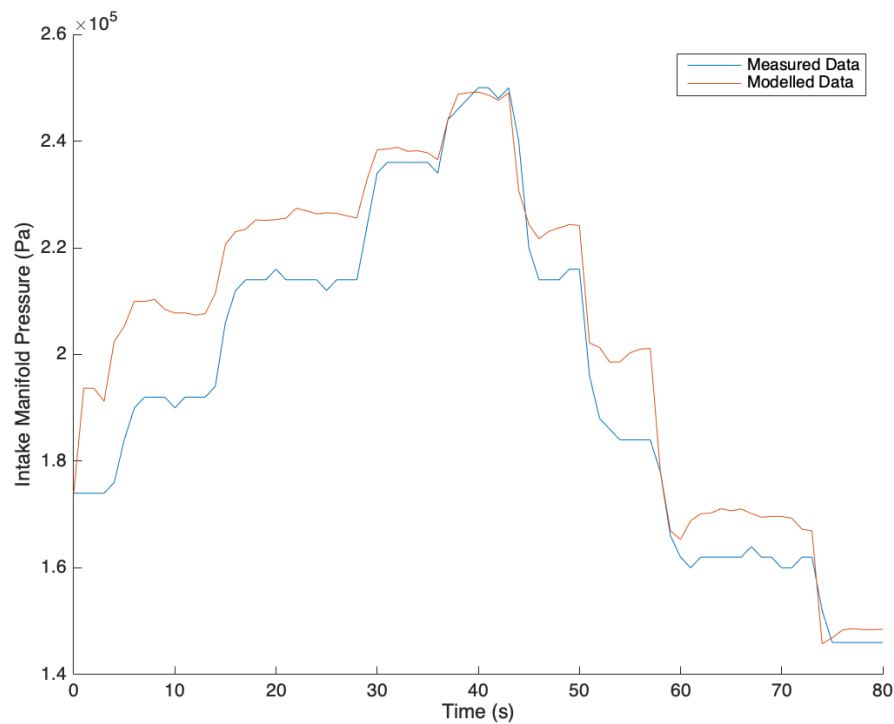


Figure 5.21: Intake Manifold Pressure after the unknown parameter initialization

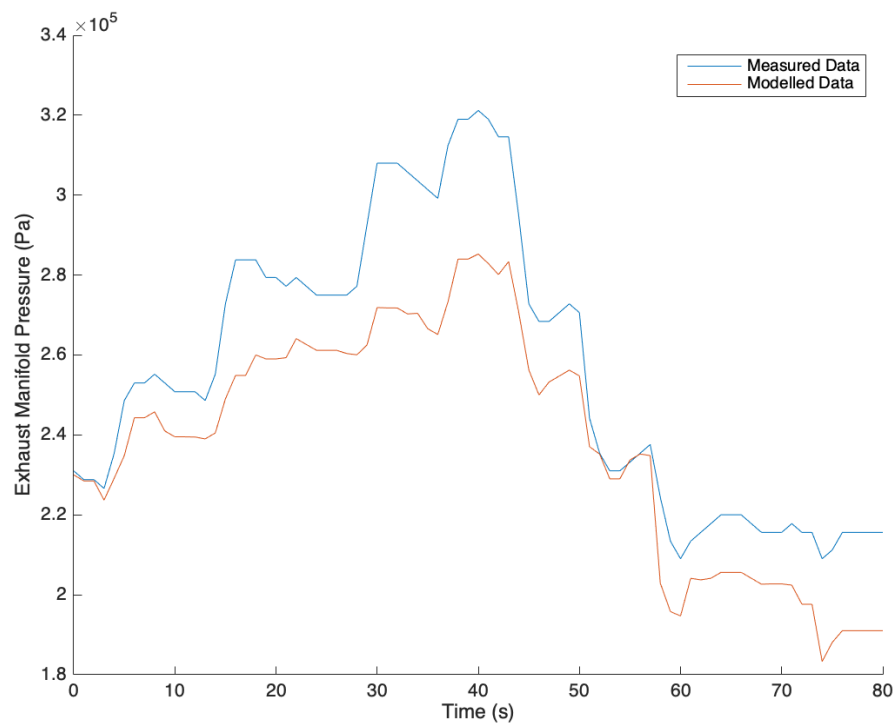


Figure 5.22: Exhaust Manifold Pressure after the unknown parameter initialization

Table 5.7: Automatic parameter tuning, after first cost function

Parameter	Value
V_{im}	0.0255 m^3
V_{em}	0.0165 m^3
J_{turbo}	$1.23 * 10^{-3} \text{ kgm}^2$
$c_{u,opt}$	0.365
h_{tot}	$95.2 \frac{\text{W}}{\text{m}^2\text{K}}$

5.3 Tuning the static parameters using the complete model

On the previous chapter, the tuning process for the dynamic unknown parameters is shown using Data Set 5. At this point, all the parameters of the complete model are initialized and close to their tuned value. At this step, the static parameters are further tuned using the complete engine model, the dynamic measurements from the Lab and the static measurements. The goal is the same as the previous chapter, to minimize the cost function between measured and modelled Data with the difference that at this point, only the static parameters are getting tuned and that the cost function is aiming to minimize both the dynamic submodel errors as well as the static submodel errors. Namely, the parameters that are getting tuned are:

- $c_{u,opt}$
- vol_1, vol_2, vol_3
- a_1, a_2, a_3
- h_{tot}
- $\eta_{turbine,max,actual}$
- $d_{1,venturi}$
- $d_{2,venturi}$
- $xO_{im,init}$
- $xO_{em,init}$

The cost function that is used for this step of the tuning process is shown below:

$$V(\theta) = \frac{1}{KJ} \sum_{k=1}^K \sum_{j=1}^J \frac{1}{L_j} \sum_{l=1}^{L_j} \left(\frac{y_k^{meas,dyn}[l] - y_k^{mod,dyn}[l]}{\frac{1}{M} \sum_{m=1}^M y_k^{meas,stat}[m]} \right)^2 + \frac{1}{NM} \sum_{n=1}^N \sum_{p=1}^M \left(\frac{z_n^{meas,stat}[p] - z_n^{mod,stat}[p]}{\frac{1}{M} \sum_{m=1}^M z_n^{meas,stat}[m]} \right)^2 \quad (5.3.1)$$

Where θ represents the static model parameters.

The first part of 5.3.1 minimizes the errors between the dynamic measurements and the dynamic submodel results. The second part minimizes the errors between the stationary measurements and the stationary submodels.

There are three dynamic simulations but only two measurements (we have no sensor for the turbocharger speed, which is the third dynamic simulation), namely $y_1 = p_{im}$, $y_2 = p_{em}$. There are multiple outputs in the stationary models. However, one that is actually measured and is of importance for the creation of this model is the Lambda output which describes the air to fuel ratio inside the cylinder. As static input to the cost function the lambda and the Torque measurement were added, the because it is the only available measurement from the Lab's sensors that can describe the overall accuracy of the complete model. Since if the Lambda model is accurate compared to the measured value, the flows both inside and out the cylinder are also

accurate (and therefore also the turbocharger mass flows are accurate). Therefore we get $z_1 = \textit{Lambda}$ and $z_2 = M_e$. The number J of Data sets is 5 and the data for both the measurements and simulations are sampled at $1000Hz$ or 1000 times a second. The number M designates the number of stationary operating points in the stationary measurements. The number differs from Data set to Data set. The first portion of the equation is normalized with KJ and the second with NM so that the stationary and dynamic measurements are penalized equally.

The optimization problem was solved using a standard Matlab non-linear least-squares solver (for more see [19]) with the tuning parameters initialized from the submodel tuning that happened beforehand. The results of the optimization are shown in Table 5.8.

Table 5.8: Automatic parameter tuning, using the second cost function

Parameter	Final Value
vol_1	-0.0001097
vol_2	-0.00681
vol_3	1.2865
$c_{u,opt}$	0.36108
h_{tot}	$95.26 \frac{W}{m^2K}$
a_1	2173.73
a_2	-3.055
a_3	116.305
$\eta_{turbine,max,actual}$	0.6784%
$d_{1,venturi}$	0.01803
$d_{2,venturi}$	0.0116
$xO_{im,init}$	0.218
$xO_{em,init}$	0.152

5.4 Tuning Completion

After the tuning process is over, the model is fairly capable of describing the dynamic tuning data that were used. The mean errors between the model and the measured dynamic data are all shown below in table 5.10. The diagrams presenting the model performance against the measured data are also presented below for each corresponding Data set that was acquired in the Lab. It should be noted that all the data sets are specific quantity sequences in the time Domain. From the acquired Data from the Lab setup the following were used as inputs to the model:

- n_{eng} , since the model doesn't include engine shaft dynamics, the engine speed is needed as input.
- W_f , the fuel rate.
- p_{es} , the pressure in the entrance of the After Treatment Unit, just so that the factor of the turbine pressure is more accurate (p_{es} is slightly increased over the atmospheric pressure).
- $p_{im,init}$, $p_{em,init}$, $n_{turbocharger,init}$, the initial dynamic states for each submodel. The first two initial states are easily acquired just by the Data Set. The third, $n_{turbocharger,init}$, the turbocharger rotor speed, is estimated manually using the compressor map and the initial $p_{im,init}$ intake manifold pressure as inputs.
- $p_{em,meas}$.

The last input's use was inevitable since the Waste Gate control signal was not measured from the Lab's setup sensors. The built up boost had to be controlled, therefore the Waste Gate flow and dynamics had to be somehow specified, this was the only way to compute the required flow. Without the correct Waste Gate flow, in other words, with inaccurate boost control, the modelled exhaust manifold pressure always reaches extremely high values, which cause the intake manifold pressure to skyrocket as well, which is of course inaccurate. More on the calculation of the Waste Gate flow will be shown in the Appendix A.

Table 5.9: The mean absolute relative error % between the dynamic tuning data and diesel engine model simulation where the model parameters are tuned according to the method in this section.

Data Set:	A	B	C	D	E
$n_e(rpm)$	1800	1800	1900	1900	2000
p_{im}	3.4262	3.4283	2.77	2.2867	4.17
p_{em}	0.07	0.986	0.0002	0.0003	0.68
$Lambda$	3.1284	5.12	2.9563	4.1	4.27
$Torque$	3.47	4.69	3.2452	3.28	3.43

5.4.1 Result Diagrams

Data Set A:

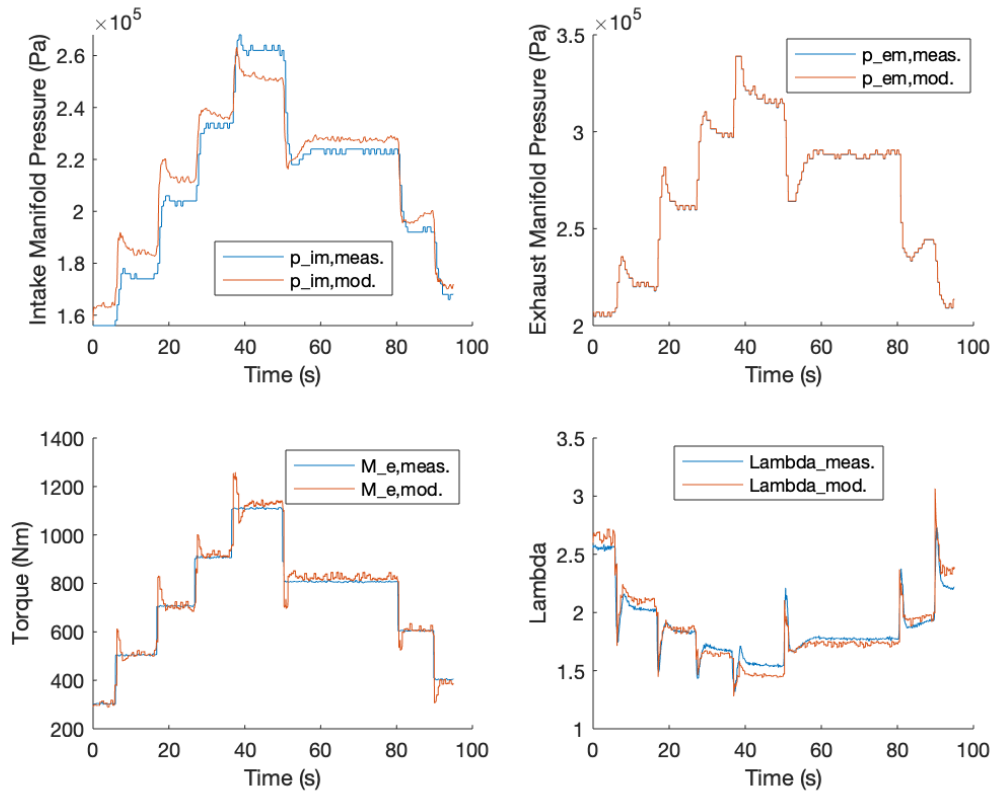


Figure 5.23: Data Set A, Model vs. Measurements

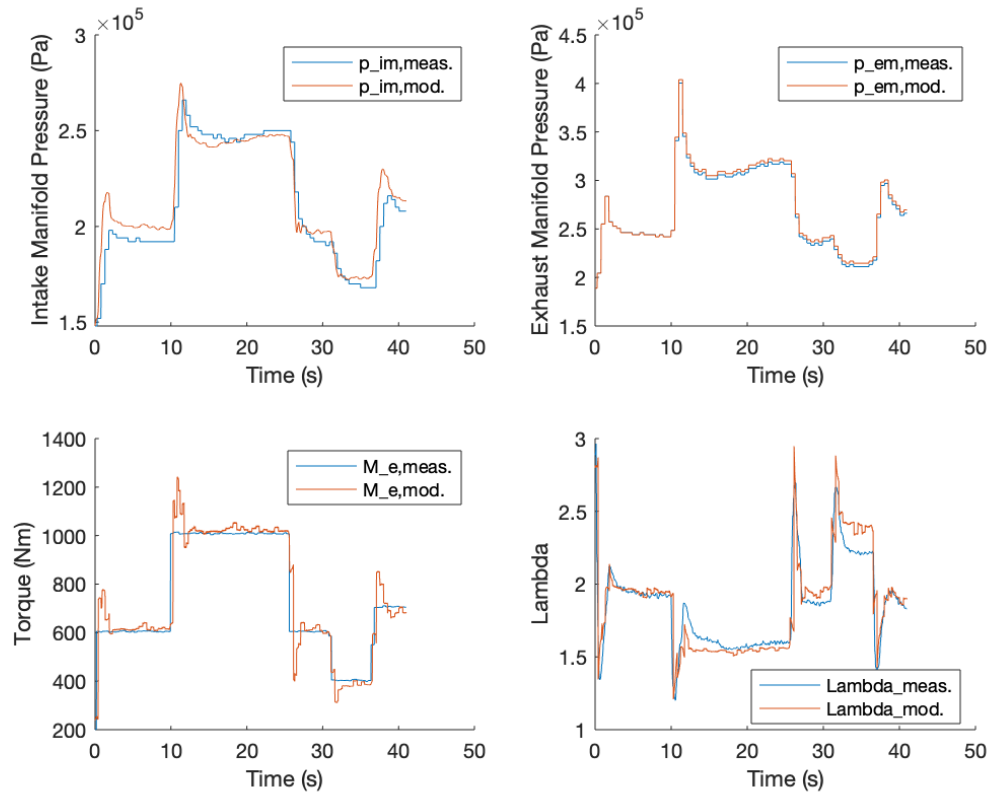
Data Set B:

Figure 5.24: Data Set B, Model vs. Measurements

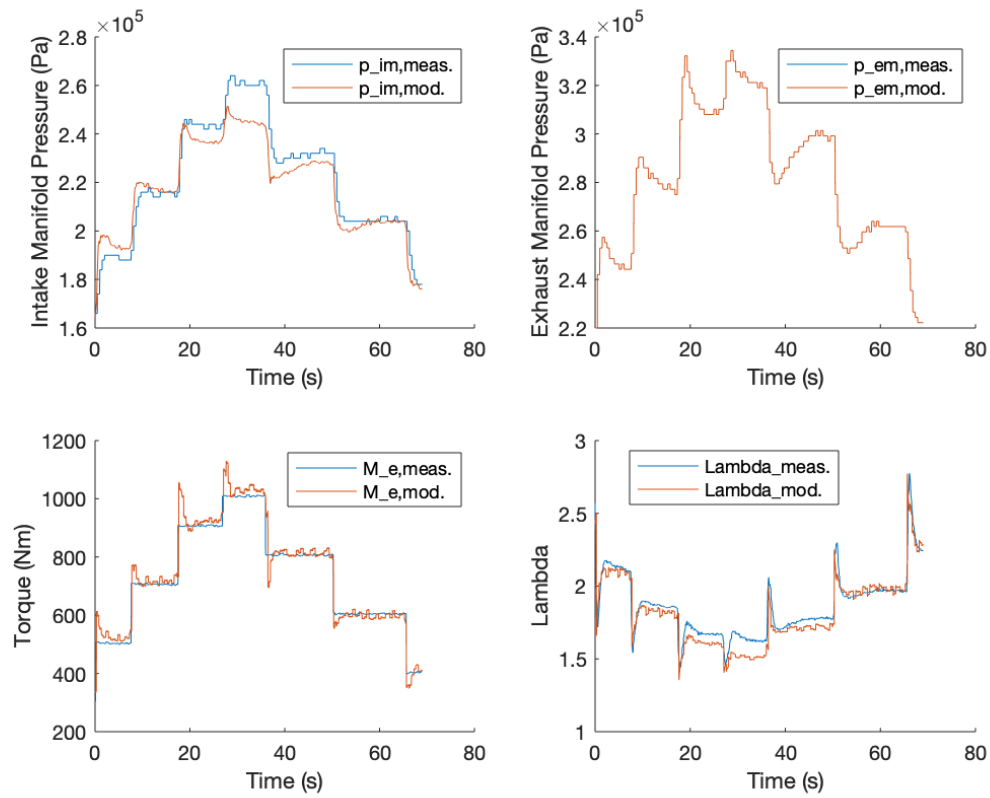
Data Set C:

Figure 5.25: Data Set C, Model vs. Measurements

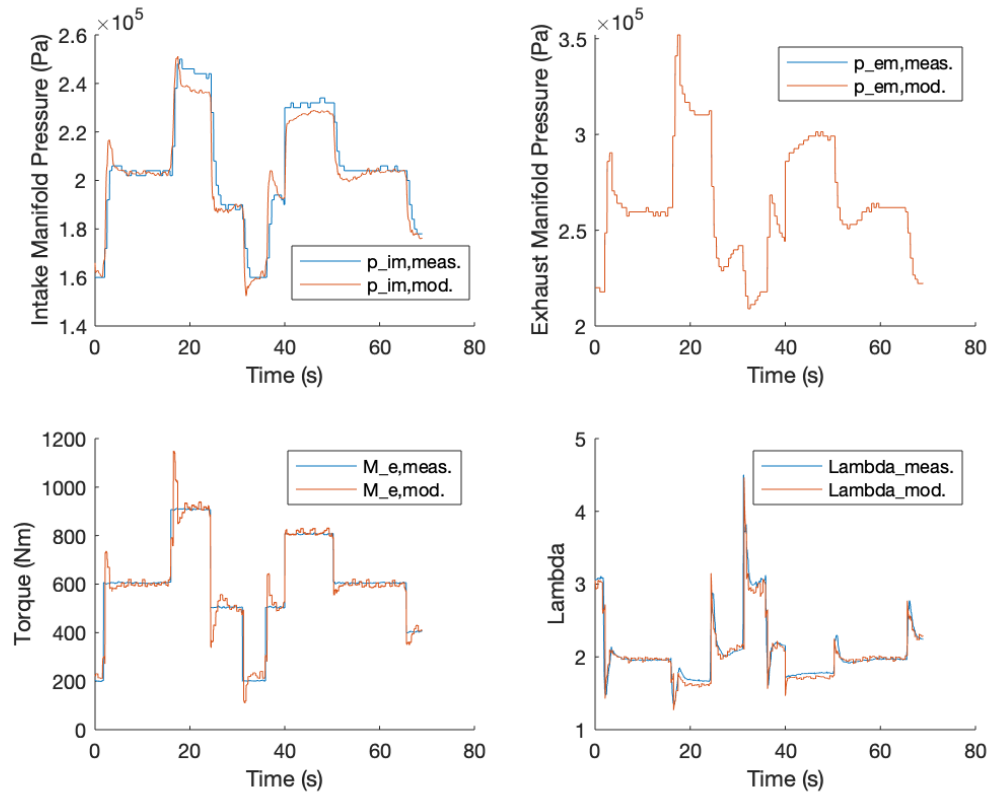
Data Set D:

Figure 5.26: Data Set D, Model vs. Measurements

Data Set E:

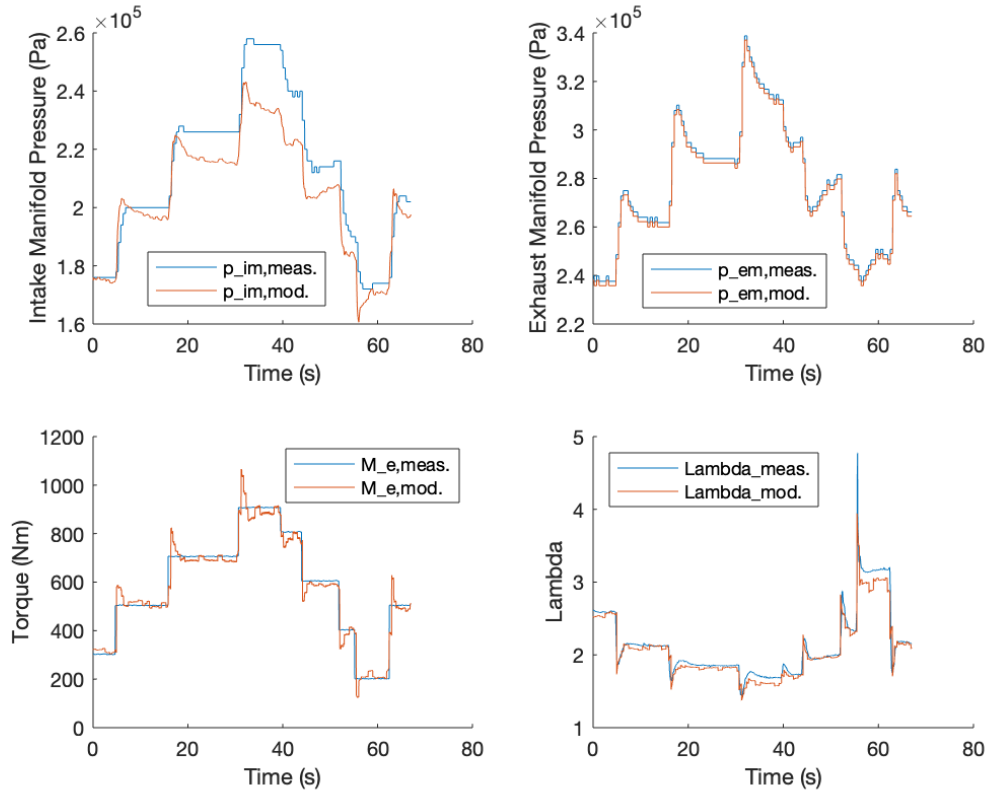


Figure 5.27: Data Set E, Model vs. Measurements

It is evident, that the model is tuned since the modelled results are very close to the measured ones. The transitional spikes that are witnessed mainly in the torque figures are not signs of bad fitment, they are present because of two reasons. Firstly, the measured data that were fed to the optimization procedure were all sampled in 10Hz (10 measurements per second) mainly to reduce the vector sizes and the calculation times, if the sampling was to be done with higher frequencies the optimization times would be considerably longer. Secondly, the dynamic transitions are almost instant and cause the exhaust pressure sensor to measure transitional spikes before it settles down to the correct stationary measurement.

5.5 The importance of the initialization and the cost function

The tuning process for the engine model relies on two different processes. First, the initialization of the tuning parameters using the separate submodels and the corresponding data. Second, the automatic parameter tuning (both dynamic and static) using dynamic tuning data. This is the most efficient way of tuning [3], since it is considerably more accurate to a tuning process that relies only on one of the above processes.

Solely initializing the model parameters using the independent submodels will yield great errors when the complete model is simulated, since the errors that are taking place during the fitting processes of the independent submodels will accumulate together resulting in a greater final error on the complete model. Moreover, some parameters are affecting the model accuracy more than others, since some parameters are parts of more submodels than others.

On the other hand, singlehandedly, using the cost function as a mean of tuning, will result in a probably accurate model as regards the dynamic measurements but it will also be a model that doesn't present logical values for the separate submodel parameters. If the submodel parameters are not initialized at the beginning and no logical or computational bounds are set to each parameter, the solver of the cost function will tune each parameter recklessly just so the final model is accurate on describing the dynamic tuning measurements, meaning that the independent submodels will have no logical outputs whatsoever. Additionally, it is very difficult and time consuming, estimating completely unknown parameters that haven't been firstly initialized using least squares cost functions. The solver needs to have accurate initial estimations for each parameter to reach a conclusion, that is especially true if the problem is relying on many parameters.

The way the tuning was done, both complete engine model and independent submodel accuracy is achieved.

5.6 Validation

The model parameters at this point, are all tuned and the complete model is ready. The model's accuracy compared to the dynamic and stationary (Lambda and Torque) tuning data, in the form of the 5 Data Sets, was calculated and the maximum mean error presented was at 5.12%. To test the model's accuracy overall, the model will be simulated for new Data Sets, that the model was not tuned for beforehand.

Similarly with the tuning process, the errors will be calculated and the relative diagrams will be presented for each new Data Set. For the validation process, three different Data Sets were used, namely Data Sets F,G,H. These Data Sets cover a heavily transient operation on 1800, 1900 and 2000 engine revolutions per minute respectively. In the validation chapter, the complete model inputs and outputs will also be presented.

It should be noted that the fifth input to the model, $p_{ATU,in}$ could be omitted and instead use the atmospheric pressure. In that case the model would have to

be tuned from the beginning to account for that minor difference in inputs. It's usage was mainly for accuracy reasons and is generally needed for engines that are equipped with an ATU (After Treatment Unit). If the engine was not equipped with an ATU this value would be the same with the atmospheric pressure.

Additionally, having the required parameters tuned, for better presentation of the results, the model input pressures were filtered using Matlab's function *sgolayfilt(data,order,framelength)* (Savitzky-Golay filtering) which is basically fitting a polynomial of specified order and frame length (frame-length must be an odd number) to the given Data in order to filter them. The engine speed input is unfiltered, as the submodels that use the engine speed as input were all initialized and tuned with unfiltered data. Additionally, it should be noted that in the validated data the transitional steps are much smoother and realistic eliminating any unwanted "spikes" in the results.

Data Set F:

The required inputs for the model to run:

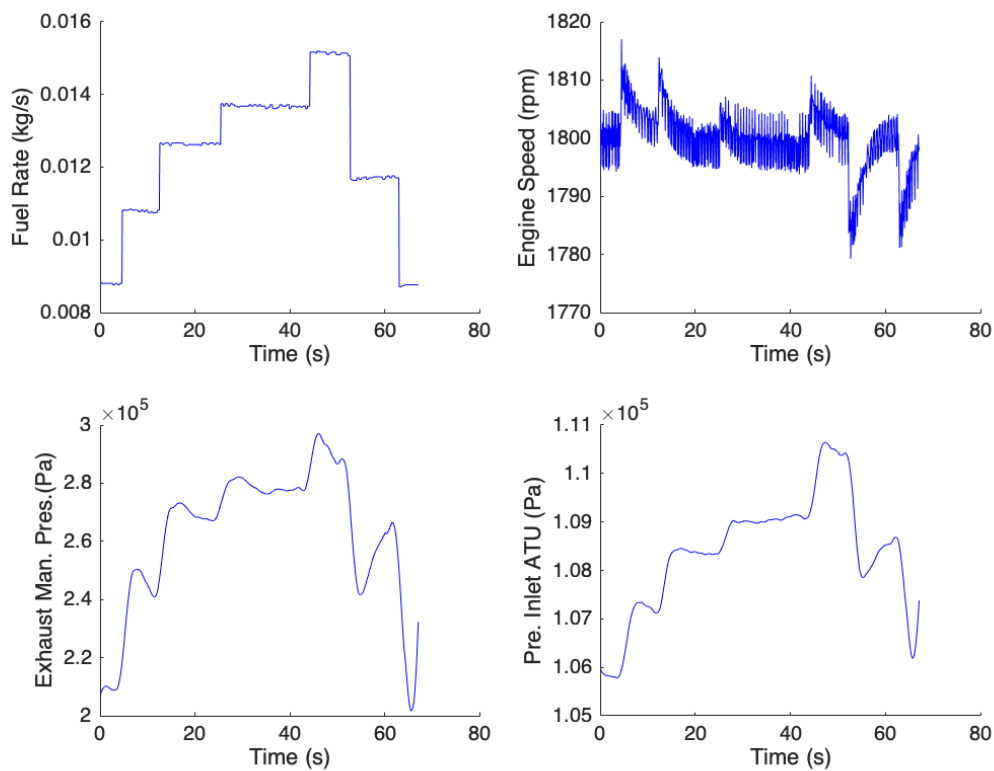


Figure 5.28: Data Set F: Inputs to the Model

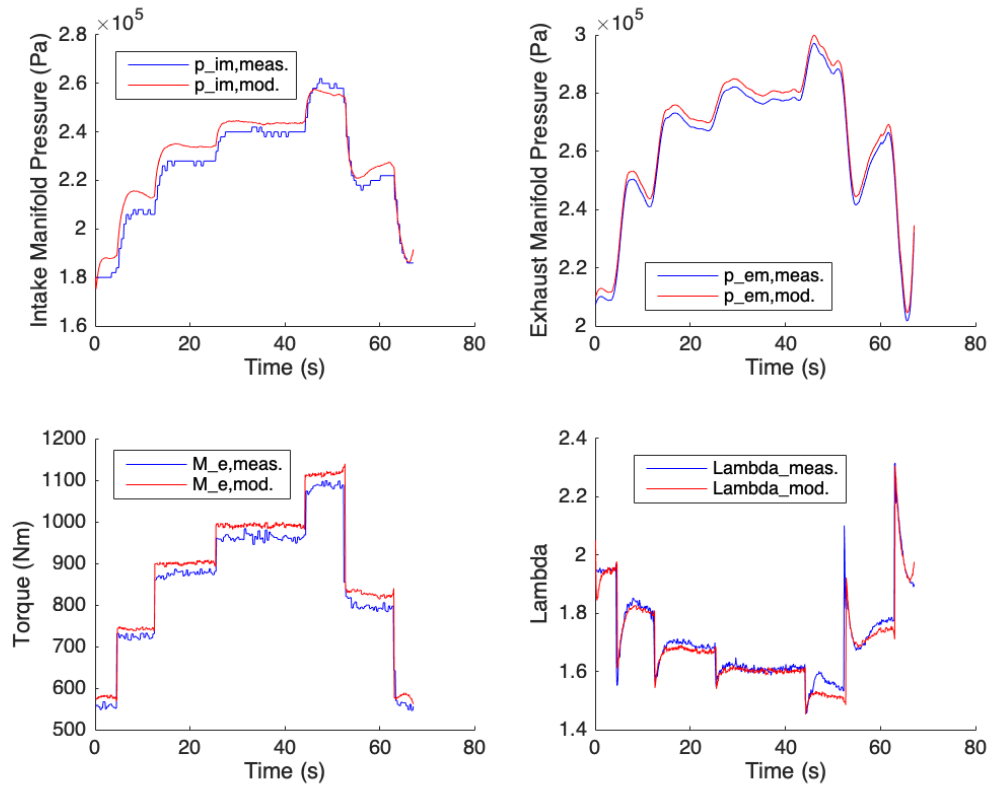


Figure 5.29: Data Set F, Model vs. Measurements

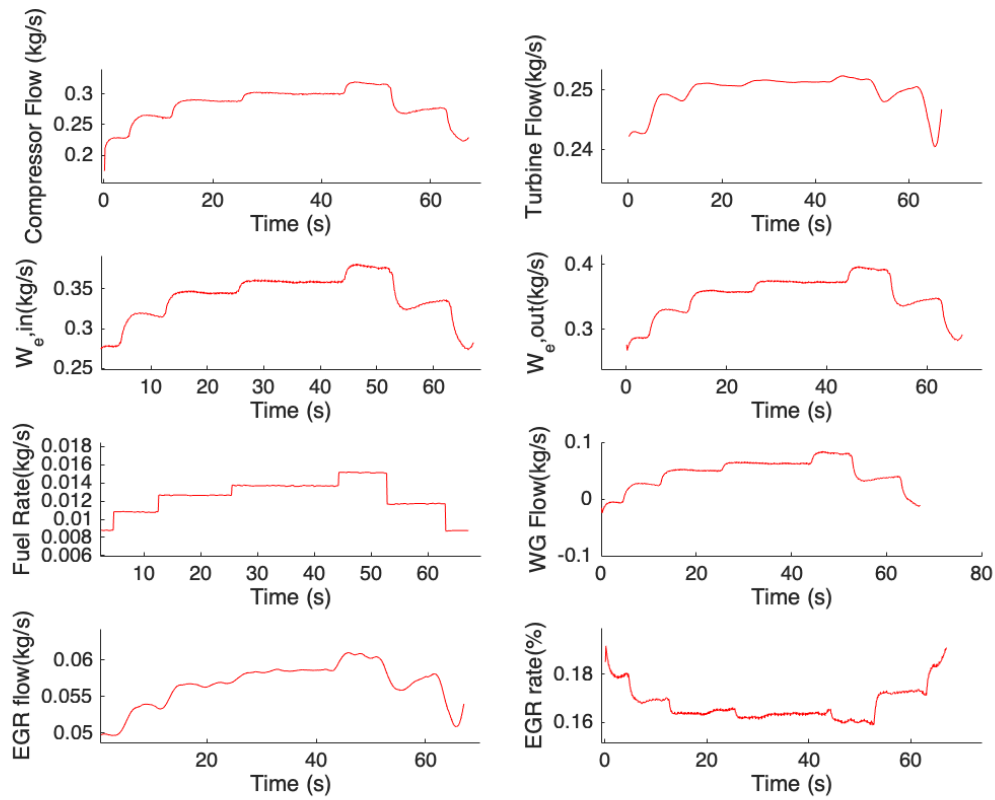


Figure 5.30: Data Set F, Flows

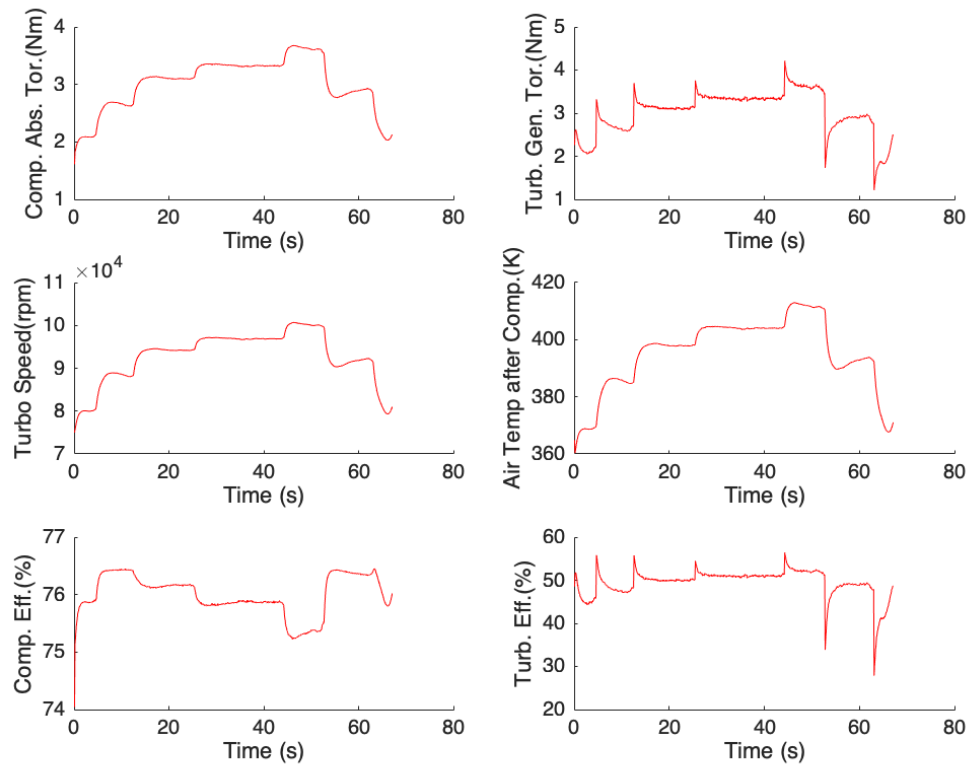


Figure 5.31: Data Set F, Turbocharger Modelled Outputs

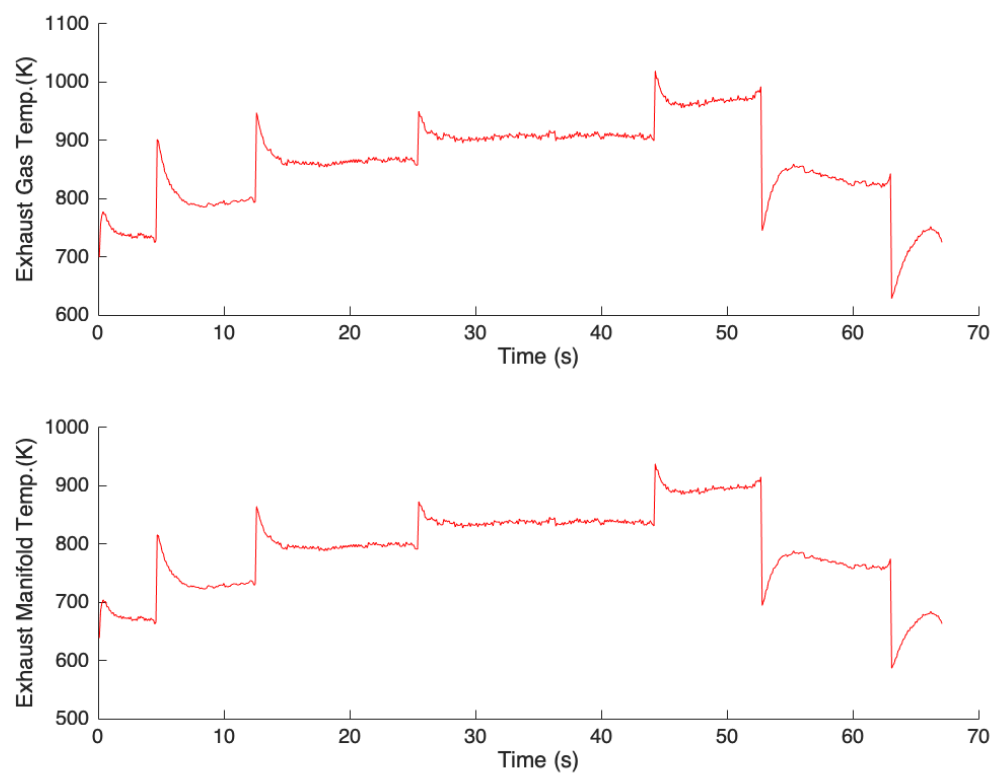


Figure 5.32: Data Set F, Exhaust Temperatures

Data Set G:

The required inputs for the model to run:

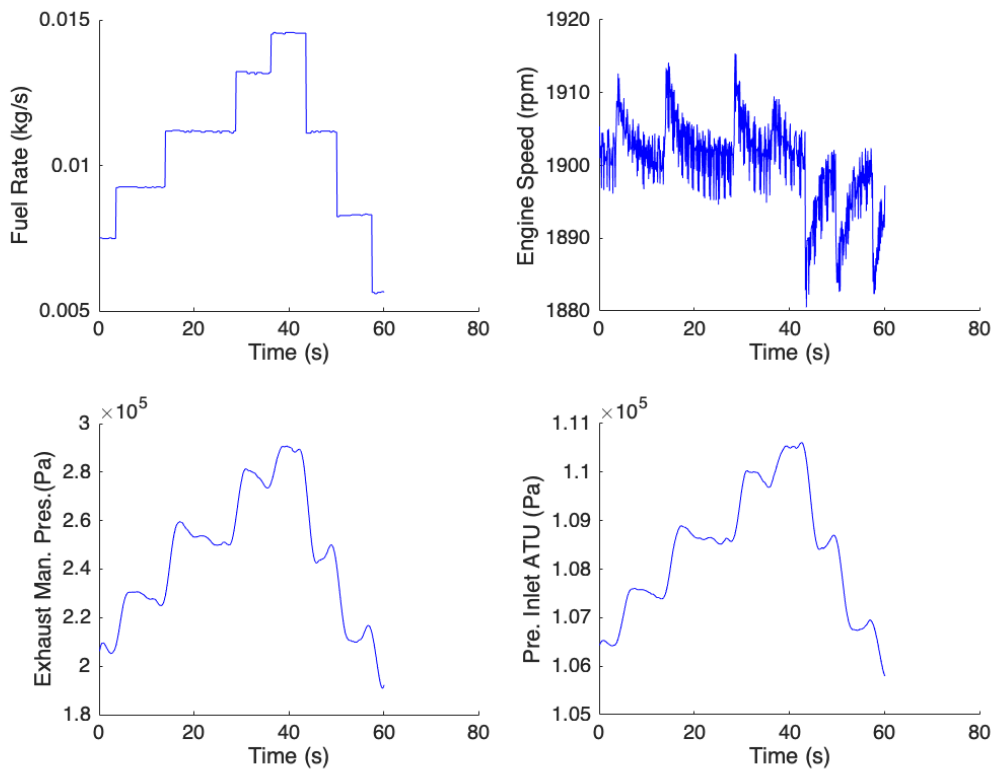


Figure 5.33: Data Set G: Inputs to the Model

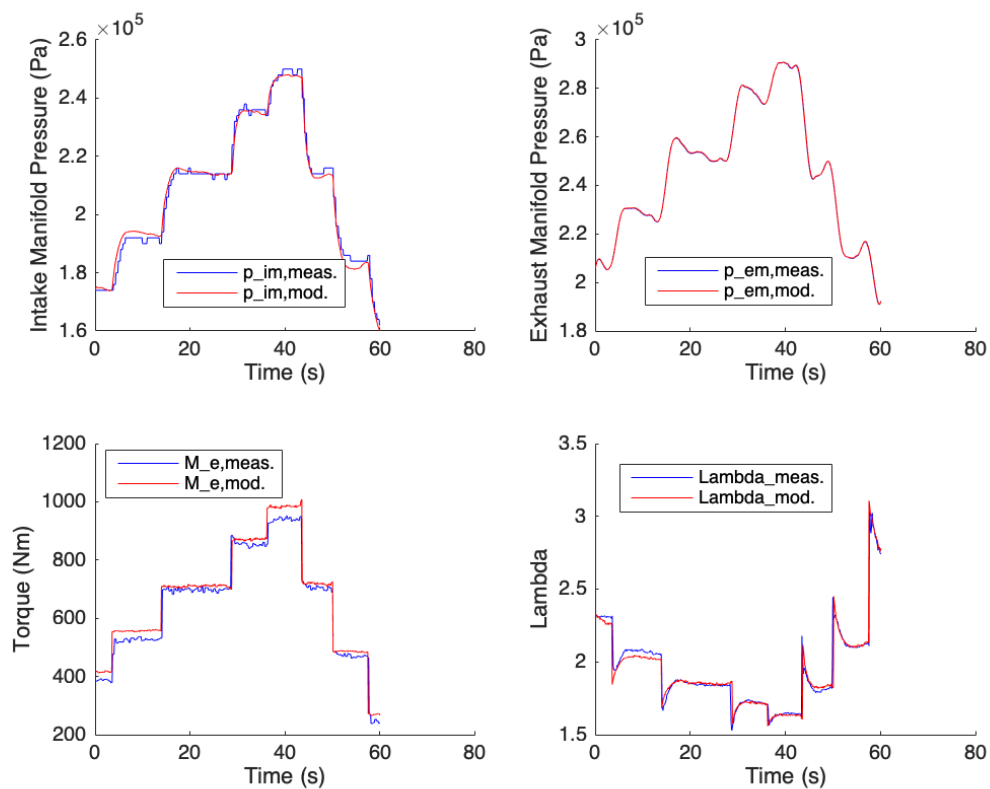


Figure 5.34: Data Set G, Model vs. Measurements

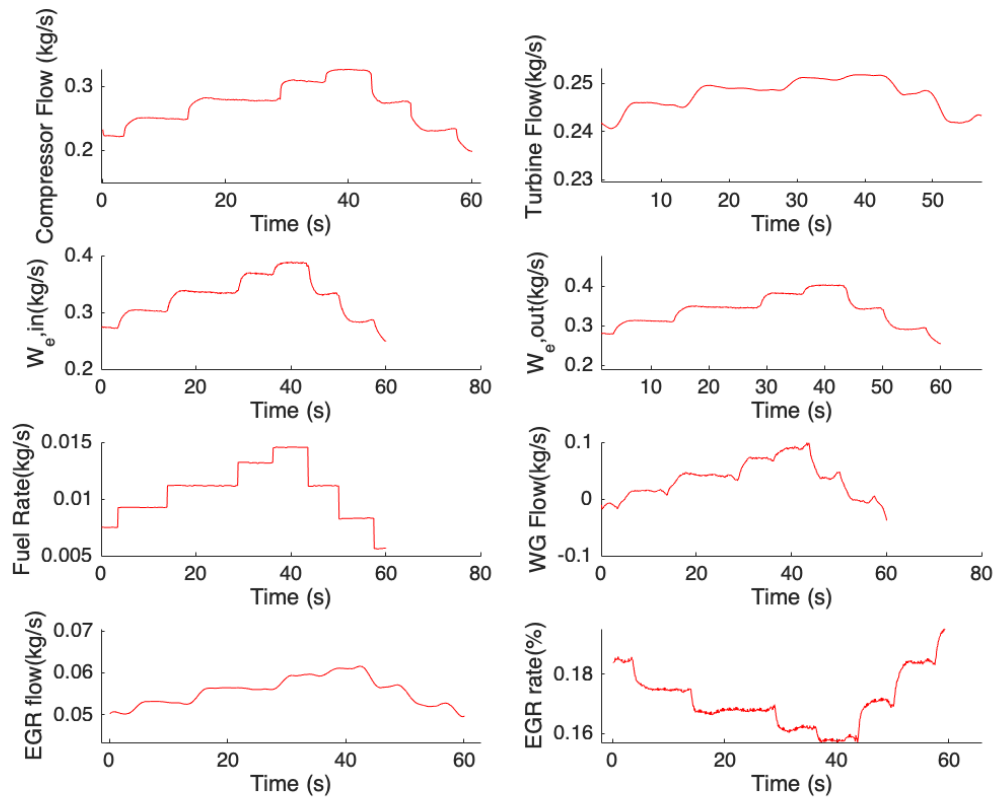


Figure 5.35: Data Set G, Flows

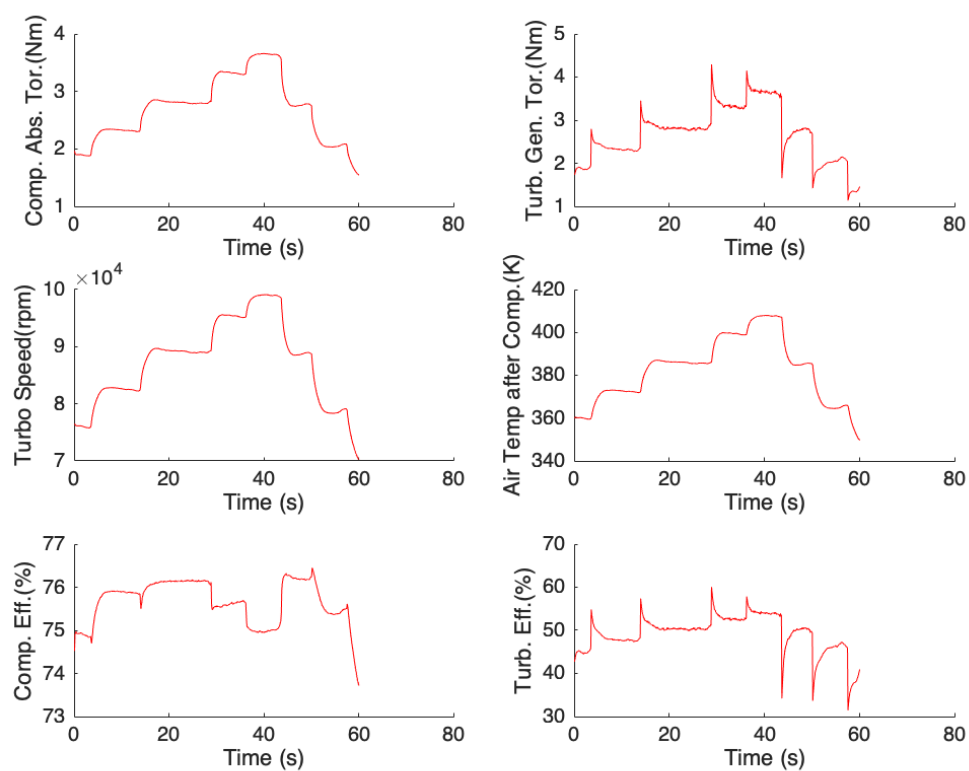


Figure 5.36: Data Set G, Turbocharger Modelled Outputs

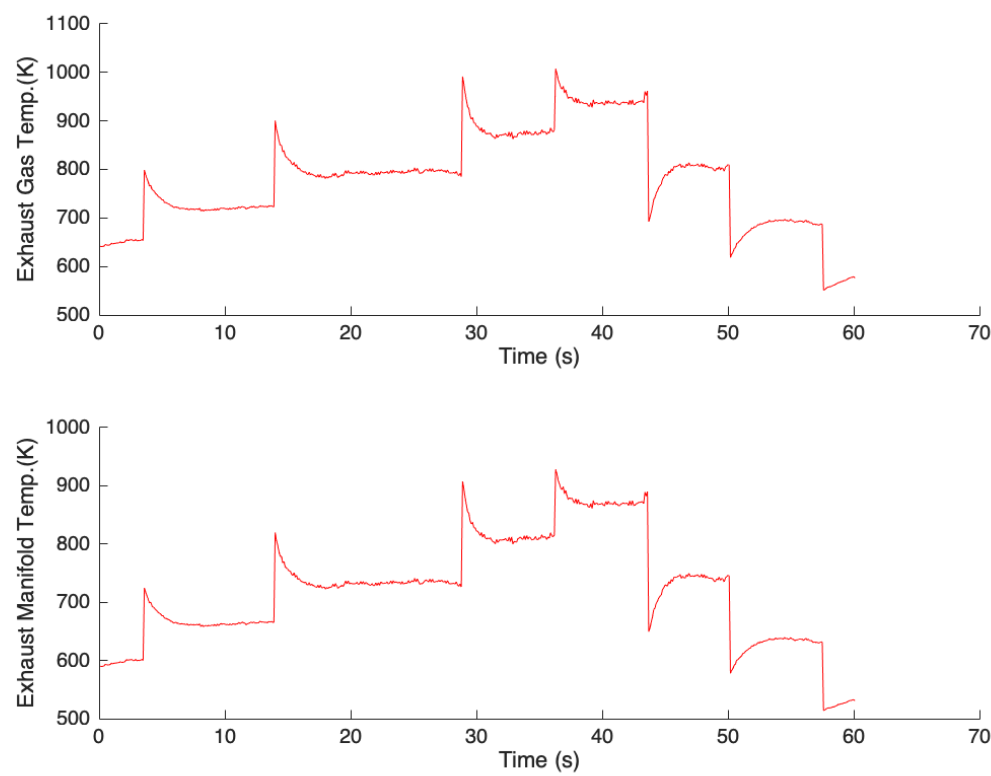


Figure 5.37: Data Set G, Exhaust Temperatures

Data Set H:

The required inputs for the model to run:

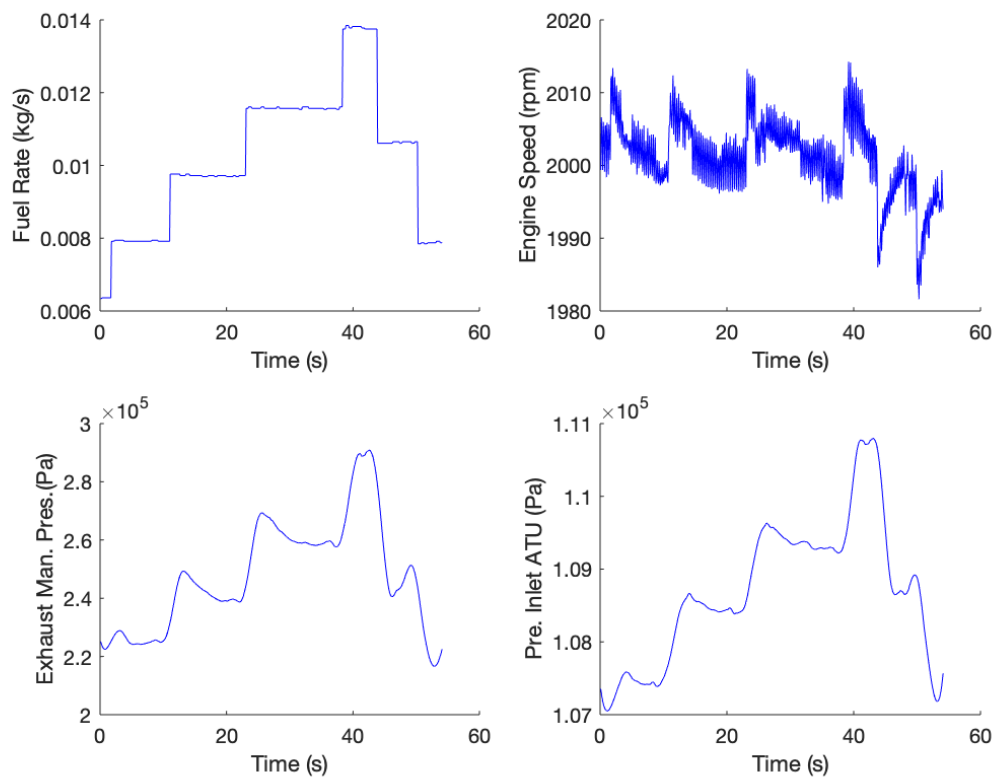


Figure 5.38: Data Set H: Inputs to the Model

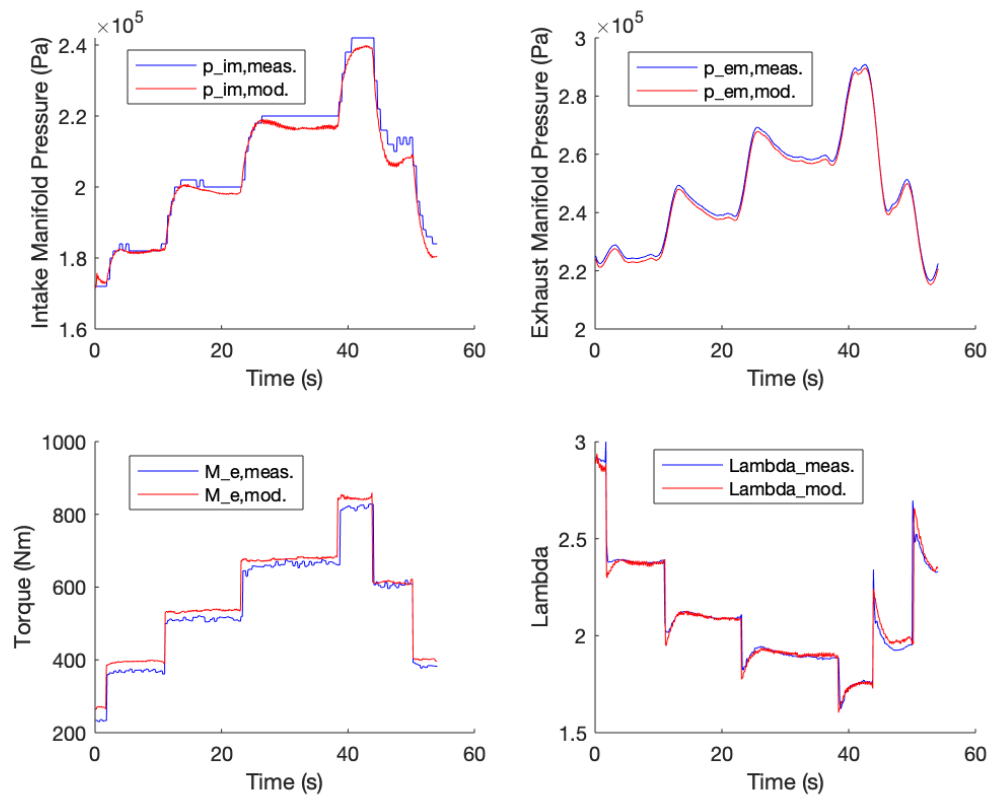


Figure 5.39: Data Set H, Model vs. Measurements

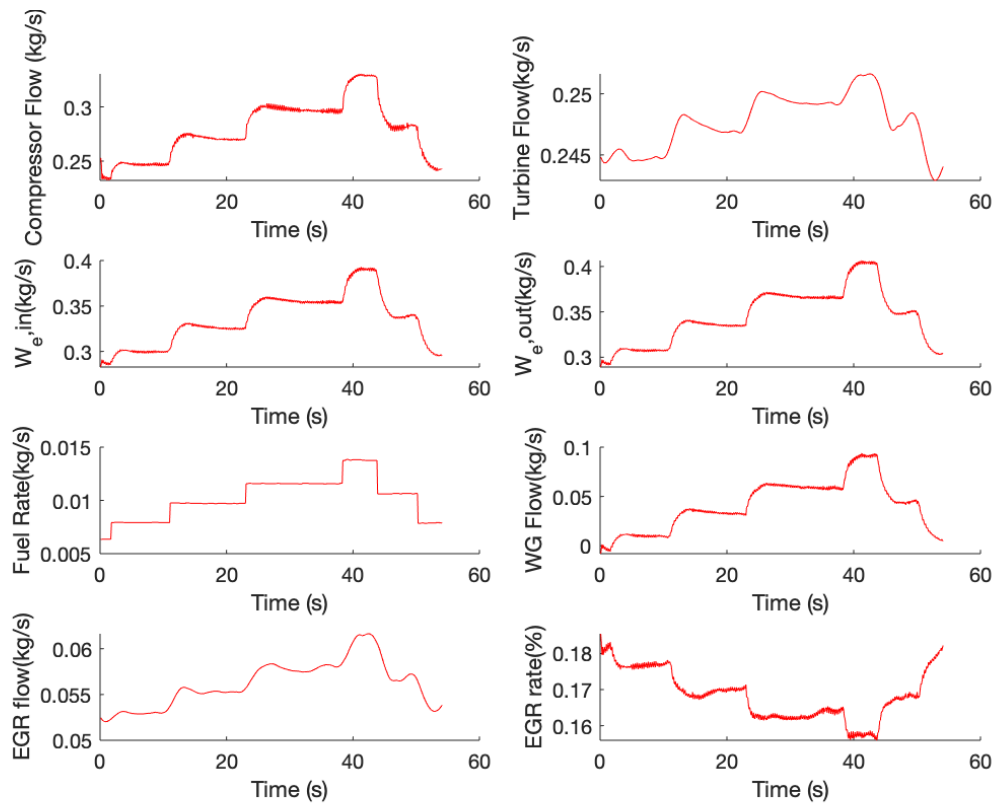


Figure 5.40: Data Set H, Flows

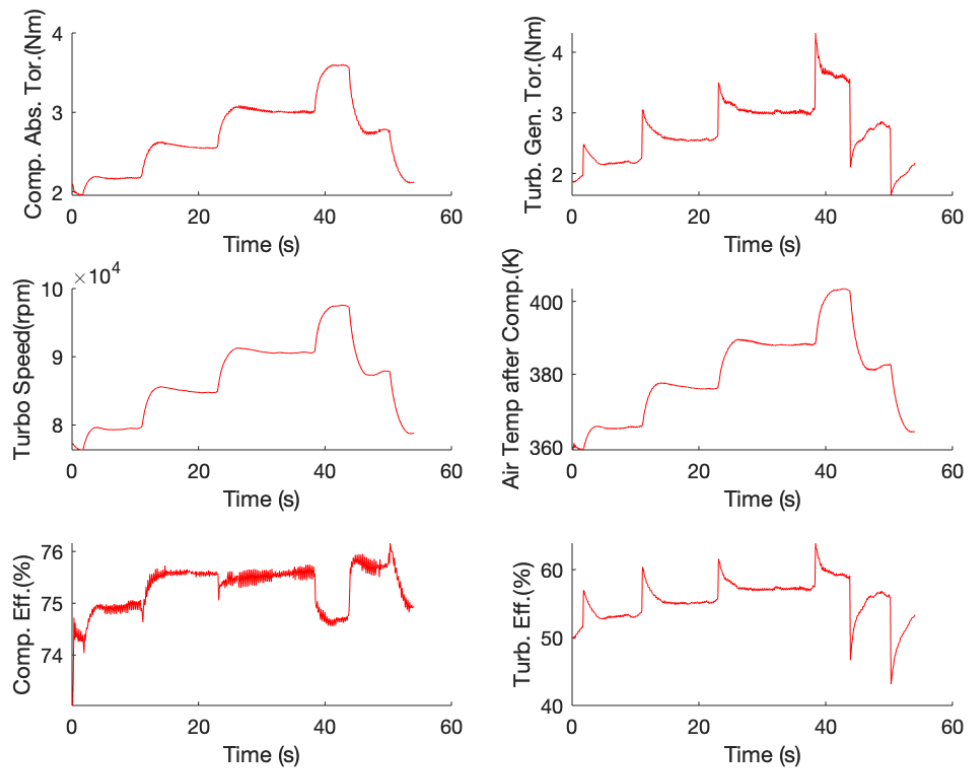


Figure 5.41: Data Set H, Turbocharger Modelled Outputs

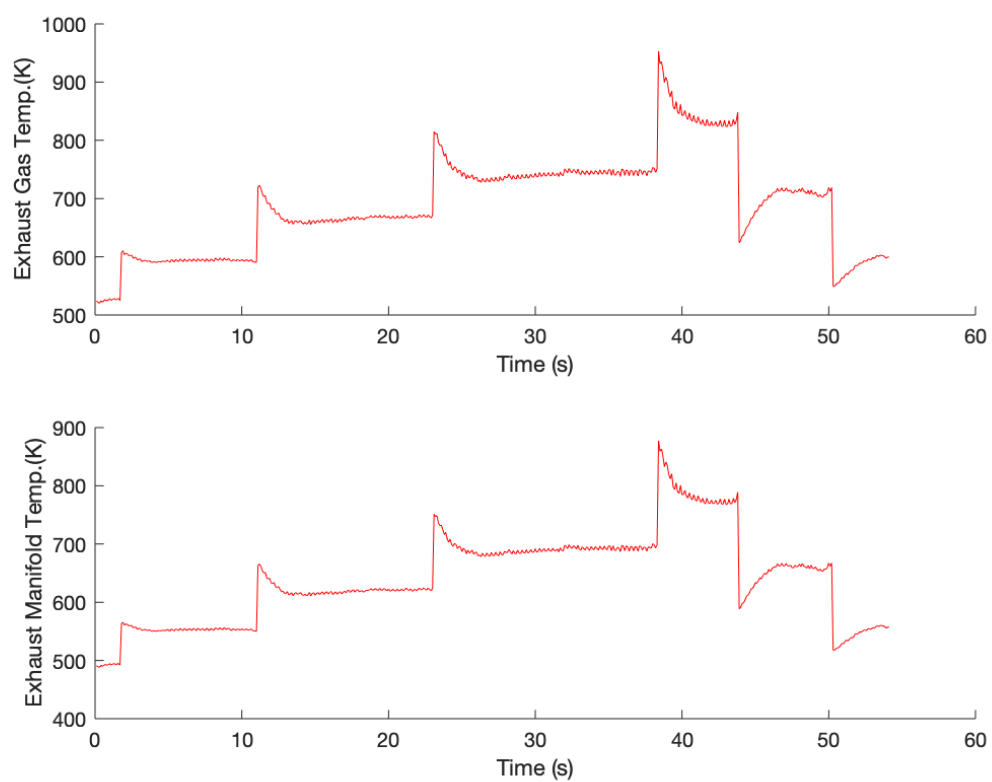


Figure 5.42: Data Set H, Exhaust Temperatures

Table 5.10: The mean absolute relative error % between the dynamic validation data and the diesel engine model simulation

Data Set:	F	G	H
$n_e(rpm)$	1800	1900	2000
p_{im}	3.76	1.47	1.3
p_{em}	1.34	0.11	0.947
$Lambda$	1.87	1.28	1.1
$Torque$	4.13	3.88	4.02

Note: It is rather interesting witnessing the effort that has been put in the design of the engine to minimize the exhaust pollutants (NOx). The lambda measurements are very low for a diesel engine, especially compared to one with no optimization for cleaner emissions, unoptimized engines run considerably leaner, having much higher Lambdas, therefore much higher Oxygen concentrations, therefore much higher combustion temperatures and much higher NOx emissions.

5.7 Conclusions

A mean value engine model of the Caterpillar C9.3 ACERT Diesel Engine was created. The goal was to create a model that is accurate in its results especially the Torque and Lambda outputs. Special care was given on the efficiency of the model in terms of speed, as short simulation times were needed. The intended applications of the model are system analysis, simulation and the development of model-based control systems. The task was very hard to complete due to the lack of many needed sensor based Data. Most important being the Waste Gate and EGR mass flow and the actuator specific parameters. In order to overcome these problems, the exhaust manifold reference pressure, that was measured in the Data Sets is used as input to the model to estimate the WG flow and WG dynamics, a derivative controller submodel acting as a "boost" controller was built, in order to quickly estimate the WG flow using the rest of the exhaust based mass flows as inputs. The resulted WG flow is fairly accurate considering the lack of sensor Data, topping at around 30% for heavy engine loads of around 80%. Also it is always positive meaning that it is a good and true physical estimation (considering the usage of an integrator it is easy for a value to become negative if it is needed to by the simulation)

Each specific subsystem is included with its individual Simulink implementation. Additionally the Simulink implementation for the complete engine model is also included in Appendix B.

Many models in the literature, which have similar complexities as the model that was developed in this instance, use two or three states for each control volume in order to also describe the Temperature dynamics. However this model uses only two states for the description of the manifolds, the pressure and the oxygen mass fraction. For the pressure estimation, the *filling and emptying* methodology is used, calculating the pressure as a result of incoming and outgoing flows. Since the use of EGR is implemented, the Lambda output from the system had to also consider the EGR oxygen concentration, therefore the oxygen fraction was implemented as

a second state in the manifolds. The EGR fraction was not available from the engine sensors, therefore, the EGR flow was estimated using the Venturi flow meter that is installed in the engine. Luckily, the pressure drop over the Venturi was logged by the ECU and therefore the flow was estimated. The error lies on the accurate diameters of the Venturi flow meter, since only the outer diameters could be measured. According to [3] estimation of more parameters describing i.e. the pressure drop over the intercooler or the pressure drops inside the piping system is not needed as they only present a marginal increase in accuracy. The cylinder out temperature was estimated through a model found in [17]. Its working principle is based on the Seiliger cycle and it relies on simulating the Diesel cycle as one. The exhaust gas temperature calculation is an iterative process with two iterations being the optimal number. However, using the model with two iterations deemed it inapplicable to the Simulink model as it was. To overcome this, a second model described in [16] was used on top of the calculated data from the first model to make its implementation achievable. The second model for the exhaust gas Temperature implies that for Diesel engines the T_e can be accurately described as a fraction of Lambda.

All the modeling equations are described for each subsystem. In order to minimize the modeling parameters, the flows and efficiencies were described using physical relationships and parametric models instead of Lookup Tables that need extensive tuning to be realistically useful in the Simulink environment. The parameters for each subsystem were initialized automatically using several Least Squares optimizations in Matlab. Afterwards, the unknown and uninitialized parameters (physical unknown parameters such as J_{turbo} or V_{im} , V_{em}) were tuned using the complete model and a Cost function that minimized the errors between the dynamic measurements and the dynamic models also through nonlinear least squares optimization. Finally, the static parameters of the individual submodels were automatically tuned using a second cost function that minimized the errors of the Dynamic measurements and models and the Stationary measurements and models that were available. Special attention was given on the importance of this whole process and the problems of singlehandedly using just one of them were explained.

The model was lastly validated using three Data Sets that were not used in its tuning process beforehand. The results were all accurate and the relative errors were below 4.13% for the complete simulations.

It should be noted that the model created in this thesis was the best possible implementation of all the modeling first principles that describe such a problem since similar solutions from the corresponding literature make use of numerous stationary and dynamic measurements (from sensors) that were not available on this engine (namely the EGR fraction x_{EGR} and the WG opening signal, as well as the actuator specifics), since its ECU is pre-tuned from the Factory for all its available operating points, making the installment of sensors such as a MAF redundant.

Appendices

.1 APPENDIX A

In this appendix of the thesis the Reverse integrator control model that was used for the Simulation of the $W_{wastegate}$ and Wastegate dynamics, since with the available data the modelling and the tuning of a separate Wastegate submodel was impossible. This lack of physical Data regarding the Wastegate flows and Wastegate orifice in particular led to the use of $p_{em,meas}$ as an input argument to the complete model in order to both control its stability and calculate an estimation of the $W_{wastegate}$. The fourth integrator, submodel, uses as inputs all the gas flows that enter the actual exhaust manifold and the measured value of the exhaust Manifold. It's output is an estimation of the $W_{wastegate}$ which also includes the dynamic stabilization factor of a well tuned Wastegate submodel. The output signal cannot be validated as there are no related measurements from the engine's or lab's sensors but it is thought to be fairly accurate and indicative of the actual value as it is always positive (except in some cases where the operating engine point is close to the idle speed and the load is near zero but again it is marginally negative) and on loads close to 70-80% it reaches 30-35% of the flow of W_{eo} , which of course would be increased for higher loads. This submodel represents a temporary solution to the problem of lack of Data. If in the future, more sensors are installed concerning the Wastegate physical system of the engine, the model's conversion to a model that doesn't need p_{em} as input would be easy and fairly simple.

The submodel is described in Fig.43:

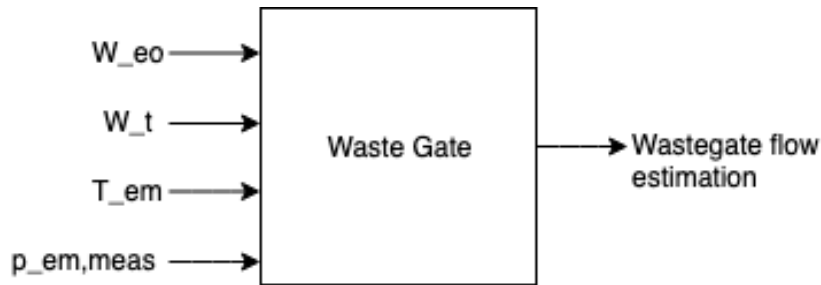


Figure 43: I/O arguments for the $W_{wastegate}$ estimation model

The simulink implementation of the Model is shown below:

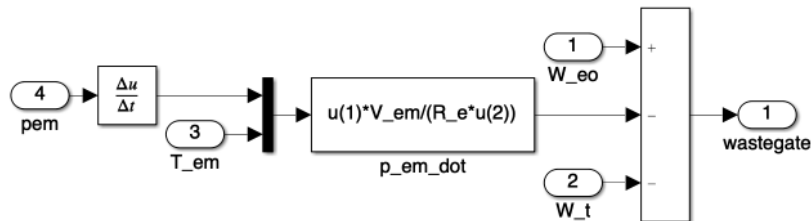


Figure 44: Simulink implementation for the $W_{wastegate}$ estimation model

Before the implementation of this submodel, the tuning of the complete model was not possible, since the p_{em} would always climb too high (which would physically

be true). After the realization that the engine did indeed use a Wastegate system, its implementation was mandatory.

.2 APPENDIX B

In this section, the implementation of the complete engine model is presented:

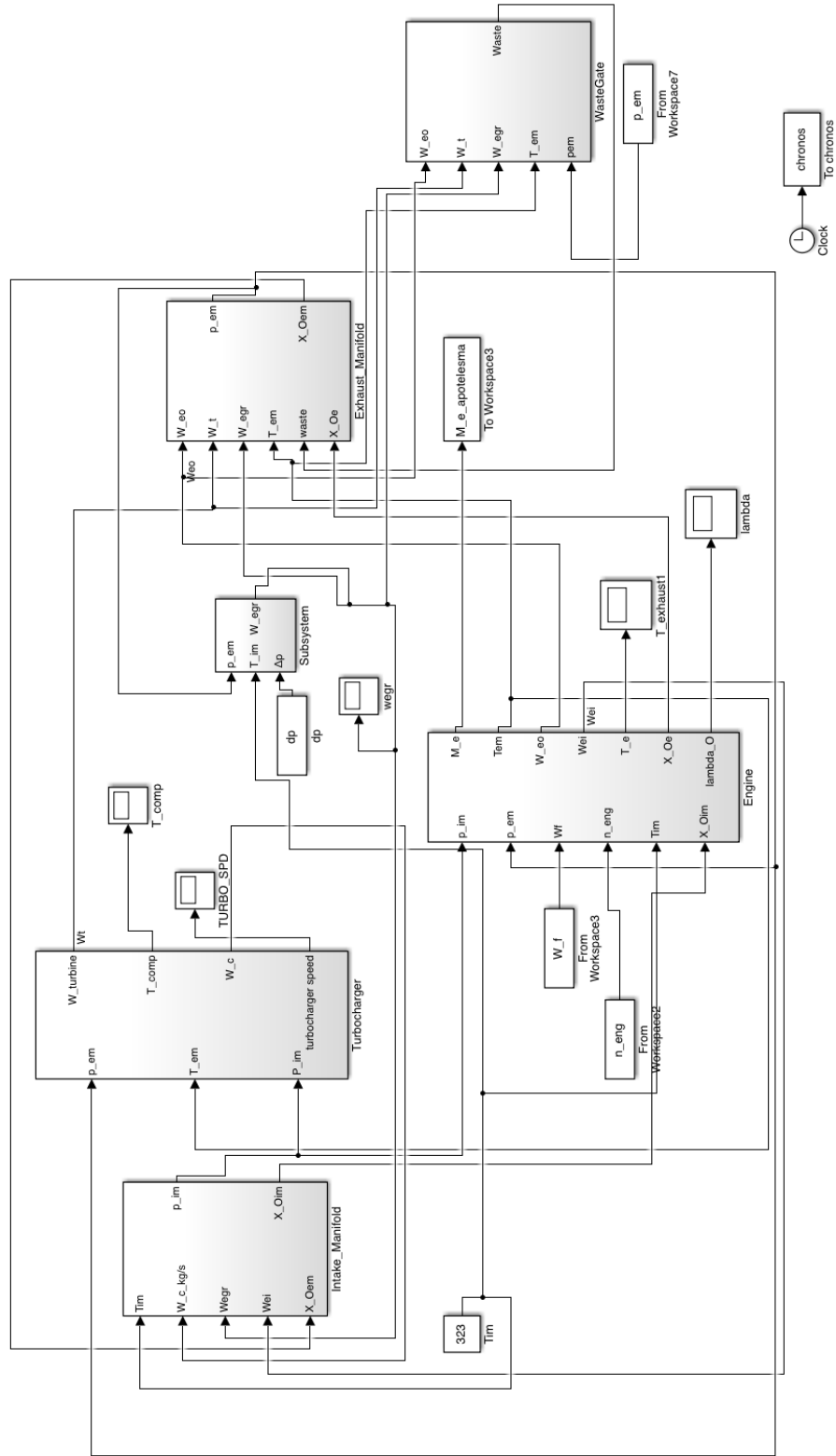


Figure 45: Simulink implementation of the complete model

Bibliography

- [1] C. H. O. Lino Guzzella, *Introduction to Modeling and Control of Internal Combustion Engine Systems*. Switzerland: Springer, 2010.
- [2] G. T. Systems, *Garrett Performance Product Catalog*. 2005.
- [3] J. Wahlström and L. Eriksson, “Modelling diesel engines with a variable geometry turbocharger and exhaust gas recirculation by optimization of model parameters for capturing non-linear system dynamics,” 2011.
- [4] I. K. Diana Yanakiev, “Engine and transmission modeling for heavy-duty vehicles,” 1995.
- [5] H. E., “A compact comprehensive model of a large turbocharged two-stroked engine,” 1986.
- [6] Caterpillar, “C9.3 acert industrial engine tier 4 interim/stage iiib,” 2010.
- [7] J. Asprion, *Optimal control of diesel engines modelling, numerical methods, and applications*. Zurich, Switzerland: ETH, 2013.
- [8] Winterbone, “A wholly dynamic model of turbocharged diesel engine for transfer function evaluation,” 1977.
- [9] Jensen, “Mean value modelling of a small turbocharged engine,” 1991.
- [10] Ledger, “Dynamic modelling of a turbocharged diesel engine,” 1971.
- [11] Mathworks, *Simulink User’s Guide*.
- [12] Ljung, *System Identification Toolbox*. Mathworks, 2015.
- [13] V. C., “The internal combustion engine: modelling, estimation and control issues,” 2001.
- [14] J. B. Heywood, *Internal combustion engine fundamentals*. 1988.
- [15] H. C. Huan Tu, “Modeling of a compressor’s performance map by fitting function methodology,” 2011.
- [16] A. A. Lino Guzzella, “Control of diesel engines,” 1998.
- [17] P. Skogtjarn, “Modelling of the exhaust gas temperature for diesel engines. master’s thesis,” 2002.

- [18] L. Eriksson, “Mean value models for exhaust system temperatures,” 2002.
- [19] A. Bjork, “Numerical methods for least squares problems,” 1996.