

Institutionen för systemteknik

Department of Electrical Engineering

Examensarbete

Tuning and Validation of an MVEM for a Turbocharged Gasoline Engine

Examensarbete utfört i Fordonssystem
vid Tekniska högskolan i Linköping
av

Stefan Tuve

LITH-ISY-EX--09/4167--SE

Linköping 2009



Linköpings universitet
TEKNISKA HÖGSKOLAN

Tuning and Validation of an MVEM for a Turbocharged Gasoline Engine

Examensarbete utfört i Fordonssystem
vid Tekniska högskolan i Linköping
av

Stefan Tuve

LITH-ISK-EX--09/4167--SE

Handledare: **Oskar Leufven**
ISK, Linköpings universitet

Examinator: **Lars Eriksson**
ISK, Linköpings universitet

Linköping, 30 January, 2009

	Avdelning, Institution Division, Department Vehicular Systems Department of Electrical Engineering Linköpings universitet SE-581 83 Linköping, Sweden	Datum Date 2009-01-30
	Språk Language <input type="checkbox"/> Svenska/Swedish <input checked="" type="checkbox"/> Engelska/English <input type="checkbox"/> _____	Rapporttyp Report category <input type="checkbox"/> Licentiatavhandling <input checked="" type="checkbox"/> Examensarbete <input type="checkbox"/> C-uppsats <input type="checkbox"/> D-uppsats <input type="checkbox"/> Övrig rapport <input type="checkbox"/> _____
URL för elektronisk version http://www.vehicular.isy.liu.se/ http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-16610		
Titel Title Tuning and Validation of an MVEM for a Turbocharged Gasoline Engine Författare Stefan Tuveesson Author		
Sammanfattning Abstract <p>One of the tools that is used when developing an engine control system is the Mean Value Engine Model (MVEM). The thesis examines the possibility of using an MVEM that originally was designed to model a SAAB B235R engine to model the considerably different GM Ecotec LNF engine.</p> <p>Engine data was measured at Vehicular Systems laboratory and the measurements were then used to tune the parameters of the MVEM's submodels.</p> <p>The majority of the components in the original MVEM could be tuned to a good or acceptable performance without any modifications. The submodels describing the port air mass flow, compressor air mass flow and turbine efficiency would all benefit from being modified or replaced, this was however only done for the turbine efficiency model, leaving the improvements of the remaining two submodels as future work.</p>		
Nyckelord Keywords measurements, parameters		

Abstract

One of the tools that is used when developing an engine control system is the Mean Value Engine Model (MVEM). The thesis examines the possibility of using an MVEM that originally was designed to model a SAAB B235R engine to model the considerably different GM Ecotec LNF engine.

Engine data was measured at Vehicular Systems laboratory and the measurements were then used to tune the parameters of the MVEM's submodels.

The majority of the components in the original MVEM could be tuned to a good or acceptable performance without any modifications. The submodels describing the port air mass flow, compressor air mass flow and turbine efficiency would all benefit from being modified or replaced, this was however only done for the turbine efficiency model, leaving the improvements of the remaining two submodels as future work.

Acknowledgments

I would like to thank my supervisor Oskar Leufven and my examiner Lars Eriksson for a lot of help and for making it possible for me to do this thesis.

I would also like to thank all friends and loved ones for their support.

Contents

1	Introduction	1
1.1	Background	1
1.2	The Engines	1
2	Mean Value Engine Model	3
2.1	Model Structure	3
2.2	Air Filter	3
2.3	Compressor	4
2.4	Intercooler	5
2.5	Throttle	6
2.6	Engine	7
2.6.1	Port air-mass flow	7
2.6.2	Torque	7
2.6.3	Exhaust mass flow	8
2.7	Turbine	9
2.8	Wastegate	10
2.9	Exhaust System	11
2.10	Turbocharger speed	11
2.11	Control Volumes	11
3	Measured Data and Maps	13
3.1	Non tunable parameters	16
4	Tuning and Validation	17
4.1	Air Filter	17
4.2	Compressor	19
4.3	Intercooler	21
4.4	Throttle	21
4.5	Engine	23
4.5.1	Port air mass flow	23
4.5.2	Torque	25
4.5.3	Exhaust mass flow	28
4.6	Turbine	29
4.7	Wastegate	30
4.8	Exhaust system	30

4.9 Turbocharger speed	31
4.10 Control volumes	31
4.11 Summary of submodels	33
4.12 Full model validation	34
4.12.1 Stationary validation	34
4.12.2 Dynamic validation	37
5 Conclusions and future work	39
5.1 Conclusions	39
5.2 Future work	40
Bibliography	41
A Nomenclature	43
B MVEMparameter	46
C MVEMturboparameter	52

Chapter 1

Introduction

This Masters Thesis examines the possibilities and problems of using an MVEM originally designed to be a good model of a specific engine to model a significantly different engine, and in the process develops a reusable MATLAB-file for the determination of the model parameters.

1.1 Background

In order to develop new and better engines that can fulfill our demands for cleaner and more efficient cars, a well functioning engine control unit (ECU) is a key element. One of the tools available to the developers of the ECU is a Mean Value Engine Model (MVEM), that simulates the function of a real world engine. This allows the developers to evaluate and tune their controllers without the need of running an expensive engine. As new engine models are being produced it's of course desirable to, if possible, use an already developed and tested MVEM, rather than develop a new one from scratch. The goal of this thesis is to use the MVEM from [1], in which it was tuned and validated for a SAAB B235R engine, to model the significantly different GM Ecotec LNF engine.

1.2 The Engines

The B235R is a 2.3 liter engine that produces a maximum power of 170 kW @ 6200 RPM [1] while the LNF is a 2.0 liter engine with a maximum power of 194 kW @ 5300 RPM [4]. Both the B235R and the LNF are 4 cylinder turbocharged spark ignited gasoline engines, the two largest differences in the operation of the engines are that the LNF engine uses what GM call SIDI and VVT.

SIDI, or Spark Ignited Direct Injection, means that the fuel is not injected in the intake manifold, but instead directly into the cylinders. There are several advantages with direct fuel injection, among other things it allows a more precise amount of fuel to be injected into the cylinder and since the air entering the

cylinder doesn't contain any fuel the engine can have a high valve overlap without unburnt fuel passing through the exhaust valve.

VVT, or Variable Valve Timing, means that the timing for the opening of the intake and exhaust valves is variable. This allows the ECU to continuously adjust the valve timing to the operating conditions, compared to fixed valve timing where it's set as a compromise to work reasonably well at all conditions.

The turbocharger of the LNF engine is also quite different from that of the B235R, as the turbine is a twin scroll turbine, meaning that the exhaust gases from the engine is led trough two separated channels until it reaches the actual turbine.

The LNF engine is installed in Vehicular Systems laboratory and has several differences compared to the engine installed in a car, for example different exhaust system and layout of intercooler installation. The engine is also controlled by an experimental control system, rendering the engine a different performance than the production engine.

Chapter 2

Mean Value Engine Model

The MVEM presented in [1] (see also: [3] and [2]) is a component based model, based on restrictions and control volumes. In this thesis, as in [1], it's assumed that there only is a positive mass flow through the engine components. In the actual Simulink model modifications are done so that negative flows also can be simulated.

2.1 Model Structure

The engine is made of components that, if you follow the air flow through the engine, are: air filter, compressor, intercooler, throttle, engine, turbine/wastegate and exhaust system. These components are modeled as restrictions and the pipes or manifolds between the components are modeled as control volumes.

2.2 Air Filter

The air filter is used to clean the air going to the engine, preventing dust and particles from damaging and wearing on the engine and its components.

As the air passes through the air filter there is a pressure loss that in [1] is described as:

$$\Delta p_{af} = p_a - p_{af} = \mathcal{H}_{af} \frac{T_a W_{af}^2}{p_a} \quad (2.1)$$

For a pressure drop $\Delta p_{af} < p_{lin_{af}}$ the resulting mass flow is linearized to avoid simulation problems for small Δp_{af} resulting in:

$$W_{af} = \begin{cases} \sqrt{\frac{\Delta p_{af} p_a}{\mathcal{H}_{af} T_a}} & \Delta p_{af} > p_{lin_{af}} \\ \sqrt{\frac{p_a}{\mathcal{H}_{af} T_a}} \frac{\Delta p_{af}}{\sqrt{p_{lin_{af}}}} & 0 \leq \Delta p_{af} \leq p_{lin_{af}} \\ 0 & \Delta p_{af} < 0 \end{cases} \quad (2.2)$$

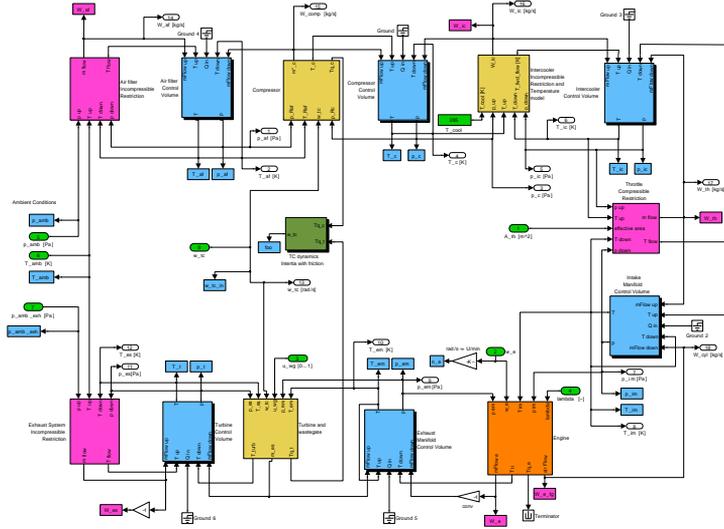


Figure 2.1. The Simulink implementation of the MVEM, showing its composition of restrictions and control volumes.

Parameters to tune	Known parameters	Inputs & Outputs
\mathcal{H}_{af} $p_{lin_{af}}$		p_a p_{af} T_a W_{af}

2.3 Compressor

In the compressor the density of the intake air is increased, allowing the engine to produce more power than if it had been naturally aspirated.

The compressor efficiency, air mass flow, temperature of outgoing air mass flow and the loading torque are in [1] modeled as:

Compressor efficiency:

$$\begin{aligned}
 W_{c_{corr}} &= W_c \frac{\sqrt{T_{af}/T_{std}}}{p_{af}/p_{std}} \\
 Q &= \begin{bmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{bmatrix} \\
 \chi &= \begin{bmatrix} W_{c_{corr}} - W_{c_{\eta_{max}}} \\ 1 + \sqrt{\Pi_c - 1} - \Pi_{c_{\eta_{max}}} \end{bmatrix} \\
 \eta_c &= \eta_{c_{max}} - \chi^T Q \chi
 \end{aligned} \tag{2.3}$$

Compressor air mass flow:

$$\begin{aligned}
 \Phi_c &= \frac{W_c}{\frac{p_{af}}{R_a T_{af}} \frac{\pi}{4} D_c^2 U_c} \\
 U_c &= \omega_{tc} \frac{D_c}{2} \\
 \Psi_c &= c_{p_a} T_{af} \frac{\Pi_c^{\frac{\gamma_a-1}{\gamma_a}} - 1}{\frac{1}{2} U_c^2} \\
 \Pi_c &= \frac{p_c}{p_{af}} \\
 \Psi_c^2 K_1 + \Phi_c^2 K_2 &= 1
 \end{aligned} \tag{2.4}$$

Compressor temperature out:

$$T_c = T_{af} \left(1 + \frac{\Pi_c^{\frac{\gamma_a-1}{\gamma_a}} - 1}{\eta_c} \right) \tag{2.5}$$

Compressor torque:

$$T_{q_c} = \frac{c_{p_a} W_c (T_c - T_{af})}{\omega_{tc}} \tag{2.6}$$

Parameters to tune	Known parameters	Inputs & Outputs
a_{11}, a_{12}, a_{22}	T_{std}	W_c
$W_{c_{\eta_{max}}}$	p_{std}	p_{af}
$\Pi_{c_{\eta_{max}}}$	R_a	p_c
$\eta_{c_{max}}$	c_{p_a}	ω_{tc}
K_1, K_2	γ_a	T_{af}
	D_c	p_{af}
		η_c

2.4 Intercooler

When the pressure is increased in the compressor, so is the temperature. In order to lower the temperature of the air, an intercooler is installed between the compressor and throttle. The most common type of intercooler used in automotive engines, and that is used in the LNF, is a cross-flow air to air heat exchanger in which the air going through the intercooler (W_{ic}) is cooled by an airflow of cooling air (W_{cool} , with temperature T_{cool}).

There are two main reasons for lowering the air temperature. One is to decrease the risk of knocking as the risk of knocking is increased with higher intake air temperatures. The other is to increase the density of intake air.

The pressure drop and temperature change over the intercooler are in [1] described as:

Intercooler temperature change:

$$\begin{aligned} T_{ic_{out}} &= \max(T_{cool}, T_c + \epsilon(T_c - T_{cool})) \\ \epsilon &= a_0 + a_1 \left(\frac{T_c + T_{cool}}{2} \right) + a_2 W_{ic} + a_3 \frac{W_{ic}}{W_{cool}} \end{aligned} \quad (2.7)$$

Intercooler pressure drop:

$$\Delta p_{ic} = p_c - p_{ic} = \mathcal{H}_{ic} \frac{T_c W_{ic}^2}{p_c} \quad (2.8)$$

For a pressure drop $\Delta p_{ic} < p_{lin_{ic}}$ the resulting mass flow is linearized to avoid simulation problems for small Δp_{ic} resulting in:

$$W_{ic} = \begin{cases} \sqrt{\frac{\Delta p_{ic} p_c}{\mathcal{H}_{ic} T_c}} & \Delta p_{ic} > p_{lin_{ic}} \\ \sqrt{\frac{p_c}{\mathcal{H}_{ic} T_c}} \frac{\Delta p_{ic}}{\sqrt{p_{lin_{ic}}}} & 0 \leq \Delta p_{ic} \leq p_{lin_{ic}} \\ 0 & \Delta p_{ic} < 0 \end{cases} \quad (2.9)$$

Parameters to tune	Known parameters	Inputs & Outputs
a_0, a_1, a_2, a_3		W_{cool}
\mathcal{H}_{ic}		W_{ic}
$p_{lin_{ic}}$		T_c
		T_{cool}
		p_c
		p_{ic}

2.5 Throttle

The throttle is used to control air flow. In older cars the throttle plate was directly connected to the accelerator pedal, today it's controlled via the ECU.

The air flow over the throttle depends on the effective opening area of the throttle plate and the pressure difference over the throttle. This is in [1] and [5] described as:

$$W_{im}(\alpha, p_{im}, p_{ic}, T_{ic}) = \frac{p_{ic}}{\sqrt{R_a T_{ic}}} \Psi(\Pi) C_d(\Pi) C_d(A(\alpha)) A(\alpha) \quad (2.10)$$

$$\Pi = \begin{cases} \frac{p_{im}}{p_{ic}} & p_{im} < p_{ic} \\ 1 & \text{otherwise} \end{cases} \quad (2.11a)$$

$$\Psi^*(\Pi) = \sqrt{\frac{2\gamma}{\gamma-1} \left(\Pi^{\frac{2}{\gamma}} - \Pi^{\frac{\gamma+1}{\gamma}} \right)} \quad (2.11b)$$

$$\Psi(\Pi) = \begin{cases} \Psi^* \left(\left(\frac{2}{\gamma+1} \right)^{\frac{\gamma}{\gamma-1}} \right) & 0 < \Pi \leq \left(\frac{2}{\gamma+1} \right)^{\frac{\gamma}{\gamma-1}} \\ \Psi^*(\Pi) & \left(\frac{2}{\gamma+1} \right)^{\frac{\gamma}{\gamma-1}} < \Pi \leq \Pi_{lin_{th}} \\ \frac{\Psi^*(\Pi_{lin_{th}})}{\Pi_{lin_{th}} - 1} & \Pi_{lin_{th}} < \Pi \leq 1 \end{cases} \quad (2.11c)$$

$\Psi(\Pi) C_d(\Pi)$ is in [1] replaced with $\Psi_\gamma(\Pi)$ which is Eq. 2.11 in which the physical value of γ is replaced with a value determined so that 2.10 becomes:

$$W_{im}(\alpha, p_{im}, p_{ic}, T_{ic}) = \frac{p_{ic}}{\sqrt{R_a T_{ic}}} \Psi_\gamma(\Pi) C_d(A(\alpha)) A(\alpha) \quad (2.12)$$

Parameters to tune	Known parameters	Inputs & Outputs
γ Π_{linth}	R_a	W_{im} p_{ic} p_{im} T_{ic} $C_d(A(\alpha)) A(\alpha)$

2.6 Engine

The air flows through the intake port into the cylinder where it's compressed and fuel is added and burnt after which the gases flow through the exhaust port out into the exhaust manifold.

The engine model consists of three parts:

- Port air-mass flow. The mass-flow from the intake manifold to the cylinder.
- Torque. The produced engine torque.
- Exhaust-mass flow. The mass-flow and temperature of the exhaust gases flowing from the engine to the exhaust-manifold.

2.6.1 Port air-mass flow

The port air-mass flow presented in [1] is:

$$W_{cyl} = p_{im} C_1 \frac{1}{1 + \frac{1}{\lambda \left(\frac{A}{F}\right)_s}} \frac{r_c - \left(\frac{p_{em}}{p_{im}}\right)^{\frac{1}{\gamma_a}}}{r_c - 1} V_d \cdot \frac{N_e}{R_a (T_{im} - C_2 \frac{\lambda - 1}{\lambda^2})} \quad (2.13)$$

Parameters to tune	Known parameters	Inputs & Outputs
C_1, C_2	R_a γ_a $\left(\frac{A}{F}\right)_s$ V_d r_c	W_{cyl} p_{im} p_{em} T_{im} N_e λ

2.6.2 Torque

The torque produced by the engine is in [1] modeled as:

$$Tq_e = \frac{BMEP \cdot V_d}{4\pi} \quad (2.14)$$

$$BMEP = IMEP - PMEP - FMEP \quad (2.15)$$

$$IMEP = W_{cyl} \frac{60 \cdot 2}{N} \frac{q_{HV}}{\lambda \left(\frac{A}{F}\right)_s} \frac{\min(\lambda, 1)}{V_d} \eta_e \quad (2.16)$$

$$PMEP = p_{em} - p_{im} \quad (2.17)$$

$$mps = 2 \frac{N}{60} \text{Stroke} \quad (2.18)$$

$$(2.19)$$

$$FMEP = \zeta_{aux} \sqrt{\frac{75}{1000 \cdot \text{Bore}}} \cdot \left((0.464 + 0.0072mps^{1.8}) \Pi_{bl} \cdot 10^5 + \underbrace{0.0215 BMEP}_{BMEP \approx W_{cyl} \frac{60 n_r}{N V_d} C_{Tq1} + C_{Tq2}} \right) \quad (2.20)$$

Parameters to tune	Known parameters	Inputs & Outputs
η_e	V_d	Tq_e
ζ_{aux}	n_r	N_e
C_{Tq1}, C_{Tq2}	$\left(\frac{A}{F}\right)_s$	p_{em}
	Bore	p_{im}
	Stroke	λ
	q_{HV}	W_{cyl}
	Π_{bl}	

2.6.3 Exhaust mass flow

The exhaust gas mass flow is the sum of the air mass flow and the fuel mass flow to the engine, and is in [1] described as:

$$W_e = W_{cyl} + W_f = W_{cyl} \left(1 + \frac{1}{\lambda \left(\frac{A}{F}\right)_s} \right) \quad (2.21)$$

The temperature of the exhaust gases after the exhaust manifold are in [1] described as:

$$T_e = T_{e_0} + \Delta T_{e_{max}} W_e \quad (2.22)$$

$$T_{em} = T_a + (T_e - T_a) e^{-\frac{h_{tot} \pi D_{em} l_{em} n_{cyl}}{W_e c_{peg}}} \quad (2.23)$$

$$h_{exh_{int}} = HT_{exh_0} \lambda_{exh} \left(\frac{4}{\pi} \frac{W_e}{\mu_{exh} n_{cyl} D_{em}} \right)^{HT_{exh_1}} \frac{1}{D_{em}} \quad (2.24)$$

$$h_{tot} = \frac{1}{\frac{1}{h_{exh_{int}}} + \frac{1}{h_{exh_{ext}}}} \quad (2.25)$$

Parameters to tune	Known parameters	Inputs & Outputs
T_{e_0}	$\left(\frac{A}{F}\right)_s$	W_e
$\Delta T_{e_{max}}$	D_{em}	W_{cyl}
$h_{exh_{ext}}$	l_{em}	T_a
	n_{cyl}	T_{em}
	c_{peg}	λ
	HT_{exh_0}, HT_{exh_1}	
	λ_{exh}	
	μ_{exh}	

2.7 Turbine

The turbine uses the exhaust gases from the engine to power the compressor that is mounted on the same axis as the turbine.

In [1] the turbine gas mass flow, efficiency, gas flow temperature out and produced torque are described as:

Turbine mass flow:

$$\begin{aligned} \Pi_t &= \frac{p_t}{p_{em}} \\ W_{t_{corr}} &= W_t \frac{\sqrt{T_{em}}}{p_{em}} \\ W_{t_{corr}} &= \begin{cases} k_1 \sqrt{1 - \Pi_t^{k_2}} & \Pi_t^{k_2} \leq 1 \\ 0 & otherwise \end{cases} \end{aligned} \quad (2.26)$$

Turbine efficiency:

$$BSR = \frac{D_t}{2} \frac{\omega_{tc}}{\sqrt{2c_{peg} T_{em} \left(1 - \left(\frac{1}{\Pi_t} \right)^{\frac{1-\gamma_{eg}}{\gamma_{eg}}} \right)}} \quad (2.27)$$

$$\eta_t = \eta_{t_{max}} \left(1 - \left(\frac{BSR - BSR_{\eta_{t_{max}}}}{BSR_{\eta_{t_{max}}}} \right)^2 \right) \quad (2.28)$$

Turbine temperature out: In the model the combined mechanical and turbine efficiency is used as turbine efficiency. This will however cause an error in the turbine temperature as that model should use a turbine efficiency without the included mechanical efficiency.

$$T_t = T_{em} - T_{em} \left(1 - \Pi_t^{\frac{\gamma_{eg}-1}{\gamma_{eg}}} \right) \eta_t \quad (2.29)$$

Turbine torque:

$$Tq_t = \frac{\eta_t W_t T_{em} \left(1 - \Pi_t^{\frac{\gamma_{eg}-1}{\gamma_{eg}}} \right)}{\omega_{tc}} \quad (2.30)$$

In [6] an alternative turbine efficiency model is presented:

$$\eta_t = a_0 + a_1 N_{tc} + (a_2 + a_3 N_{tc}) \cdot BSR + (a_4 + a_5 N_{tc}) \cdot BSR^2 \quad (2.31)$$

Parameters to tune	Known parameters	Inputs & Outputs
k_1, k_2 $\eta_{t_{max}}$ $BSR_{\eta_{t_{max}}}$ $a_0, a_1, a_2, a_3, a_4, a_5$	D_t c_{peg} γ_{eg}	p_t p_{em} T_{em} W_t ω_{tc}, N_{tc} η_t T_t Tq_t

2.8 Wastegate

The mass flow through the wastegate is in [1] described as:

$$W(u_{wg}, p_t, p_{em}, T_{em}) = \frac{p_{em}}{\sqrt{R_{eg} T_{em}}} \Psi(\Pi) C_d A_{wg_{max}} u_{wg} \quad (2.32)$$

$$\Pi = \begin{cases} \frac{p_t}{p_{em}} & p_t < p_{em} \\ 1 & \text{otherwise} \end{cases} \quad (2.33a)$$

$$\Psi^*(\Pi) = \sqrt{\frac{2\gamma_{eg}}{\gamma_{eg}-1} \left(\Pi^{\frac{2}{\gamma_{eg}}} - \Pi^{\frac{\gamma_{eg}+1}{\gamma_{eg}}} \right)} \quad (2.33b)$$

$$\Psi(\Pi) = \begin{cases} \Psi^* \left(\left(\frac{2}{\gamma_{eg}+1} \right)^{\frac{\gamma_{eg}}{\gamma_{eg}-1}} \right) & 0 < \Pi \leq \left(\frac{2}{\gamma_{eg}+1} \right)^{\frac{\gamma_{eg}}{\gamma_{eg}-1}} \\ \Psi^*(\Pi) & \left(\frac{2}{\gamma_{eg}+1} \right)^{\frac{\gamma_{eg}}{\gamma_{eg}-1}} < \Pi \leq \Pi_{lin_{wg}} \\ \frac{\Psi^*(\Pi_{lin_{wg}})}{\Pi_{lin_{wg}}^{-1}} & \Pi_{lin_{wg}} < \Pi \leq 1 \end{cases} \quad (2.33c)$$

Parameters to tune	Known parameters	Inputs & Outputs
C_d $\Pi_{lin_{wg}}$	R_{eg} $A_{wg_{max}}$ γ_{eg}	p_{em} T_{em} u_{wg} p_t

2.9 Exhaust System

As the exhaust gases go through the catalyst, mufflers and pipes of the exhaust system there is a pressure drop described in [1] as:

$$T_{esin} = \frac{T_t W_t + T_{em} W_{wg}}{W_t + W_{wg}} \quad (2.34)$$

$$W_{es} = W_t + W_{wg} \quad (2.35)$$

$$\Delta p_{es} = p_t - p_a = \mathcal{H}_{es} \frac{T_{esin} W_{es}^2}{p_t} \quad (2.36)$$

$$W_{es} = \begin{cases} \sqrt{\frac{\Delta p_{es} \mathcal{H}_{es} p_t}{T_{esin}}} & \Delta p_{es} > p_{lines} \\ \sqrt{\frac{p_t}{\mathcal{H}_{es} T_{esin}}} \frac{\Delta p_{es}}{\sqrt{p_{lines}}} & 0 \leq \Delta p_{es} \leq p_{lines} \\ 0 & \Delta p_{es} < 0 \end{cases} \quad (2.37)$$

Parameters to tune	Known parameters	Inputs & Outputs
\mathcal{H}_{es} p_{lines}		T_t T_{em} W_{wg} W_t W_{es} T_{esin} p_t p_a

2.10 Turbocharger speed

The dynamics of the turbocharger is in [1] described as:

$$I_{tc} \frac{d\omega_{tc}}{dt} = (Tq_t - Tq_c - \omega_{tc} \cdot \text{friction}_{tc}) \quad (2.38)$$

The friction_{tc} parameter is small as the friction of the turbocharger already is a part of the turbine efficiency, the main purpose of the parameter is to add stability to the model.

Parameters to tune	Known parameters	Inputs & Outputs
I_{tc} friction_{tc}		Tq_t Tq_c ω_{tc}

2.11 Control Volumes

There are six control volume's representing the pipes and manifolds between the engine components:

- V_{af} , Air filter to compressor.
- V_c , Compressor to intercooler.
- V_{ic} , Intercooler to Throttle.
- V_{im} , Throttle to Engine.
- V_{em} , Engine to turbine.
- V_{es} , Turbine to the end of the exhaust system.

All control volumes emptying and filling are in [1] described as:

$$m = \frac{pV}{RT} \quad (2.39a)$$

$$\frac{dT}{dt} = \frac{1}{mc_v} (W_{in}c_v(T_{in} - T)) + \frac{1}{mc_v} (R(T_{in}W_{in} - TW_{out}) + \dot{Q}) = f_{\frac{dT}{dt}}(p, T, T_{in}, W_{in}, W_{out}, m, R, v, c_v, \dot{Q}) \quad (2.39b)$$

$$\frac{dp}{dt} = \frac{RT}{V} (W_{in} - W_{out}) + \frac{mR}{v} \frac{dT}{dt} = f_{\frac{dp}{dt}}(p, T, T_{in}, W_{in}, W_{out}, m, R, v, f_{\frac{dT}{dt}}) \quad (2.39c)$$

Chapter 3

Measured Data and Maps

An engine map with 135 points has been measured from the LNF Engine in Vehicular Systems Engine Test Cell. During the measurements the engine used an experimental control system that used a very retarded spark timing for the points with high intake manifold pressure, this causing fuel to, in these cases, be burnt in the exhaust. This resulted in a number of phenomena that the model cannot describe, such as higher temperature after the turbine than in the exhaust manifold. This also prevented measurements for higher power output, in the mapped data the maximum power output is 73kW, i.e. the mapped data's maximum air flow is less than half of the air flow when the LNF engine is running at its maximum rated power of 194kW.

Because of the fuel being burnt in the exhaust, the values from the “hot” side should be considered uncertain and no proper validation of those models have been done, as in those cases only a small part of the available data has been used.

A smaller map in which the throttle is at a constant value has been created to tune the throttle model.

For the tuning and validation of the compressor and turbine models, a supplied turbocharger map has been used. The turbocharger map also contained values for turbine and compressor efficiency.

The states and inputs that have been measured in the engine map are:

- N_e , Engine speed [rpm]
Taken from the dyno-bench.
- Tq_e , Engine torque [Nm]
Taken from the dyno-bench.
- λ , Normalized air/fuel ratio [-]
Measured with lambda sensor located between the turbine and catalyst.
- p_a , Ambient pressure [Pa]
Measured with the LFE3.

- T_a , Ambient temperature [K]
Measured with a temperature sensor placed near the intake of the LFE3.
- p_{af} , Pressure after air filter [Pa]
Measured with a pressure sensor connected to the pipe going from the air filter box to the compressor, approx. halfway between the two components.
- T_{af} , Temperature after air filter [K]
Not measured. The temperature change over the air filter is considered negligible and the temperature T_a is used as T_{af} .
- p_c , Pressure after compressor [Pa]
Measured with a pressure sensor connected to the inlet of the intercooler.
- T_c , Temperature after compressor [K]
Measured with a temperature sensor placed next to where the p_c sensor is connected in the inlet to the intercooler.
- p_{ic} , Pressure after intercooler [Pa]
Measured with a pressure sensor connected to the outlet of the intercooler.
- T_{ic} , Temperature after intercooler [K]
Measured with a temperature sensor placed next to where the p_{ic} sensor is connected in the outlet of the intercooler.
- p_{im} , Intake manifold pressure [Pa]
Measured with a pressure sensor mounted on the intake manifold.
- T_{im} , Intake manifold temperature [K]
Measured with a temperature sensor placed in the intake manifold.
- p_{em} , Exhaust manifold pressure [Pa]
Measured with a pressure sensor connected to the beginning of one of the exhaust manifold pipes very close to the engine block.
- T_{em} , Exhaust manifold temperature [K]
Measured with a temperature sensor placed next to where the p_{em} sensor is connected in the beginning of one of the exhaust manifold pipes.
- p_t , Pressure after turbine [Pa]
Measured with a pressure sensor connected to the pipe shortly after the turbine.
- T_{esin} , Temperature after turbine [K]
Measured with a temperature sensor placed next to where the p_t sensor is connected shortly after the turbine.
- N_{tc} , Turbocharger speed [rpm]
Measured with a sensor mounted on the compressor.

- W_{af} , Air mass flow [kg/s]

Measured with the LFE3. As the measurements are steady state and there are no additional air intakes or outlets the air mass flow is the same through all components of the engine (if the wastegate is open the air mass flow is divided into two parallel flows through the turbine and wastegate).

All temperature sensors are type K thermocouples, all pressure sensors except for the ambient pressure are Kistler Kristall sensors. A laminar air mass flow sensor, LFE3, was connected on a pipe to the air filter box of the LNF. The signals from these sensors were measured using a Hewlett-Packard HP E1415A measurement system with the engine connected to a Schenk Dynas dynamometer. The engine was controlled with a MATLAB/Simulink interface via a dSPACE rapid pro box using an experimental prototype engine control system.

The exhaust manifold pressure and temperature, p_{em} and T_{em} , are on the test engine measured in the beginning of the exhaust manifold near the engine block and not just before the turbine in the end of the manifold as it is assumed in the model and that was the case with the B235R engine. As T_{em} is the temperature upstream the turbine, this will cause the temperature change in the exhaust manifold not to be described by the exhaust-mass flow temperature change model described in section 2.6.3 but instead affect the turbine model in section 4.6.

The fuel used during the measurement was normal commercial gasoline, the same fuel was used for all measurements but the exact values of the air to fuel ratio, $(\frac{A}{F})_s$, and energy contents, q_{HV} , is not known.

3.1 Non tunable parameters

There are a number of engine and gas parameters with known physical values that are measured or taken from data sheets with the exception of T_{std} , p_{std} that are taken from the turbo map and HT_{exh_0} , HT_{exh_1} , μ_{exh} and λ_{exh} that are taken from [1].

V_d	= 0.001998	Swept volume	$[m^3]$
r_c	= 9.2	Ratio of compression	[-]
n_{cyl}	= 4	Number of cylinders	[-]
Stroke	= 0.086	Stroke	[m]
Bore	= 0.086	Bore	[m]
q_{HV}	= $4.4 \cdot 10^7$	Specific energy contents of fuel	$[J/kg]$
$(\frac{A}{F})_s$	= 15.1	Stoichiometric air/fuel ratio	[-]
c_{peg}	= 1256.7	Specific heat, exhaust gas	$[J/(kg \cdot K)]$
γ_{eg}	= 1.3	Ratio of specific heat (c_p/c_v)	[-]
R_{eg}	= 290	Universal gas constant, exh. gas	$[J/(kg \cdot K)]$
c_{pa}	= 1005.2	Specific heat, air	$[J/(kg \cdot K)]$
γ_a	= 1.4	Ratio of specific heat (c_p/c_v)	[-]
R_a	= 287.2	Universal gas constant, air	$[J/(kg \cdot K)]$
D_{em}	= 0.04	Exhaust manifold pipe diameter	[m]
l_{em}	= 0.42	Exhaust manifold pipe length	[m]
μ_{exh}	= $4.23 \cdot 10^{-5}$	Dynamic viscosity of exh. gas	$[kg/(m \cdot s)]$
λ_{exh}	= $6.85 \cdot 10^{-2}$	Thermal conductivity of exh. gas	$[W/m \cdot K]$
D_t	= 0.05	Turbine diameter	[m]
D_c	= 0.056	Compressor diameter	[m]
p_{std}	= 98100	Correction pressure	[Pa]
T_{std}	= 293	Correction temperature	[K]
HT_{exh_0}	= 0.26	Heat transfer parameter	[-]
HT_{exh_1}	= 0.6	Heat transfer parameter	[-]
Π_{bl}	= $\max\left(\frac{p_{im}}{p_a}\right)$	Engine boost layout ¹	[-]

¹The boost layout is an engine parameter that is calculated from the engines maximum boost pressure

Chapter 4

Tuning and Validation

Because of the problems with the exhaust gas temperatures it has not been possible to do a good validation of the turbo on the full model. The models on the hot side were all tuned with the same uncertain data, so even those parts of the MVEM on the hot side of the engine that follows the mapped data quite well should be recalculated when better measurements are available.

As only one set of data was available the same data was used for both tuning and validation.

The full implementation of the parameter calculation is found in appendix B and C. Appendix B contains MATLAB code for calculations made using data from the engine map and appendix C contains the MATLAB code used to calculate parameters using the turbo map data.

For all modeled values the correlation of the modeled value to the measured is calculated using the *corr2* function in MATLAB, the mean and largest absolute relative error are taken from the tool Data statistics after plotting the absolute error with the plot function in MATLAB.

The plots have all been slightly scaled in order not to reveal any true engine parameters.

4.1 Air Filter

The pressure head loss parameter over the filter, \mathcal{H}_{af} , in the air filter model described by Eq. 2.1 is calculated using the least square method in MATLAB with data taken from the engine map. In the mapped data, the pressure drop over the air filter seems to be linear unlike the model (see Fig. 4.1). It's clear when looking at the data that something is wrong as the pressure difference is negative for low airflows.

The linearity can be caused by a number of different things, the pressure drop over the air filter can in fact be linear, there can be some problems in the sensor signal and conversion during the measurement, the placement of the sensors may not be ideal and dynamic pressure effects may cancel the nonlinearity of the pressure drop. It's also possible that the pressure drop would be less linear for higher

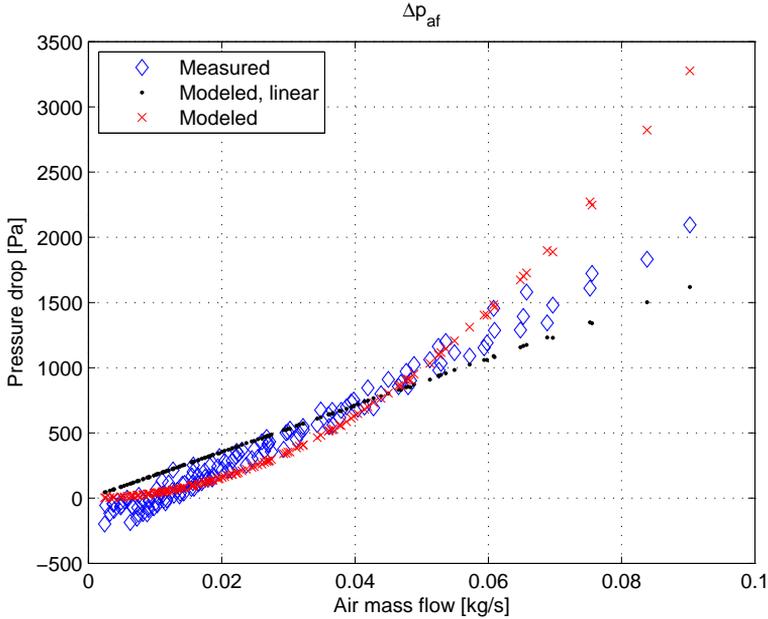


Figure 4.1. Pressure drop over the air filter. The measured drop, the modeled and linearized modeled drop ($p_{lin_{af}}$ set to 800 Pa). Notice the linearity and negative values of the measured pressure drop.

airflows than those mapped.

The negative pressure difference can be attributed to a number of things, most likely is however a sensor bias. In the rest of the validation and parameter tuning it's assumed that there is a sensor bias. The value of the bias is unknown but since it's impossible to have a negative pressure drop it's likely to be larger than the largest measured negative pressure drop. To compensate for this the ambient pressure, p_a , in the data has been modified by simply adding a value to it so that the minimum pressure drop is just above zero.

The linearization that is done to prevent simulation problems for low pressure differences will, because of the linear pressure drop in this case, result in a better fit than the nonlinear pressure drop. By manually setting \mathcal{H}_{af} and then calculating $p_{lin_{af}}$ using *lsqcurvefit* in MATLAB, the linearized model is tuned to fit the measured data. The value of \mathcal{H}_{af} only affects the behavior of the model for pressure drops larger than $p_{lin_{af}}$, so by setting \mathcal{H}_{af} to a low value the pressure drop in all of the measured points is below that of the calculated $p_{lin_{af}}$. This makes the resulting model (Eq. 2.2) for the measured air flows to be described by the linearized model (see Fig. 4.2). The correlation between the modeled and measured pressure drop is 99.1%. The mean relative error is 40.2% and the largest relative error is 240%. The value for $\frac{\max|e_i|}{\text{mean}(\Delta p_{af_i})}$, where i is the sample index and e is the error, is 0.36. The very large relative errors mainly occur for low pressure

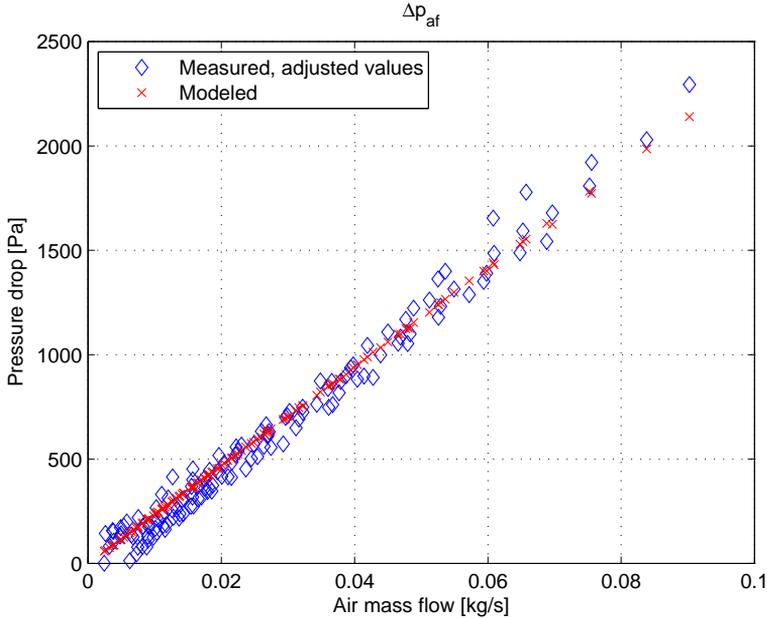


Figure 4.2. Pressure drop over the air filter. The measured drop with adjusted p_a and the modeled pressure drop tuned using the adjusted data and a calculated $p_{lin_{af}}$.

drops where a small absolute error leads to large relative error.

4.2 Compressor

The compressor efficiency model parameters in Eq. 2.3, $\Pi_{c_{\eta_{max}}}$, $W_{c_{\eta_{max}}}$, $\eta_{c_{max}}$ and a_{11}, a_{12}, a_{22} are calculated with MATLAB's *lsqcurvefit* using data from the turbo map. The resulting model gives a reasonable fit to the mapped data. The modeled and the mapped efficiency shown in Fig. 4.3, have a correlation of 95.2% and a mean relative error of 6.4%. The error is smallest for the medium compressor speeds in the turbo map and is clearly much larger for the higher and lower compressor speeds (see Fig. 4.3). It's possible that a modified model for the compressor efficiency that also have a dependency of the turbo speed would perform considerably better, this however has not been investigated in this thesis.

The air mass flow parameters K_1 and K_2 from Eq. 2.4 are calculated with MATLAB's *lsqcurvefit* using turbo map data. The resulting model gives a poor fit with a correlation of 90.1% and a mean relative error of 23%, see Fig. 4.4.

One of the reasons for the poor fit is that the model doesn't capture the fact that the mass flow - pressure ratio curve goes from a positive to a negative slope but instead describes the curve with only a negative or zero slope. It's also easy to think that the fit is better than it really is when looking at the plots, as it's easy to just look at the modeled point nearest the measured instead of that with

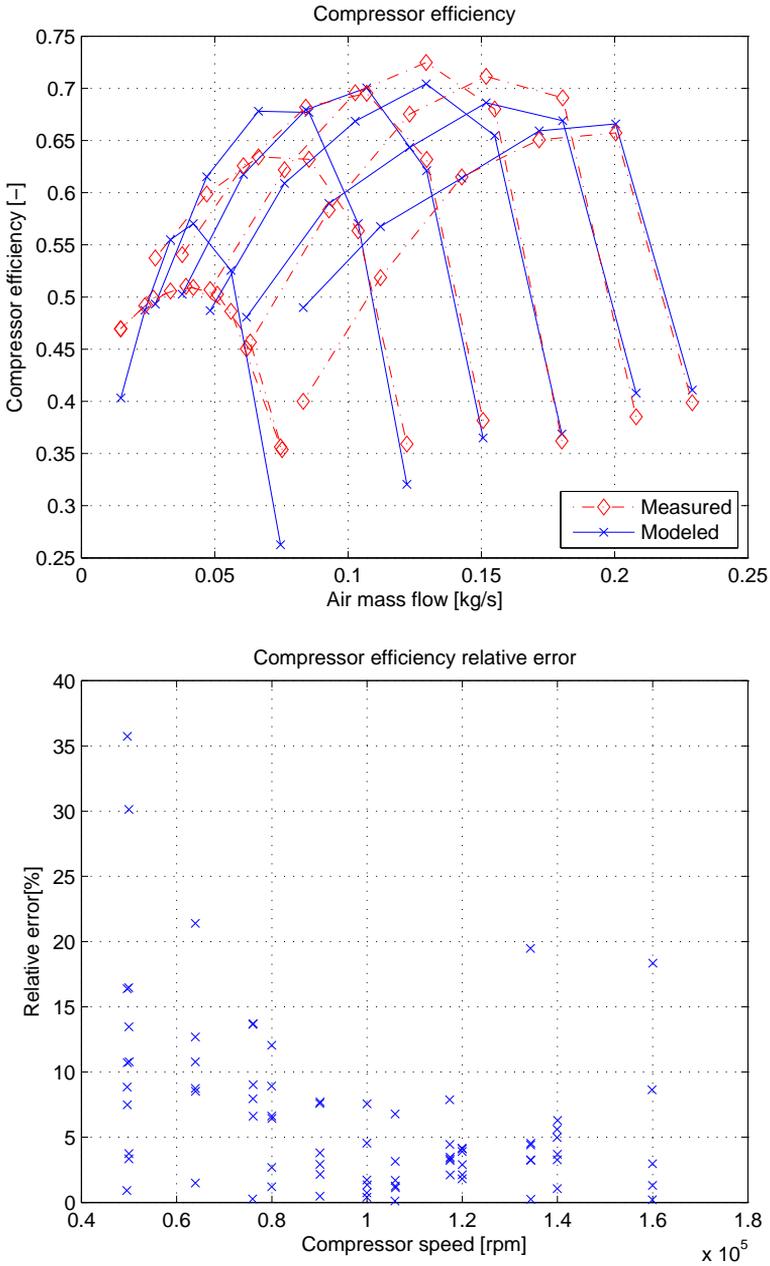


Figure 4.3. *Top:* The compressor efficiency gives a reasonable fit to measured data. *Bottom:* The relative error is clearly larger for the higher and lower mapped compressor speeds.

the same pressure ratio.

4.3 Intercooler

The pressure head loss parameter over the intercooler \mathcal{H}_{ic} described in Eq. 2.8 is calculated using the least square method in MATLAB with the data from the engine map. As with the air filter the measured pressure drop seems to be linear. Therefore, just as with the air filter, \mathcal{H}_{ic} is set low so that all measured pressure drops will be smaller than, the with MATLAB's *lsqcurvefit* thereafter calculated, $p_{lin_{ic}}$ (see Fig. 4.5).

The modeled pressure loss follows the mapped data well and have a correlation of 98% a mean relative error of 18.44% and a largest relative error of 99.2%. The value for $\frac{\max|e_i|}{\text{mean}(\Delta p_{ic_i})}$, where i is the sample index and e is the error, is 0.55. The very large relative errors come from the fact that the pressure drop is very small which makes a small absolute error result in a large relative error.

In the model for the temperature drop over the intercooler described in section 2.4, one of the inputs is the mass flow of the cooling air. This has in the test cell a constant but unknown value. Therefore the parameter a_3 can be set to zero and the effect of $a_3 \frac{W_{ic}}{W_{cool}}$ will instead be included in the parameter a_2 . T_a , the ambient temperature measured with the temperature sensor in the LFE3, is used as T_{cool} as it is the temperature measurement that is closest to that of the cooling air going through the intercooler in the test cell. The resulting model then becomes:

$$\begin{aligned} T_{ic_{out}} &= \max(T_a, T_c + \epsilon(T_c - T_a)) \\ \epsilon &= a_0 + a_1 \left(\frac{T_c + T_a}{2} \right) + a_2 W_{ic} \end{aligned} \quad (4.1)$$

The calculation of the temperature parameters is done with the least square method in MATLAB using data from the engine map, and the resulting model gives a temperature change with a correlation of 99.67% and the resulting T_{ic} a mean relative error of 0.27% and a largest relative error of 0.82% compared to measured data (see Fig. 4.6).

4.4 Throttle

When tuning the parameter γ in the throttle model, described by equation Eq. 2.12, the map with constant throttle angle is used, as then $C_d(A(\alpha))A(\alpha)$ is constant. $C_d(A(\alpha))A(\alpha)$ can therefore be treated as an unknown constant parameter when determining γ using MATLAB's *lsqcurvefit*.

For the points with a pressure ratio of less than $\left(\frac{2}{\gamma+1}\right)^{\frac{\gamma}{\gamma-1}}$, that with the calculated γ is ≈ 0.425 , the resulting model is, as described by Eq. 2.12 and Eq. 2.11, only depending on $p_{ic}/\sqrt{T_{ic}}$. The resulting model gives a reasonable fit to measured data with a correlation of 97.9%, a mean relative error of 4.87% and a largest relative error of 5.56% (see Fig. 4.7).

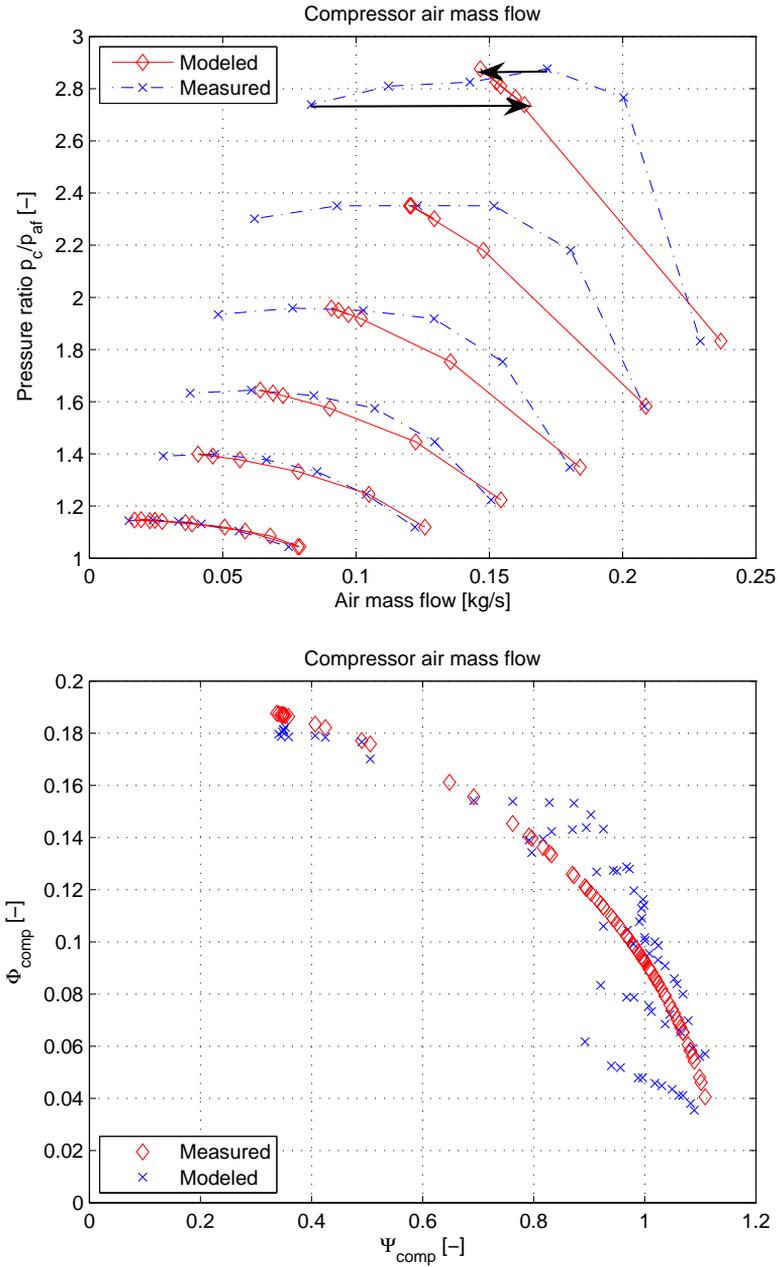


Figure 4.4. *Top:* The compressor air mass flow model gives a poor fit, notice arrows indicating difference in modeled and measured flow. *Bottom:* The basis for the compressor air mass flow model is that the dimensionless numbers Φ_{comp} and Ψ_{comp} follow a quarter of an ellipse.

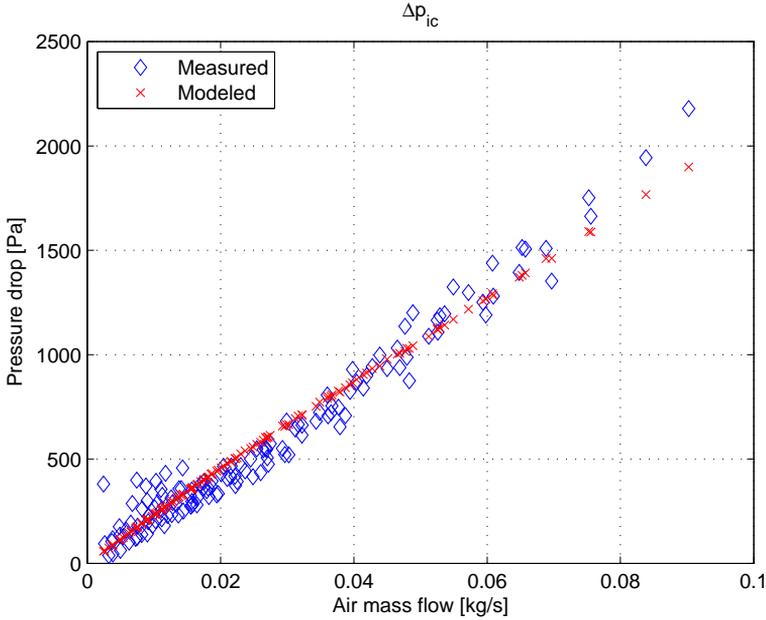


Figure 4.5. Pressure drop over the intercooler. As the measured drop is linear the linearized model is used to model it.

4.5 Engine

The engine submodels are tuned and validated separately.

4.5.1 Port air mass flow

The port air mass flow model described in section 2.6.1 is one of the models that is most directly influenced by the two biggest differences in the operation of the LNF engine compared to the B235R, SIDI and VVT.

VVT will affect the port air mass flow, and the model could be improved by taking this into account. This is however not further investigated in this thesis.

SIDI has a number of effects on the model described by Eq. 2.13. As the fuel isn't injected into the intake manifold there is no fuel vaporization taking place there, ie. no charge cooling effect, and the air will not contain any fuel, ie. no fuel fraction in the intake air. Eq. 2.13 modified according to these effects results in the port air mass flow model:

$$W_{cyl} = p_{im} C_1 \frac{r_c - \left(\frac{p_{em}}{p_{im}}\right)^{\frac{1}{\gamma}}}{r_c - 1} V_d \cdot \frac{N}{R_a T_{im} 120} \quad (4.2)$$

However as the fuel can be injected into the cylinder at the same time as the airflow is going through the open intake valve, it's possible that the effects from

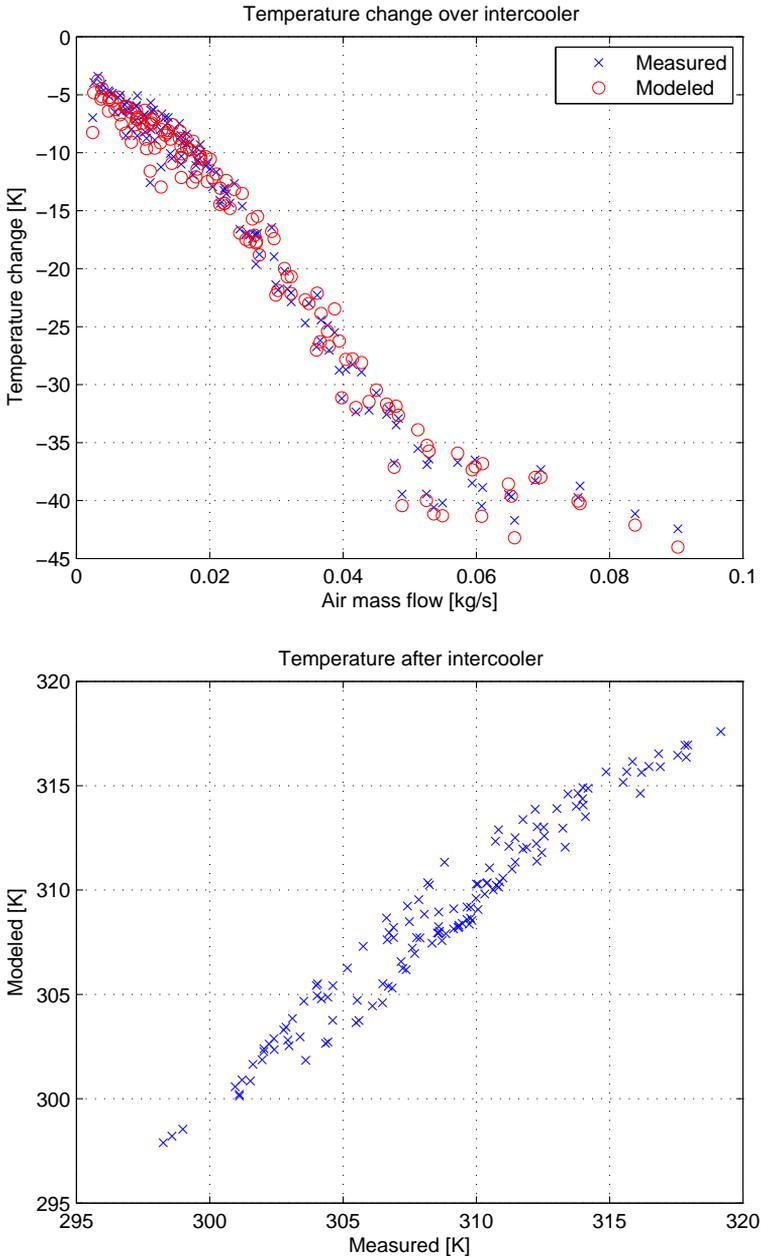


Figure 4.6. The temperature change over (top) and temperature after (bottom) the intercooler showing a good fit between the modeled and measured temperature.

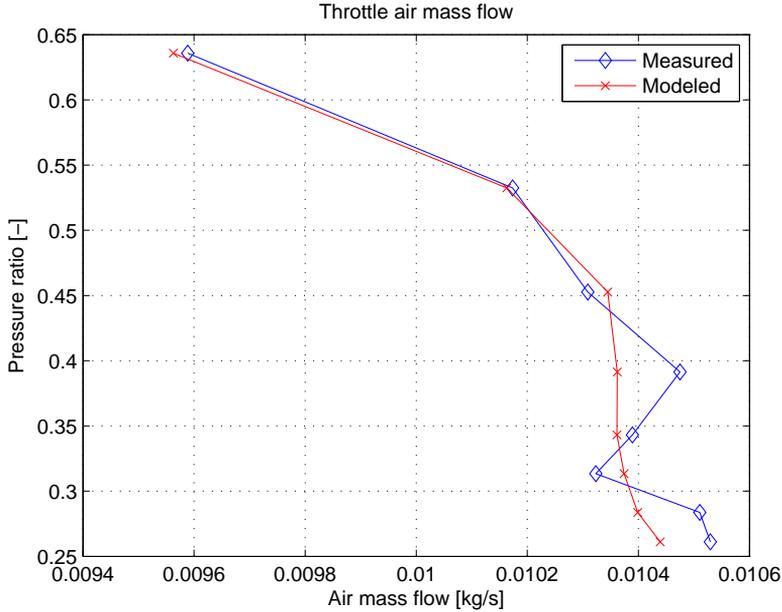


Figure 4.7. The modeled and measured flow over the throttle with a constant throttle angle as a result of the pressure difference over the throttle.

the injection of the fuel are similar to those of the fuel fraction and charge air cooling. Therefore both the original and the modified port air mass flow model are tuned and validated.

The tunable parameters are determined with MATLAB's *lsqcurvefit* using data from the engine map (Fig. 4.8). The original model gives a slightly better fit to measured data with a correlation of 98.1%, a mean relative error of 10.9% and a largest relative error of 41.9% compared to the modified model's correlation of 97.7%, mean relative error of 12.8% and a largest relative error of 47.5%. None of the models gives a very good fit as both model mainly estimates too high flows for low and medium air flows, and too low flows for higher air flows (see Fig. 4.9). Two of the reasons for the original models slightly better performance could possibly be that it accidentally captures some of the VVT's effects and not necessarily those of the fuel injection, or that the fact that it has two tunable parameters makes it possible for *lsqcurvefit* to calculate a better fit.

4.5.2 Torque

The first of the torque model parameters from section 2.6.2 that is calculated is the boost layout Π_{bl} . This is a parameter that represents the extra strengthening on the engine that is required because it is supercharged, and therefore this should be the max p_{im}/p_a ratio that the engine is subjected to during the full spectrum of its operation. In the engine map the max boost pressure is a modest 40kPa, at

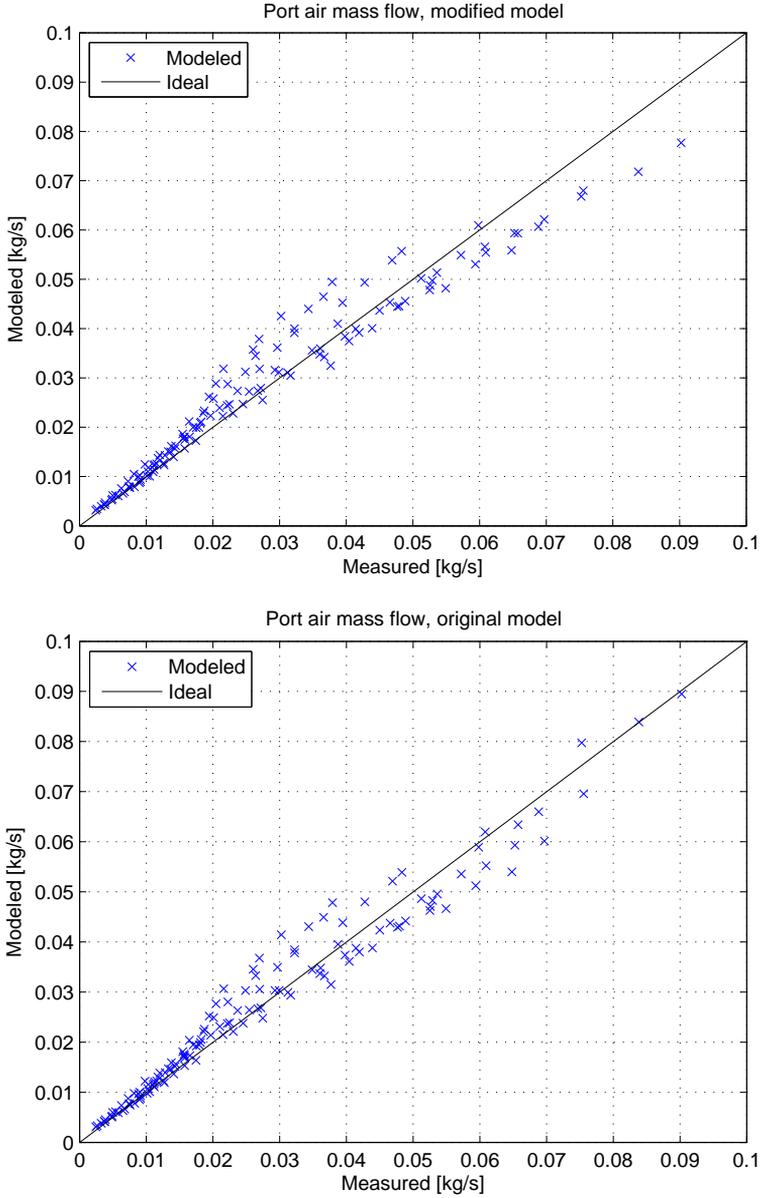


Figure 4.8. The modeled and measured air flow plotted against each other. The original model gives a slightly better fit.

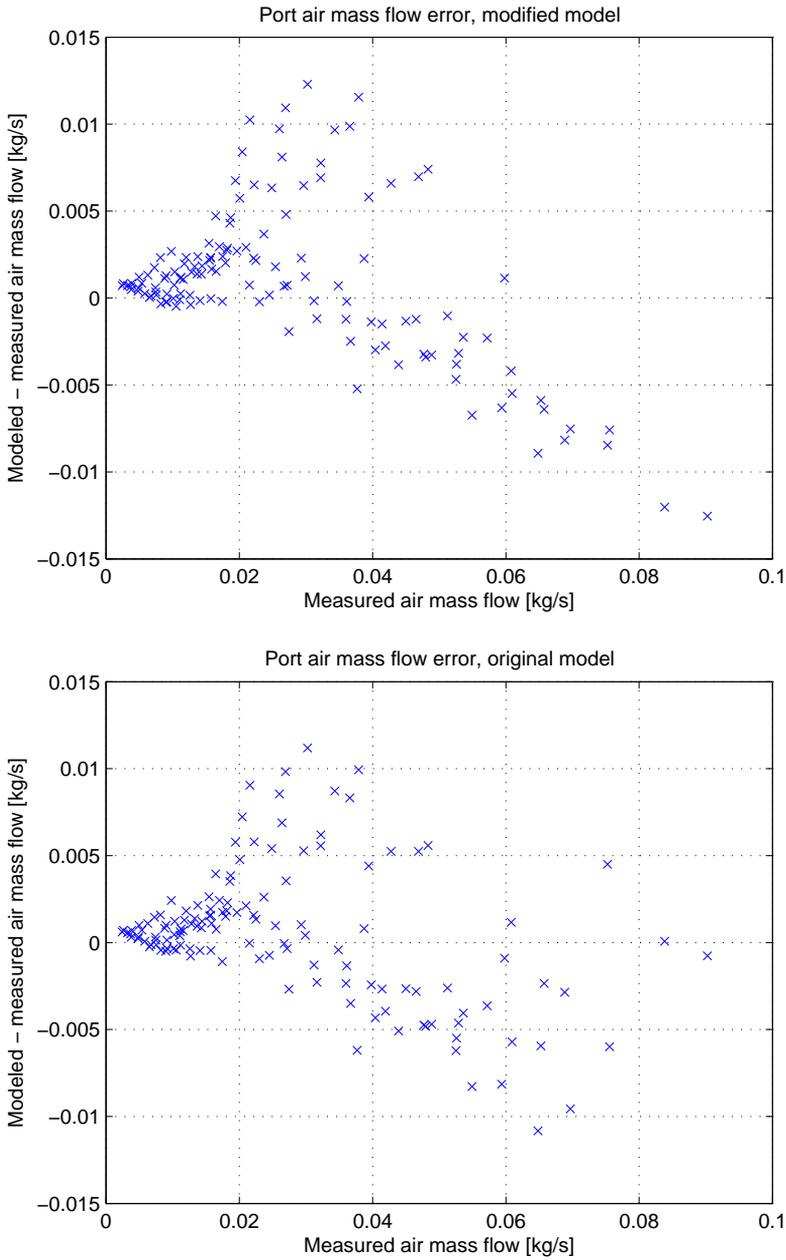


Figure 4.9. The error of the modeled port air mass flow varies for different air mass flows.

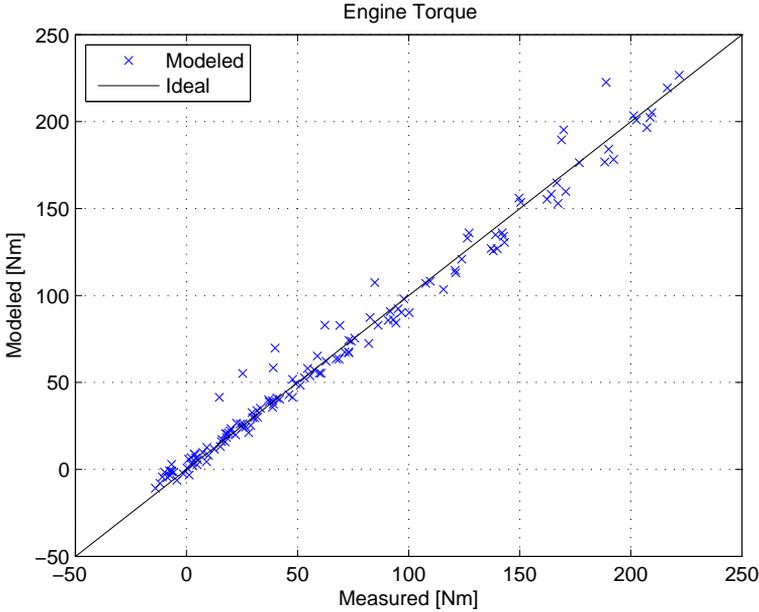


Figure 4.10. The modeled torque gives a reasonable fit to the measured engine torque.

full power the LNF engine has a boost pressure of up to 140 kPa, Π_{bl} is therefore set to 2.4. After that the parameters η_e and ζ , are calculated with the least square method in MATLAB using engine map data. Then the parameters $C_{T_{q_1}}$ and $C_{T_{q_2}}$ are calculated using MATLAB's *lsqcurvefit* using the same data as used when calculating η_e and ζ .

The resulting model gives a reasonable fit to the measured data with a correlation of 99.1% (see Fig. 4.10). The mean relative error is 5.4% and the largest relative error is 516%. The value for $\frac{\max|e_i|}{\text{mean}(T_{q_{e_i}})}$, where i is the sample index and e is the error, is 0.49. The very large relative errors mainly occur for low engine torques where a small absolute error results in large relative errors (see Fig 4.11). If one disregards the points with a torque of less than 15 Nm the mean relative error is 1.9% and the max relative error 118%.

4.5.3 Exhaust mass flow

The exhaust mass flow temperature parameters $h_{exh_{ext}}$, $\Delta T_{e_{max}}$ and T_{e_0} are determined with MATLAB's *lsqcurvefit* using data from the engine map. Because of the issues mentioned in sec. 3 not all data is used. All points where the temperature in the exhaust manifold, T_{em} , is less than 60K higher than the temperature after the turbine, T_t , are discarded.

The resulting model gives a reasonable fit to the selected data with a correlation of 95.7% (see Fig. 4.12), a mean relative error of 3.58% and largest relative error

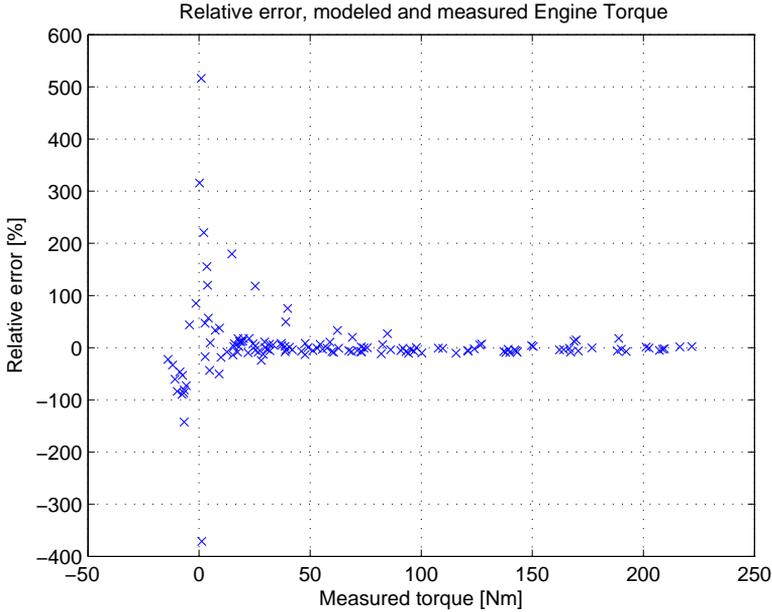


Figure 4.11. The relative error is very large for small torques where a small absolute error gives large relative errors.

of 13.1%.

4.6 Turbine

The turbine mass flow parameters k_1 and k_2 from Eq. 2.26 are calculated with MATLAB's *lsqnonlin* using turbo map data.

The resulting model gives a good fit to measured data (see Fig. 4.13) with a correlation of 99.5%, a mean relative error of 1.79% and a largest relative error of 5.34%.

The turbine efficiency parameters $\eta_{t_{max}}$ and $BSR_{\eta_{t_{max}}}$ are determined with MATLAB's *lsqcurvefit* using turbo map data. However this model gave an extremely poor fit to measured data with a non existing correlation(see Fig. 4.14).

The parameters in the alternative model described by Eq. 2.31, a_0 , a_1 , a_2 , a_3 , a_4 and a_5 , are calculated with MATLAB's *lsqcurvefit* using turbo map data. The resulting model gives a reasonable fit (see Fig. 4.14) with a correlation of 92.4% a mean relative error of 1.87% and a largest relative error of 6.42%.

The alternative model is clearly much better than the original model.

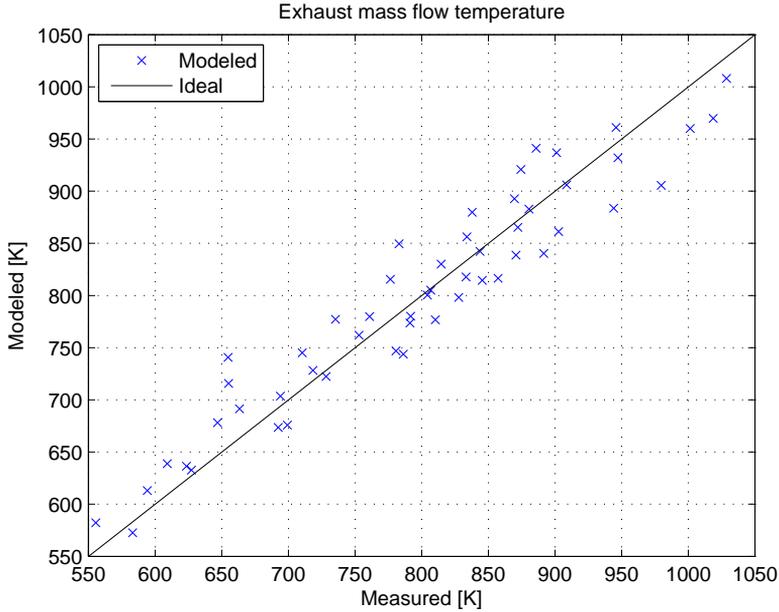


Figure 4.12. The modeled exhaust mass flow temperature gives a reasonable fit to measured data.

4.7 Wastegate

As it has not been possible to measure the wastegate flow, a validation has not been done. $A_{wg_{max}}$ has been approximated from measurements of the wastegate valve and C_d is set to 0.9 [1]. Because $A_{wg_{max}}$ is very approximative it's unnecessary to try to determine a more precise C_d .

4.8 Exhaust system

In the engine map data the pressure drop over the exhaust system is in many points negative. This is probably because the ambient pressure, here measured in the LFE3 should actually be the ambient pressure of the air outside of the test cell where the exhaust pipe ends. The ambient pressure of the air where the exhaust pipe ends is unknown, but assuming that the pressure after the turbine, p_t , isn't faulty, p_t must be bigger than p_a in a steady state measurement.

Therefore for the calculation of the pressure head loss parameter in the exhaust system, \mathcal{H}_{es} , the ambient pressure is adjusted by adding a value so that the pressure drop is bigger than zero in all points.

Using this adjusted data \mathcal{H}_{es} is calculated using the least square method in MATLAB. The resulting model gives a good fit to measured data considering that the exact value of the ambient pressure is unknown (see fig. 4.15) with a correlation

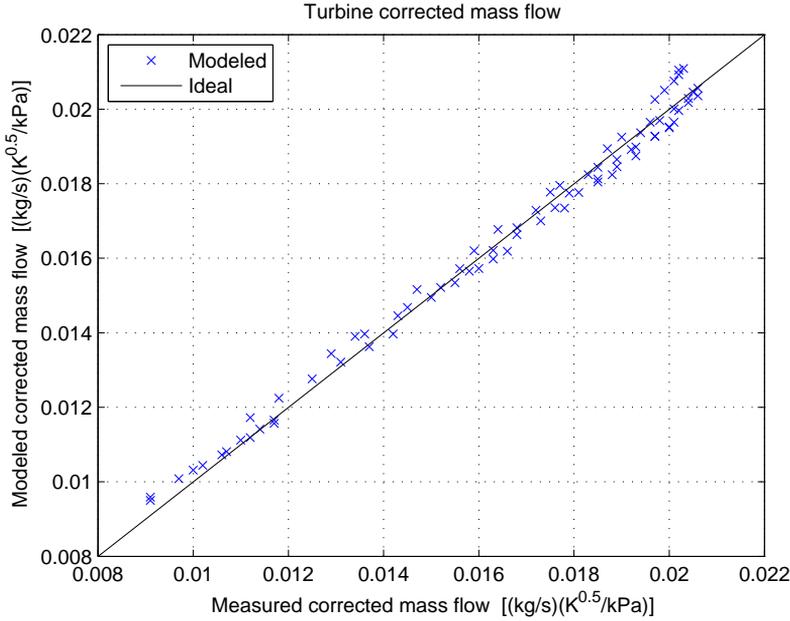


Figure 4.13. The modeled turbine mass flow shows a good fit to measured data.

of 98.55%, a mean relative error of 43% and a max relative error of 87%. The large relative errors are caused by the fact that the pressure drop is very small and a small absolute error therefore will result in a large relative error.

4.9 Turbocharger speed

The parameter I_{TC} is tuned manually by step response experiments. friction_{TC} is estimated using engine map data and the fact that when in a stationary condition Eq. 2.38 can be rewritten as $\text{friction}_{TC} = \frac{Tq_t - Tq_c}{\omega_{TC}}$.

4.10 Control volumes

All control volumes have been determined using measurements of the engine and known engine geometry. As you need to know the mass flow to and from each control volume to individually validate them it has not been possible to do so. Instead they are validated in the full model validation.

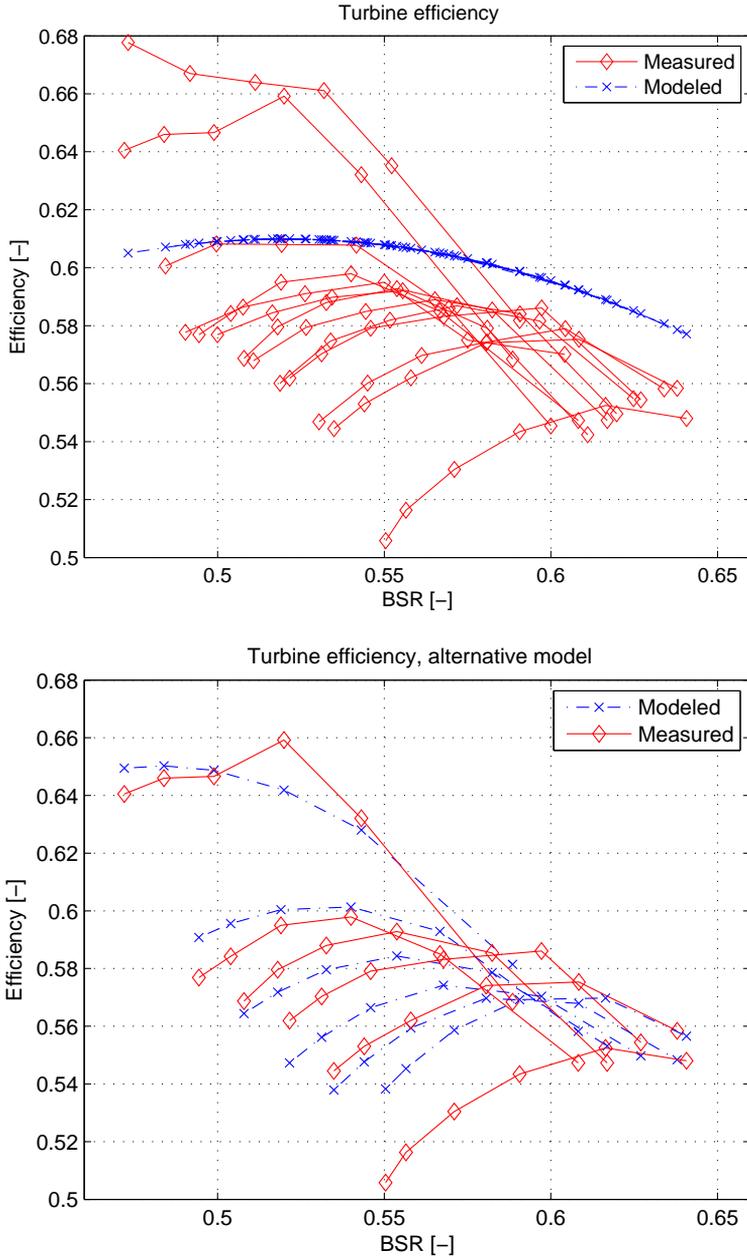


Figure 4.14. *Top:* The original turbine efficiency model fails to model the measured efficiency. *Bottom:* The modified turbine efficiency model gives a reasonable fit to measured data (to improve visibility, half of the data is removed from the plot).

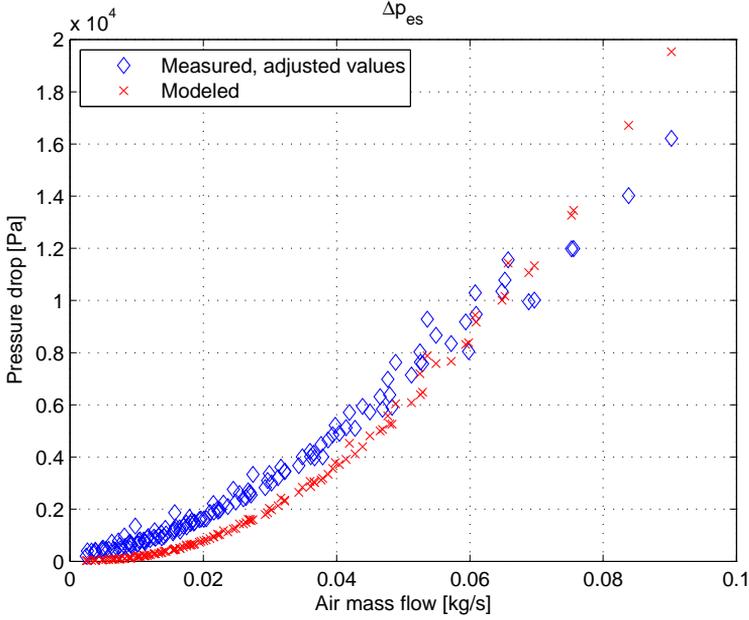


Figure 4.15. The modeled pressure drop over the exhaust system gives an acceptable fit to measured data.

4.11 Summary of submodels

Modeled value	Correlation	Mean relative error	Largest relative error
Δp_{af}	99.1%	40.2%	240%
η_c	95.2%	6.4%	35.7%
W_c	90.7%	23%	109%
Δp_{ic}	98%	18.44%	99.2%
ΔT_{ic}	99.67%	7.45%	37.9%
W_{th}	97.9%	4.87%	5.56%
W_{cyl}^1	98.1%	10.9%	41.9%
W_{cyl}^2	97.7%	12.8%	47.5%
Tq_e	99.1%	5.4%	516%
T_{em}	95.7%	3.58%	13.1%
W_t	99.5%	1.79%	5.34%
η_t^1	38.1%	5.7%	20.2%
η_t^3	92.4%	1.87%	6.42%
Δp_{exh}	98.55%	43%	87%

¹Original model as presented in [1]

²Modified model as presented in sec. 4.5.1

³Alternative model as presented in [6]

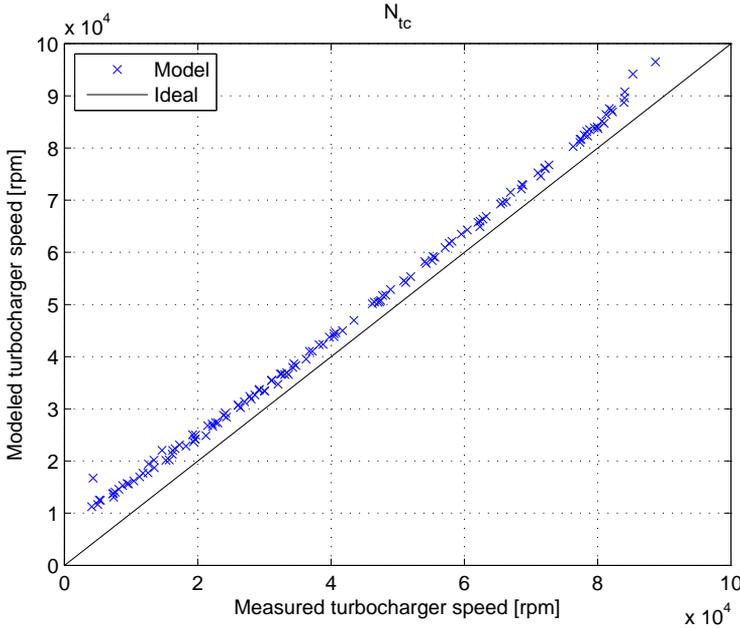


Figure 4.16. To achieve the same pressure after the compressor as in the engine map data, the turbocharger needs to have a higher speed than the measured.

4.12 Full model validation

The model is validated for both stationary and dynamic conditions. When validating the model for stationary conditions the throttle is controlled by a PI controller to achieve the same air mass flow as in the measurement. For the validation of the hot side of the engine the same selection of data that was used to tune the exhaust temperature parameters in sec. 4.5.3 is used.

4.12.1 Stationary validation

When running the full Simulink model with the determined submodel parameters it's immediately clear that the turbocharger model fails, as it is unable to reach the desired speeds. Therefore the speed of the turbocharger is simply controlled by a PI controller so that the desired pressure after the compressor, p_c , is achieved.

The turbocharger speed, N_{tc} , required to achieve the desired p_c is for all measured points higher than the measured N_{tc} (see fig.4.16). This indicates that the compressor air mass flow model has an error, and if the compressor air mass flow models parameters are changed manually a better fit, at least for the higher airflows, can be achieved.

On the cold side of the engine the intake manifold temperature is the only measurement that the model fails to describe as accurately as expected (see fig. 4.18). One possible explanation for this may be that the temperature sensor is

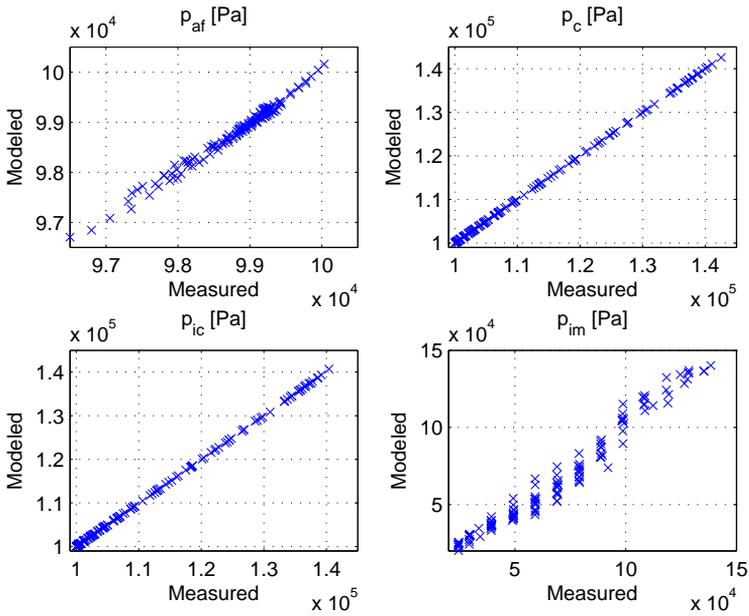


Figure 4.17. The modeled pressures on the cold side all have as good fit to measured data as expected, following the measured pressures well with the exception of the intake manifold pressure p_{im} .

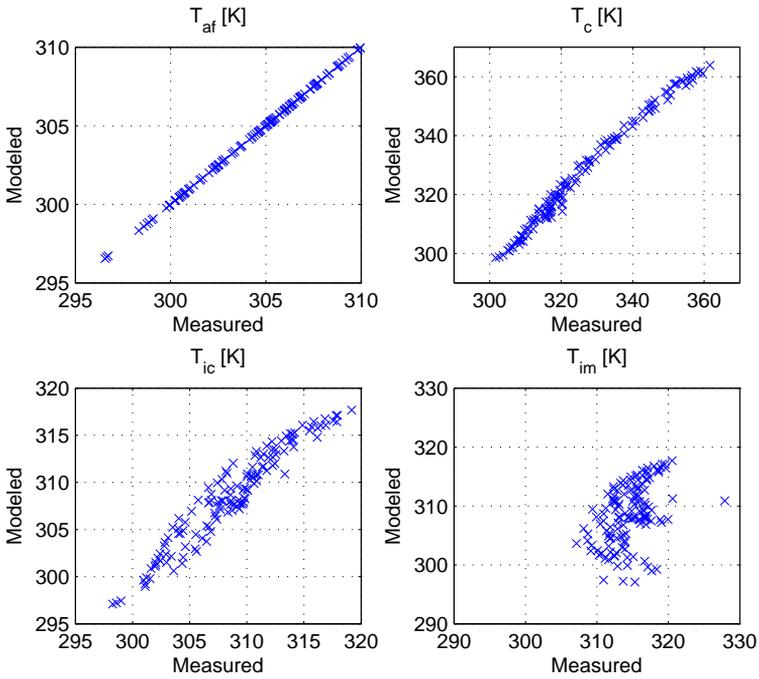


Figure 4.18. The modeled temperatures on the cold side follows the measured data well with the exception of the intake manifold temperature T_{im} .

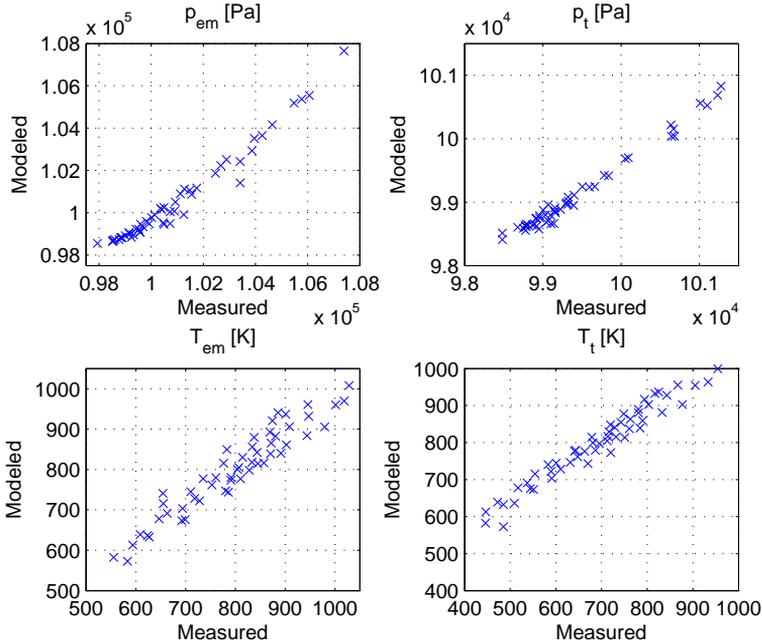


Figure 4.19. With the exception of the temperature after the turbine, the pressures and temperatures after the engine give a good fit to measured data.

heated by the intake manifold and the air flow cools it. All modeled pressures on the cold side have as good fit as expected with a good fit for all except the intake manifold pressure p_{im} . This is because of the port air mass flow model's mean relative error of $\approx 10\%$, resulting in a different modeled intake manifold pressure than the measured to achieve the measured air mass flow.

On the hot side of the engine (see fig. 4.19) the temperature after the turbine T_t is clearly not modeled correctly, probably because of a too low η_t (which is expected as the calculated turbine efficiency also includes the mechanical efficiency).

4.12.2 Dynamic validation

Because of the problems with the turbocharger model the dynamics of the turbocharger couldn't be tuned or validated.

For the validation of the control volumes the turbocharger speed is taken directly from the measurements. The control volumes dynamics of the model on the cold side follows the measured well (see fig. 4.20), on the hot side the dynamics seems to be right, but it is harder to determine the accuracy of the dynamics because of noisier signals.

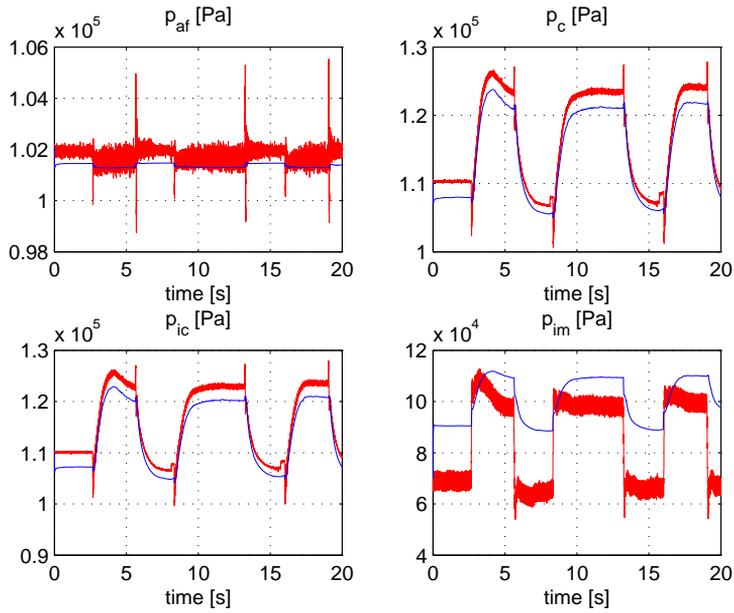


Figure 4.20. The modeled and measured dynamics of the control volumes have a good fit.

Chapter 5

Conclusions and future work

5.1 Conclusions

The MVEM from [1] has been tuned and validated for the LNF engine. As there were problems because of the experimental engine control system used during the measurements the data doesn't cover the entire engine operating region and a large part of the measurements from the hot side of the engine are unusable.

Most of the submodels of the MVEM from [1] gave a good or reasonable performance with minor or no modifications.

The compressor efficiency model gave a reasonable performance without any modifications, it's possible that a modified model for the compressor efficiency that also have a dependency of the turbo speed would perform considerably better, this however has not been investigated in this thesis. The model describing the temperature change over the intercooler gave a good fit to measured data, especially considering that the exact temperature and mass flow of the cooling air was unknown. The turbine gas mass flow model was able to give a good fit to the turbocharger map data, however it was not, as the rest of the turbocharger models, fully validated in the full MVEM model. The engine torque model, the exhaust gas mass flow temperature model and the model describing the pressure drop over the exhaust system all gave a reasonable performance without any modifications.

The models describing the pressure drops over the airfilter and the intercooler were able to give a good fit to measured data when their linearization pressures were treated as a tunable parameter instead of a manually set value. The turbine efficiency model failed to describe the turbine efficiency and a different model was tried and it was able to achieve a better fit to measured data. The model could still benefit from further improvements. The compressor air mass flow model performed very poorly but no alternative model or modifications were investigated in this thesis. The port air mass flow model was modified by removing the effects of the fuel fraction in the intake air and the charge cooling effect, as these effects don't exist in the LNF engine, however the unmodified model did give a slightly better performance. Both models have a mean relative error of $\approx 10\%$ and would benefit from further improvements.

5.2 Future work

The main obstacle for the MVEM to successfully model the LNF engine are the turbocharger models. The submodel (apart from the turbocharger models) that the MVEM probably would benefit most from if it was improved is the port air mass flow model.

All models would benefit from being validated with data covering higher air mass flows than those that were available, and a data map with more usable values for the data from the hot side of the engine would allow the turbocharger models to be properly validated.

Bibliography

- [1] Per Andersson. *Air Charge Estimation in Turbocharged Spark Ignition Engines*. PhD thesis, Linköpings Universitet, December 2005.
- [2] Lars Eriksson. Modeling and control of turbocharged SI and DI engines. *Oil & Gas Science and Technology - Rev. IFP*, 62(4):523–538, 2007.
- [3] Lars Eriksson, Lars Nielsen, Jan Brugård, Johan Bergström, Fredrik Pettersson, and Per Andersson. Modeling of a turbocharged SI engine. *Annual Reviews in Control*, 26(1):129–137, October 2002.
- [4] http://media.gm.com/us/powertrain/en/product_services/2008/08car.htm.
- [5] John B. Heywood. *Internal combustion engine fundamentals*. McGraw-Hill, 2005.
- [6] Paul Moraal and Ilya Kolmanovsky. Turbocharger modeling for automotive control applications. 1999.

Appendix A

Nomenclature

Symbol	Description	Unit
N_e	Engine speed	<i>rpm</i>
α	throttle angle	<i>rad</i>
λ	Normalized air/fuel ratio	–
p_a	Ambient pressure	<i>Pa</i>
T_a	Ambient temperature	<i>K</i>
p_{af}	Pressure after air-filter	<i>Pa</i>
T_{af}	Temperature after air-filter	<i>K</i>
W_{af}	Air mass flow through air-filter	<i>kg/s</i>
p_c	Pressure after compressor	<i>Pa</i>
T_c	Temperature after compressor	<i>K</i>
W_c	Air mass flow from compressor	<i>kg/s</i>
p_{ic}	Pressure after intercooler	<i>Pa</i>
T_{ic}	Temperature after intercooler	<i>K</i>
W_{ic}	Air mass flow through intercooler	<i>kg/s</i>
p_{im}	Intake manifold pressure	<i>Pa</i>
T_{im}	Intake manifold temperature	<i>K</i>
W_{im}	Air mass flow into intake manifold	<i>kg/s</i>
p_{em}	Exhaust manifold pressure	<i>Pa</i>
T_{em}	Exhaust manifold temperature	<i>K</i>
W_e	Air mass flow from engine	<i>kg/s</i>
p_t	Pressure after turbine	<i>Pa</i>
T_t	Temperature after turbine	<i>K</i>
T_{esin}	Temperature in to the exhaust system	<i>K</i>
W_t	Air mass flow from turbine	<i>kg/s</i>
W_{wg}	Air mass flow from wastegate	<i>kg/s</i>
W_{es}	Air mass flow through exhaust system	<i>kg/s</i>
ω_{tc}	Turbocharger speed	<i>rad/s</i>
N_{tc}	Turbocharger speed	<i>rpm</i>
W_{cyl}	Air mass flow to cylinders	<i>kg/s</i>

<i>continued from previous page</i>		
Symbol	Description	Unit
n_{cyl}	Number of cylinders	–
V_d	Swept volume	m^3
r_c	Ratio of compression	–
Stroke	Stroke	m
Bore	Bore	m
n_r	Number of revolutions per cycle	–
q_{HV}	Specific energy contents of fuel	J/kg
$(\frac{A}{F})_s$	Stoichiometric air/fuel ratio	–
c_{peg}	Specific heat, exhaust gas	$J/(kg \cdot K)$
γ_{eg}	Ratio of specific heat (c_p/c_v)	–
R_{eg}	Universal gas constant, exhaust gas	$J/(kg \cdot K)$
c_{pa}	Specific heat, air	$J/(kg \cdot K)$
γ_a	Ratio of specific heat (c_p/c_v)	–
R_a	Universal gas constant, air	$J/(kg \cdot K)$
D_{em}	Exhaust manifold pipe diameter	m
l_{em}	Exhaust manifold pipe length	m
μ_{exh}	Dynamic viscosity of exh. gas	$kg/(m \cdot s)$
λ_{exh}	Thermal conductivity of exh. gas	$W/m \cdot K$
T_{e0}	Exhaust gas temperature parameter	K
$\delta T_{e_{max}}$	Exhaust gas temperature parameter	$K/(kg/s)$
$h_{exh_{ext}}$	Exhaust gas heat transfer parameter	m
D_t	Turbine diameter	m
D_c	Compressor diameter	m
p_{std}	Correction pressure	Pa
T_{std}	Correction temperature	K
HT_{exh_0}	Heat transfer parameter	–
HT_{exh_1}	Heat transfer parameter	–
Tq_e	Engine torque	Nm
Tq_c	Compressor torque	Nm
Tq_t	Turbine torque	Nm
\mathcal{H}_{af}	Pressure head loss parameter	$Pa^2 s^2 / Kkg^2$
\mathcal{H}_{ic}	Pressure head loss parameter	$Pa^2 s^2 / Kkg^2$
\mathcal{H}_{es}	Pressure head loss parameter	$Pa^2 s^2 / Kkg^2$
$p_{lin_{af}}$	Linearization pressure	Pa
$p_{lin_{ic}}$	Linearization pressure	Pa
$p_{lin_{es}}$	Linearization pressure	Pa
$K1, K2$	Compressor air mass flow parameter	–
η_c	Compressor efficiency	–
$W_{c_{max}}$	Compressor efficiency parameter	kg/s
$\Pi_{c_{\eta_{max}}}$	Compressor efficiency parameter	–
$\eta_{c_{max}}$	Compressor efficiency parameter	–
a_{11}, a_{12}, a_{22}	Compressor efficiency parameters	–
W_{cool}	Mass flow of cooling air	kg/s
T_{cool}	Temperature of cooling air	K

<i>continued from previous page</i>		
Symbol	Description	Unit
a_0, a_1, a_2, a_3	Intercooler heat transfer parameters	–
γ	Ratio of specific heat, nonphysical value	–
$\Pi_{lin_{th}}$	Throttle linearization pressure	Pa
$C_d(A(\alpha))$	Discharge coefficient for the throttle	–
$A(\alpha)$	Throttle area	m^2
C_1, C_2	Port air mass flow parameters	–
η_e	Engine efficiency parameter	–
ζ_{aux}	Engine friction parameter	–
C_{Tq_1}, C_{Tq_2}	BMEP approximation parameters	–
Π_{bl}	Engine boost layout	–
$\Pi_{lin_{wg}}$	Wastegate linearization pressure	Pa
C_d	Discharge coefficient for the wastegate	–
u_{wg}	Wastegate position	–
$A_{wg_{max}}$	Maximum open wastegate area	–
$k1, k2$	Turbine mass flow parameter	–
$\eta_{t_{max}}$	Turbine efficiency	–
$\eta_{t_{max}}$	Turbine efficiency parameter	–
$BSR_{\eta_{t_{max}}}$	Turbine efficiency parameter	–
a_0, \dots, a_5	Turbine efficiency parameters	–
I_{tc}	Turbo moment of inertia	$kg \cdot m^2$
$friction_{tc}$	Turbo friction	Js/rad

Appendix B

MVEMparameter

```
function result = MVEMparameter(calcnr, EngMap, EngData)
%Calculates MVEM parameters from enginemap data

switch calcnr
%-----
case 1

    % Throttle Gamma

    Psi = (EngMap.mDotA.*(sqrt(287.*EngMap.T_ic))./ ...
            (EngMap.p_ic);
    storapi = EngMap.p_im./EngMap.p_ic;

    f = @(gamma,grandpi) gamma(2).*(sqrt(2*gamma(1)/ ...
            (gamma(1)-1).*((grandpi.^(2/gamma(1)))-(grandpi.^( ...
            ((gamma(1)+1)/gamma(1))))));

    result = lsqcurvefit(f,[2 1e-4],storapi,Psi);

%-----
case 2

    % h_exh_ext
    % Lambda_exh och mu_exh must be known
    %[exhTemp.zeroFlowTemp,exhTemp.tempChangeWithFlow,h_exh_ext]

    hexhi = 0.26 .*EngData.exhMan.exh_lambda.*(EngMap.mDotA.*...
            (1+1./(EngMap.lambda.*EngData.fuel.AFs))./ ...
```

```

        (pi*EngData.exhMan.exh_mu*EngData.exhMan.eManDiam)).^...
        0.6.*(1/EngData.exhMan.eManDiam);

    htot = @(hexhe) (1./((1./hexhi)+(1./hexhe)));

    Tem = @(blaj,foo) 298+((blaj(1) +blaj(2).*EngMap.mDotA.*...
        (1+1./(EngMap.lambda.*EngData.fuel.AFs)))-298).* ...
        exp((-pi.*EngData.exhMan.eManDiam.* ...
        EngData.exhMan.eManLen.*4.*htot(blaj(3)))./(foo.* ...
        (1+1./(EngMap.lambda.*EngData.fuel.AFs)).* ...
        EngData.gasProp.exh.cp));

    result = lsqcurvefit(Tem, [1100 3000 100], EngMap.mDotA, ...
        EngMap.T_em);

%-----
case 3

% Engine Torque parameters Ctq1 och Ctq2
% In the simulinkmodel Ctq1 and Ctq2 are reversed!
% The real Ctq2 i.e. the simulinkmodels Ctq1 should change
% sign

    BMEP = EngMap.Tq.*4.*pi./EngData.geometry.V_d;

    BMEPapp = @(x,Wcyl) Wcyl.*(120./(EngMap.N.* ...
        EngData.geometry.V_d)).*(x(1))+x(2);

    result = lsqcurvefit(BMEPapp, [1e5 1e5],EngMap.mDotA ,BMEP);

%-----
case 4
    % Engine Torque parameter boost layout

    result = max(EngMap.p_im./EngData.ambient.p);

%-----
case 5

    % Engine Torque parameters zeta och eta

```

```

% Tq > 0 och lambda ~= 1

mps = 2.*EngMap.N./60.*EngData.geometry.stroke;

PMEP = EngMap.p_em - EngMap.p_im;

BMEP = EngMap.Tq.*4.*pi./EngData.geometry.V_d;

% N.B. this is FMEP / zeta :
FMEP = sqrt(75/(1000*EngData.geometry.bore)).*((0.464 + ...
    0.0072.*(mps.^1.8)).*EngData.torque.Pi_bl * ...
    1e5+0.0215.*BMEP);

% N.B. this is IMEP / eta :
IMEP = (EngMap.mDotA.*(120.*EngData.fuel.q_lhv.* ...
    min(EngMap.lambda,1)./(EngMap.N.*EngMap.lambda.* ...
    EngData.fuel.AFs.*EngData.geometry.V_d)));

result = lsqr([IMEP (FMEP.*-1)], (BMEP + PMEPEP), [],20);

%-----
case 6
% Port air-mass flow
% No Charge air cooling effect!
% For model using Charge air cooling effect, use case 14
% No fuel-fraction in mass flow because of SIDI

Awcyl = EngMap.p_im.*EngMap.N.*EngData.geometry.V_d.* ...
    (EngData.geometry.r_c - (EngMap.p_em./EngMap.p_im).^ ...
    (1/EngData.gasProp.air.kappa))./(120.* ...
    ((EngData.geometry.r_c -1).*(EngData.gasProp.air.R .*...
    EngMap.T_im)));

result = lsqr(Awcyl, EngMap.mDotA, [],20);

%-----
case 7
% Turbo data, friction
% Wastegate closed and Tq positive

BSR = (EngData.turbine.D/2).*1./sqrt(2.* ...
    EngData.gasProp.exh.cp.*EngMap.T_em.*(1-(1./ ...
    (EngMap.p_t./EngMap.p_em).^...

```

```

        ((1 - EngData.gasProp.exh.kappa)/ ...
        EngData.gasProp.exh.kappa))) * EngMap.N_tc * 2 * pi / 60;

etat = EngData.turbine. efficiency. fit. maxEfficiency * ...
        (1 - ((BSR - ...
        EngData.turbine. efficiency. fit. maxEfficiencyBSR) ./ ...
        EngData.turbine. efficiency. fit. maxEfficiencyBSR ) . ^ 2);

Tqt = etat * (EngMap.mDotA * (1 + (1 ./ (EngMap.lambda * ...
        EngData.fuel.AFs)))) * EngData.gasProp.exh.cp * ...
        EngMap.T_em * (1 - (EngMap.p_t ./ EngMap.p_em) . ^ ...
        ((EngData.gasProp.exh.kappa - 1) ...
        / EngData.gasProp.exh.kappa) ./ EngMap.N_tc * 2 * pi / 60;

Tqcomp = EngData.gasProp.air.cp * EngMap.mDotA * ...
        (EngMap.T_comp - EngMap.T_af) ./ (EngMap.N_tc * 2 * pi / 60);

result = (Tqt - Tqcomp) \ (EngMap.N_tc * 2 * pi / 60);

%-----
case 8

% Pressure head loss Air filter

result = (EngMap.T_amb * (EngMap.mDotA . ^ 2) ./ EngMap.p_amb) \ ...
        (EngMap.p_amb - EngMap.p_af);

%-----
case 9

% Pressure head loss Intercooler

result = (EngMap.T_comp * (EngMap.mDotA . ^ 2) ./ EngMap.p_comp) \ ...
        (EngMap.p_comp - EngMap.p_ic);

%-----
case 10

% Intercooler temperature change param.

A = [(EngMap.T_comp - EngMap.T_amb) ...

```

```

        (EngMap.T_comp-EngMap.T_amb).*(EngMap.T_comp + ...
        EngMap.T_amb)./2 (EngMap.T_comp-EngMap.T_amb) .* ...
        EngMap.mDotA] ;

    result = A\ (EngMap.T_ic-EngMap.T_comp);

%-----
case 11

    % Exhaust system pressure head loss

    result = (EngMap.T_t.*(EngMap.mDotA.^2)./EngMap.p_t)\ ...
        (EngMap.p_t - EngData.ambient.p);

%-----
case 12

    % Port air-mass flow
    % Whith Charge air cooling effect!

    Awcyl = @(C) (C(1).*EngMap.p_im.*EngMap.N.* ...
        EngData.geometry.V_d.*(EngData.geometry.r_c - ...
        (EngMap.p_em./EngMap.p_im).^ ...
        (1/EngData.gasProp.air.kappa))./ (120.*((1+(1./ ...
        (EngMap.lambda.*EngData.fuel.AFs)))* ...
        (EngData.geometry.r_c - 1).*(EngData.gasProp.air.R.*...
        (EngMap.T_im - C(2).*((1 - EngMap.lambda)./ ...
        EngMap.lambda )))))) - EngMap.mDotA;

    result = lsqnonlin(Awcyl,[0.95 100]);

%-----
case 13

    % Pressure drop linearization limit Air filter

    flindeltap_af = @(plin,foo) sqrt((EngData.airFilter.H.*...
        EngMap.T_amb.*plin)./EngMap.p_adj).*EngMap.mDotA;
    deltap_af = (EngMap.p_adj-EngMap.p_af);

    result = lsqcurvefit(flindeltap_af,2000,[],deltap_af);

%-----
case 14

```

```
% Pressure drop linearization limit Intercooler

flindeltap_ic = @(plin,foo) sqrt((EngData.interCooler.H*...
    EngMap.T_comp.*plin)./EngMap.p_comp).*EngMap.mDotA;
deltap_af = (EngMap.p_comp-EngMap.p_ic);

result = lsqcurvefit(flindeltap_ic,2000,[],deltap_af);

%-----
otherwise
    result = 'Wrong case number';
end
```

Appendix C

MVEMturboparameter

```
function result = MVEMturboparameter(calcnr, Map, EngData)
% Calculates MVEM turbocharger parameters from turbomap data

switch calcnr
%-----
case 1
    % Compressor eff
    %[eta_max w_comp_eta_max PI_comp_eta_max a_11 a_12 a_22]

    x = Map.wN_SI;
    k = sqrt(Map.prc-1);
    inpa = [x k];
    foo = @(abc, inp) abc(1) - ((abc(4).*(inp(:,1).^2 -2.* ...
        inp(:,1).*abc(2) + abc(2).^2)) + (2*abc(5).*(1 + ...
        inp(:,2) - abc(3)).*(inp(:,1)-abc(2))) + (abc(6)* ...
        ((1 + inp(:,2) -abc(3)).^2)));
    options = optimset('lsqcurvefit');
    options.MaxFunEvals = 2000;
    result= lsqcurvefit(foo, [0.73 0.14 2.03 39 -3.7 0.94], ...
        inpa, Map.etac, [], [], options);

%-----
case 2
    % Turbine eff

    turbfoo = @(turbpar, indata) turbpar(1).*(1-((indata - ...
        turbpar(2))./turbpar(2)).^2);

    result = lsqcurvefit(turbfoo, [1 1], Map.BSR, Map.etat);
```

```

%-----
case 3
    % compressor mass-flow

    Ucomp = Map.omega.*Map.dc./2;

    Phic = Map.wN_SI ./ (Map.p1c./ (EngData.gasProp.air.R .* ...
        Map.T1c.*4 ).*pi .* EngData.compressor.D^2.*Ucomp);

    Psic = EngData.gasProp.air.cp .*2.*Map.T1c.*((Map.prc).^ ...
        ((EngData.gasProp.air.kappa-1)./ ...
        EngData.gasProp.air.kappa)-1)./Ucomp.^2;

    % plot(Psic,Phic,'x')

    compfunc = @(k,foo) Psic.^2.*k(1) + Phic.^2.*k(2);

    result = lsqcurvefit(compfunc,[1 1] , [], ...
        ones(size(Psic),1));

%-----
case 4
    % turbine mass-flow

    wtfun = @(k) (k(1).*sqrt(1-(1./Map.prt).^k(2)))-Map.wN_SI;

    result = lsqnonlin(wtfun,[1 1]);

%-----
case 5
    %Turbo data, friction

    Tqt = Map.etat.* Map.W_t.* ...
        EngData.gasProp.exh.cp.*Map.T_em.*(1 - (Map.prt.^ ...
        ((Map.kappa - 1)./ Map.kappa)))./ ...
        Map.w_tc;

    Tqcomp = EngData.gasProp.air.cp.*Map.W_comp.*(Map.T_comp-...
        Map.T_af)./(Map.w_tc);

    result = (Map.w_tc)\(Tqt-Tqcomp);

%-----

```

```
case 6
    %turbine eff. v2.

    ajteff = @(teffa, ntobsr) teffa(1)+teffa(2).*ntobsr(:,1)+...
        (teffa(3)+teffa(4).*ntobsr(:,1)).*ntobsr(:,2) + ...
        (teffa(5)+teffa(6).*ntobsr(:,1)).*ntobsr(:,2).^2;

    options = optimset('lsqcurvefit');
    options.MaxFunEvals = 4000;
    options.MaxIter = 800;

    result = lsqcurvefit( ajteff, [1 1 1 1 1 1],[Map.Nt_SI ...
        Map.BSR],Map.etat,[],[],options);
%-----
otherwise
    result = 'wrong func number';

end
```