

A Framework for Evaluation of Cylinder Balancing Controllers

Niclas Lindström

Master of Science Thesis in Electrical Engineering
A Framework for Evaluation of Cylinder Balancing Controllers

Niclas Lindström
LiTH-ISY-EX--17/5070--SE

Supervisor: **Xavier Llamas**
ISY, Linköpings universitet
Vaheed Nezhadali
Scania AB

Examiner: **Professor Lars Eriksson**
ISY, Linköping university

Vehicular Systems
Department of Electrical Engineering
Linköping University
SE-581 83 Linköping, Sweden

Copyright © 2017 Niclas Lindström

Abstract

Cylinder speed variations in a combustion engine is an unwanted phenomenon caused by a number of different reasons. Inaccurate fuel delivery from the individual injectors, resonance frequencies in the drive train and faulty sensor readings are some probable causes. There is a need to investigate the potential of different cylinder balancing controllers in a simulation environment before implementing them in the ECU hardware. The thesis is about developing a simulation framework where different controllers can be tested. The framework will generate an engine speed signal based on injected fuel mass to the individual cylinders. A PI controller that makes individual fuel adjustments to the cylinders is implemented in the framework and tested for three different operating points and three different types of disturbances. The results show that the framework is able to generate an accurate engine speed signal based on the commanded fuel amount. Moreover the controller is able to eliminate imbalances caused by error in injected fuel mass as well as specific type of periodic load disturbances in the drive line. Some disturbances can not be handled by the PI controller, as they lie outside of its controllable region. The simulation framework shows promising results and while further work is needed in some areas, it can work as a foundation for future development and controller evaluation.

Acknowledgments

I would first of all like to thank Vaheed Nezhadali for his supervision during this project. His knowledge and experience within the subject area has been of great aid and helped keeping the project moving in the right direction. I also owe thanks to my supervisor at Linköping University, Xavier Llamas who has given constructive advice during the project and fast feedback on questions. I would also like to thank my examiner Professor Lars Eriksson whose courses in engine modelling as well as published literature have shown to be of great help in this thesis work.

*Södertälje, May 2017
Niclas Lindström*

Contents

Notation	ix
1 Introduction	1
1.1 Background	1
1.2 Purpose And Goals	1
2 Related Research	3
2.1 In-Cylinder Processes	3
2.1.1 Intake stroke	3
2.1.2 Compression	4
2.1.3 Combustion	4
2.1.4 Exhaust	6
2.2 Pressure	6
2.3 Torque	7
2.4 Cylinder Balancing Control	7
3 Method	9
3.1 Modelling Approach	9
3.1.1 In-Cylinder processes	9
3.2 Crankshaft Dynamics	10
3.3 Cylinder Pressure Model	11
3.3.1 Compression	11
3.3.2 Expansion	13
3.3.3 Valve Model	14
3.3.4 Summarizing the Cylinder Pressure Model	14
3.4 Parameter Estimation	14
3.4.1 Manifold Pressures and Temperatures	15
3.4.2 Combustion Parameters	16
3.5 C-bal Controller	17
3.6 The simulation framework	19
3.6.1 Subsystem Description	19
3.6.2 Simulation user manual	22

4	Results and Discussion	25
4.1	Modelling Validation	25
4.1.1	Pressure Validation	25
4.1.2	Changing SOI and EOI	28
4.1.3	Engine Speed Signal	29
4.1.4	Parameter Estimation	32
4.2	Controller results	35
4.2.1	Disturbance in injected fuel	35
4.2.2	Disturbance in load torque	36
4.2.3	Error in engine speed sensor	38
4.3	Discussion	40
5	Conclusions	43
5.1	Future Work	44
	Bibliography	47

Notation

ABBREVIATIONS

Abbreviation	Description
TDC	Top Dead Centre
BDC	Bottom Dead Centre
ATDC	After Top Dead Centre
ABDC	After Bottom Dead Centre
BBDC	Before Bottom Dead Centre
BTDC	Before Top Dead Centre
SOI	Start Of Injection
EOI	End Of Injection
SOC	Start Of Combustion
ECU	Electronic Control Unit
CAD	Crank Angle Degree
IVC	Inlet Valve Closing
IVO	Inlet Valve Opening
SI	Spark Ignition
CI	Compression Ignition
C-bal	Cylinder Balancing

1

Introduction

1.1 Background

Nowadays direct injection engines use high pressure fuel injection systems that feature high control accuracy. Even though these injectors permit high control of fuel delivery slight deviations between different injectors exist because of limited manufacturing accuracy, and consequently cylinder to cylinder combustion torque variations can occur. These variations will cause unpleasant vibrations and ultimately put unnecessary strain on the moving parts of the engine and driveline.

However unequal fuel distribution is not the only cause of cylinder imbalances. Asymmetries in the intake system can impose a variability in air charge to individual cylinders and thereby result in unequal combustion. Moreover, soot deposits occur as the engine ages which can affect air flow through the valves as well as the flow through the injector nozzle.

Even though unequal fuel quantity is not the only reason for cylinder imbalances it is by far the most common and since injectors are the fastest available actuators at hand, this project will focus on reducing the imbalances by adjusting the amount of fuel injected in each cylinder.

It is also important to note that while the engine is idling, i.e. when the driveline is not connected to the engine, the balancing problem has a different nature than when the driveline and engine are connected because of the resonance frequencies that occur in a closed driveline.

1.2 Purpose And Goals

There is an interest to be able to conveniently test and evaluate different cylinder balancing controllers in a controlled environment. Testing and implementing all

the different solutions directly in a real engine is both inconvenient and expensive. This supports the idea of building a simulation framework for evaluating the controller before implementing in the ECU software.

As a request from Scania CV, the goal of this project is to design such a framework in Matlab/Simulink. The injected fuel will result in a pressure build up. From the pressure the combustion torque and a crankshaft model, based on Newton's second law of motion will be used to calculate the crankshaft acceleration. From the crankshaft acceleration the engine speed signal can be integrated. A PI-controller will be implemented, which measures the engine speed signal, detects a provoked imbalance and makes the necessary fuel adjustments to the individual cylinders. The objectives of the project are listed below.

1. **Modeling** - Must be able to generate an engine speed signal that captures the effect of disturbances in engine load, fuel amount and engine speed sensor.
2. **Controller** - Must have stability in the desired operating range of the engine and should be able to remove the stationary error (engine speed imbalance).
3. **The interface between model and controller** - The framework should enable testing and evaluation of various controllers. Therefore it should be possible to replace the controller without having to make changes to the model itself. I.e. the input to the model from the controller will be adjustments in fuel amount, and the input from the controller will be an engine speed signal generated by the model.

The framework will be modelled for a six cylinder engine and will primarily be evaluated on its ability to generate an engine speed signal and relevant disturbances.

The complete simulation framework with the implemented controller will also be tested on four different test scenarios to verify proper functionality.

2

Related Research

2.1 In-Cylinder Processes

The pressure trace in the cylinder during the four stroke cycle is a consequence of the in-cylinder processes. In this section these processes will be covered from a thermodynamic perspective. The processes are based on the First Law of thermodynamics

$$\frac{dE}{dt} = \frac{dQ_w}{dt} - \frac{dW}{dt} + \sum_j \frac{dm_j}{dt} h_j \quad (2.1)$$

where $\frac{dQ_w}{dt}$ is the total heat transfer across the system boundaries, $\frac{dW}{dt}$ is work transfer rate and equals $P \frac{dV}{dt}$. The heat released from the combustion is accounted in the energy and enthalpy terms.

2.1.1 Intake stroke

The intake stroke typically starts at around 10 degrees before top dead centre (BTDC) as the intake valve opens. The exhaust valve typically closes at around 15-20 degrees after top dead centre (ATDC) which means that there is a period of valve overlap where the intake valve and the exhaust valve is open together. This valve overlap will improve volumetric efficiency, especially as engine speed increases [10]. As the exhaust valve closes the pressure successively becomes closer to the intake manifold pressure as the piston approaches the bottom position. The intake valve will remain open all the way until around 50-70 degrees after bottom dead centre (ABDC) so that fresh charge can be inducted for as long as possible.

2.1.2 Compression

The compression stroke will start at bottom dead centre (BDC) and initially the inlet valve will continue to remain open during the first part of the compression stroke. As the inlet valve closes a slight pressure increase will result. From there on all the valves are closed and the pressure will continue to increase as the volume decreases. As the piston approaches TDC the compression pressure will be high enough for the diesel fuel to self-ignite.

2.1.3 Combustion

Understanding the combustion development in the four stroke cycle is a key for accurately modelling the pressure trace. The event is highly complex and a complete understanding of all the underlying processes is still not fully achieved. The diesel combustion is significantly different from the gasoline combustion, partly because of the chemical differences between the fuels but also because of the way that the fuel is injected and ignited. In an SI-engine, the fuel mixture is injected early in the compression stroke and is ignited by the spark plug about 10 to 40 degrees BTDC and the burning processes is typically between 40 to 60 crank angle degrees. In a CI-engine air alone is inducted into the cylinder and the fuel is injected close to top dead centre at the end of the compression stroke. The fuel is self-ignited due to the high compression temperature. The load is primarily varied by controlling the amount of fuel that is injected while the air charge is essentially unchanged. This differs from the SI-engine where the air and fuel ratio inside the cylinder is of greater importance.

The burning process

The combustion in the CI-engine can be split into three distinct parts, [10]. As the fuel is injected into the cylinder just before TDC, the fuel first evaporates and mixes with the compressed air. The time between when the diesel fuel is injected into the cylinder and when the actual combustion is initiated is called ignition delay. Spontaneous combustion within the non-uniform mixture initiates the burning event and the pressure increases rapidly. This part is referred to as the *premixed phase*. The initiation of the burning process facilitates the flame to spread. During the expansion the fuel continues to be injected and is ignited almost instantly because the initiated burning. This part up until the injection of fuel stops is typically referred to as the *diffusion phase*. In the final phase, usually referred to as *late diffusion phase*, the remaining fuel is combusted, usually well into the late parts of the expansion.

During the combustion some of the air-fuel mixture is left unburnt and is left as residual gases for the next cycle. The amount of residual gas is typically a function of inlet and exhaust pressures, engine speed, compression ratio and valve timing, [10]. The work in [9] suggests an iterative method for determining the residual gas fraction and the residual gas temperature. Even though the residual gas fraction is substantially lower for CI-engines than SI-engines, due to the

higher compression ratio and leaner air-fuel mixture, its effects are still to be considered.

Heat Release Analysis

The combustion process is closely related to the heat release profile, which describes the heat released during the combustion event as a function of crank angle. There are typically two different alternatives for describing the heat release rate. The first is the gross heat release rate, which includes the chemical energy released from the fuel, as well as heat transfer effects. The other alternative is to use the apparent heat release rate, which only takes in account for the chemical energy released and discards the heat transfer effects. The apparent heat release is typically favoured because of its computational simplicity. Values obtained from the apparent heat release model are typically 15% lower than those from the gross heat release model, [3].

The heat release profile is closely related to the mass fraction burned, which essentially describes the rate of the fuel that is burnt. Describing the mass fraction burned is typically accomplished using a single-zone model or a multi-zone model. The single-zone model treats the gases in the cylinder as one uniform composition whilst multi-zone models have different zones with different thermodynamical characteristics. For example the unburned region and the burned region of fuel is modeled differently. The single-zone model is often restricted by its capacity to match both the slower combustion that occur in the cooler boundary areas adjacent to the cylinder walls, and the faster combustion in the cylinder core, [18].

The accuracy of the single-zone model has been investigated in [4]. Such models are usually favoured compared to the more detailed multi-zone models, as they are numerically more efficient. Three different methods of calculating heat release were tested and evaluated on real data. The models included one gross First Law, single-zone model with included heat transfer. The second model was a net First Law, single zone-model, which discards the effect of heat transfer, and the final was a polytropic index first law model. The results showed that both the gross First Law model and the polytropic index model were able to capture the most important behaviour of the heat release profile.

The burned mass fraction is closely related to the heat release profile and a common way to apply a distribution function for the burned mass fraction is to use the well known Wiebe function, which can be described as

$$x_b(\theta) = \exp \left[-a \left(\frac{\theta - \theta_{soc}}{\Delta\theta} \right)^{m+1} \right] \quad (2.2)$$

x_b is the mass fraction burned as a function of crank angle θ , θ_{soc} is the angle at start of combustion, $\Delta\theta$ is the combustion duration and m and a are tuning parameters to be determined. The angle at start of combustion is of great importance for accurate combustion modelling and is depending on many variables. The work done in [2] proposes a well known relationship for estimating ignition delay based on the pressure and temperature just before combustion is initiated.

In the work done in [11] pressure was used to estimate relevant heat release parameters. The heat release profile was estimated using a double Wiebe function. The reason for the double Wiebe function is covered in [10] and [14] and is related to the distinct burning phases in CI combustion. Each Wiebe function describe the respective burning phase in order to more accurately capture the behaviour of the burning process.

A method for determining the mass fraction burned using the ratio between the motored pressure and the combustion pressure is covered in [13]. The method is based on the assumption that the pressure increase due to combustion is proportional to the burned fuel mass mixture.

Another way of describing the burned mass fraction is covered in [12]. The method proposed there is non-standard, where the premixed combustion is approximated by a quadratic function and the diffusion is interpolated between a quadratic and an exponential function. There is also a newer formula for estimating the heat release rate, that is covered in [5], where the heat release rate is calculated based on the concept of mixing controlled combustion. This model considers the injection rate of the fuel as well as the influence of the kinetic jet energy.

Although mathematical functions such as the Wiebe function and the ones proposed in [12] can effectively be used to represent the combustion process there are inherent issues associated with them, typically associated with being able to accurately determining the unknown parameters. Since the functions themselves are mathematical functions and are not derived from physical relationships, estimating the unknown parameters is usually accomplished by performing a least squares fit. A systematic way of identifying these parameters using a combination of algebraic expressions and least squares fit are covered in [6] and [7].

2.1.4 Exhaust

The exhaust phase will start at around 40-60 degrees before bottom dead centre (BBDC) and as the valve opens the pressure difference between the cylinder and the exhaust manifold will drive the gas out of the cylinder. After BDC the exhaust gases will be driven by the upward motion of the piston and the pressure will get close to the exhaust manifold pressure as the piston approaches the top position, [10].

2.2 Pressure

The in-cylinder processes will result in a pressure build up in the cylinder. An approach for estimating the in-cylinder pressure using seven analytical expressions for different parts of the four stroke cycle is covered in [8]. The burned mass fraction is modelled as a single zone Wiebe function. One advantage with this method is that the pressure can be calculated analytically, as opposed to solving

the differential equation that describes the relationship between heat release and pressure. This makes the approach numerically favourable.

2.3 Torque

The pressure build up from each cylinder will cause a torque acting on the crankshaft. The work done by [16] examined how to design a virtual crank angle sensor from pressure data. The model that was implemented includes torque models for calculating the cylinder torque, friction losses and losses from other components. A rigid crankshaft model and a lumped mass torsion model are two alternative models that were used. The rigid model ignores torsional effects and approximates the crank shaft as a rigid body while the lumped mass model involves dividing the crankshaft into separate nodes that are allowed to move independently from each other.

Modelling the crankshaft dynamics has also been done in [17]. The work included a simulation model for describing the relationship between individual cylinder pressure and the resulting torque on the crankshaft. The purpose with the work was to examine how instantaneous torque measurements can be used to deduce information of the cylinder-wise combustion processes. The model is based on a system of first order non-linear differential equations, where the crankshaft is divided into several interconnected mass-spring elements.

2.4 Cylinder Balancing Control

The operating principle of an internal combustion engine usually results in periodic signals with the engine's firing frequency. Consequently the engine speed signal will consist of pulses from the individual cylinders. When the engine is running at steady state operation, the pulses should ideally have an equal amplitude and overall characteristics. When a disturbance is present that causes variations in the cylinder speeds, the pulses will differ in amplitude and behaviour.

In order to correct this imbalance, fuel quantity corrections can be made to the individual cylinders. The work done in [12] presents three different approaches for cylinder balancing based on crankshaft acceleration, in-cylinder pressure and exhaust manifold pressure measurements. Using crankshaft acceleration measurements, the imbalance is calculated based on the passage time between two teeth on the flywheel. These two teeth make up the crank window and is chosen to have maximum information about crankshaft acceleration. From the imbalance the necessary fuel adjustments are calculated to the individual cylinders. Experimental results showed that the controller is able to quickly eliminate the imbalance when one injector is biased by +20%.

Another approach for detecting cylinder imbalance is covered in [1]. The measured engine speed signal is used as an input for estimating the individual cylinder pressure traces using a sliding mode observer. The results show that the design is accurate enough and robust to disturbances in the measured signal.

In the work done by [15], a cylinder balancing method is presented for a medium-speed power plant engine. The method is based on recreating the oscillating torque from torsional vibrations on the crankshaft, and adjust the cylinder-wise fuel injections to minimize these vibrations.

3

Method

This section will systematically cover the modelling strategy that was chosen to model the crankshaft dynamics, in-cylinder pressure, estimation of parameters as well as the cylinder balancing controller. The modelling was done for the DC13155 engine, which is a six cylinder with a displacement of 12.74 litres, producing around 500 hp. The resulting simulation framework, including a thorough explanation of its belonging systems will also be covered.

3.1 Modelling Approach

3.1.1 In-Cylinder processes

In order to determine what modelling choices are suitable, one needs to take in consideration what the model is going to be used for and where focus needs to be drawn to, as well as where simplifications can be made. In order to justify the design choices the following was taken into consideration.

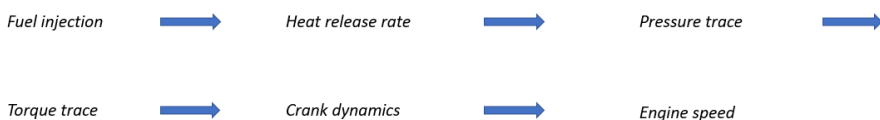


Figure 3.1: A graphical representation of the the resulting processes from injected fuel to the engine speed signal.

Figure 3.1 gives an overlook of the processes that are occurring between in-

jected fuel mass and engine speed. The fuel injection dynamics, as for example start of injection, end of injection as well as details regarding the fuel spray is closely related to the heat release rate. When evaluating heat release these details would need to be modelled in order to accurately capture this behaviour.

However the details in the heat release trace are not as sensitive to the dynamics in the engine speed signal. Because of this, a highly sophisticated model for describing the fuel spray will most likely not yield in any significant improvements in accuracy. Likewise the details in the combustion process around top dead centre, where the premixed combustion is typically present might not affect the engine speed signal that much.

Consequently the decision was to attempt to model the combustion with a single Wiebe function as described in section 2.1.3. Unknown parameters which could not be analytically expressed such as manifold temperatures and pressures were to be determined by measurement data from a range of operating points.

Since the model ideally should have capacity to be simulated for several minutes it in turns put certain restrictions to the numerical complexity of the model.

3.2 Crankshaft Dynamics

The crankshaft dynamics describes how the resulting torque contributions from each cylinder affect the angular acceleration of the crankshaft. The crankshaft can be modelled in different ways with varying complexity. The simplest way is to model the crankshaft as a rigid crankshaft.

$$\ddot{\theta} = \frac{T_{gas}(\theta, t) + T_{fric}(\dot{\theta}) + T_{mass}(\theta, \dot{\theta}) + T_{load}(t)}{J(\theta)} \quad (3.1)$$

Equation (3.1) describes the angular acceleration of a rigid crankshaft. The rigid crankshaft model discards the torsional effects that can occur in the crankshaft. A lumped mass crankshaft model can be used to include these torsional effects and is described as following

$$\ddot{\theta} = \frac{T_{gas}(\theta, t) + T_{fric}(\dot{\theta}) + T_{mass}(\theta, \dot{\theta}) + T_{load}(t)}{\mathbf{J}(\theta) + \mathbf{C}_d\theta + \mathbf{K}\theta} \quad (3.2)$$

Besides the inertia term \mathbf{J} , the lumped mass model also includes a damping \mathbf{C}_d and a stiffness \mathbf{K} . The lumped mass model splits the crankshaft up into several node elements, where each node has a specific inertia, damping and stiffness and can therefore move relative to the other node elements. The three terms are marked with bold font to emphasise that they are not constants, but matrices consisting of the node element values.

The model that was chosen to be implemented is based on a previous thesis, see [16]. The model includes functionality to calculate the torque contribution from each cylinder as well as models to account for the different torque losses. The model also included functionality for a rigid crankshaft model as well as a lumped mass crankshaft model. Because of that both the rigid and the lumped

mass crankshaft model was chosen to be integrated in the framework, allowing the user to manually choose which one to use. For a detailed explanation of the crank shaft model the reader is referred to the authors work.

3.3 Cylinder Pressure Model

The cylinder pressure model is based on the modelling proposition described in [8]. The pressure trace is described by a number of analytical expressions divided into different parts of the engine cycle. These parts are typically linked to specific angles of interest, such as when valves open or closes, start of combustion etc.

3.3.1 Compression

The compression part is modelled as a polytropic process and is described by the polytropic exponent k_c and the reference pressure when the intake valve closes, p_{ivc} . The resulting pressure and temperature can be expressed with the following relationship.

$$p_c(\theta) = p_{ivc} \left(\frac{V_{ivc}}{V(\theta)} \right)^{k_c} \quad (3.3)$$

$$T_c(\theta) = T_{ivc} \left(\frac{V_{ivc}}{V(\theta)} \right)^{k_c-1} \quad (3.4)$$

These relationships will describe the pressure up until the start of combustion. Determining the initial pressure p_{ivc} is of great importance in order to position the compression phase correctly.

At bottom centre just before the compression stroke, the cylinder pressure can be approximated with the intake manifold pressure. The point at which the intake valve closes (IVC), the pressure can be expressed as

$$p_{ivc} = p_{im}(\theta_{ivc}) + C_1 + C_2 N \quad (3.5)$$

where $p_{im}(\theta_{ivc})$ is the manifold pressure at IVC, N is the engine speed and C_1 and C_2 are tuning parameters. Sometimes p_{ivc} is set equal to $p_{im}(\theta_{ivc})$ but this simplification did not yield results of sufficient accuracy in all operating regions. Instead a least squares fit was performed to fit C_1 and C_2 to a particular engine speed, where $p_{im}(\theta_{ivc})$ was used as measured data.

Determining the initial temperature T_{ivc} is much harder than the initial pressure as it is dependent on heat transfer and residual gases. As fresh charge enters the cylinder the temperature of the air will undergo a slight increase due to the temperature differences in the cylinder and the intake manifold. For simplification purposes however this temperature T_a is approximated to be the same as T_{im} . As the piston approaches IVC the temperature of the air charge has been heated by the residual gases in the cylinder. If assuming that the difference between the

specific heats c_p of the air and the residual gases are negligible, the temperature at IVC can be expressed as

$$T_{ivc} = T_a(1 - x_r) + T_r x_r \quad (3.6)$$

T_r is the temperature of the residual gases and x_r is the residual gas fraction which is defined as

$$x_r = \frac{m_r}{m_r + m_f + m_a}$$

Determining the residual gas fraction The residual gas fraction can have a significant effect on the temperature increase from the manifold temperature to the temperature at IVC and cannot be neglected or approximated to a fixed value. An iterative method for determining the residual gas fraction x_r as well as the residual gas temperature T_r is suggested in [9]. The following relationships are used

$$x_r = \frac{1}{r_c} \left(\frac{p_{em}}{p_{im}} \right)^{1/\gamma} \left(1 + \frac{q_{in}}{c_v T_1 r_c^{\gamma-1}} \right)^{-1/\gamma} \quad (3.7)$$

$$q_{in} = \frac{1 - x_r}{1 + \lambda(A/F)_s} q_{LHV} \quad (3.8)$$

$$T_r = \left(1 + \frac{q_{in}}{c_v T_1 r_c^{\gamma-1}} \right)^{1/\gamma} \quad (3.9)$$

$$T_1 = x_r T_r + (1 - x_r) T_{im} \quad (3.10)$$

where q_{in} is the specific heat supplied to the system and T_1 is the temperature at the start of the intake stroke.

The iterative algorithm for determining the residual gas fraction and temperature works as following.

1. Set $x_r = 0$ as the initial value.
2. Use (3.8) to find the initial value of q_{in} .
3. With the initial conditions determined, use (3.7), (3.8), (3.9) and (3.10).
4. Repeat the previous step until the value of x_r and T_r converges, i.e. the values do not continue to change when the process is repeated.

This method quickly converges to the final value if the initial guess for the residual gas fraction is close enough to the final value.

3.3.2 Expansion

The expansion phase is just like the compression phase modelled as a polytropic process with exponent k_e

$$p_e = p_3 \left(\frac{V_3}{V(\theta)} \right)^{k_e} \quad (3.11)$$

$$T_e = T_3 \left(\frac{V_3}{V(\theta)} \right)^{k_e-1} \quad (3.12)$$

p_3 , V_3 and T_3 refer to the state 3 in the ideal Otto cycle, [8]. The temperature increase associated with the combustion is going to determine the resulting pressure p_3 . From state 2 to 3 in the ideal Otto cycle the temperature increase can be expressed as

$$T_3 = T_2 + T_{comb}$$

where the temperature increase is defined as

$$T_{comb} = \frac{m_f Q_{LHV} \eta_f(\lambda)}{c_v m_{tot}} \quad (3.13)$$

and the fuel conversion efficiency is expressed as

$$\eta_f(\lambda) = 0.95 \min(1, 2\lambda - 0.2) \quad (3.14)$$

The pressure at state 3 can then be expressed as

$$p_3 = p_2 \frac{T_3}{T_2} \quad (3.15)$$

where $p_2 = p_c(\theta_{soc})$ and $V_3 = V(\theta_{soc})$.

Determining start of combustion (SOC)

Start of combustion is an important parameter to determine in order to accurately capture the combustion behaviour. The underlying processes that affect ignition delay are highly complex. The correlation presented in [2] is used to estimate the ignition delay. The ignition delay is described as

$$t_{id} = 3.45 p^{-1.02} \exp\left(\frac{2100}{T}\right) \quad (3.16)$$

where p and T are the pressure and temperature at the start of injection. This relationship is fairly simple but has shown to give a good estimation of the ignition delay. When the ignition delay has been determined the angle at start of combustion can be found using

$$\theta_{soc} = \theta_{soi} + 2\pi t_{id} \frac{N}{60}$$

where N is the engine speed.

3.3.3 Valve Model

Modelling the pressure changes that occur when the valves open and closes can be done using an interpolation cosine function

$$x_i(\theta, \theta_0, \theta_1) = 0.5 \left(1 - \cos \left(\pi \frac{\theta - \theta_0}{\theta_1 - \theta_0} \right) \right), \quad \theta \in [\theta_0, \theta_1] \quad (3.17)$$

This interpolation function is used when the intake valve is closing, during the overlap period when exhaust valve is closed and the intake valve is open, as well as when the exhaust valve opens.

3.3.4 Summarizing the Cylinder Pressure Model

Summarizing all the expressions for the different parts of the cycle, the full pressure model for one cycle takes the form

$$\begin{aligned}
 p_{cyl} = & \\
 p_{im} & , \theta_{evc} \leq \theta < \theta_{int} \\
 p_{im}(1 - x_i(\theta, \theta_{int}, \theta_{ivc})) + p_c(\theta)x_i(\theta, \theta_{int}, \theta_{ivc}) & , \theta_{int} \leq \theta < \theta_{ivc} \\
 p_c(\theta) & , \theta_{ivc} \leq \theta < \theta_0 \\
 p_c(\theta)(1 - x_b(\theta)) + p_e(\theta)x_b(\theta) & , \theta_0 \leq \theta < \theta_{evo} \\
 p_e(\theta)(1 - x_i(\theta, \theta_{evo}, \theta_{exh})) + p_{em}x_i(\theta, \theta_{evo}, \theta_{exh}) & , \theta_{evo} \leq \theta < \theta_{exh} \\
 p_{em} & , \theta_{exh} \leq \theta < \theta_{ivo} \\
 p_{em}(1 - x_i(\theta, \theta_{ivo}, \theta_{evc})) + p_{im}x_i(\theta, \theta_{ivo}, \theta_{evc}) & , \theta_{ivo} \leq \theta < \theta_{evc}
 \end{aligned} \quad (3.18)$$

The cylinder pressure results in a gas torque, which can be expressed as

$$T_{gas} = (p_{cyl} + p_{crank})A_{cyl}L(\theta) \quad (3.19)$$

where p_{crank} is the pressure in the cylinder on the bottom side of the piston, A_{cyl} is the cylinder's cross sectional area and $L(\theta)$ is the momentary lever arm.

3.4 Parameter Estimation

There are a number of parameters that can not be determined with analytical expressions and therefore need to be estimated. Many of these parameters belong to the combustion process and the Wiebe function. Parameters like intake pressure, intake temperature and SOI were available from a range of operating points from test data but values for the operating points that were not available from test data needed to be estimated.

The test data that was used for parameter estimation was measured at four different engine speeds, 800, 1200, 1600 and 2000 rpm. For each engine speed six tests were performed with varying load, which means that a total of 24 different operating points were available. The available data from the measurements can be seen in Table 3.1.

Measured Data
CAD
cylinder pressure
rpm
SOI
EOI
intake pressure
lambda
exhaust temperature
inlet temperature
fuel amount
exhaust pressure
pilot injection
engine temperature

Table 3.1: The available data from measurements

In order to estimate a parameter value for an arbitrary operating point, the data was sorted in tables based on engine speed and load (fuel amount) as a look up parameter. The sorted table of a look-up parameter can be seen in Table 3.2. The estimated value of the parameter is interpolated from the table based on the input engine speed and fuel amount. Note that data was not available for 500 rpm, but an estimation of the parameter was needed for that speed as it is roughly the value of the idling engine speed. For some look-up parameters the values for 500 rpm were set to the same as for 800 rpm, whilst others were linearly extrapolated from the values at 800 rpm.

Engine Speed [rpm] /Load [mg/s]	op 6	op 5	op 4	op 3	op 2	op 1
500 rpm	$p_{1,1}$	$p_{1,2}$	$p_{1,3}$	$p_{1,4}$	$p_{1,5}$	$p_{1,6}$
800 rpm	$p_{2,1}$	$p_{2,2}$	$p_{2,3}$	$p_{2,4}$	$p_{2,5}$	$p_{2,6}$
1200 rpm	$p_{3,1}$	$p_{3,2}$	$p_{3,3}$	$p_{3,4}$	$p_{3,5}$	$p_{3,6}$
1600 rpm	$p_{4,1}$	$p_{4,2}$	$p_{4,3}$	$p_{4,4}$	$p_{4,5}$	$p_{4,6}$
2000 rpm	$p_{5,1}$	$p_{5,2}$	$p_{5,3}$	$p_{5,4}$	$p_{5,5}$	$p_{5,6}$

Table 3.2: The general view of the look-up tables that were used where $p_{i,j}$ is an arbitrary parameter. **op 1 to op 6** represent the different load operating points were a higher number indicates a lower load.

3.4.1 Manifold Pressures and Temperatures

Using the measurement data seen in Table 3.1 manifold pressures and temperatures were fitted to a specific operating point using the look-up table principle shown in Table 3.2. Since data from manifold pressure and temperature were already available from measurements, the values were simply sorted by engine

speed and fuel amount and could then be interpolated as Figure 3.2 shows.

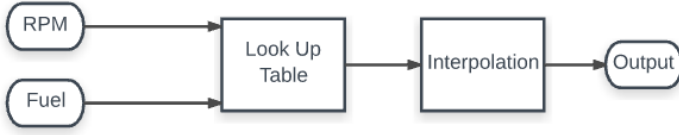


Figure 3.2: The basic principle used for estimating manifold pressures and temperatures

3.4.2 Combustion Parameters

From equation (2.2) the unknown parameters that are to be determined are the burn duration angle $\Delta\theta$, a , m and θ_{soc} . [6] suggests five different methods for fitting Wiebe parameters. The parameters are determined using least squares fitting method and direct algebraic solutions.

Burn duration angles The burn duration angle is often described as $\Delta\theta = \theta_{90} - \theta_{10}$, which is the difference between when 90% and 10% of the fuel is burnt. A profile for the mass fraction burned needs to be available to find these angles. The work done in [13] suggests a method for estimating the mass fraction burned. It suggests that the pressure increase between two samples can be expressed as the sum of the pressure increase due to combustion and the volume change

$$\Delta p = \Delta p_c + \Delta p_v \quad (3.20)$$

The pressure associated with the volume change, often denoted as the motored pressure, is the pressure in the cylinder when no combustion takes place. When no combustion takes place the motored pressure between two samples i and $i + 1$, can be represented as

$$\Delta p_v = p_i \left(\left(\frac{V_i}{V_{i+1}} \right)^n - 1 \right) \quad (3.21)$$

where the polytropic exponent $n \in [1.25, 1.35]$. From equation (3.20) the pressure increase from the combustion can be solved.

The relationship between the burned mass fraction and the pressure ratio is described as

$$x_b(i) = \frac{m_b(i)}{m_b(\text{total})} = \frac{\sum_0^i p_c}{\sum_0^M p_c} \quad (3.22)$$

where i can analogously be switched to crank angle θ or time. M is the number of total samples that exists within the burn duration interval. The burn duration

angles θ_{10} and θ_{90} are found for

$$x_b(\theta_{10}) = 0.1$$

$$x_b(\theta_{90}) = 0.9$$

Tuning parameters a and m

If the burn duration angles are available the parameters m and a can be calculated by algebraic expressions

$$m = \frac{\ln\left(\frac{\ln(0.1)}{\ln(0.9)}\right)}{\ln\left(\frac{\theta_{90}-\theta_{soc}}{\theta_{10}-\theta_{soc}}\right)} - 1$$

$$a = -\ln(1 - 0.1) \left(\frac{\theta_{90} - \theta_{10}}{\theta_{90}}\right)^{m+1}$$

However results during implementation did not show good agreement with the measured pressure data using these expressions. Instead the parameter a was set to a fixed value of $a = 5$, as this value have been previously used in [10]. For each operating point the value of m was optimized using a least squares fit.

Each of the combustion parameters θ_{10} , θ_{90} and m were thereafter sorted by engine speed and fuel amount to be used as a look-up parameter as in Figure 3.2.

Table 3.3 shows the parameters that were used and are based on look-up tables as well as how the table values were found.

Parameter	Table Data
SOI	From measurements
Intake Pressure	From measurements
Exhaust Pressure	From measurements
Intake Temperature	From measurements
Exhaust Temperature	From measurements
wiebe parameter m	Fitted with least squares method
Ca_{10}	Fitted with least squares method
Ca_{90}	Fitted with least squares method

Table 3.3: A summary of all the parameters that were based on look-up tables

3.5 C-bal Controller

The C-bal controller consist of six separate PI-controllers, one for each cylinder. In a real engine the controller would not have access to such a high resolution rpm signal that the simulation environment could generate. In a production engine setup, the rpm sensors readings are based on the number of teeth on the flywheel. There are 60-2 teeth on the flywheel, where the 2 indicates the missing teeth that are used to detect start and end of one revolution. This means that the sensor will

receive a pulse reading every 6 degrees. Therefore, the rpm signal generated in the simulation environment is first modified to resemble the real sensor readings.

The control algorithm for cylinder balancing is based on a similar approach as described in [12]. There are however some differences to how the imbalance is measured. The cylinder imbalance is calculated based on the deviation of each pulse amplitude from the mean pulse amplitude. More specifically the imbalance is expressed as vectors containing the minimum and maximum values from each pulse. Three values are averaged for the minimum and maximum value of each pulse. Figure 3.3 illustrates roughly which values that are chosen.

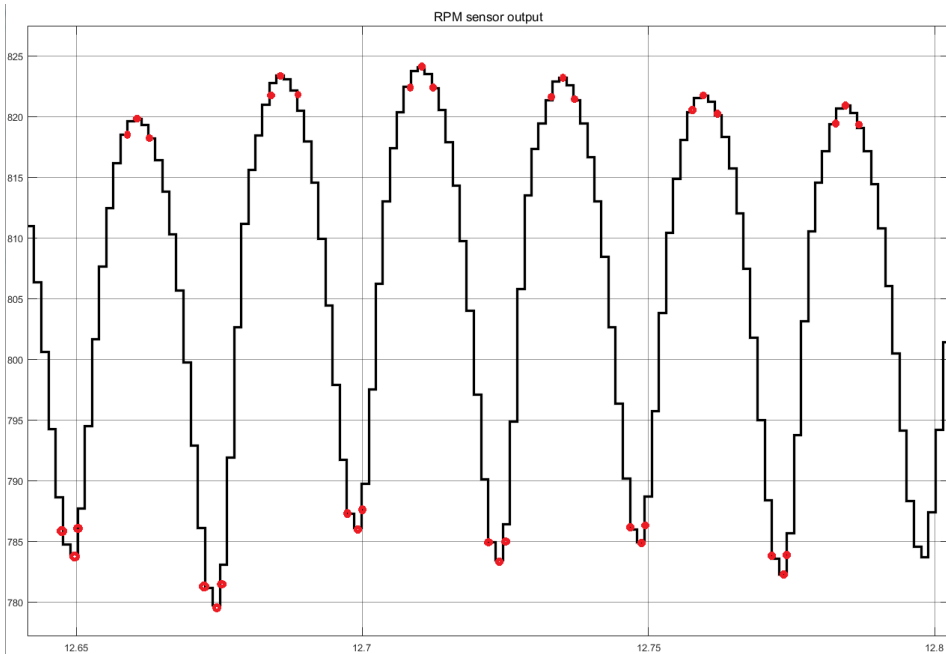


Figure 3.3: The raw rpm values detected by the flywheel sensor together with the values (in red) that are used to measure the imbalance. Note that the signal is not based on measurements but is reconstructed from the simulation framework.

The difference between the minimum and maximum are calculated and the mean value of all the differences are subtracted from each term. The resulting values are the imbalance in each cylinder. The control tuning parameters K_P and K_I were manually tuned to give a sufficient response time but without risking overshoots that could cause instability issues. The control actuator was also limited to adjustments of ± 10 mg per cylinder. This is necessary to avoid engine damage and soot generation in the cylinders.

3.6 The simulation framework

The simulation framework was implemented in Simulink. It is an integrated environment of the cylinder pressure model, crankshaft dynamics and the PI cylinder balancing controller. Figure 3.4 gives an overview of the main model blocks and the signal flow between them. A number of other subsystems are also included and a thorough description of all the subsystems are covered in section 3.6.1. Section 3.6.2 gives a guide for using the simulation environment and the files that are included.

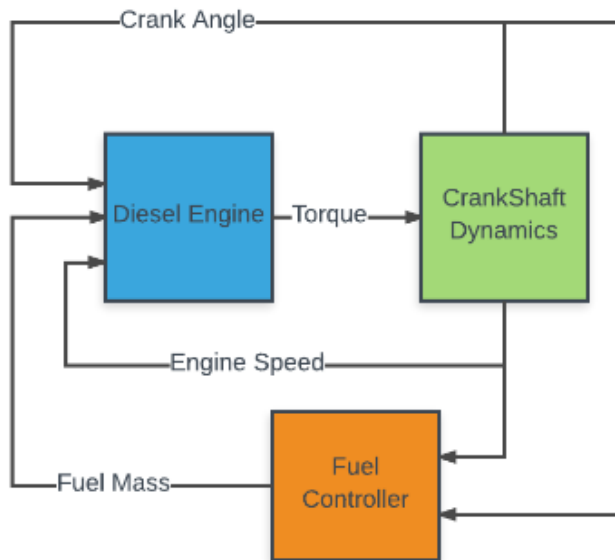


Figure 3.4: Flowchart of the signals and the main blocks in the simulation framework.

3.6.1 Subsystem Description

Figure 3.4 shows the top level view of the simulation framework. A thorough description of all the main model blocks and their belonging subsystems will be covered below. The listed names corresponds to the block names in the Simulink model.

1. **Diesel Engine** calculates all the in-cylinder processes, from injected fuel mass to pressure and then to torque.
 - 1.1. **In-Cylinder Processes** generates the pressure trace of each cylinder.

- 1.1.1. **Pressure Calculations** calculates the pressure trace for the upcoming engine cycle. The block is enabled once every engine cycle, at the start of a new cycle. Injector fuel offsets are added to the respective cylinder and parameters that are depending on fuel amount and rpm are calculated using look up tables and used as inputs to the pressure calculations. A Matlab function block calculates all the six pressure vectors in one time step.
 - 1.1.2. **Pressure Index from CAD** uses the crank degree for each cylinder to select which index from the pressure vector that are to be delivered at each time step.
 - 1.2. **Look Up Torque Inputs** The fuel pump mass flow, rail pressure and oil temperature is needed in some of the torque models and these parameters are depending on the fuel amount and the engine speed. The parameters are calculated from look up tables based on test data.
 - 1.3. **Torque Calculations** The torque contributions are gas torque from the cylinder pressure, mass torque from the reciprocating masses inside the cylinder, friction torque and load torque. The output torque components are represented as a vector of eight elements, where each torque act on a specific node on the crankshaft. For a more detailed explanation of how the torque is distributed the reader is referred to [16].
2. **Crankshaft Dynamics** Performs the calculations of CAD and engine speed based on the torque output from the Diesel Engine. This block can be simulated either using a rigid crankshaft or a non-rigid crankshaft model through a manual switch.
 - 2.1. **Torsion** is used when simulating a non-rigid crankshaft. It is a state-space block that from the input torque vector calculates the displacement (CAD) and the angular speed at each node. The matrices used in this block is calculated beforehand by the Matlab script `TorsionSS`.
 - 2.2. **Rigid** is used when simulating a rigid crankshaft. Unlike the torsion block that uses the torque vector to calculate the respective CAD and speed at each node, this block uses the sum of all torque components to calculate the engine speed.
 - 2.3. **calculate CAD** integrates the angular speed at the flywheel and output the CAD.
3. **Fuel Controller** is responsible for calculating the fuel mass input to the Diesel Engine for each cylinder. It contains one block that is the rpm controller, which computes the main fuel supply to the system. The cylinder balancing controller computes the individual adjustments to each cylinder and its output is added to the total fuel supply.
 - 3.1. **RPM Controller** is a PI-controller responsible for adjusting the overall fuel amount so that the desired engine speed can be kept. During the first few seconds of the simulation, the average RPM value will be

calculated with some inaccuracies until the RPM signal has generated a sufficient amount of values. To minimize this effect, the initial fuel value is set to be constant for the first second and thereafter the PI-controller makes the necessary adjustments. The initial value has been determined manually for each operating point and is an interpolated look-up table based on load torque and engine speed.

- 3.2. **C-bal Controller** is the block that calculates the cylinder imbalance and performs the cylinder balancing control. The cylinder balancing controller can be activated in two ways through a manual switch. Either it is activated when the rpm signal has reached a steady state value, alternatively it is activated at a specific time during the simulation. The rpm signal is manipulated to be sampled at each 6 degrees, as this is the current sampling frequency in the engine. This is the signal that the controller will use to measure the imbalance.
 - 3.2.1. **findMin/findMax** uses three different crank angle values from the minimum and maximum value of each rpm pulse. The three values are averaged and used to calculate the minimum and maximum peak of each pulse. These values are stored in two vectors and are used to calculate the imbalance.
 - 3.2.2. **Calculate Imbalance** is responsible for calculating the imbalance based on the min and max vectors.
 - 3.2.3. **Controller Output Trigger** This block keeps track of the current position on the flywheel. Since the control signal should only deliver its control signal at the end of each engine cycle, the trigger signal is set to true when the CAD reaches 719 degrees. This is to make sure that the imbalance is only based on measurements from the same cycle.
 - 3.2.4. **PI-Controller** contains six PI controllers, each responsible for controlling its respective cylinder. This block is enabled by an external trigger signal that detects when the rpm signal is within a steady state operating range. This is to ensure that the controller only operates at steady state. In order to not receive any faulty imbalance values before one engine cycle has been simulated, the imbalance is set to zero up until one engine cycle is completed.
 - 3.2.5. **Subtract Mean and Scale** subtracts the mean value of the fuel adjustment from the PI-controllers to ensure that the sum of all adjustments are equal to zero. A scaling factor is also calculated to ensure this in case one adjustment reach the output boundaries. This block is enabled by an external trigger signal. The control signal for the current cycle is based on the calculated imbalance from the previous engine cycle.
- 3.3. **Add Fuel Adjustments** adds the fuel adjustments from the C-bal controller to the main fuel amount. It is of importance to notice that the fuel adjustments are added to each cylinder based on the engines firing order. The pulse order will therefore not necessarily match the

corresponding cylinder order.

3.6.2 Simulation user manual

There are two main files that are used when running the simulation. `testSetup` is the Matlab script that initializes the simulation parameters and enables the user to manually configure the test setup. Table 3.4 gives a description of the different configurable test parameters.

Parameter	Description	Input	Unit
<code>logStart</code>	Determine when to start logging data	double	s
<code>logEnd</code>	Determine when to stop logging data	double	s
<code>rpm</code>	The target engine speed	double	rpm
<code>loadTorque</code>	The load torque on the engine	double	Nm
<code>load_amp</code>	The amplitude of the load disturbance	double	Nm
<code>loaf_freq</code>	The frequency of the load disturbance	double	Hz
<code>fuel_offset</code>	The injector fuel offset vector	double	%
<code>sensor_err</code>	The sensor error in maximum pulse value	double	%
<code>switchOn</code>	C-bal controller is activated at this time	double	s

Table 3.4: The parameters that are used to set up the test.

The other file is `simulation_model`, which is the Simulink model that runs the simulation. Below is a step by step guide on how to use the simulation framework.

Simulation User Guide

1. Open up Matlab
2. Go to the folder `2017_Simulation_Framework_C-bal_Niclas_Lindström` and then into the folder `Matlab` and then into the folder `simulation framework`.
3. Open the file `testSetup`
4. The user can now scroll down to the parameters that belong to test setup and manually adjust them. As a default, the parameters that are being logged are the engine speed signal, the cylinder imbalance, the fuel adjustments from the C-bal controller and the trigger signal that shows when the controller is activated.
5. Choose values for when to start and stop data logging.
6. If load disturbance is chosen, assign values for the load disturbance parameters. If no load disturbance is used, assign `load_amp` to 0.

7. Choose whether to use a fuel offset or not. The fuel offset is given as an input vector of size 1×6 and represents the offset in each injector. For example, the input `[0 -5 0 10 0 0]` suggests that cylinder 2 will inject 5% less and cylinder 4 will inject 10% more than the desired amount. The other cylinders inject the correct amount.
8. If choosing to use sensor error, the percentage offset can be adjusted. The sensor error represents an offset in maximum pulse value and is represented as a vector, just as the fuel offset. For example, `[5 0 0 0 0 0]` means that the maximum pulse value of cylinder 1 is increased by 5%.
9. Set the parameter `switchOn` to the desired value. Note that this value is unused if the controller is set to be activated by the rpm signal by the manual switch in the Simulink subsystem block **C-bal Controller**.
10. Run the script and when the message `Ready to run simulation` is displayed the user can go to the simulink model `simulation_model` and run the simulation.
11. When the simulation has finished, the logged parameters can be plotted by scrolling down in file `testSetup` and run the section `Plot Data`.

4

Results and Discussion

This chapter is divided into two distinct parts. The first section will cover the results associated with the modelling implementation. The second section will cover the results from the controller tests in the complete simulation framework.

4.1 Modelling Validation

The modelling accuracy will be determined by how well the modelled pressure will match the pressure data for the different operating points. For presentation purposes the pressure trace from four different operating points will be shown. The engine speed signal is validated against measured engine speed data and its capacity is also tested using the rigid and the non-rigid crankshaft for two different engine speeds. The model will also be judged on its ability to show a different behaviour when changing SOI or changing EOI. Lastly parameter estimations that were done are represented as 3-dimensional graphs to verify that the interpolation method can be justified.

4.1.1 Pressure Validation

The pressure model is validated using the measurement data from Table 3.1 for 800, 1200, 1600 and 2000 rpm. For each operating point the model parameters such as intake/exhaust pressure, λ -value, fuel amount etc. were set equal to the corresponding data from Table 3.1. The pressure data and the modelled pressure have been scaled in order to be publishable in this report.

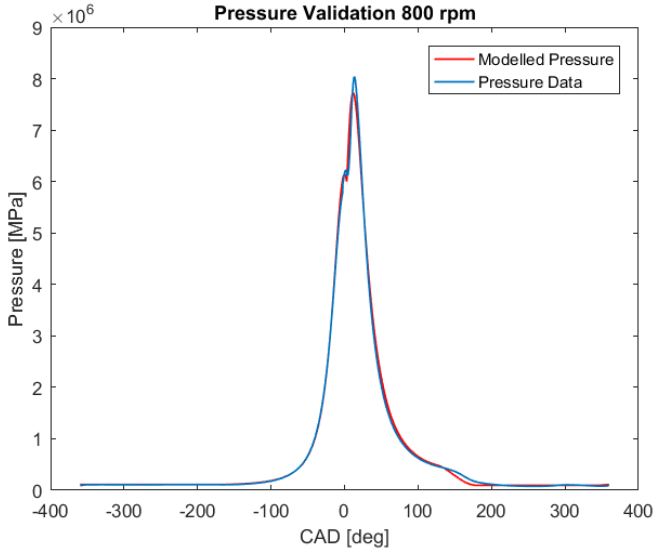


Figure 4.1: Comparison between the modelled pressure and the measured pressure for 800 rpm and 175 mg/stroke

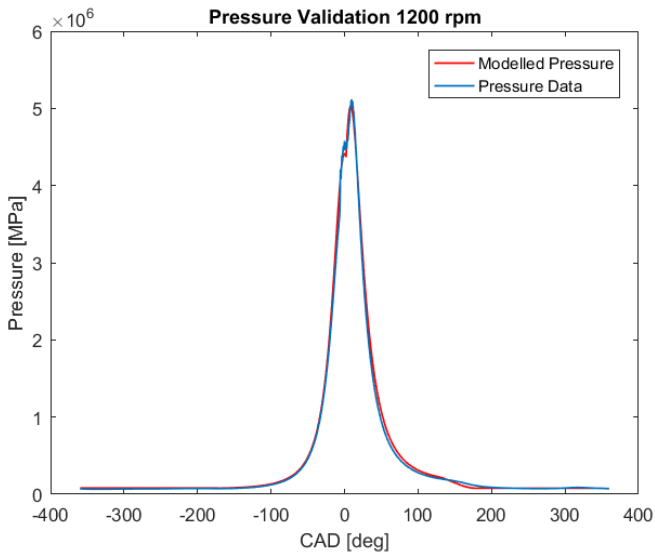


Figure 4.2: Comparison between the modelled pressure and the measured pressure for 1200 rpm and 52 mg/stroke

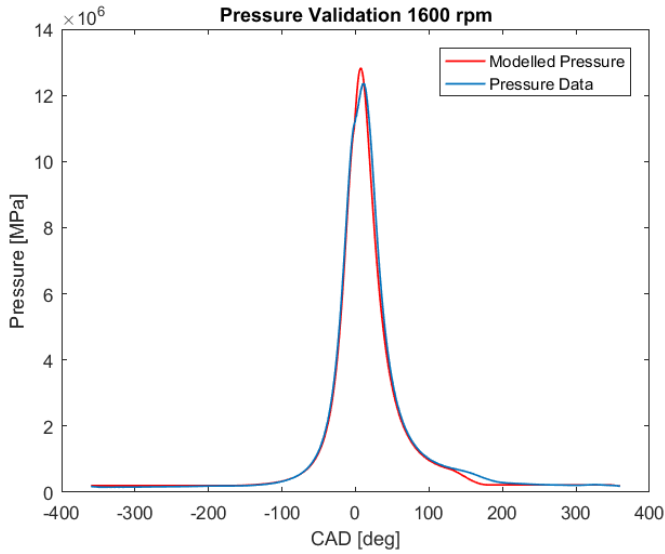


Figure 4.3: Comparison between the modelled pressure and the measured pressure for 1600 rpm and 249 mg/stroke

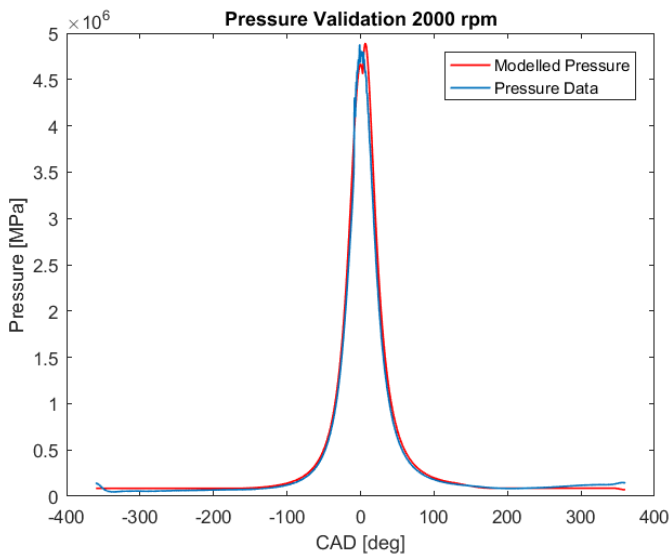


Figure 4.4: Comparison between the modelled pressure and the measured pressure for 2000 rpm and 9 mg/stroke

For each of the four different pressure plots the modelled pressure is able to match the measured pressure with good accuracy. At lower loads the measured pressure around top dead center has a rather noisy behaviour. Since the combustion parameters are fitted against the measured data for each individual operating point, the estimation of these parameters become less accurate at low loads and consequently this can be seen in the modelled pressure. At medium to higher loads the accuracy around top dead center is improved. However between 140 to 200 degrees, which is when the exhaust valve is opened, the model accuracy is decreased. This suggests that the cosine function in equation (3.17) is not as accurate for higher loads.

4.1.2 Changing SOI and EOI

To verify that the maximum pressure can not only be controlled by changing the fuel amount but also through changing the start of injection, the pressure is modelled for two different SOI. In both cases the fuel mass is the same and all the other parameters are set equal.

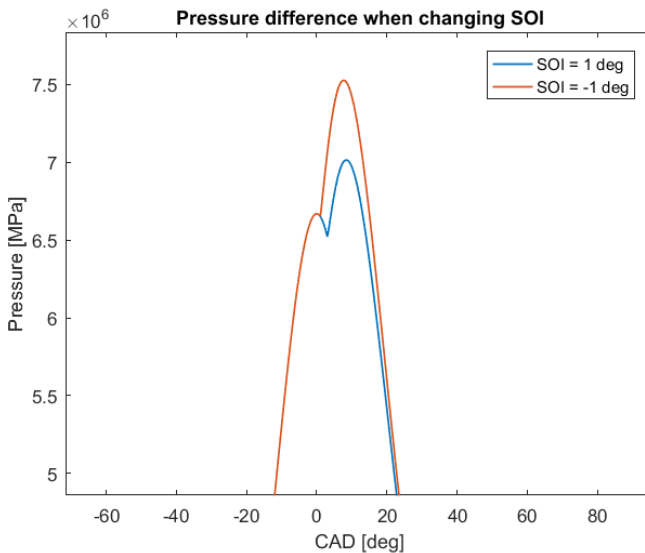


Figure 4.5: The modelled pressure for two different SOI. In both cases the fuel mass was set to 100 mg/stroke and the engine speed was 1200 rpm.

Figure 4.5 shows that the model gives different behaviour when changing SOI whilst keeping the fuel mass the same. This is to be expected as SOI determines SOC which is a parameter that is sensitive to changes in the Wiebe function. When SOI is advanced the combustion initiates earlier which results in a higher maximum pressure. The result would be similar to increasing the fuel mass for a fixed SOI, only that an increased fuel mass suggests a delayed EOI which would prolong the burning process and result in a wider pressure curve.

4.1.3 Engine Speed Signal

The engine speed is first validated against measurement data for 800 rpm. The measurement data is from an engine idling, without any drive train connected to it. The signal is not affected by any cylinder balancing control, which leads to that the signal can include imbalances from a variety of causes. In order to recreate such an imbalance, the modelled engine speed signal is provoked with fuel offset in some of the cylinders in order to mimic the behaviour in the measured signal. The modelled and the measured signal is plotted for one complete engine cycle, i.e. 720 degrees. The modelled engine speed was simulated for a rigid crankshaft.

The engine speed signal is also tested for 800 rpm and 1600 rpm, using both the rigid and non-rigid crankshaft. In this case, no disturbances are affecting modelled engine speed signal

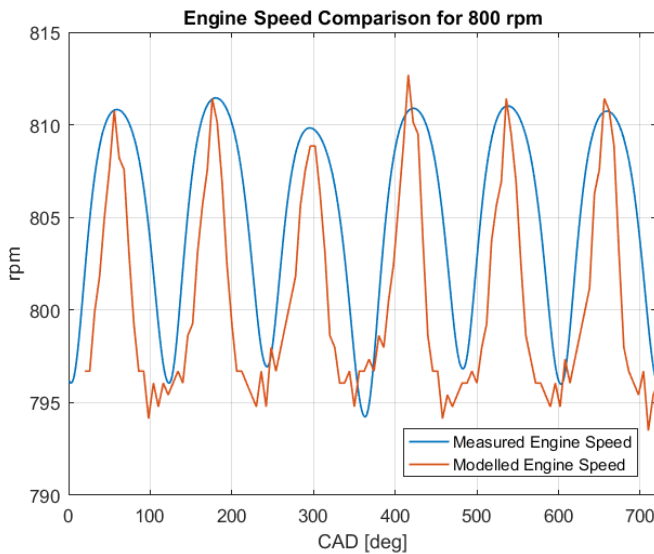


Figure 4.6: Comparison between the modelled and measured engine speed signal for 800 rpm.

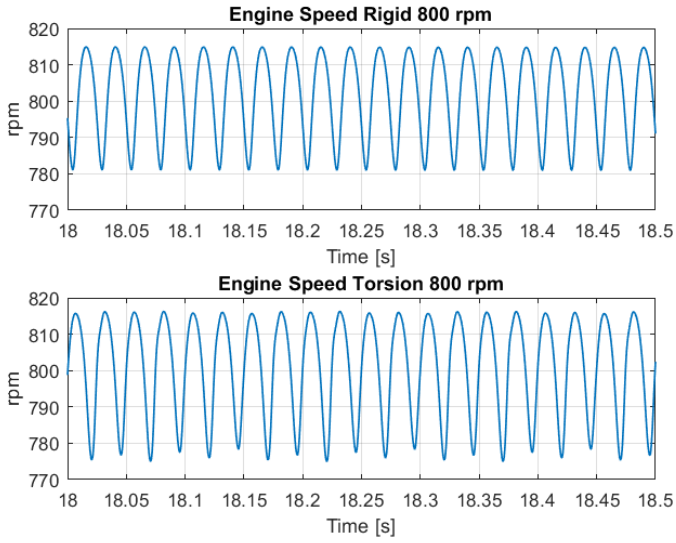


Figure 4.7: The engine speed signal for 800 rpm for rigid mode and torsion mode

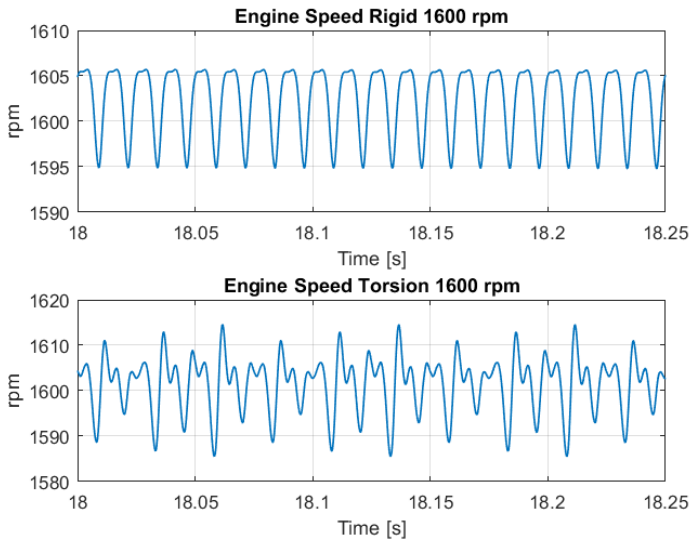


Figure 4.8: The engine speed signal for 1600 rpm for rigid mode and torsion mode

The modelled engine speed from Figure 4.6 was deliberately provoked with a fuel error in order to resemble the behaviour of the measured signal. The measured engine speed contains quite a bit of noise and does not have the same high resolution that the modelled signal can offer. One can however see when comparing the two signals that the behaviour is still similar. Cylinder 3 (which also has fire order 3) is slightly lower maximum pulse value than the rest of the pulses. The amplitude of the pulses are fairly similar between the modelled and the measured signal. The two signals have been plotted in such a way that the phasing between them is as small as possible. One can note that the frequency of the pulses are similar and that both the modelled and the measured signal have completed one full engine cycle at 720 degrees.

From Figure 4.7 one can see the difference between the engine speed signal when using a rigid crankshaft compared to when including torsion. When using a rigid crankshaft, the pulses have an equal amplitude. For the non-rigid crankshaft the pulses slightly deviate in amplitude. However, both signals show well defined pulses without any noise and the average amplitude of the pulses are similar.

When engine speed increases as can be seen Figure 4.8, the signal shows a different behaviour. This is particularly evident for the non-rigid crankshaft, where the pulses have become largely distorted. For the rigid crankshaft the pulses are still well defined and mostly unaffected by the increased engine speed. One can however spot a slight flattening effect on the top part of pulse, compared to the results in Figure 4.7

4.1.4 Parameter Estimation

The estimated look-up parameters are here represented as 3-dimensional graphs. Each parameter is a function of fuel mass and engine speed.

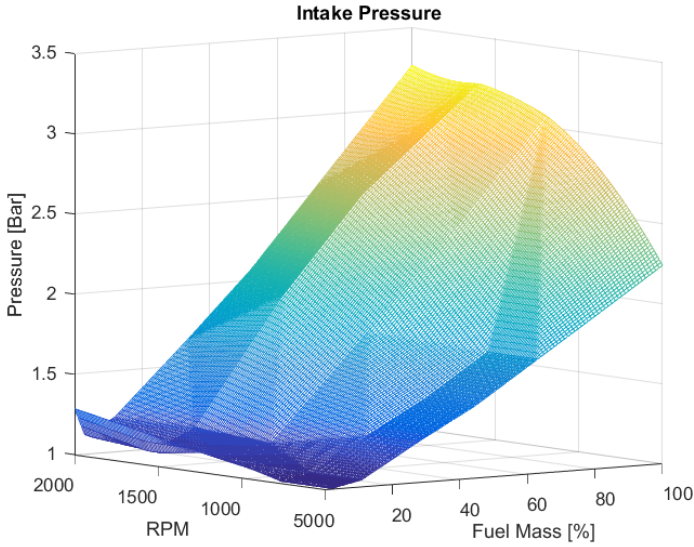


Figure 4.9: Interpolation map for the intake pressure as a function of fuel mass and engine speed

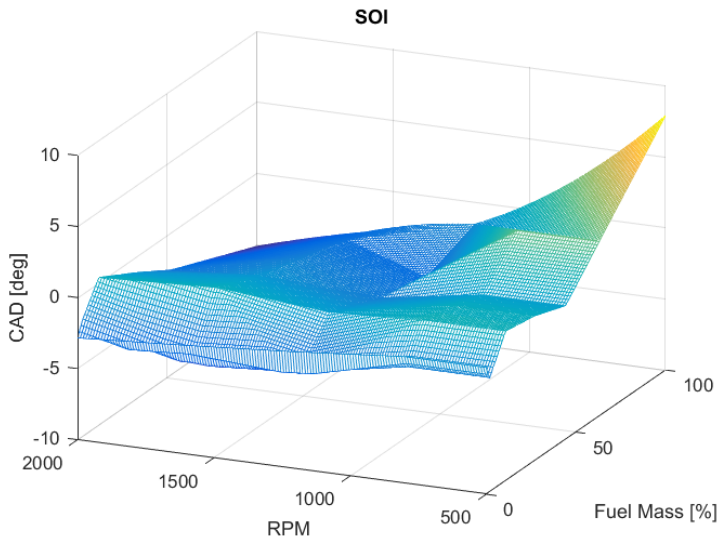


Figure 4.10: Interpolation map for SOI as a function of fuel mass and engine speed

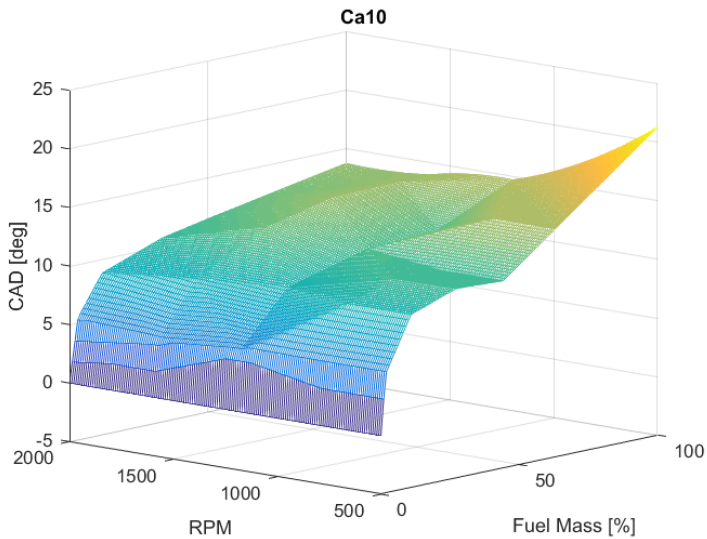


Figure 4.11: Interpolation map for burn duration angle θ_{10} as a function of fuel mass and engine speed

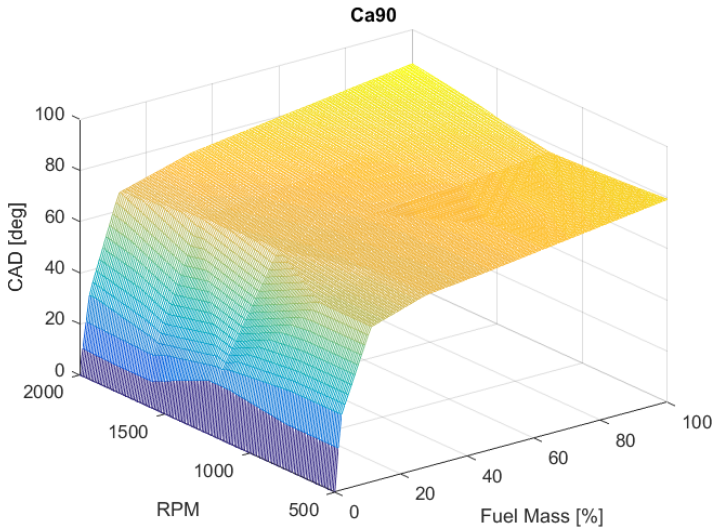


Figure 4.12: Interpolation map for burn duration angle θ_{90} as a function of fuel mass and engine speed

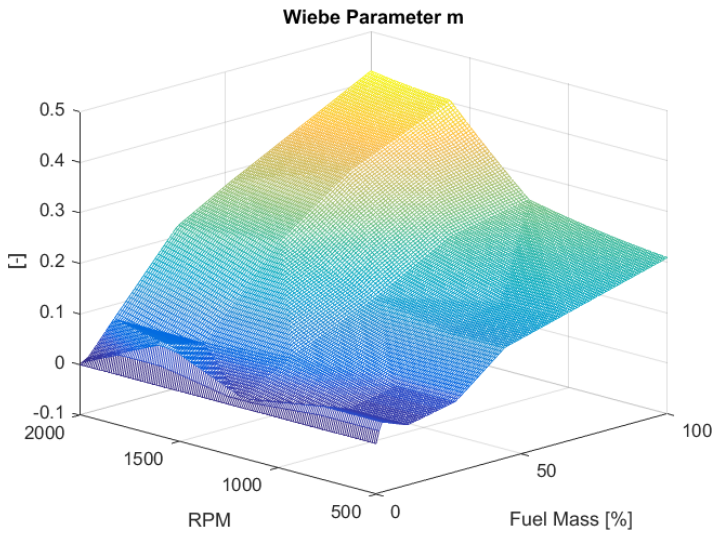


Figure 4.13: Interpolation map for Wiebe parameter m as a function of fuel mass and engine speed

The interpolated maps represents a surface which suggests that estimating the parameters based on engine speed and load is a valid approach. The surfaces are not completely smooth though which indicate that other variables besides engine speed and load have a definite effect. The intake manifold pressure in Figure 4.9 for example shows a rather linear behaviour at high loads and high engine speed while non-linearities are more predominant at lower speeds and loads. The air charge is typically more steady as engine speed and load increases which supports why the model has better accuracy at these operating points.

The SOI is typically mapped based on load and engine speed on a real engine, which implies the method is appropriate to use for this parameter. In Figure 4.10 the SOI occurs earlier for higher speeds and later for lower speeds. This is because when engine speed increases the ignition of the fuel will occur later in the cycle, which means that timing has to be advanced.

The burn angles θ_{10} and θ_{90} from Figure 4.11 and 4.12 grow rapidly between 0 to 30% load. Between 30 to 100% load the duration time increases much slower.

The Wiebe parameter m in Figure 4.13 has the largest value at higher speeds and loads. A larger value implies a delayed combustion [9], which is what to be expected at higher engine speeds.

4.2 Controller results

In this section the results for the controller tests are presented. The tests were designed to evaluate the controller performance on different operating points for four different types of disturbances, injected fuel error, two types of load disturbances and a speed sensor disturbance. Because of certain restrictions in the engine speed quality when using the non-rigid crankshaft, the controller tests were only carried out for a rigid crankshaft. Each plot shows the engine speed signal, fuel adjustments from the controller, the imbalance in each cylinder as well the trigger signal which activates the controller.

4.2.1 Disturbance in injected fuel

Here the fuel injector for cylinder 2 is set by default to inject 20% more fuel to produce the imbalance. The engine is idling at 500 rpm, meaning that the load torque is 0 Nm. The controller is activated at 25 seconds and makes the necessary adjustments for correcting the imbalance.

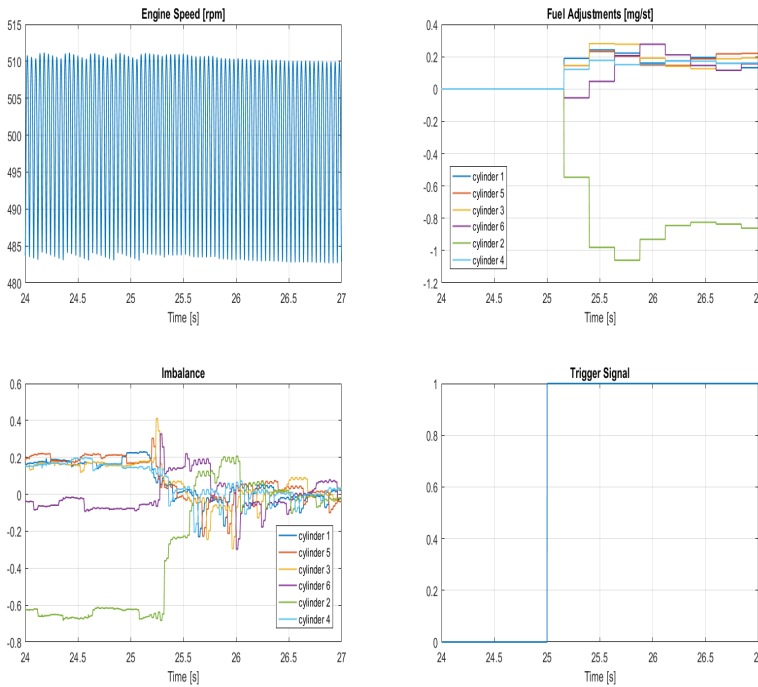


Figure 4.14: Test results for the controller using disturbance in injected fuel amount. Cylinder 2 injects 20% more fuel than desired.

The controller is able to quickly minimize the engine speed error, which can be seen both from the engine speed signal and the imbalance. The fuel adjustments reaches relatively stationary values at around 26 seconds. As the disturbance is a constant percentage deviation of the desired fuel amount, it is not affected by any variables other than the commanded fuel amount from the controller. This enables the controller to individually adjust the fuel amount to each cylinder until the imbalance is eliminated. The reason why the measured imbalance will never go down to zero is because the engine speed signal has a limited resolution and that the imbalance is based on sampled values on specified crank angles.

4.2.2 Disturbance in load torque

A disturbance in load torque would resemble a resonance frequency in the drive-line that propagates to the crankshaft. The test is carried out for two different cases. One case when the frequency of the load disturbance matches the frequency of the engine speed and another when the frequency does not match the engine speed frequency. The engine speed is running steady at 900 rpm. The load disturbance is modelled as a sine wave that is added to the static load torque act-

ing on the engine. In this scenario the static load was set to 1000 Nm and the load disturbance was set to have an amplitude of 30 Nm. In the first test the frequency was set to 15 Hz in Figure 4.15 to match the engine's rotational frequency, and in the second test it was set to $\frac{15}{\pi}$ Hz (to arbitrary scale the engine frequency) in Figure 4.16 as the non-matching frequency.

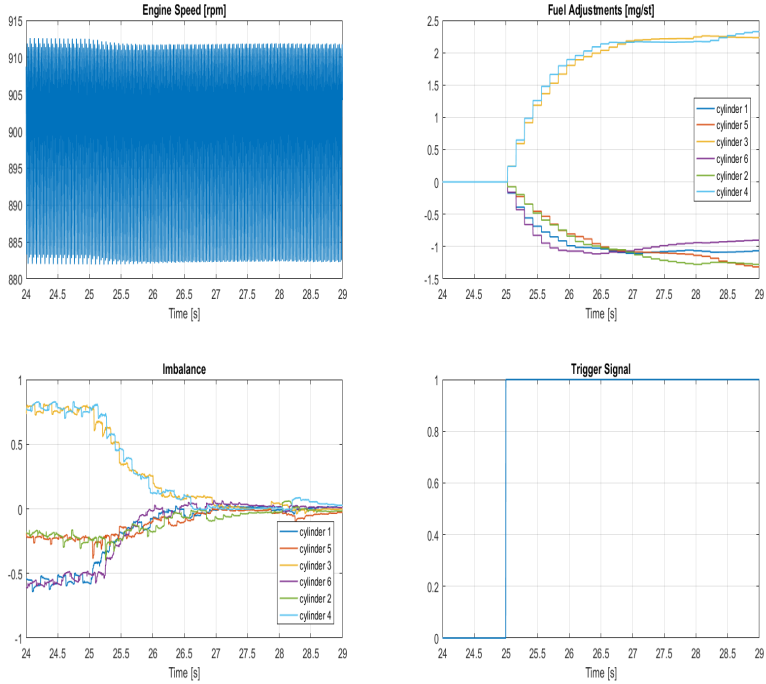


Figure 4.15: Test results for the controller using disturbance in load torque. The disturbance has an amplitude of 30 Nm and a frequency of 15 Hz, matching the frequency of the engine speed.

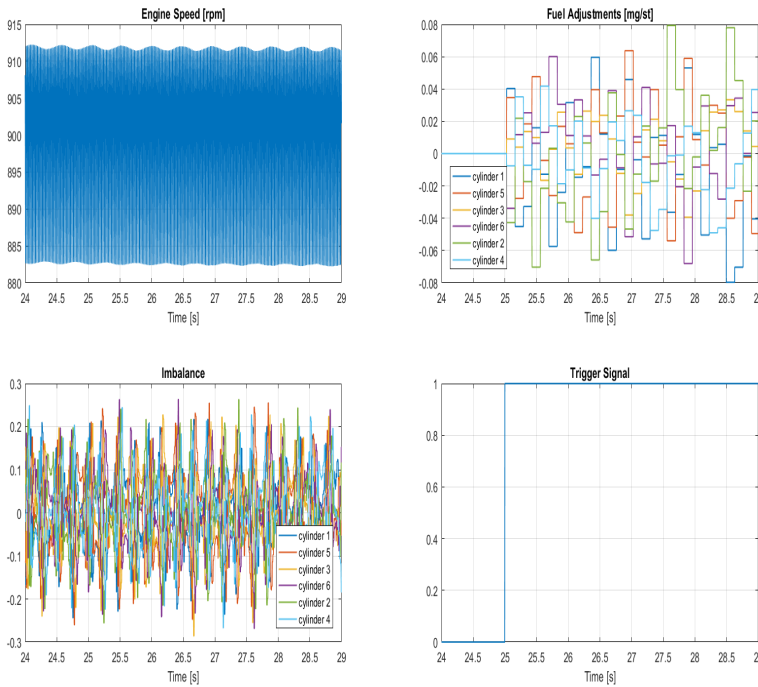


Figure 4.16: Test results for the controller using disturbance in load torque. The disturbance has an amplitude of 30 Nm and a frequency of $\frac{15}{\pi}$ Hz, which is not synced with the frequency of the engine speed.

It is clear that the controller performance is completely depending on the frequency of the load disturbance. When the frequency of load torque matches the frequency of the engine speed, as in Figure 4.15, the disturbance affects the amplitude of each pulse equally each engine revolution. The fuel adjustments calculated by the controller will dampen the effect of the load disturbance and minimize the imbalance. However when the load disturbance has another frequency, as in Figure 4.16, the amplitude of the pulses will not be the same for each engine cycle and any suggested corrections from the controller will not minimize the imbalance. There is even a possibility that the controller can cause instability in some instances if the disturbance is large enough.

4.2.3 Error in engine speed sensor

Here the engine speed is affected by a particular kind of sensor reading error. The maximum pulse value of cylinder 1 and cylinder 6 is increased by 5 % relative to the actual value. This sort of disturbance could resemble a manufacturing error on a specific region on the flywheel, causing sensor reading in this region to be

distorted. It is therefore necessary that two cylinders are going to be affected by such an error as one four stroke cycle is completed in two revolutions. Because of the ignition order 1-5-3-6-2-4, an error affecting cylinder 1 is going to also affect cylinder 6.

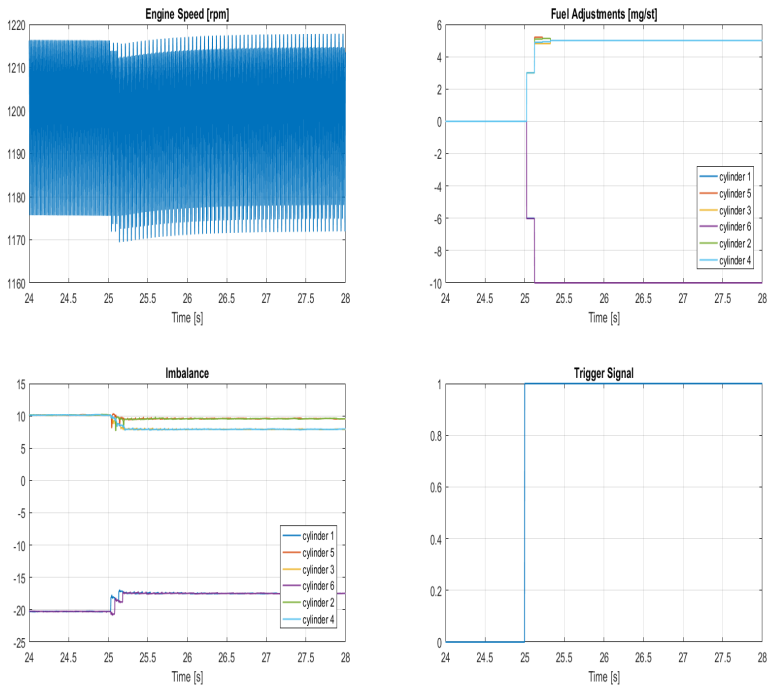


Figure 4.17: Test results for the controller using disturbance in sensor readings. The peak values of cylinder 1 and 6 are increased with 5 %.

The engine speed signal shown in Figure 4.17 is the actual correct signal, which is completely balanced. However the controller reads a quite significant imbalance due to the sensor reading errors. Note that since the imbalance is based on the mean value of the all the pulses, the cylinders which aren't affected by the disturbance are also going to be calculated as having an imbalance. When the controller is activated at 25 seconds, it attempts to make adjustments which cause the engine speed signal to become largely imbalanced. The imbalance is now affected by both the sensor error and the true imbalance that the controller has provoked when attempting its corrections. Such a disturbance will negatively affect the true engine speed signal when the controller is activated.

4.3 Discussion

Pressure Model The pressure model shows good accuracy when compared to the pressure data. The compression part of the curve, which is modelled as a polytropic process, is surprisingly accurate for all operating points. The ignition delay from equation (3.16) seems to also be able to give a good estimation of the start of combustion, which can be seen in end part of the compression stroke. Around TDC the modelled pressure lies close to the data. The accuracy of the combustion part is quite depending on operating point and load. The estimation of the combustion parameters from the Wiebe function seem to more accurately describe the combustion at medium loads and lower engine speeds. Since the combustion parameters are explicitly estimated from each operating point it is evident that any noise in the measured data will directly affect the accuracy of the estimation. Also since the pressure can only be validated against the accessible data, little can be known about the model accuracy in other operating points. The expansion part of the curve, is just like the compression modelled as a polytropic process and gives excellent results. However the dynamics of the exhaust valve open at around 120 degrees to 200 degrees is not a very good approximation at higher loads. This could possibly be tuned to better agreement by adjusting the angle at when the exhaust phase starts (θ_{exh}).

Engine Speed Signal Attempting to recreate the behaviour of the measured engine speed signal is quite hard. The main reason for this is that in real life, cylinder imbalances can be caused by a number of reasons which are very difficult to detect and measure. Because of this there is no way to completely validate the model's accuracy against measurement data. One can only attempt to show that a typical real life disturbance shows a similar behaviour as the measurement data. The comparison between the measured and modelled engine speed in Figure 4.6 tells that simulation framework is able to pretty accurately recreate the behaviour of such a disturbance.

The pulse amplitude is typically a result of the crankshaft acceleration. A larger pulse amplitude would therefore resemble a higher acceleration, i.e. a higher engine load. Since the measurements were logged when the engine was idling, the modelled signal was simulated for a load torque of 0 Nm. As can be seen when comparing the signals the amplitude of the pulses are still fairly close. The amplitude is also going to be directly affected by the accuracy of the inertia of the crankshaft, as it directly affects the acceleration, which can be seen from Equation (3.1). In reality the inertia is not constant throughout the engine cycle, but a function of crank angle. In this thesis the inertia of the crankshaft was set to be constant throughout the engine cycle.

It is important that the modelled engine speed match the frequency of the measured signal and that it can capture the behaviour of a relevant disturbance, which it manages to do.

When evaluating the engine speed performance at high and low speeds, as in Figure 4.7 and Figure 4.8, one can that the speed signal shows good behaviour

for lower engine speeds. At higher engine speeds though the signal is not as accurate, especially when including torsion. This sets certain limitations of the operating ranges that the simulation framework can be used within at the current state. The inaccuracies at higher engine speeds are most likely caused by the torsion model itself. The torsion model is based on a previous work by [16] and in the report the author has concluded that the torsion model have areas of improvement. At higher engine speeds, the effects of torsion are more prominent and errors in the model will have a larger affect on the engine speed signal. There is a also a possibility that the torque models, which are also based on the work by [16] have inaccuracies that can have an impact at higher engine speeds. This could explain the flattening affect in Figure 4.8 when using the rigid crankshaft. The model that is used to calculate the mass torque is quite sensitive and can grow to large values when engine speed increases. The decision was therefore made to limit the maximum amplitude from the mass torque model to ± 800 Nm. This restricted the flattening affects of becoming any worse than what is seen in Figure 4.8. This restriction enables that if using the rigid crankshaft the simulation framework can to be used at higher engine speeds as well, up to 2000 rpm if one wishes. When using the non-rigid crankshaft the engine speed signal starts to get distorted when the speed exceeds 1000 rpm. In order to increase the operating range for the non-rigid crankshaft, the torsion model itself has to be improved, which is something that has not been within the area of this thesis work.

Parameter Estimation The parameters that were estimated as three dimensional interpolation maps all seem to represent some type of surface. This indicates that the parameters can be expressed as a function of load and engine speed, at least to some extent. In all plots the measured or estimated values that the map uses as interpolation values can be located in the graph as edges or peaks. The smoother the transition between the different peaks are, the more likely it is that the interpolated values are accurate. Of all the interpolation maps shown in Section (4.1.4), the intake pressure and SOI are the maps that are based on measured values. The maps for the combustion parameters are all based on estimated values. This makes it clear that the parameter estimation for the intake pressure and SOI are more likely to be accurate than the combustion parameters. In order to improve the accuracy of the parameters with this type of modelling approach the maps need to be based on more measurement data. Alternatively, the parameters could be decided using mathematical expressions, which could be more favourable than the extensive use of interpolation maps that are currently used. SOI for example, should ideally be calculated in the block that contains the main fuel controller, preferably as a function expression of fuel mass and engine speed.

Controller The four different tests for the controller were meant to test the controller's performance to handle different disturbances in some different operating points. The results show that the controller can minimize imbalances that are caused by error in injected fuel mass and load disturbances with a frequency that match the engine speed frequency. However it can not handle disturbances where the load disturbance has a frequency that lies outside of the engine speed's

or disturbances caused by faulty sensor readings. This is a limitation that the controller has and is not caused by any modelling errors. Since the controller only looks at the previous cycle when calculating its control signal, it is limited to handling disturbances that have a behaviour that is repeated in every cycle. In order to be able to handle disturbances of other frequencies, one would need to detect this frequency and its effect on the engine speed signal. The controller would then have to make different adjustments in different engine cycles to an individual cylinder. Detecting this behaviour and calculate such a control signal requires a far more sophisticated controller. It might be that the even if such a behaviour could be detected, the behaviour of the disturbance could change if the controller attempts to make corrections. Regarding faulty sensor readings, the effects of noise could possibly be minimized using an engine speed signal with higher resolution, i.e. using a flywheel with 120 teeth. The hardest part in real situation is to know what type of disturbance that is causing the imbalance. In a simulation environment the characteristics of these disturbances are known completely but in real life it is hard to know whether insufficiency depends on the controller or is caused by a disturbance which can not be handled.

5

Conclusions

Overall the simulation framework shows promising results. The pressure model is able to pretty accurately recreate the cylinder pressure trace when compared to the measurement data. Parameters that were interpolated through look-up tables represents a surface when plotted, which supports that idea that they have a dependence on fuel mass and engine speed.

The framework is able to generate an engine speed signal that captures the behaviour of the measured engine speed, as far as pulse size and frequency. Some deviations exist when it comes to the width of the pulses that may come from some inaccuracies in the torque models.

When using a rigid crankshaft the framework is able to generate an engine speed signal up to 2000 rpm if one wishes. However the non-rigid crankshaft can only be used up to around 1000 rpm. At engine speeds above this, the inaccuracies in the lumped mass model become so significant that they distort the rpm-signal. Since cylinder imbalance in real life is an issue most prominent at lower engine speeds, the need to simulate at higher engine speeds than around 1200 rpm might not be a necessity.

The framework also enables a convenient test setup and is computationally manageable. A simple PI-controller was implemented to test the framework. The results showed that the controller is able to eliminate imbalances caused by disturbances that are within its control-ability limits, i.e. imbalances that are caused by disturbances whose frequency lie within one engine cycle.

The results from this work supports the idea of continuing the development of the simulation environment as well as come up with new control strategies for testing and implementation.

5.1 Future Work

In order to improve the simulation framework there are some particular areas that could be addressed.

1. More data would be necessary to make better interpolation for parameters that are based on measurements. This includes for example intake pressure, SOI, λ -value, exhaust pressure, intake temperature etc.
2. Improving the estimation of combustion parameters in the Wiebe function. The proposed solution of estimating the burn angles as well as the tuning parameter m is just one of many options that exist. One could attempt to make better estimations if having access to more pressure data. What is worth pointing out however is that putting excessive work into details in the combustion part is probably not necessary. The engine speed signal will most likely not be affected much anyway. It can however be worth looking in to alternative solutions since the least squares fit were only able to give an acceptable solution, and for many operating points the value of m had to be manually tuned.
3. Improving the torsion model would enable the framework to be used at higher engine speeds for the non-rigid crankshaft. One could possibly look into improvements for the mass torque model as well so that the current saturation at 800 Nm will not be needed.
4. The engine speed controller that is currently implemented is a simple PI-controller used to find the main fuel delivery for running steady at the desired speed. It is able to find the suitable fuel value to keep the desired engine speed within a few seconds and its performance is mainly depending its initial fuel value. The initial fuel value has been found through simulating various operating points and allow the rpm controller to find the steady state fuel amount. This initial value is fairly close to the final fuel value for any desired operating point and seems to work well enough. It is worth mentioning though that an alternative rpm controller that is able to more quickly lead the engine speed signal to steady state could be favourable.
5. Making the simulation framework compatible with other engine types. This thesis has been limited to investigating the DC13155 engine. In order to make the framework compatible with other engines there are a few things that need to be clarified. All parameters that are based on measurements or estimated from data are unique for a particular engine. Switching to a new engine means that these new parameters would need to be calibrated for the new engine. If the engine uses the same main components as the DC13155 engine and have an equal displacement, some minor changes could be done in the engine parameters and the engine would be compatible with all the torque models. If changing to a larger or smaller engine that uses different components, the torsion model would not resemble that engine since it is

strictly based on geometrical and mechanical features in the crankshaft. In such case the torsion model would have to be calibrated for the new engine.

It is however of importance to note that in the current state the torsion model is not sufficiently accurate to be used for simulation purposes and the need to include a non-rigid crankshaft might not be of highest importance for controller evaluation. If the framework could use the rigid crankshaft for evaluation purposes, the complications of changing to a new engine type become far smaller.

Bibliography

- [1] Ahmed Al-Durra, Lisa Fiorentini, Marcello Canova, and Stephen Yurkovich. A model-based estimator of engine cylinder pressure imbalance for combustion feedback control applications. In *American Control Conference (ACC)*, 2011, pages 991–996. IEEE, 2011. Cited on page 7.
- [2] K Alkhulaifi and M Hamdalla. Ignition delay correlation for a direct injection diesel engine fuelled with automotive diesel and water diesel emulsion. *Journal World Academy of Science, Engineering and Technology*, 6(58):905–917, 2011. Cited on pages 5 and 13.
- [3] Usman Asad and Ming Zheng. Real-time heat release analysis for model-based control of diesel combustion. Technical report, SAE Technical Paper, 2008. Cited on page 5.
- [4] Michael FJ Brunt and Kieron C Platts. Calculation of heat release in direct injection diesel engines. Technical report, SAE Technical Paper, 1999. Cited on page 5.
- [5] Franz G Chmela and Gerhard C Orthaber. Rate of heat release prediction for direct injection diesel engines based on purely mixing controlled combustion. Technical report, SAE technical paper, 1999. Cited on page 6.
- [6] C Cooney, J Worm, D Michalek, J Naber, et al. Wiebe function parameter determination for mass fraction burn calculation in an ethanol-gasoline fuelled si engine. *Journal of KONES*, 15:567–574, 2008. Cited on pages 6 and 16.
- [7] Lars Eriksson. Requirements for and a systematic method for identifying heat-release model parameters. Technical report, SAE Technical Paper, 1998. Cited on page 6.
- [8] Lars Eriksson and Ingemar Andersson. An analytic model for cylinder pressure in a four stroke si engine. Technical report, SAE Technical Paper, 2002. Cited on pages 6, 11, and 13.
- [9] Lars Eriksson and Lars Nielsen. *Modeling and control of engines and drivelines*. John Wiley & Sons, 2014. Cited on pages 4, 12, and 35.

- [10] John Heywood. *Internal combustion engine fundamentals*. McGraw-Hill Education, 1988. Cited on pages 3, 4, 6, and 17.
- [11] Tobias Johansson. Virtual sensors for combustion parameters based on in-cylinder pressure. Master's thesis, Linköping University, 2015. Cited on page 6.
- [12] I Kolmanovsky. Detecting and correcting cylinder imbalance in direct injection engines. *Journal of Dynamic Systems, Measurement and Control*, 2001. Cited on pages 6, 7, and 18.
- [13] Krzysztof Z Mendera, Andrzej Spyra, and Michał Smereka. Mass fraction burned analysis. *Journal of KONES Internal Combustion Engines*, (3-4): p193–201, 2002. Cited on pages 6 and 16.
- [14] Noboru Miyamoto, Takemi Chikahisa, Tadashi Murayama, and Robert Sawyer. Description and analysis of diesel engine rate of combustion and performance using wiebe's functions. Technical report, SAE Technical Paper, 1985. Cited on page 6.
- [15] Fredrik Östman and Hannu T Toivonen. Active torsional vibration control of reciprocating engines. *Control Engineering Practice*, 16(1):78–88, 2008. Cited on page 8.
- [16] Tobias Roswall. Design of virtual crank angle sensor based on torque estimation. Master's thesis, KTH, 2016. Cited on pages 7, 10, 20, and 41.
- [17] Stefan Schagerberg and Tomas McKelvey. Instantaneous crankshaft torque measurements-modeling and validation. Technical report, SAE Technical Paper, 2003. Cited on page 7.
- [18] H Yasar, HS Soyhan, H Walmsley, B Head, and C Sorousbay. Double-wiebe function: An approach for single-zone hcci engine modeling. *Applied Thermal Engineering*, 28(11):1284–1290, 2008. Cited on page 5.