Combustion Engine Identification and Control

Daniel Blasco Serrano

LUND UNIVERSITY

Department of Automatic Control

MSc Thesis
ISRN LUTFD2/TFRT--5943--SE
ISSN 0280-5316

Department of Automatic Control
Lund University
Box 118
SE-221 00 LUND
Sweden

Printed in Sweden by Media-Tryck
Lund 2014

Abstract

The topic of this thesis is system identification and control of two different internal combustion engines, Partially Premixed Combustion (PPC) engine and a more conventional Combustion Ignited (CI) diesel engine. The control of both engines is aimed to emission reduction and to increase the efficiency.

There is an introduction to the internal combustion engine, as well as theory used about system identification and Model Predictive Control (MPC).

A physical model of a PPC engine was designed. With this model, it was possible to perform simulations and to obtain data to design a cycle-to-cycle state space model used subsequently by the MPC controller.

A black box model of the CI engine was designed using data of a real CI engine from another project. After designing the model, a MPC controller was created. The aim of this controller was to reduce emissions and to increase the efficiency of the engine.

Acknowledgements

This master thesis would not have been possible without the guidance of my examiner, Rolf Johansson, who helped me to define the objectives and to define this thesis. Rolf has also taught me uncountable automatic control concepts in two different courses that have been essential for this thesis.

I would like to thank my supervisor, Gabriel Ingesson, who has spent a huge amount of hours with me and this master thesis. Gabriel has given me thousands of ideas for this thesis, has always shown me the correct way in the most difficult moments, and has reviewed and corrected the report of the thesis until it was perfect.

I would like also to thank Per Tunestål, who has shared his knowledge about combustion engines and control with me, and who has provided useful suggestions for this master thesis.

I am very grateful to the people of the master thesis room and all the friends I have met in Lund that have made my stay in Lund unforgettable. I also thank my friends from Barcelona, who have been with me during the best and the hardest moments in the degree.

Last, but not least, I would like to thank Adriana for her unconditional support from Barcelona, my parents, José and Elena, my brother and sister, Ricardo and Patricia, and the rest of my family. I truly appreciate your support and encouragement.

Contents

Part I

Introduction

1

Introduction to Internal Combustion Engines

The internal combustion engine is today the most used technology for transportation of goods and people worldwide [**7**]. The need for transportation and the contamination produced by this type of engines force the society to improve the technology of the internal combustion engines. For this reason, more efficient internal combustion engines that can comply with the increasingly restrictive emission level standards are under research. A way of optimizing engine efficiency and lower engine emissions is of course to improve the design of the engine control system. However, controlling these new types of engines can be challenging [**10**]. With the help of the increasingly available computer power and the amount of engine sensor information available today more advanced control methods such as MPC could be used to control the engines of today and tomorrow.

1.1 Internal Combustion Engine

Figure 1.1 shows basic geometry and the operation principle of an internal combustion engine. The internal combustion engine produces mechanical power from the chemical energy stored in the fuel. The energy is released by burning the fuel in a combustion chamber resulting in high pressures inside the cylinder, allowing the gas mixture to produce work by pushing a piston inside the combustion chamber. This linear movement is transformed into rotatory movement by a crankshaft-rod device, allowing to put the wheels of the vehicle in motion. In a four-stroke engine the thermodynamic cycle which produces work is completed during two revolutions of the crankshaft, or four piston strokes:

- **Intake stroke:** during the first stroke of the cycle the piston goes from TDC (Top Dead Center) to BDC (Bottom Dead Center) while the

intake valve is opened to draw fresh air into the combustion chamber from the intake manifold of the combustion chamber (see Figure 1.1).

- **Compression stroke:** after BDC the piston compresses the unburned mixture while going to TDC with the valves closed. During this stroke the work is done on the gas by the piston (negative work). At the end of this stroke, when the piston is close to TDC, the combustion process is initiated.

- **Power stroke:** most of the combustion takes place after TDC which increases the in cylinder pressure and temperature. The high pressure forces the piston down towards BDC. During this stroke work is done by the gas mixture on the piston (positive work).

- **Exhaust stroke:** in order to replace the burnt gases with a new fresh air/fuel mixture the exhaust valve opens and lets the burnt gas mixture exit the cylinder to the exhaust manifold, while the piston goes from BDC to TDC. This completes two revolutions of the crankshaft and the four strokes of the engine. After the exhaust stroke has ended the four-stroke cycle is restarted with an intake stroke.

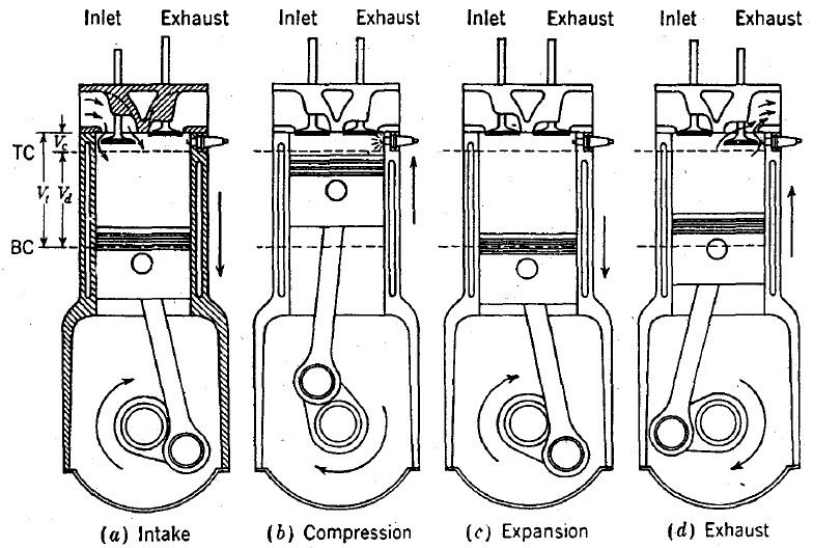

Figure 1.1 The four-stroke operating cycle. Source: [8]

There are many ways of classifying an internal combustion engine, however one of the most common is by the ignition method. The two most conventional types are the gasoline (Otto) engine and the Diesel engine.

1.2 Gasoline engine

These engines run on gasoline fuels. The fuel and the air are premixed before the piston reaches TDC. In addition, gasoline engines use an external source of energy to ignite the air-fuel mixture, done by spark. After the ignition the combustion proceeds through the combustion chamber as an exothermic reaction. In the Spark Ignited (SI) engines, the timing of the start of combustion (SOC) is chosen by controlling the timing of the spark, which is usually set just before TDC during the compression stroke. In a controller design perspective the spark timing is a trade-off between thermodynamic efficiency, the risk of knock and NO_x formation [**8**]. The efficiency of gasoline engines is lower than Compression Ignited (CI) engines, which results in a higher fuel consumption and higher emissions of CO_2. However, NO_x and particulate emission levels are lower than those from diesel engines when operated together with a catalyst [**8**].

1.3 Diesel engine

These engines run on diesel fuels which are injected into the combustion chamber close to TDC. After the fuel has vaporized and mixed sufficiently with the air, the combustion the combustion takes place with a heterogeneous fuel and air mixture [**8**]. These engines are Compression Ignited (CI) and this thesis will only cover this type of engine operation. In comparison to the Otto engine, the efficiency of the Diesel engine is higher and the emissions of CO_2 are lower. Nevertheless, diesel emissions of NO_x are typically higher [**14**].

1.4 HCCI and PPC

There exist more engine concepts under research. Examples of newer engine concepts are: Homogeneous Charge Compression Ignition and Partially Premixed Combustion engines. These try to improve the efficiency and to lower the emissions of the conventional engines. Figure 1.2 shows how these type of engines are related to the conventional engines explained above.

Homogeneous Charge Compression Ignition (HCCI) engines are a combination of SI and CI concepts. Here, the fuel is injected earlier in the cycle so that the fuel-air mixture is homogeneous prior to the start of combustion in

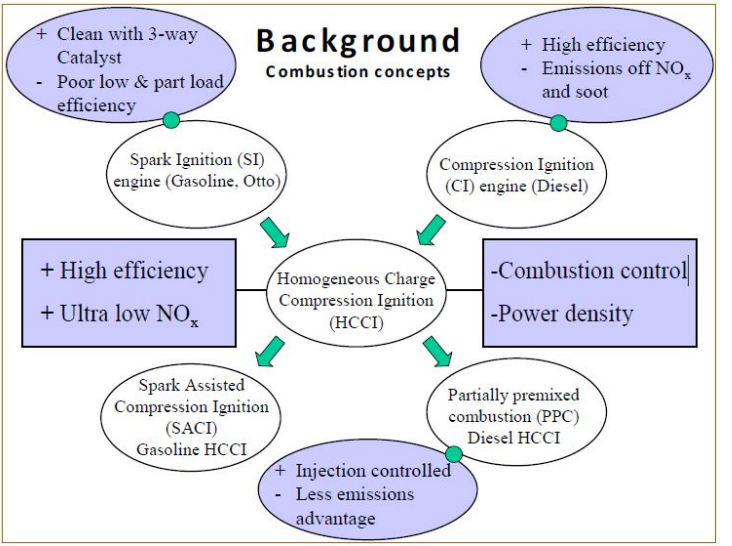

Figure 1.2 Different combustion processes in internal combustion engines Source: [**9**]

the same way as SI. However, in HCCI the mixture will be compression ignited and the timing of ignition will depend on the pressure, the temperature and the properties of the charge. HCCI has the advantage of a homogeneous combustion without hot zones which reduces NO_x emissions, and, since the charge is homogeneous, no locally rich zones occur, reducing soot formation [**8**]. However, it is not trivial to control the point of auto-ignition. This makes HCCI more difficult to control compared as to SI and CI engines.

Partially Premixed Combustion (PPC) is achieved by having a higher resistance to auto-ignition than in conventional CI and injecting the fuel early enough for substantial mixing to occur before combustion starts, but not so early that the mixture is completely homogeneous. In addition, if the injection is sufficiently close to the autoignition point it is possible to find a connection between injection timing and ignition [**13**]. This results in a combustion mode with better controllability than HCCI without increasing the emissions of NO_x and soot to the levels of traditional Diesel engines [**13**].

2

Introduction to Physical and Black Box Modelling and MPC

As mentioned in Section 1.4, in order to increase efficiency and reduce emissions of the engines it is crucial to have a good automatic control system. The engine is a complex system with multiple inputs and outputs and the goal of the controller is to achieve a high efficiency with low emissions. This could be formulated as an optimization problem. This suggests that Model Predictive Control could be a suitable controller choice (Section 2.3). This controller is model-based, which means that in order to design the parameters for the controller, a model of the engine is needed. This chapter introduces physical and black box modelling and Model Predictive Control theory.

2.1 Physical Modelling

The knowledge about physical laws can help to describe the behaviour of a system. One can describe a system using physics applied to mathematical equations. For instance, the ideal gas law is widely used to model the state of a thermodynamic system:

$$pV = nRT \tag{2.1}$$

There also exist general principles that can be used as guidelines for physical modelling. For example, the conservation of mass, energy (used in Section 3.1) and momentum lead to physical balance equations on the form

$$\text{accumulation flow} = \text{inflow} - \text{outflow} \tag{2.2}$$

which are useful for mechanical, electrical, chemical and thermodynamical systems [**12**].

Complex systems like combustion engines may be described with more than one equation. Given that to put all the equations together to design a single model is not easy, it is possible to design a simulation program. Then, one can simulate the model to obtain data which can be used to identify simpler black box models (explained in the next section) with lower complexity and more suitable for controller design purposes. In this thesis physical laws have been used to design a simulation program (see Section 3.1). Afterwards, it has been possible to obtain a state-space model for the controller that later was used for MPC design.

2.2 Black Box Modelling

Black box modelling is used when there is not enough knowledge of the systems internal workings. Therefore the system is treated as a black box, that transfers an input to an output (Figure 2.1).

Figure 2.1 Black box schematics

In order to model the system transfer characteristics, input output data is obtained from the real process after making different experiments, although can be obtained from a simulation program, as explained in the last section.

The input used in the identification must have enough excitation in order to provide all the information of the dynamics of the system. Usually a good choice is to use Pseudorandom binary sequences [**12**].

Once one has sufficient input-output data, it is possible to fit a priori chosen models to this data, for example state-space or ARMAX models of different orders. A higher order model (with more parameters) usually fits the input / output data more accurately, but it does not mean that a higher order model is better than a lower one. If the number of model parameters is chosen too high there is a risk that the the fitted model could get too adapted to the specific input / output data set and would not be successful in predicting outputs of new input/output experiments. This is called over-fitting and could be avoided by doing cross-validation experiments which also were done in this thesis. In order to choose the model order, there are criteria

that can be followed. In this thesis, the Akaike information criterion (AIC) was used, where the following formula should be minimized:

$$AIC(p) = \log \widehat{\sigma}^2(\widehat{\theta}_N) + \frac{2p}{N}, \quad \widehat{\sigma}_N \in \mathbb{R}^p \tag{2.3}$$

where $\sigma^2(\widehat{\theta}_N)$ is the variance of the estimated parameters, p the number of model parameters and N the number of data samples. Although this criteria may lead in favour of higher order models, it helps to choose the optimal number of parameters needed to describe the desired model [**12**].

2.3 Model Predictive Control

"In MPC, the current control action (at each sampling instant) is obtained by solving on-line open-loop finite-horizon optimal control problem with current measurements as the initial state, the solution providing an optimal control sequence where only the first control step is applied to the process. At each new measurement, the optimal control problem is recalculated over the prediction horizon, which provides an open-loop finite-horizon control plan. Again, the controller actuates the first move of the control plan, thus providing an iterative nature to MPC. As the iteration proceeds, the prediction horizon moves ahead, a feature denominated receding horizon control [**11**].

In other words, MPC predicts the outputs during a determined prediction horizon (p) and computes also the control signals from the actual sample during the control horizon (m), although only the first move is applied. MPC is model-based, meaning that it is necessary to have a good model describing the process in order to design a controller.

The MPC algorithm consists in solving an optimization problem where a cost function (2.4) is minimized with respect to the system input. This cost function contains the weights of the inputs and outputs and the penalization for the soft constraints for each sample in the prediction horizon. This cost function must be solved subject to the constraints mentioned previously. The cost function is minimized at each sample instant and the first control signal is then applied to the process [**11**].

$$\begin{aligned} J(u) = \sum_{i=0}^{p-1}(\sum_{j=1}^{n_y}|w_{i+1,j}^{y}\left(y_{k+i+1|k}^{j} - r_{k+i+1}^{j}\right)|^2 + \sum_{j=1}^{n_u}|\left(w_i^{\Delta u}\right)^j \Delta u_{k+i|k}^{j}|^2 \\ + \sum_{j=1}^{n_u}|(w_i^u)^j\left(u_{k+i|k}^{j} - \left(u_{k+1}^{r}\right)^j\right)|^2 + \rho_\varepsilon \varepsilon^2) \end{aligned} \tag{2.4}$$

Here, y are the outputs, r the references, u the inputs, Δu the inputs variations and w the weights assigned to each parameter. The $\rho_\varepsilon \varepsilon^2$ expression is the penalization when any soft constraint is violated.

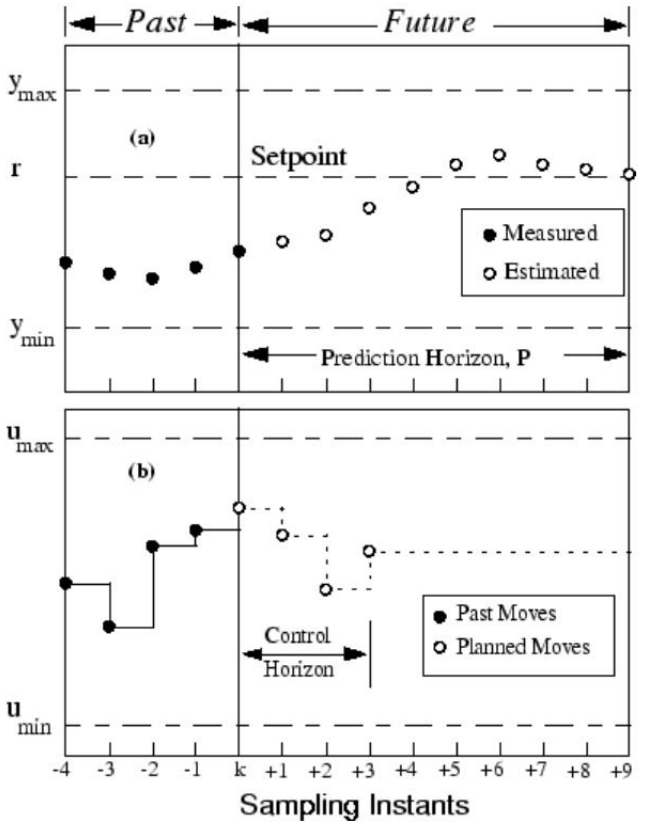

Figure 2.2 Model Predictive Control. Source: [1]

MPC has some advantages with respect to other control methods:

- It can be used with MIMO systems, having multiple inputs and multiple outputs.
- MPC works with a function to minimize, called cost function. With this function it is possible to specify controller priorities in order to obtain a desired behaviour of the process.

- With the cost function it is possible, for example, to have different weights for each output. This way some outputs are more penalized than others if they don't reach its reference.
- One can also have weights in the inputs in order to prioritize some more than others. It is also possible to set a weight in the input variation.
- One can add constraints to inputs or outputs, in order not to exceed a determined value for an input or output. These constraints can be softened, letting the controller violate the constraints but increasing the value of the cost function.

Part II

PPC Engine Identification and Control

3

Physical Cycle-to-Cycle Modelling of PPC

This chapter presents a mathematical model of a PPC cylinder based on physical laws (e.g. thermodynamic laws). All the relevant variables of the system are presented and explained. Finally, the physical model obtained is used in different simulations to obtain a state-space model which will be useful to design an engine controller.

3.1 Physical Model

The model used in this chapter describes the behaviour of the engine during the closed part of the cycle. This model aims to predict the cylinder pressure during the closed part of the cycle given the injection timing and the injected fuel mass. Nevertheless, to reduce the amount of time and data needed by the controller to compute the control signal, the inputs given to the controller will act as a summary of the whole cycle. For example, the pressure trace obtained with the half-cycle model will be used to calculate the Indicated Mean Effective Pressure (IMEP) and the maximum pressure of the whole cycle. These two parameters act as a summary of the whole pressure signal. These variables will be used by the controller to compute the control signal of the next cycle.

To build a model of an engine it is necessary to define the variables that are going to be relevant for the control of the engine. Those are the inputs to the engine, outputs, and other variables whose value should be taken into account. A schematics diagram of the controller and the system is presented in Figure 3.1.

Inputs

The parameters which one can choose freely to modify the behaviour of the engine are three, all connected to the fuel injection system. These are

the duration of the injection, the injection pressure and the crank angle of SOI (start of injection), θ_{SOI}. To choose the mass fuel injected, the injection pressure and injection duration must be determined. However, the relation between duration of injection and injection pressure to the injected fuel mass has not been modelled here. Instead, the fuel mass (m_f) has been regarded as a system input.

Outputs

There are mainly two sensors in the engine that are going to be considered. The first one is a crank angle encoder which will measure the angular position of the crank. Given that the chosen time scale for this system is not the absolute value of time, but the crank angle, one must take into account that if the engine is running at a higher speed, the computer will have to calculate the control signal faster because the cycles duration will be shorter.

The other sensor is an in-cylinder pressure sensor. This will give a relative pressure inside the cylinder. Considering the pressure in the inlet manifold equal to the pressure in the cylinder during the intake process, it is possible to compare relative pressure in the cylinder with the absolute pressure in the inlet manifold. This will give the absolute pressure (p) during the cycle. With those values it will be possible to calculate more relevant parameters for the controller. Directly one can find the maximum pressure (p_{max}) which will be used by the controller in order not to exceed a too high pressure in the cylinder. The Gross Indicated Mean Effective Pressure (IMEP_g) will be also calculated. This is the indicated work on the piston during the closed part of the cycle (compression/expansion stokes) divided by the displacement volume (V_d) [**14**]:

$$\text{IMEP}_\text{g} = \frac{1}{V_d} \oint p \text{d}V \tag{3.1}$$

Another relevant variable that will be treated as an output will be the crank angle in which half of the heat (Q) has been released from the combustion process (θ_{CA50}). This parameter is used to control the combustion timing. Given that there is not a sensor to measure the heat release, an observer that uses cylinder pressure to estimate the heat will be designed.

Cylinder Movement

As it has been said, the absolute value of time is not taken into account for the description of the system in this thesis. Instead, the crank angle of the engine is used. It is trivial to see that each crank angle will correspond to an unequivocal position of the piston, and this will lead to the volume of the combustion chamber. By taking the cylinder geometry into consideration, it is possible to obtain the following formula to find the cylinder volume (V) depending on the crank angle (θ) [**8**]

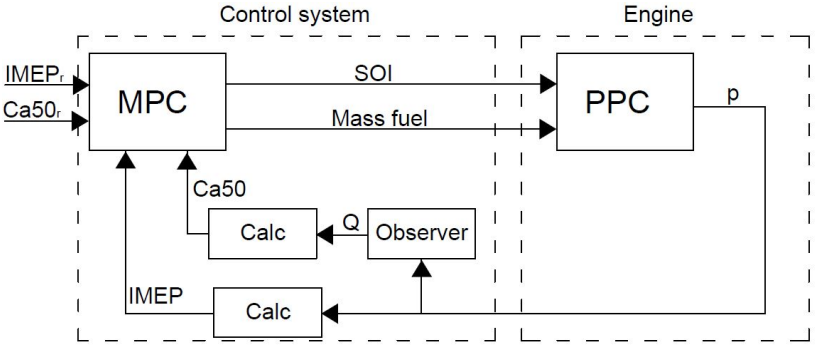

Figure 3.1 Schematics of the system of the engine and the controller

$$V = V_c + \frac{V_d}{2}\left(R_v + 1 - \cos(\theta) - \sqrt{(R_v^2 - \sin^2(\theta)}\right) \tag{3.2}$$

where V_c is the minimum volume in the cylinder obtained at TDC (clearance volume) and R_v the ratio between the connection rod length and the crank radius (see Figure 1.1).

It is also possible to define the derivative of the volume with respect to crank angle

$$\frac{dV}{d\theta} = \frac{V_d}{2}\sin(\theta)\left(1 + \frac{\cos(\theta)}{\sqrt{R_v^2 - \sin^2(\theta)}}\right) \tag{3.3}$$

which may be integrated during the simulation in order to obtain (3.2).

Heat Release Model

The heat released from the combustion process has been modelled as a Gaussian function centred in the half of the heat released (θ_{CA50})

$$\frac{\mathrm{d}Q}{\mathrm{d}\theta} = \frac{e^{\frac{-(\theta-(\theta_{SOI}+\tau))^2}{2\sigma^2}}}{\sqrt{2\pi\sigma^2}} Q_{LHV} \cdot m_f \tag{3.4}$$

where τ is the ignition delay between the injection and the θ_{CA50}, σ is the width of the Gaussian function that represents the duration of the combustion and Q_{LHV} the lower heating value of the fuel. Note that this function only describes the heat released well in cases when it is fairly symmetric. An example of the heat release together with the pressure during a cycle is shown in Figure 3.2.

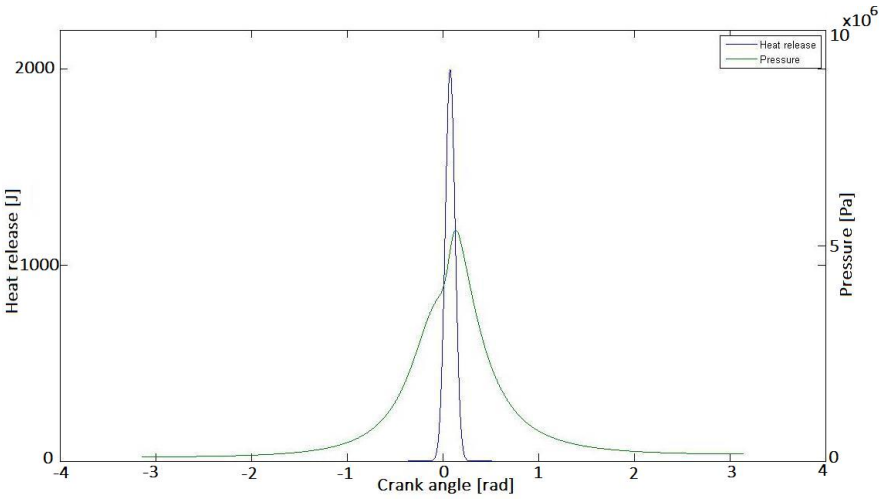

Figure 3.2 Example of heat release during simulations

The relationship between the heat released and the pressure is given by [8]

$$\frac{\mathrm{d}Q}{\mathrm{d}\theta} = \frac{\gamma}{\gamma - 1} p \frac{\mathrm{d}V}{\mathrm{d}\theta} + \frac{1}{\gamma - 1} V \frac{\mathrm{d}p}{\mathrm{d}\theta} \tag{3.5}$$

where γ is the heat capacity ratio of the mixture.

Observer

There are no sensors measuring the heat released in the engine. Therefore, an observer for the model has to be designed. In order to find the heat release from the measured pressure, Eq. (3.5) is used.

Having this in mind, it is intuitive to find θ_{CA50} by integrating (3.5).

Heat Losses to the Cylinder Walls

An example of cycle-to-cycle dynamics is caused by the temperature of the cylinder walls. This temperature is affected by the last cycle and will influence the pressure inside the cylinder in the next cycle. Therefore, it is necessary to model the heat losses of the gas in the combustion chamber to the cylinder walls and the coolant (Figure 3.3) [**14**].

The heat losses are modelled as convection from the gas to the wall

$$\dot{Q}_{ht} = h_c A_c \left(T - T_w\right) \tag{3.6}$$

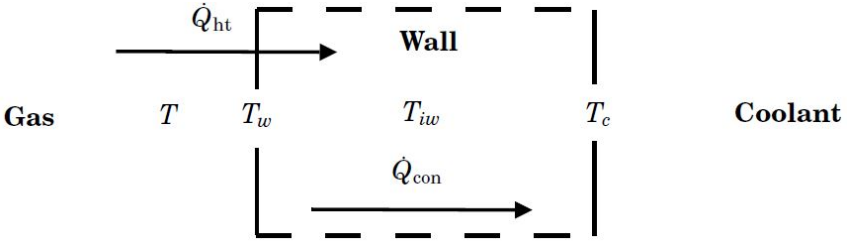

Figure 3.3 Heat losses through walls. Source: [14]

where h_c is the convection coefficient that could be obtained experimentally (although it depends on T, p and the in cylinder flow [8]), A_c the cylinder surface area, T the gas temperature and T_w the wall temperature.

The conduction through the cylinder wall was modelled as

$$\dot{Q}_{con} = \frac{(T_w - T_c)\, k_c A_c}{L_c} \tag{3.7}$$

where T_c is the coolant temperature, k_c the conduction coefficient and L_c the wall thickness. After obtaining these two heats, it is possible to find the variation of the wall temperature with

$$\dot{T}_w = 2\frac{\dot{Q}_{ht} - \dot{Q}_{con}}{m_c C_p}, \quad T_w(0) = T_{w_0} \tag{3.8}$$

where m_c is the wall mass and C_p the specific heat. However, given that only half of the cycle was modelled, the wall temperature that is obtained corresponds to the temperature at half of the cycle. In order to get the final temperature of the cycle (and the initial temperature of the following cycle), it is necessary to add some first order dynamics. It can be done using the equation

$$T_w = T_{w_{half}}\alpha + T_c\,(1-\alpha) \tag{3.9}$$

where α should be tuned accordingly to the real engine.

Ignition Delay

The last dynamics that was implemented to the model was the variation of the ignition delay, τ. This delay depends on the fuel used, but also depends on the temperature and the pressure during injection. It is possible to correlate this ignition delay to equations of the form

$$\tau = Ap^{-n}e^{\frac{E_A}{RT}} \tag{3.10}$$

where E_A is the activation energy of the fuel for auto ignition, R the universal gas constant and A and n two tuning parameters that could be found empirically [8].

3.2 State-Space Model

The MPC controller will use a model in state-space represention (3.11), where $x(k)$ is the state vector, $u(k)$ the input vector and $y(k)$ the output vector at sample k.

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) \end{aligned} \tag{3.11}$$

Input-Output maps

In this part of the thesis, the cylinder wall temperature and the variable ignition delay have not been considered because of time constraints in the master thesis. This means that there is no cycle-to-cycle dynamics. Therefore, each input to the system will correspond to one output independently of the previous inputs and the state of the system. For this reason, an input-output map was found by doing simulations (Figure 3.4).

The first output, IMEP is clearly non-linear with an absolute maximum value for a determined θ_{SOI}, $\arg\max_{\theta_{SOI}} \text{IMEP}_\text{g}$. This graph shows that there is an optimal θ_{SOI} point in order to obtain the maximum work done to the piston. In order to fit a linear model for this output, one could easily fit two planes (one for $\theta_{SOI} < \arg\max_{\theta_{SOI}} \text{IMEP}_\text{g}$ and another one for $\theta_{SOI} > \arg\max_{\theta_{SOI}} \text{IMEP}_\text{g}$).

The second output corresponds to the maximum pressure achieved in the cylinder. The graph shows that in order to reduce the maximum pressure, the mass fuel should be reduced, or the θ_{SOI} increased. It is also non-linear, although it is possible to find a good approximation by fitting a linear regression.

The last output corresponds to the crank angle in which half of the heat has been already released. This output has been measured with an observer, as explained in Section 3.1. This graph shows a linear relationship between θ_{CA50} and θ_{SOI}, and shows no connection between the output and the mass fuel injected. It is trivial then to fit a linear regression to find a good model.

Regressions

Finding a linear model for each output makes possible to find a state-space model when there is not dynamic cycle-to-cycle. Each output has been found as an equation of the form

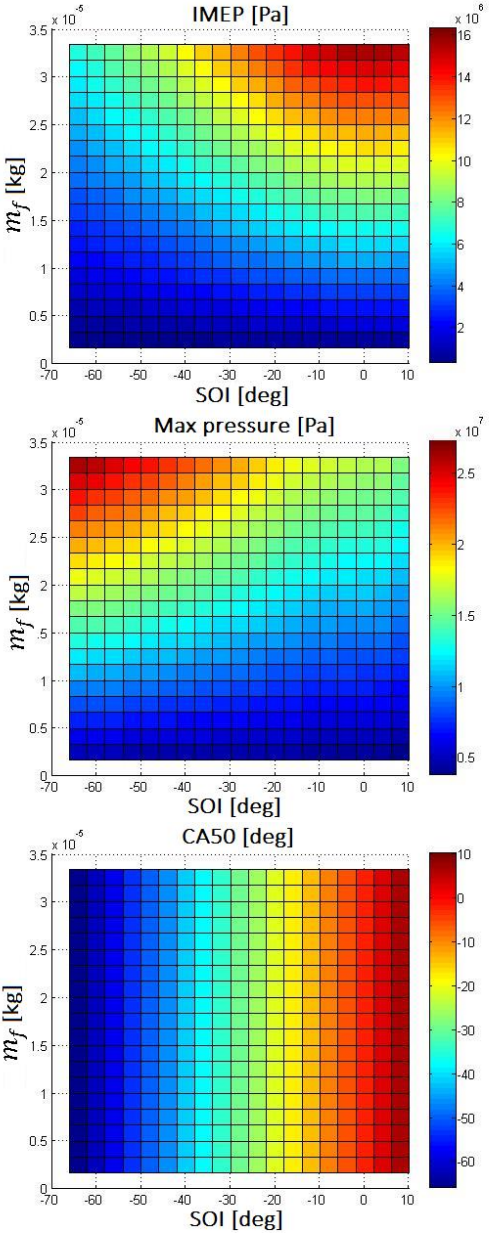

Figure 3.4 Input-Output maps

$$y_i = a_i \cdot m_f + b_i \cdot \theta_{SOI} \tag{3.12}$$

where a_i and b_i are the regression coefficients using least-squares estimation. Putting all the outputs together leads to a matrix equation of the form

$$Y = BU \tag{3.13}$$

where

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} \quad B = \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_3 \end{bmatrix} \quad U = \begin{bmatrix} m_f \\ \theta_{SOI} \end{bmatrix} \tag{3.14}$$

Finally, to transform (3.13) into a valid state-space model is trivial.

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) \end{aligned} \tag{3.15}$$

Here $A = 0$, B is the same matrix as in (3.14) and C is an identity matrix. To avoid control design problems caused by having $A = 0$ it is also possible to treat the inputs incrementally:

$$\begin{aligned} x(k+1) &= Ax(k) + B\Delta u(k) \\ y(k) &= Cx(k) \end{aligned} \tag{3.16}$$

Finally, A and C are identity matrices and

$$B_1 = \begin{bmatrix} 3.99 & -8.03 \\ 4.88 & 7.77 \\ 0.00 & 1.00 \end{bmatrix} \quad B_2 = \begin{bmatrix} 3.99 & -8.03 \\ 4.88 & -2.33 \\ 0.00 & 1.00 \end{bmatrix} \quad \Delta u(k) = \begin{bmatrix} \Delta m_f \\ \Delta \theta_{SOI} \end{bmatrix} \tag{3.17}$$

Here, $B = B_1$ when $\theta_{SOI} < \arg\max_{\theta_{SOI}} \mathrm{IMEP_g}$ and $B = B_2$ when $\theta_{SOI} > \arg\max_{\theta_{SOI}} \mathrm{IMEP_g}$.

Figure 3.5 and Figure 3.6 show the quality of the obtained model for each output. The colour in each graph represents the absolute value of the difference between the real output and the regression, being blue a good approximation and red a poorer approximation. As said before, θ_{CA50} fits the model perfectly. For this reason θ_{CA50} is not shown in the graphs. Meanwhile, IMEP and P_{max} contain some areas where the approximation is not so good. However, the general results are good enough, and this model could be used for controller design.

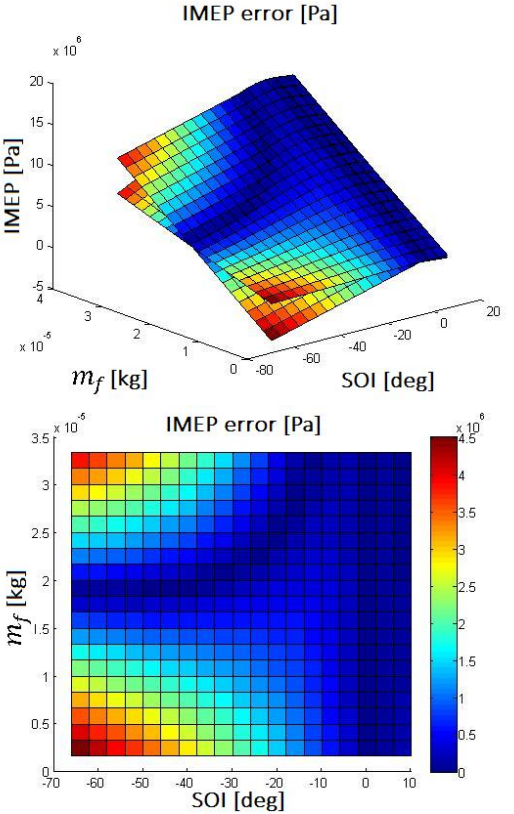

Figure 3.5 IMEP model regressions

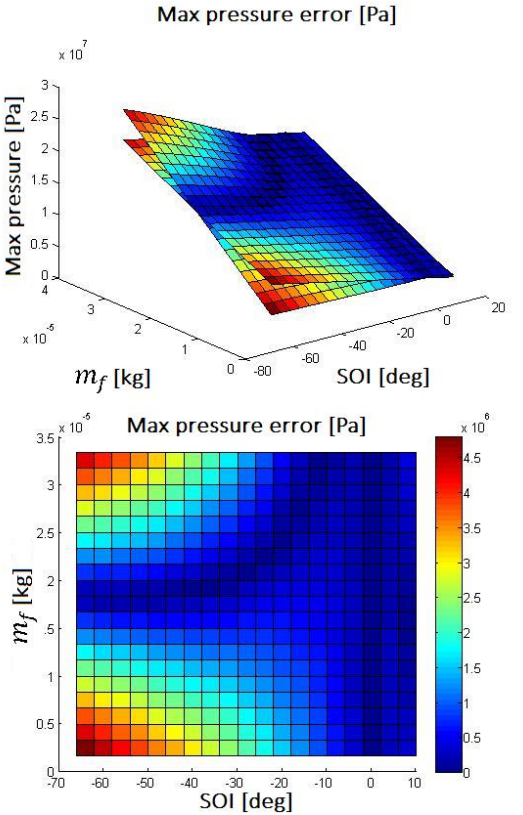

Figure 3.6 Maximum pressure model regressions

4

PPC Control Simulations

In the last chapter a model that approximates the behaviour of the real engine has been found. Now it is possible to design a controller based on MPC. In this chapter, the design of the MPC is explained, as well as some simulation results. These simulations can help to prepare for experimenting with the PPC engine.

4.1 MPC design

The model found in Section 3.2 consists of:

- 2 inputs: injected fuel mass variation (Δm_f) and start of injection variation ($\Delta\theta_{SOI}$)
- 3 outputs: Gross Indicated Mean Effective Pressure (IMEP_g), maximum pressure (p_{max}) and crank angle of half of the heat released (θ_{CA50})
- 3 states: given that the C matrix of the state-space model is an identity matrix, these states correspond to the same 3 outputs

It was possible to simplify the linear model found in the previous chapter with a minimal realization, obtaining 2 states instead of 3. However, this won't affect the results as both models are equivalent. However, it is necessary to design two different MPCs for controlling the engine. This is because there exist two different models for the engine, one for $\theta_{SOI} < \arg\max_{SOI} \text{IMEP}_\text{g}$ and the other for $\theta_{SOI} > \arg\max_{SOI} \text{IMEP}_\text{g}$. Each model describes the behaviour of IMEP in a different side of its maximum value (see Section 3.2). Each MPC will control one model and will swap every time the θ_{SOI} changes from one side to the other, as a smart switch.

As explained before, the sampling time for the MPC is not constant. The MPC will carry out its calculations once per each cycle of the engine. This means that if the engine is running at a higher speed, the controller will

have less time to compute the next control signal. This should be taken into account when working with the real engine.

To improve the behaviour of the closed loop, the controller must be tuned. This tuning has been made heuristically. To find the suitable parameters that would work with the model, many simulations have been done, tuning the following controller parameters simultaneously:

- **Prediction and control horizon:** given that this model has no cycle-to-cycle dynamics, if the controller manages to find a solution to bring the outputs to its references, it will be done rapidly. However, given that this will not be the real case, prediction and control horizons have been chosen. In these simulations, the MPC will predict 50 samples ahead the outputs for the system, and it will compute the following 3 control signals.

- **Output constraints:** it is clear that if there are not constraints the controller will have more options to bring the process to its reference. However, the pressure inside the engine should not exceed a determined limit. Therefore the output that refers to the maximum pressure will have an upper-bound. In addition this constraint will be considered as a hard constraint, because higher pressures in the cylinder could damage the engine.

- **Output weights:** these parameters are important because they will affect the controller so that the process behaves as the user wants to. It is important to note that this model has two inputs and three outputs, which means that the controller has two degrees of freedom to control three signals, which means that the control problem could be ill-posed. Nevertheless, the maximum pressure does not have to follow a reference, but only fulfil a constraint which should not be violated. Therefore, the weight chosen for this output will be considered to be 0. The other two outputs can be chosen freely, depending on if it is more interesting to obtain the reference of the IMEP_g or the θ_{CA50}.

- **Input constraints:** given that the inputs have been treated as incremental inputs, to add a constraint to one of these inputs will only affect the variation of the value of the input, but not its absolute value. The solution taken in this case has been to add two extra states and two extra outputs to the model. These extra outputs will keep the absolute values of the inputs. This way in order to add absolute constraints to the inputs it is enough to add them to the extra outputs. The mathematical description of the extended model is in (4.1).

$$\begin{bmatrix} x(k+1) \\ x_n(k+1) \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & I_2 \end{bmatrix} \begin{bmatrix} x(k) \\ x_n(k) \end{bmatrix} + \begin{bmatrix} B \\ I_2 \end{bmatrix} \Delta u(k)$$
$$\begin{bmatrix} y(k) \\ u(k) \end{bmatrix} = \begin{bmatrix} C & 0 \\ 0 & I_2 \end{bmatrix} \begin{bmatrix} x(k+1) \\ x_n(k+1) \end{bmatrix}; \quad x_n(0) = u(0) \tag{4.1}$$

Here, $x_n(k)$ are the two new states which keep track of the absolute value of the inputs.

- **Input weights:** in a similar way as with the input constraints, the weight to the inputs can also be added to the extra outputs. In this system it is interesting to minimize the fuel consumption, using the injection timing (θ_{SOI}) to make the outputs reach its reference when it is possible.

- **Kalman filter:** the Kalman filter helps to predict the actual states despite the noise in the system. Given that only simulations have been carried out, the Kalman filter has not taken an important role in this part. However, when working with the real engine it will be necessary to tune the Kalman filter in order to obtain the correct values of the states. Note that it is also possible to use the Kalman filter obtained from system identification.

4.2 Simulations

In order to improve the behaviour of the controller, a large number of simulations have been performed. An example is shown in Figure 4.1, where 100 cycles were simulated. In these graphs all the inputs and outputs are shown as well as the gross efficiency (η) of the engine (calculated from the gross IMEP). The outputs corresponding to the IMEP and the θ_{CA50} follow a reference (red line), meanwhile the third output (p_{max}) has only an upper bound that should not be exceeded (green line). The inputs are shown with their absolute value, although the controller works with incremental inputs. The efficiency is calculated using

$$\eta = \frac{\text{IMEP}_\text{g} \cdot V_d}{m_f Q_{LHV}} \tag{4.2}$$

In this example, the outputs are followed by its reference rapidly except for one section (cycles 50-75) where the maximum pressure is on its limit (around cycle 55 the constraint is violated because of model error). In this section the θ_{CA50} is close to its reference but cannot reach it. The efficiency remains almost constant during the simulation. This example also includes

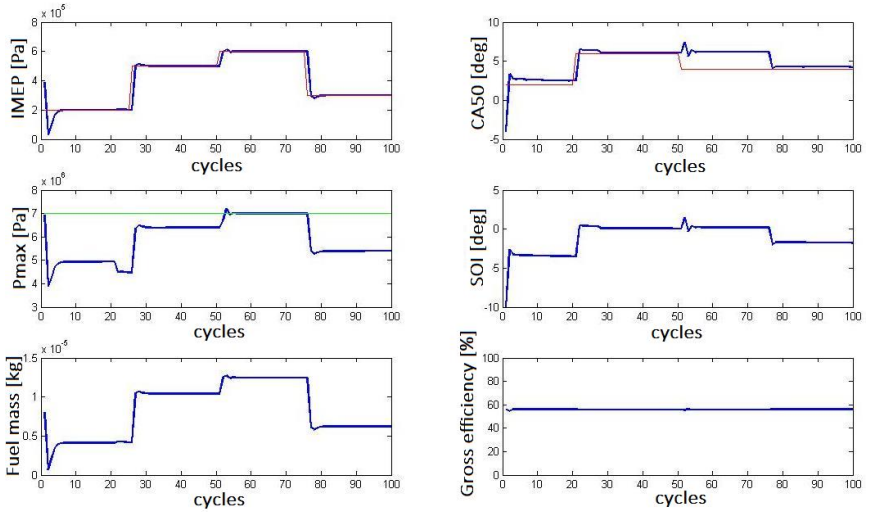

Figure 4.1 MPC simulation

the application of the smart switch described before, although the engine is mainly working in the side corresponding to $\theta_{SOI} > \arg\max_{SOI} \text{IMEP}_\text{g}$.

To make the controller more robust and more realistic, some white noise has been applied to the ignition delay and the combustion duration, simulating more realistic engine behaviour with stochastic cycle-to-cycle variations (Figure 4.2).

The performance here is similar to the noise-free simulation, except when the maximum pressure is closer to its limit. Here, the limit is exceeded because of the noise, and the θ_{CA50} output becomes less stable during this section. However, the MPC is able to control the engine showing robustness to the white noise.

It is clear that the efficiency increases when the fuel consumption decreases and the IMEP remains constant. This may be achieved by increasing the weight to the fuel mass input, so that the controller tries to increase the power of the engine by modifying the θ_{SOI} instead of the fuel mass.

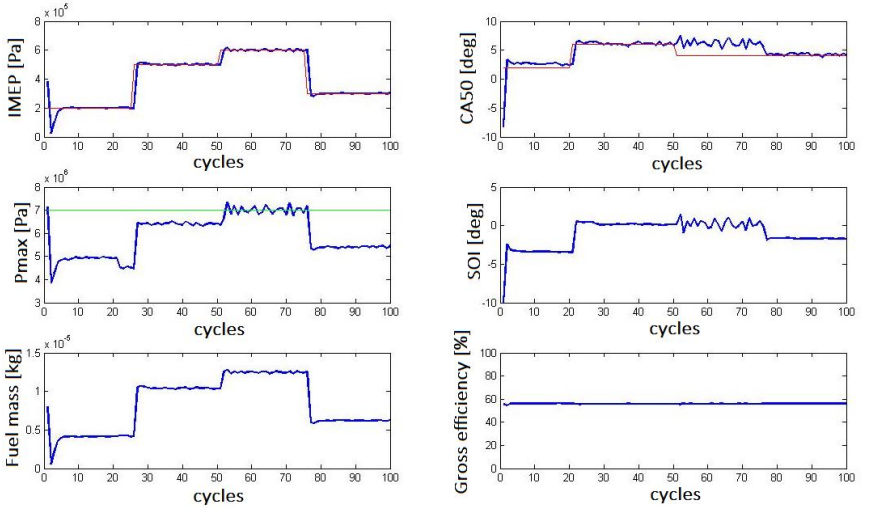

Figure 4.2 MPC simulation with white noise

5

Conclusions

The design of a model and a controller for simulations is the first step to control a PPC engine. The results achieved from these simulations could be used as a guideline for real engine experiments and controller design. After the study of the simulations, some conclusions were made:

- Using MPC techniques for controlling a PPC engine is a good option. The characteristics of the MPC and the chance to add constraints and weights to some parameters in the system will be useful in order to improve the efficiency reducing the fuel consumption which is the main objective of the PPC engine. Reducing the emissions would also be possible if the model is extended covering that part.

- The process has shown to be stable at least during the simulations. Although a validation model with the real process has to be done, this first approximation shows that it is possible to control this kind of systems.

- The last step before working with the real process is to add cycle-to-cycle dynamics to the physical model. The equations that should be used in this more complex model are already stated in Section 3.1. Some data could be obtained from these simulations and might be used to get a new state space model. In the next part of this thesis, it is shown how to work with data extracted from a real engine and to use it for designing a MPC controller.

- The MPC will need a lot of tuning when working with the real process. Kalman filter design will have to be taken into account, apart from all the other parameters (e.g. horizons, weights). The model used for the MPC will also need the tuning of some of its parameters that are unknown at the beginning, but can be found after some experiments are done.

- Given that the MPC works with linear models and the engine has shown a non-linear behaviour, more than one linear model may be needed. Smart switches as the designed before will be needed that swaps control between different controllers. However it could be hard to find the switching point.

- Although the computers can calculate large quantities of mathematical operations, it is important to check that the MPC is able to solve its optimization problem when the engine is running at the highest speeds. An optimization problem can be time consuming depending on the number of parameters involved, which can lead to long calculation times. For example, if the engine is running at 800 rpm the computer has about 75 ms to find the control signal, but if it runs at 2000 rpm the time for finding the control signal is reduced to 30 ms.

- This controller is able to deal with constraints, which is useful to restrict some variables. However, the noise can cause this variables to exceed its limit when they are too close of the bound. For this reason it may be interesting to choose a limit according to the noise variance so that if the constraint is violated, the real hard constraint will not be violated.

Part III

CI Engine Identification and Control

6

Compression Ignited data

In this part of the thesis the engine studied is a conventional compression ignition (CI) engine, i.e. a typical diesel engine. Experimental data extracted from the engine is analysed and used to design mathematical models in order to create a MPC controller. Using the models and the controller, it is possible to draw conclusions about the best way to increase the efficiency of the engine and to reduce the emissions. This chapter describes the engine used and presents a first approach to the data that will be used later.

6.1 Diesel engine

The data used in this thesis was obtained from a six-cylinder Volvo D12 heavy-duty engine. This engine is a standard production engine, although some modifications were made in order to extract the data. Engine specifications are given in Table 6.1 [**7**].

All the data used in this chapter has been taken from the experiments done in [**7**].

Parameter	Value
Operated cylinders	6
Total displacement volume	12.2 dm^3
Bore	131 mm
Stroke	150 mm
Connecting rod length	260 mm
Valves per cylinder	4
Compression ratio	18.5 and 14.1

Table 6.1 Specifications for the Volvo D12 engine

6.2 Data

The data extracted from the engine contains the following information:

- **INPUTS**
  - **Fuel injection duration:** this variable will be treated as an input of the system. It represents the injection duration of the fuel in each cycle (DOI).
  - **Start of injection (θ_{SOI}):** the crank angle in which the injection is taking place. $\theta_{SOI} = 0$ represents the injection at TDC. This variable will also be treated an input to the system.
- **OUTPUTS**
  - **Net Indicated Mean Effective Pressure ($\mathrm{IMEP_n}$):** the total work output during the whole cycle divided by the volume displaced by the piston. This variable will be the main output of the system and will be guided by the controller in order to follow a reference.
  - **Crank angle of half of the heat released (θ_{CA50}):** gives information about the timing of the combustion.
  - **Fuel consumption:** the consumption for each cylinder in each cycle in mg (m_f). This output will not follow a reference, but will be used by the controller to minimize fuel consumption.
  - **Ignition delay (τ):** the delay between the injection and the start of combustion can give useful information about the combustion process.
  - **Maximum pressure rise rate (dp):** this rate is related to the noise caused by the engine and should be minimized.
  - **Indicated net efficiency (η):** the efficiency is calculated using the following equation:

$$\eta = \frac{\mathrm{IMEP_n} \cdot V_d}{m_f Q_{LHV}} \tag{6.1}$$

    Therefore, the model used to calculate the efficiency can be obtained directly from the $\mathrm{IMEP_n}$ and m_f.
  - NO_x **emissions:** this output will be used by the controller to minimize or put a constraint to the emission level of $\mathrm{NO_x}$.
  - **Smoke emissions:** similar to $\mathrm{NO_x}$, this will variable will be minimized or constrained by the MPC controller.

To identify system dynamics two data sets were used in this thesis. In the first one, the fuel injection duration input is excited, while θ_{SOI} remains constant. In the second one, the θ_{SOI} was excited meanwhile fuel injection is constant. This means that each output of the data will have two models, one corresponding to the first output and one to the second output. Afterwards, both models can be combined in order to get the complete model of each output. Figures 6.1 and 6.2 are the two data sets used in this thesis. Note that in the second figure the fuel flow sensor does not work properly and the efficiency goes then to infinity during a section of the experiment. The data have been arranged suitably in order to obtain the mathematical models.

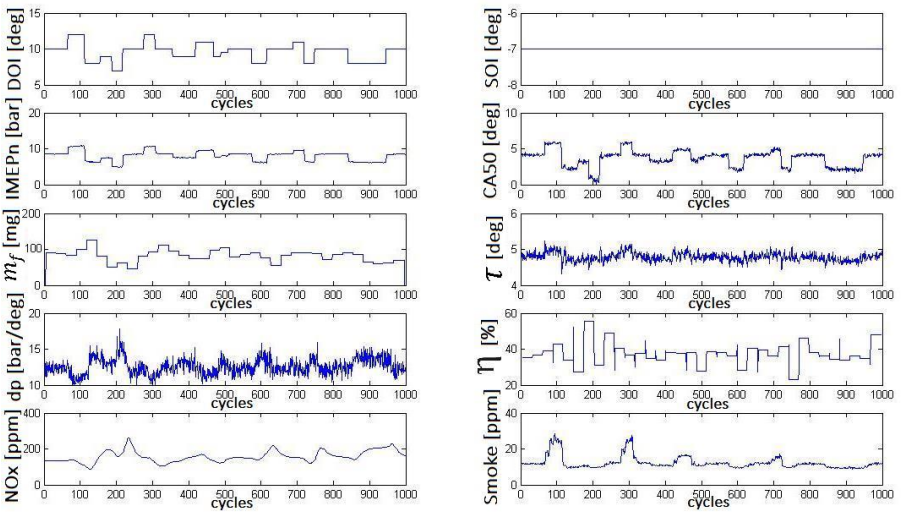

Figure 6.1 Data extracted from V12 Volvo diesel engine with varying fuel injection duration

Before identifying the models, some correlations between signals have been done. These correlations can show information about possible relationships between data. The correlation used here is the Pearson product-moment correlation coefficient

$$\rho_{X,Y} = \frac{\mathrm{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{\mathrm{E}[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X \sigma_Y} \tag{6.2}$$

where, cov is the covariance, σ_X is the standard deviation of X, μ_X is the mean of X, and E is the expectation.

In Tables 6.2 and 6.3 those pair of signals with correlations higher than $|\pm 0.5|$ are marked with a $+$ or $-$ sign, as well as those with really strong

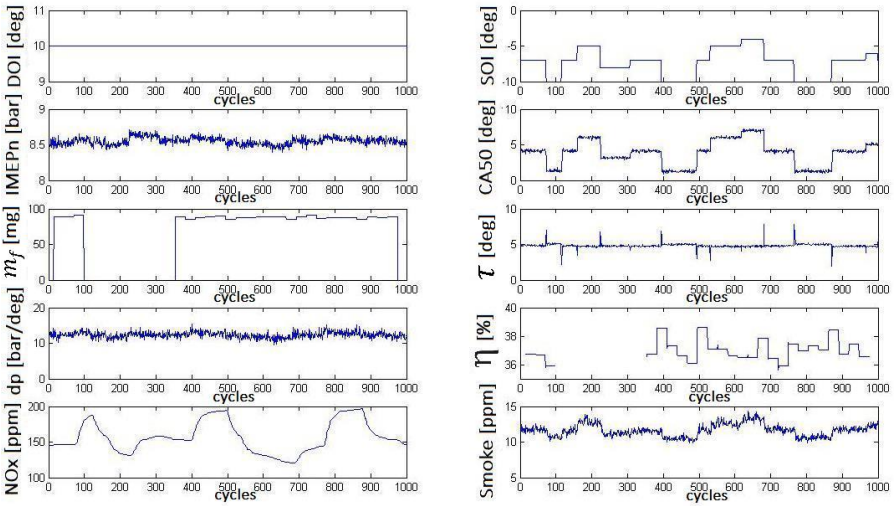

Figure 6.2 Data extracted from V12 Volvo diesel engine with varying θ_{SOI}

	DOI	θ_{SOI}	IMEP_n	θ_{CA50}	m_f	τ	dp	η	NO_x	Smoke
DOI			++	++	0	+	−	0	0	+
θ_{SOI}										
IMEP_n				++	0	+	0	0	0	+
θ_{CA50}					0	+	−	0	0	+
m_f						0	0	0	−	0
τ							0	0	0	0
dp								0	0	−
η									0	0
NO_x										0
Smoke										

Table 6.2 Correlations between signals for data set #1

correlations (higher than $|\pm 0.9|$) are marked with ++ or −−. The pair of signals that do not have enough correlation are marked as 0.

After looking at the tables, it is possible to draw some conclusions:

- There is a strong positive correlation between DOI and IMEP_n and θ_{CA50}. This may have sense given that if there is more fuel in the cylinder, the combustion process will cause higher pressures, which will cause more work output and the combustion duration will be extended.

	DOI	θ_{SOI}	IMEP_n	θ_{CA50}	m_f	τ	dp	η	NO_x	Smoke
DOI										
θ_{SOI}			−	++	0	0	−	0	−	++
IMEP_n				−	0	0	0	0	0	−
θ_{CA50}					0	0	−	0	−	++
m_f						0	0	−−	0	0
τ							0	0	0	0
dp								0	0	−
η									0	0
NO_x										−
Smoke										

Table 6.3 Correlations between signals for data set #2

- With more fuel it is logic that there are more smoke emissions since the fuel air mixture is richer.
- θ_{SOI} and θ_{CA50} are obviously correlated because if there is a time shift in θ_{SOI} the combustion timing will also be shifted.
- Although it is not reflected in both tables, the efficiency decreases if more fuel mass is used.
- It seems that there is no correlation between DOI and m_f at all. In fact there exist correlation, but there is a delay in the m_f signal. This delay hides the logical correlation between the duration of the injection and the mass fuel used.

7

Black Box Cycle-to-Cycle Modelling of CI

The data provided in the last chapter can be used to identify different mathematical models that can be used in simulations with a MPC controller. Although ideally the inputs should have been excited simultaneously, it is possible to obtain one model for each input-output pair, and finally merge all of them in one single state-space model that will be useful for the MPC. This chapter describes with detail each model obtained and how to get the combined state-space model.

Given that no physical model has been used for the diesel engine, the best way to identify the different models is to use black box modelling (see Section 2.2). Before any identification, all the means have been removed from all the data. This will have to be taken into account in the future, because the models obtained will be according the data with the mean removed. To make this report lighter, all the graphs showing the behaviour of each model with its respective data have been set in appendix A.

The different pairs input-outputs have been modeled by means of *autoregressive models with eXternal Input* (ARX) (7.1) and *autoregressive moving average models with eXternal Input* (ARMAX) (7.2) [**6**].

$$A(z)y = B(z)u + e \tag{7.1}$$

$$A(z)y = B(z)u + C(z)e \tag{7.2}$$

The use of one model or another depends on if it has been necessary to make a model of the noise. These models are characterized by the order of its polynomials (n_A, n_B and n_C for A, B and C respectively) and by its internal delay (n_k). In order to find a good model with not too high order, the Akaike information criterion has been used (see Section 2.2).

Net Indicated Mean Effective Pressure

This is an important model because IMEP_n has to follow a reference given by the driver work output request. Therefore it is necessary to obtain a model that fits the real data well enough. The models found have the following orders:

IMEP_n	na	nb	nk
DOI	5	7	1
θ_{SOI}	4	2	2

Table 7.1 IMEP_n-Input models orders

As can be seen in Figure A.2, DOI will affect more than θ_{SOI} when controlling IMEP_n because of the amplitude of the output. The first model is really well-fitted, and although the second one is noisier, it captures the mean. Using an ARMAX equation didn't improve the model.

Combustion timing

θ_{CA50} will not be forced to follow a reference signal in these experiments, although information about the combustion timing is important. The models found have the following orders:

θ_{CA50}	na	nb	nk
DOI	9	6	1
θ_{SOI}	1	3	1

Table 7.2 θ_{CA50}-Input models orders

The first model is of high order, but fits the data well despite the noise. The second model is lower order with similar results (Figure A.3).

Fuel mass

Given that the fuel consumption is directly related to DOI, there is no model from θ_{SOI}.

m_f	na	nb	nk
DOI	1	1	6
θ_{SOI}	0	0	0

Table 7.3 m_f-Input models orders

Because of the delay in the signal the model cannot be more accurate. The model obtained from the identification procedure has the output shown in Figure A.4. The delay in this signal is due to the way of measuring the fuel consumption. Therefore the gases where measured with delay. As a consequence, this delay will be removed from the model without adding any kind of error. In addition, to improve the performance of the controller, the time constant of this first order model has been decreased. The knowledge about the logical relation between DOI and m_f lets to make this modification without adding any kind of uncertainty to the model.

Ignition delay

Although the ignition delay is more useful in PPC engines, it can provide information in diesel engines as well [**13**]. The models orders are as follows:

τ	na	nb	nc	nk
DOI	5	9	-	1
θ_{SOI}	1	2	3	1

Table 7.4 τ-Input models orders

Given that the data are very noisy, ARMAX equations have been used. It has been found that in the second model, using ARMAX instead of ARX fitted better and with lower order. The spikes in the second graph have been captured given that these are not considered to be noise (Figure A.5).

Maximum pressure rise rate

This output will have to be minimized in order to decrease the noise of the engine.

dp	na	nb	nk
DOI	1	5	10
θ_{SOI}	1	1	1

Table 7.5 dp-Input models orders

Given that this signal is obtained from a derivative, it is very noisy. However, both models manage to obtain the mean of the data (Figure A.6). Note that the first model is delayed 10 samples.

Efficiency

The efficiency can be calculated directly with

$$\eta = k\frac{\text{IMEP}_\text{n}}{m_f} \tag{7.3}$$

where k is the constant that relates the efficiency with the IMEP_n and m_f. This constant is supposed to be

$$k = \frac{V_d}{Q_{LHV}} \tag{7.4}$$

although given that the characteristics of the fuel used are not known, this constant has been obtained from previous calculations of η.

Given that the model that would be obtained from (7.3) would be non linear, the controller would not be able to deal with the efficiency. Therefore, in order to maximize the efficiency, a cost on m_f will be introduced to minimize fuel consumption.

NO_x

There are legal restrictions on NO_x emissions that have to be fulfilled. Therefore, this model should at least capture the tendency of the output:

NO_x	na	nb	nk
DOI	4	10	3
θ_{SOI}	4	1	7

Table 7.6 NO_x-Input models orders

The first model is a high order model, although it approximates well enough the real response (Figure A.7). The controller will manage to minimize NO_x emissions using these models.

Smoke

Although it is interesting to know the smoke emissions, it has be found to have a non-linear behaviour and there was no success in finding linear models for this output. Therefore, it was decided to exclude smoke emissions from the model.

Merging of the models

Given that the engine will be controlled by only one MPC controller, all the models found previously will have to be combined in only one big model. In addition, it will have to be translated into a state space model, because the MPC software used in this thesis needs this format to work.

The state-space model obtained will have the following form:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) \end{aligned} \tag{7.5}$$

where

$$y(k) = \begin{bmatrix} \text{IMEP} \\ \theta_{CA50} \\ m_f \\ \tau \\ dp \\ \text{NO}_\text{x} \end{bmatrix} \quad u(k) = \begin{bmatrix} \text{DOI} \\ \theta_{SOI} \end{bmatrix} \tag{7.6}$$

and A a 55x55 matrix (55 states), B a 55x2 matrix and C 6x55 matrix.

8

CI Control Simulations

After obtaining the mathematical models that describe the behaviour of the engine it is possible to go further and do controller experiments. In order to understand how each parameter affects the efficiency and the emissions, it is necessary to simulate the designed model controlled by MPC. By tuning the MPC in different ways it is possible to draw conclusions regarding possible trade off effects, for instance what happens with the emissions when one wants to maximize the efficiency, or how the prediction horizon affects the computational time.

8.1 MPC Design

As explained in Chapter 4, a MPC has a wide variety of tuning parameters that lets the user to prioritize certain control aspects over others. In this chapter simulation results obtained using different combinations of controller parameters are presented. In the presented figures all the outputs (blue) and inputs (red) are shown. θ_{SOI} was limited to $-15 < \theta_{SOI} < 5$ due to model uncertainties outside this region. IMEP is always shown together with its reference (black) as well as the net efficiency (green) and fuel consumption that are displayed in the same graph. Finally, the constraints (if any) are shown in red. The simulations were done during 500 cycles, although the first 100 cycles of each simulation have been removed given they were considered as a transient.

Using the same nomenclature as in Eq. (2.4) the cost function becomes

$$J(u) = \sum_{i=0}^{p-1}(|w_{\text{IMEP}}\left(y^{\text{IMEP}}_{k+i+1|k} - r^{\text{IMEP}}_{k+i+1}\right)|^2 + |w_{m_f}\left(y^{m_f}_{k+i+1|k} - r^{m_f}_{k+i+1}\right)|^2 + |w_{dp}\left(y^{dp}_{k+i+1|k} - r^{dp}_{k+i+1}\right)|^2 + |w_{\text{NO}_\text{x}}\left(y^{\text{NO}_\text{x}}_{k+i+1|k} - r^{\text{NO}_\text{x}}_{k+i+1}\right)|^2) + \sum_{i=0}^{m-1}(|w_{\Delta\text{DOI}}\Delta u^{\text{DOI}}_{k+i|k}|^2 + |w_{\Delta\theta_{SOI}}\Delta u^{\theta_{SOI}}_{k+i|k}|^2) \quad (8.1)$$

where w_{IMEP}, w_{m_f}, w_{dp} and w_{NO_x} are the weights to their respective outputs, $w_{\Delta\text{DOI}}$ and $w_{\Delta\theta_{SOI}}$ are the weights to the variation of the inputs, considering them to be constant during all the simulations. The reference of IMEP will be similar to the identification data, while the reference to the other outputs will be set to 0 in order to minimize those outputs.

Figure 8.1 will be used as a reference for all the experiments. In this experiment only IMEP has a reference (IMEP_{ref}), and there are not any constraints or other weights, except for the input rate of change (in order to avoid overly aggressive controller action). There is a change of IMEP_{ref} every 50 cycles. This frequency has been chosen according to the frequency of IMEP and input changes in data. The prediction horizon of the reference experiment has been chosen also of 50 samples given that the longest time constant of all the signals is around 50 samples. The control horizon has been chosen as the half of prediction horizon (25).

8.2 Prediction horizon and computational time

To begin with, prediction and control horizon were varied. These are important parameters that may affect the stability of the process if chosen incorrectly. If the prediction horizon was chosen too short, IMEP_{ref} was followed too slowly (Figure 8.2) and the fuel consumption was higher, while if chosen too large, the computational times may grow too much.

In order to understand how the horizons affect the computational time, 3 different experiments were done. The first will have the prediction and control horizons used as reference, the second will have shorter prediction and control horizons and the last one will have longer prediction and control horizons. For each of the experiments the time needed by the controller to compute the control signal in each cycle was calculated. The ratio between both horizons has been kept constant during all the experiments ($m = p/2$). The results are given in Table 8.1.

The table shows that the computational time increases exponentially with the horizons. However, this increase is not very strong for this case. The

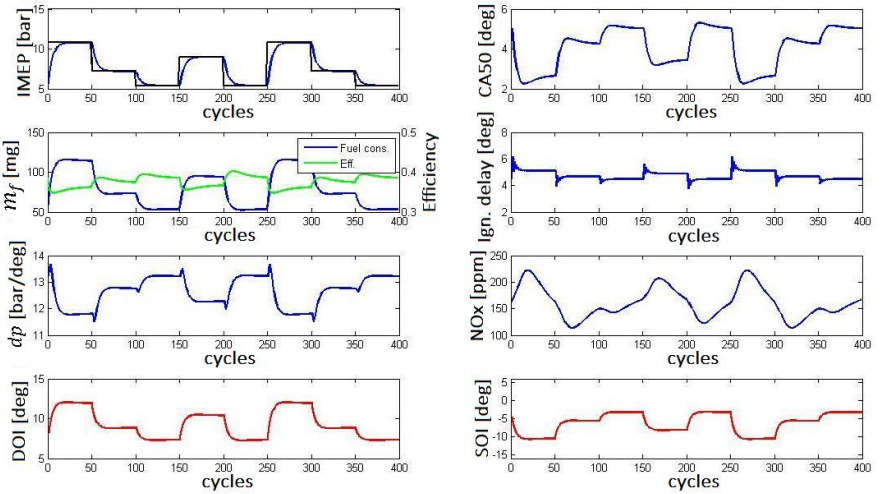

Figure 8.1 Control with default parameters

Experiment	Prediction horizon	Control horizon	Elapsed time (ms)
1	2	1	4.834
2	50	25	7.632
3	100	50	13.532

Table 8.1 Computational time for different prediction and control horizons

times obtained in these experiments cannot be compared with the real time needed for the controller because the experiments have been performed with a laptop that probably will not have the same characteristics as the processor of the engine. Nevertheless, it is interesting to see the time constraints that a controller in a real engine would have to perform calculations. Considering an engine running at 4000 rpm, which means 2000 cycles per minute:

$$t = \frac{2 \cdot 60}{4000} = 0.030s = 30\text{ms} \tag{8.2}$$

Considering that the controller only has the half of the cycle to perform calculations, the controller should calculate the control signal in less than 15ms.

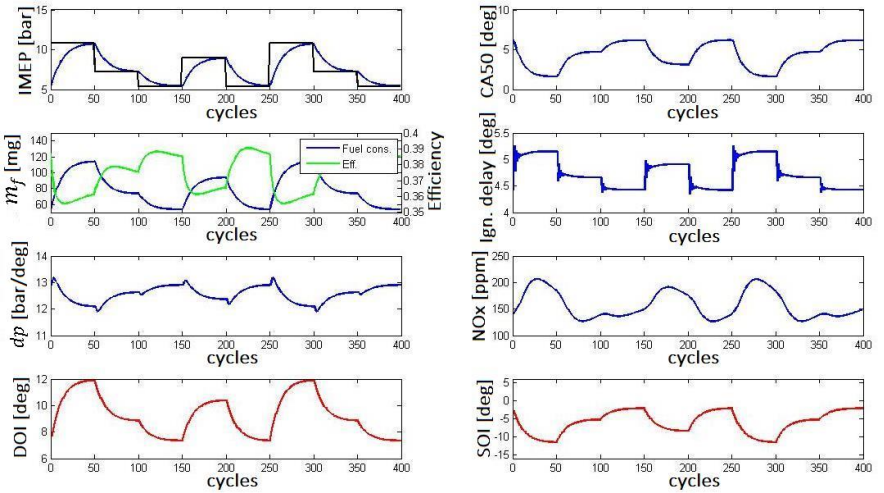

Figure 8.2 Control with small prediction horizon

8.3 High efficiency control

A high efficiency is imperative for environmentally friendly engines. As explained before, the efficiency will be maximized by minimizing the fuel consumption (i.e., increasing w_{m_f}). Figure 8.3 shows the behaviour of the engine when only the efficiency is maximized (without having any restrictions on NO_x emissions or the maximum pressure rise rate). θ_{SOI} is advanced and NO_x levels increase over 300 ppm (which is higher than the default experiment). The efficiency is improved compared to the default case, but the NO_x levels make this type of control environmentally unfriendly.

The physical reason for this behaviour is that if the combustion takes place before TDC the pressure in the cylinder increases a lot given that the piston is still comprising the mixture. Nevertheless, this can lead to some negative work (caused by the explosion when the piston is still comprising) that it has not taken into account.

If the efficiency is increased even more, the results are as shown in Figure 8.4. NO_x levels increase a lot and the most important is that the $IMEP_{ref}$ is not longer followed correctly by the controller, since the cost function penalizes fuel consumption more than having zero IMEP control error. Therefore this is a practically unwanted solution. Nevertheless, it is possible to find out that the best efficient working point for the engine is when the θ_{SOI} is being advanced.

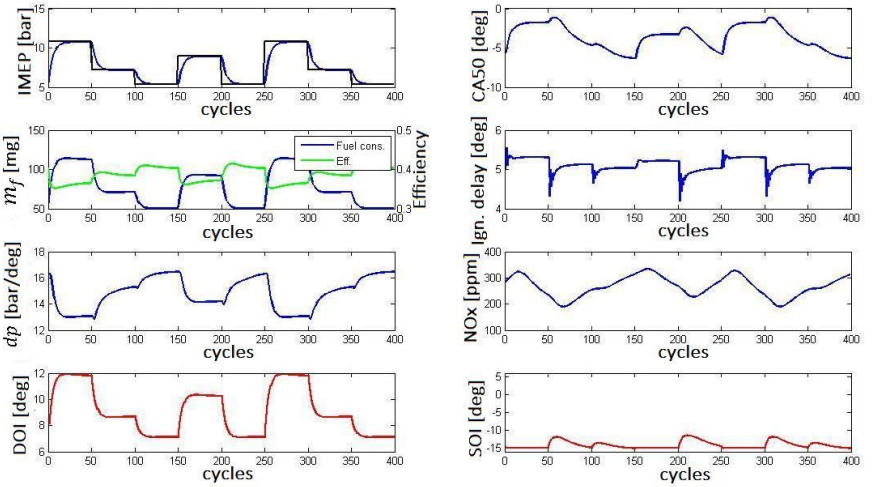

Figure 8.3 Control with high efficiency

8.4 Low NO_x emissions control

Instead of trying to find the best efficiency, it can be interesting to study how the controller behaves if one wants to minimize NO_x emissions by increasing w_{NO_x}. In Figure 8.5 it is possible to see how the NO_x emissions can be decreased with respect to the default case. In order to achieve this behaviour, the controller delays θ_{SOI} although DOI is not affected too much. Therefore the fuel mass and finally the efficiency are not reduced excessively.

The physical reason for the reduction of NO_x emissions has to do with the temperature of the mixture. Given that the combustion takes place after TDC, the pressure in the cylinder will not increase too much, and therefore the temperature of the mixture neither. Since NO_x formation takes place when the temperature is high [**9**], delaying θ_{SOI} will reduce NO_x emissions.

The model used to compute NO_x emissions exhibit some transients that might not be close to those of real emission levels (this model uncertainty could be fixed using larger identification data sets). For example, it is possible to see that NO_x emissions reach 0 level when they should not. This is because the data used to identify the models lack excitation for lower frequencies. However, given that the purpose of the model is to give a quantitative view, the results have been considered acceptable.

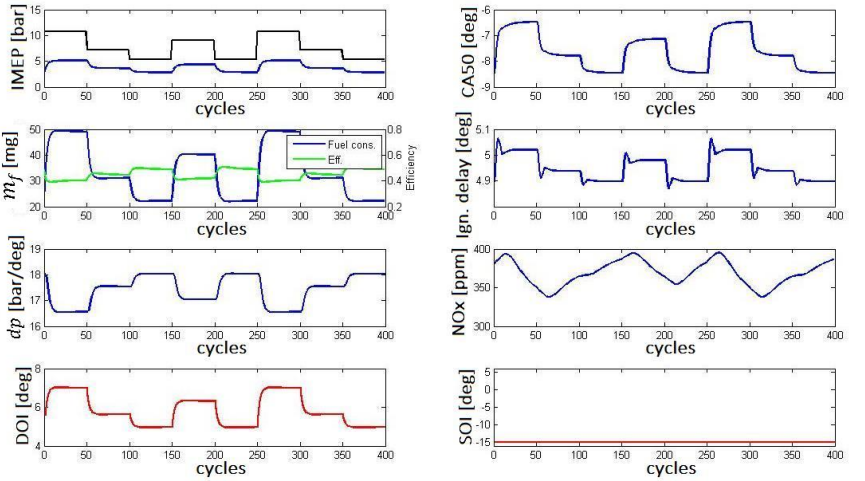

Figure 8.4 Control with extremely high efficiency

8.5 Balanced behaviour

In a real engine, it will be interesting to have a high efficiency while keeping low NO_x emissions. In addition, the maximum pressure rise rate should also be reduced in order to maintain the noise caused by the engine at acceptable levels.

Figure 8.6 shows a tuned MPC controlling the IMEP while having lower fuel consumption, maximum pressure rise rate and NO_x emissions than the reference experiment. Given that in some cases NO_x levels could increase more than necessary, a constraint has been added to this output (red line), limiting its value to 150 ppm. In the figure it is possible to see that when NO_x emissions reach its boundary, IMEP has a small error. This could be solved by increasing w_{IMEP} on the controller (which will decrease the efficiency) or by softening the constraint set on NO_x emissions. Nevertheless, given that the difference between IMEP_{ref} value and its signal is really small, the MPC has not been modified.

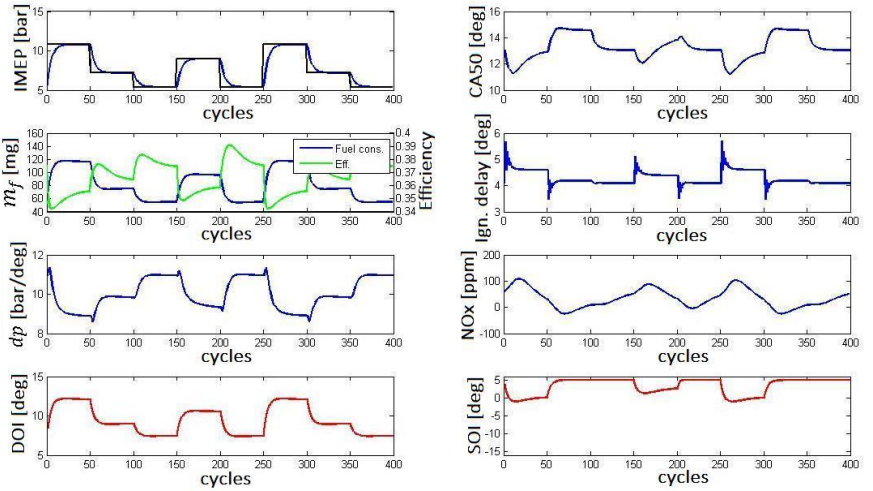

Figure 8.5 Control with low NO_x emissions

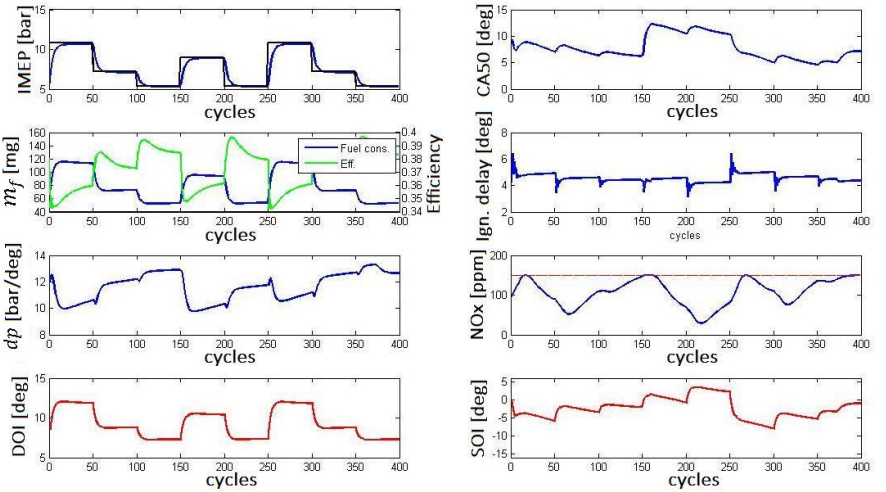

Figure 8.6 Balanced control with constraint in NO_x emissions

9

Conclusions

The experiments done in the last chapter lead to some useful conclusions that can be used in the future to improve diesel engine performance and that may be used as guidelines for CI MPC tuning. The conclusions obtained from the experiments are the following:

Data

- All the simulations done are subject to the characteristics of the identification data. The results can be uncertain if these notes are not taken into account:

  - It is not feasible to perform experiments with different reference or input frequencies than those of the extracted data. Several models obtained in the identification procedure may not behave correctly if the frequency is too low or too high compared to the frequency of the data.

  - All the signals (specially the inputs) should be kept around the operation point they were in the original data. Using different input values may lead to strange behaviour due to non-linearities in the real process. This is the main reason the θ_{SOI} has been constrained in the simulations.

MPC extra tuning parameters

- In some situations the controller may try to force the IMEP reach its reference too fast (few cycles). In order to avoid this sharp behaviour, a weight for the rates of the inputs has been added. In this way the inputs may not change drastically and therefore IMEP and the rest of outputs will not change so drastically either.

Horizons and computational times

- A suitable prediction horizon was found to be around 50 samples. This number has been got from the longest time constant of the engine model, which corresponds to NO_x emissions. The control horizon has been chosen as the half of the prediction horizon, and this ratio has been kept constant during all the simulations. These horizons could be used as starting point when tuning the MPC in the real engine.

- The relation of the computational time with the horizons has been found to be exponential with the software and laptop used, although with low rise rate. Therefore, it is possible to increase the horizons if needed without compromising the time for the controller to compute each control signal.

- Given that the laptop used in these simulations is not optimized for this algorithm, shorter computational times may be found when using a different MPC implementation. In addition, the software used to minimize the cost function may not be optimal [**1**]. However, it has been shown that with the computational times obtained in these simulations, it is possible to deal with an engine running at high speeds (4000 rpm).

MPC weights and constraints tuning

- Only to maximize engine efficiency is not a good option because NO_x emissions may grow too much. In addition, if the efficiency is set too high, the IMEP output may not follow its reference, losing too much engine power.

- The best efficiency point is with earliest θ_{SOI} possible. This point will not be reached by the engine in most cases though, because of high NO_x levels. However, it is interesting to know how to achieve the best efficiency for especial cases where it is extremely important to save fuel, even if it means to contaminate more or not to reach high engine power.

- Minimizing NO_x without having any weight in the fuel consumption does not cause the efficiency to decrease too much. For this reason, it is possible to decrease NO_x emissions without compromising the efficiency of the engine.

- The tuning of the weights of the MPC can be summed up in a trade-off between the NO_x emissions and maximum pressure rise rate and the fuel consumption (or efficiency). This will be the most important parameter to be tuned in the real engine. However, it has not been found to be difficult to tune in simulations. It is also important to note

that if too much importance is given to these parameters, the IMEP_{ref} may not be followed correctly.

- Adding an upper constraint on NO_x emissions helps to keep w_{NO_x} low. This way it is possible to have both high efficiency and NO_x emissions below a certain level. When the emissions reach the constraint, the IMEP signal cannot follow the reference perfectly. However, the error is kept low and the engine will give almost all the power requested. Nevertheless, if this is not desirable, there exist two options to avoid this:

  - Increase w_{IMEP}. This will lead finally to a relative decrease on the weight of the rest of the signals, which will provoke a reduction of the efficiency.
  - Use soft constraint for the NO_x emissions. This may be applied only if the top levels of NO_x can be exceeded a bit. However, the constraint could be set in a lower level in order to accomplish with the NO_x restrictions.

A

Identified model output graphs

The first graph shows the input data used to validate all the models. The following graphs show how each model fits each real output. The units for each graph are relative.

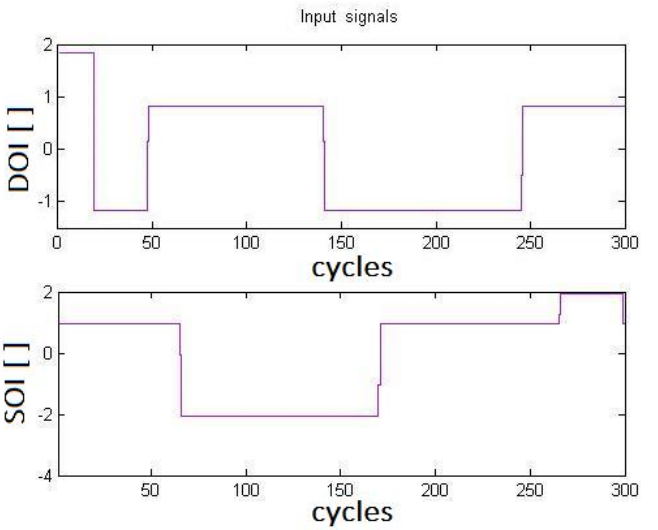

Figure A.1 System inputs

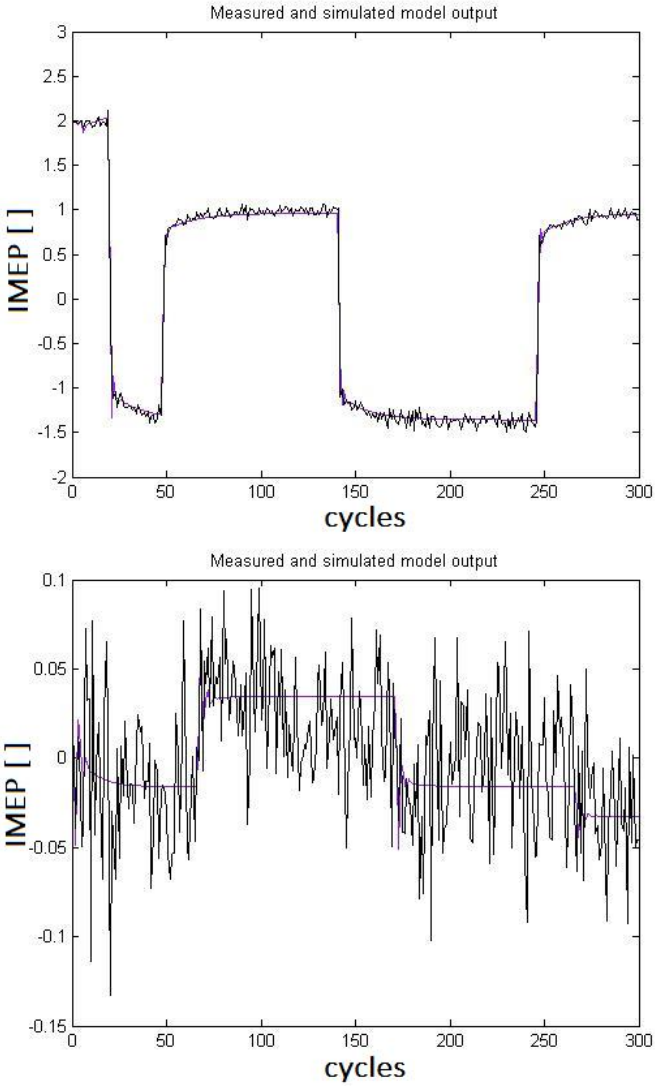

Figure A.2 IMEP

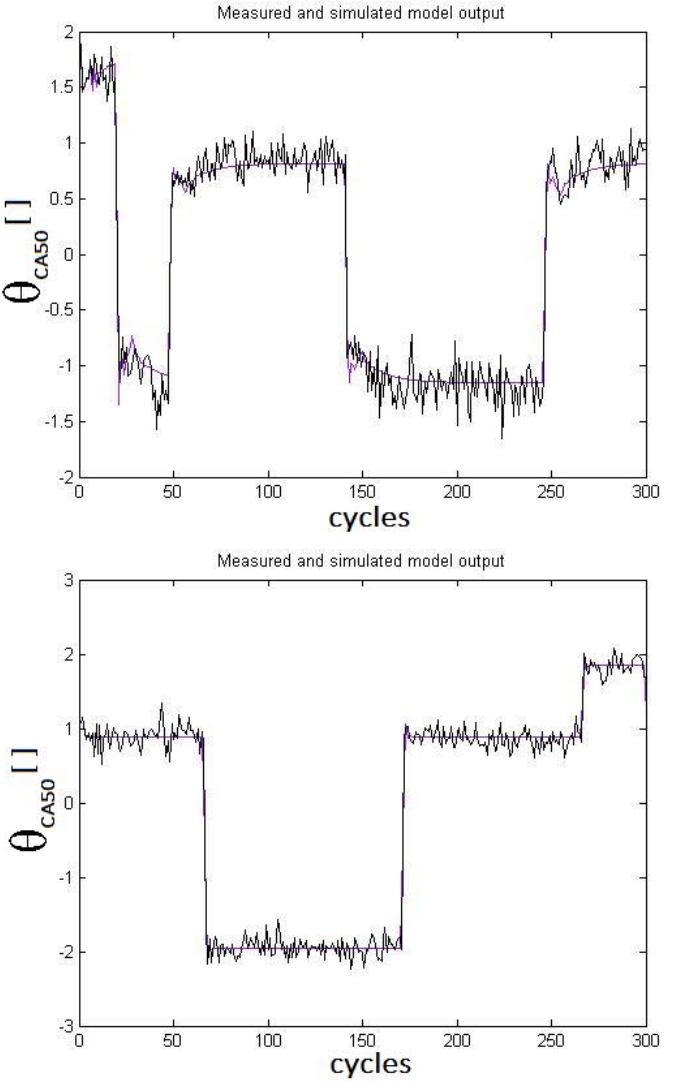

Figure A.3 θ_{CA50}

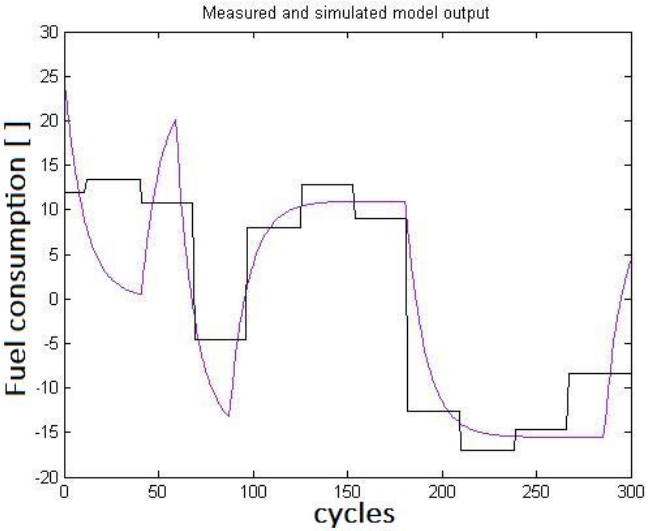

Figure A.4 Fuel consumption

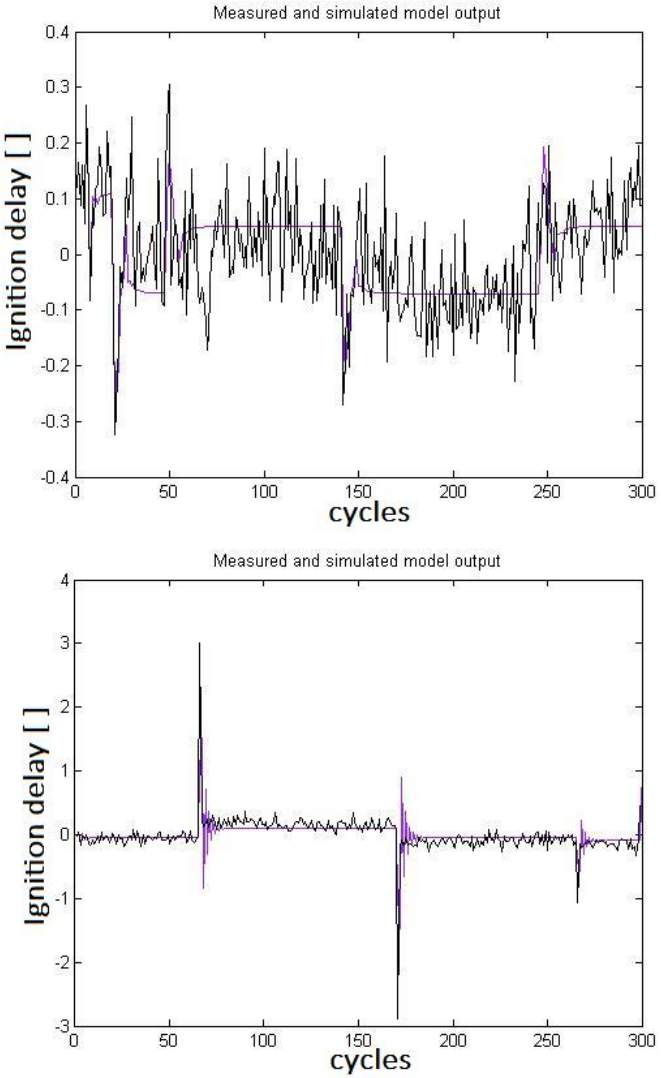

Figure A.5 Ignition delay

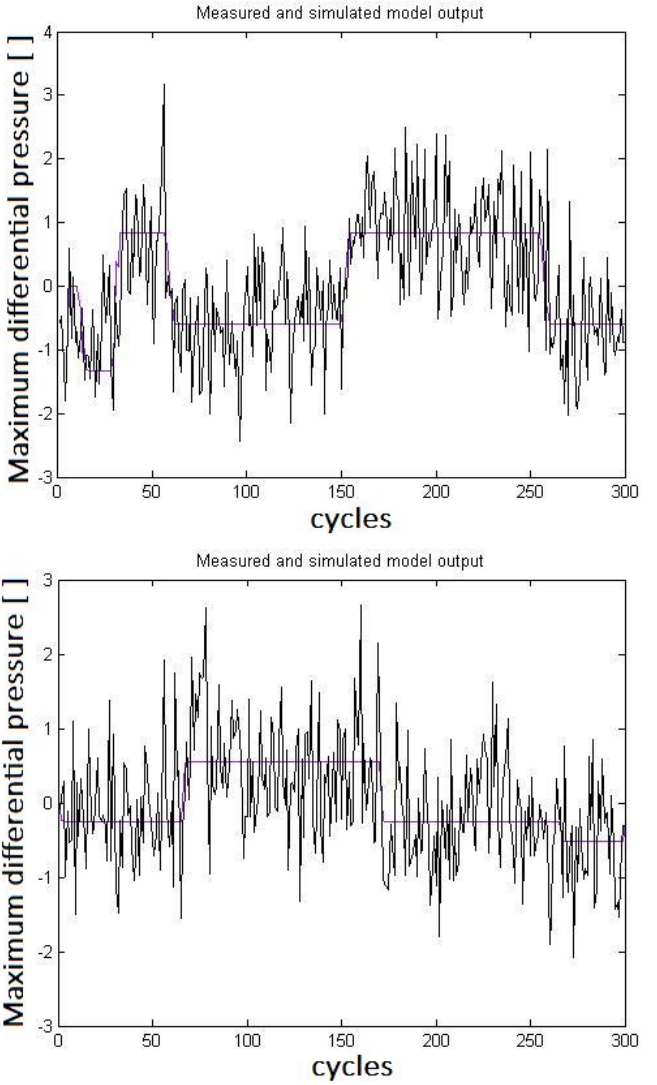

Figure A.6 Maximum differential pressure

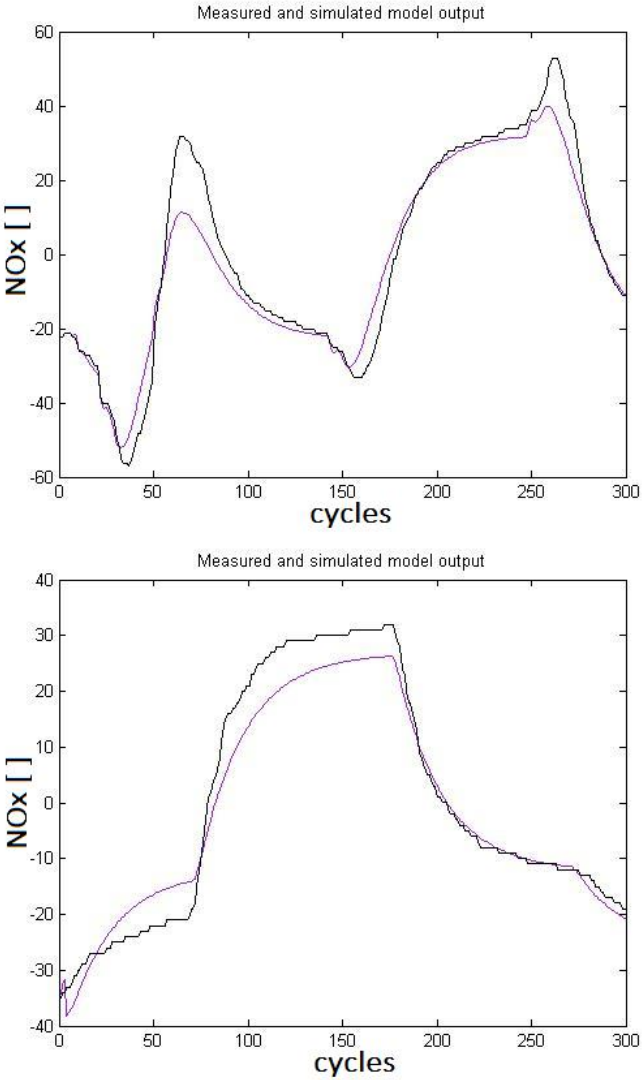

Figure A.7 NO_x

Bibliography

[1] A. BEMPORAD, M. MORARI, N. L. RICKER, *Model Predictive Control Toolbox User's Guide*, The Mathworks, Inc., 3 Apple Hill Drive, Natick, MA 01760-2098, U.S., 2014.

[2] J. BENGTSSON, P. STRANDH, R. JOHANSSON, P. TUNESTÅL, B. JOHANSSON, *Multi-Output Control of a Heavy Duty HCCI Engine using Variable Valve Actuation and Model Predictive Control*, SAE Paper 2006-01-0873, April 2006.

[3] J. BENGTSSON, P. STRANDH, R. JOHANSSON, P. TUNESTÅL, B. JOHANSSON, *Model Predictive Control of Homogeneous Charge Compression Ignition (HCCI) Engine Dynamics*, Proc. IEEE Int. Conf. Control Applications (CCA 2006), Munich, Germany, October 4-6, 2006, pp. 1675-1680.

[4] J. BENGTSSON, P. STRANDH, R. JOHANSSON, P. TUNESTÅL, B. JOHANSSON, *System Identification of HCCI Engine Dynamics*, Proc. IFAC Symp. Advances in Automotive Control (AAC04), Salerno, Italy, April19-23, 2004, pp.250-255.

[5] J. BENGTSSON, P. STRANDH, R. JOHANSSON, P. TUNESTÅL, B. JOHANSSON, *Control of Homogeneous Charge Compression Ignition (HCCI) Engine Dynamics*, Proc. 2004 American Control Conference (ACC04), Boston, Massachusetts, June 30 - July 2, 2004, pp. 4048-4053.

[6] J. BENGTSSON, P. STRANDH, R. JOHANSSON, P. TUNESTÅL, B. JOHANSSON, *Cycle-to-cycle Control of a Dual-Fuel HCCI Engine*, SAE Paper 2004-01-0941, 2004.

[7] M. HENNINGSSON, *DataRich Multivariable Control of HeavyDuty Engines*, TFRT–92–SE, Dept. Automatic Control, Lund University, Sweden, 2012.

[8] J. B. HEYWOOD, *Internal Combustion Engines Fundamentals*, Automotive Laboratory, Massachusetts Institute of Technology, U.S., 1988.

[9] B. JOHANSSON, *HCCI*, Dept. Energy Sciences, Lund University, Sweden.

[10] V. MANENTE, *Gasoline Partially Premixed Combustion*, TMHP-10/1071-SE, Dept. Energy Sciences, Lund University, Sweden, 2010.

[11] R. JOHANSSON, *Predictive and Adaptive Control*, Dept. Automatic Control, Lund University, Sweden, 2013.

[12] R. JOHANSSON, *System Modeling and Identification*, Dept. Automatic Control, Lund University, Sweden, 2012.

[13] M. LEWANDER, *Characterization and Control of Multi-Cylinder Partially Premixed Combustion*, TMHP–11/1083–SE, Dept. Energy Sciences, Lund University, Sweden, 2011.

[14] A. WIDD, *Physical Modelling and Control of Low Temperature Combustion in Engines*, TFRT–1090–SE, Dept. Automatic Control, Lund University, Sweden, 2012.

Nomenclature

Symbols

Engine variables

θ	Crank angle
θ_{SOI}	Crank angle of the start of injection
θ_{CA50}	Crank angle which half of the heat has been released
m_f	Mass fuel injected
IMEP	Indicated Mean Effective Pressure
τ	Ignition delay
η	Efficiency
dp	Maximum pressure rise rate

General variables

Q	Heat
p	Pressure
T	Temperature
V	Volume
A	Area

Engine related constants

V_c	Clearance Volume
V_d	Displacement Volume
R_v	Ratio between the connection rod length and the crank radius
Q_{LHV}	Lower heating value of the fuel
γ	Heat capacity ratio
h	Convection coefficient

MPC parameters

p	Prediction horizon
m	Control horizon
w	Weight of a parameter in MPC

Acronyms

TDC: Top Dead Center
BDC: Bottom Dead Center
SOC: Start of Combustion
SOI: Start of Injection
DOI: Duration of Injection
SI: Spark Ignition
CI: Compression Ignition
HCCI: Homogeneous Charge Compression Ignition
PPC: Partially Premixed Combustion
MPC: Model Predictive Control

Lund University
Department of Automatic Control
Box 118
SE-221 00 Lund Sweden

Document name
MASTER´S THESIS

Date of issue
June 2014

Document Number
ISRN LUTFD2/TFRT--5943--SE

Author(s)
Daniel Blasco Serrano

Supervisor
Gabriel Ingesson, Dept. of Automatic Control, Lund University, Sweden
Rolf Johansson, Dept. of Automatic Control, Lund University, Sweden (examiner)

Sponsoring organization

Title and subtitle
Combustion Engine Identification and Control

Abstract

The topic of this thesis is system identification and control of two different internal combustion engines, Partially Premixed Combustion (PPC) engine and a more conventional Combustion Ignited (CI) diesel engine. The control of both engines is aimed to emission reduction and to increase the efficiency.

There is an introduction to the internal combustion engine, as well as theory used about system identification and Model Predictive Control (MPC).

A physical model of a PPC engine was designed. With this model, it was possible to perform simulations and to obtain data to design a cycle-to-cycle state space model used subsequently by the MPC controller.

A black box model of the CI engine was designed using data of a real CI engine from another project. After designing the model, a MPC controller was created. The aim of this controller was to reduce emissions and to increase the efficiency of the engine.

Keywords

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title
0280-5316

ISBN

Language
English

Number of pages
1-73

Recipient's notes

Security classification