# The Application of Black Box Models to Combustion Processes in the Internal Combustion Engine

by

Bastian Maass

A Doctoral Thesis

Submitted in partial fulfilment of the requirements
for the award of
Doctor of Philosophy (PhD) of Loughborough University

June 2011

## Abstract

The internal combustion engine has been under considerable pressure during the last few years. The publics growing sensitivity for emissions and resource wastage have led to increasingly stringent legislation. Engine manufacturers need to invest significant monetary funds and engineering resources in order to meet the designated regulations.

In recent years, reductions in emissions and fuel consumption could be achieved with advanced engine technologies such as exhaust gas recirculation (EGR), variable geometry turbines (VGT), variable valve trains (VVT), variable compression ratios (VCR) or extended aftertreatment systems such as diesel particulate filters (DPF) or $\mathrm{NO_x}$ traps or selective catalytic reduction (SCR) implementations.

These approaches are characterised by a highly non-linear behaviour with an increasing demand for close-loop control. In consequence, successful controller design becomes an important part of meeting legislation requirements and acceptable standards. At the same time, the close-loop control requires additional monitoring information and, especially in the field of combustion control, this is a challenging task. Existing sensors in heavy-duty diesel applications for in-cylinder pressure detection enable the feedback of combustion conditions. However, high maintenance costs and reliability issues currently cancel this method out for mass-production vehicles. Methods of in-cylinder condition reconstruction for real-time applications have been presented over the last few decades. The methodical restrictions of these approaches are proving problematic.

Hence, this work presents a method utilising artificial neural networks for the prediction of combustion-related engine parameters. The application of networks for the prediction of parameters such as emission formations of $\mathrm{NO_x}$ and Particulate Matters will be shown initially. This thesis shows the importance of correct training and validation data choice together with a comprehensive network input set. In addition, an application of an efficient and accurate plant model as a support tool for an engine fuel-path controller is presented together with an efficient test data generation method.

From these findings, an artificial neural network structure is developed for the prediction of in-cylinder combustion conditions. In-cylinder pressure and temperature provide valuable information about the combustion efficiency and quality. This work presents a structure that can predict these parameters from other more simple measurable variables within the engine auxiliaries. The structure is tested on data generated from a GT-Power simulation model and with a Caterpillar C6.6 heavy-duty diesel engine.



**Keywords:** Artificial Neural Networks, Optimum Network Topology, Diesel Engine, Combustion Modelling, Emission Modelling, Virtual Sensing, Parameter Observer

## Acknowledgements

It is a pleasure to thank those who made this thesis possible. First I want to thank my supervisors Professor Richard Stobart and Dr. Jiamei Deng who had a large impact on my journey through this project. For their support of my work with discussions and helpful suggestions I am very grateful. I also appreciate their confidence in me and the given responsibilities over this period.

I also want to thank all the staff of the Department of Aeronautical and Automotive Engineering at Loughborough University. They have been great support and very friendly and helpful during all times. Special thanks go to Gale Haywood who always found me a meeting slot in the busy time schedules of my supervisors. Another big thank you is due to Edward Winward who has been great help during my experimental phase of my project and has been a good discussion partner during my work.

I also would like to acknowledge the support of the Caterpillar Incorporation during my project. Special thanks go to Tom Langley and his group at Peterborough. Without their support parts of this thesis could not have been possible.

And last but by no means least I have to owe my deepest gratitude to my family and friends in Germany and the UK. Special thanks go to my parents and my brothers who have been outstanding support throughout the time and who always made me believe in this route. Without their encouragement and advice this work would not have been written. I also would like to thank my friends here in the UK who made Loughborough to a place I enjoyed and found my inspiration. Namely to thank are Matthew Rogers, Benedikt Knauf and Michel Azoulay. Thanks guys!

## List of Publications

- *Modeling techniques, to support fuel path control in medium duty diesel engines,* textbfSAE Technical Paper No. 2010-01-0332
- *Diesel Engine Emissions Prediction Using Parallel Neural Networks,* **American Control Conference, June 2009, St. Louis, Missouri, USA**
- *Prediction of NOx Emissions of a Heavy-Duty Diesel Engine with a NLARX Model,* **SAE Powertrains, Fuel and Lubricants Meeting, November 2009, San Antonio, Texas, USA: No. 2009-01-2796**
- *Artificial Neural Networks - The applications of Artificial Neural networks to Engines* **Intech, ISBN: 978-953-7619-003, January 2011**
- *Accurate and Continuous Fuel Flow Rate Measurement Prediction for Real Time Application,* **SAE World Congress, April 2011, Detroit, Michigan, USA, No. 2011-01-1303**
- *In-Cylinder Pressure Modelling with Artificial Neural Networks,* **SAE World Congress, April 2011, Detroit, Michigan, USA, No. 2011-01-1417**
- *Single NLARX Model for Particulate Matters Prediction of Diesel Engines,* **18th IFAC World Congress, accepted for Publication in September 2011, Milano, Italy**
- *The Challenge of Fuel Path Control at High Load Conditions,* **18th IFAC World Congress, accepted for Publication in September 2011, Milano, Italy**
- *Soot Prediction of Both Steady-State and Transient Operation in Diesel Engines,* **IMECHE Part D, submitted for publication**

# Contents

# List of Figures

# List of Tables

# 1 Introduction

The internal combustion engine is exposed to vast pressure from more stringent emission regulations and the demand for more efficient fuel consumption. Recognisable changes in climate and the increasing demand on fossil resources leave no doubt that internal combustion engines need to be developed into another level in order to confirm their outstanding importance in particular within the domain of transport and heavy-duty applications. The key for further application capabilities is the reduction of emission levels and fuel consumption. The improvement of combustion quality and efficiency is imperative as control of combustion in-cylinder parameters such as pressure and temperature characteristics which contain a considerable amount of information about the combustion process that can be used for monitoring and consequently control.

Particularly in the domains mentioned, diesel engines are the dominant combustion engine type. Nevertheless, over the last couple of years it has been noticed that a "dieselisation" is occurring [1]. The classic sector of heavy-duty applications both on- and off-road has known about the innate advantage of lower fuel consumption and higher efficiency for years. Now, this benefit has also impacted the number of sales for light and medium road vehicles as Schindler mentions in his work [2].

The effect of increasing emissions and, especially, hazardous emissions generated by diesel engines such as $\mathrm{NO_x}$ or smoke puts additional pressure on this engine type. Current advanced engine technologies enable engines to be made more adaptable for instantaneous and varying engine conditions. This includes more flexible component behaviour such as: variable valve timing (VVT), variable geometry turbine (VGT) and variable exhaust gas recirculation (EGR) as well as the control of these applications [1]. However, here is where the challenge starts to grow more severe. With an increase of control demanding technologies, a rise in complexity and non-linearity can be noted and engine processes require an improved closed-loop control.

At the same time, monitoring of engine behaviour is required, ideally with additional sensor applications for feedback. In order to avoid additional sensors and use the explanatory power of certain engine parameters, the in-cylinder conditions became the point of focus. The conditions during combustion enable the detection of engine misfires, power leakages, and can be used to draw conclusions on emission levels. Monitoring the in-cylinder process is therefore crucial and has been realised with special methods such as direct measurement methods in form of pressure transducers, fibre optic cables or strain gauges. Another approach is the indirect pressure recovery where vibration signals in the crank-shaft kinematics or on the cylinder head are used for reconstructing the pressure trace. In addition, modelling approaches have been developed to create a virtual representation of the system. However, these latter methods are not very applicable in on-board diagnostics.

This research work presented for the degree of Doctorate of Philosophy investigates the feasibility of predicting pressure and/or temperature characteristics within heavy and medium-duty diesel engines using artificial neural networks (ANN). ANN are classified in the Black-Box-Modelling domain and are therefore a promising approach for on-board implementation with its real-time capabilities. At the same time, accuracy levels are still sufficient for adequate monitoring and control purposes.

## 1.1 In-Cylinder Conditions and Characteristics

The in-cylinder conditions of a combustion engine are highly dependent on the stochastic nature of 1. the dynamics of entering fluids such as air and fuel and 2. the successive chemical processes. In addition, there are a number of external factors that can influence these stochastics to some extent. The air and fuel path have a considerable effect on these but also timings of fuelling and valvetrain can control some of the stochastics. The considered four-stroke cycle consists of four phases: 1. induction 2. compression, 3. expansion, 4. exhaust.

During the cycle, certain events govern the actual combustion process. During the induction process, the initial conditions for combustion are determined by controlling intake valve opening (IVO) and closure (IVC). These events allow the air to stream in and out of the cylinder

chamber. Depending on piston heads and air intake ports, the air is set into a tumbling or swirl motion in order to create an active environment during fuel injection and, consequently, better fuel distribution throughout the cylinder. With the closure of the intake valve (IVC) and the reciprocating cylinder the compression phase starts. Air is compressed and heated up. When the piston approaches top dead centre (TDC), fuel is injected (start of injection - SOI) into the chamber and mixes with the air and ignites once it has reached the temperature and pressure required to start the chemical reactions. A sudden rise in pressure and temperature due to heat release can be observed. This moment is known as start of combustion (SOC) as shown in the sample pressure and heat release trace of a compression ignition event in figure 1.1. This ideally happens at TDC or after TDC, during the expansion stroke in order to create minimal pumping losses due to created reaction forces. The combustion forces the piston to move downwards and towards the bottom dead centre (BDC). Around this point the exhaust valve opening (EVO) takes place and combustion ends. The formed gases and emissions are finally pushed out during the exhaust stroke. The cycle is completed towards the TDC where the intake valve opens again.

### 1.1.1 The combustion process and its impact on engine performance

The core of the combustion engine is the chemical process of discharging the enclosed energy in fuel in order to push a piston down the cylinder. The control of this process has a significant effect on the power output, efficiency, emissions formation, and engine life. A smooth and evenly spread combustion over the length of the expansion stroke is more beneficial than harsh and immediate reactions during the start phase. Those kind of effects can be achieved with multiple injection events in order to spread the fuel load evenly over the combustible period. This can also prevent abnormal combustion behaviour such as knock or immediate complete combustion where extreme material stresses can occur. In addition, fuel efficiency can be increased due to more precise fuel injections and hence overall engine efficiency can be improved. The product of combustion are emission gases such as $\mathrm{CO}$, $\mathrm{CO_2}$, $\mathrm{NO_x}$, $\mathrm{HC}$ and, characteristic for direct injection diesel engines $(\mathrm{PM})$. The variables affecting emissions formation directly in the cylinder chamber are air-to-fuel ratio, temperature and duration of combustion. The emissions within the main focus of current emission regulations are carbon

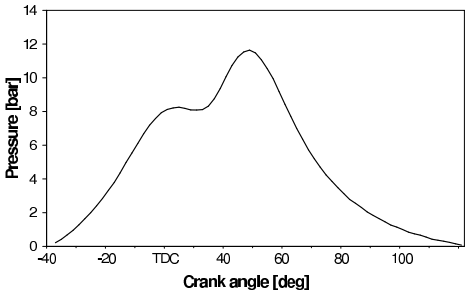


***Figure 1.1:*** *Example of Pressure Trace*

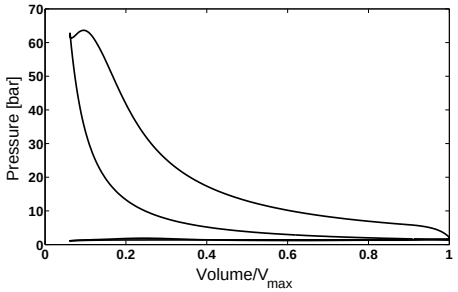


***Figure 1.2:*** *Example of Pressure-Volume Diagram*

dioxide $\mathrm{CO_2}$, nitrogen oxide $\mathrm{NO_x}$ and particulate matters $(\mathrm{PM})$. The formation of $\mathrm{CO_2}$ is mainly dependent on the amount of fuel used during combustion and therefore also a function of engine size. The other two emission types can only be tackled in a trade-off since $\mathrm{NO_x}$ is most likely to be formed at high combustion temperatures whereas $\mathrm{PM}$ is reduced at high temperatures due to better vapourisation of droplets. A more detailed description of emissions formation of nitrogen oxides and particulate matters is picked up within the methodology in chapter 4.

### 1.1.2 Content of information of In-cylinder Pressure and Temperature

As emissions formation and fuel efficiency is reliant on the combustion process, it is of interest to find variables to describe its quality and characteristics. Different parameters, such as burn rates, angles of fuel burned, or end of combustion are used for in-cylinder monitoring. However, they are determined from parameters such as in-cylinder pressure and temperature. Monitoring these two parameters and understanding their characteristics can be used to influence overall combustion quality. The information they provide about engine behaviour can be included in engine control and engine diagnostics.

**In-cylinder Pressure Trace** In-cylinder pressure information is usually presented in a waveform pressure characteristic over the engine crank angle of 720°CA as presented in figure 1.1. This pressure characteristic appears to be similar in phase and magnitude from cycle to cycle.

Slight variations can be detected in steady conditions and underline the stochastic nature

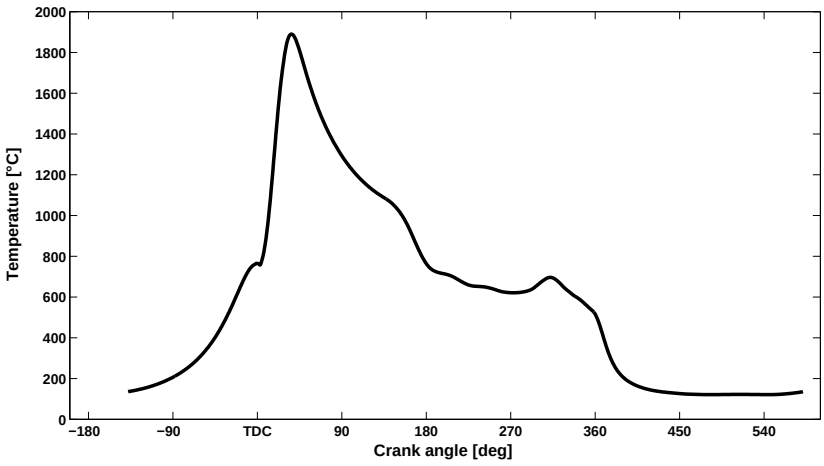


***Figure 1.3:*** *Example of temperature trace*

of the process. Another analytic presentation of the pressure characteristic is the pressure-volume (P-V) diagram. This diagram enables us to calculate the actual work undertaken during the combustion process as well as the pumping losses due to intake and exhaust processes. An exemplary P-V diagram is shown in figure 1.2. With the help of these presentations, considerable changes in pressure characteristics can be detected. A sudden rise may be the reason for a misfire. On the other hand, a drop in pressure may be the reason for a leaking seal due to piston ring failure or valve misfit. Along with a change in engine power output due to abnormal pressure changes, unnecessary engine stresses cause premature wear of parts and costly damage. In addition for lower pressures, emissions formation can also be highly influenced. Fuel spray might not ideally break up and bigger droplets of fuel resulting from that increase the possibility of incomplete combustion and therefore increase PM emissions.

**In-cylinder heat characteristics** Heat development during combustion is usually defined through the instantaneous heat release with starting ignition. In figure 1.3 an exemplary characteristic of a heat release is shown. The temperature characteristic of combustion has a significant effect on emissions formation and material wear. The trade-off between high and low temperatures as mentioned before can have considerable effects on the formation of either $\mathrm{NO_x}$ or $\mathrm{PM}$. At the same time, high temperatures cause higher material stresses during

combustion inside the cylinder as well as in the following passages in the exhaust part of the engine.

## 1.2 Conventional In-cylinder Condition Detection

Since the start of the combustion engine era it has been of interest to determine the in-cylinder pressure that acts in the piston and the actual temperature in the combustion chamber. However, the combustion process is complex and the pressure and temperature distribution varies greatly within the chamber. Consequently, it is difficult to detect the instantaneous value, and measurements can only represent an excerpt of the situation. Therefore, the positioning of the measurement equipment at key locations and choosing the right sensitivity are crucial.

### Pressure Detection Systems

Different systems for pressure detection were developed throughout the years as described in [3]. Bae et al. [4] introduced a sensor integrated into spark plugs using a fibre optic cable. Every pressure perturbation that affects the cable changes the amount of transferred light. This in turn can be related to the pressure. Another approach used a film strain gauge sensor that promised good and accurate results similar to piezoelectric transducers [5]. These transducers are well established in modern experiments. Since the transducer extends into the chamber, the classic approach is to apply the sensor as flush with the cylinder wall as possible [6]. However, in current experimental pressure detection, the sensors integrated into the spark or glow plugs are increasingly used as shown in the work of Schindler et al. [2]. A typical pressure acquisition system can be seen in figure 1.4 (redrawn from [6]).

At the core of these systems is a piezoelectric transducer that senses pressure variations during combustion cycles. The heart of the pressure transducer is a piezoelectric crystal that generates electric charges if it is exposed to an acting load, in this case in-cylinder pressure. The observable electrical charge is proportional to the force and enables us to deduce the instantaneous pressure. It is fitted into a metallic housing with a diaphragm that conducts the pressure to the crystal. An amplifier is required to make the pressure characteristic visible.

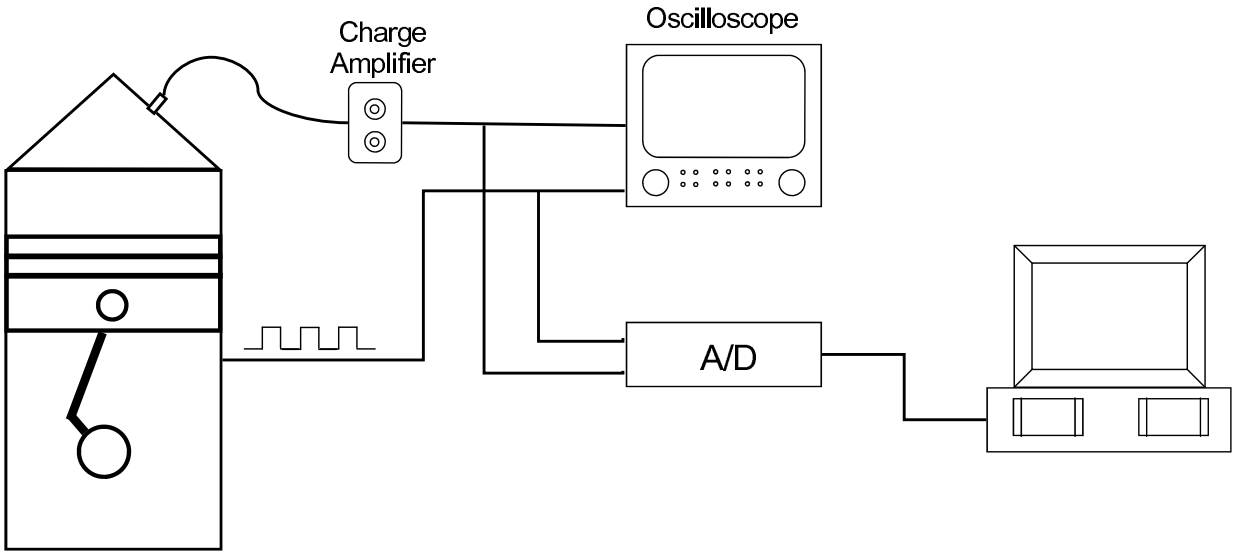


***Figure 1.4:*** *Pressure measurement experiment set-up with pressure transducer and periphery (partially redrawn from [6])*

At the same time, an additional encoder records the crank angles to transfer the pressure into the crank-angle domain [6].

This measurement technique brings a number of significant disadvantages, meaning that this approach is still a technique mostly used in laboratories. Firstly, the method is intrusive. Although the sensor is mounted as flush as possible to the cylinder wall, the intrusive character still exists due to the increased size of the spark or glow plugs. On the one hand, an extension of the sensor would decrease the in-cylinder volume. This changes the conditions of measurement because a decrease in volume affects the pressure characteristic. On the other hand, the sensor requires some assembly space. In the case of existing engines, this can lead to collisions with cooling paths in the engine block.

The sensitivity of the sensors is a second disadvantage. Even though sensors varying in sensitivity are available, the sensor is exposed to extreme environmental forces, which in turn makes it difficult to protect. For instance, pressure forces deform the housing and may falsify measurements. In addition to the mechanical force, thermal effects impact the accuracy. Due to temperature fluctuations the metal expands and may add extra pressure to the sensing unit that influences the signals of the crystal. This issue is overcome by adding extra cooling paths into the housing, which in turn leads to an increase in the dimensions of the sensor itself. Hence, required assembly space and an increase in complexity and costs of sensor development

are encountered. Another disadvantage is low reliability due to the exposure to these heavy acting forces. The extreme conditions shorten the life cycle, meaning that the equipment needs to maintained on a regular basis. This factor applies in particular when it comes to mass series production.

Altogether these concerns render pressure determination costly and less appealing, especially for the purpose of monitoring pressure characteristics in combustion engines, which are in everyday operation. The use of direct in-cylinder measurements can be driven by the performance and efficiency improvements potentially achieved through combustion control based on those sensor applications. However, several strategies have currently been proposed to bypass the direct measurement approach. They are realised through prediction, estimation or reconstruction of pressure waveforms in the combustion chamber. The range of methods is presented in more detail in the following literature review.

### Temperature Detection Systems

Currently, there is no direct temperature detection in combustion engines specially designed for application in a mass production environment. In the field of experimental laboratory set-ups, approaches are mostly based on an optical assembly. A window in the cylinder head or engine block enables the user to access and monitor the combustion process with cameras or light beams.

Alongside the main established optical methods today, another approach was developed prior to laser technology. Livengood et al. used the velocity of sound to measure the temperature inside the combustion chamber [7]. However, this method is not practical when it comes to non-stationary equipment where other noises may falsify the measurements and increase the effort required to detect the signal of interest.

The optical methods referred to as radiation thermometers or pyrometers can be split into two different approaches. The first utilises the existing radiation of material - radiation thermometry. In the second approach, the radiation is additionally excited by a light beam, often a laser, and is known as - scattering method. An extensive outline of those methods is described in the literature by Zhao [6].

The two-colour-method is one of many that have been in use for radiation thermometry for many years [6, 8]. It uses the diesel engine's characteristics of thermal radiation of soot particles and is based on the assumption that the particle temperature is the same as the flame temperature. Two wavelengths are detected, one of which is a reference used to determine a meaningful ratio. In addition, the flame temperature is also capable of detecting the instantaneous soot density.

The other approach is distinguished by the fact that a laser beam is sent into the combustion chamber to excite spectral lines scattering the light. Methods used include spontaneous Raman Scattering, Laser Rayleigh Scattering, and the Temperature Measurement by PLIF as described in [6, 9].

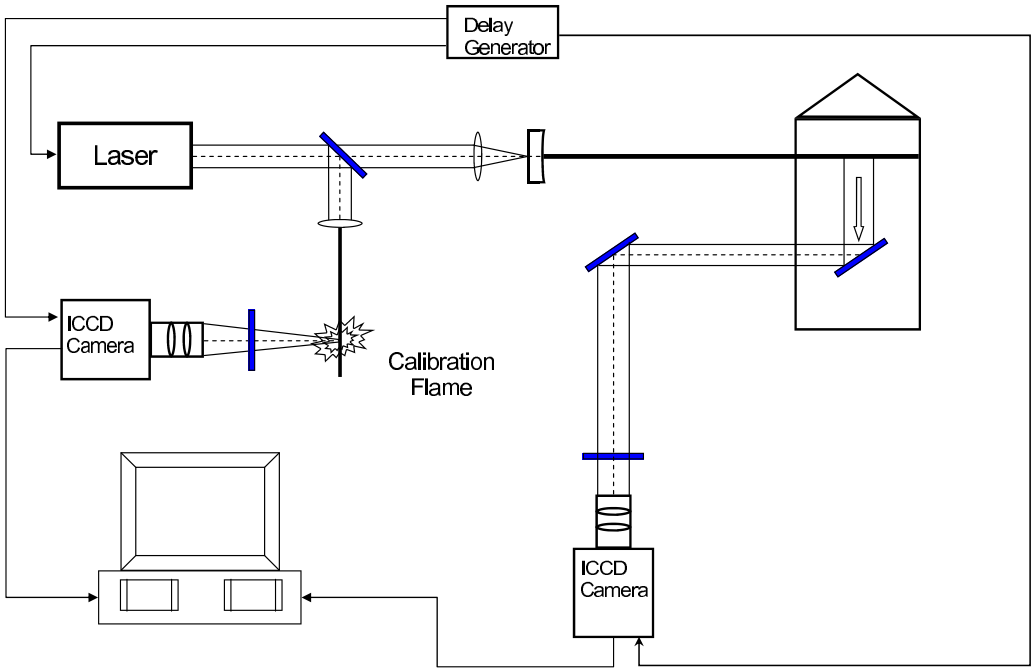


***Figure 1.5:*** *Example of Temperature measurement (partially redrawn from [6])*

All of these optical methods only apply to laboratory work. Even the advantage that they are non-intrusive cannot overcome the fact that their accuracy is highly dependent upon the quality of sight into the cylinder. This is especially apparent in diesel engines where higher soot density makes it more likely that depositions affect the sight. Continuous maintenance of the window would therefore be required. An additional disadvantage is the elaborate equipment of lasers, cameras and mirrors illustrated in figure 1.5. Hence, these technologies are only viable for experimental work rather then an on-board application.

## 1.3 Virtual Sensors for Detection

To date, no suitable approach for measuring temperature and pressure in mass produced cylinder combustion engines has been found. The difficulties encountered in the detection of engine parameters in real-time for control purposes lead to the methods of reconstruction, estimation or prediction. Alternative techniques have been investigated that can be summarised as virtual sensing. This approach is determined by an estimator that predicts costly or immeasurable data from available sensor signals and serves as a virtually sensing diagnostic tool. The long-standing approach of observers is based on physical equations or data maps such as those used in the work of Stephant et al. [3] in a vehicle simulator. The more recent term of 'virtual sensors' is marked by their less required understanding of physical processes. They enable the user to virtually sense a parameter determined on the basis of existing engine and environmental signals. Different techniques, from numerical models to empirical and Black-Box models, have been used to implement this method of diagnosing difficult to measure engine parameters [10, 11, 12].

## 1.4 Non-linearity of the Process

The combustion process itself is highly dynamic and variable in transient engine operation. But even in steady-state points a strong variation in flame and flow characteristics make it difficult to predict an exact combustion process. In addition, the impact of parameters such as EGR or the delay between injection and start of combustion characterise the system as non-linear. Also the complex input choice indicates that the system contains non-linearities. An increase in intake manifold pressure does not necessarily lead to a higher combustion pressure since the time of injection and the mass of injection can considerably diverge at different operating points. Hence, with a retarded injection the peak pressure differs from a previous scenario where the injection took place earlier. In this work the virtual sensor that is generated is outlined as a non-linear predictor in order to satisfy the mapping requirements.

## 1.5 Objectives of the Research Project

The aim of this research is to develop a virtual sensor in order to determine in-cylinder pressure and temperature conditions. Its implementation is based on the recent and current research activities in this field. The research focused on several objectives.

The objectives of this research were:

1. Investigation of applicability of virtual sensing theory to variables in the engine environment such as air and fuel path, engine emissions and in-cylinder combustion parameters.
2. Definition of inputs of interest depending and based on various parameters' impact on engine behaviour and operation characteristics.
3. Analysis of data features for the definition of efficient and minimum training costs.
4. Definition of a virtual sensor structure: preferable model structure in the field of artificial neural networks. Consideration of possible drawbacks of architectures and their applicability.
5. Creation of lean model structures representing least-possible complexity which consequently keeps computational demands to a minimum.

The findings of this research work are partially presented in chapter 4 what includes several extracts of research papers published at conferences in the field of automotive and control. They include findings on the definition of training and validation data for successful training of network structures. In addition, input definition and network construction have been part of the investigations and are presented as well as a contribution to the main results found in chapter 5, 6 and 7. Here, the implementation of a pressure and temperature trace model are presented. One of the main contributions is the choice of inputs such as the training and validation data definition and generation. A new approach of network validation is shown where simulation data not measurable in the real engine environment is combined with real measured data in order to validate a network structure for prediction of in-cylinder temperatures. In addition, a network for in-cylinder pressure prediction is presented. The chosen architecture

enables satisfactory predictive results. One part of these results has been presented in another publication at the SAE congress 2011.

## 1.6 Outline of the Thesis

### Chapter 1: Introduction

The first chapter provided a quick and broad introduction into the importance of in-cylinder condition monitoring, control, and its effect on engine performance followed by an initial outline of possible alternative approaches to existing laboratory methods not applicable to on-board diagnostic tools.

### Chapter 2: Literature Review

The literature review section presents initial work and recent activities on the indirect detection of in-cylinder conditions such as in-cylinder pressure and temperature. Here, a distinction is made between reconstruction and prediction and the advantages and disadvantages of both approaches are highlighted. Together with the motivation and objectives from the introduction, this chapter shall point out the idea and technological gap this research project aims to fill.

### Chapter 3: Theory of Artificial Neural Networks - Structures and Optimisation

This chapter presents the theory of Artificial Neural Networks (ANN). It describes the structures, networks and architectures that have been developed over the last few decades and determines possible candidates for the current problem. In addition, training methods and algorithms are presented that can be considered for the chosen networks. The chapter concludes with an overview of the use of ANN in automotive applications. On the basis of this chapter the reader shall understand the model choice and implementation approach.

### Chapter 4: Methodology and Model Structure

This chapter presents some findings on the topic of network development which are published. They show the feasibility of implementation and capability of ANN on the topic of virtual sensing applications. Main findings in these publications are the training and validation data variation as well as the input definition. In addition, the results of an implementation of ANNs for the design of model predictive control are shown to underline the range of practicability of the chosen modelling approach. These introductory examples form the basis for the resulting model structure in the form of a mixed parallel network, which is also presented here. Its inputs, outputs and specifications are described. This chapter shows the idea and implementation of this project in detail.

### Chapter 5: Data Generation and Acquisition

Chapter 5 covers the field of data generation and acquisition for training and validation sets in order to develop the model. It describes the necessity of data variety and different test scenarios such as steady-state pressure measurements and transient high-load cycles. To generate experimental, data a 6 cylinder in-line heavy duty diesel engine (C6.6) is fitted with the equipment required to detect in-cylinder pressure data. To compensate for missing in-cylinder temperature data, a software model based on GT-Power is set-up - in the form of engine 1D simulation software. This model is validated against data from the C6.6 engine and a validated Dynasty model of this particular engine model. The chapter concludes with data processing and the principal parameters required for the model training.

### Chapter 6: Modelling Results with GT-Power Generated Data

This chapter shows results for a variety of network applications aimed at data generated with a GT-Power model. The in-cylinder parameters, temperature and pressure, are predicted based on engine parameters identified in the previous section. Each model is presented together with its results for both parameters.

### Chapter 7: Modelling Results with Real-Engine Generated Data

Chapter 7 shows the results for the same network applications as presented in the previous chapter. It validates the applicability of the networks based on experimental and noise contaminated signals. Data from the diesel engine are used as inputs and outputs. In addition, the temperature trace generated with a GT-Power model is used to train networks. The in-cylinder parameters, temperature and pressure, are predicted based on engine parameters identified in the previous section. Each model is presented with its results for both parameters.

### Chapter 8: Summary and Conclusion

The conclusions for the developed models are presented here and contain the novelty of the model structure approach and the data set generation by mixing computed and measured data for development purposes. An outline for its potential applications is made in the field of engine controller design or virtual on-board sensing along with available implementation methods and required modifications.

# 2 Literature Review

In-cylinder condition detection and monitoring faces serious limitations due to factors such as implementation complexity, engineering feasibility, reliability and costs as described in the introductory chapter 1. Therefore, the idea of indirect detection has been developed for several years now. In order to avoid restrictions, different approaches have been employed to find relationships between the more accessible engine parameters and the in-cylinder conditions. The relations can be of statistical nature or expressed through a model that describes the physical relationship between the in-cylinder parameter of interest and the initiated or related signal recorded. The techniques used to identify relations will depend on the application of the model as to whether for example, high accuracy and information content is required to study engine behaviour or if a control scenario needs to be designed where fast and mainly reliable results are required. Within this area, several authors have presented their work which is summarised in this review.

Throughout this chapter, distinction is made between the methods "reconstruction" and "prediction". The term reconstruction is used in combination with the analysis and recovery of an initiated or amplified signal by the in-cylinder pressure. This recovered signal is characterised by the fact that its origin lies in the past. A prediction of in-cylinder conditions can however be based on the pre-combustion parameters available for measurement.

The following sections 2.1 and 2.2 present a summary of the applied research for both the indirect temperature and pressure data acquisition.

## 2.1 Literature Review of Indirect Temperature Detection

Direct temperature detection is a difficult and imprecise process as shown in the previous chapter 1.2. The current applications are mainly applicable to laboratories that are inappropriate for application in an industrial environment. Hence, research is focused on determining the in-cylinder heat-release from the combustible mixture. Different approaches have been suggested based, for example, on 1. physical models and 2. statistical relationships identified from heat flux measurements. In recent years, the broad range of engine developments for meeting emissions legislation and fuel consumption limitations have been met by, for example, common rail injection, variable structured fuel injection, exhaust gas recirculation or inter-cooling, and new engine construction designs. However, these developments have outpaced the adaptation of available thermal modelling approaches as stated in the work of Finol et al. [13]. Therefore, it is necessary to find advanced approaches to better describe in-cylinder thermal conditions and achieve potential benefits in terms of:

- Enhanced cooling systems (smaller and lighter pumps or heat exchangers)
- Reduced thermal distortion (lower friction and optimised piston-ring assembly)
- Improvement of computational methods in CFD or FEA

This section creates an overview of the development of in-direct temperature detection and the methods available. In an early stage of investigation into reconstruction methods, Livengood et al. [7] introduced the method of using the speed of sound to determine the in-cylinder temperature. In their approach, the known speed of sound and its changing propagation through a gaseous medium with varying temperatures is used. They focused on the flame front conditions that are most significant for the development of diesel knock which occurs in the event of a sudden temperature increase. A pulse trigger is used to excite sound and its propagation is then measured with a crystal transducer. The time difference over the known distance is caused by varying temperature and, hence, gas conditions. Amongst others, Hickling et al. [14] and Carryer et al. [15] adopted and adapted this approach and introduced pressure transducers into the set-up. This usage is considered a disadvantage for this method because of the limitations of pressure transducers introduced in section 2.2.

In terms of modelling and predicting or simulating the thermal behavior of combustion engines, a wide range of applications has been developed. The different character of models lies in the accuracy and complexity of how closely the relations are investigated. For instance, Huang et al. [16] developed a zero-dimensional model based on the thermodynamic equations of combustion. The combustion process is divided into premixed and diffusive combustion and the formulae for instantaneous combustion and mechanical friction losses are empirical. The aim of the work was to find the optimum heat-release pattern in terms of fuel consumption, maximum combustion pressure and pressure rise rate. Restrictions of these approaches are the assumptions of ideally mixed and uniformly distributed in-cylinder conditions such as gas and temperature. These have an effect on overall accuracy since the spatial heat distribution is important for the detection of knock patches within the flame front which in turn has an impact on combustion performance and efficiency. However, the advantage is the least computational demand in comparison to much more complex models when taking into consideration the spatial distribution of gas and temperature composition.

In the work of Stiesch et al. [17], a phenomenological or quasi-dimensional modelling approach is applied. The goal of the paper is to predict the heat-release and emissions formation during the combustion process of a direct injection diesel engine. The two processes are described using separate models in order to trade the accuracy and validity against the computational demand. The spatial approach enables the determination of the in-cylinder conditions more precisely. With fuel entrainment, additional zones are formed and generate improved resolution of the combustion chamber. Still, the drawback of this model is the uniform pressure assumption over the entire cylinder chamber volume. On the other hand, the computational demand rises and therefore makes real-time application infeasible.

Another quasi-dimensional model is introduced by Hountalas et al. [18] with the introduction of submodels for air entrainment, fuel injection, droplet breakup and combustion and gas exchange. This model is similar to Stiesch's where the fuel entrainment zones of temperature and fuel-air composition are built to include the time history. This method showed considerable errors at the beginning and the end of the combustion where the heat-release was

underestimated and overestimated respectively. This error was explained by the highly heterogeneous temperature and combustion composition. The ideal gas law and assumed uniform in-cylinder pressure do not represent the real situation and thus make it very difficult to estimate the combustion conditions more accurately. However, the overall accumulated accuracy for temperature history and emissions formation were all estimated. This work claims to have identified the error and quantified its impact.

The work of Tamilporai et al. [19] also takes into account the actual heat transfer and applies a two-zone combustion model improved in terms of its swirl characteristic. With this additional heterogeneous information, instantaneous heat flux and the variation of gas velocity can be incorporated into the calculations of heat-release. The goal of the paper is to investigate the effect of conventional and low heat rejection engines. This work shows that with increasing complexity and incorporation of additional information regarding the heterogeneity of in-cylinder conditions, the accuracy of the heat-release can be improved. However, the computational demand and expenditure make the application less favourable than, for example, the online controller design. In addition, the presented models do not accommodate the features of modern engines.

Here, the work of Chmela et al. [20] shows an advanced view of the dependence of heat-release on in-cylinder conditions that are influenced by advanced control strategies. Their simplified model without any spray development, evaporation or mixture formation results in the fact that it is only applicable to high-load diffusion combustion phases because at low loads the premixed combustion phase cannot be described. Nevertheless, the work points out the high proportional relation of heat-release rate to the fuel injection and the kinetic energy. In an adapted version of the model in the work of Lakshminarayanan et. al. [21] the addition of possible wall impingement is made at high speed-high load operation. It undermines the assumption of the previous work and achieves even more accurate results. The model is tested over a range of five different engines varying in type and size. In some cases considerable improvements were made. This shows the modern engine technologies can have an significant effect on the engine performance characteristics and need to be included in the modelling

strategies.

Another aspect of modelling is the inclusion of EGR that is now almost universally applied to diesel engines. Ogawa et al. [22] points out the importance of low temperature combustion achieved through exhaust gas recirculation. Looking at the combustion calculation and emission formation the application of those technologies has a significant effect. Consequently, models complexity has to be revised and increased since increasing numbers of factors influence the in-cylinder thermal development expressed either through the actual flame temperatures or the heat release. With additional complexity the accuracy versus computational demand trade off becomes significant again. Here, solutions in the form of zero or one-dimensional models are favorable in terms of their cost. However, multidimensional models result in higher accuracies over the whole combustion chamber with greater assumptions but require considerable amounts of computational power for the rapid calculations.

## 2.2 Literature Review of In-direct Pressure Detection

The measurement of in-cylinder pressure characteristics is far more routinely and widely applied than temperature measurement. However, reliability and intrusion issues still make the actual measurement unsuitable for mass application. These constraints shifted the focus to different approaches to find a reliable, real-time and less costly method to capture aspects of the in-cylinder pressure characteristics.

The conventional way of modelling and simulating in-cylinder behaviour is based on comprehensive understanding of physical and chemical phenomena. This comes with computational demand and expenditure and is mainly used to investigate combustion and the effects of parameters that influence combustion such as air motion, fuel entrainment, flame propagation. Due to the recent and pressing requirements for on-board diagnostics (OBD) and advanced control strategies, there has been an increasing need for reliable and fast data. Here, two methods are predominant in the research field. Both rely on less costly sensors and try to

capture the in-cylinder characteristic through the effects of combustion as opposed to a direct measurement.

- Cylinder block vibrations initiated through occurring knock and combustion
- Crank angle kinematics recognised through pressure changes during the combustion phase

The following section presents the current research undertaken up to now with regard of in-direct in-cylinder pressure detection either through reconstruction or predictive modelling.

### 2.2.1 Pressure Reconstruction through Cylinder Block Vibrations

In-cylinder characteristics, in particular sudden pressure changes, create structure-borne noise which becomes detectable at the surface using sensors on the cylinder block. This method is established as knock detection, especially in commercial SI-engines as Villarino et al. [23] state in their work. The goal is to find a relationship between the vibration captured at the surface and the actual pressure waveform developed during the combustion phase. This relies on the fact that quick and sudden pressure variations such as those found during the combustion event are transferred through cylinder walls, the cylinder head or the piston. The challenge with this approach is the identification of the pressure-initiated signal within the signal that is contaminated with noise. This is difficult because of two main reasons as concluded by Antoni et al. [24]:

- Piston slap or inertial forces perturbate the signal on the surface.
- Low frequencies carrying the main energy of the pressure trace are not conveyed due to the rigidity of the cylinder block.

In several applications different locations have been tested such as cylinder head surface or cylinder head bolts. The acquired signals have been analysed and used to reconstruct of pressure traces through inverse filtering with neural networks. One of the early investigations was carried out by DeJong et al. [25] in the mid 1980s in which a vibration signal of a heavy-duty diesel engine was measured. They discovered that the deconvolution of the pressure

signal with a frequency response function provided promising results. This approach is called inverse filtering where a source $x(n)$ is convoluted with a response function $h(n)$ into the actual measured vibrations $y(n)$ - see equation 2.1:

$$y(n) = x(n) * h(n) \tag{2.1}$$

However, the significant deviation of the reconstructed pressure trace in comparison to the measured conditions gave rise to the need for further investigation.

The measured vibrations are transformed into the frequency domain using Fourier Transforms in order to detect the frequency bands initiated through the pressure variations. This signal can be inversely deconvoluted with the response function into the the pressure trace. However, the definition of the response function can be challenging because of the mentioned perturbation signals, the engine assembly and time-varying conditions. The source signal is converted into engine block vibrations the moment it impinges on the cylinder walls. The propagation path through the engine block and the engine running conditions such as temperature and speed have a significant impact on the signal measured on the surface of the cylinder block. In addition, separate force sources create and add perturbations to the signal such as the piston slap or valve impacts and are documented in several studies [24, 26, 27].

The combination of time and angle-based events create a challenge for this method. Application to different engine types is required, which in turn imposes different assembly and noise path characteristics. Kim et al. [28] present an approach that takes this structural variability into account. They developed a method using cepstral analysis. This approach suffered significant drawbacks in signal processing where data smoothing affected the overall accuracy when reconstructing the pressure waveform.

El-Gahmry et al. [29] presented studies where cepstral analysis was applied to two engines with extreme size differences - a 102 bhp engine was used in comparison with a 10 000 bhp engine. In both cases the approach delivered results with acceptable and sufficient correlation in relation to the measured pressure signal. This investigation points out the feasibility of cepstral

analysis to overcome structural variability issues. Furthermore, the authors concluded that this method is less reliable in the application of SI engines which generally operate on a lower compression ratio and the pressure signals contribute lower energy to block vibrations than in higher pressure CI engine applications. In addition, it was found that the frequency content is more useful than the energy content of the signal. The recognition of this distinction leads to the capability not being able to reconstruct low-energy signal parts such as during the compression.

Multi-cylinder engines face multiple impacts from valves, pistons and combustion within an engine cycle. Some of these events overlap and the signal analysis gains complexity. An investigation by McCarthy et al. [27] introduces a so-called cepstral comb window. Here, the time domain is used rather than the frequency domain as in the other methods. This is due to the time-based nature of impacts. The cepstral comb window is applied in order to reconstruct signal sections with multiple impacts. In this work, a robust reconstruction approach is created for reciprocating engines with broadband transients. However, they also identified the questionable case where impacts are not equally spaced over time as for example with variable valve timing at different speed and load conditions.

Gao et al. [30] compared three different approaches against each other: 1. time-domain smoothing, 2. direct inverse filtering, 3. cepstral smoothing. The newly developed method of time-domain smoothing is intended to achieve a more robust transfer function by multiplying the signal with an exponential window. The resulting output of the reconstructed pressure trace can be compared to the results of the more complex cepstral method. However, the authors also state that the varying conditions of engines may cause some problems in the reconstruction of the pressure trace.

Another approach is implemented in the work of Antoni et al. [24, 26]. A periodically varying filter is applied that utilises the fact that the vibrational signal is characterised by a non-stationarity which is addressed using a cyclostationary paradigm. The signal is sampled in the crank angle domain, allowing a correlation with engine events and kinematics. This angle-domain-based signal is transferred into the frequency domain where it results in considerably

improved performance than the more conventional approaches of invariant inverse filters. One reason for this is its accommodation of low-frequency components in the signal. These components are described as ill-posed issues of the block vibration approach because the engine block tends to transfer higher frequency components ($>$ 500 Hz) due to rigidity. Another work using the crank-angle domain is presented by Zurita et al. [31] which uses multivariate data analysis (MVDA). In the investigation, this method is compared to the cepstral analysis described earlier. The MVDA consists of two projection methods: 1. Principal Component Analysis, 2. Partial least squares analysis. Both methods are combined in order to enable identification of a relationship between the initiating signal and the response. The results show that this method can be used to successfully reconstruct the in-cylinder pressure in a six-cylinder diesel engine whereas the cepstral analysis failed to reconstruct pressure curves for some of the engine cylinders.

A completely different approach is pursued by Du et al. [32] by using artificial neural networks (ANN). The studies described the implementation of a radial basis function network including k-means clustering for the inputs and a gradient descent algorithm for centre and network training. The results from the developed network show sufficient accuracy. However, processes can be improved through further investigations in the form of additional training data and data pre-processing in order to reduce networks complexity. Another ANN approach is presented by Vulli [33] who uses a Non-Linear Autoregressive with Exogenous Input (NLARX) model structure. Although the previous ANN approach is known for its good generalisation capability in non-linear applications, this approach achieves somewhat better results. In addition to Du the data is pre-processed with Fourier transformation and different training algorithms were investigated: 1. Back-Propagation-Through-Time (BPTT) and 2. Extended Kalman Filter (EKF).

Overall, the methods presented can be used to detect the pressure waveform. However, additional noises from engine parts, overlapping engine events, structural variability and different signal domains raise the question of final achievable accuracy. Another disadvantage of the general methodology of reconstructing the signal from block vibrations is the weak representa-

tion of low-frequency parts due to the engine block characteristics. Parallel to this approach, researchers have tried the other mentioned method of reconstructing the in-cylinder cycle characteristics through the angular speed fluctuations. This particular method promises access to the lower frequencies.

### 2.2.2 Pressure Reconstruction through Angular Speed Fluctuations

The use of angular speed fluctuations in order to monitor combustion conditions has been mainly driven by the aim to detect abnormal combustion as stated in a review by Williams [34]. However, recently researchers tried to reconstruct the whole pressure trace or quantify the generated torque [35]. Measurements are most commonly taken from the flywheel ring gear but are enhanced with recordings taken from the crankshaft nose or driven inertia. This is to achieve a more comprehensive characterisation of the signal. A fluctuating signal is generated through sudden pressure changes that occur at the start of combustion accompanied by changing force amplitudes acting on the piston. With this increase in force amplitudes the crank shaft connected to the piston is accelerated and causes a temporary speed change captured by sensors. As in the previous method, the recorded signal can then be related to the corresponding pressure characteristics causing the output. Different methods for reconstruction have been developed and presented by researchers. Mathematical models, inverse filtering, pattern recognition and, more recently, the application of artificial neural networks have been the focus when it comes to accurate reconstruction.

One of the early approaches used to find the relation between speed fluctuations and pressure waveform was developed by Rizzoni [36]. This model considers the physical impacts of gas pressure forces, friction and pumping losses and reciprocating inertia forces. These formulations are transferred from the mechanical quantities into their electrical analogues and the model is fed with the crankshaft acceleration measurements. As a conclusion of this model it is established that pumping losses and time-varying friction can be negelected. Overall, the result of the model shows a valid description over a wide range of transient engine conditions for time-average and instantaneous torque. In an extension of this work, Rizzoni's [37] model

was refined for a single-cylinder combustion engine. In addition, the inertia, stiffness, damping parameters and the frictional losses of piston rings and bearings of the engine were incorporated into the model. In this work, the transfer of vibrations through the connection of the flywheel and the dynamometer were an additional focus since perturbation signals can be introduced through external sources into the crankshaft. Furthermore, the model was simplified as only discrete frequencies were considered by filtering noise and emphasising the actual vibrations to access the pressure-initiated signal. This model is applicable to multi-cylinder engines.

In the studies of Connolly et al. [38, 39], a model in the form of a continous, linear, first-order differential equation is defined to relate the combustion pressure characteristics to the resulting angular crankshaft speeds. The model was developed by validating it in 'forward direction' where the angular speed is calculated from measured pressures. The model application is performed inversely by deconvolving the pressure from the measured angular crankshaft speed. Through this inverse approach, a distinction can be made between abnormal and normal combustion. The drawback of this modelling tool is the limitation in terms of steady-state operation.

Another approach was developed by Moro et al. [40] who defined different patterns in the crankshaft speed characteristic in relation to pressure combustion conditions. Each pattern is related to a frequency response function. These functions are stored in a map in order to determine the signal transfer characteristics through the engine structures. The map then enables the derivation of the pressure waveform. This less expensive method of reconstruction is an approach for on-board diagnostics if applied on a map basis. However, depending on the required accuracy vast amounts of data need to be generated to cover different conditions and different fault characteristics.

In the work of Lee et al. [41], a pattern recognition method has been applied in comparison with a frequency analysis. The pattern recognition method employs a stochastic analysis in which the physical complexity of the engine is ignored. This was achieved by building a relationship between combustion and crankshaft in the form of polynomial expression, enabling

a fast and non-costly calculation of pressure traces. In contrast, they described a frequency analysis approach where the advantage lies in transferring the signal into the frequency domain and being able to define frequency components relevant to the measured signal. Consequently, the computation time was shortened because the calculation expenditure required fewer components. Both approaches were capable of being applied on a low sampling resolution and met the criteria for real-time online estimation and control.

In the three reported works of Taraza et al. [42, 43, 44], the issue of non-rigidity of the crankshaft is discussed. The first investigation [42] establishes the importance of harmonic orders for representing the in-cylinder parameters of gas pressure and torque. This approach enables the distinction of the orders initiated in the cylinder chamber and orders generated through non-rigidity and resonance oscillations. Different harmonics are caused by varying engine phenomena such as combustion pressure, faulty cylinder conditions or power imbalance. Hence, patterns of harmonics can be used to isolate certain engine problems. In [42, 44], the authors also state that lower frequency components measured in the crankshaft fluctuation contain information about the indicated gas pressure. For these frequency regions in particular, the crankshaft can be assumed to be a rigid body that does not introduce resonance frequencies. In addition, specific harmonic orders can be used to determine varying power contributions of individual cylinders. One disadvantage of this method is the limitation in terms of steady-state conditions. In the case of additional engine operation points there would be a requirement for far more data in the form of harmonic patterns for each condition.

Zeng et al. [45] used this information about harmonics and created a speed-load curve with fitting factors for a polynomial relation under defined conditions. This was achieved by measuring in-cylinder pressure on tested points and capturing the corresponding harmonics on the crankshaft. With the corresponding map the in-cylinder pressure can be reconstructed with a trade-off in accuracy of conditions between defined points where interpolation is applied.

Another approach of reconstruction is the application of ANN. The method radial basis function already mentioned in 2.2.1 was proposed by Jacob et al. [46] and eventually applied by

Gu et al. [47]. The non-linear relation and potentially necessary application to a wide range of engine conditions makes ANN with its generalisation capabilities and non-linear mapping characteristic a promising choice. The results showed that the RBF operated in a consistent and robust manner over a wide range of engine operations.
A similar discovery was made by Potenza et al. [48] who applied the Non-Linear Autoregressive model with exogenous input (NLARX) as mentioned earlier in 2.2.1. They applied two different training algorithms, wheras the Extended Kalman Filter was found to be more efficient and resulted in better network performance than the Back-Propagation-Through-Time algorithm. In these applications, problems were identified in the complex structure and computationally expensive structures of the network. Accuracy is thought to be improved by additional training data with higher feature density recorded over a wider engine operation range. At the same time, any additional inputs and data features model structures will grow in complexity, which will consequently increase computational costs.

To summarise the method of reconstruction of cylinder pressure through crankshaft velocity fluctuations, it appears there are several reconstruction approaches that lead to promising results in terms of:

- Abnormal combustion detection
- Faulty cylinder detection
- Torque reconstruction
- Gas pressure reconstruction.

However, there are challenges in recovering the information required. Assumptions in models and relations or limitations to certain engine conditions or operating ranges take their toll on accuracy. In the event of more comprehensive models or maps for engine conditions, the computational expenditure increases. This in turn disqualifies the algorithm for on-board diagnostics or controller design purposes. An additional problem is the measurement location which is ill-posed for several possible signal perturbations. Engine phenomena such as overlapping cylinder excitations introduce noise into the measurement or vibrations introduced from the road and transferred through the clutch generated disturbances. In addition, the

crankshaft is considered a non-rigid body. On the one hand, this is a reasonable assumption for lower frequencies introduced through cylinder pressure that can pass through the engine components and then be detected. On the other hand, higher frequencies cause resonance problems. A third problem arises in the form of the singularity phenomena that appears while the cylinder moves through the Top-Death-Centre (TDC) and Bottom-Dead-Centre (BDC) points. At these points torque is instantaneously zero because of the change of direction. As a consequence, there is no pressure information available at these moments via the crankshaft. This effect becomes crucial since the pressure propagation process is at its peak around TDC and the instantaneous information cannot be detected.

Taking into consideration the problem of low frequency blocking in the cylinder block vibration method and the ill-posed high frequency part in the crankshaft speed fluctuation method Johnsson [49] developed a method combining those signals.

### 2.2.3 Combination of Block Vibration Signal and Crankshaft Speed Signal

The reconstruction of the pressure trace through vibrational analysis on the cylinder block or fluctuation analysis on the crankshaft speed suffer from frequency range restrictions. This is either caused by the fact that the transferring medium is too rigid (engine block) or too flexible (crankshaft). Hence, the engine block does not transfer the low frequencies of the pressure trace and the crankshaft has a resonant behaviour at high frequencies. When both features are combined, they theoretically enable a higher accuracy and more comprehensive system representation. Johnsson [49] presents a complex valued RBF network that is fed with both measured signals. The signals are pre-processed with Fourier transformation in order to access the necessary frequency bands. For training purposes k-means clustering is applied along with a recursive hybrid learning algorithm in order to optimise the networks performance. The results are accurate but would require more training data to improve performance. This however would lead to bigger networks and more computational expense that goes against the needs of real-time applications.

### 2.2.4 Prediction through Comprehensive Models

The predictive character of a model theoretically enables the result to be taken into account for instantaneous control actions. The predictive detection of in-cylinder pressure has to be based on parameters that are available prior to the combustion event and have an effect on the outcome of the combustion rather than reconstructing them from signals available after the event. Different models have been developed in recent years. The pressure predictive models are mainly integrated into fully comprehensive engine models used for engine calibration and investigation of different engine processes. They are handled as a sub-model with varying detail and complexity. In the case of comprehensive implementations, the computational demand may be very high which disqualifies those methods from any real-time operation in OBD or control. However, models have been developed with good prediction accuracies depending on the model type. Their complexity can be classified into: 1. zero-dimensional, 2.quasi-dimensional and 3. multi-dimensional approaches. These are determined via the degree of detail in which the in-cylinder process is analysed. Zero-dimensional models usually assume ideally mixed states of in-cylinder behaviour. Quasi-dimensional models use the notion of different zones of gases such as unburned, burned or flame zone. The multi-dimensional approach is most comprehensive and utilises a mesh of separate zones that describes the cylinder. The latter is usually applied in computational fluid dynamics (CFD) models and can give precise predictions for each step of combustion for different zones in the combustion chamber.

The studies of Sing et al. [50, 51] generated a comparison of three different approaches to diesel engine combustion modelling. Their comparison parameters are in-cylinder pressure, $\mathrm{NO_x}$ and simultaneous optical diagnostic images from a heavy-duty diesel engine. A characteristic time combustion model, a representative interactive flamelet model and a direct integration using detailed chemistry are implemented with a KIVA code. Each model is tested on five different engine operation points. The predictions of all three models show a reasonable trend for cylinder pressure and other comparison parameters. However, the actual in-cylinder phenomena were considerably different between each of the models. The cost of these results with CFD codes comes with the computational expenditure which is significant and renders them not applicable for OBD or controller design purposes.

A less detailed model is presented by Pariotis et al. [52]. This quasi-dimensional model uses a multi-zone phenomenological approach with a procedure normally used in CFD models. The combustion process is described in more fundamental terms than a plain quasi-dimensional approach but is less computationally demanding than in a CFD model. It also takes into consideration the spatial distribution of air-fuel mixing and temperature and gas mixture concentrations. In the paper, three different load and speed points are presented and the models are correlated with a turbocharged diesel engine. The results for pressure prediction are described as sufficient. In most engine models, the pressure trace correlation is a measure of the models' quality rather a focus on the models' development. Models implemented simply to predict of the pressure are rare and mainly developed for low-cost simulations and to develop control algorithms in the lab.

Zero-dimensional models require the least computational performance. Therefore, models with generalisation of complex gas mixtures, fuel-entrainment systems or temperature distribution are preferred for real-time simulations. The model presented by Grondin et al. [53] focused on the pressure and torque generation of a single cylinder CI engine. The model is described by physical phenomena based on the filling and emptying method and the ideal gas law is applied as stated in equation 2.2:

$$PV = nRT. \tag{2.2}$$

Here $P$ represents the absolute pressure, $V$ the volume, $n$ the number of moles of the gas, $R$ the absolute gas constant, and $T$ the absolute temperature. The model is described as favourable for hardware-in-the-loop (HIL) applications. The simplification generates a fast, reliable and, over a reasonable engine operation range, accurate model. However, model validation for transient application still needs to be verified.

Allmendinger et al. [54] developed a model based on energy balance equations. However, their approach was to split the calculations of an engine cycle into different phases in order to

reduce the number of parameters actually required and, consequently, calculation time. Their model can calculate the in-cylinder pressure, temperatures during compression, expansion and the heat-release phase. Their model shows some deviations from measured results which are due to incorrect initial conditions and neglecting the cylinder wall heat transfer as stated in the conclusions. Nevertheless, it is claimed their model is reasonable and reliable for control purposes.

This approach is utilised by Erikssons et al. [55] who instead developed a model for a SI engine. The Otto cycle enables the authors to solely consider the compression and expansion processes that need to be modelled. The gap in-between is interpolated using a Vibe function. Instead of describing the two processes numerically, an analytical approach is applied. The aim here is to develop a simple, reliable and accurate model.

Another important factor in this combustion model is described in Chen's work [56]. The model is developed for a CI-engine application and is used to investigate the impact of variation of inlet conditions on the peak cylinder. It correctly predicts the in-cylinder pressure with varying intake manifold temperatures and pressure. For peak pressure composition, this simplified model uses three parameters: 1. intake manifold pressure, 2. pressure rise from compression to expansion with combustion (motoring pressure rise), 3. pressure rise added through the combustion heat added to the cylinder. The model closely agrees with trends and magnitudes of the peak cylinder pressure. The drawback of this method is that the prediction is restricted to peak cylinder pressure rather than the whole pressure trace over the engine cycle.

Zweiri et al. [57] describe an analytical, non-linear dynamic model for a single-cylinder diesel engine. It is validated using the cylinder pressure and instantaneous engine speed under transient operating conditions. The model can describe the dynamic behaviour of varying fuelling strategies and the resulting engine speed. For those operations it includes dynamometer dynamics, friction parameters and cylinder thermodynamics. The results are found to be in good agreement with the previously mentioned validation parameters. The authors state that the model can be used as a investigation tool for transient fuel control design and fault diagnostics as a model-based estimator but also for OBD and control operation.

This overview shows that predictive models are mainly developed for investigation and understanding of engine behaviour. Although the need for non-linear models as on-board diagnostic tools or for hardware-in-the-loop controller design has become greater, current models are still not sophisticated enough to cover the entire engine operation range. The application of ANN is promising since the recognised generalisation capabilities can overcome the demands of transient engine conditions. At the same time, efficient and lean network design can achieve fast and real-time-capable models applicable to on-board diagnostics.

## 2.3 Summary Literature Review

- The in-cylinder temperature estimation is achieved using heat release models. These can be distinguished by the number and types of parameters and the spatial resolution of CFD or FEA simulations.
- The in-cylinder pressure simulation is classified into two main parts. The reconstruction utilises signals that are excited by the pressure characteristics inside the cylinder, whereas the prediction determines the pressure signal from relevant engine parameters measured prior to the combustion event.
- The reconstruction is segmented into the approaches of engine block vibration and angular crank-shaft speed fluctuation reconstruction. In both approaches, several techniques have been employed, such as inverse filtering, mapping, mathematical, numerical or neural network modelling.
- Prediction can be categorised into zero-dimensional, quasi-dimensional, multi-dimensional and black-box modelling. The first three modelling approaches can be distinguished from one another by the considered parameters and their treatment. In contrast, black-box models in the form of neural networks, simply neglect the complexity of the observed system.

## 2.4 Advantages of Simulation Approaches

- Heat-release models permit further understanding of combustion processes and enable the investigation of varying engine behaviour with changing parameters, such as air entrainment or injection rate and timing.
- The reconstruction approaches, such as engine block vibration and angular speed fluctuation measurements, can be readily acquired because the required sensor hardware has already often been installed. In addition, the signal recovery techniques can be based on data processing that cuts down the required understanding of engine processes in terms of their complexity. Consequently, the techniques can be considered for a real-time on-board application purpose.
- Modelling approaches to predict in-cylinder pressures increase the understanding of engine combustion processes and achieve high accuracies in terms of mapping the outputs.
- Artificial neural networks achieve remarkable results for both parts, reconstruction and prediction. Their adaptiveness and ability to accommodate the process and systems complexity push them into a promising position.

## 2.5 Disadvantages of Simulation Approaches

- Indirect temperature measurement is characterised by elaborate models that require substantial computational resources.
- Strong noise perturbations from other engine parts, high- and low-frequency constraints in both approaches: 1. cylinder block vibrations 2. crank fluctuations, and model simplifications in the reconstruction techniques lead to an overall restriction of both methods.
- The prediction is denoted by the trade-off between accuracy and computational demand. Increasing utilisation of submodels aids accuracy but simultaneously increases the required computing effort.

- The approaches identified from the literature are not appropriate when it comes to on-board emissions control.

# 3 Theory of Artificial Neural Networks - Structures and Optimisation

The theory of Artificial Neural Networks (ANN) has been in development over the last few decades. Its origins are based in the field of neuroscience from where it spread into other sciences such as engineering, but also economics for stock market prediction [58].

The idea of ANN is laid out in the paradigm of the human nervous system. Therefore, it resulted from an interdisciplinary work between mathematicians and neurophysiologists who determined that the nervous system can be mapped by logical procedures represented in maths. Here, the breakthrough was achieved by McCulloch and Pitts [59] in the 1940s. It was shown that a simple unit in the form of a neuron composed within a sufficient network of neurons could map any computable function. However, after two decades of pursuing this new approach, the level of interest slowed down due to lack of computing power. The increasingly required computational accuracy of learning algorithms for the rising complexity of problems could not be met. Especially within the engineering domain this field was put on hold until the end of the 1980s before becoming increasingly popular again in the 1990s as a result of the cheap and sufficient computing power. The ability to solve complex problems on desk machines made the theory highly applicable for everyday tasks. Since then, the method of ANN has been utilised in system identification tasks for creating models or controllers in a number of different fields as the following studies suggest [59, 60, 61, 62].

The following chapter outlines the theory of artificial neural networks. The first section describes the formulation of a single processing unit. In the second section, architectures of comprehensive parallel processing compounds are presented. The third section covers the optimisation of a system-representative network through specific training algorithms. A fourth

section illustrates compilations of ANN structures and their application. The final section summarises the chapter.

## 3.1 Artificial Neural Network Principles

The idea behind ANN is based on the human nervous system found in the brain: the neural network. This composition of neurones can be described as a magnificent parallel distributed processing unit. The human brain's natural ability to store knowledge and use it to map complex input-output relations makes it unique. Hence, the development of artificial counterparts for simple and narrower field application can be a powerful tool. Haykin [59] defines a neural network as follows:

> *A neural network is a massively parallel distributed processor made up of simple processing units, which has a natural propensity for storing experiential knowledge and making it available for use. It resembles the brain in two respects:*
>
> 1. *Knowledge is acquired by the network from its environment through a learning process.*
> 2. *Inter-neuron connection strengths, known as synaptic weights, are used to store the acquired knowledge.*

From this definition it follows that an ANN, similar to its natural paradigm, learns from stimulating inputs and a desired output it can relate to. It is taught by the existing input-output relation and, based on the teaching, it generates a mapping function that enables it to predict the output based on new, unseen inputs. The corresponding relation of this input and output is hidden within the structure of the ANN. There is no information given by the network about the inner behaviour of the actual relation it maps between input and output. Therefore it can be counted into the field of black-box modelling approaches. This approach is especially desirable for not fully understood physical relations where definitions of phenomena cannot be described through mathematical formulation of the problem. In addition, in cases of extreme complex numerical relations requiring vast computing power, neglecting this information by using ANN can reduce computing time considerably.

The performance outcome of an ANN is dependent on different parts the user can choose and train. The core unit of such a network is a so-called neuron as presented in figure 3.1.

### 3.1.1 The Neuron and its Peripherals

The neuron is the basic processing unit that, in connection with other neurons forms a network. This unit consists of several parts as they are also displayed in figure 3.1:

- Input signals - $x_j$
- Connecting links with assigned synaptic weights - $w_{kj}$
- Bias input - $b_k$
- Summing junction - $\sum_{j=1}^{n}$
- Activation function $\varphi(\cdot)$
- Output signal $\hat{y}_k$.

Each of these parts therefore play a role in finding a network that performs sufficiently on a selected problem, i.e. modelling task.

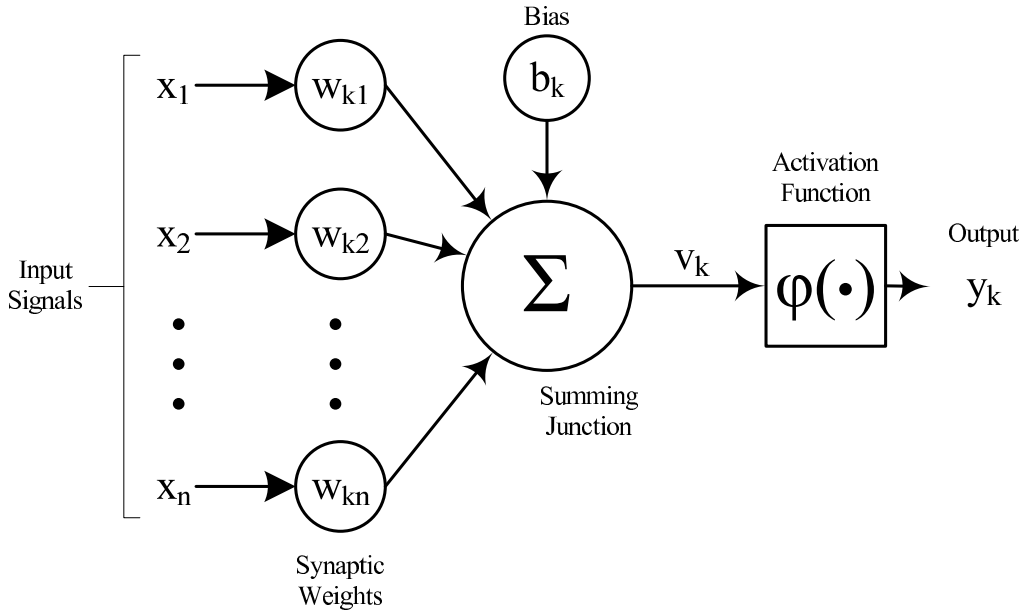


***Figure 3.1:*** *Single neuron scheme with input, weight, bias, summing junction, activation function and output*

**Inputs and Outputs** The inputs $x_j$ and outputs $\hat{y}_k$ are key parameters in order to successfully train an ANN. The information content incorporated in the inputs of the system is crucial in order to find an operational point with sufficient performance levels. The inputs provide stimulating information as to what causes a reaction within the system resulting in an output. Therefore it is important that the scope of information covers the maximum required details. This enables the trained ANN to perform more accurately on new and unseen data known as the generalisation capability. Common pre-processing procedures are the normalisation of inputs and outputs in order to reduce the input value range and avoid saturation of the activation functions. The output range of many functions lies between [0, 1] or [-1, 1]. Hence, the system output value is also required to be within the target values of the activation function of the output layer to avoid saturation of output values. In case of multiple inputs it is recommended to choose uncorrelated information with covariances that are approximately equal. A popular method of finding uncorrelated inputs is the principal-component analysis.

**Connecting links** The connecting links between inputs and neurons and the inter connections between neurons are the ANN memory. Links are assigned with weighting values according to the importance of the corresponding connection. This synaptic weight $w_{kj}$ is multiplied by the input value from either the input $x_j$ or the output $\hat{y}_k$ of a predecessor neuron. Their value is changed during training until a specified accuracy level with respect to the reference system output is found. In the final training state, the value for each weight describes the importance of each input to the neurons' output. Initialisation of the weights is important to ensure training process efficiency. Initial states can lead to saturation of the activation function. Consequently, this may cause a longer and possibly unsuccessful training process that results in not finding the weight allocation. Haykin [59] states that if the values are chosen from the uniform distribution they should have a mean of zero and a variance that is equal to the reciprocal of the number of synaptic connections into a neuron as expressed through 3.1:

$$\sigma_w = m^{-\frac{1}{2}}. \tag{3.1}$$

An additional value for tuning is the bias input that can be chosen as a perturbation signal

into the summing junction.

**Summing junction** The summing junction is the combining element of the neuron $k$ where the different connecting links either from the input or from a predecessor neuron are combined into the activation potential $v_k$. This is represented through the mathematical equations 3.2 and 3.3:

$$u_k = \sum_{j=1}^{n} w_{kj} x_j \tag{3.2}$$

and

$$v_k = u_k + b_k. \tag{3.3}$$

The neuron inputs $x_1, x_2, ..., x_n$ are multiplied with the correlating input weight $w_{k1}, w_{k2}, ..., w_{kn}$ and then summed up into the resultant signal $u_k$ shown in 3.2. In the follow-up equation 3.3 the activation potential $v_k$ of neuron $k$ is formed by the sum of $u_k$ and $b_k$.

**Activation or Transfer Function** The activation function or sometimes also called transfer function acts as an amplitude limiter and maps the activation potential $v_k$ into a finite value range of output $\hat{y}_k$. Typically these ranges are between [0, 1] or [-1, 1]. The transformation can be mathematically expressed through the formulation of 3.4:

$$\hat{y}_k = \varphi(v_k). \tag{3.4}$$

This varies from the type of activation function which is described into more detail in the following section 3.1.2.

### 3.1.2 Types of Activation Functions

The activation function determines the relationship between the input and output by mapping the activation potential $v_k$ through a functional relation $\varphi(\,\cdot\,)$ into the output $\hat{y}_k$. Within the field of ANN, different functions are preferably used:

- Threshold functions
- Piecewise linear functions
- Sigmoidal functions
- Pure linear functions

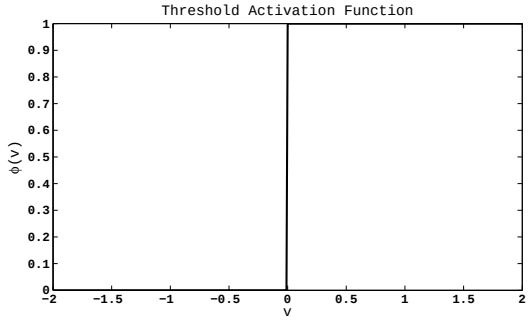


***Figure 3.2:*** *Typical threshold activation function*

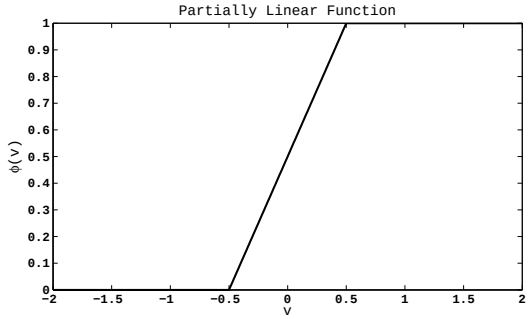


***Figure 3.3:*** *Typical piecewise linear activation function*

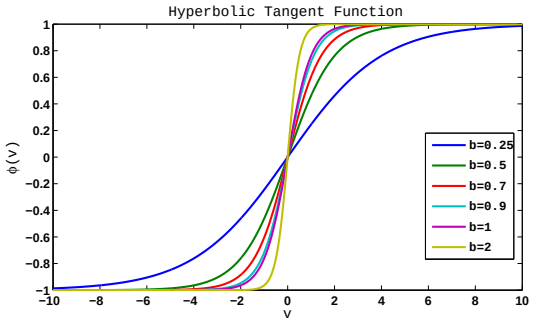


***Figure 3.4:*** *Typical hyperbolic tangent activation function*

Their choice is dependent on the problem of the system that needs to be mapped, whether the system is classifying, is linear, or shows some non-linear characteristics. In figure 3.4, three of the main functions are represented. The threshold function is typically used for classification problems whereas the piecewise linear function can be applied to linear problems. The pure

linear function represents a special case of the piecewise linear functions. The most commonly used function especially within non-linear problems is the sigmoidal function, which is based on the logistic function described by 3.5:

$$\varphi(v_k) = \frac{1}{1 + exp(-a \cdot v_k)}, \tag{3.5}$$

where the activation potential is represented through $v_k$ and $a$ which is describing the slope of the function. If this value becomes infinitely positive, the function represents a simple threshold. An infinitely negative value of $a$ represents a pure linear function. Another commonly used function from the group of the sigmoidal functions is the hyberbolic tangent function shown in 3.6:

$$\varphi(v_k) = b \cdot tanh(a \cdot v_k), \tag{3.6}$$

where $a$ defines the slope and $b$ determines the output range of the function. A positive or negative infinite value leads to a threshold or linear characteristic respectively. The actual output form of an hyperbolic tangent activation function can be described as being anti-symmetric since a negative value of $v_k$ is mapped into a negative value of $\varphi(\cdot)$ as figure 3.4 shows. The other output form is a non-symmetric form that is represented through the logistic function where the input range, e.g. [-10, 10], is mapped into the function space [0, 1].

In general, every function that is continuously differentiable could be a possible candidate for an activation function. In theory, this enables specific custom functions to be defined for individual problems. Nevertheless, current practice shows that for most ANN applications the presented function types are used [62].

## 3.2 Neural Network Architectures

A composition of neurons and connecting links into networks is considered to be an artificial neural network (ANN). Compositions of those networks have been evolved during the last few decades to create more and more complex architectures of combinations. Different architectures fit different tasks such as: pattern association and classification, function approximation

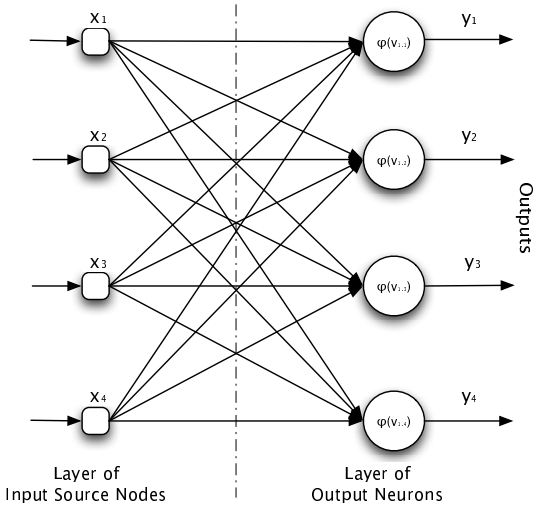


***Figure 3.5:*** *Single-layer feedforward network*

finding associative memories or generate new meaningful patterns. Within the field of ANN, a distinction is generally made between three main architectures:

- Single-layer feedforward networks
- Multi-layer feedforward networks or Multi-Layer Perceptrons (MLP's)
- Recurrent neural networks (RNN)

### 3.2.1 Single-Layer Feedforward Networks

The architecture of a single-layer feedforward network represents the basic structure of neuron composition as illustrated in figure 3.5. The alignment of neurons is organised in layers. The initial layer contains source nodes that connect the inputs towards a layer of neurons generating the network output. Literature does not consider the inputs as an independent layer with computation of data, so here it is simply called the input layer but is not considered during layer numbering.

Figure 3.5 shows a single-layer network connecting the input layer with four source nodes to an output layer of four neurons. The processing direction is strictly forward which results in the name of single-layer feedforward network. Single-layer structures can accommodate single input single output (SISO), which are basically a neuron, multiple input - single output (MISO)

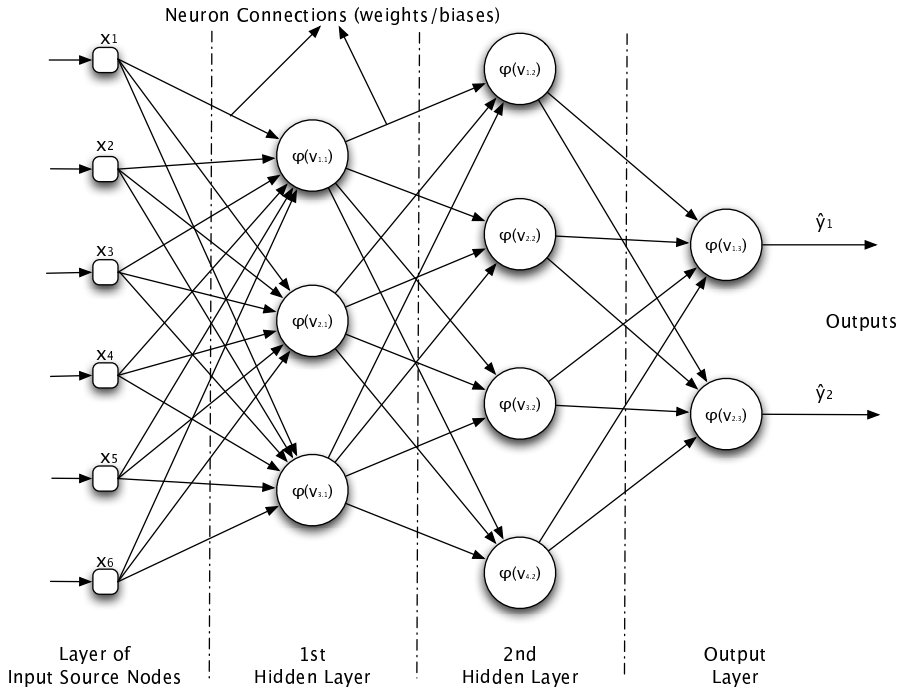


***Figure 3.6:*** *Multi-layer feedforward network or Multi-Layer Perceptrons (MLP's)*

or multiple input - multiple output (MIMO) networks. They are used for pattern classification or simple function approximations.

### 3.2.2 Multi-Layer Feedforward Networks

The extension of the basic network version contains additional layers. Figure 3.6 shows a network with an input layer, two hidden layers and one output layer. The introduction of hidden neurons between the input and output increases the networks' capabilities of mapping higher-order statistics and introduces a global perspective that increases modelling power. In general, it is stated that most practical neural networks have just two to three hidden layers [62]. Such limitation ensures a complexity within manageable boundaries and computational expenditure that is both reasonable and acceptable for training procedures and operation.

The network in figure 3.6 can be expressed by the abbreviation $6-3-4-2$ which means it consists of 6 source nodes, 3 hidden neurons in layer one, 4 hidden neurons in layer two, and 2 output neurons. The general term for a network with $j$ inputs, $h_1$ hidden neurons in layer one, $h_2$ hidden neurons in layer two and $y$ outputs is consequently $j-h_1-h_2-y$. Figure 3.6 represents a fully connected network where all the nodes of each layer are connected to all of

the nodes in the subsequent layer. In case of missing connections a network is called partially connected. A popular name for the multi-layered networks is also the expression MLP which stands for multi-layer perceptrons.

### 3.2.3 Multi-Layer Feedforward Networks with Temporal Behaviour

Conventional feedforward networks are limited to representing dynamical characteristics of an input-output relation. Their standard implementation does provide for the inclusion of any time-related information. This can be solved by a short-time memory represented through a tapped delay line storing preceding inputs which generates a temporal dimension for the networks performance [59, 62]. The output $y(n)$ includes some temporal information from delayed inputs for a single hidden layer of the size $m$ and can be described by 3.7:

$$\hat{y}(n) = \sum_{j=1}^{m} w_j y_j(n) = \sum_{j=1}^{m} w_j \varphi \left( \sum_{l=0}^{p} w_j(l) x(n-l) + b_j \right) + b_0 \tag{3.7}$$

where $w_j$ defines the weights of the output neuron, $x(n-l)$ the delayed input, $b_j$ stands for the neuron bias, and $b_0$ for the input bias if applicable.

### 3.2.4 Recurrent Neural Networks

Recurrent neural networks (RNN) are defined by a feedback loop. This loop can be either between output and input layer, between hidden layers, or even between the neuron's output and input. This structure has a profound learning advantage. Its feedback enables the implementation of time dimension and, hence, system dynamics. The delaying capability of inputs and outputs defines a long-term memory.

Figure 3.7 displays an example of an RNN where two outputs are fed back to the input layer. The introduced connection branches require a unit-delay element that is denoted by $z^{-1}$.

A further detailed distinction can be drawn between three different types:

- Input-Output Recurrent Model
- State-Space Model

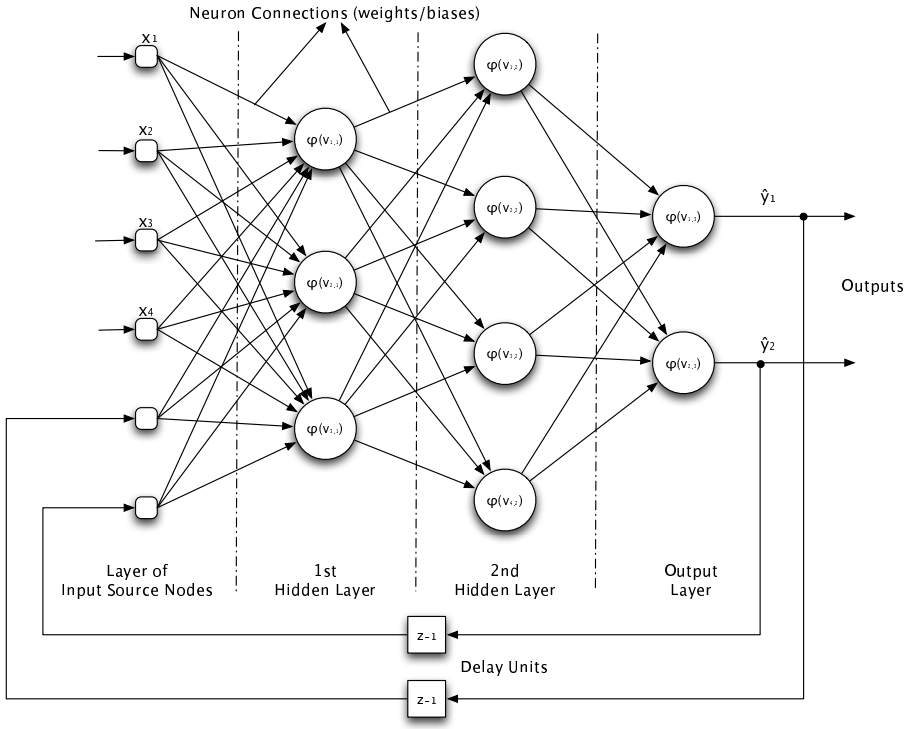


***Figure 3.7:*** *Recurrent neural network (RNN)*

- Recurrent Multilayer Perceptron

**Input-Output Recurrent Model** - This model is derived from an MLP. The inputs are delayed as in an MLP with temporal behaviour. In addition, the output is fed back into the input layer with delay units. Due to the exogenous inputs $x(n)$ and its predecessors $x(n-1), ..., x(n-q+1)$ on the one hand and the output $\hat{y}(n+1)$ that is regressed in terms of its previous values $\hat{y}(n), \hat{y}(n-1), ..., \hat{y}(n-q+1)$ the network is called Non-Linear Autoregressive with Exogenous Input Model - NARX or NLARX. Figure 3.8 shows a general canonical model of the structure.

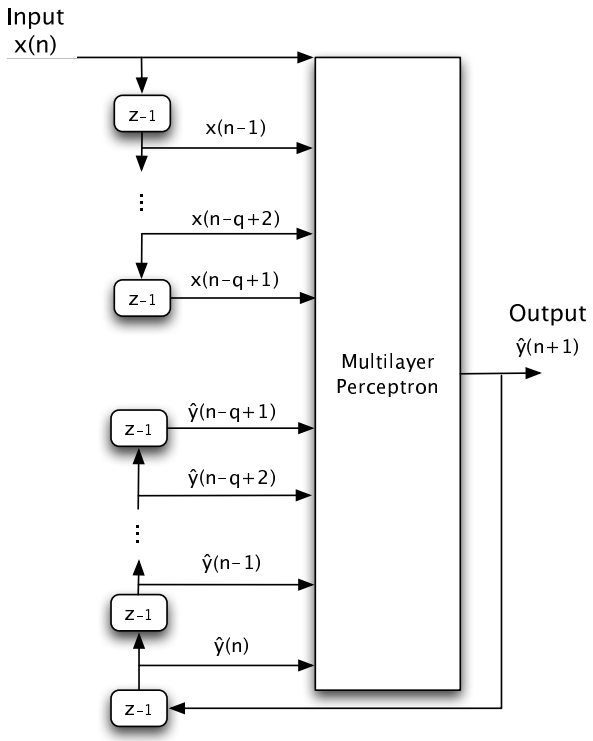


***Figure 3.8:*** *Nonlinear Autoregressive with Exogenous Inputs Model - NARX or NLARX: canonical representation (partially redrawn from [59])*

This network structure can grow considerably depending on the number of recurrent outputs and delayed inputs. Hence, network performance is a trade-off between the computational expenditure and the required dynamics.

**State-Space Model** - The state-space model (SSM) differs from the NLARX model on account of its state-based feedback. The states of the SSM are defined through the outputs of hidden neurons. These states are looped back to the input layer. Figure 3.9 visualises the assignment of the state feedback from the hidden layers. Its structure can be expressed by the following equations 3.8 and 3.9:

$$\mathbf{x}(n+1) = \mathbf{f}(\mathbf{x}(n), \mathbf{u}(n)) \tag{3.8}$$

$$\mathbf{y}(n) = \mathbf{C}\mathbf{x}(n) \tag{3.9}$$

where $\mathbf{x}(n+1)$ is the hidden layer output generated through a non-linear function, $\mathbf{f}$, which is dependent on $\mathbf{x}(n), \mathbf{u}(n)$, the hidden layer output previously fed back and the environmental input respectively. The output layer transfer function is of linear character which leaves the output $y(n)$ in equation 3.9 in a simple multiplication of the output neuron weight matrix $\mathbf{C}$ and the current hidden layer output $\mathbf{x}(n)$ - see figure 3.9.

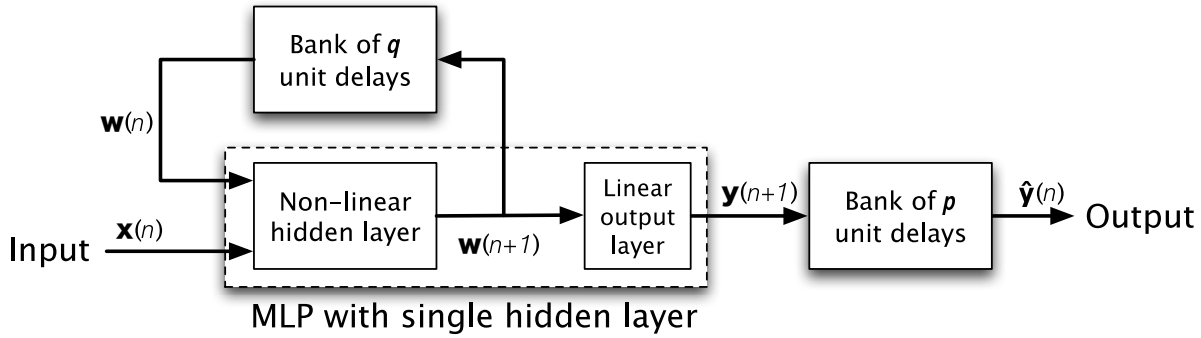


***Figure 3.9:*** *State space model with MLP and a single hidden layer (partially redrawn from [59])*

This operation of feedback enables information of previous network states to be stored which may influence the forthcoming process and hence take dynamical behaviour into account.

In addition, recurrent multi-layer perceptrons (RNN) and second-order networks can be mentioned as an extension of the state-space model. The former is characterised by a local feedback around each hidden layer which permits free choice of the transfer function of each layer. The latter shows a multiplication of the nodes from external and feedback inputs with weight $w_{kij}$ resulting in an activation potential $v_k$:

$$v_k = \sum_i \sum_j w_{kij} x_i u_j \tag{3.10}$$

with $x_i$ as the external input and $u_j$ for the feedback input.

### 3.2.5 Combining Artificial Neural Networks

The growing complexity of single ANN becomes bigger, more difficult to train, and may lose its computational efficiency. Hence, in some cases the development of network compounds

becomes inevitable. Network compounds as presented in Maass et al. [63] can improve the overall task solutions considerably. A compound may include the arrangement of several neural networks which can be allocated different tasks or even the same ones in case of required redundancy [64]. This method enables ANN capabilities to be broadened by building network ensembles or modular combinations. The former is defined as a system of redundant networks, which ensures a definite result in case one of the networks fails to perform. In case of a modular combination, individual networks are designed to perform a superior task and contribute to a solution. A third way is the combination of ensembles and modular combinations. The definition of those combinations is either made as a decomposition of a task from top to bottom or bottom-up for sensor fusion. There are different methods of combining ANN modules which are presented in figure 3.10.

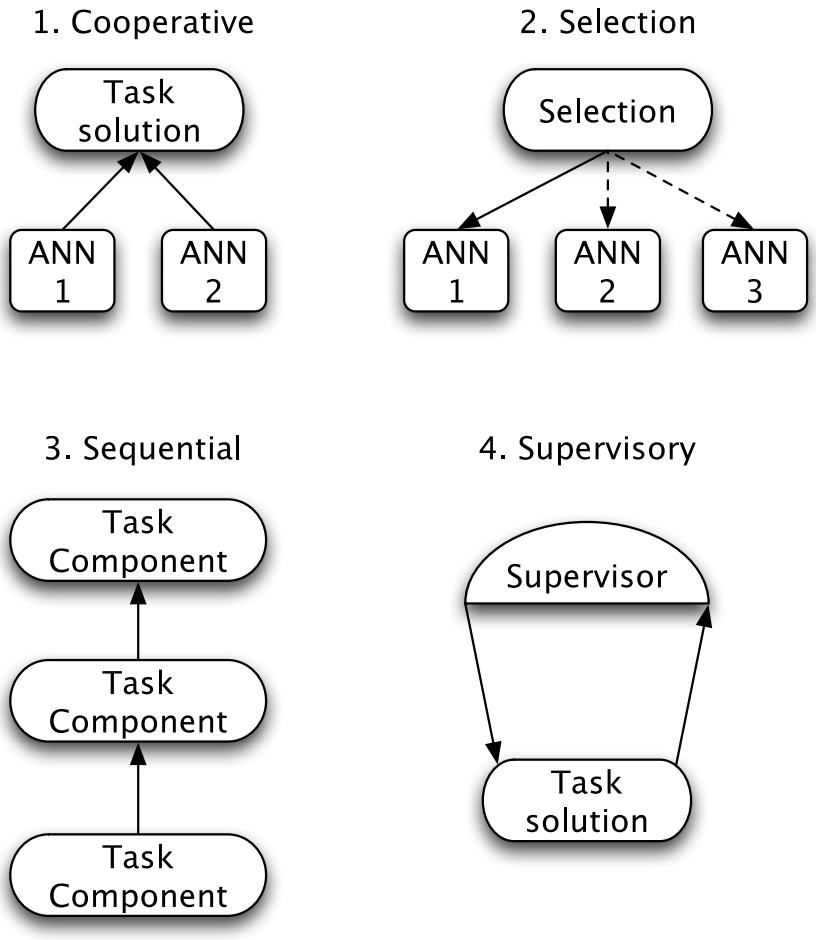


***Figure 3.10:*** *Different methods of combining ANN into multi- network systems*

These combinations can be used for a variety of purposes in order to find system behaviour solutions:

- Modularity for optional exchange
- Voting capabilities
- Several non-similar outputs
- Signal variability

Here, the modularity for optional exchange describes the fusion of a variety of signals within an overall solution. A subsystem may provide an output to another subsystem that combines other inputs to a system's output. An example of such a scheme is represented in figure 3.11. In the literature it is also known as the modularity-based approach as described by Sharkey et al. [64].

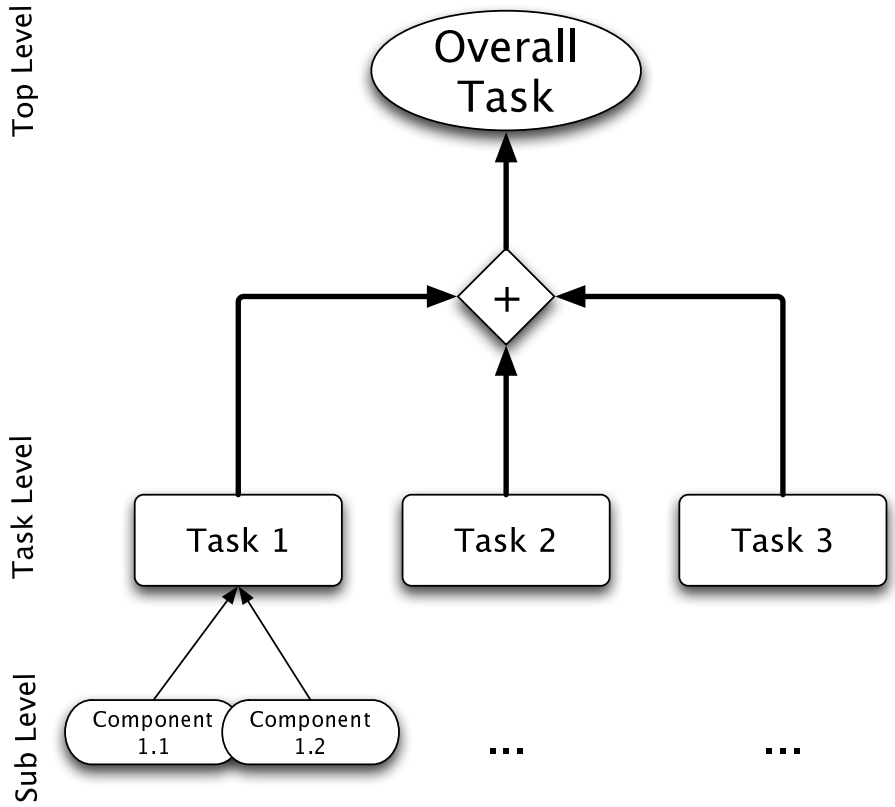


***Figure 3.11:*** *Scheme of network modularity: task distribution for finding overall task solutions*

A network structure with voting capabilities is also described as an ensemble [64]. Several networks map the same output redundantly. An overlooking system selects the strongest "vote", i.e. the best result from all of the different subsystems. These subsystems can have different inputs or be trained differently - figure 3.12.

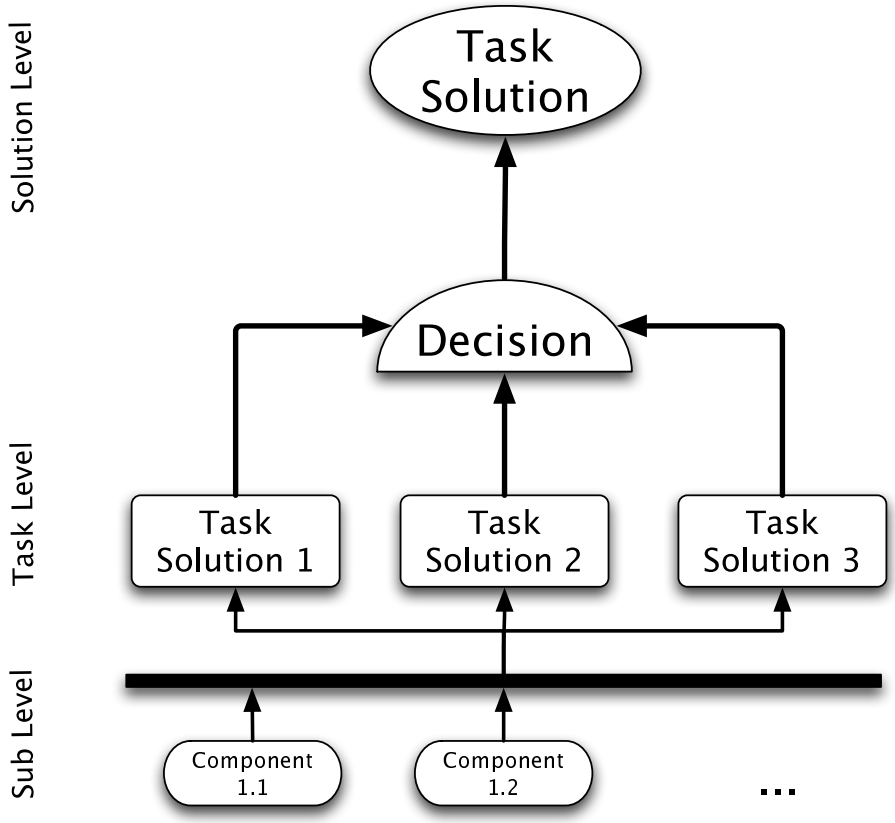


***Figure 3.12:*** *Scheme of network output voting: Several redundant networks predict the same output and a vote for the best or the majority result is applied*

The third bullet of non similar outputs is represented in section 4.3. An engine model contains four independent ANN for four outputs. A single ANN for all four outputs cannot generate the output performance of several independent networks. Hence, a modular compound is required in order to find the tasks' solution.

Another possible use for network compounds is signal variability. This type of network compound is especially helpful for wide operation ranges such as, for example, powertrain diagnostics. Different settings (start of injection, fuel rail pressure, fuel ratio) or operation points (speed and torque) may be modelled with separate networks. Depending on the system's state, different subsystems are chosen to generate the system's desired output. This subsystem can

again be split into redundant networks in order to find the best solution for the task. A possible structure is presented in figure 3.13.

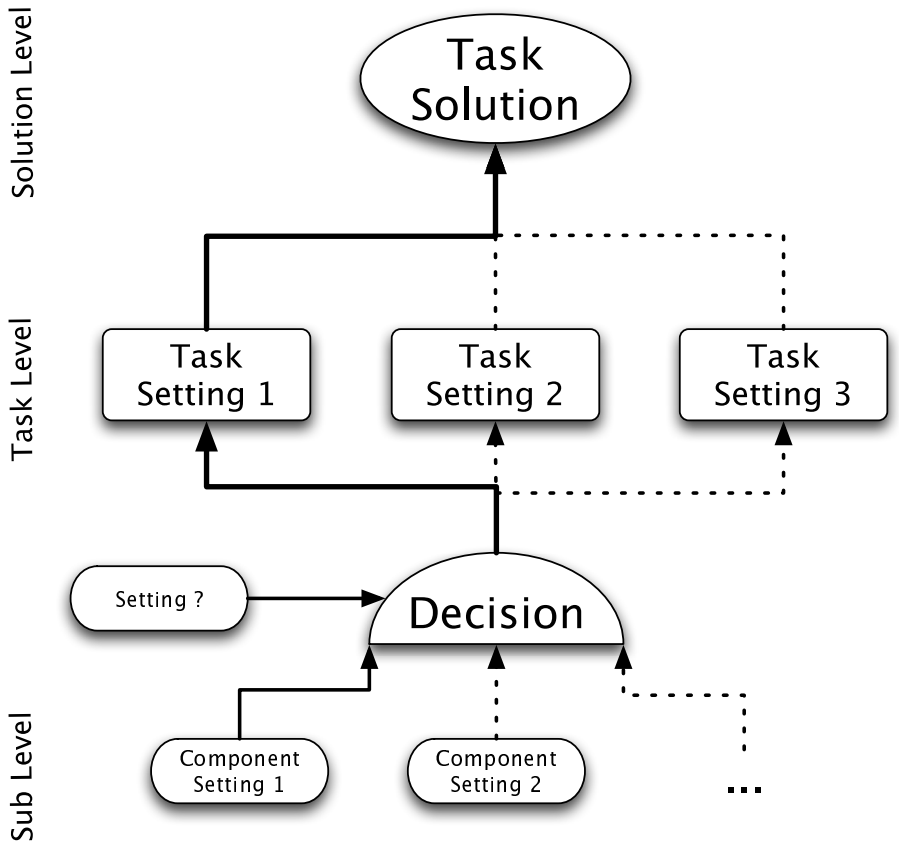


***Figure 3.13:*** *Scheme of operational variability: Several networks predict the same output for different operation cases and are trained on signal variability*

Examples within literature can be found such as subtask modelling of a system described in the work by Soumelidis et al. [65], which discusses the development of an initial modular powertrain model including a model of an electric motor and a model of a transmission. Each of the models was replaced by a neural network in the following in order to investigate if similar results can be achieved. Guoyin et al. [66] introduced definitions for parallel neural network structures that are classified into three variations:

- Parallel network system with single task
- Parallel network system with multiple tasks
- Mixed parallel system networks

A parallel network system with a single task can be applied for redundancy problems. Such a system structure enables the classification or detection of certain signal patterns. Each network is trained on different input variations. This approach ensures the broadest spectrum of the signal is covered and redundancy is created in case a network fails to determine the output. In those cases, networks can also be used for generating voting results that are analysed and combined towards a system output. In their work [67, 68], Sharkey et al. state that different training patterns applied to different modules increase the generalisation capability of the whole multi-net compound and, consequently, the overall classification or approximation performance.

The second item, i.e. the parallel network with multiple tasks, is chosen e.g. by Sharkey et al. [69] who developed a fault diagnosis tool for a marine diesel engine where a combination of network modules forms a multi-net system. Each module is trained for a particular subspace question, in this case a certain engine fault. The actual problem can then be detected by combining the results.

This technique has also been applied in the conference paper: Diesel Engine Prediction Using Parallel Neural Networks by Maass et al. [63] which is presented in section 4.2.3. In this case, the network compound consists of independently trained NLARX networks covering certain signal parts. The results are summed up to provide an overall result. Lee et al. [70] state that the application of several networks can be used to find the optimum on the error surface. This decreases the risk of converging into local minimum on the error surface. Also, the application of networks on subspaces reduces the overall computational performance of the network due to complexity reduction in each network.

## 3.3 Optimisation Methods of Artificial Neural Networks

The next step after choosing the ANN architecture is the network's optimisation which is aimed at reducing the error $e$ between the teaching data output $y$ and the network output $\hat{y}$. The optimisation of an ANN can be controlled through different parameters:

- Number of layers
- Number of neurones
- Activation function definition
- Weight matrix assignment

Whereas the choice of numbers for layers and neurons is experience-based and trial-and-error dominated, the actual weight assignment is typically solved through numerical optimisation.

As already mentioned in the previous section 3.2, a reasonable network consists of up to 2 or 3 layers. The literature describes that networks with more than 3 layers are considerably more difficult to train and optimise due to increasing complexity and pure computational expenditure as some of the training algorithms require computing-intensive operations. The number of neurons within each layer or how many delayed inputs and recurrent outputs a network requires for optimum performance is also trial-and-error dominated. However, the literature describes two approaches of finding optimal numbers of neurons as: 1. network growing or 2. network pruning. The former is based on the idea of building a network with as few neurons as possible, while the latter approaches the solution from the opposite direction by introducing a large network with numerous neurons that in turn result in an over-fitted network. This is characterised through a very close match of network outputs with presented training data, but the ability to generalise over unseen data is be poor. The pruning technique systematically eliminates irrelevant links between neurons until an adequate network performance is achieved The definition of transfer functions is dependent on the actual system characteristics. As described in subsection 3.1.2, the commonly used activation function for non-linear system behaviours are sigmoidal functions for hidden layers and linear activation functions for the output layer. In case of classification problems the domain of threshold functions plays an important role.

The major task for network optimisation lies in the weight assignment. Several algorithms have been proposed over the last few decades. The idea is to find a minimum for a cost function that is based on the error between the training data and the network output. Training a network can be carried out either in a supervised or unsupervised manner. In the former, the network is taught with teaching data that incorporates the system's behaviour for as many characteristics

of input and output data as possible. It is used to teach the network the system's response towards certain input features by adjusting connection weights until the minimum value of a function of the errors between the teaching output data $y$ and the network processed output $\hat{y}$ is achieved. The latter method, unsupervised learning, applies competitive rules in order to build and distinguish different classes in the input data of the environment. These unsupervised approaches are not further investigated in this work.

Different ANN may require certain training algorithms in order to satisfy different learning tasks such as pattern recognition or classification, function approximation, control application or filtering. Consequently, feed-forward architectures require different techniques to recurrent and temporal networks which include dynamics of systems. However, the basic approach for weight optimisation and reduction of the cost function can be described on the gradient descent algorithm used in feed-forward structures. The aim of all algorithms is to reduce the cost function and hence the error $e$ between training data output and network output.

The error at the network output node $j$ is defined by the error $e_j(n)$:

$$e_j(n) = d_j(n) - y_j(n) \tag{3.11}$$

where $d_j(n)$ is the desired response and $y_j(n)$ the current output of node $j$ at step $n$.

The cost function $\xi(w, n)$ of the network is defined by the sum of the square of the output errors associated with the nodes at the output layer:

$$\xi(w, n) = \frac{1}{2} \sum_{j=1}^{k} e_j^2(n) \tag{3.12}$$

where w is a vector of adjustable weights of the network. Hence, with every presentation of training data it is the aim to choose w to reduce the cost function $\xi(w)$ to a local optimum where:

$$\nabla \xi(\mathbf{w}) = 0 \tag{3.13}$$

with the gradient operator $\nabla$ represented by:

$$\nabla = \left[\frac{\delta}{\delta w_1}, \frac{\delta}{\delta w_2}, \frac{\delta}{\delta w_3}, ..., \frac{\delta}{\delta w_m}\right]^T \tag{3.14}$$

This cost function needs to be differentiable in respect of w in order to find an optimum on the weight space.

The first-order partial derivatives may be used to define the search direction on the error surface in order to find the minimum. By applying this information, the following definition of the gradient descent algorithm can be made:

$$\mathbf{w}(n-1) = \mathbf{w}(n) - \eta \mathbf{g}(n) \tag{3.15}$$

where g:

$$\mathbf{g} = \nabla \xi(\mathbf{w}) \tag{3.16}$$

and $\eta$ is a learning-rate parameter that defines how big the search steps on the error surface are, which is also referred to as the weight space. The temporal performance of algorithms is a trade-off between small and large learning-parameter step sizes. In case of a small change, the search route will be smoother, resulting in a slower convergence time. On the other hand, if the learning rate is defined through a big step size, the algorithm may be become unstable and will not meet the target.

This basic algorithm can be improved by using second-order derivatives or, as described in Newton's method-using quadratic approximation in order to minimise the cost function [59]. The general problem with gradient methods is the slow convergence time. Depending on the error tolerance and the size of the network, the number of iterations to find the optimum can be computationally intensive. In addition, these algorithms are not protected against running into local minima on the error surface. Hence, the resulting minimum of one run of the algorithm is not necessarily the global optimal solution.

### 3.3.1 Back-Propagation Algorithm

The back-propagation algorithm consists of two main phases which are repeated iteratively until the optimal of the cost function is found. The core of the back-propagation is the gradient descent described in equations 3.17 to 3.22. The aim is to reduce the function:

$$\xi_{av} = \frac{1}{N}\sum_{n=1}^{N}\xi(w,n) \tag{3.17}$$

where $\xi_{av}$ describes the average squared error over the set of training samples $1, ..., N$. The process of the back-propagation algorithms is the computation of network outputs including all hidden layer signals before back-propagating the local gradients in order to calculate weight changes for connecting links between the nodes. An initial training example with an input vector $\mathbf{x}(n)$ and a desired output vector $\mathbf{d}(n)$ is presented to the network. This is followed by calculating the activation potentials for each neuron defined by the general expression in equation 3.18.

$$\mathbf{v}_j^{(l)}(n) = \sum_{i=0}^{m_0}\mathbf{w}_{ji}^{(l)}(n)\mathbf{y}_i^{l-1}(n) \tag{3.18}$$

where $\mathbf{y}_i^{l-1}(n)$ is the output signal of a neuron $i$ in the previous layer $l-1$ that is multiplied by the weights $\mathbf{w}_{ji}^{(l)}(n)$ assigned to the connection from layer $l$ to layer $l-1$ between the neurons $i$ and $j$. The output of a neuron $j$ in layer $l$ is consequently defined by 3.19:

$$\mathbf{y}_j^l = \varphi(\mathbf{v}_j^{(l)}(n)). \tag{3.19}$$

where the $\mathbf{y}$ is an output of the transfer function $\varphi$ in dependency of the activation potential $\mathbf{v}_j^{(l)}(n)$ of the corresponding neuron.

For $y_j^m$ in a network with $m$ layers, two special cases occur. If the neuron $j$ is in the first hidden layer the output signal of the input layer is equal to the $j^{th}$ element of input vector $\mathbf{x}(n)$:

$$\mathbf{y}_j^0 = \mathbf{x}_j(n).$$

The other special case is if the neuron is located in the output layer $l = L$. Then:

$$\hat{\mathbf{y}}_j^L = \mathbf{o}_j(n)$$

with $\mathbf{o}_j(n)$ representing the output signal of a neuron $j$ from the network. With this signal the output error can be subtracted from the desired response $\mathbf{d}_j(n)$ - equation 3.20:

$$\mathbf{e}_j(n) = \mathbf{d}_j(n) - \mathbf{o}_j(n). \tag{3.20}$$

This forward computation is then followed by a backward calculation of the local gradients $\delta$ that is defined by 3.21:

$$\delta_j^{(l)}(n) = \begin{bmatrix} \mathbf{e}_j^L \varphi_j'(\mathbf{v}_j^{(L)}(n)) & \text{for neuron j in output layer L} \\ \varphi_j'(\mathbf{v}_j^{(l)}(n)) \sum_k \delta_k^{l+1}(n) \cdot \mathbf{w}_{kj}^{(l+1)}(n) & \text{for neuron j in hidden layer l} \end{bmatrix} \tag{3.21}$$

With this gradient, the adjustment for the individual weights can be calculated on the basis of the generalised delta rule as shown in equation 3.22.

$$\mathbf{w}_{ji}^{(l)}(n+1) = \mathbf{w}_{ji}^{(l)}(n) + \alpha[\mathbf{w}_{ji}^{(l)}(n-1)] + \eta \cdot \delta_j^{(l)}(n) \cdot \mathbf{y}_i^{(l-1)}(n) \tag{3.22}$$

where $\alpha$ represents a momentum constant and $\eta$ a learning-rate parameter. These parameters are adjusted over the number of iterations in order to increase the precision in finding an optimum. A momentum coefficient can be applied as a filter in order to smoothen the gradient oscillation on the trajectory in the weight space. Further explanations about the definition of the delta rule and the local gradients can be found in [59].

### 3.3.2 Back-Propagation Through Time

An extension of the popular back-propagation algorithm is the version for temporal MLP networks that adds the dimension of time delays to the optimisation process. This is achieved by unfolding the network into its time dimension [59, 71]. The unfolded network grows with each processing step as displayed in figure 3.14 where the schematic development of the

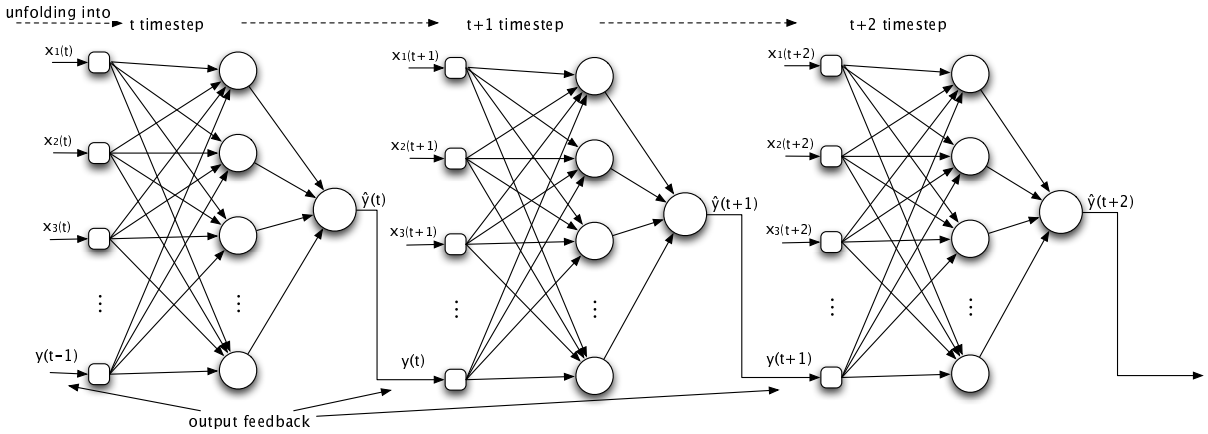


***Figure 3.14:*** *Back-Propagation Through Time processing scheme*

algorithm is visualised.

The network consists of layers $n_0, ..., n_1$ where the former is the start time and the latter the end time of the training example. As a consequence, each layer consists of a time step rather than a neuron set and hence the back propagation is operated in the time dimension rather than processing information backwards through the neural network. The corresponding cost function for this algorithm is defined by 3.23:

$$\xi_{total}(w) = \frac{1}{N} \sum_{n=1}^{N} \sum_{j \in \mathcal{A}} \mathbf{e}_j^2(w) \tag{3.23}$$

where $\mathcal{A}$ is the set of indices $j$ for all samples of output neurons. Hence, the cost function depends on the sum of error output of the neurons in $\mathcal{A}$ over all time steps $1, ..., N$. The local gradients represent the sensitivity through the partial derivatives of the cost function $\xi_{total}(w)$ with respect to the network's connection weights and is defined by equation 3.24:

$$\delta_j(n) = \begin{bmatrix} \varphi_j'(\mathbf{v}_j(n))\mathbf{e}_j(n) & \text{for } n = N \\ \varphi_j'(\mathbf{v}_j(n)) \left[\mathbf{e}_j(n) + \sum_{k \in \mathcal{A}} \delta_k(n+1) w_{kj}(n)\right] & \text{for } 1 < n < N \end{bmatrix}. \tag{3.24}$$

Following the gradient calculation in the backward computation, the weights of the network are adjusted according to the rule 3.25:

$$\triangle \mathbf{w}_{ji} = \eta \sum_{n=n+1}^{N} \delta_j \mathbf{w}_i(n-1) \tag{3.25}$$

where $\eta$ is the learning rate and $w_i(n-1)$ represents the input into neuron $j$ at timestep $n-1$. For a more elaborate explanation of the back-propagation algorithm the author refers to the literature [59]. Here the definitions are plainly for support of the understanding of the gradient descent search algorithms.

The back-propagation shows a good computational efficiency but can lead to intensive storage requirements depending on the network depth and size or length of the training examples.

### 3.3.3 Numerical Optimisation Methods

The slow convergence and restriction to first-order gradient information of the steepest descent search used in classic back-propagation can be overcome by Newton's method which is known as a second-order optimisation method. Second order optimisation methods utilise the Hessian matrix, which is a matrix of second derivatives of the cost function $\xi(\mathbf{w})$ with respect to the decision vector w.

$$\mathbf{A_n} = \nabla^2 \xi(\mathbf{w})|_{\mathbf{w}=\mathbf{w_n}} \tag{3.26}$$

In case of a quadratic function with a strong minimum, Newton's method can converge and find the optimal minimum in one step. However, if the function is not quadratic it cannot be assumed that the method converges at all [62]. Another disadvantage with Newton's method is that the storage requirement of the second derivative grows quadratically [59]. This, consequently, is impractical in case of complex input/output relationships with a wide range of connections. These disadvantages can be overcome by variations of some methods. The Quasi-Newton method that uses an estimate of the Hessian matrix and its inverse requires less storage. However, its computational expenses restrict it from use with large and complex networks. For more detailed information on Newton's and the Quasi-Newton method, the literature of [59, 62] can be consulted. Here, two other classes of second-order methods are presented, the Conjugate Gradient method and Levenberg-Marquardt method.

**Conjugate Gradient Method** - The conjugate gradient method avoids the processing, storage and inversion of the Hessian matrix and searches along conjugated vectors for the

optimal minimum. First, the gradient of the function $\nabla\xi(\mathbf{w}) = g_n$ is derived at each step $n$ for each iteration, which results in a search direction along the steepest descent. Secondly, each new direction $p_n$ is a combination of the gradient and the previous direction $p_{n-1}$ defined by 3.27:

$$p_n = -g_n + \beta_n p_{n-1} \tag{3.27}$$

where $\beta_k$ is a scaling factor determining the length of the search step along the defined vector. It can be chosen by several different methods. Two common definitions are defined by 3.28 or 3.29in [62]:

1. Polak-Ribiere:

$$\beta_n = \frac{\Delta \mathbf{g_{n-1}^T g_n}}{\mathbf{g_{n-1}^T g_{n-1}}} \tag{3.28}$$

2. Fletcher-Reeves:

$$\beta_n = \frac{\Delta \mathbf{g_n^T g_n}}{\mathbf{g_{n-1}^T g_{n-1}}} \tag{3.29}$$

Finally, the search direction defines the next step of the iteration, as seen in 3.30:

$$\mathbf{w}_n = \mathbf{w_{n-1}} + \alpha_n \mathbf{p_n} \tag{3.30}$$

where $\mathbf{w_n}$ represents the new point on the error surface determined by the previous point $\mathbf{w_{n-1}}$, the current search direction $\mathbf{p_n}$, and the learning rate $\eta$ which is defined by 3.31:

$$\eta_n = -\frac{\mathbf{g}_n^T \mathbf{p_n}}{\mathbf{p}_n^T \mathbf{A}_n \mathbf{p_n}}. \tag{3.31}$$

**Levenberg-Marquardt Algorithm -** Another variation of Newton's method is the Levenberg-Marquardt algorithm. However, it avoids the computationally expensive Hessian matrix by

approximating it with 3.32:

$$\mathbf{A} = \mathbf{J}^T\mathbf{J} \tag{3.32}$$

where $\mathbf{J}$ is the Jacobian matrix that consists of first-order derivatives of the network errors with respect to the weights and biases [62]. This becomes possible if the cost function represents the form of a sum of squares as in the case of feedforward training where the cost function is represented by the equation 3.17.

The algorithm is operated in a back-propagating manner. A training example is presented to the network resulting in a network error. This error is used for the calculation of the Jacobian matrix through determination of so-called Marquardt sensitivities $\mathbf{s}_j^l$, the matrix elements that are defined by 3.33:

$$\mathbf{s}_j^l = \frac{\delta \mathbf{e}_j^l}{\delta \mathbf{v}_j^l} \tag{3.33}$$

with the derived error $\mathbf{e}_j^l$ over the activation potential $\mathbf{v}_j^l$. The sensitivities are determined through the back propagation procedure as presented earlier. For each neuron within the network a sensitivity is determined. Those sensitivities define the Jacobian matrix $\mathbf{J}$:

$$\mathbf{J} = \begin{bmatrix} \frac{\delta \mathbf{e}_1^1}{\delta \mathbf{v}_1^1} & \frac{\delta \mathbf{e}_1^2}{\delta \mathbf{v}_1^2} & \cdots & \frac{\delta \mathbf{e}_1^L}{\delta \mathbf{v}_1^L} \\ \frac{\delta \mathbf{e}_2^1}{\delta \mathbf{v}_2^1} & \frac{\delta \mathbf{e}_2^2}{\delta \mathbf{v}_2^2} & \cdots & \frac{\delta \mathbf{e}_2^L}{\delta \mathbf{v}_2^L} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\delta \mathbf{e}_j^1}{\delta \mathbf{v}_j^1} & \frac{\delta \mathbf{e}_j^2}{\delta \mathbf{v}_j^2} & \vdots & \frac{\delta \mathbf{e}_j^L}{\delta \mathbf{v}_j^L} \end{bmatrix} \tag{3.34}$$

Where $j$ is the neuron within the layer and $l$ stands for the actual layer of the sensitivity.

With this information, the weight vector for the next step is determined by equation 3.35:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - [\mathbf{J}^T\mathbf{J} + \mu_k\mathbf{I}]^{-1}\mathbf{J}^T\mathbf{e}. \tag{3.35}$$

The parameter $\mu_k$ may be adapted depending on the result of the next propagation. In case of decreasing squared errors, network performance improves and the step $\mathbf{w}_{k+1}$ is accepted

and parameter $\mu_k$ is adapted by reducing it by a previously defined value. In case of increasing errors, $\mu_k$ is multiplied by this value.

This procedure is iterated until the algorithm converges to a predefined value for the sum of squared error. The drawback with this method is the storage of the Jacobian matrix which is an n $\times$ n matrix. In case of substantial networks, this can lead to memory problems. However, the computational costs can be reduced by this method which is designed for feedforward networks. In addition, the Levenberg-Marquardt algorithm has been shown as an efficient alternative to back-propagation and the newton optimisation algorithm as shown by [59, 62, 72]. For this reason the Levenberg-Marquardt algorithm is used throughout this work for optimisation. It is also shown that the algorithm also is applicable to recurrent neural networks if trained in a serial manner.

### 3.3.4 Further Optimisation Capabilities for Increasing Network Complexity

An attractive alternative to gradient based methods is the DIRECT algorithm, which is a deterministic global optimisation algorithm which does not assume that the cost function is differentiable. Originally developed by Jones et al. [73], it became popular in the optimisation of cost functions of neural networks. DIRECT stands for DIviding RECTangles, which captures the main feature of the algorithm that is described as dividing multidimensional spaces into rectangles. This basic approach is also found in Lipschitzian optimisation. However, the described algorithm avoids the determination of a Lipschitzian constant that may either be not easy to determine or non-existent. For further explanations of the algorithms, refer to the literature of [73].

In addition, genetic algorithms are often applied for training artificial neural networks. These algorithms are inspired by the field of evolution [74] and can be used for function optimisation and a broad variety of applications. A solution is found by selection of the so-called "fittest" solutions which are allowed to create a new generation of the network parameters. This evolutionary path is also considered global optimisation algorithm due to its evolutionary approach. Further informationcan be found in the literature as for example of, e.g. [74].

The above-mentioned global optimisation methods are known for their more reliable and robust search approach that avoids local minimum. However, in this work the optimisation approach with common gradient based algorithms provided sufficently good results, so that it was not necessary to employ global optimisation methods.

## 3.4 Summary and Conclusions

This chapter outlined the general idea and theory behind artificial neural networks. It showed some of the more common structures in use and highlighted the distinction between applications and preferred network structures. In addition, current research directions are described where the combination of ANN is within the focus of reducing the complexity of single networks and increasing the performance. The chapter also outlines the theory of gradient-based algorithms such as simple back-propagation but also the more numerical based methods of Levenberg-Marquardt.

This chapter provides the theoretical background about ANN that is required and applied throughout the rest of this work. The focus lies within the application of simple feed-forward networks up to the recurrent NLARX structure. Single network structures are presented as well as combined networks for improved performances. The training algorithms used is the Levenberg-Marquardt code which provides good training results in view of the proximity between network output and teaching output. In addition, the optimisation time is of sufficient speed.

The next chapter will present some initial work on the application of ANN in the field of emissions prediction or engine parameter estimation. Within the scope of this work, the focus is on the choice of training and validation data sets as well as the choice of inputs in order to find a correct mapping capacity of the ANN for the desired output. The work presented within the next chapter is published material.

# 4 Methodology and Model Structures

The previous chapter outlined the theory behind ANN and their working mechanism. It also explained a variety of methods for different applications. An increasing complexity of problems and demands for computational performance cause neural networks to grow and result in large network compounds. Today's applications, especially in powertrain technology, are highly non-linear. Common modelling techniques find it difficult to perform without trade-off between either accuracy or simulation time. The advantage here initially lies with ANN in the simulation performance. Although complex problems are expensive in terms of training, their simulation time can be fast due to simple mathematical relations and information stored within neuron connections. However, ANN are developed for increasingly complex relations and Multiple-Input and Multiple-Output networks lead to huge network compounds if incorporating recurrent characteristics. The next step involves combinations of networks that are arranged to spread complexity over several networks or let them compete for the best solution. Simulation expenditure for complex networks mapping highly non-linear processes can be reduced by creating combinations of networks that either:

1. work in unison towards superior tasks or
2. compete with other networks for task solution

The distribution of networks over a range of operation points can be beneficial for simulation performance because smaller and less expensive ANN can be developed. Training times might increase due to an increasing number of networks but simulation time can be improved due to the least complex networks covering smaller data scopes which in turn require less training information. In terms of control structures, the decentralisation of computationally intensive algorithms such as an ANN can be also an advantage due to savings with regard to memory and processor power requirements.

This chapter describes applications of ANN methodology for engine parameter monitoring and prediction of a medium-sized diesel engine. It is shown how to define a model structure, then train it and validate it for $\mathrm{NO_x}$, PM prediction and fuel path control design. These examples cover the importance of input choice, data characteristics and the advantage of combining networks in different cases. This paper shows initial work on the topic of neural network application in automotive on-board diagnostics and engine control design. The work presented also contains contributions in the field of input choice, the training and validation set desing such as the finding the model architecture with the least error and best coefficient of determination.

## 4.1 Neural Networks in Automotive Application

The automotive sector has applied these kind of models in several different problems. Their main implementation can be seen in control design in the area of engine operation. Hence, in engine development, neural networks are used for control problems such as fuel injection, output performance or speed [75, 76]. In addition, advanced control strategies such as variable turbine geometry (VGT), exhaust gas recirculation (EGR) or variable valve timing (VVT) have been a focus of ANN modelling [77]. Nevertheless, the application of ANN is also used for virtual sensing such as emissions [11, 12] or as described in Prokhorov [10] for misfire detection, torque monitoring or tyre pressure change detection.

The combustion process itself has been investigated and parameters modelled with neural networks by different authors. Potenza et al. [48] developed a model estimating Air-to-Fuel Ratio (AFR) for in-cylinder pressure and temperature on the basis of crank shaft kinematics and its vibrations. Winsel et al. [78] present a method with artificial neural networks. Static and time delay neural networks are implemented and trained for modelling in-cylinder pressure and engine torque. In their work they the show the capability of this modelling approach for spark ignition engines. Here, different parameters are influential than in comparison to the compression ignition process. In the work of He et al. [79], combustion parameters and emissions are modelled under the consideration of boost pressure and EGR.

## 4.2 Emissions Modelling with Artificial Neural Networks

Emissions regulations have become increasingly stringent over the past decade and legislation will pursue this trend in the upcoming years. Within the field of diesel combustion, $\mathrm{NO_x}$ and particulate matter emissions are a particular focus of reduction enforcement. Consequently, new goals are being set for engine manufacturers in order to comply with global emissions standards. These new goals require comprehensive understanding and control procedures for advanced engine technologies and their parameters. However, intensified control therefore leads to growing complexity and costs [1]. In particular, additional sensor systems and their hardware implementation for monitoring and diagnosis purposes contribute towards costs and computational demand. In addition, data acquisition might be ill-posed by slow sensors or slow changing parameters. Here, the prediction and estimation of parameters may be the solution. The method of virtual sensing can overcome those drawbacks. Measurements of more readily accessible data used and made available for engine control management (ECM) can be implemented to map a model relation between available influencing parameters and signals such as emissions.

### 4.2.1 Accuracy targets and measurements

An accuracy target was formulated by an industry partner who had initial experiences with neural network modelling approaches. Their experience showed a 95% accuracy towards the measured target output would suit their needs for application in controller design or on-board diagnostics.

Hence, the performance of the trained network is measured through the coefficient of determination $\mathrm{R}^2$:

$$R^2 = 1 - \frac{\sum_{n=1}^{N}(y_n - \hat{y}_n)^2}{\sum_{n=1}^{N}(y_n - \bar{y}_n)^2}. \tag{4.1}$$

where $\bar{y}_n$ describes the mean value of the desired output data. It defines the amount of explained variability of the system's output by the current network. A value of $\mathrm{R}^2$ =1 represents a perfect fit of desired output and network output which means the model is able to explain

the system's response in full whereas a value close to zero or even negative would mean the variability in the output is not well described by the model.

This coefficient of determination is used throughout the rest of this work as a measurement of accuracy. In certain modelling parts an additional comparison technique is used. A comparison between measured and predicted output shows a linear behaviour in an ideal case. This mapping enables a visual comparison and detection of outliers.

### 4.2.2 $NO_x$ Emission Prediction with a NLARX Structure

In literature, $NO_x$ modelling is presented using different approaches. Comprehensive models representing physical relations such as elaborated CFD models or empirical mappings. Here an NLARX structure is shown as was presented in the work Maass et al. [80] for the SAE meeting on Powertrains, Fuels and Lubricants in 2009. Data is generated from two different heavy-duty diesel engines. One data set results from a Non-Road-Transient Cycle (NRTC) presented in figure 4.1, another data set is created from a set representing a composition of cycles shown in figure 4.2.

**Data Sets** - The first data set consists of 12 inputs such as: torque, boost pressure, engine speed, pilot fuel quantity, final fuel quantity, back pressure, intake manifold pressure and temperature, exhaust temperature and coolant temperatures. The data is recorded at a sample rate of 1Hz over a period of 1200 seconds, the length of an NRTC cycle. The resulting $NO_x$ is presented in figure 4.1.

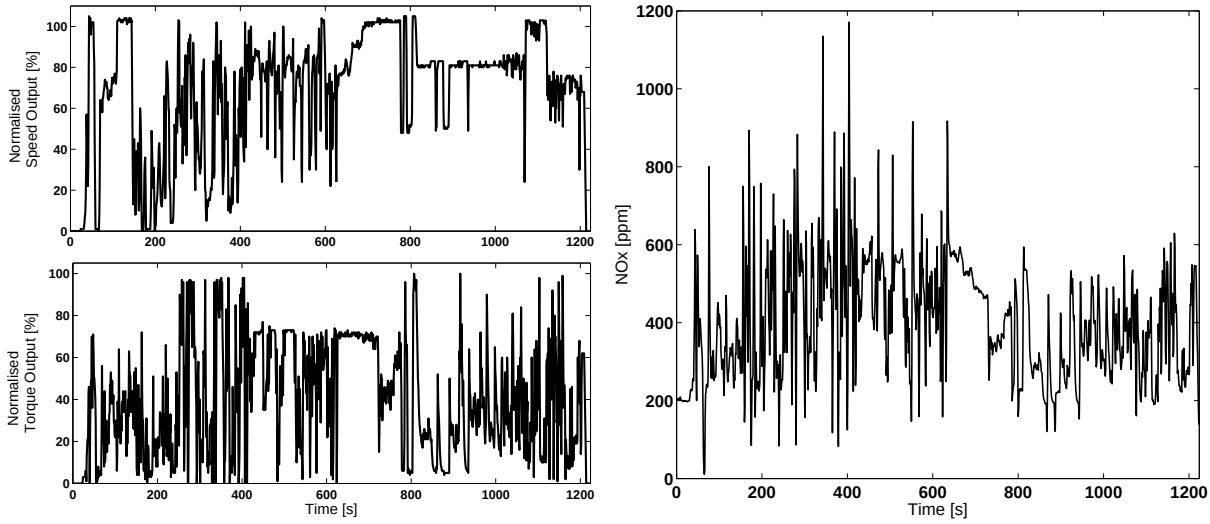


***Figure 4.1:*** *NRTC engine test cycle and corresponding $\mathrm{NO_x}$ output - Data set I*

The second data set consists of 9 inputs and is sampled at 1 Hz over a time period of 2378 seconds. The operating cycle seen in figure 4.2 is a composition of an NRTC, a ramped modal cycle, a full-load curve and some key steady-state points. The set contains 29 repetitions of the cycle in which the engine calibration maps for start of injection (SOI), fuel rail pressure (FRP), and fuel quantity are changed. An exemplary cycle result is shown in figure 4.2 with a resulting $\mathrm{NO_x}$ emissions output.

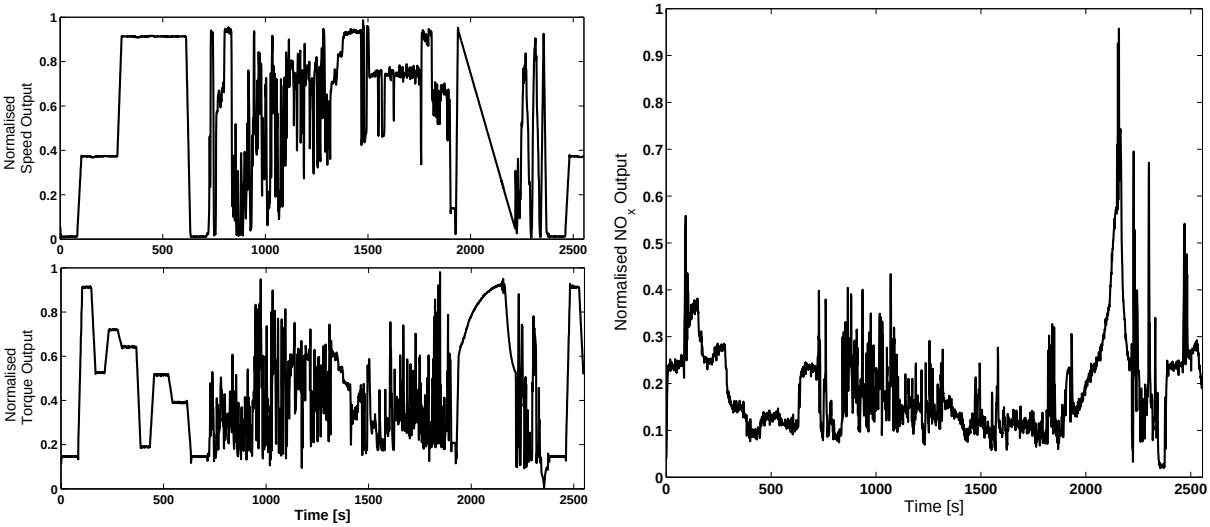


***Figure 4.2:*** *Composition of engine test cycles and corresponding $\mathrm{NO_x}$ output - Data set II*

**Data-Processing** - Both data sets are pre-processed before being used to train an NLARX structure. They are normalised into a range of [0, 1] for reduction of data variability and, hence, improved network performance. At the same time, a training and validation set is formed from each data set.

The first data set is split into quarters and rearranged as shown in figure 4.3. Quarters 1 & 3 and 2 & 4 form the training and validation set respectively. Each set has a length of 612 seconds. The reason for this processing lies in re-distributing training data characteristics. As seen in the original signal in figure 4.1, the first half of the output signal shows high-frequency oscillations over a wide range whereas the second half is characterised through fewer oscillations with smaller amplitudes. In order to present both characteristics to the network and train the network, this new arrangement is set up. It covers both characteristics evenly. The boundaries of the quarters are not processed separately. The main focus was laid that the cuts are made at samples which show similar characteristics in order to reduce the risk of training wrong system behaviour. For the input and output delays at the first quarter the first value is assumed as initialisation value. This strategy is applied also in the next sections and with the same data set for particulate matters prediction.

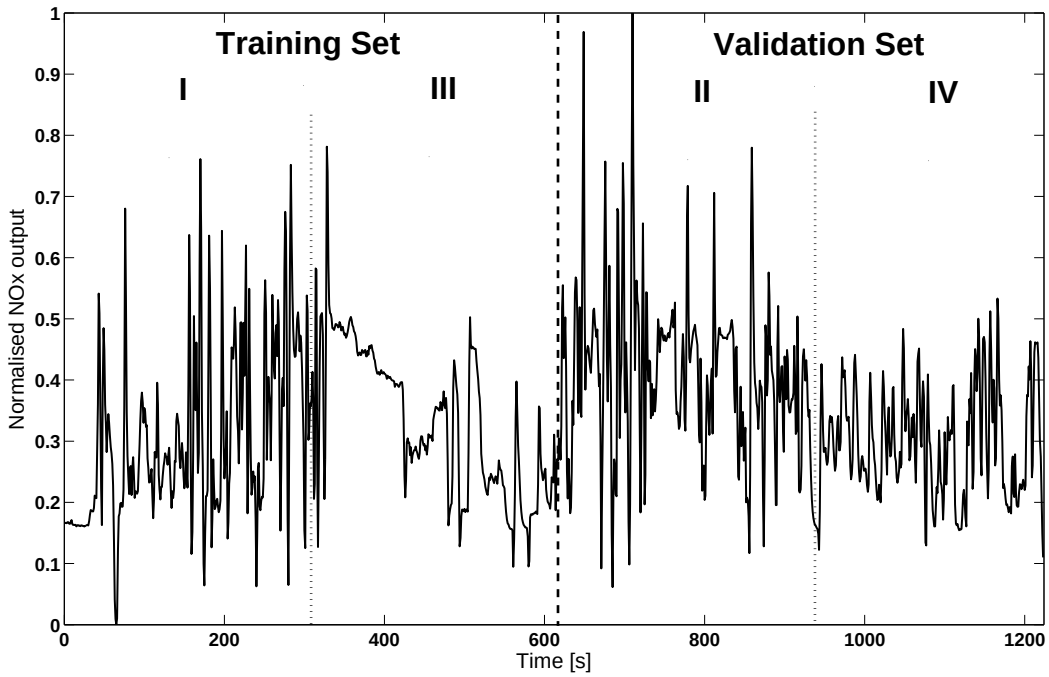


***Figure 4.3:*** *Processed $\mathrm{NO_x}$ output for data set I into training and validation sets*

The second data set is processed differently into training and validation sets. The second set consists of 29 different cycle repetitions. Each cycle has a variation due to calibration changes. Therefore each cycle should represent different characteristics. Initially, only one cycle is chosen for training and the residual 28 cycles are used for validating the network and confirm its generalisation capability. Later the impact of increased information content in the training set is investigated by including several cycles in the training set.

**Training and Validation Results** - The training procedure is operated through a Matlab integrated optimisation. During each training run, the number of neurones per layer, the number of hidden layers, and the delay of input and output feedback can be manipulated by the user. As literature suggests networks with more than three layers do not create sufficient improvement in the view of predictive capability. Hence an initial set-up with three layers was implemented. The number of neurons is systematically varied from 20 neurons per layer down to 4. After reduction of neurons to a minimum of 12 in a three-layer network the layers are reduced to two layers and neurons are pruned from 20 per layer down to 4 per layer. The results are compared and the best structure is defined based on the total number of layers. In case of comparison between feedforward and recurrent networks the recurrency needs to be considered as additional cost due to networks training complexity and the additional number of neurons in the input layer. Here, the best results are achieved by two-layer networks and second-order delays of inputs and output feedback. The training data is presented to the network with tuned parameters and the optimisation algorithm searches for a minimum of the cost function for the NLARX structure as defined by 4.2:

$$\min_{\mathbf{w}} \xi(\mathbf{w}, Z_N) = \frac{1}{N} \sum_{n=1}^{N} \|y(n) - \hat{y}(n|\mathbf{w})\|^2. \tag{4.2}$$

where $Z_N = [y, \mathbf{x}_k]$ with n = 1,...,N is the data set of N samples that is split into training and validation parts. The $y$ represents the desired measured output as shown in figures 4.1 and 4.2. The input vector $\mathbf{x}_k$ contains k inputs also from the recorded data set whereas $\hat{y}(n|\mathbf{w})$ is the network output for sample n and the weight vector $\mathbf{w}$.

During training of a network the measured outputs are used instead of a true recurrent output

feedback. This approach makes it easier and faster to find a valid system mapping. However, in the validation process true recurrence is applied to assure a valid structure. The actual network output is fed back to the input layer and used for the output mapping.

Data set I is the initial step in order to show the NLARX structure's capability to approximate the relationship between the inputs and the $\mathrm{NO_x}$ output. The network parameters found through training achieve a good correspondence between the measured and network output. Figure 4.4 shows the training and validation set comparisons of the outputs. The coefficient of determination for the training set, $\mathrm{R}^2_{\mathrm{train}} = 0.96$ and $\mathrm{R}^2_{\mathrm{valid}} = 0.94$ for the validation set.

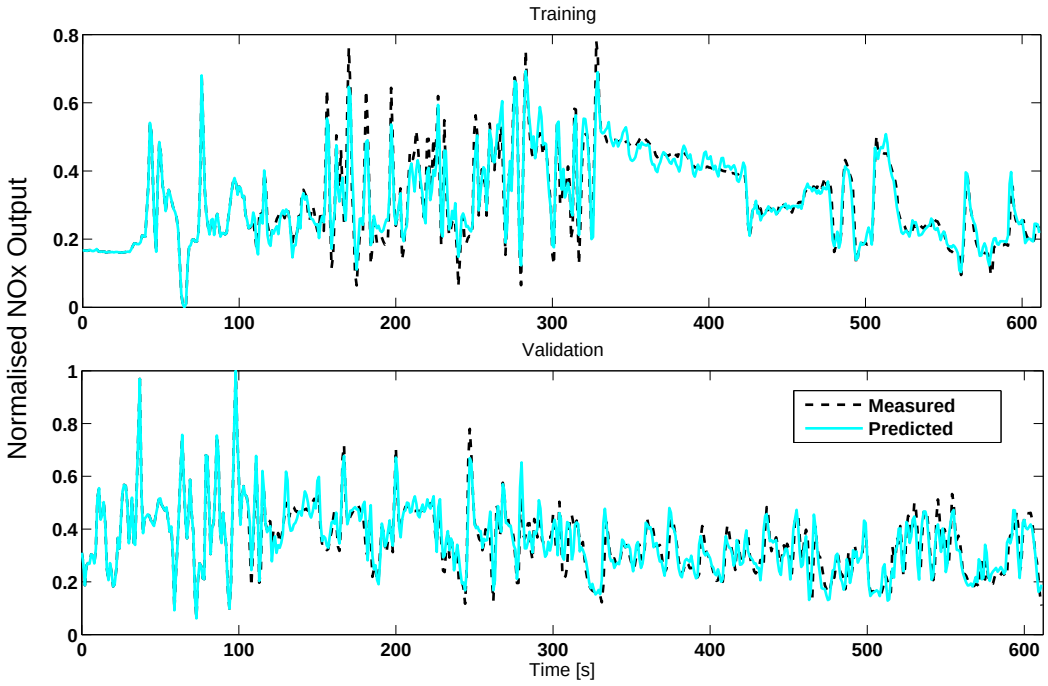


***Figure 4.4:*** *Comparison results for $\mathrm{NO_x}$ output of data set I for training and validation set: $\mathrm{R}^2_{\mathrm{train}} = 0.96$ (top) and $\mathrm{R}^2_{\mathrm{valid}} = 0.94$ (bottom)*

The top graph shows the comparison of the training data against the network output. It can be seen that the network output in light blue follows the desired output well. Although some of the signal peaks are not covered, the characteristic of the signal is represented by the modelled output. The missing signal peaks can be linked to a lack of information in the training inputs. Hence, the importance of data features in inputs is crucial in order to train a network. This is also shown in the second network developed for the second data set.

This data set is used to investigate the flexibility and generalisation capability of the NLARX

structure on varying data sets. This data set covers cyclic variances over the runtime of a cycle but with different calibration settings affecting the output. Firstly, one cycle is used to train the network's weight parameters resulting in a comparison value of $\mathrm{R}^2_{\mathrm{train}}$=0.95 for training as presented in the top graph in figure 4.5.

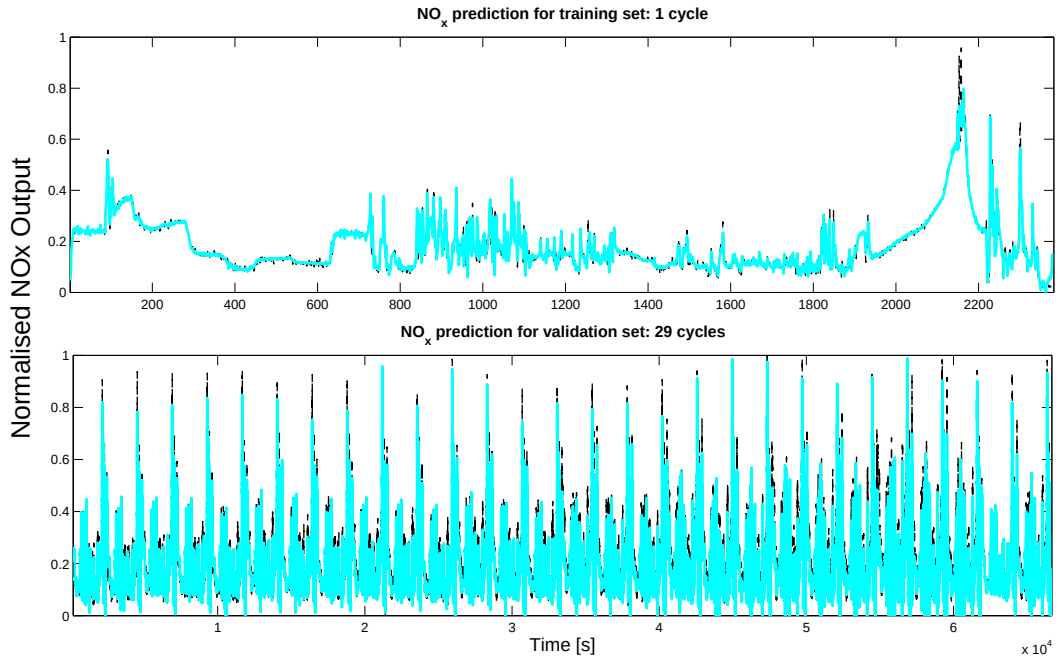


***Figure 4.5:*** *Comparison results for $\mathrm{NO_x}$ output of data set II for training and validation set with 1 cycle for training and 29 cycles for validation*

Subsequently the network is applied to all residual 28 cycles in order to validate the model structure. The visual results are shown in the bottom graph in figure 4.5. The graph shows a lack of model accuracy at high $\mathrm{NO_x}$ outputs. In addition, an overview of all $\mathrm{R}^2_{\mathrm{train}}$ coefficients is plotted over the number of cycles in figure 4.6. The black curve shows a decreasing trend of comparison from the first cycle used for training until the last cycle where the calibration settings are changed significantly. This decrease shows that a single cycle does not contain enough information for generalisation over varying engine settings. When SOI changes, the rail pressure and fuel quantity are reset to the initial settings of the training cycle. This can be observed in the network's performance since the comparison rises at cycle number 10 and 21 where this change occurs. Here the engine settings are closest to the training set behaviour.

Consequently, the next step is to provide additional training information of different engine

settings. Hence, four additional cycles are chosen for training the network. This increases the amount of data available for finding an optimal network set-up. The training set now contains information about different SOI, FRP and fuel quantity settings. The training result is sufficient - $R^2_{train}$= 0.98. For comparison purposes, the validation results are plotted in the same graph as the previous results in figure 4.6 from the single cycle training set. The difference is significant as the results are within the designated accuracy that vary between $R^2_{min}$= 0.94 and $R^2_{max}$=0.97.

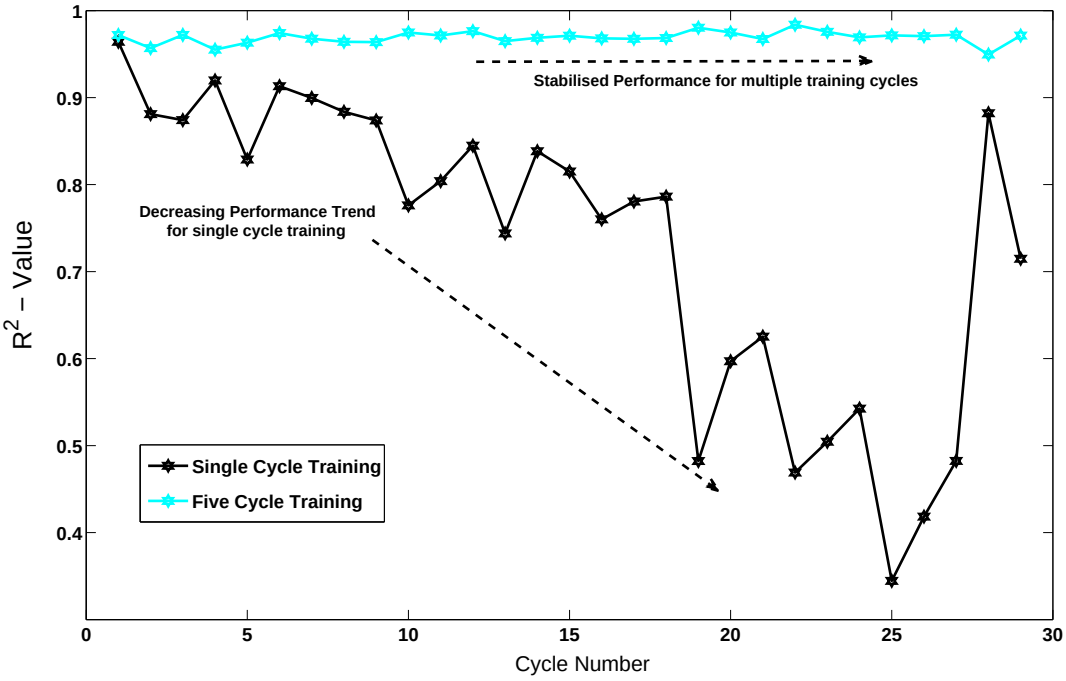


***Figure 4.6:*** *Comparison results for $NO_x$ output of data set II for training and validation set with 1 (black line) and 5 (bright blue line) cycles for training and 29 cycles for validation*

**Conclusion** - The first conclusion of this work is that a NLARX structure is capable of mapping a relation between engine parameters and the emission output $NO_x$. This is shown on a single NRTC cycle data set that is processed into training and validation data. The trained network achieves sufficient results. The second part of this investigation shows the importance of available training information. A single cycle used for training a network resulted in sufficient comparison results on the training set but failed over a cycle batch that includes cyclic variations of engine performance and engine settings of SOI, FRP and fuel quantity. The adaptation of the training set with additional information from other cycles with different

engine settings resulted in an overall improvement of network performance. Only 14% of the whole data set are required in order to achieve a sufficient generalisation capability of the network.

### 4.2.3 Particulate Matter Emissions Prediction with Parallel NLARX Structures

The following investigation presents a model for prediction of particulate matter (PM) emissions which in this case are represented by the term smoke and are stated as being as good indicator for the comprehensive emissions group of PM. Modelling of PM emissions has been tried with several modelling techniques, ranging from comprehensive physical descriptions of the process down to less computationally demanding procedures with quasi-dimensional models or empirical studies. He et al. [79] describe a model that estimates engine output parameters amongst others along with smoke emissions from available engine parameters such as boost pressure and EGR. This work was published at the ACC conference 2009 in the engine diagnostic session titled: Diesel Engine Emissions Prediction Using Parallel Neural Networks [63].

**Data Set and Data Pre-processing** - The data set consists of the same data as described for the previous $\mathrm{NO_x}$ problem - an NRTC set recorded at 1Hz data over 1200 seconds. For initial modelling, the same inputs are used such as: torque, boost pressure, engine speed, pilot fuel quantity, final fuel quantity, back pressure, intake manifold pressure and temperature, exhaust temperature and coolant temperatures. The data is normalised into the range of [0, 1] in order to reduce data variability.

In terms of data partitioning, the same approach as described above is chosen due to the fact that just one set of data is available. This is divided into training and validation parts by rearranging the quarters into 1 & 3 and 2 & 4. In figure 4.7 the smoke output signal and the processed version are presented respectively.

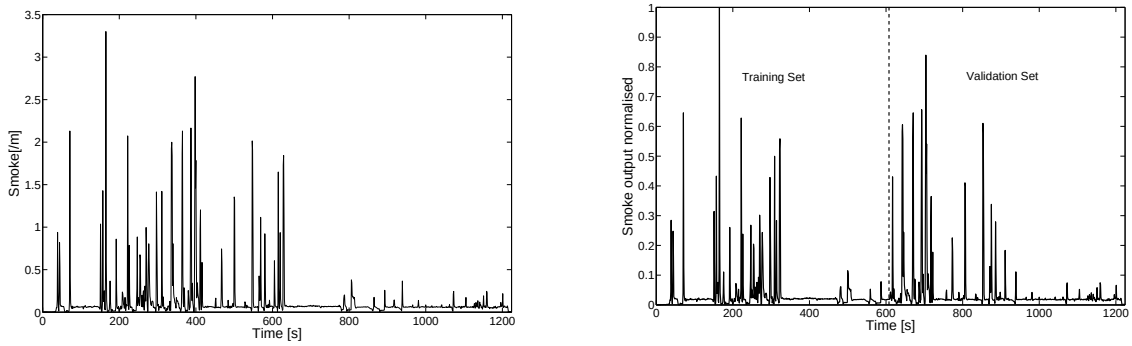


***Figure 4.7:*** *Original and processed smoke output split into training and validation set*

The output signal is characterised by two different parts. In the first half of the signal, high peaks and fluctuations are introduced through wider feature distribution and more transient behaviour in the two variables torque and speed up to 600 s into the cycle as shown in figure 4.1. The second half of the cycle consists of steady-state parts and hence minimal fluctuations are introduced. The behaviour identified can be traced back to the fact that with a rapid change of speed, the combustion conditions also change. Soot formation is regulated by a number of different parameters, which are indirectly influenced by the change to the engine's loading conditions. On the one hand the amount of oxygen that is available for forming organic compounds by oxidation reactions is critical. On the other hand the formation of the spray is crucial. High injection pressures ensure that a sufficient atomisation of the fuel can take place because smaller droplets are less likely to lead to soot formations. A third feature is a high combustion temperature that leads to complete combustion and less in-cylinder soot formation by breaking up fuel droplets through oxidisation [81]. Taking these thoughts into account, the smoke signal can be explained as follows. The first half of the smoke signal is a result of rapid changes in engine speeds. During transients the engine control requires some delay time until a stable condition is achieved that allows for minimum emissions formation. In this phase, the amount of oxygen that flows into the cylinder settles towards a steady-state, whereas the fuel injected may rise due to a load increase. This initial excessive fuel may coincide with a reduction in oxygen flow and, consequently, soot is more likely to be formed. In addition, the duration of combustion is dependent on the amount of oxygen present and the engine speed. As a result, a shorter period of combustion with a decrease in required oxygen can lead to incomplete combustion. The second half of the signal is dominated by steady speed resulting

in a flat output signal. Small visible peaks breaking this signal are due to the sporadic fast speed changes. Due to high speeds of around 80% of the rated speed, the temperature can rise and be kept at a high level. Hence, the conditions during the combustion process are more likely to break-up the fuel droplets and create a more homogenous mixture within the combustion compartment. At the same time, less fluctuations mean less transient states with varying conditions which in turn result in less soot formation.

Other parameters such as fuel injection timing and duration of in-cylinder pressures and temperatures also have an impact on observed engine behaviour. In this case however, the formation is initiated through the two controlled variables torque and speed.

**Parallel Modelling Approach** - The presented final parallel modelling approach has been developed based on the fact that no sufficient results for a single NLARX structure could be found. Extreme signal fluctuations in the first half of the the output signal introduce a so-called hypersensitivity. This leads to high-frequency oscillations with an underlying lower frequency in the prediction signal as it can be seen in figure 4.8. The network becomes inaccurate in steady situations as they are present during the second half of the signal. The approach to overcome this drawback is developed from the work that Guoyin et al. [66] present. Here, a parallel network system with multiple tasks is chosen. Lee [41] states that the operation of several individual networks reduces the risk of getting stuck in a local minimum. Sharkey et al. [64] determine different approaches such as ensembles and modular structures. In this case a modular network structure is set up where each network is assigned with an individual task.

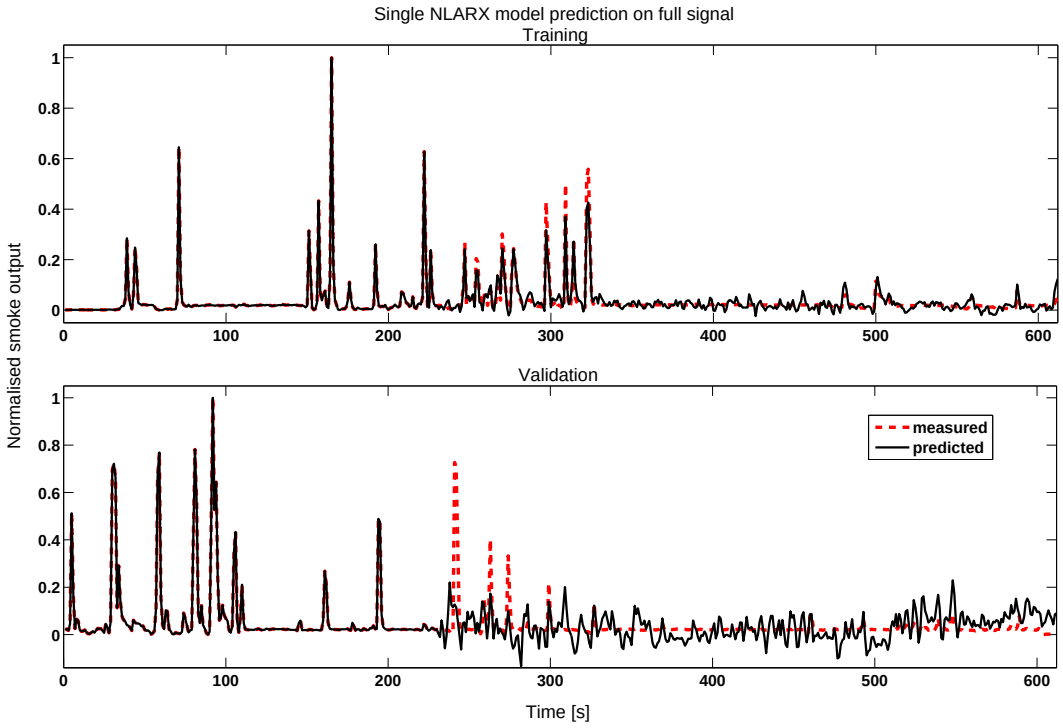


***Figure 4.8:*** *Performance of a single NLARX network designed on present data*

In the presented work the smoke output signal is divided into three vertical regions. Consequently, the amplitudes of signal spikes are cut while the frequencies of residual parts are decreased as shown in figure 4.9. The division of three regions is determined by trial and error giving the best trade-off between results and computational expenditure.

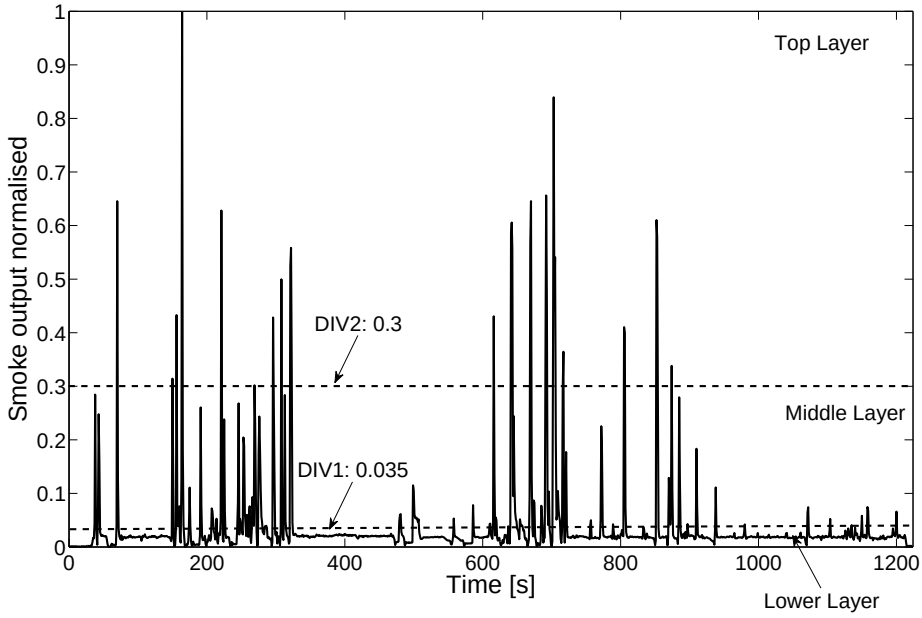


***Figure 4.9:*** *Region division of smoke output signal*

The graph in figure 4.9 shows the division lines and region contents. The first region, referred to here as lower region (LL), consists of signal noise and low frequencies. The remaining part is split into a middle region(ML) and a top region (TL). The ML covers the part of the signal with medium density of oscillations and peaks of a normalised smoke value up to y=0.3. The residual peaks are covered by the top region. In the TL some characteristic peaks are present without any noise or smaller peaks that perturb the signal-to-noise ratio. In table 4.1 the chosen region borders are presented.

***Table 4.1:*** *Division borders of the approach*

| | | |
|---|---|---|
| $0 < \text{LL} < 0.035$ | $\Rightarrow$ | $\Delta y_{LL} = 0.035$ |
| $0.035 < \text{ML} < 0.3$ | $\Rightarrow$ | $\Delta y_{ML} = 0.265$ |
| $0.3 < \text{TL} < 1$ | $\Rightarrow$ | $\Delta y_{TL} = 0.7$ |

A separate network is developed for each output division receiving input information over the whole signal range. The inputs are not divided since information for the output is spread over the whole input range. The parallel processing model structure is presented in figure 4.10. The input vector is the same for all three NLARX networks whereas each network will predict a region output: $\hat{\mathrm{y}}_{\mathrm{LL}}$, $\hat{\mathrm{y}}_{\mathrm{ML}}$ or $\hat{\mathrm{y}}_{\mathrm{TL}}$. This predicted information is combined into an overall signal $\hat{\mathrm{y}}_{\mathrm{overall}}$ that is compared against the overall measured output.

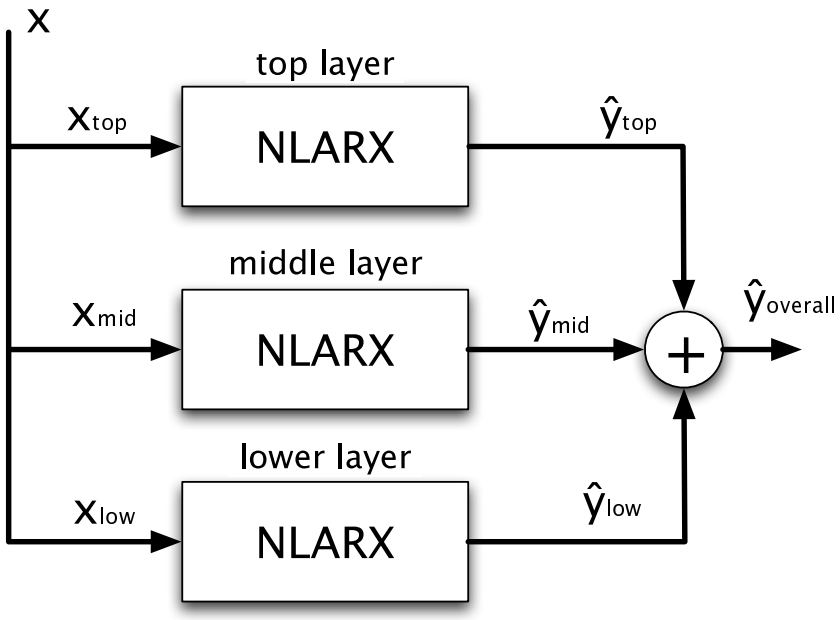


***Figure 4.10:*** *Schematic representation of parallel NLARX model structure*

**Training and Validation Results** - For each region an NLARX model is trained with a corresponding output region. The performance of the network is measured with the previously mentioned comparison coefficient $\mathrm{R}^2$ expressed in equation 4.1.

The lower region (LL) is indicated by (1) the lower part of peak oscillations as well as (2) low peaks and noise. The signal range is reduced by dividing it into three regions which in turn achieves a more homogenous amplitude distribution. This approach favours the choice of NLARX structures for estimation. Consequently, the comparison between measured and predicted output for the training set is sufficient with $\mathrm{R}^2 = 0.97$. The validation set demonstrates the practicability of the chosen structure with $\mathrm{R}^2 = 0.92$. The visual comparison of the two signals is presented in figure 4.11.

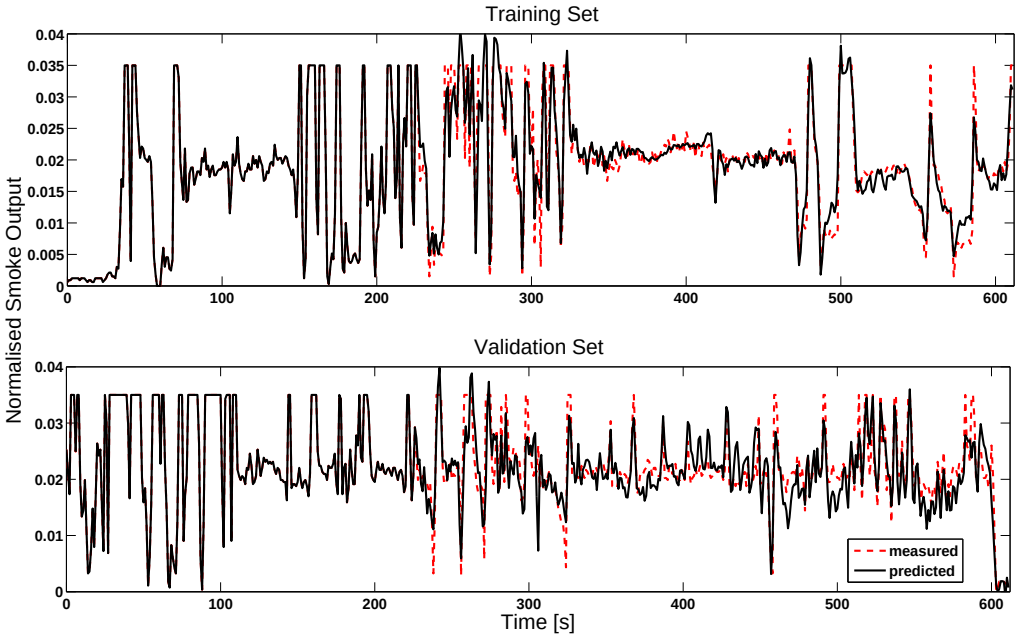


***Figure 4.11:*** *Comparison between measured and predicted model output for lower region of smoke output*

While the first part of the signal is well predicted, the second half is characterised by a number of discrepancies. This observation is present in the other regions as well.

The middle region (ML) represents the middle section of peak oscillations and the medium peaks of the signal. In this region the NLARX achieves a training set comparison between measurements and the model output of $\mathrm{R}^2 = 0.93$. This model's validity is confirmed by the training set comparison of $\mathrm{R}^2 = 0.90$. As expected, this validation value is lower than in the LL due to a broader frequency range in the signal determined by a wider scope of y-values. Here, $\Delta y_{ML}$ is 0.265 wide whereas the first region covers $\Delta y_{LL} = 0.035$.

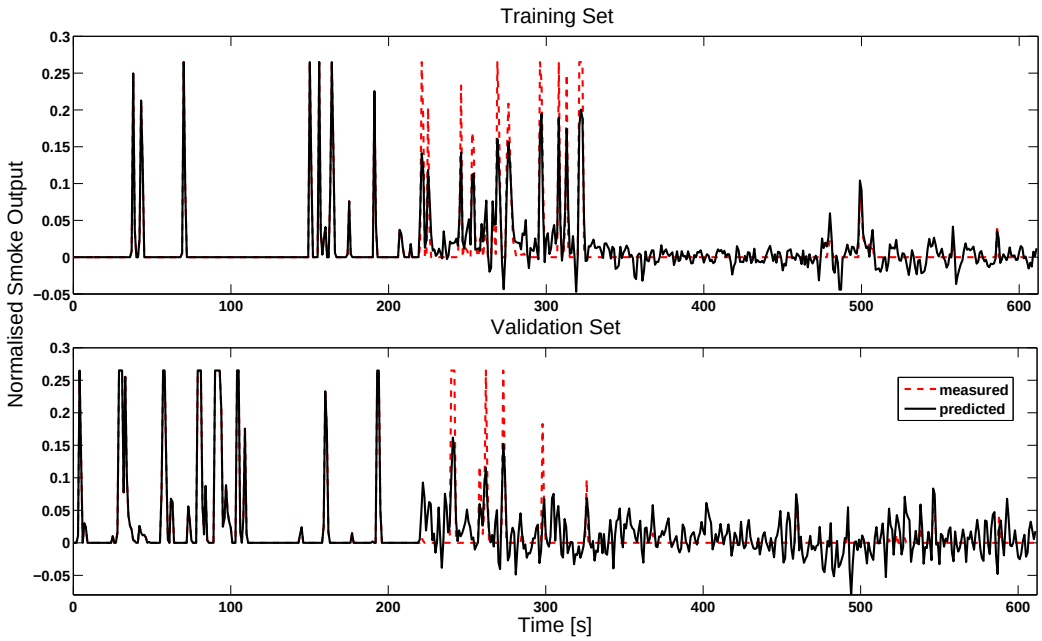


***Figure 4.12:*** *Comparison between measured and predicted model output for middle region of smoke output*

The graph in figure 4.12 that visualises the ML output comparison shows similar characteristics of the predicted output signal as in figure 4.11. The first half of the prediction correlates closely to the measurement whereas the second half is marked by fluctuations. It is assumed this fading of the signal is introduced as a result of the network structure approach. The data within the first half requires different network characteristics to the data in the second half. After introducing the fast-response data and training the network thereon the response is quicker and noise is introduced within the second half.

The top region covers the high peaks of the output signal. The range of $\Delta y_{TL}$ is 0.7. Hence, a higher range of output data leads to a wider frequency range. The training comparison drops to $\mathrm{R}^2 = 0.99$ in comparison to a sufficient $\mathrm{R}^2 = 0.97$ for the validation data set as presented in figure 4.13. The results show a very close comparison between the model and the measured system output. In fact, the peaks marking the smoke output peaks are covered sufficiently and the introduced noise in the second half is present but kept lower since there are no low signal fluctuations.

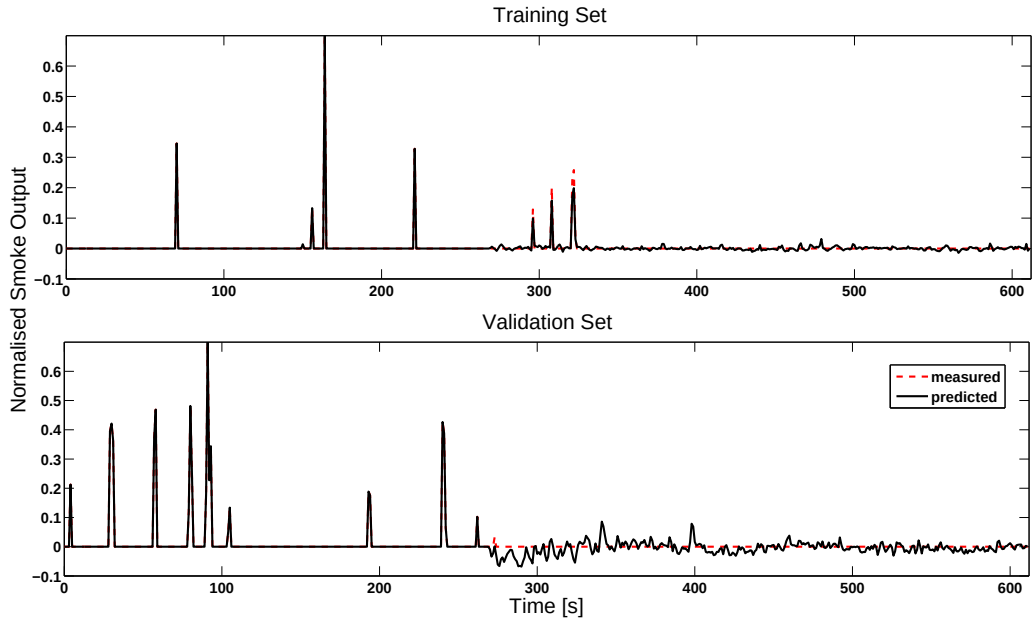


***Figure 4.13:*** *Comparison between measured and predicted model output for top region of smoke output*

The overall result for the output is created by adding together the three network estimation outputs. The comparison with the measured output shows a sufficient result of $\mathrm{R}^2 = 0.97$ for the training and $\mathrm{R}^2 = 0.96$ for the validation set. Here, in addition a linear comparison determines the prediction accuracy as shown in figure 4.14. The data forms a scatter cloud close to the origin in the graphs due to the characteristics of the output signal that is based at zero. The scatter distribution fits a linear comparison close to the unit vector. In figure 4.15 it can be seen that parts initially classified as difficult due to their wide amplitude differences and high frequencies are described sufficiently by the calibrated model. Patterns with high peaks and high density of oscillations show appropriate comparison. However, the less oscillating parts are marked by noise introduced through the calibration approach. The networks are designed for responses on high and fast responses in the first half and overshoot at small oscillations as present in the second half of the signal.

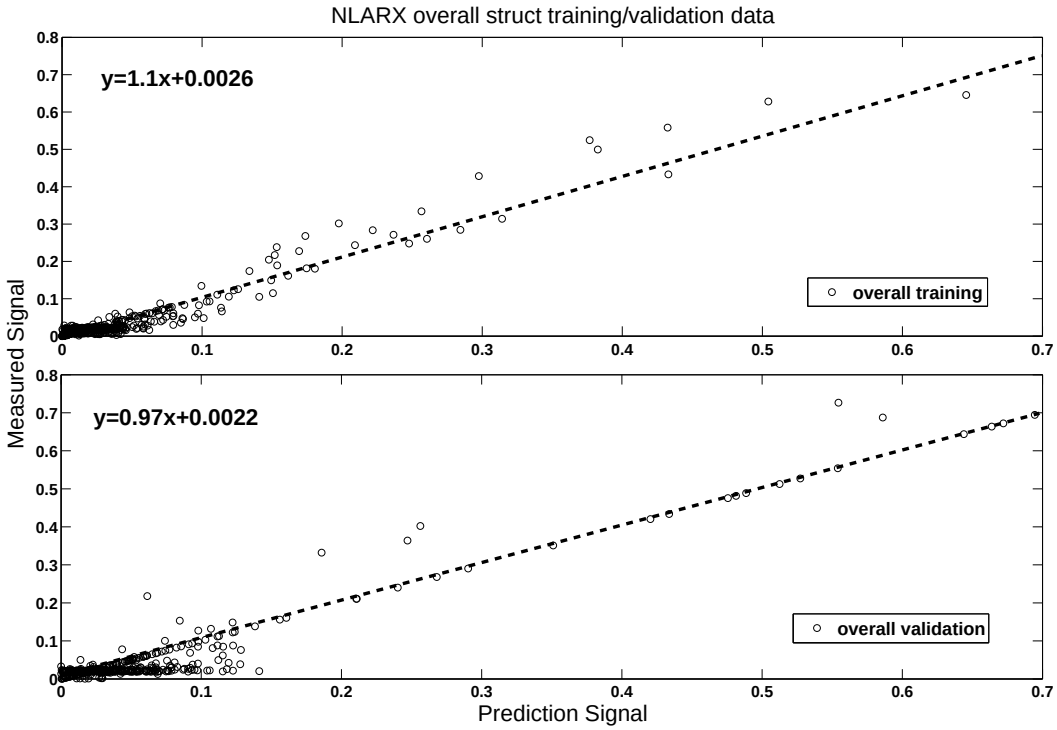


***Figure 4.14:*** *Comparison between measured and predicted model output in linear plot*

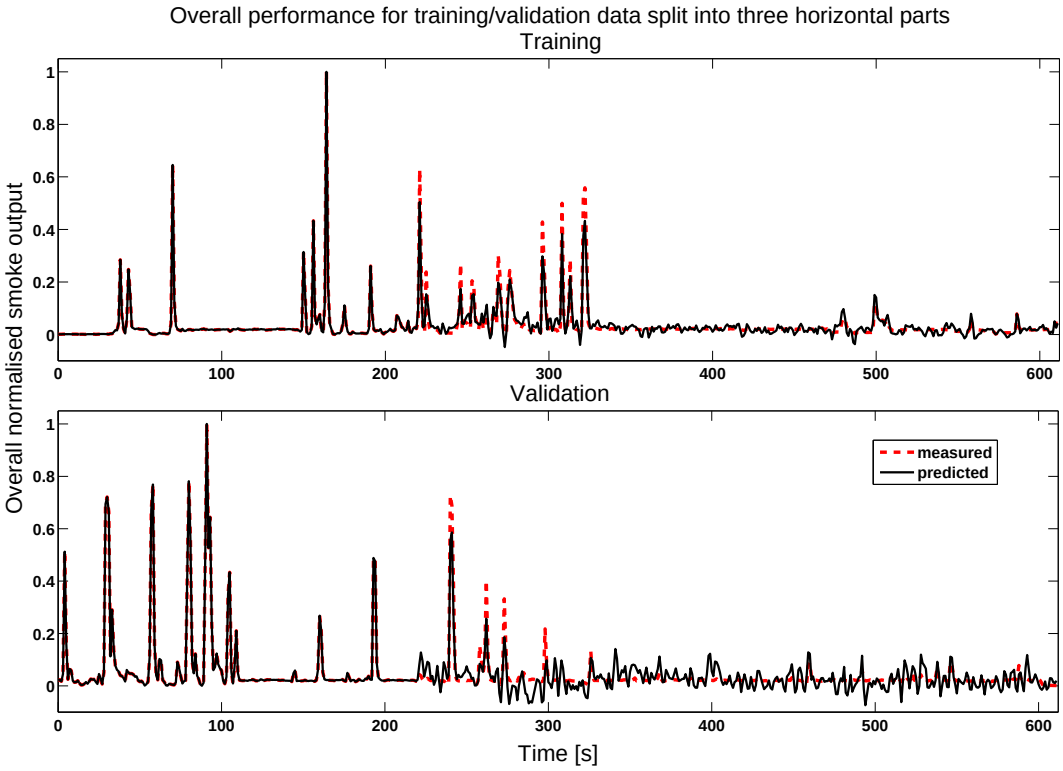


***Figure 4.15:*** *Overall comparison between measured and predicted model output*

**Conclusion** - This work shows the importance of network design on performance. A single NLARX structure cannot create sufficient estimation performances due to signal characteristics. Analysis of the signal and the generation of a parallel network structure where each network is allocated an individual task enables a reduction in the complexity of the signal relations. This in turn results in improved prediction performance for the available data. A further investigation will show the effect of choosing the network inputs more strategically. The next subsection will present additional training and validation data together with a different approach for choosing the network input.

### 4.2.4 Identification of Input Parameters for Soot Prediction

This section describes a further investigation of the virtual sensors for online prediction of smoke emissions of medium and heavy-duty diesel engines. The test section shows results for a variety of engine test cycles and training validation scenarios. The analysis of input data results in an improved model complexity with fewer inputs. It defines the inputs with the highest information density required for sufficient prediction of soot along with the minimum requirements in terms of inputs for meeting a predictive comparison coefficient accuracy target of 95%.

**Initial data generation and model development based on a C6.6 engine test** - For an initial model set-up, four test cycles were run on a C6.6 engine at the test facilities at Loughborough University. The ECM is set to an industrial calibration. The cycles are run in order to create a range of engine response characteristics relating to soot formation. The test cycles are:

**Part A - Random Walk (figure 4.16)** The random walk test was operated in two different versions:

1. '1-slow' test: The slow random walk test runs over a duration of 6218 seconds. The original test cycle covers the complete engine speed-load map. However, due to the engine being in operation for these tests, the maximum load is reduced down to 70 % at the speed points 800, 1000, 1100 RPM.

2. '2.5-fast' test: The fast random walk test runs over a duration of 1001 seconds. This test also incorporates a reduced speed-load. Due to faster ramp times the maximum load is reduced at speed points 800, 1000 and 1100 RPM points. This action is taken in order to avoid engine stall problems in this stage of the test.

**Part B - Constant Speed Load Acceptance (CSLA) test (figure 4.17)** The constant speed load acceptance test runs for a duration of 45 minutes. The engine speed is increased from 1000 RPM to 2200 RPM. At each speed step, torque is stepped up to peak torque, in this case 70% of maximum torque. The peak torque is applied for 500 seconds and the engine response is measured to determine soot emissions.. The ramp times of these tests are presented in the appendix A in the table A.1, whereas the graph with the speed and load changes is presented in figure 4.17

**Part C - Idle to Full Throttle (figure 4.18)** The idle to full throttle test is characterised by a step change from an idle state with no load applied to a full throttle condition with peak torque.

**Part D - NRTC (figure 4.19)**

1. Complete NRTC test with 70% load and full speed range covering a wide range of engine transients in different frequencies and combinations.

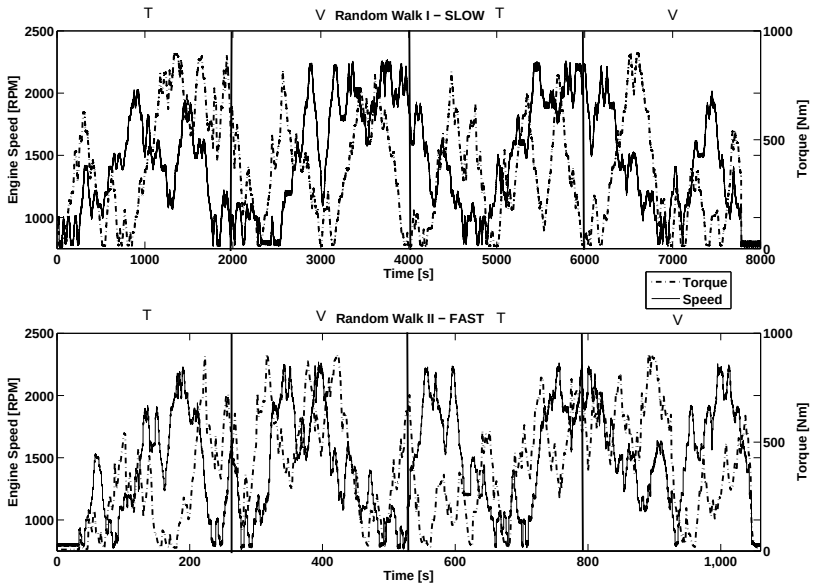


***Figure 4.16:*** *Random Walk 1 & 2 training and validation part distribution*

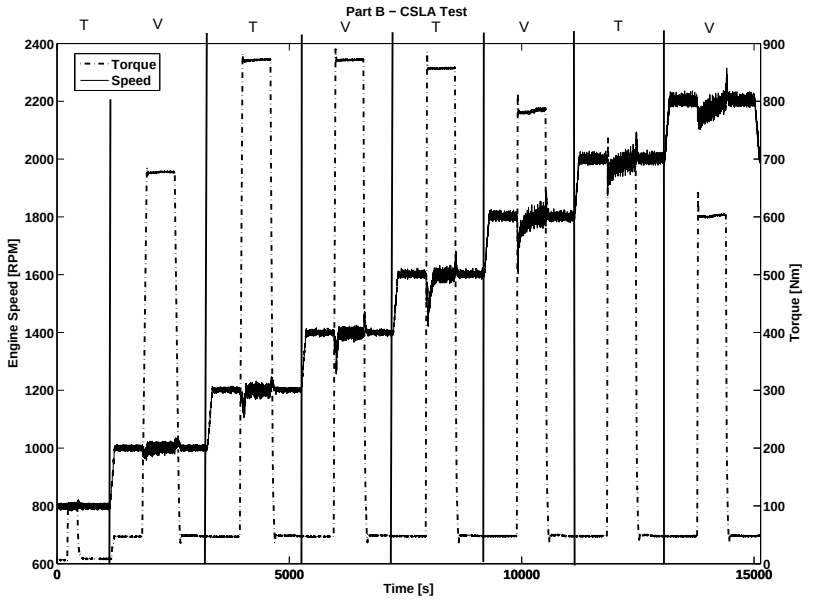


***Figure 4.17:*** *CSLA Test - training and validation part distribution*

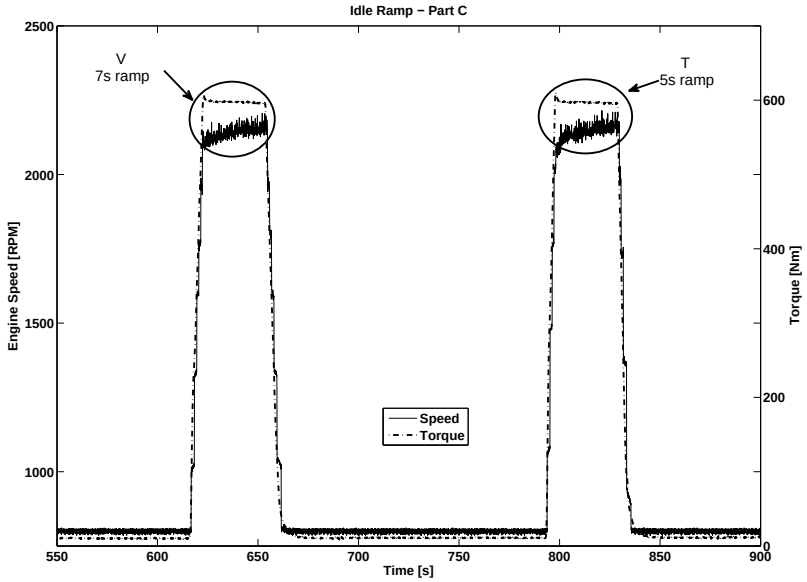


***Figure 4.18:*** *Idle Ramp Test training and validation ramps*

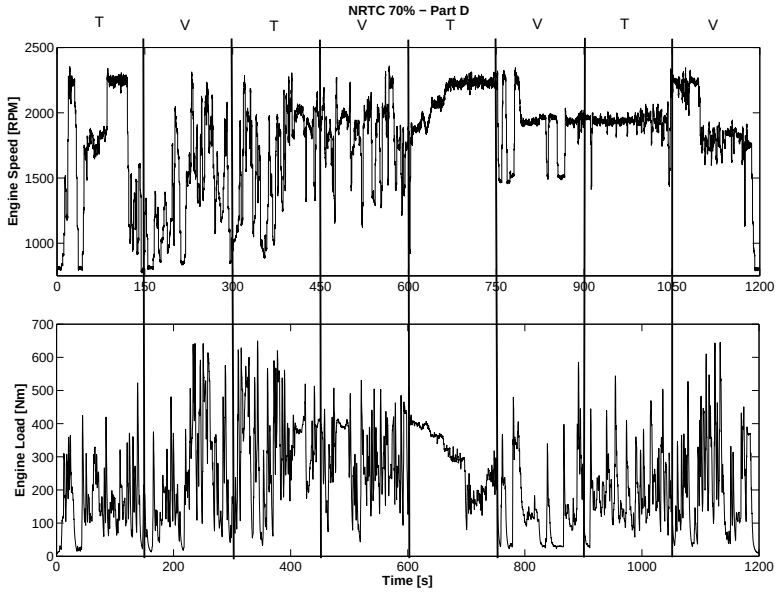


***Figure 4.19:*** *NRTC Test showing speed and torque signal*

**Test Cycle Data Processing** The cycle test data is processed into a training and validation set. Each cycle is split into training (T) and validation (V) parts. These parts are recompiled as shown in the scheme in Table 1 in the appendix. The resulting training and validation sets

are shown in 4.20 and 4.21. Each set contains the same amount of data for each of the test cycles. The feature density covers a wide scope of engine operation behaviour in steady-state (CSLA and Idle Ramp) and transient operation (RW and NRTC). In the graphs the concrete line shows the engine speed curve whereas the dashed-dotted curve represents the torque, the engine load. Both curves show the complete range of the present engine at 800-2300 RPM and 0-900 Nm. Each data set is initially processed into 1 Hz data for the initial model identification.

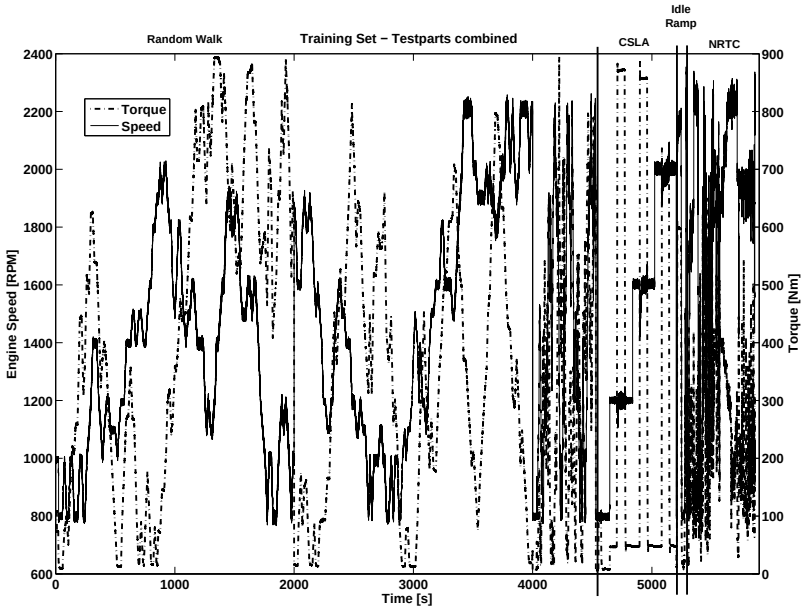


***Figure 4.20:*** *Training Set showing a combination of all parts of the mentioned cycles*

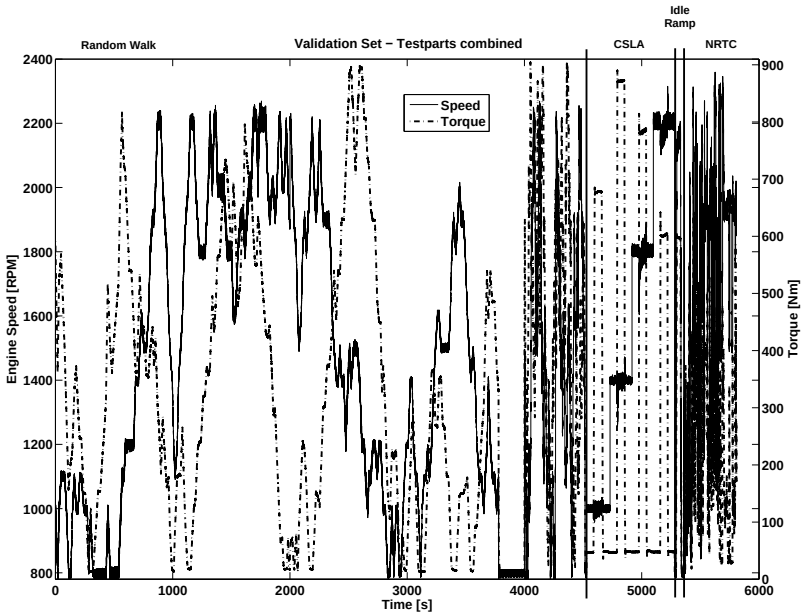


***Figure 4.21:*** *Validation Set showing combination of all parts of the mentioned cycles*

**Initial Model Identification with Seven Inputs** For the initial model identification seven inputs were chosen:

1. Torque/ Load
2. Engine Speed
3. Intake Manifold Temperature (IMT)
4. Mass-Air-Flow (MAF)
5. Air-to-Fuel Ratio (AFR)
6. Boost Pressure
7. Exhaust Pressure.

In the operated test cycles, smoke output of the engine is represented by measurements using an AVL 439 opacity meter.

Two different approaches were tested for this initial model identification in order to find a suitable model structure. Each set was reduced to 1 Hz data and the inputs and outputs are normalised for data range reduction. The three different modelling approaches are:

1. Single NLARX structure
2. Three-layer parallel NLARX structure

Each model was trained with the training set shown in figure 4.20 and validated against the validation set (figure 4.21). The comparison between the desired measured test data and the model-predicted output is determined through the coefficient of determination $\mathrm{R}^2$.

In addition a linear regression plot is presented to show the direct value-to-value comparison of measured and predicted output. The comparison is shown by a diagonal regression line. The closer the value-to-value comparison, the closer it fits the line, which in turns shows a perfect fit of predicted output to the measured output.

The prediction results for each of the approaches for training and validation show similar characteristics:

1. Initial modelling with single NLARX structure [1Hz data]
   a) Training $\mathrm{R}^2$= 0.88
   b) Validation $\mathrm{R}^2$= 0.67
2. Three-layer approach with NLARX structures [1 Hz data] (see [63])
   a) Training $\mathrm{R}^2$= 0.86
   b) Validation $\mathrm{R}^2$= 0.69

The achieved results are not sufficient and require further investigation. In particular the data at 1 Hz does not seem to provide enough information to generate a comparison between inputs and outputs. In addition, the current list of inputs may need further investigation. The next step incorporates two additional inputs.

**Model Identification with nine Inputs** The additional inputs are:

1. Fuel-Rail Pressure (Common Rail Pressure)
2. Fuel Quantity.

Those inputs may increase the information content available to predict the actual opacity output and lead to an input list of nine inputs:

1. Torque/ Load
2. Engine Speed
3. Intake Manifold Temperature (IMT)
4. Mass-Air-Flow (MAF)
5. Air-to-Fuel Ratio (AFR)
6. Boost Pressure
7. Exhaust Pressure
8. Common Rail Pressure
9. Fuel Quantity

Due to the signal variety the inputs and outputs are normalised into a range of [0, 1].

For this input set the single NLARX structure shows similar results to the three-layer approach. Hence, the three-layer approach is neglected here due to the similarity of the results in comparison to the single NLARX approach. This provides evidence to the effect that a single NLARX structure is capable of predicting the full data scope and a task distribution for different data ranges is not necessary.

To summarise, the single NLARX approach provided the following results:

- Initial modelling with single NLARX structure [10 Hz data]
    1. Training $\mathrm{R}^2$= 0.99
    2. Validation $\mathrm{R}^2$= 0.96

Here the visual comparison of the single NLARX structure is shown in figure 4.22. The blue line represents the desired normalised opacity output whereas the bright green dashed line shows the model output for training and validation. In addition, a value-to-value comparison is plotted in figure 4.23 in order to provide a better overview of the comparison.

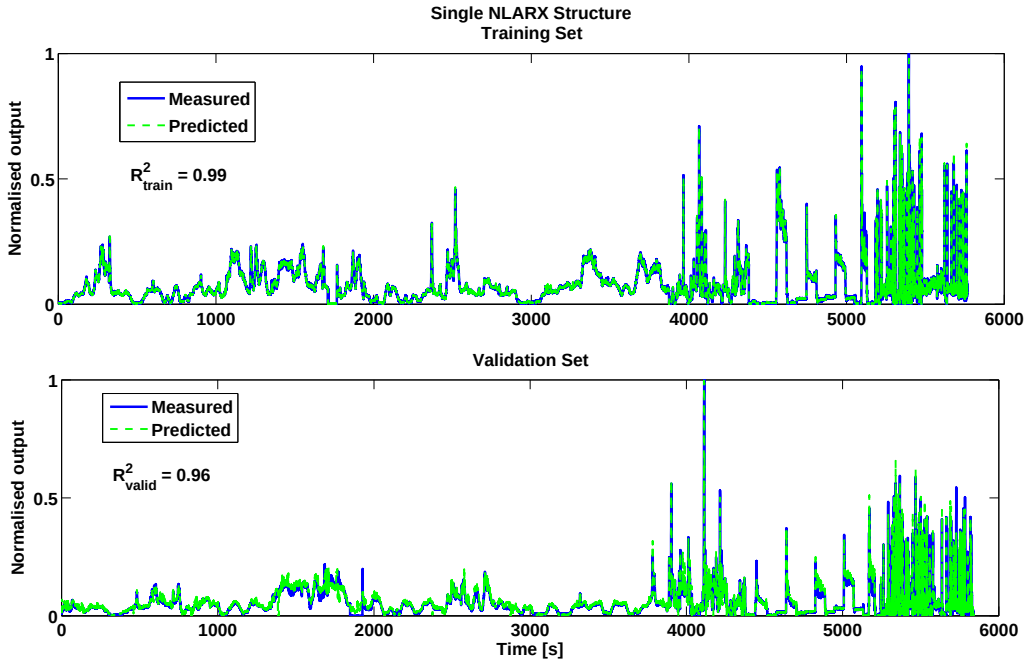


***Figure 4.22:*** *Comparison results for nine-input NLARX structure with data sampled at 10 Hz; Training comparison:* $\mathrm{R}^2$ *= 0.99 - Validation comparison:* $\mathrm{R}^2$ *= 0.96*

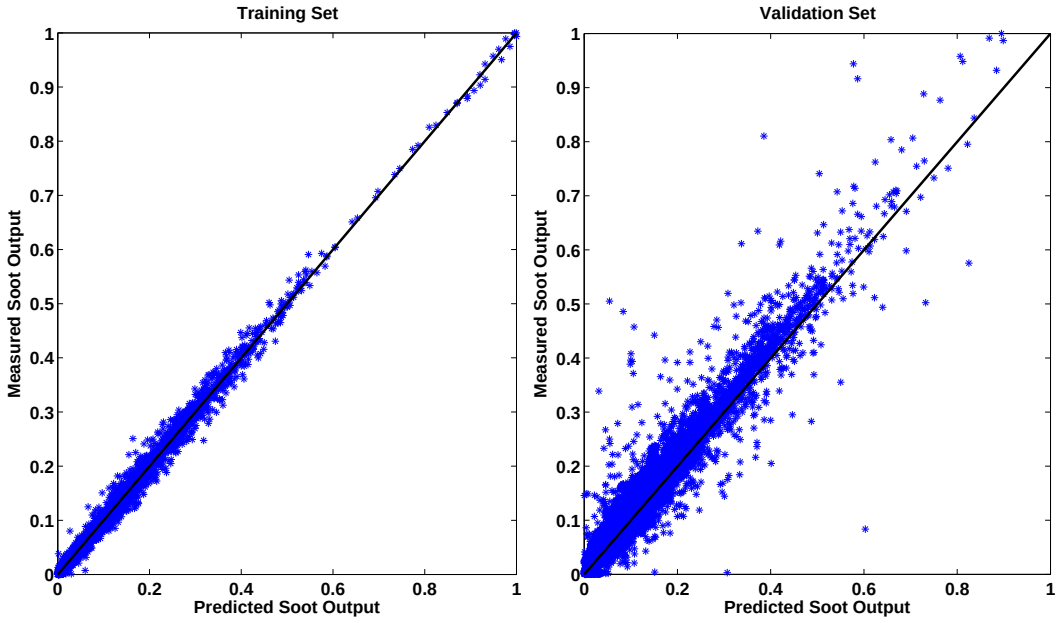


***Figure 4.23:*** *Value-to-value comparison results for nine-input NLARX structure with data sampled at 10 Hz; Training comparison:* $\mathrm{R}^2$ *= 0.99 - Validation comparison:* $\mathrm{R}^2$ *= 0.96*

This result leads to the conclusion that for a sufficient information density a higher sampling rate is required together with additional fuelling information. In the following, investigations

are conducted into reducing the number of inputs and data samples. The idea behind reducing the number of inputs is to find a less complex network structure for faster training and less costly data processing of sensor signals.

**PCA Pre-Processing for Inputs for Model Identification** The inputs, influence on the model are determined with a Principal Component Analysis (PCA) applied to the set of nine inputs. This leads to a possible reduction by four inputs down to five crucial inputs for sufficient predictive accuracy of the model. The PCA predicts the influence of each input on the systems behaviour. The following list shows the ranking for the PCA result on the 9 inputs chosen initially. The PCA functionality of the MATLAB's Neural Network Toolbox was used in order to determine the principal components. The PCA method is based on determining the maximum signal variability by subtracting the mean value for each input signal and creating a zero mean signal. A next step incorporates the calculation of the covariance matrix before the eigenvectors are found. Based on those eigenvectors the input signals are ordered with the largest variation first and least variation last. In this case the variability is set to a 60 $\%$ threshold for the principal components i.e. every component with less than 60 $\%$ variability is neglected. The inputs PC6, PC7, PC8 and PC9 were dropped in order to determine the performance change without their influence. As shown in the following figures, the performance improves. This effect can be defined by reducing "waste" information from inputs neglected. These inputs may contain information that does not relate to the output, in turn making it difficult to find an optimal solution.

1. PC1 – Air-to-Fuel Ratio
2. PC2 – Speed
3. PC3 – Torque/Load
4. PC4 – Exhaust Manifold Pressure
5. PC5 – Common Rail Pressure
6. PC6 – Boost Pressure
7. PC7 – Fuel Quantity

8. PC8 – Intake Manifold Temperature

9. PC9 – Mass-Air-Flow

The result for a five-input model can be seen in 4.24 and 4.25.

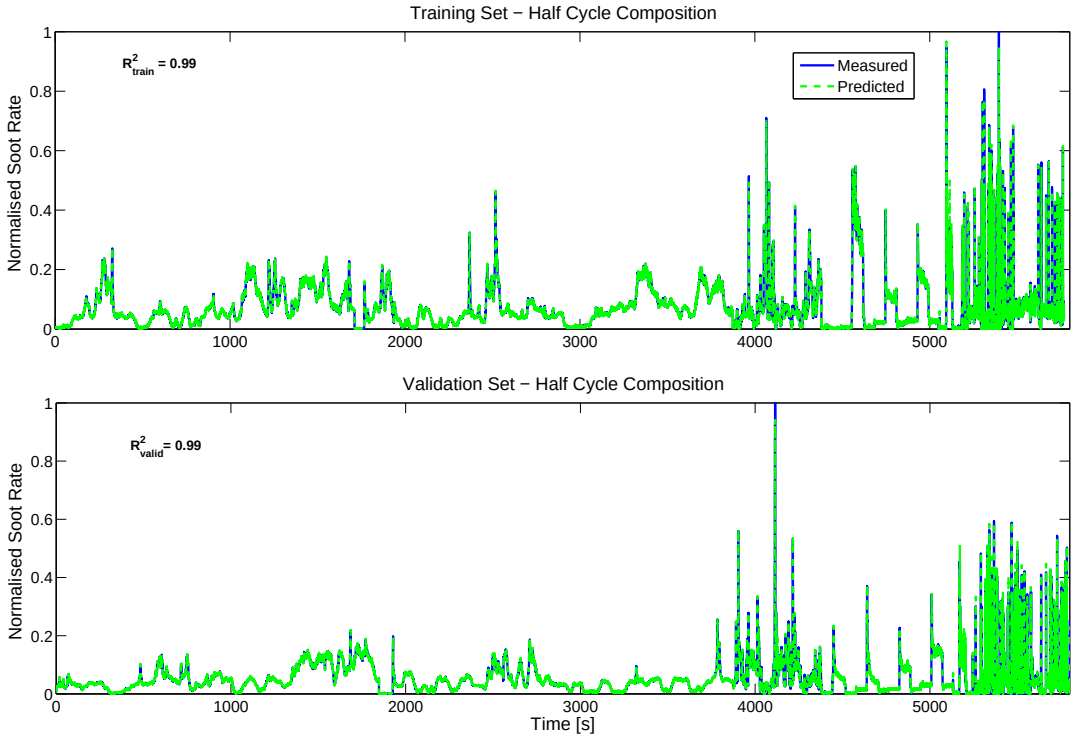


***Figure 4.24:*** *Comparison results for five-input NLARX structure based on the results of principal-component analysis with data sampled at 10 Hz; Training comparison:* $\mathrm{R}^2$ *= 0.99 - Validation comparison:* $\mathrm{R}^2$ *= 0.99*

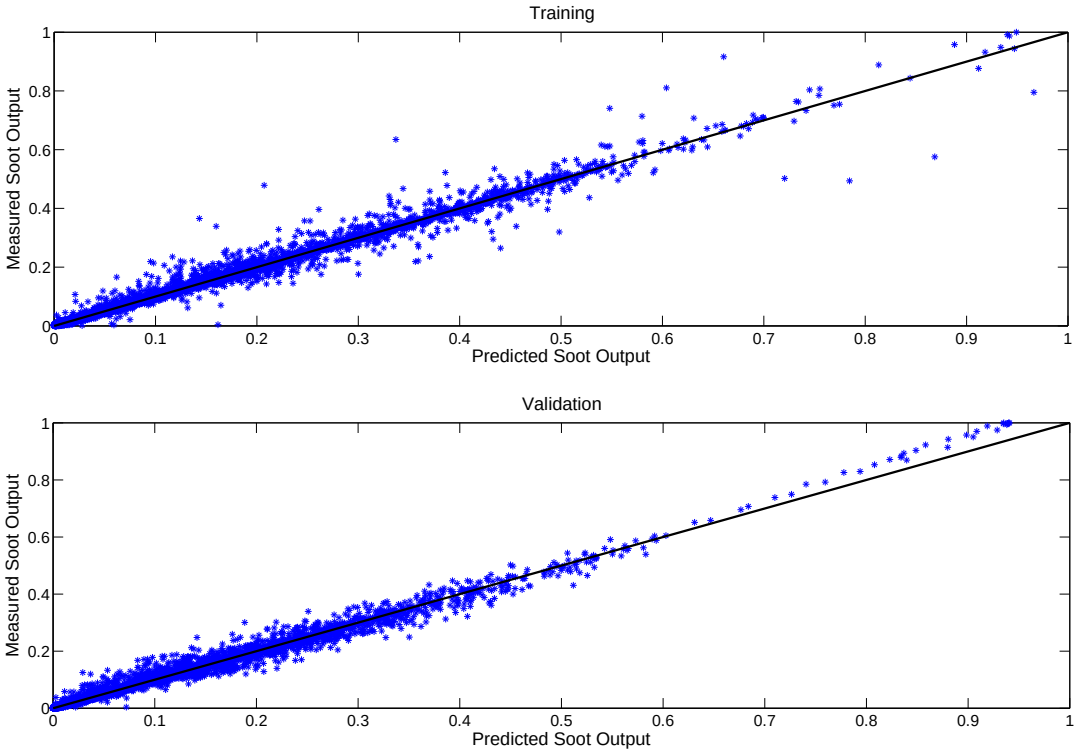


***Figure 4.25:*** *Value-to-value Comparison results for five-input NLARX structure based on results of principal-component analysis with data sampled at 10 Hz; Training comparison:* $\mathrm{R}^2$ *= 0.99 - Validation comparison:* $\mathrm{R}^2$ *= 0.99*

Training and validation results for five inputs:

1. Torque
2. Speed
3. Air-to-Fuel Ratio
4. Exhaust Manifold Pressure
5. Common - Rail - Pressure

Results:

1. Training $\mathrm{R}^2$=0.9961
2. Validation $\mathrm{R}^2$=0.9961

**Conclusions and summary** The results show a sufficient predictive accuracy of the NLARX structure based on five inputs. The investigation also confirms the importance of the choice of inputs for the correct representation of system behaviour. It is shown that the previous findings of using a three-layer network structure can be reduced in complexity by identifying the correct inputs. Inputs with little impact on a system's behaviour may contaminate input information and create more complex relations, making it difficult to find optimum network training points. The inputs also show an image of the engine parameters that are directly related to the behaviour of formation of soot during the combustion process. Torque and speed have a comprehensive expression capability for many engine conditions. Hence, they showed the highest values within the PCA. The three other parameters help to define certain operating conditions that are known to favour soot formation. Low air-to-fuel ratios indicate excessive fuel entrainment which may cause unburned carbon and, consequently, increased soot rates.The exhaust pressure reflects on the possible after-burn and oxidation processes within the exhaust part. A considerable number of chemical reactions occurred after the exhaust gas left the cylinder environment, meaning that the exhaust manifold pressure is an indicator for the conditions within the exhaust system. The fifth input parameter as a result of the PCA is the common - rail - pressure. The injection pressure considerably affects the break-up of the fuel jet and an improved break-up can result in reduced droplet size and hence the risk of soot formation. The methodology presented can be applied to find the correct set of inputs for model identification.

## 4.3 Neural Network Modelling for Fuel Path Control Design

The previous section showed the practicability of neural networks in the emission modelling and hence in the complex and non-linear, highly dynamic field of engine behaviour. As outlined earlier, emissions regulations are progressively being enforced to meet increasingly and more stringent targets. In addition, fuel economy is a driving factor in engine development in order to meet customers' expectations and reduce running costs. A method of reducing these parameters in diesel engines, especially passenger cars, has been the multi-pulse injection technology. This is, increasingly applied in the field of medium and heavy-duty diesel engines. Along with the rising number of injection events, the dynamics of fuelling become more and

more complex. Look-up tables (LUT) are still in place with modern production diesel engines which define a setting for certain engine modes through a system of listings. These have been calibrated for optimised emission and fuel economy, often manually. Finding the optimum behaviour for a closed-loop control in a system with a high degree of freedom such as with multiple and varying injection settings, is an increasingly difficult task. The optimisation of such a controller can be be investigated through a model-based approach. A real-time modelling approach can ease the design task by simulating the controller's behaviour over a range of modelled engine behaviour. Here, the model's accuracy is crucial.

The following section presents an extract of a technical paper presented at the SAE Congress 2010 by Deng et al. [82] with the title "Modeling Techniques, to Support Fuel Path Control in Medium Duty Diesel Engines". This paper proposes a state-space model for representing of the fuel-path dynamics within the control algorithm. In terms of controller design and validation, an ANN structure is created that acts as an engine plant model.

This structure is shown in figure where the ANN is represented through the engine model. The outputs of the model are fed into the controller which adapts the engine model inputs in order to find the best operating point for $\mathrm{NO_x}$. Amongst $\mathrm{NO_x}$ the designed ANN structure can predict compressor mass flow rate, exhaust manifold pressure, exhaust manifold temperature. Those parameters are based on the following model inputs: start timing of main injection, dwell between the injections, rail pressure and the fuel ratio between main and pilot injection.

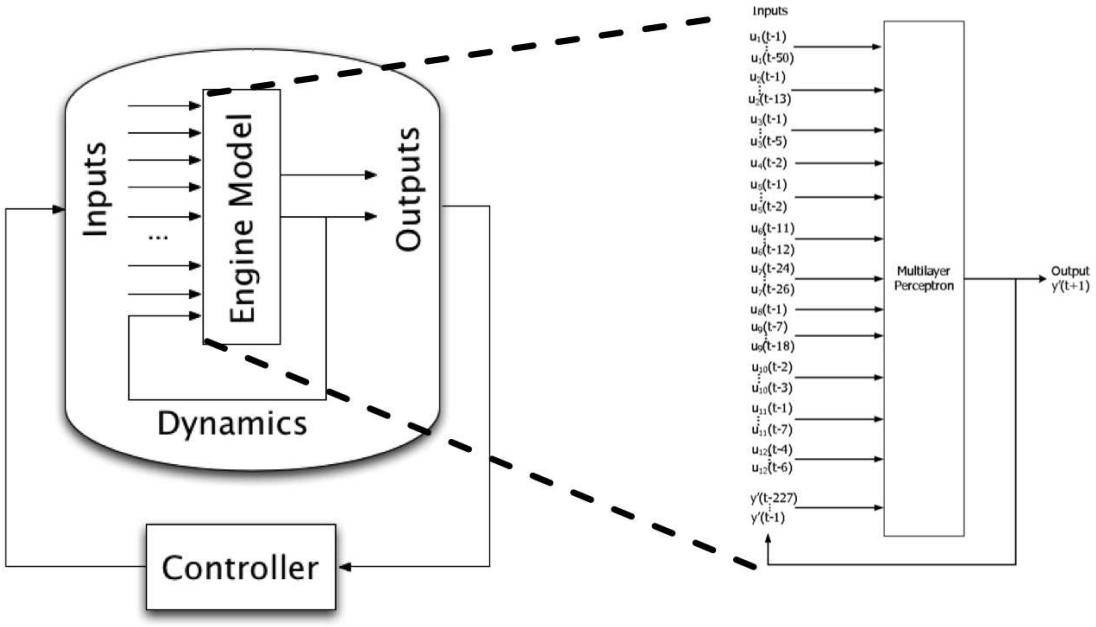


***Figure 4.26:*** *Real-time engine plant model for controller design*

**Data Generation** - The data required for training and validation of the ANN are recorded with a Caterpillar C6.6 heavy-duty off-highway engine. In order to capture a broad variety of features for the model calibration, the output parameters are recorded as a response of random step input signals as shown e.g. for the start of injection timing in figure 4.27. The signal is defined as a sequence of random magnitudes with sampling instants at a probability of $p$ - equation 4.3:

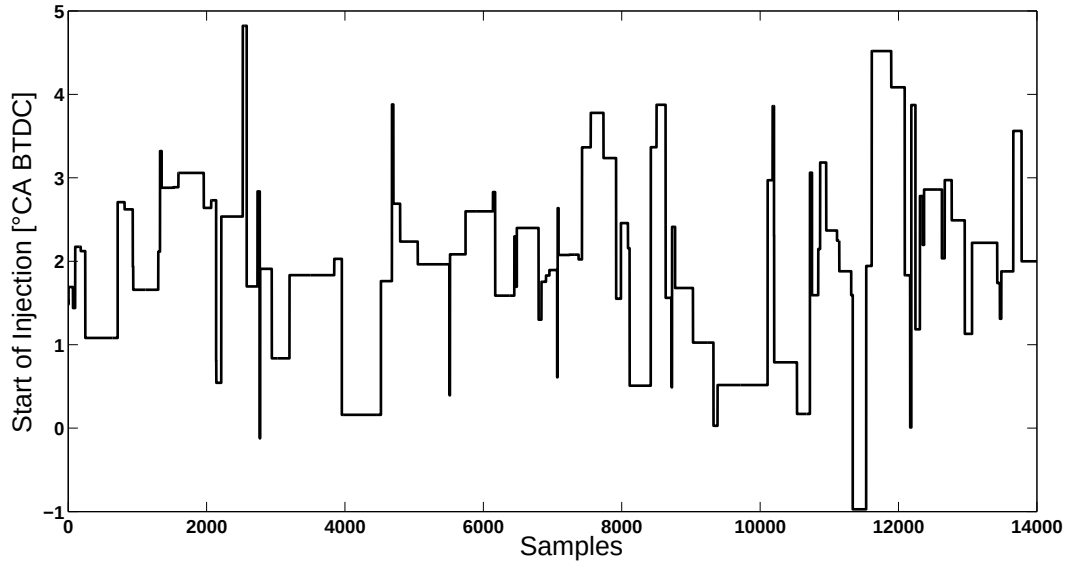


***Figure 4.27:*** *Random perturbation signal for start of injection for data generation*

$$r(k) = r(k-1) \quad with\ distribution \quad 1-p \tag{4.3}$$

$$e(k) \quad with\ distribution \quad p \tag{4.4}$$

where $k$ is an integer and $e$ is a discrete time noise process with zero mean and standard deviation. The signal magnitudes are designed to cover the whole range of fuel injection space for fixed speed and torque. This ensures a wide variety of features for this engine condition and takes into consideration the model's ability to interpolate between set points of calibration as well as its inability to extrapolate beyond the range of presented calibration data. Here, data acquisition is operated at 1440 RPM and 466 NM. The injection time ranged from -6° to 3° TDC, rail pressure was operated at between 45 MPa and 75 MPa, dwell from 0.4 ms to 0.5 ms and the fuel injected was distributed from 0.5 to 1. The training set consists of 2000 seconds and the validation set was logged for 2500 seconds while initially sampled at 33.3 Hz before being resampled at 10 Hz.

**Training and Validation Results -** To this end, the known NLARX structure is used in a parallel manner. Each output, exhaust manifold pressure, compressor mass flow rate, exhaust manifold temperature, and $\mathrm{NO_x}$ is predicted by an individually trained NLARX network.

In figure the design of the engine plant model is shown. The four inputs are combined in a vector before being fed into the individually trained network for the four outputs. Each output is predicted through a separate NLARX ANN.

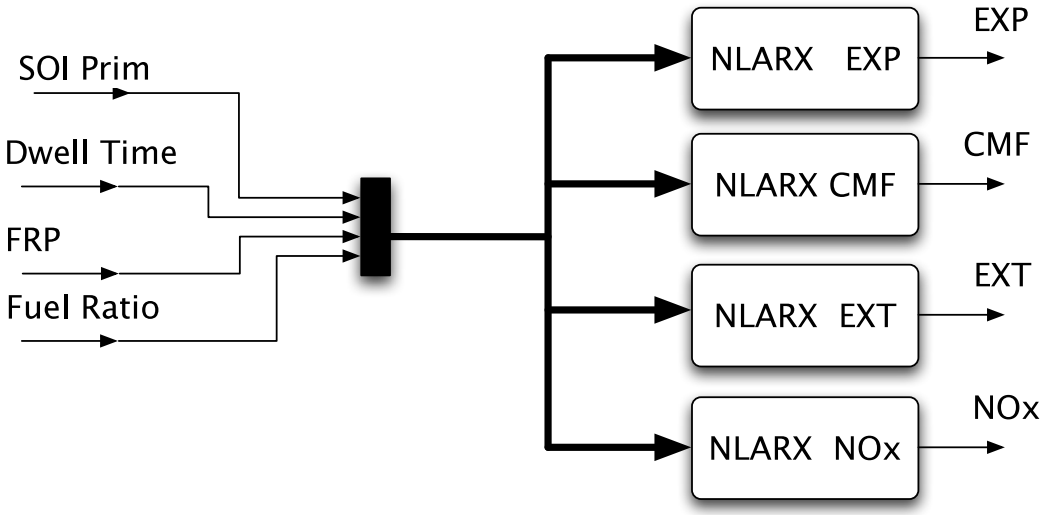


***Figure 4.28:*** *Neural networks for engine parameter prediction for engine fuel-path controller design*

The training and validation comparison results are shown in figures 4.29 to 4.32. It is shown the accuracy of the trained NLARX model in order to predict the corresponding output.

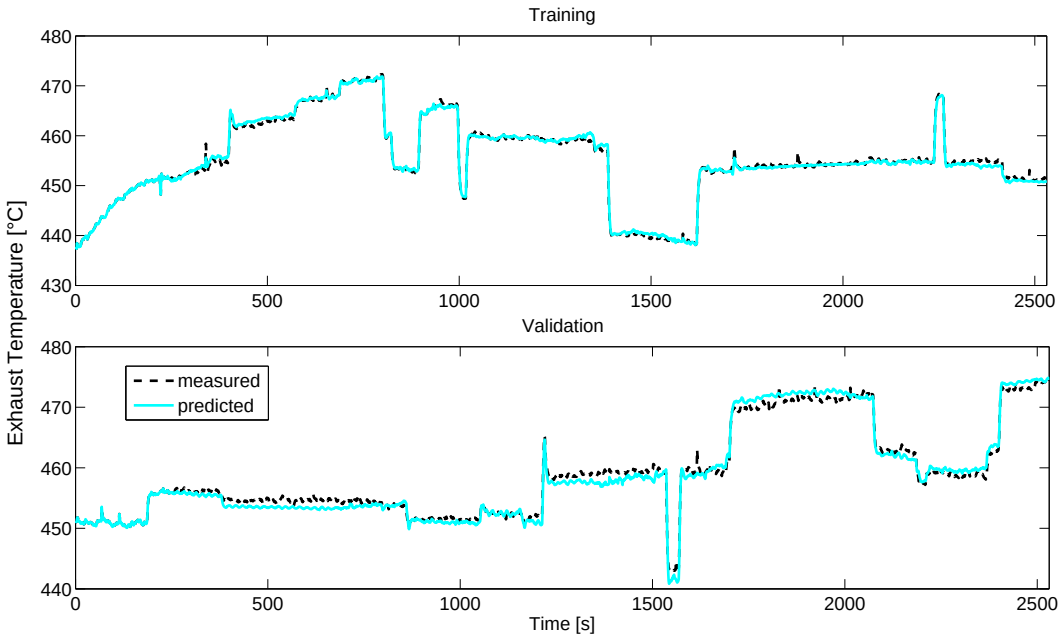


***Figure 4.29:*** *Comparison of exhaust temperature between measured and model-predicted output*

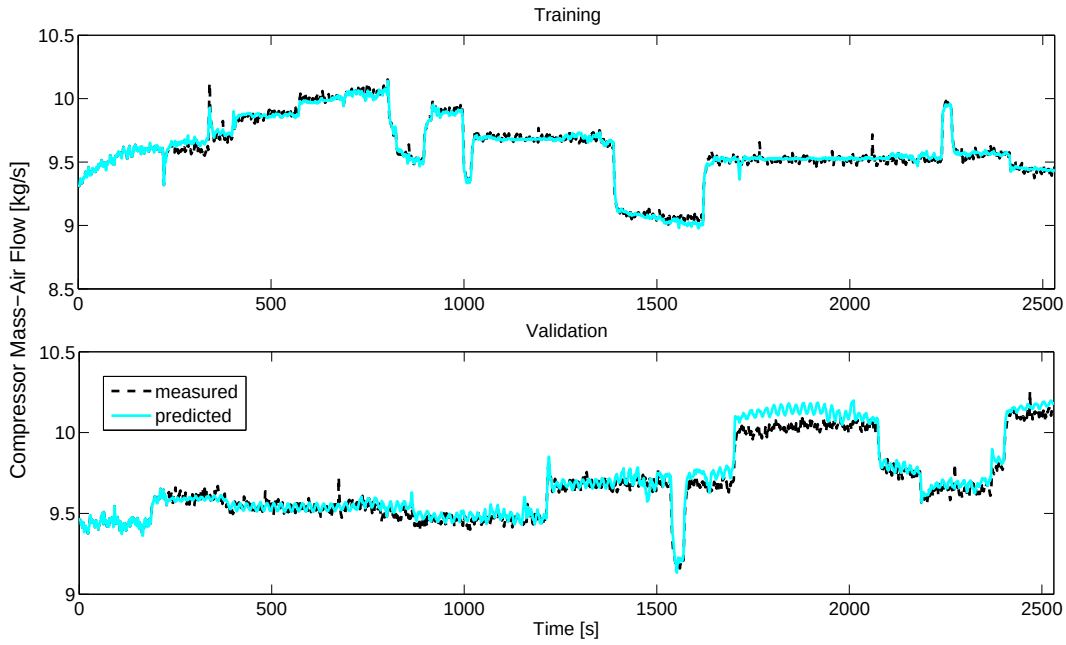


***Figure 4.30:*** *Comparison of compressor mass - air flow between measured and model-predicted output*

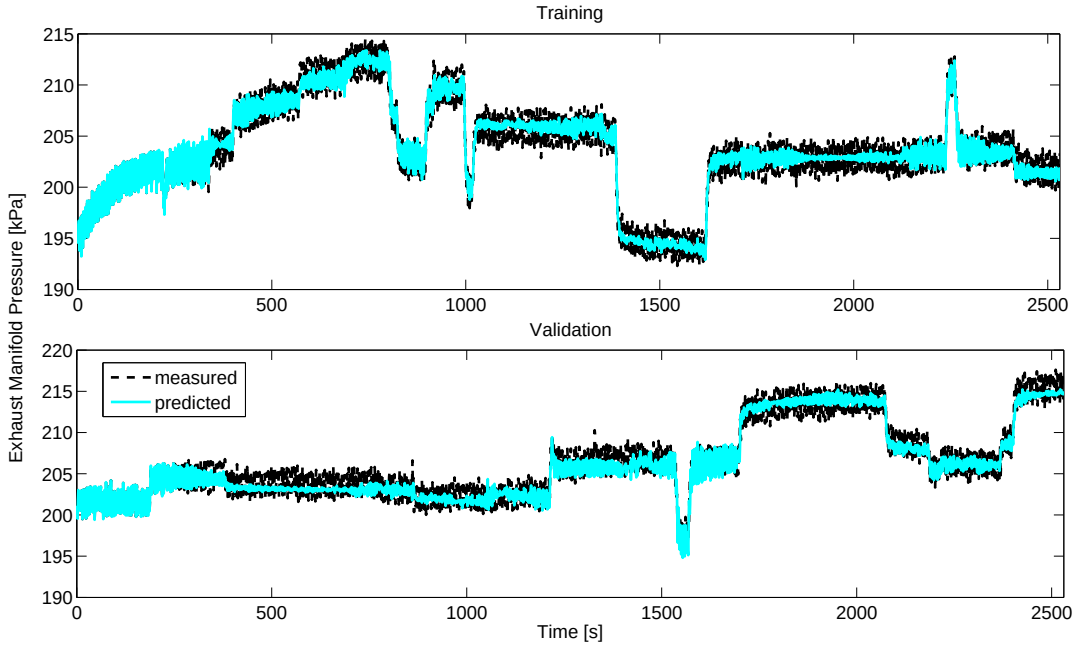


***Figure 4.31:*** *Comparison of exhaust pressure between measured and model-predicted output*

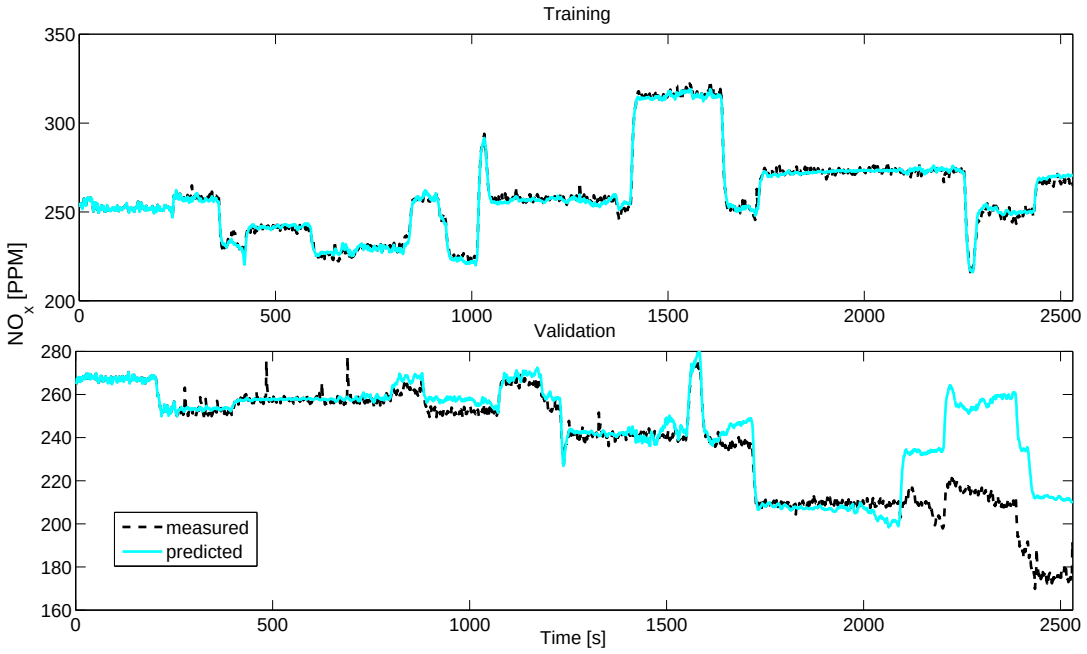


***Figure 4.32:*** *Comparison of* $\mathrm{NO_x}$ *between measured and model-predicted output*

These models are used in order to design a control algorithm which is based on state space models predicting each of the outputs. They show also sufficient accuracy and can be used

in the application of model predictive control in order to define a stable horizon for control parameters.

Table 4.2 shows the comparison between performance results for the neural networks and the state space model. It can clearly be seen that the neural networks show a better modelling capability and close comparison with the system's behaviour. Hence, they are used as an engine plant model in order to design an control algorithm based on state space models.

| | Output | $R^2$ Training | | $R^2$ Validation | |
|---|---|---|---|---|---|
| | | NN | SS | NN | SS |
| 1 | EXT | 0.99 | 0.75 | 0.99 | 0.68 |
| 2 | CAF | 0.99 | 0.74 | 0.99 | 0.66 |
| 3 | EXP | 0.99 | 0.79 | 0.99 | 0.72 |
| 4 | $NO_x$ | 0.99 | 0.69 | 0.99 | 0.72 |

***Table 4.2:*** *Comparison of ANN and SS performances*

**Conclusion** - In this work an experiment design is presented that enables random creation of engine operation points. By randomly changing engine calibration settings a response of the engine is provoked. This approach requires knowledge about the system boundaries in order to avoid critical scenarios with potential engine failure. In this example, the results are achieved from a single speed and load operation point. Additional data would be required in order to be able to predict the networks' capability more comprehensively. Nevertheless, the type of ANN presented here shows good basic capabilities to support the controller design. Due to its superior prediction performance in comparison to the steady-state model that was also tested, the controller design can be more accurate. Trends and characteristics can be tested more comprehensively within the controller development.

## 4.4 Conclusions

This chapter outlined the investigation of several applications of ANN in the field of engine parameters related to the combustion process. It has been established which model structure is appropriate for the prediction of non-linear parameter characteristics. In addition, the sections about emissions formation draw a conclusion about the importance of input choice and how to

reduce potential input lists that incorporate waste information. The work about $\mathrm{NO_x}$ formation also draws a conclusion on the importance of engine calibration settings and their impact on the network's predictive performance if changed. Each section emphasises the importance of training and validation data. The choice of test cycles for data generation is described in the work about emissions formation. In addition, the fuel-path control work describes an additional data acquisition procedure with a random signal generator. This creates a random variance of input parameters for coverage of engine conditions across the operation range. The fuel-path work also describes the implementation of an ANN structure as a plant model for controller design support purposes.

In this chapter the applicability of ANN is shown for engine parameters related to the combustion process. These findings are now applied to the detection of an applicable structure for prediction of in-cylinder pressure and temperature conditions based on a GT-Power simulation model and the previously mentioned Caterpillar C6.6 engine.

# 5 Data Acquisition and Generation

Data acquisition is a key element for successful modelling of system's behaviour. In the field of neural network modelling the training data is crucial for creating a good generalising network that covers a broad range of the system's behaviour. The previous chapter outlines the importance of the analysis of the system's parameter output range in correspondence to the input response. Hence, sufficient experiment design is key to successful neural network design. For efficient and yet sufficient training data generation, it is necessary to find the least required data covering the broadest engine operation range. This data set does not need to contain all different operating states as they can be generalised by the optimised network. They will, however, miss out extreme states in the operation map, which means a lack of training information. Neural networks generally cannot extrapolate states which are not covered by the training data as shown in subsection 4.2.4.

In addition, a design of experiment can be varied by pseudo-random signal generation for engine parameters. The variation of control parameters such as engine speed, torque, SOI, FRP or FR can be used to create different engine operation scenarios - see section 4.3. Depending on the parameter to be modelled, the operation makes a considerable difference. High transient load and speed changes can cause extreme soot output peaks as shown in subsections 4.2.3 and 4.2.4. On the other hand steady-state operation with increasing load can cause rising combustion temperatures resulting in excessive $\mathrm{NO_x}$ formation during diesel combustion - see subsection 4.2.2. Hence, the data generation for network training is highly dependent on the parameter to be modelled.

In case of in-cylinder condition acquisition such as the peak combustion pressure or temperature it is both difficult and expensive to record these two parameters. Although cylinder pressure recordings are available on some larger industrial diesel engines, the temperature within the combustion chamber is not measurable on common production engines. In order to overcome

this difficulty, a simulation model is applied and validated against the real engine. The software package GT-Power enables simulation of the missing parameter and generates the possibility of additional data acquisition in case of downtimes of the test cell equipment. Hence, two data acquisition systems are available:

1. GT-Power Engine Simulation Model of Real C6.6 Engine
2. Caterpillar 6-Cylinder 1106D Industrial Diesel Engine

This chapter describes the data requirements for in-cylinder pressure and temperature prediction. The chapter outlines factors that affect combustion and describes the experiment design along with the most influential parameters. In addition, the chapter outlines the two systems used for acquiring data: the GT-Power simulation model validated against a verified Dynasty 973 model and the C6.6 Caterpillar diesel engine.

## 5.1 Parameter Identification - Network Inputs and Outputs

The combustion process is dependent on several parameters. For the modelling process it is crucial to define the principal component parameters in order to generate a network of minimal complexity and to avoid correlation between input parameters. An initial approach is the physical understanding of the combustion process as described by the graph in figure 5.2. Certain parameters influence the initial conditions of the combustion process, while others control the start of combustion. This section lists the engine parameters which are considered for the neural network training and modelling of in-cylinder pressure and temperature. The understanding of combustion processes is based in this case on the corresponding literature on combustion - [81, 83, 84].

**In-cylinder conditions** The in-cylinder temperature and pressure describe the in-cylinder conditions during combustion. They follow the process of compression up to the moment combustion initiation where a sudden expansion of gases due to exothermic reactions causes pressure and temperature to increase as the compression ignition engine cycle is described in figure 5.1.

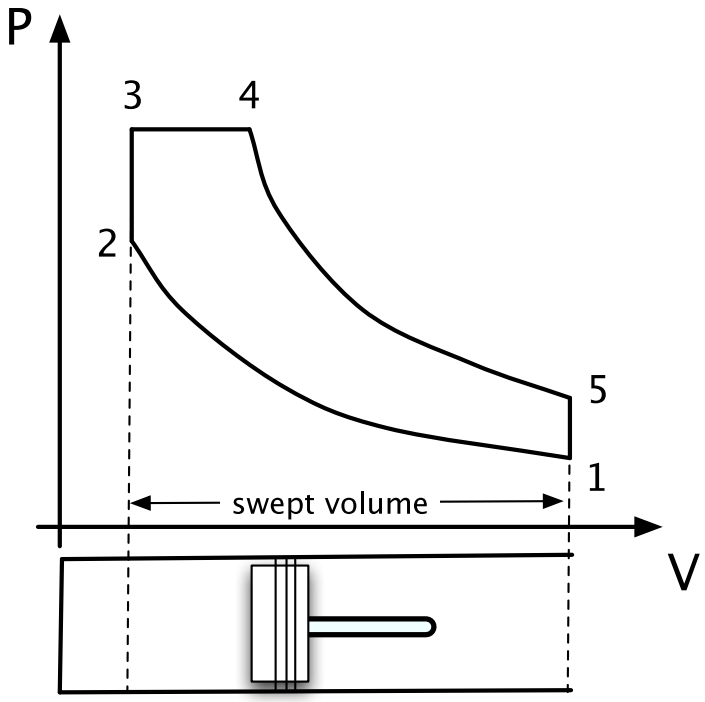


***Figure 5.1:*** *Ideal engine cycle pressure-volume diagram: 1-2 compression phase, 2-3 combustion - constant volume heat release, 3-4 combustion - constant pressure heat addition, 4-5 expansion and combustion abates, 5-1 gas exchange and pressure drop at valve opening*

The process of combustion is a result of pre-combustion conditions which are set by the following parameters.

**Mass - Air - Flow** The mass - air - flow determines how much air is made available for combustion within the cylinder and hence influences the quality of combustion. At the same time the air-flow has an impact on the initial gas density in the cylinder and therefore pressure and temperature development during the compression stroke. The mass - air - flow is dependent on the engine design. In turbocharged engines the air flow can be controlled more specifically to emerging engine operation needs.

**Intake Manifold Conditions** The intake manifold conditions can be described by the pressure and temperature. The manifold conditions have a considerable impact on the initial in-cylinder conditions prior to the intake valve closure. The higher the pressure, the higher the initial compression pressure. The higher the temperature, the higher the initial in-cylinder

temperatures. Both parameters affect the in-cylinder compression process and how fast combustion status is achieved. In addition, the intake manifold conditions are influential during the overlap period of intake and exhaust valve opening. In case of lower pressures in the intake system, exhaust gases can be dragged back into the combustion chamber and used to influence the gas exchange.

**Exhaust Manifold Conditions** The exhaust manifold is described by the temperature and pressure that predominate in this part of the system. High pressures caused by turbochargers may cause back pressures and hinder the gas exchange between cylinder and exhaust system. Temperatures in the exhaust are an indication of combustion temperatures and whether after-combustion is taking place in the exhaust system.

**Valve Lift Profiles** The intake valve closure defines the start of the compression and the rise of pressure and temperatures within the cylinder until combustion. The exhaust valve opening following the combustion terminates closed-cylinder conditions and defines the end of in-cylinder combustion. These two events mark the start and end point of combustion and are therefore crucial to system's behaviour.

**Injected Mass - Flow Profile** The mass flow profile defines the amount of fuel injected into the combustion chamber. The injection of fuel, usually defined as a flow rate during the injection period, includes information about the start of injection. It also defines the time period in which start of combustion will occur. Instead of the the mass flow profile, two other similar and closely connected events can be utilised. The injector activation current gives an indication of injection events such as start of injection or the injection duration. Another important parameter is the needle lift profile which shows the activity of the injection process. A raised needle indicates fuel flow. These parameters are crucial to distinguish between the compression process and the actual start of combustion. Depending on the mass flow profile, the combustion parameters are determined such as start-of-injection (SOI) or start-of-combustion (SOC).

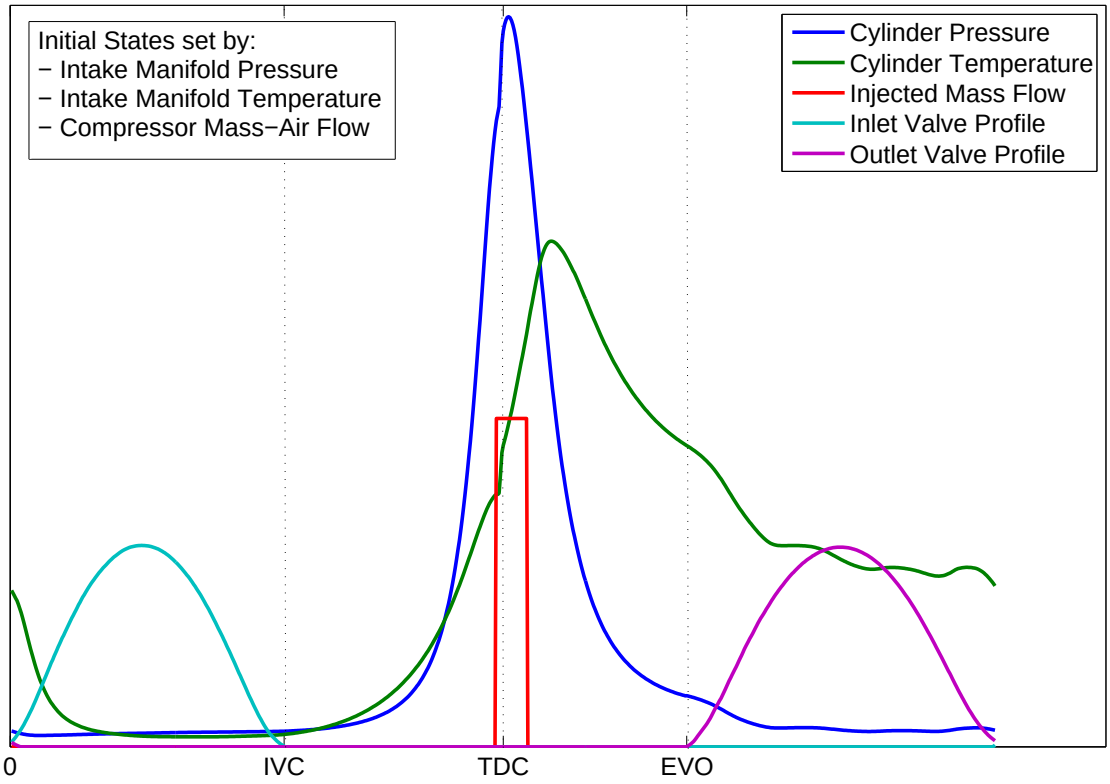


***Figure 5.2:*** *Exemplary combustion process events - 2200 RPM and 480Nm (70 %)*

## 5.2 Data Acquisition Systems

The acquisition of data for training and validation of the neural network is realised through two different approaches. The simulation software GT-Power from Gamma Technologies provides a platform for accurate data generation. In addition, a C6.6 medium-duty diesel engine is employed equipped with in-cylinder pressure sensors and additional injector measurements such as needle lift-activating injector current.

These two approaches are required due to the limitations in each method. Through the application of two different approaches these restrictions are partially overcome. The simulation only generates non-noisy data. Although the artificial introduction of noise signals can overcome this, the real engine environment is the benchmark for the neural network application. Hence, the modelling approach needs to be tested on real data generated during test cell operation.

On the other hand, the test cell engine cannot provide in-cylinder temperature data due to the difficulties in engine temperature measurements as described in the introduction in chapter 1. For this reason the simulation model is designed in order to provide this missing parameter. The simulation model in this case is validated against a verified model implemented in the software Dynasty. Engine implementation 973 is validated against the test cell. However, the Dynasty model is restricted, which makes it necessary to include the more comprehensive GT-Power software. The following section highlights the GT-Power implementation and the characteristics of the simulation model [85].

### 5.2.1 GT-Power Simulation Model

The GT-Power simulation tool is part of the GT-Suite from Gamma Technologies [85]. The model is implemented as a one dimensional simulation model. The model calculates an average for the flow direction of different engine parts such as pipes, valves or cylinder. Its predictive accuracy depends on the discretisation resolution of sub-volumes and how comprehensively the model designer defines the individual part specifications. However, the discretisation resolution also affects the computational cost of simulation and therefore it is a trade-off between accuracy and computation.

The present GT-Power simulation model has been validated against the independent Dynasty model implementation 973, which is fully validated against the real engine. This model serves as a data source for parts and calibration data for different load and speed cases. The calibration parameters and set parameter are listed in table 5.1. The calibration parameters are taken from the Dynasty model as reference points and are the validation values for the GT-Power implementation. The engine settings are also read from Dynasty and are used as control parameters for the calibration parameters together with engine part specifications.

The GT-Power model is set up from different template groups which require certain specifications on which the calculations are based. The following brief descriptions of the templates illustrates their main features. In some cases an individual calibration of a component or subsystem of the engine was needed. For further explanations of templates, refer to the GT-Suite user manuals.

***Table 5.1:*** *Calibration parameters for GT-Power engine simulation model*

| Calibration Parameter | Unit |
|---|---|
| In-cylinder pressure | $\mathrm{bar}$ |
| EGR flow | Fraction and valve opening |
| Intake conditions | Temperature and pressures |
| Exhaust conditions | Temperature and pressure |
| Turbocharger - | Turbine speed, compression ratio |
| Engine torque | $\mathrm{Nm}$ |
| Brake power | $\mathrm{kW}$ |
| IMEP | $\mathrm{bar}$ |
| Compressor pressure ratio | Fraction |
| **Engine Settings** | **Unit** |
| Engine speed | RPM |
| Injection timing | degrees BTDC |
| Injected fuel mass | $\mathrm{mg}$ |
| Injection duration | $\mathrm{ms}$ |
| Injection pressure | $\mathrm{bar}$ |
| Valve opening | CA$^\circ$ |
| Turbocharger maps | (Turbine and compressor) |
| Cooler outputs | $C^\circ$ |

**Pipes, Flowsplits and Valves** The core components of the engine model are the pipe segments, flowsplits and valves which determine the gas flow and directions. Each pipe segment specification is provided by the Dynasty model: dimensions, material and hence friction coefficients and heat conduction. In addition, the model takes into account discharge effects over valves, orifices or diameter changes in the piping. The valves for EGR and the bypass section are represented by throttle or butterfly valves. The calibration values of the pipes are mass - flows, temperatures and pressures which are correlated to the predicted values of the Dynasty template.

**Turbocharger and Compressor** The engine implementation realises a turbocharger with variable geometry turbine (VGT). The turbine operation in the model is controlled by mapping values describing the turbine aperture which is assigned through a turbine map. The turbine and compressor are connected through a shaft block that also enables the introduction of inertia and the matching of turbine and compressor speed. The operation of the turbocharger is validated and correlated against rotational speed and pressure ratios before and after the turbine and compressor wheels.

**Aftercooler and EGR Cooler** The cooler units are implemented by a pipe segment that consists of a number of identical pipes in parallel application to one another. This set-up acts as a heat sink and the desired outlet temperature is imposed by the pipes' wall-temperature that simulates the water-cooling behaviour. Another calibration parameter is the desired pressure drop over the cooler unit. Therefore the intake pressure and outlet pressure are correlated against the Dynasty model data. Two cooler units are simulated in this model: 1. Aftercooler for the intake-system cooling, 2. EGR Cooler for heat reduction of exhaust gases.

**Engine Block** Here, the term engine embraces the following components: manifolds, valve train, injectors, cylinders and crank train. The manifolds are represented by pipe and flowsplit components. Special caution is required with regard to mass flow in order to to achieve closest correlation with the calibration data. Variations in the flow characteristic can have significant effects on the charge cycle and consequently on the combustion process.

The valve train is represented by individual cam-controlled seat valve blocks with two inlet and two outlet valves for each cylinder. These blocks contain the valve geometry as well as the lift and flow characteristics for the particular valve which differs depending on intake or exhaust.

The current model incorporates a multiple injection strategy that enables up to four separate injections - pilot, postpilot, main and post. For each injection event, the fuel mass is either calculated in GT-Power or controlled through a Simulink controller (multi-injection control block). In the former case, the injected fuel mass and the duration are determined on the basis of the current in-cylinder pressure, cylinder crank - angle, engine speed and fuel rail pressure. Each injector is controlled through a command referenced to the first cylinder. Injector blocks also contain data for the injector parameters such as nozzle size and holes or the injected fuel.

Cylinder blocks define the cylinder geometry and the cylinder head as well as the piston shape and bore dimensions. In addition, the combustion model is referenced in the cylinder block. In this model a predictive direct-injection diesel combustion model is used that predicts the burn rate, pressures, temperatures and emissions formation - particularly $NO_x$.

The coordination of engine operation is defined in the crank train template where the start of cycle, the firing order and the TDC reference crank angle is assigned. In addition, it contains the type of engine, 4-stroke, and the cylinder configuration, in-line. The engine model structure is presented in figure 5.3 for visualisation.

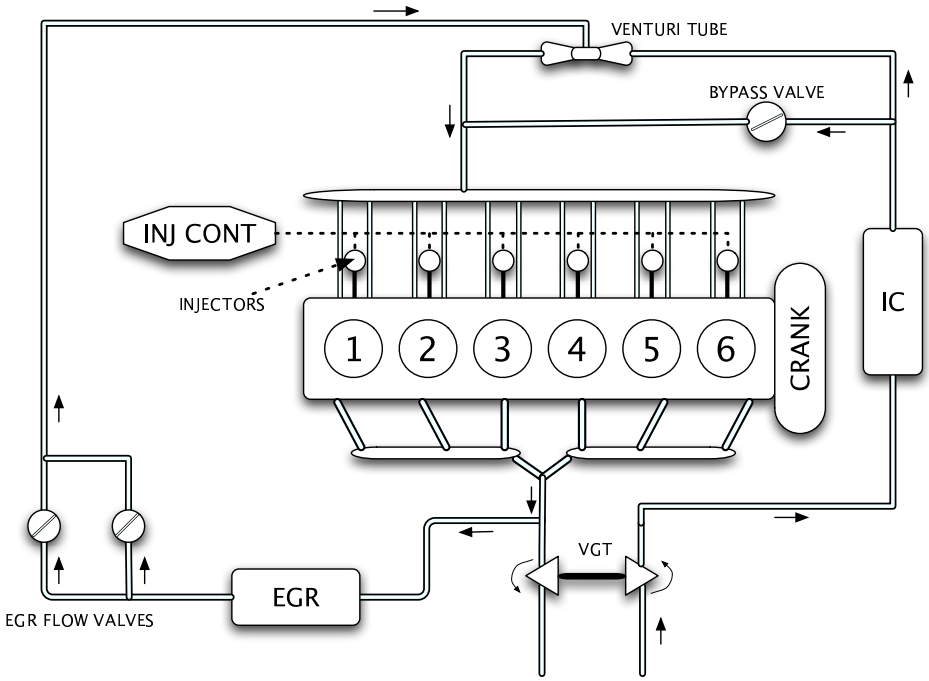


***Figure 5.3:*** *GT-Power Engine model. Map of the modelled engine parts*

Special emphasis was placed on the predictive combustion model that is specified in the cylinder blocks. The direct-injection jet combustion model requires calibration against simulated or measured cylinder pressure traces. For calibration purposes the model is fitted with a measured cylinder pressure trace for different operation points. Each of these traces is used for calculating the burn rate and other combustion parameters. This burn rate is then used to predict the in-cylinder pressure trace which is compared against another measured trace. Depending on the accuracy, several calibration parameters are monitored for error behaviour. The indicated mean effective pressure (IMEP) is an important calibration parameter. In addition, the characteristic of the pressure trace is investigated, along with initial pressure conditions at intake, intake valve closure (IVC), the available fuel and air masses and their ratio. For an accurate combustion model, the example pressure trace has to be correctly phased in order to achieve a close correlation between the predictive model and the desired signal.

As a measurement of accuracy of the current model, a pressure trace of the GT-Power model and the Dynasty model are plotted against each other in figure 5.4 and of a pressure-volume diagram is presented in figure 5.5. In addition the IMEP value is shown by way of comparison for the operation state at 2200 RPM and 685 Nm.

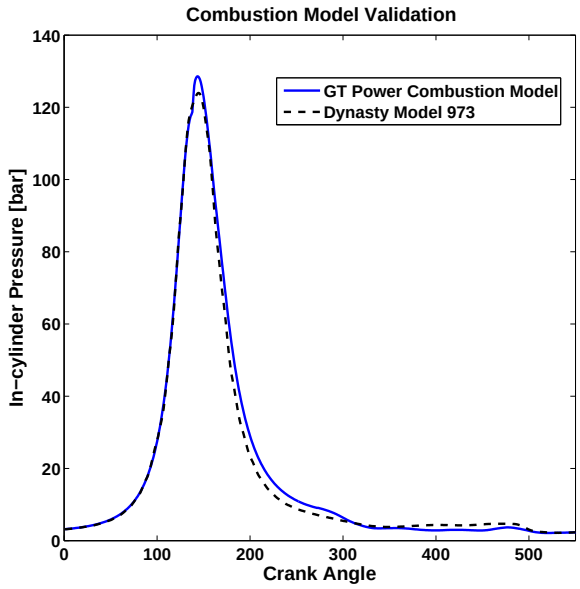


***Figure 5.4:*** *Pressure traces of GT-Power and Dynasty in Correlation of* $\mathrm{R}^2$ *= 0.99*

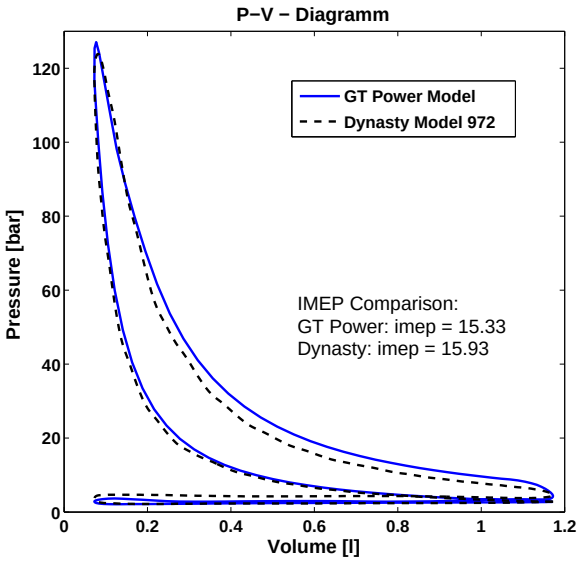


***Figure 5.5:*** *Pressure-volume relation and IMEP comparison between GT-Power and Dynasty*

The pressure trace and the P-V diagram show an overall correlation. There are slight differences in intake and combustion duration which shows the differences between combustion models. However, for the current application of data generation, these correlation characteristics are sufficient. The model can generate in-cylinder conditions, in particular pressure traces and temperature traces over the entire engine operation range as presented in figure 5.6 for low, medium and high-speed operation. It also shows close quantitative and qualitative correlation over the combustion process and correctly indicates events such as start of combustion and peak pressure. These variables are important, particularly with regard to the application of model temperature traces in combination with the real engine data. Firstly, the quantity of the prediction needs to be correct and accurate enough in order to determine the temperature from calculations based on the pressure trace. A correlation of up to 95% shows an acceptable range. Due to the fact that the temperature detection can only be an instantaneous spatial and temporal extraction of the combustion, this accuracy measure will give the correct trend of peak heat development. In addition, the quality of the signal such as characteristic events are required for training and subsequent correct prediction throughout the network.

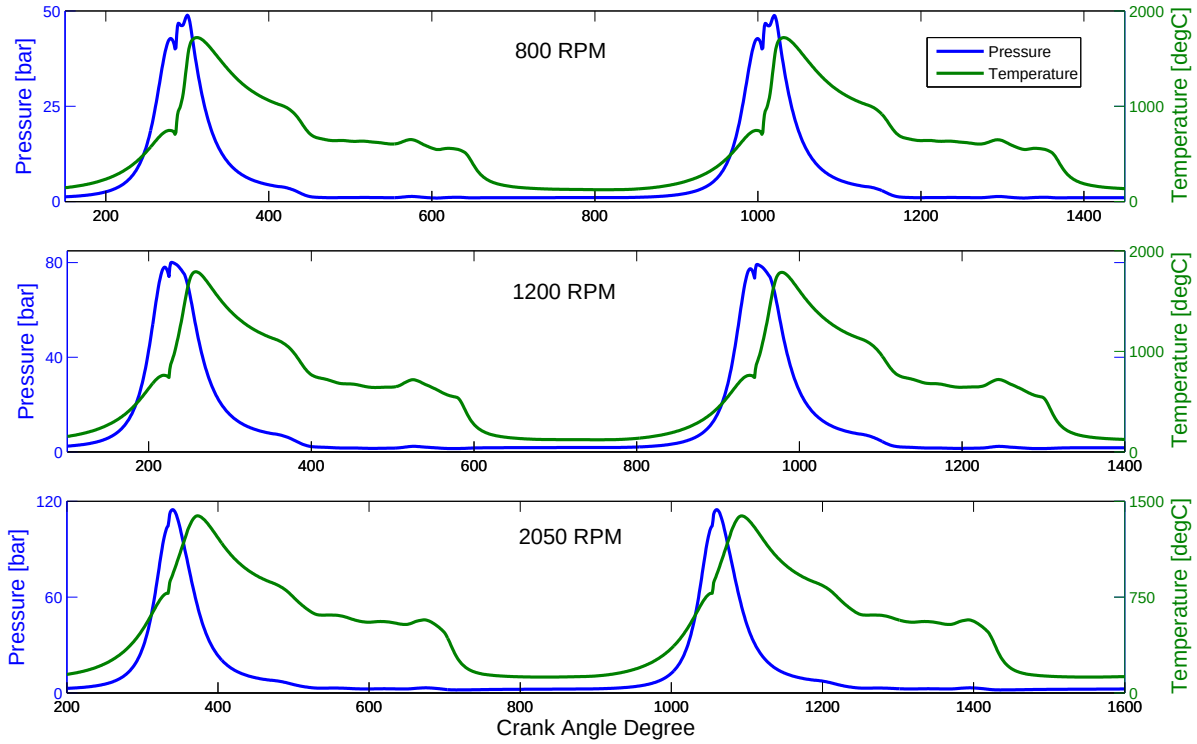


***Figure 5.6:*** *GT-Power combustion model output: pressure and temperature traces over low (800 RPM), medium (1200 RPM) and high speed (2050 RPM)*

### 5.2.2 Caterpillar C6.6 1106D Industrial Diesel Engine

The test engine is a Caterpillar six-cylinder, 6.6 litre industrial medium-duty diesel engine. Its specifications are listed in table 5.2. The engine is a Tier 3 engine that has been substantially modified to meet Tier 4 emissions specifications. These modificiations include modern advanced technology auxiliary systems such as a variable geometry turbine (VGT) turbocharger, an after- and EGR gas cooling system, an EGR control valve system with two separate controlled pathways and a throttle valve in the intake. The engine operates on direct fuel injection and was tested on different injector types and injection such as for example up to four injection events.

***Table 5.2:*** *Caterpillar 1106D Industrial HD Diesel Engine - Specifications*

| Descriptor | Value |
|---|---|
| Bore | 105 $\mathrm{mm}$ |
| Stroke | 127 $\mathrm{mm}$ |
| No. of cylinders | 6 |
| Displacement | 6.6 $\mathrm{L}$ |
| Cylinder arrangement | In-line |
| Type of combustion | Direct injection |
| Compression ratio | 16.2:1 (turbocharged/aftercooled) |
| Valves per cylinder | 4 |
| Firing order | 1-5-3-6-4-2 |

The engine is operated using the Cadet engine test system installed in the test cell. It allows the control of speed and torque via dynamometer control and high-speed data acquisition. The system supports the automatic run of transient schedules or stage-based testing with definition of setpoints, timings or test flows.

For research purposes the engine is fitted with an air- and fuel-path real-time control system that replaces the manufacturer's engine control unit (ECU). The original engine sensors are expanded with an independent sensor system comprising around 120 additional parameters within the engine and auxiliaries. Figure 5.7 shows the engine block with its original sensor locations, while Figure 5.8 shows the test cell with the engine arrangement. The engine head with sensor connections, intake and EGR system are visible.

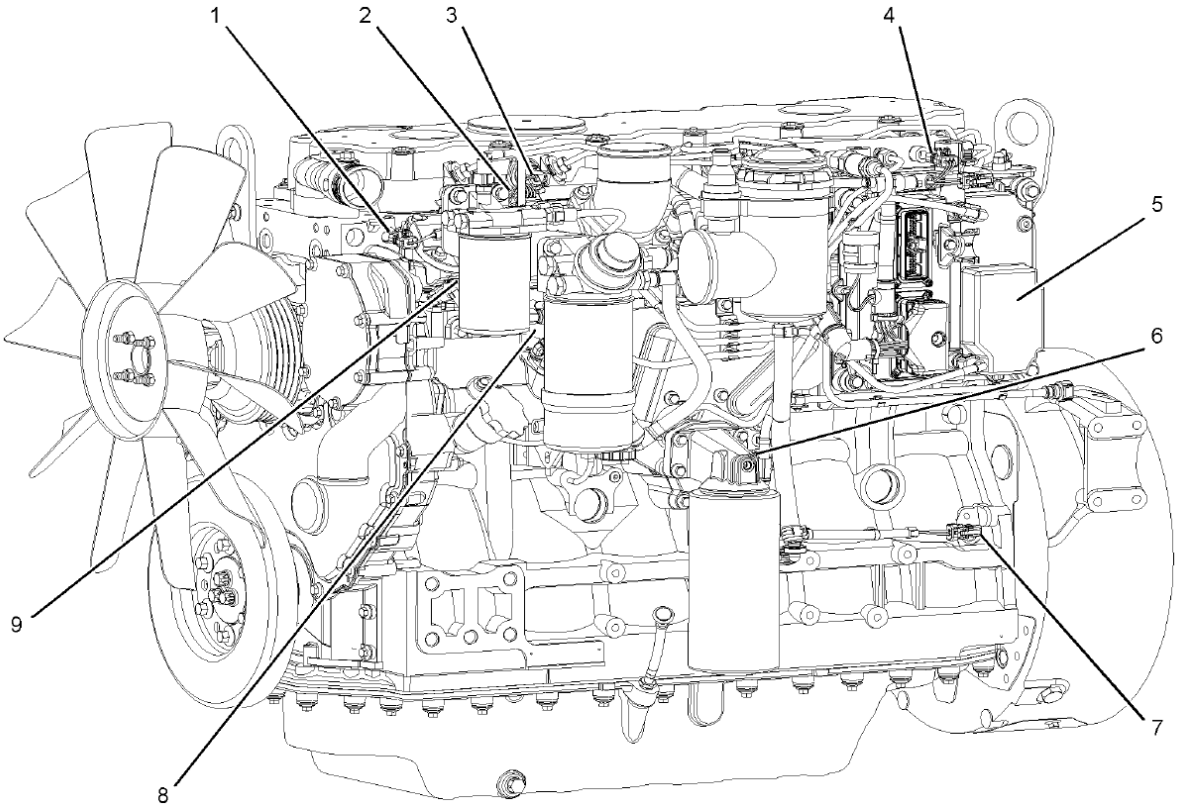


***Figure 5.7:*** *Perkins 1106D HD industrial diesel engine - sensor and ECM location: 1) Coolant temperature sensor, 2) Intake manifold temperature sensor, 3) Intake manifold pressure sensor, 4) Fuel pressure sensor, 5) Electronic Control Module (ECM), 6) Oil pressure sensor, 7) Primary speed/timing sensor, 8) Secondary speed/timing sensor, 9) Solenoid for fuel injection pump*

The data acquisition system designed for real-time control consists of an air-path control system run as an XPcTarget application. This extension of the MATLAB/Simulink environment allows control applications to be run on a commonly used personal computer in real-time and in parallel to the engine. In addition, a fuel-path control system is in operation via LabView which, similar to the XPcTarget application, enables integration with MATLAB/Simulink simulations. The current acquisition and control set-up has evolved this way and is currently under change. The data acquisition for the control operation is realised through a PXI/PCI-chassis. The schematic drawing in figure 5.9 explains the acquisition architecture.

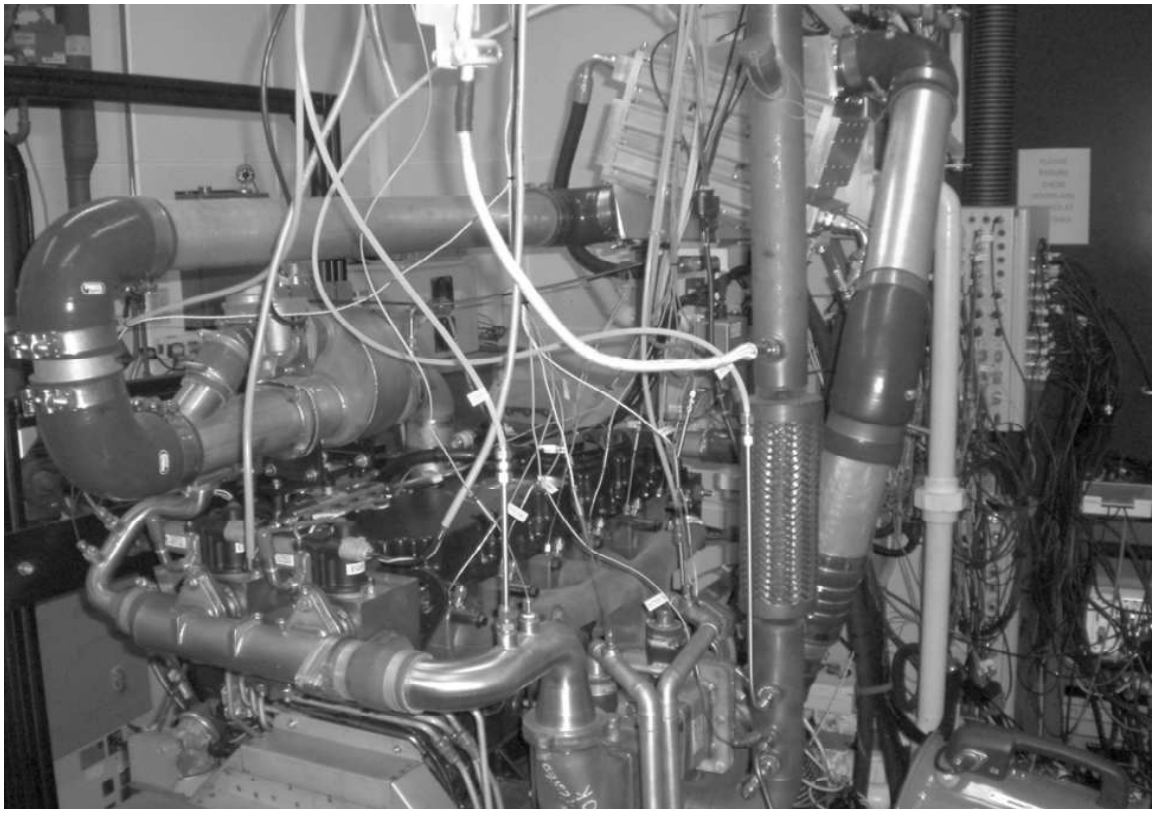

***Figure 5.8:*** *Test cell arrangement at Loughborough University for the Caterpillar C6.6 medium-duty engine.*

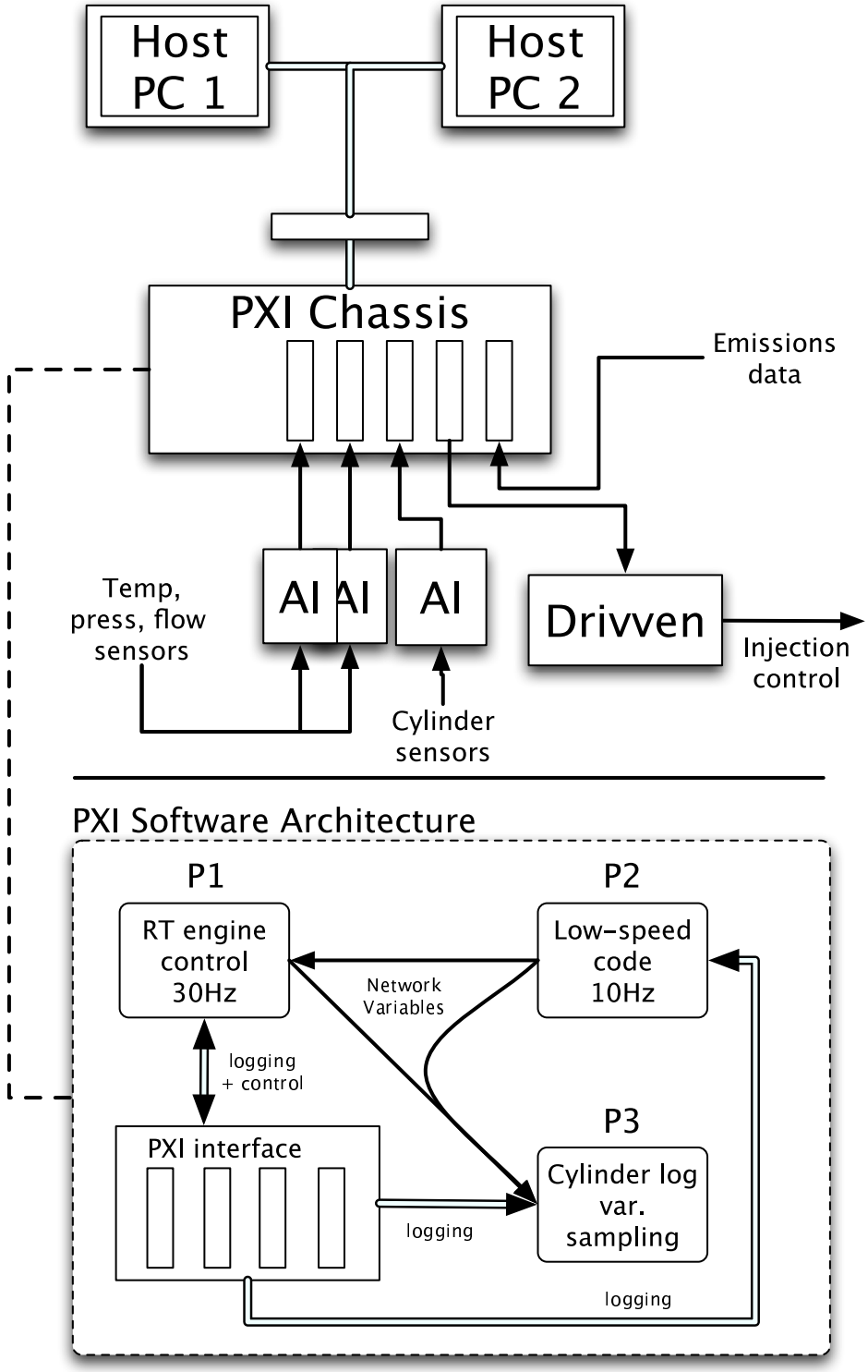


***Figure 5.9:*** *Schmematic top level representation of engine data acquisition and real-time control system*

The PXI system logs different variables at different sampling rates for the various applications. The real-time engine control logs parameters at 30 Hz (P1). For engine monitoring, the

low-speed code (P2) samples various data channels at 10 Hz for recording engine operation and behaviour. In addition, a third code (P3) samples the parameters required for in-cylinder acquisition at crank-angle resolution. Hence, the code can sample at different sampling rates and can be either triggered optically by encoder measurements or set to a fixed sampling rate corresponding to the operational engine speed. In the case of transient operation, the recording of in-cylinder pressure traces requires averaging or should not exceed a sampling window of 60 seconds due to vast amounts of data, especially at high-speed operation.

### 5.2.3 Engine Parameters Recorded

For both data acquisition applications the input identification applies differently. Due to restrictions in sensor availability at the engine, certain input information for the network structure is derived from available sensor readings. The two listings show the parameters recorded from the Caterpillar C6.6 engine and the parameters recorded from the GT-Power model.

***Table 5.3:*** *List of input parameters for ANN structures recorded from the GT-Power simulation and the Caterpillar C6.6 engine*

| Parameters from Caterpillar C6.6 engine | List of parameters recorded with GT-Power |
|---|---|
| Compressor mass air - flow | Compressor mass air - flow |
| Intake manifold pressure | Intake manifold pressure |
| Injector current | Intake manifold temperature |
| Needle lift | Injected fuel mass- flow |
| Exhaust temperature (port 1) | |
| Intake valve profile | Intake valve profile |
| Exhaust valve profile | Exhaust valve profile |

The sensor set-up at the C6.6 engine provides information about the intake manifold air - flow and the intake manifold pressure. The temperature sensor within the intake manifold does not provide a resolution high enough in order to detect temperature differences within the intake system. To define the compression and power stroke of the cycle, the intake and exhaust valve profiles are recorded as well. They serve as an indication for initiated compression and the end of the combustion process. These profiles are recorded from the C6.6 engine and the GT-Power model and are the same in both cases. For information about the combustion process, the actual start of combustion and the length of combustion can be derived from fuel injection information. The information required is incorporated in the injection profile. In the

GT-Power model the profile of injected fuel mass can be used in order to detect the start of injection and the actual mass injected. Hence, the information for the start of combustion is available as well as the duration of combustion due to the actual available fuel mass to burn. For the C6.6 modelling task a different set of parameters is available, which includes the injected fuel mass flow information used for the GT-Power modelling. The injector current measurable at cylinder 1 of the C6.6 engine can be used to define the start of injection and hence indicates the start of combustion. An additional parameter, the actual needle lift of the injector, is used as an indicator for the mass of fuel entraining the cylinder over a specific period of time. An additional parameter used for the engine modelling is the exhaust port temperature. This value is recorded at the exhaust port of cylinder 1, which is equipped with a better sensor than the intake manifold that enables the detection of event-based temperature rises such as exhaust valve opening. Hence, the sensor signal contains information about the in-cylinder temperature during the gas exchange which could be related to the combustion temperature.

The sensor signals used here are acting as network inputs and they all contain information of some relevance to combustion. The data is recorded in the crank angle domain due to the cycle-based origin of the process that is described on events such as crank TDC or BTDC. Hence, the crank position is a possible additional input defining the cycle events.

## 5.3 Design of Experiments and Data Generation

The acquisition of data for network training requires a certain operation range in order to be able to teach the network sufficient generalisation capabilities. The boundaries defining the extent of system operation also define the design of experiments. Insufficient scope of training data leads to failure of output prediction beyond the training data boundaries. Neural networks show good generalisation capabilities within the trained data range. However, outside this training range the capability of extrapolation is limited.

In the case of in-cylinder pressure and temperature, the boundaries of operation are set by the engine speed, load and the calibrated engine control settings related to fuel injection. A

change to those settings causes a variation of parameter behaviour and would require a revised version of the trained network as shown in the section about $NO_x$ prediction in 4.2.2.

The data generation for in-cylinder pressure and temperature is realised over the engine operation range described in a speed-torque map shown in figure 5.10. Specific data points are recorded. The initial data recorded from the GT-Power model covers seven speed cases and 21 load cases leading to 105 operation points. The load is increased in 5% increments from no load to peak load. Speed steps are: 800 RPM, 1000 RPM, 1200 RPM, 1400 RPM, 1550 RPM, 1800 RPM and 2200 RPM.

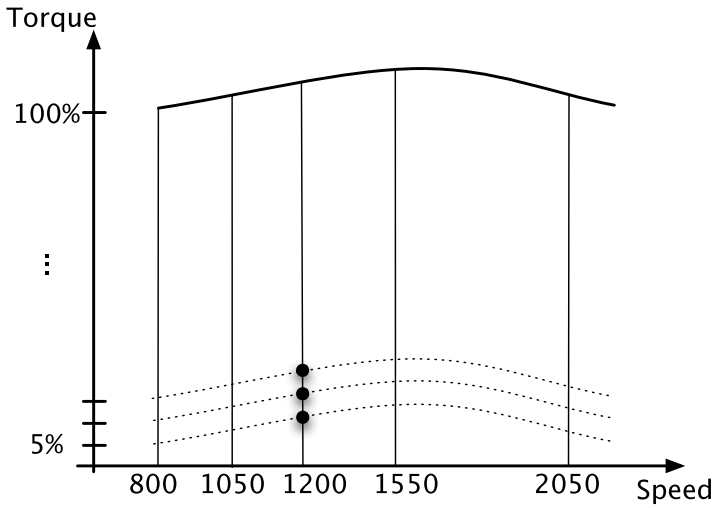


***Figure 5.10:*** *Design of Experiments (DOE) for GT-Power data acquisition: Speed-torque map for definition of engine operation range: 5 speed points with 21 load cases each - 105 operation points*

The real engine data is generated with a step load increase of 10% from zero to 70% load. Due to current system specification, the engine load was limited to 70% load application. The hardware installed on the engine was limited to certain operational ranges which did not allow engine loads above 70%. Operation above this limit would cause engine stall in certain scenarios due to loss of oil pressure or excessive emissions output due to fuelling errors. Hence, only data from zero to 70% load is considered. It is assumed that the data covers sufficient engine operation points in order to show the applicability of the presented method.

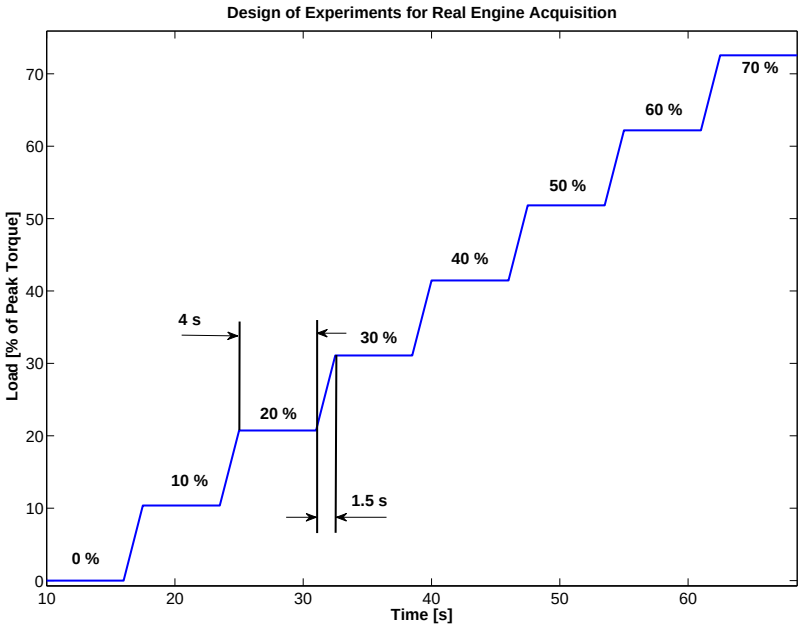


***Figure 5.11:*** *DOE: Real engine test with incremental load increase from zero up to 70% of peak load*

Figure 5.11 shows the experiments carried out on the real engine. The load is ramped up every 4 seconds for 1.5 seconds to the next load stage. In total the runtime is 68.5 seconds. Depending on the speed, this creates between 500 engine cycles for 800 RPM and 1300 engine cycles for 2200 RPM. Each data set contains steady-state cycles and cycles showing transient behaviour between load stages.

## 5.4 Conclusions

This chapter outlined the data acquisition requirements for the modelling task of in-cylinder conditions with ANN. The first section defines the engine parameters with the key impact on the combustion process. The parameters are recorded with two acquisition systems. A GT-Power simulation model that is built using a Dynasty simulation model. This model, in turn, is validated against a real Caterpillar C6.6 engine. The simulation becomes necessary for the generation of in-cylinder temperature data which is not accessible through real engine experiments. In addition, the simulation enables the fast and less cost intensive data generation for initial modelling purposes. The other acquisition system is a real Caterpillar C6.6 engine

that enables the high-speed data acquisition of cycle-based data. The engine is equipped with a cylinder pressure transducer and a comprehensive data acquisition system.

The chapter also describes the simulation and experiment procedures carried out for data generation. Due to hardware restrictions, the data is currently limited to 70% of peak load in the GT-Power application and the real engine. However, the key engine operation points are covered by a DOE. The data acquired is first used for an initial modelling with GT-Power data in chapter 6 before the network structures are tested on real engine data in chapter 7 along with transient and noisy data signals.

# 6 Modelling Results with GT-Power Generated Data

The results presented in this chapter show the performance of different network topologies on the prediction of in-cylinder pressure and temperature data. Three different network types are applied in order to detect the best possible network topology for the problem at hand.

The first structure is a simple multi-layer feed-forward network as described in 3.2. The second structure is a multi-layer feed-forward network with input time delays in order to include possible input dynamics. A third structure used is the NLARX recurrent network which has been described in 4.2.2, 4.2.3 and 4.2.4.

## 6.1 Cylinder Pressure Modelling with GT-Power Data

The cylinder pressure data from GT-Power contains the key inputs identified for this modelling task. The initial modelling approach for each network is realised with six inputs:

1. Compressor mass air - flow
2. Intake manifold pressure
3. Exhaust manifold temperature
4. Injected mass - flow
5. Inlet valve profile
6. Outlet valve profile.

The output is the in-cylinder pressure trace shown the of example in figure 5.2. The training data set is composed of 15 cycles that cover the engine operation range:

- Speed: 800 RPM to 2200 RPM

***Table 6.1:*** *Training set - speed and load scenarios*

| Speed [RPM] | Torque cases [%] | | | | |
|---|---|---|---|---|---|
| 800 | 0 | 20 | 40 | 60 | 70 |
| 1400 | 0 | 20 | 40 | 60 | 70 |
| 2200 | 0 | 20 | 40 | 60 | 70 |

- Torque: 0% to 70% of maximum rated torque.

The engine torque on the real engine is limited to 70% because restrictions are in place for certain operation scenarios. Hence, the GT-Power simulation is restricted to the identical range in order to match this data to the available data from the real engine - further explanations about specific restrictions are given in the next chapter 7 for real engine modelling. In addition, several points within the engine operation range are covered randomly. The training set covers the boundary speed, load points and additional 9 points, making a total of 15 cycles as shown in table 6.1.

The visualisation of the training set is presented in figure 6.1. It shows the in-cylinder pressure traces and the variance within different operation points. The cycles are arranged randomly and their cycle arrangement can be found in the appendix in tables B.1 and B.2 along with the actual load value for the corresponding cycle.

***Table 6.2:*** *Validation set - speed and load scenarios*

| Speed [RPM] | Torque cases [%] | | | | |
|---|---|---|---|---|---|
| 800 | 0 | 10 | 30 | 50 | 70 |
| 1200 | 0 | 20 | 40 | 60 | 70 |
| 1400 | 0 | 10 | 30 | 50 | 70 |
| 1800 | 0 | 20 | 40 | 60 | 70 |
| 2200 | 0 | 10 | 30 | 50 | 70 |

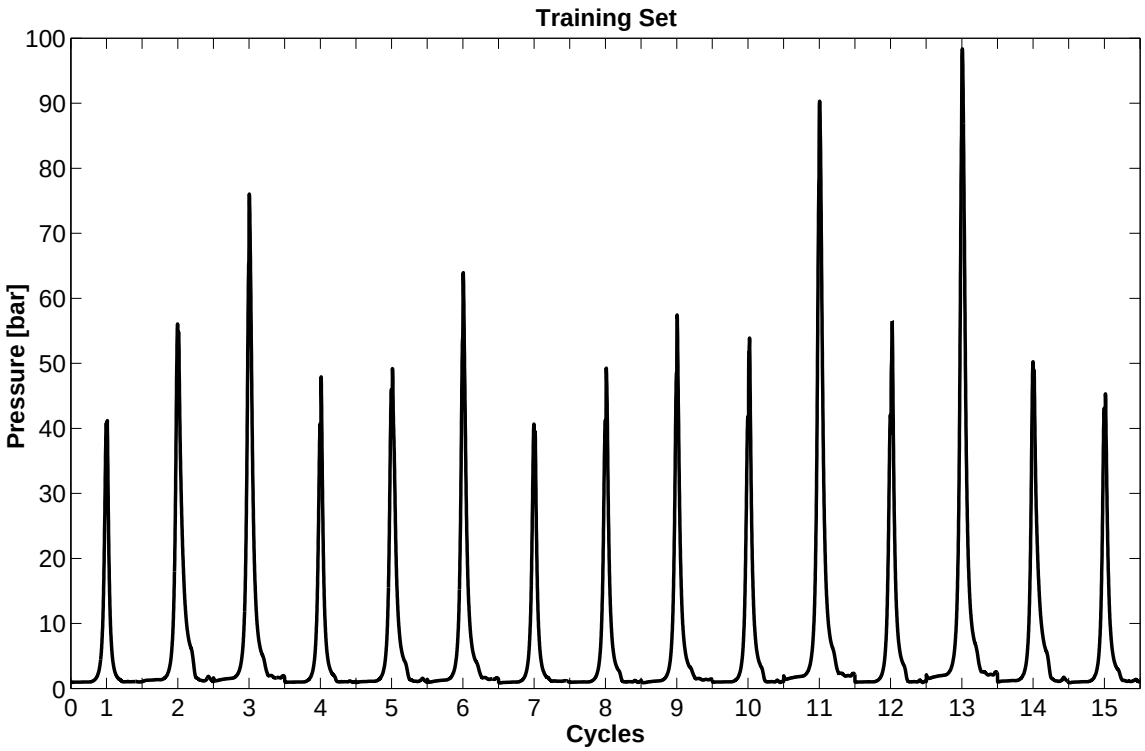


***Figure 6.1:*** *Training set for cylinder pressure modelling generated with GT-Power consisting of 15 cycles covering load scenarios at 800, 1400, 2200 RPM.*

The validation set consists of 25 cycles consisting of the scenarios shown in table 6.2 and figure 6.2. The current training and validation sets consist of steady-state cycle data.

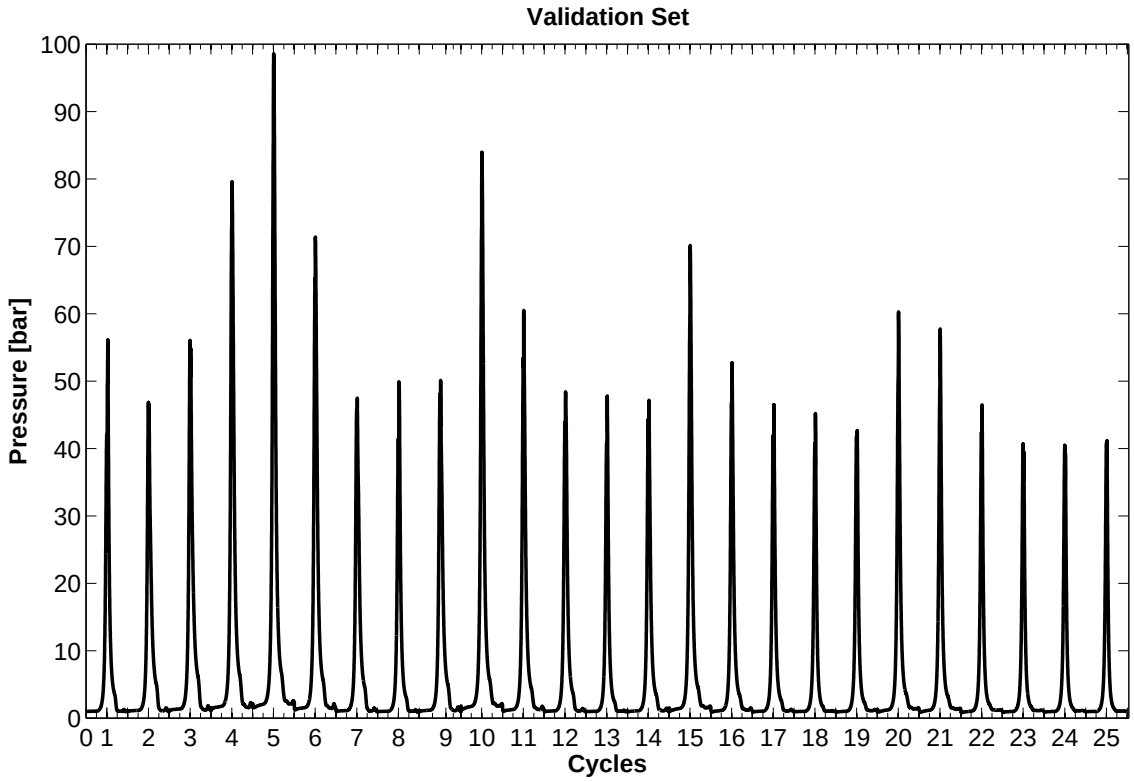


***Figure 6.2:*** *Validation set for cylinder pressure modelling generated with GT-Power consisting of 25 cycles covering load scenarios at 800, 1200, 1400, 1800, 2200 RPM.*

For each network type a corresponding set of results is presented. The network performance is measured with the comparison coefficient: coefficient of determination $\mathrm{R}^2$ as formulated in equation 4.1. In addition, a linearity check is plotted showing the value-to-value comparison between the measured and predicted output.

### 6.1.1 Network Training Approach

The goal of the network training is to find a topology for prediction of the in-cylinder pressure or temperature. As described in section 3, the topology consists of the network architecture, the number of layers and neurons, the assignment of transfer functions, and the training of the network weights and bias. For the latter the best performing points are found through a training algorithm as described in section 3.3. In terms of the transfer function, the literature [59, 62] states that within multi-layer networks the hidden layers usually are assigned non-linear functions in case of non-linear relationships with the output layer being assigned a linear

function. In this case, the logistic function defined in equation 3.5 is used consistently for the hidden layers and a pure linear function for the output layer. The number of layers and neurons is tested manually over a sensible and feasible range. The literature states that every neural network can be trained to map a differentiable function with a network that consists of no more than three hidden layers [86]. A network with up to three hidden layers and a sufficient number of neurons can be taught any non-linear differentiable function. Hence, the tests are limited to networks of up to three layers. At the same time the modelling is limited to available computational resources which restricts the number of neurons to 25 per layer. The memory of the personal computer in use cannot process networks any larger than this and hence the current training data set. The aim here is to find an efficient network structure for generating sufficient results in view of the comparison coefficient and the linearity check plot. The feed-forward networks without and with input delay is trained for three-layer topologies with 4, 10, 15, 20 and 25 neurons per layer. In addition, two-layer topologies are presented with 4, 10, 15, 20, 25 neurons per layer.

### 6.1.2 Multi-layer Feed-Forward Network Structure

The feed-forward structure is trained with a Levenberg-Marquardt algorithm. The training data is presented in 50 iterations to the network for training. Performance is measured via the mean squared error calculated from the difference between the desired output and the network predicted output. The smallest network topology with the best achieved performance is a six input network with three layers and four neurons per layer. With an $\mathrm{R}^2$ performance of 0.85 and 0.71 for training and validation respectively, the result is neither accurate nor sufficient. The visual comparison and the linearity check graphs in figures 6.3 and 6.4 show an insufficient comparison between the measured and predicted output. For better accuracy, an additional input signal is added that enables the network to relate information to periods after inlet valve closure through to outlet valve closure. The crank angle signal enables the comparison of engine cycle behaviour during the compression and expansion stroke. The injected fuel mass - flow indicates the moment of ignition and hence the start of combustion. However, the information for compression and expansion is missing. Adding the extra input enables the network to relate training input information to the training output information.

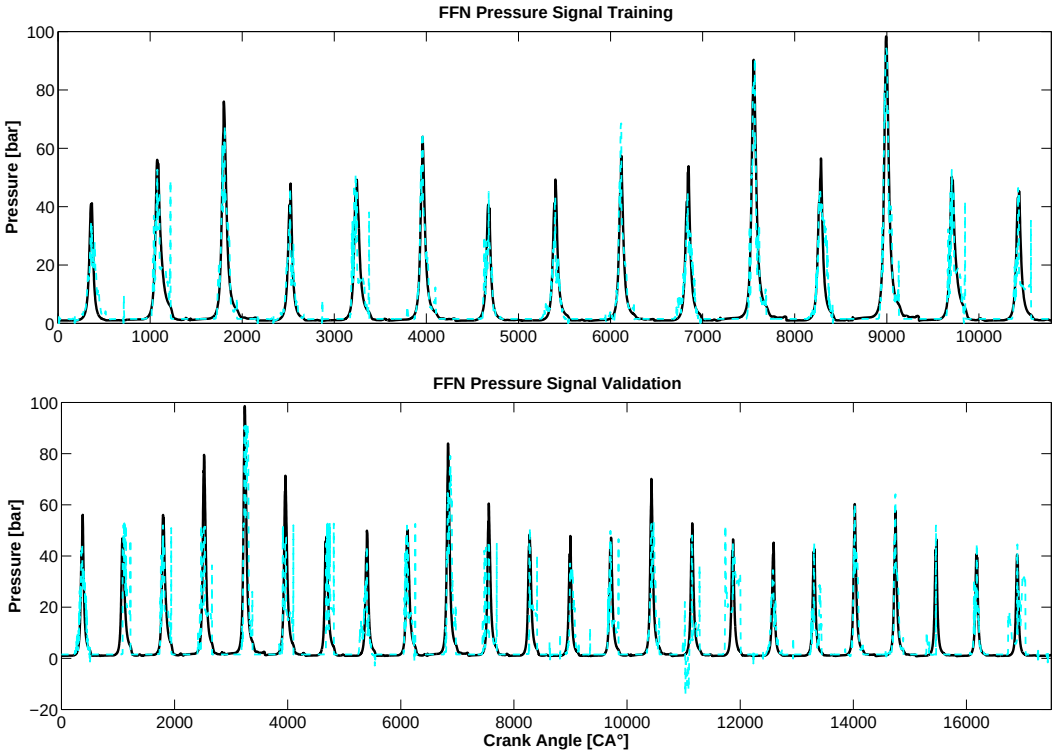


***Figure 6.3:*** *Comparison result for three-layer network [4 4 4] with six inputs: training* $\mathrm{R}^2$ *= 0.85 and validation* $\mathrm{R}^2$ *= 0.71*

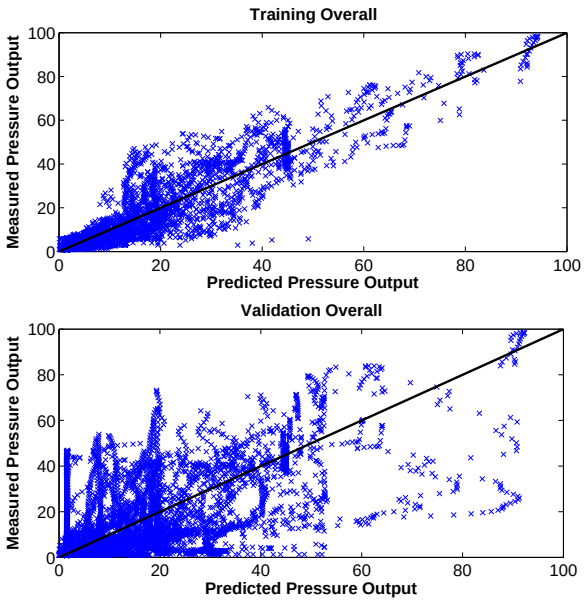


***Figure 6.4:*** *Value-to-value comparison along a linear plotline: training and validation set*

The result for seven inputs can be seen in figure 6.5. For this scenario the best result is found with a network of 10 neurons per layer trained to achieve an $\mathrm{R}^2$ of 0.99 for training and the validation set respectively. The visual comparison in figure 6.5 shows a good comparison of the training set. In the lower of the two graphs below, the validation set shows a close comparison. However, the network misses some peaks of the validation cycles. In addition, the linearity check plot in figure 6.6 indicates the closer comparison between measured and predicted values in comparison to the results with six inputs in figure 6.4.

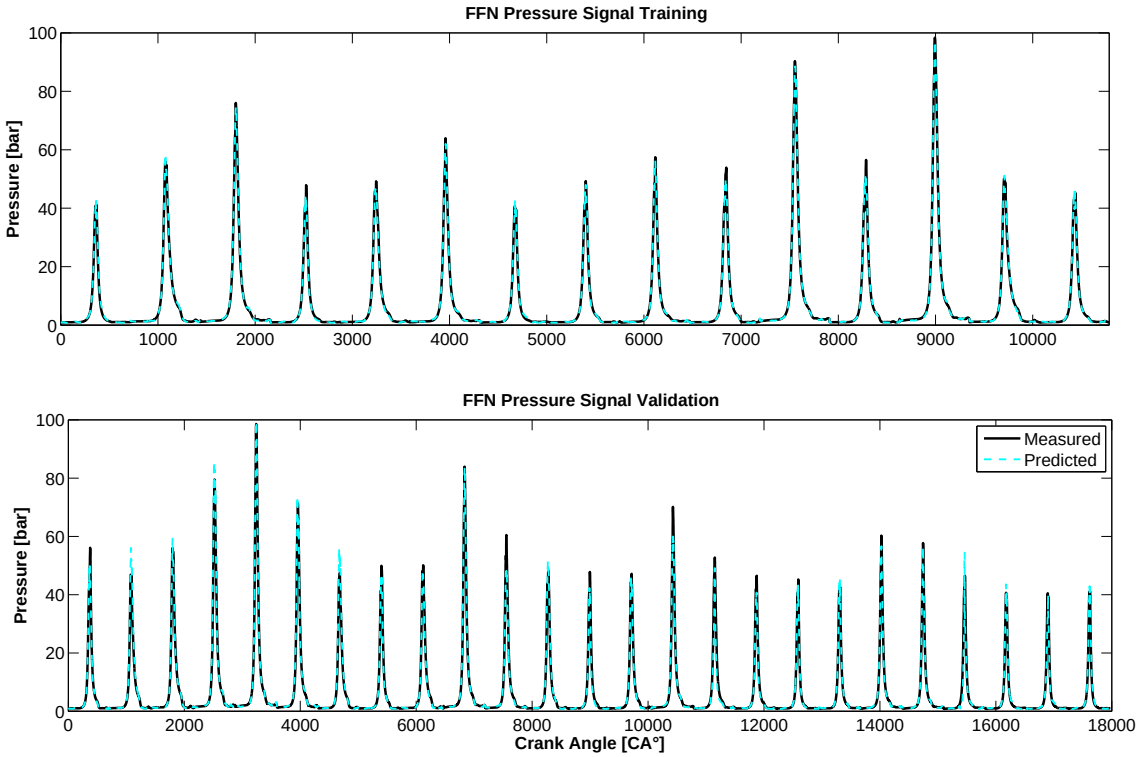


***Figure 6.5:*** *Comparison result for a FFN three-layer network [10 10] with seven inputs: training* $\mathrm{R}^2$ *= 0.99 and validation* $\mathrm{R}^2$ *= 0.99*

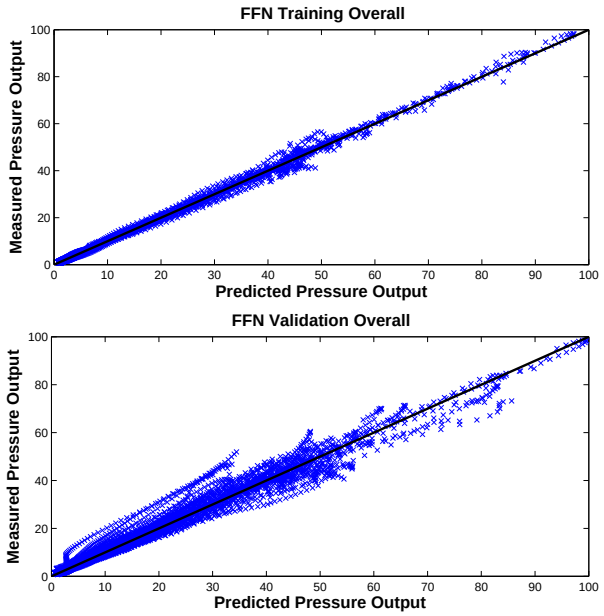


***Figure 6.6:*** *Value-to-value comparison along a linear plotline: training and validation set*

### 6.1.3 Multi-layer Feed-Forward Network Structure with Input Time - Delay

The multi-layer feed-forward structure with input time delay is characterised by an additional tuning parameter. The definition of previous input states considered for processing the current output is tuned during the optimisation process. For the investigation here, four previous input states are considered which showed the best results within a range of one to ten previous input states per input. The result presented in figure 6.7 and figure 6.8 is a two-layer network with four neurons per layer and seven inputs. The residual results for six- and seven-input network topologies are shown in the table of results in the appendix B.3. The six-input topologies show similar comparison performances as shown for the simple feed-forward structure in figure 6.3. In consequence, a seven-input approach shows the best performance for predicting the validation set of in-cylinder pressure traces.

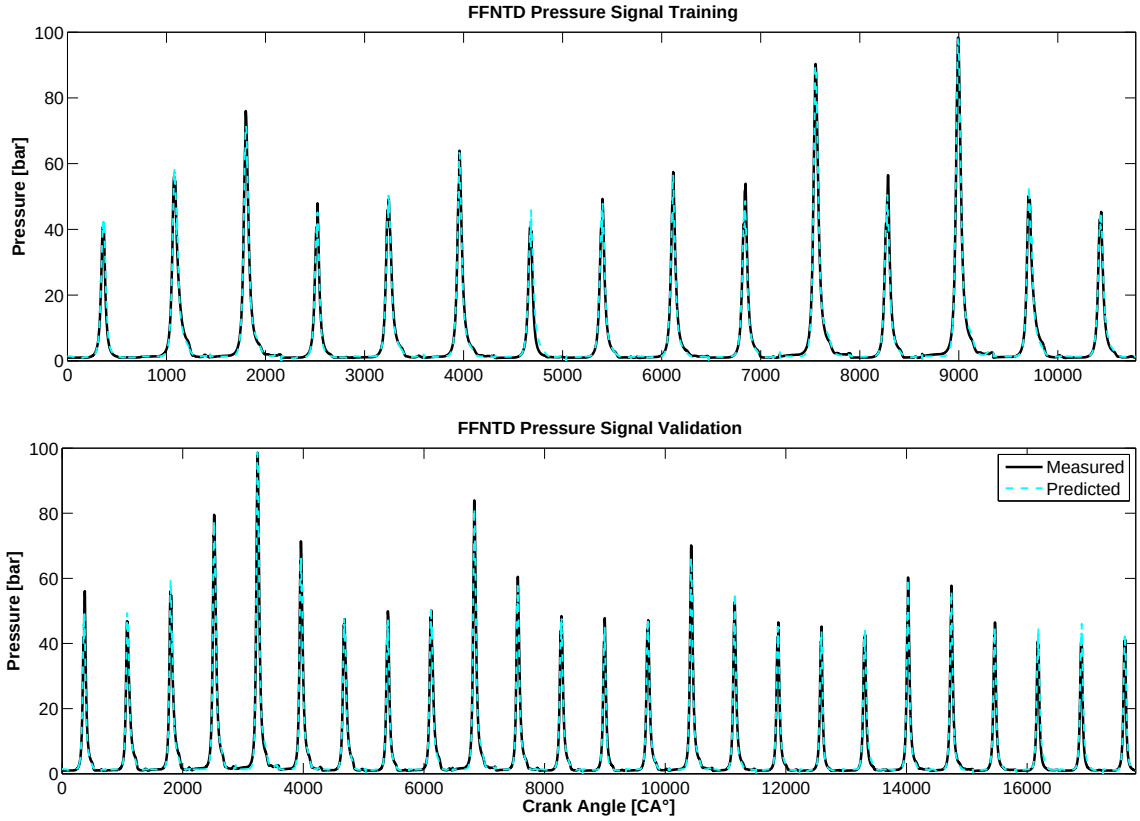


***Figure 6.7:*** *Comparison result for a FFNTD two-layer network [10 10] with seven inputs: training* $\mathrm{R}^2$ *= 0.99 and validation* $\mathrm{R}^2$ *= 0.99*

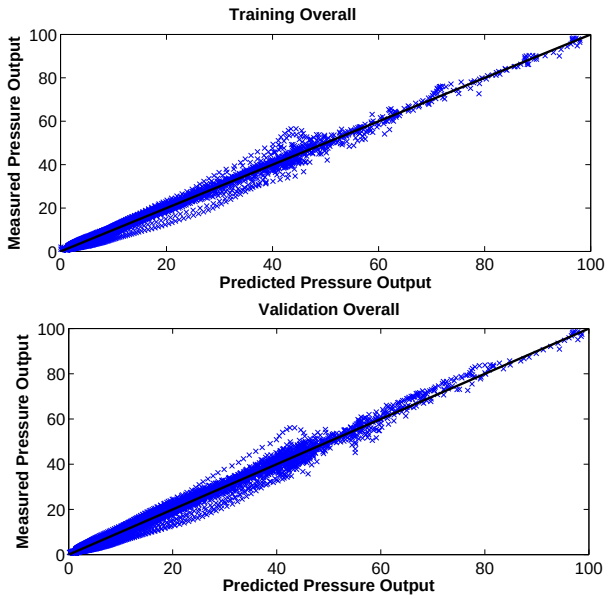


***Figure 6.8:*** *Value-to-value comparison along a linear plotline: training and validation set*

The comparison coefficient determined from the signals in figure 6.7 are 0.99 for training and validation respectively, which is similar to the previous structure. However, the network found for the input delay topology shows a slightly better value-to-value performance in comparison to the simple feed-forward structure as figure 6.8 implies. In particular, peak pressure is predicted more closely in the validation part which indicates a better generalisation capability of the network. In addition, the delay structure enables a decrease in the number of neurons per layer, resulting in a similar sufficient performance. However, the number of inputs increased due to the delayed inputs. Consequently the simple feed-forward structure is potentially preferable due to the overall simpler computational operation costs.

### 6.1.4 Non-linear ARX Structure

The NLARX network showed close comparison results in previous examples and achieves the best performance for this task. For this structure, the six initial chosen inputs are sufficient to be trained to map a relation of inputs and output. The most sufficient performance is found for a NLARX with two layers and three neurons per layer. The recurrent characteristic of this network are the output and previous states that are fed back as an additional input. This parameter is set to three previous output states that are considered for the system mapping. In addition, for each of the inputs, one previous state is used for the input-output relation mapping. In consequence, the network has 15 inputs [six current input states, six delayed input states, three recurrent output states = 15]. The result with this network is a comparison coefficient $\mathrm{R}^2$ = 0.99 for training and validation. The visualised results are plotted in figure 6.9 for the measured signal and the network output for training and validation. In addition, the value-to-value graph in figure 6.10 shows the closest fit along the regression line for all three structures presented.

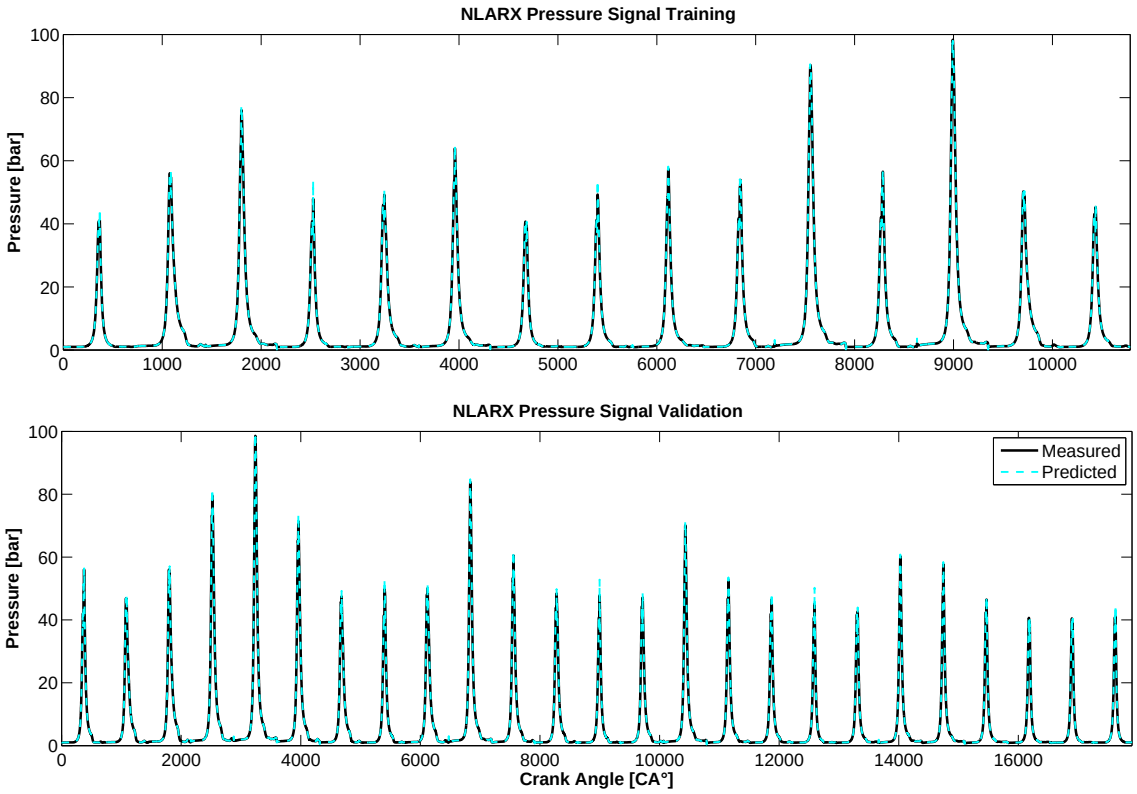


***Figure 6.9:*** *Comparison result for a NLARX three-layer network [3 3] with six inputs: training* $\mathrm{R}^2$ *= 0.99 and validation* $\mathrm{R}^2$ *= 0.99*

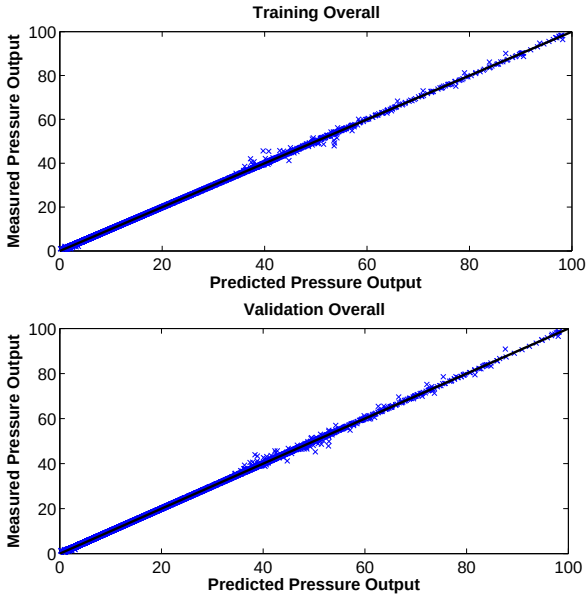


***Figure 6.10:*** *Value-to-value comparison along a linear plotline: training and validation set*

This recurrent structure shows the closest comparison in view of the value-to-value comparison.

It requires only six inputs in comparison to the two other networks that require the additional crank angle information in order to relate cylinder pressure behaviour to the input steps.

In addition, in figure 6.11 a cycle of the training set and a cycle of the validation set are picked along with the prediction signal of each of the optimal performing networks. The graph shows the measured signal plotted against three prediction signals. In the case of the training cycle, the NLARX comparison follows the signal closely through the start of combustion pressure change. At this point the feed-forward networks fail to predict the exact moment. The FFNTD shows a step in the predicted pressure, however it is too early. The FNN network creates a smooth transition from compression into combustion, which makes it difficult to detect the start of combustion. Consequently, the performance of the feed-forward networks is slightly offset in the validation set. In particular, the combustion (expansion) phase after the peak pressure shows the weakness of the feed-forward approach. In this case the network cannot generalise over the unseen case within the validation set. The NLARX network shows a close comparison throughout the trace. However, the peak pressure is overpredicted.

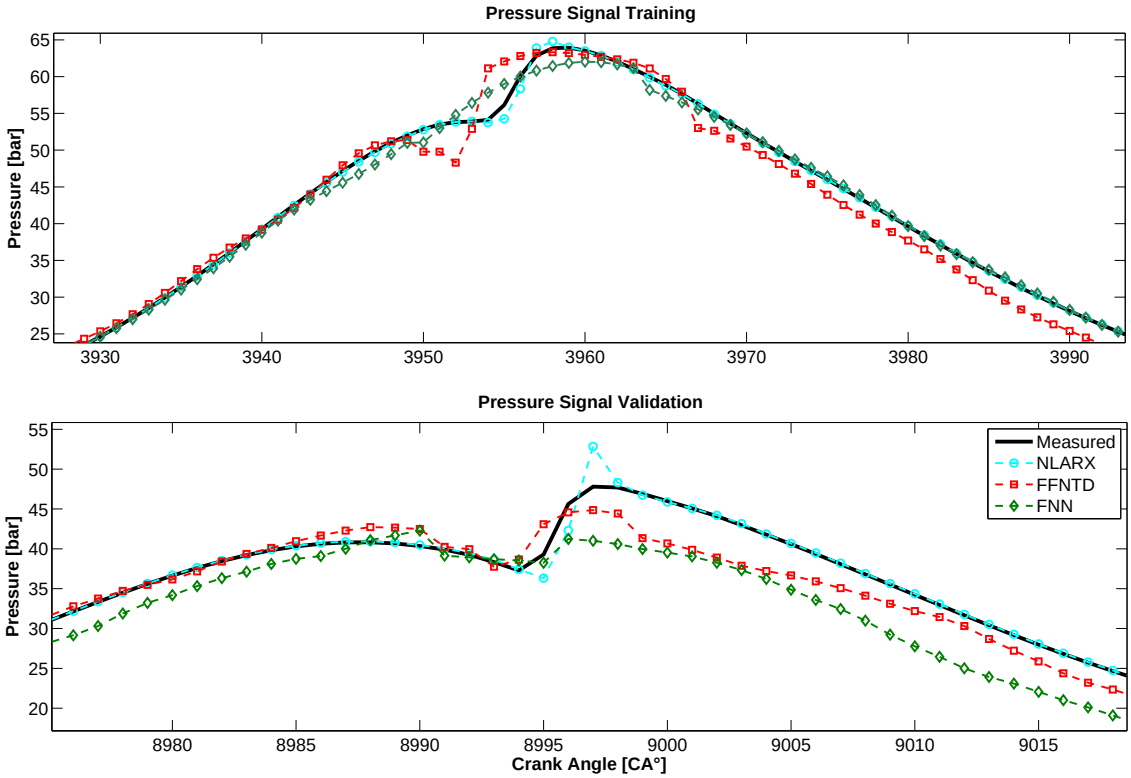


***Figure 6.11:*** *Comparison of all three best found network performances against a training and validation cycle*

In this section for each of the network structures, an best performing network topology is found with the least number of neurons and iteration runs. All of the trained networks presented in the result tables B.3, B.4, B.5 in the appendix B.2 are trained for 50 consecutive training data presentations. This is restricted in order to reduce the risk of overtraining the network and reducing its generalisation capabilities.

The next step is now the prediction of in-cylinder temperature in a similar approach.

## 6.2 Cylinder Temperature Modelling with GT-Power Data

The inputs used for the in-cylinder pressure modelling are also used for the in-cylinder temperature modelling. An exemplary temperature signal is shown in figure 5.2. The training and validation set is composed in the same approach as for the cylinder pressure data and are shown in figures 6.12 and 6.13 respectively.

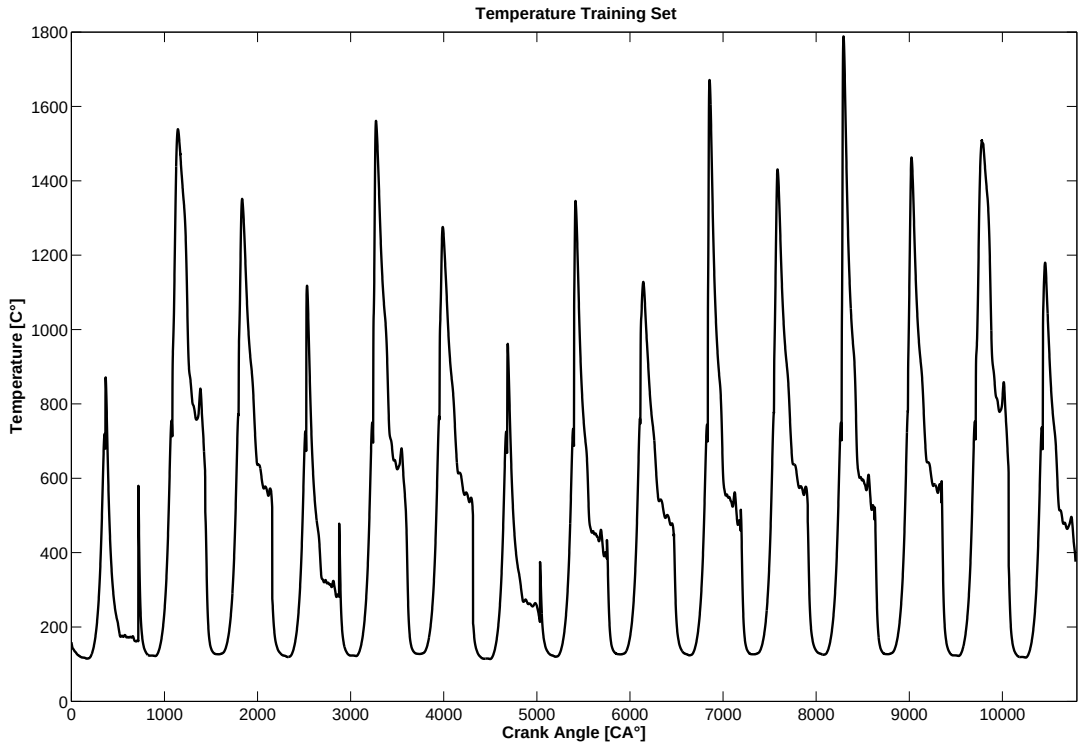


***Figure 6.12:*** *Training set for cylinder temperature modelling generated with GT-Power consisting of 15 cycles covering load scenarios at 800, 1400, 2200 RPM.*

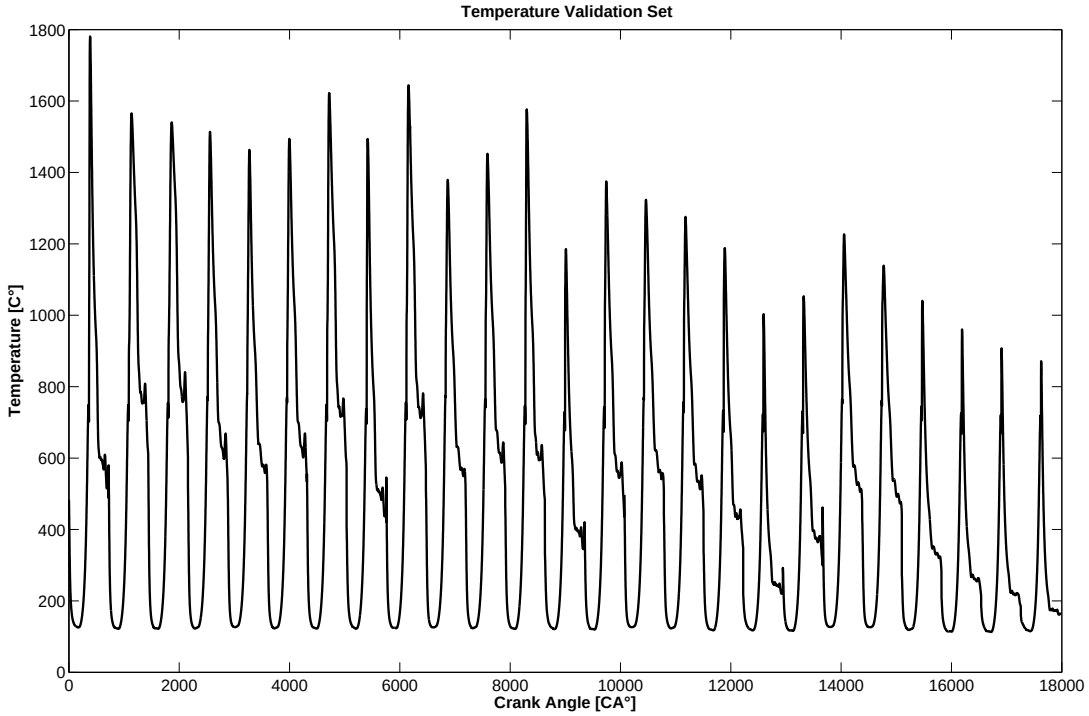


***Figure 6.13:*** *Validation set for cylinder temperature modelling generated with GT-Power consisting of 25 cycles covering load scenarios at 800, 1200, 1400, 1800, 2200 RPM.*

The in-cylinder temperature is modelled with the three network architectures as presented for the in-cylinder pressure modelling. First a topology for the multi-layer feed-forward structure is presented, followed by the input time-delay structure and, finally, the recurrent NLARX structure.

### 6.2.1 Multi-layer Feed-Forward Network Structure

With the initial six input set-up the feed-forward structure could not be optimised towards a sufficient $\mathrm{R}^2$ result. Hence, the mapping capability is increased by adding the crank angle signal to the input set. With seven inputs the network is optimised towards a performance of $\mathrm{R}^2 = 0.99$ for training and validation. However, the value-to-value comparison shows that the networks accuracy is not sufficient.

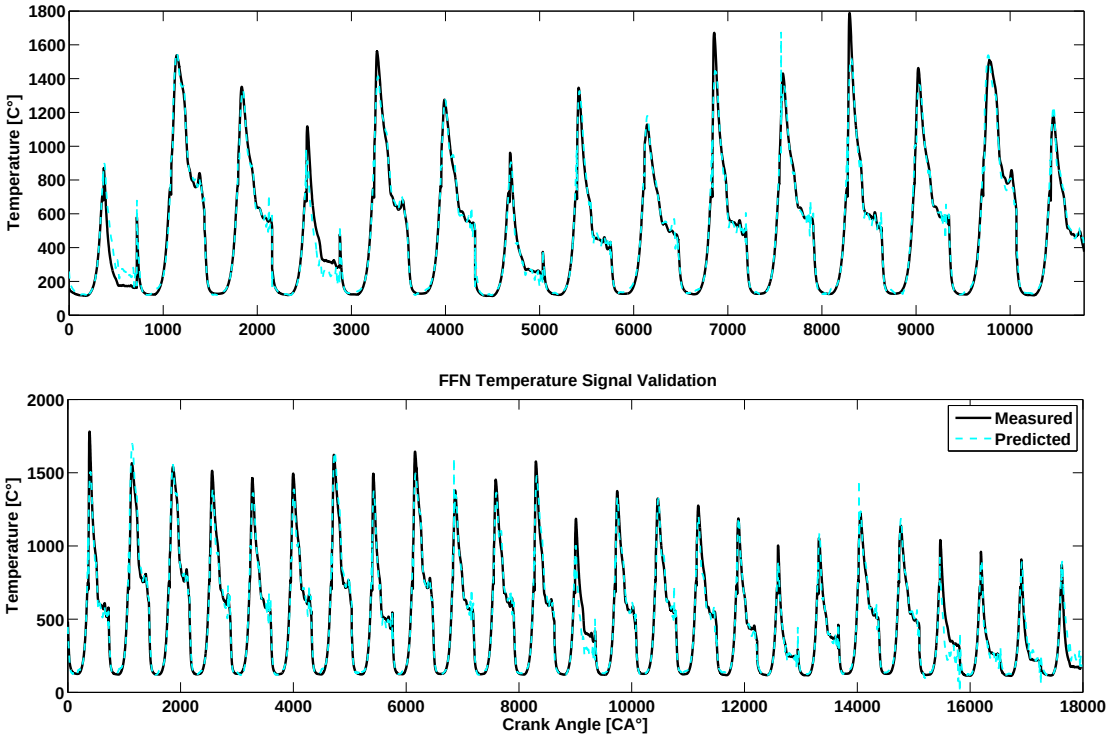


***Figure 6.14:*** *Comparison result for a FFN 2 layer network [10 10] with 7 inputs: training* $\mathrm{R}^2 = 0.99$ *and validation* $\mathrm{R}^2 = 0.99$

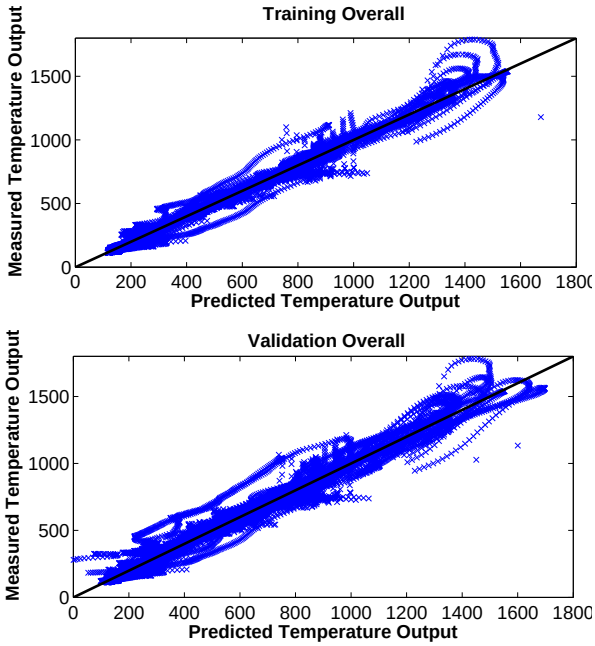


***Figure 6.15:*** *Value-to-value comparison along a linear plotline: training and validation set*

The network results presented in figures 6.14 and 6.15 show the prediction results for two-layer network with 10 neurons per layer. This network shows the closest value-to-value comparison of all tested networks. The residual test results can be found in table B.3 the appendix B.2.

### 6.2.2 Multi-layer Feed-Forward Network Structure with Input Time Delay

The multi-layer feed-forward structure with input time delay is trained for a sufficient result $\mathrm{R}^2$ of 0.99 for training and a $\mathrm{R}^2$ of 0.98 for validation shwon in figure 6.17.

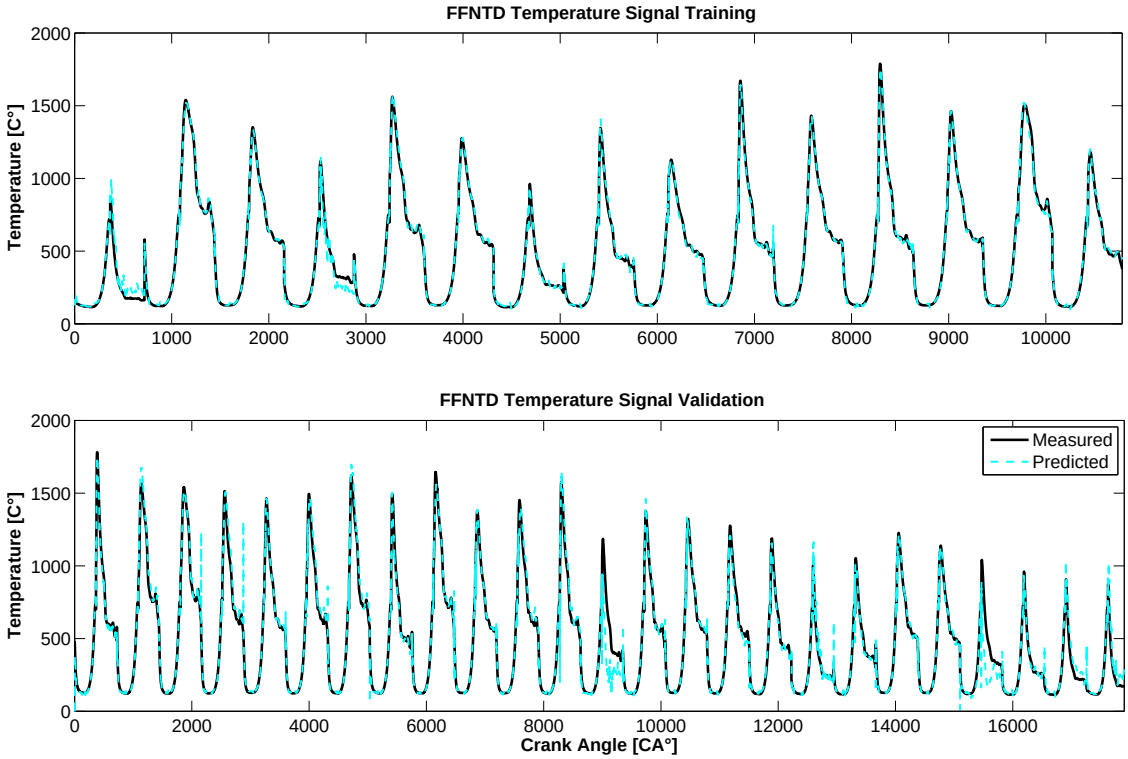


***Figure 6.16:*** *Comparison result for a FFNTD three-layer network [10 10 10] with seven inputs: training* $\mathrm{R}^2$ *= 0.99 and validation* $\mathrm{R}^2$ *= 0.98*

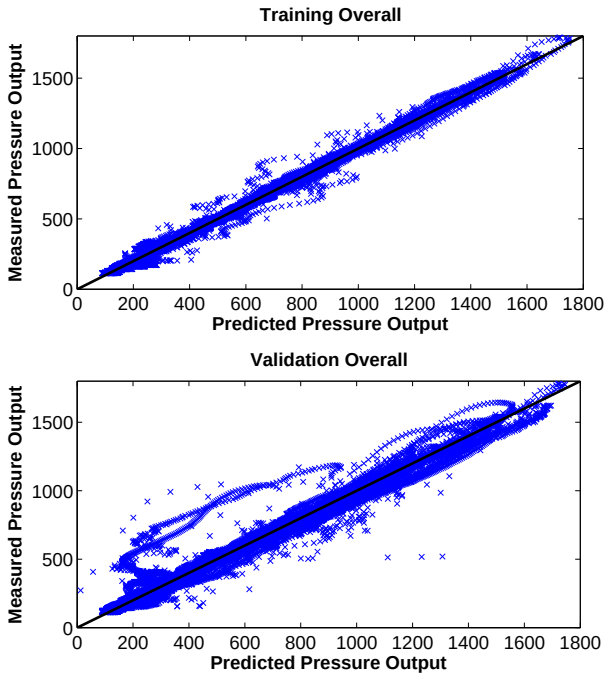


***Figure 6.17:*** *Value-to-value comparison along a linear plotline: training and validation set*

The visual comparison graph in figure 6.16 shows the good prediction capability of the network. The network generalises over the validation set except for certain temperature traces, in this case for 800 RPM and 30% load and 1800 RPM and 0% load (validation cycles 13 and 22). The results for all trained network designs are shown in the table B.4.

### 6.2.3 Non-Linear ARX Structure

The NLARX structure shows the best performance of all three structures for temperature prediction. A network trained with six inputs including two layers and four neurons per layer is the most sufficient network topology for this structure. It achieves an $\mathrm{R}^2$ of 0.99 for training and validation, and the value-to-value comparison is close for both sets. The number of outliers drifting away from the linear line in figure 6.19 is reduced significantly in comparison to the other two structures and their topology.

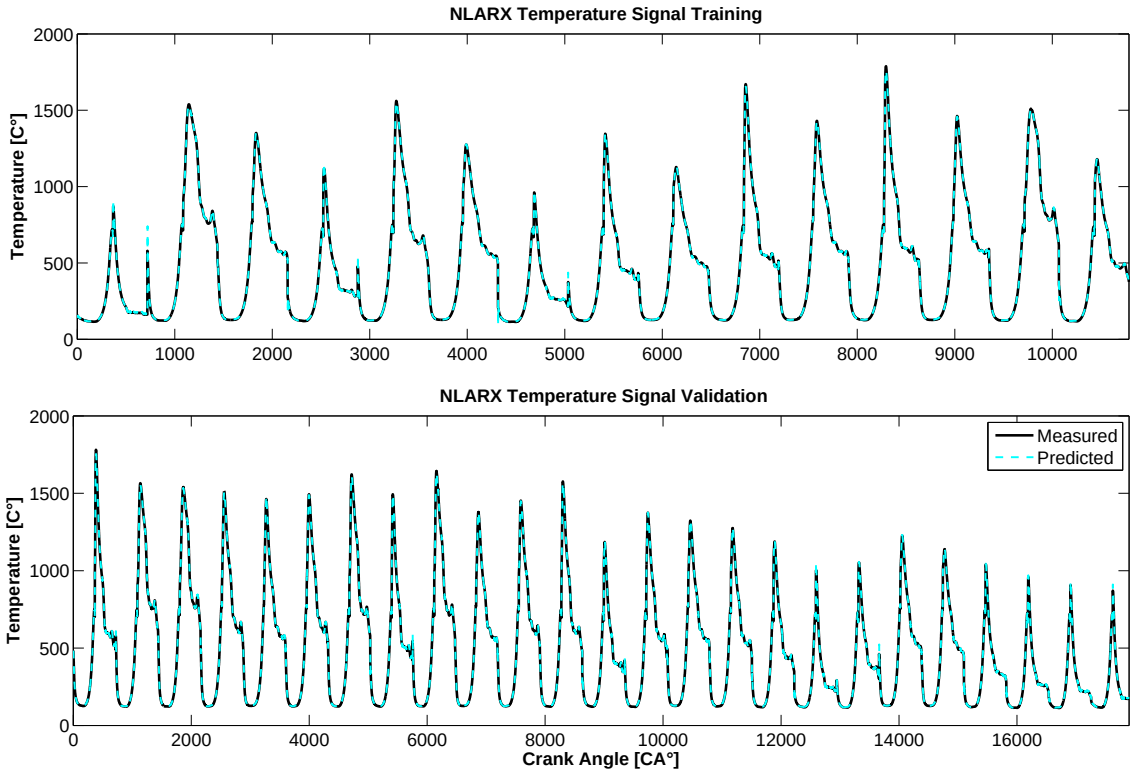


***Figure 6.18:*** *Comparison result for a NLARX two-layer network [4 4] with six inputs: training* $\mathrm{R}^2 = 0.99$ *and validation* $\mathrm{R}^2 = 0.99$

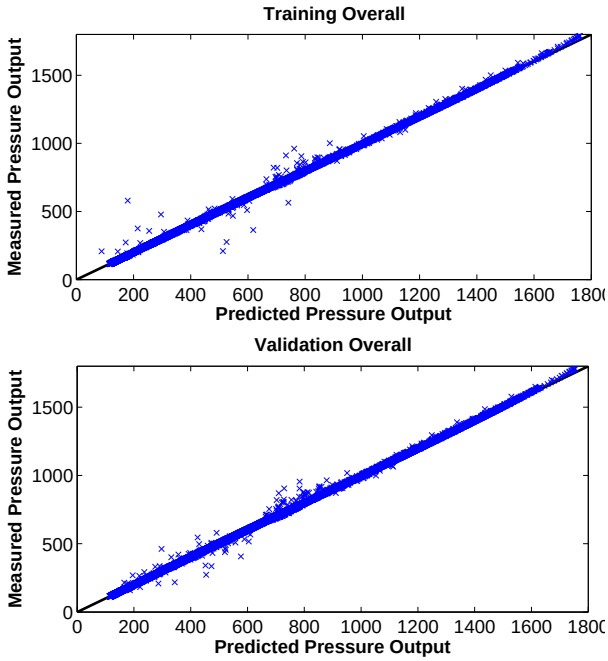


***Figure 6.19:*** *Value-to-value comparison along a linear plotline: training and validation set*

The generalisation capability of the NLARX network shows the best performance for all three structures and can predict across the engine operation range as represented by the validation set.

The performance increase of the NLARX is explicitly shown in figure 6.20 where cycles six and 13 of the training and validation set are plotted respectively. The graph shows the measured signal of the cycle and the prediction output of all three structures. The training set comparison is close for all three structures. An indication of the network's performance can be investigated at the step change of the signal during start of heat release - at sample 3955. The NLARX network fits the curve very closely whereas the simple FFN structure shows some delay in response which results in a signal step missing out the exact rate of heat increase. In comparison, the FFNTD shows a closer comparison due to increased dynamics incorporation.

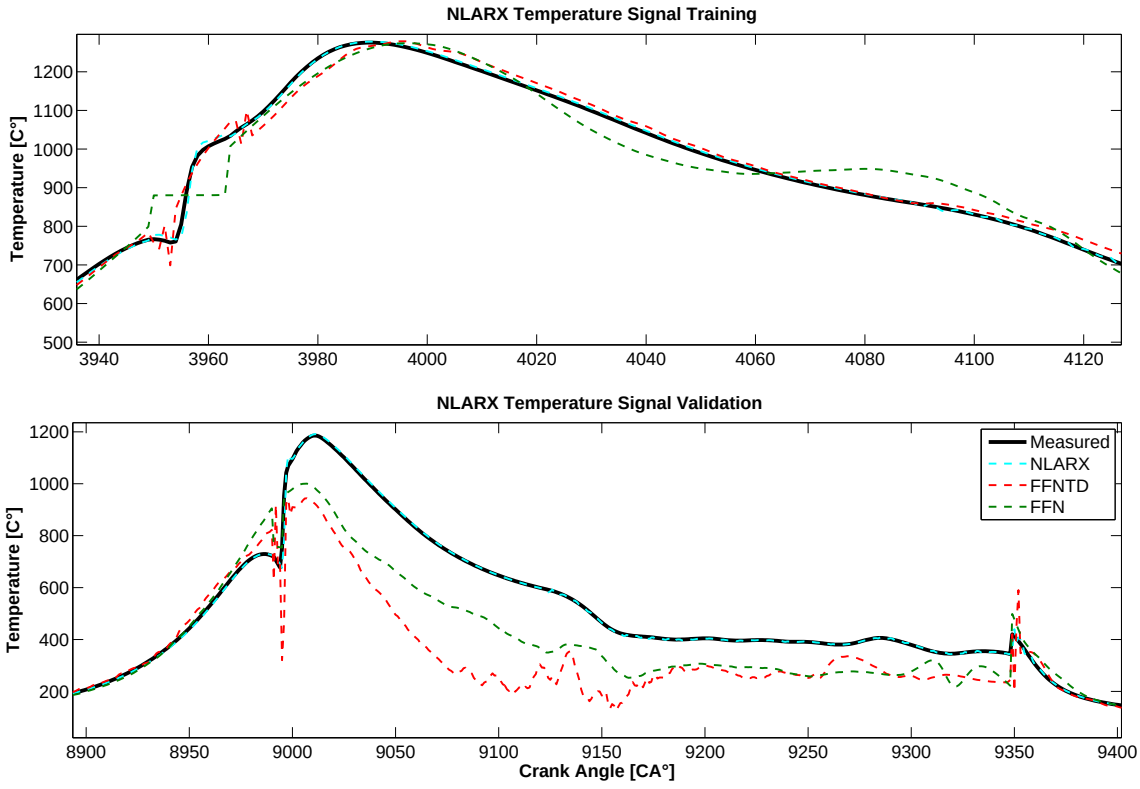


***Figure 6.20:*** *Comparison of all three network performances for temperature prediction. Zoomed in at training cycle six and validation cycle 13*

However, the validation signal shows the generalisation capabilities of the three structures. The NLARX network closely fits the cylinder temperature trace with all signal characteristics.

In the same view, the FFNTD and the FFN fail in predicting the exact temperature over the combustion cycle. This shows the lack of generalisation capability within the two feed-forward structures.

## 6.3 Conclusion on the Investigation of GT-Power Data Modelling

In this section the results of model identification for in-cylinder condition modelling are presented. Data generated with a validated GT-Power model are employed as training and validation data in order to investigate the capability of three different network structures in order to predict in-cylinder pressure and temperature.

The network structure with the best predictive and generalisation capabilities is the NLARX structure. It underlines its broad applicability amongst the other presented engine parameter modelling investigations in chapter 4. It shows good predictive capabilities with the chosen six inputs sketching the cylinder condition behaviour. In addition, the validation set that includes unseen cycle cases at different speeds and load scenarios fits closely the correlation.

Nevertheless, the presented feed-forward structures show good correspondence between measured and predicted signals for a variety of engine cases. An additional input enables the network to be trained sufficiently. By adding the current crank angle to the input list, the feed-forward networks are able to relate cylinder events and the network performance can be increased significantly. The FFN still shows delays in signal response and hence high frequency characteristics in the pressure or temperature trace such as start of combustion are not covered sufficiently. The FFNTD shows a better response capability, but the response overshoots the measured signal or does not meet the correct value. Unseen data in particular shows that the FFNTD has problems in generalising to a satisfactory level.

Overall, the network structures found display good correspondence and for each of the structures, a network topology is found that can predict the in-cylinder conditions over a wide range of engine states. Low engine speeds and low loads are of particular interest here. The input signal's rate of change and the output's response are less articulated. Therefore, it is more challenging for networks to distinguish between engine state changes. The FFN specifically

showed that at these states, the feature detection such as start of combustion or valve opening and closing events are difficult to model.

For further investigations, the simpler FFN and the NLARX network are used. The FFNTD shows a slight improvement in network performance over the FFN. However, the increase in training and prediction costs is significant since the delay structure adds four additional input feeds to the input layer. Hence, the FFNTD input layer increases up to 30 inputs instead of six as with the FFN structure. At the same time, the number of neurons within the hidden layer does increase within the FFN but the total number of neurons is kept lower than in the FFNTD for pressure and temperature prediction.

The next chapter will focus on the application of the tested network structures on real engine data recorded from the Caterpillar C6.6 engine. It is investigated whether the models are capable of dealing with noisy data signals as well as slightly different input sets.

# 7 Modelling Results with Real-Engine Generated Data

Additional data from the real engine described in chapter 5 is generated in order to validate the findings of the previous chapter. The measurements contain following channels used as input channels for the model:

1. Injector current
2. Exhaust temperature (port 1)
3. Compressor mass air flow
4. Needle lift
5. Intake manifold pressure
6. Inlet valve profile
7. Outlet valve profile

The list shows different inputs in comparison to the signals used for the GT-Power modelling. The reason for these differences lies within sensor availability and sensor capability. Hence, the intake manifold temperature sensor resolution is lacking accuracy due to the sensor type. In addition, the injected fuel mass - flow is replaced by the two sensor readings of injector current and needle lift. Despite the differences, the listed sensors contain the main information required for the modelling process as described in section 5.1.

Similar to the previous chapter, the signals to be modelled are the in-cylinder pressure and the in-cylinder temperature. The former signal is captured during the operation of the real engine with an installed in-cylinder pressure transducer. The latter is acquired by running the corresponding engine state in the simulation environment GT-Power in order to generate an in-cylinder temperature trace. The identical engine states within real-engine and engine

***Table 7.1:*** *Training set - speed and load scenarios*

| Speed [RPM] | Torque cases [%] | | | | | | |
|---|---|---|---|---|---|---|---|
| 800 | 0 | 10 | 20-30 | 40 | 50-60 | 70 | |
| 1400 | 0 | 0-10 | 20 | 30-40 | 50 | 60-70 | 70 |
| 2200 | 0 | 10-20 | 30 | 40-50 | 60 | 70 | |

simulation are supposed to generate close comparison as shown in chapter 5. The simulated trace is allocated to the respective case of measurements from the real engine. In the following, an optimum network is found for each parameter and the results are presented here. Firstly, this chapter presents the training and validation scenarios generated with real-engine operation. Secondly, the FFN and NLARX structures are pursued to find a modelling capability of in-cylinder pressure and temperature traces.

## 7.1 Real-Engine Training and Validation Data

The data recorded from the engine contains steady-state cases as found in the GT-Power results and transient phases. The test is designed to record data over 60 seconds with a load ranging from 0% up to 70. Load changes occur every four seconds after the load had been ramped up for 1.5 seconds to the current load stage.

The results of the test can be seen in figure 7.1 which shows the torque ramping up towards 70% load with the cylinder peak pressure rising along with transient behaviour during torque increase. The graph is plotted against the number of cycles recorded and available for training. The axis shows that approximately 50 cycles per load case are theoretically available for training. However, due to available computing performance and memory from each load case, one cycle of data is chosen randomly together with a cylce from the transient phase.

Hence, the training and validation data does include transient cases in addition to the steady-state scenarios. For the training set, 19 cycles are chosen randomly from 800 RPM, 1400 RPM and 2200 RPM at different loads and transients shown in table 7.1. The resulting training set is plotted in figure 7.2.

The validation set consists of 33 cycles including load cases and transients from 800 RPM,

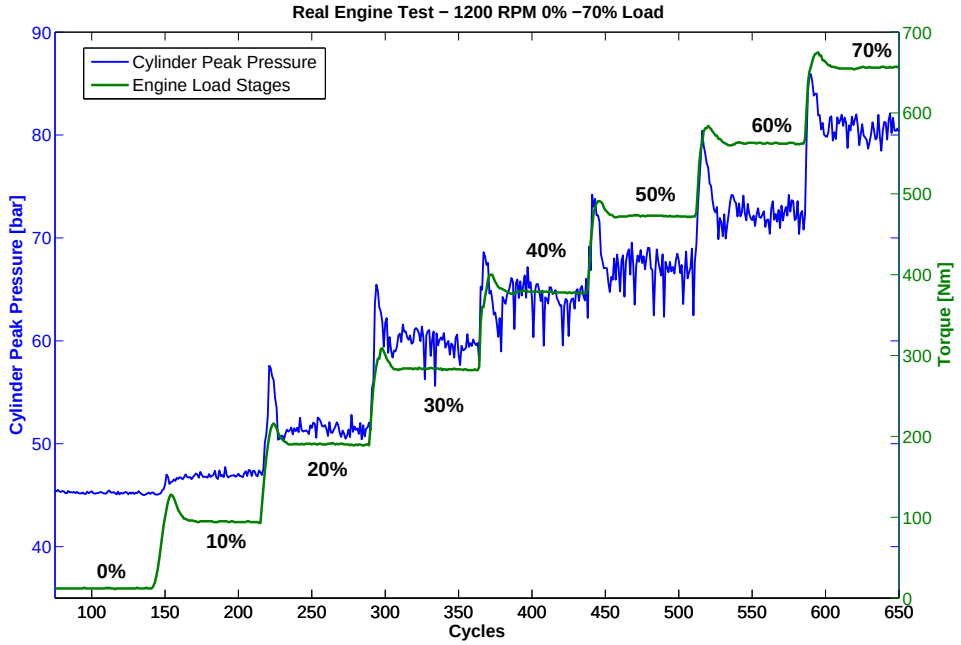


***Figure 7.1:*** *Engine test procedure with increasing torque at fix speed (1200 RPM) from 0% - 70% load*

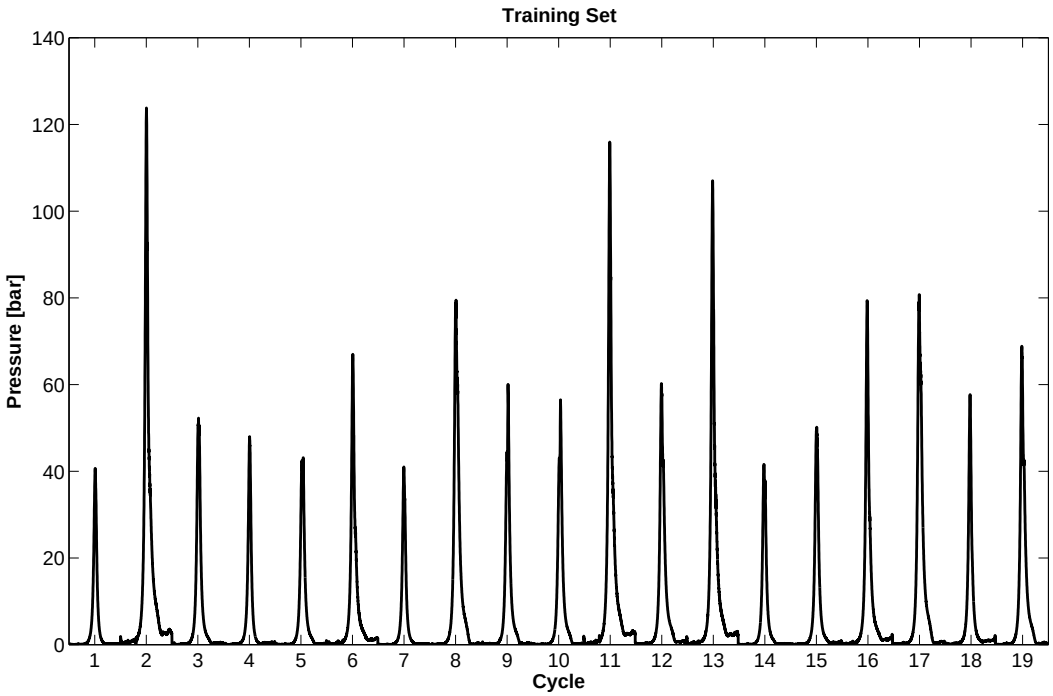


***Figure 7.2:*** *Set of pressure traces for network training recorded on the test engine*

***Table 7.2:*** *Validation set - speed and load scenarios*

| Speed [RPM] | Torque cases [%] | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 800 | 0 | 0-10 | 20-30 | 40-50 | 60-70 | 70 | | |
| 1200 | 0 | 10 | 20 | 30 | 40 | 50 | 60 | 70 |
| 1400 | 0 | 10-20 | 30-40 | 50-60 | 70 | | | |
| 1800 | 0 | 10 | 20 | 30 | 40 | 50 | 60 | 70 |
| 2200 | 0 | 0-10 | 20-30 | 40-50 | 60-70 | 70 | | |

1200 RPM, 1400 RPM, 1800 RPM and 2200 RPM - 7.2. The arrangement of cycles chosen for the test can be found in the tables C.1, C.2 of the appendix C.1 for the training and validation set respectively.

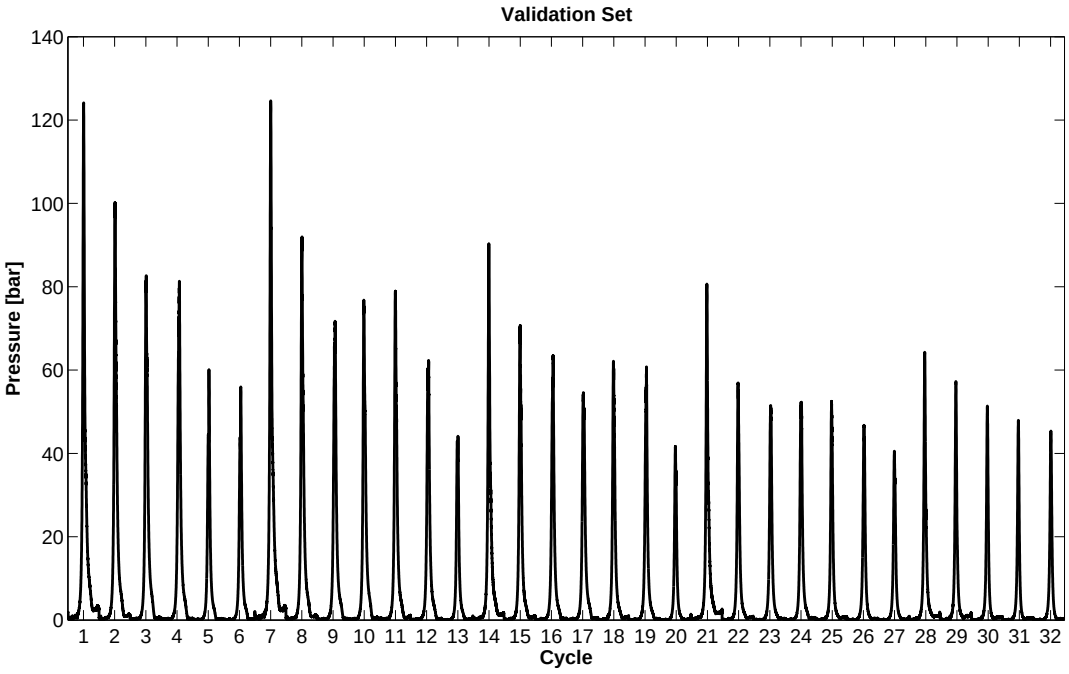


***Figure 7.3:*** *Set of pressure traces for network validation recorded on the test engine*

## 7.2 In-Cylinder Pressure Modelling with Real Engine Data

For modelling in-cylinder pressure, two network structures are investigated: the multi-layer feed-forward structure and the non-linear ARX structure.

### 7.2.1 NLARX Structure

The most sufficient NLARX structure in this search is found with a hidden two layer network with 8 neurons per layer. The input layer consists of the seven listed inputs plus one previous output state per input and three previous output states. The resulting trained network achieves a training and validation performance of $\mathrm{R}^2$=0.99. The comparison in figure 7.4 shows a good prediction of cylinder pressure for a variety of engine operation scenarios. The generalisation of the network can be seen in the lower of the two graphs in figure 7.4 where the validation set shows close comparison. Underlined is the comparison coefficient by the linearity check in figure 7.5 which shows the close fit of the value-to-value comparison with the regression line.

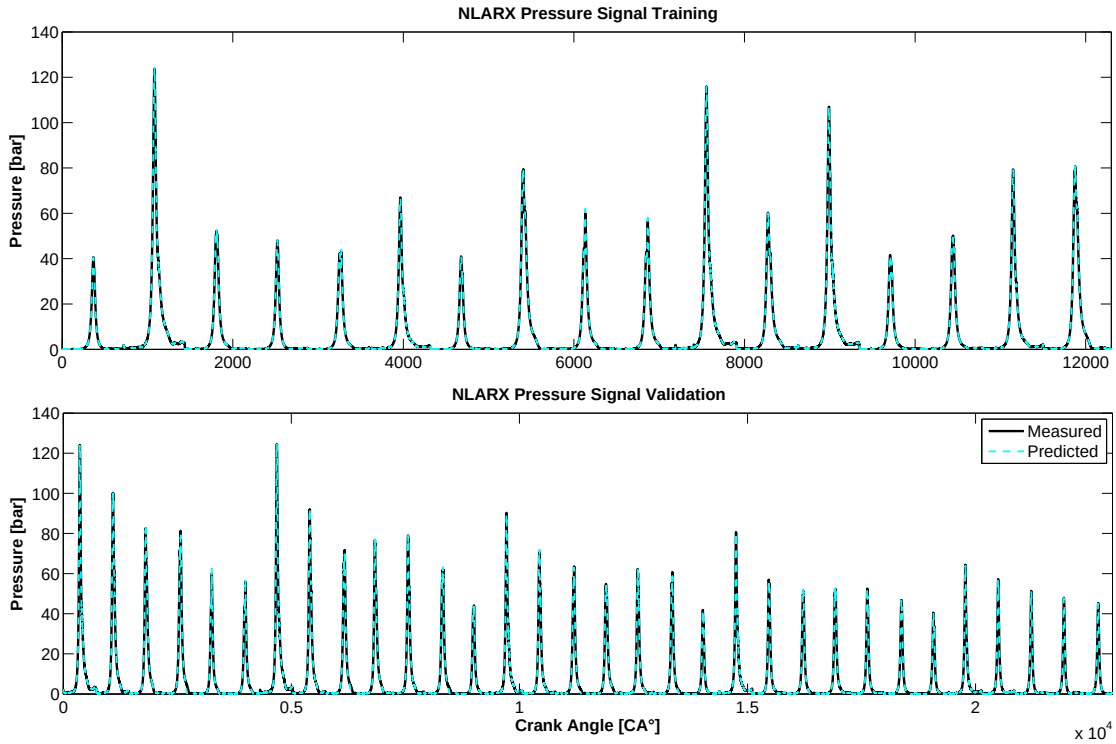


***Figure 7.4:*** *Comparison result for an NLARX two-layer network [8 8] with seven inputs: training* $\mathrm{R}^2$ = *0.99 and validation* $\mathrm{R}^2$ = *0.99*

### 7.2.2 Multi-layer Feed-Forward Structure

The feed-forward structure is trained with an additional input, the crank angle degree signal. Without this additional input, the network cannot be trained to find a relation between the

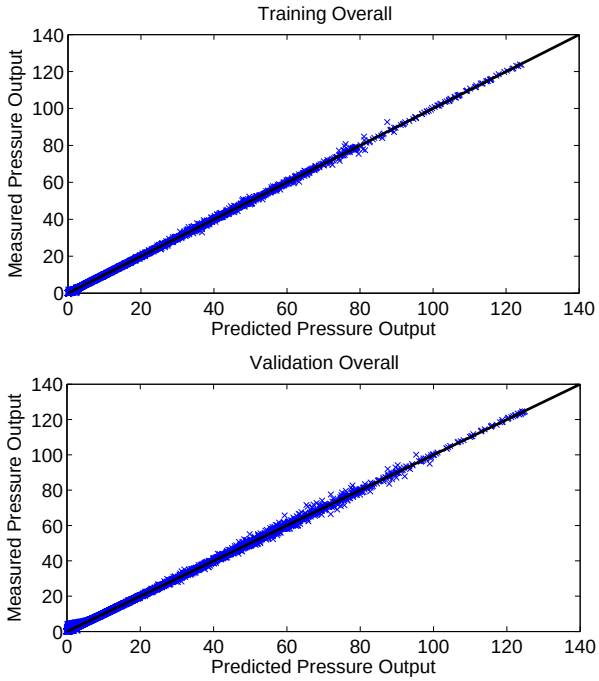


***Figure 7.5:*** *Value-to-value comparison along a linear plotline: training and validation set*

inputs and the output. Hence, this network contains an input layer of eight nodes. The hidden unit contains two layers with 10 neurons per layer. The result of the training and validation can be seen in figures 7.6 and 7.7. The comparison coefficient for the training set is $\mathrm{R}^2 = 0.98$ and for the validation set $\mathrm{R}^2 = 0.95$.

The validation signal in figure 7.7 shows the weakness of the FFN structure in generalising over unseen states. The value-to-value comparison shows a wide distribution around the regression line. In particular, states of 1200 RPM are not covered correctly. The network introduces some offset to the signal. This is assumed to be due to the unseen data at this speed state.

The results for all network topologies trained for pressure and temperature prediction can be found in tables C.3 and C.4 of appendix C.2 for the FFN and NLARX structures respectively. In the next section the results for the temperature prediction are presented.

## 7.3 In-Cylinder Temperature Modelling with Real-Engine Data

The in-cylinder temperature used is generated with GT-Power and, in the previous section, it has been shown that the FFN and NLARX structures are able to find a mapping relation

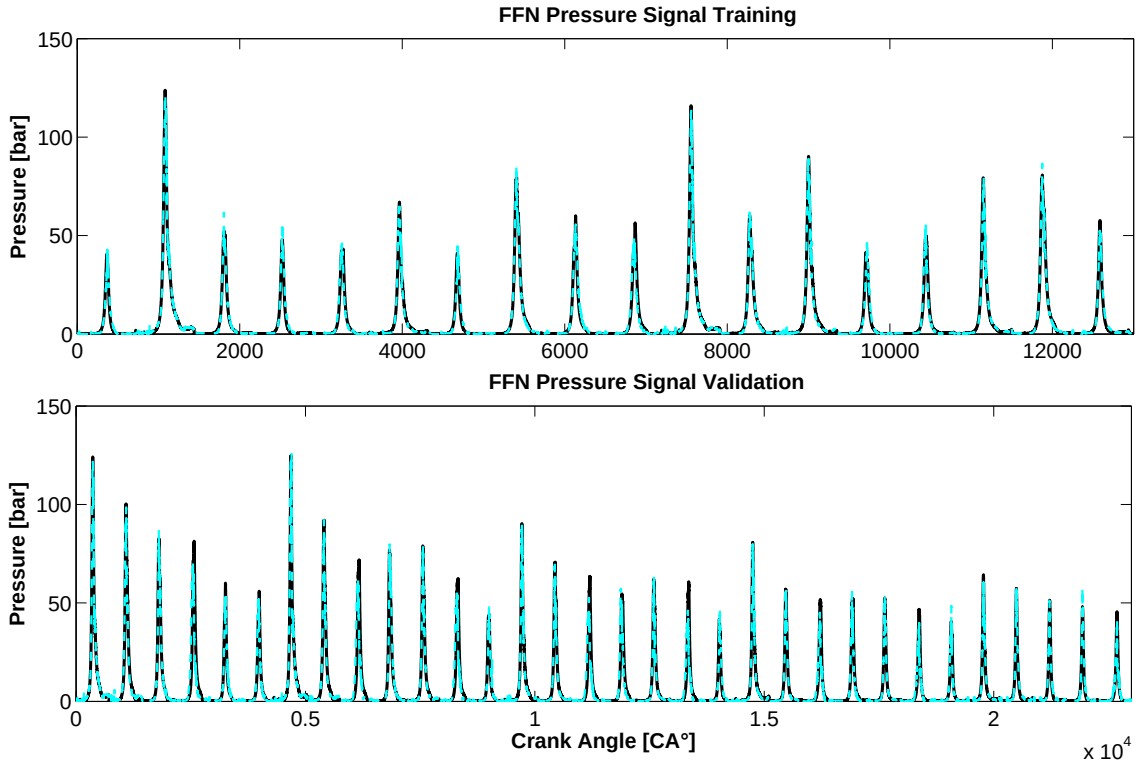


***Figure 7.6:*** *Comparison result for an FFN two-layer network [10 10] with eight inputs: training* $\mathrm{R}^2 = 0.98$ *and validation* $\mathrm{R}^2 = 0.95$

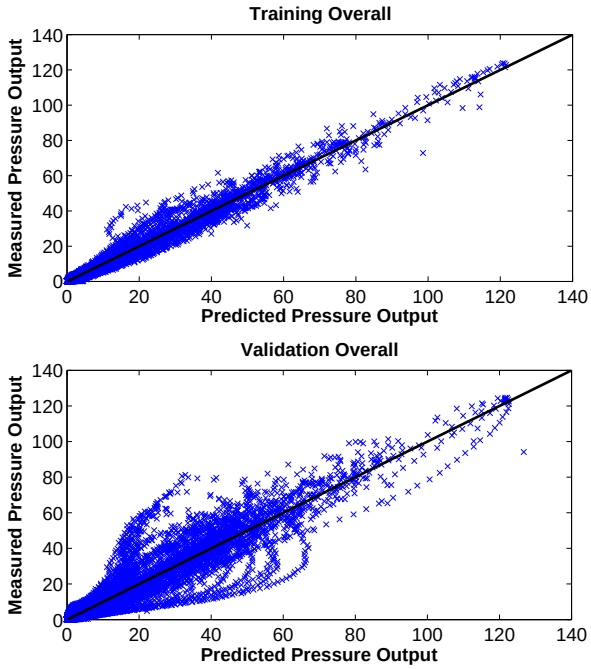


***Figure 7.7:*** *Value-to-value comparison along a linear plot line: training and validation set*

between inputs and outputs. However, in this section it is shown that the simulated cylinder temperature can be used as training data for a network based on real-engine data.

### 7.3.1 NLARX Structure

The comparison for the temperature signal shows good results with an $\mathrm{R}^2 = 0.99$ for training and validation. The same network topology is used as implemented for the in-cylinder pressure. The results are presented in figures 7.8 and 7.9.

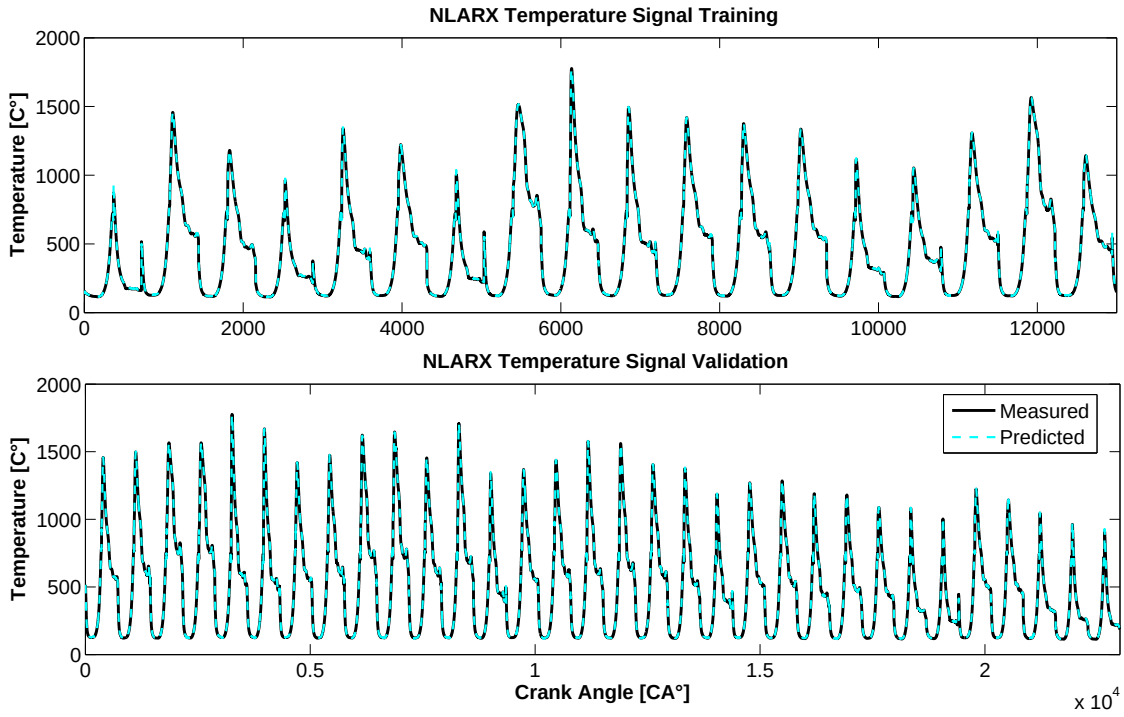


***Figure 7.8:*** *Comparison result for an NLARX two-layer network [8 8] with seven inputs: training* $\mathrm{R}^2 = 0.99$ *and validation* $\mathrm{R}^2 = 0.99$

### 7.3.2 Multi-layer Feed-Forward Structure

The feed-forward structure is changed towards a network topology with two layers and 20 neurons per layer. The results are plotted in figures 7.10 and 7.11. The performance shows similar behaviour as seen in the previous chapter. The training data shows a close fit in the visual and value-to-value comparison. However, the validation set is predicted less accurately as the value-to-value graph shows. In particular, the peak temperatures are missed repeatedly.

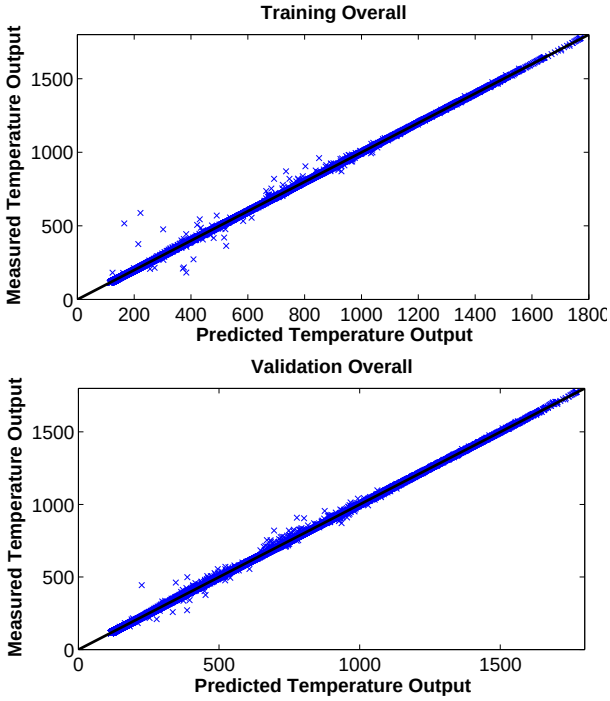


***Figure 7.9:*** *Value-to-value comparison along a linear plot line: training and validation set*

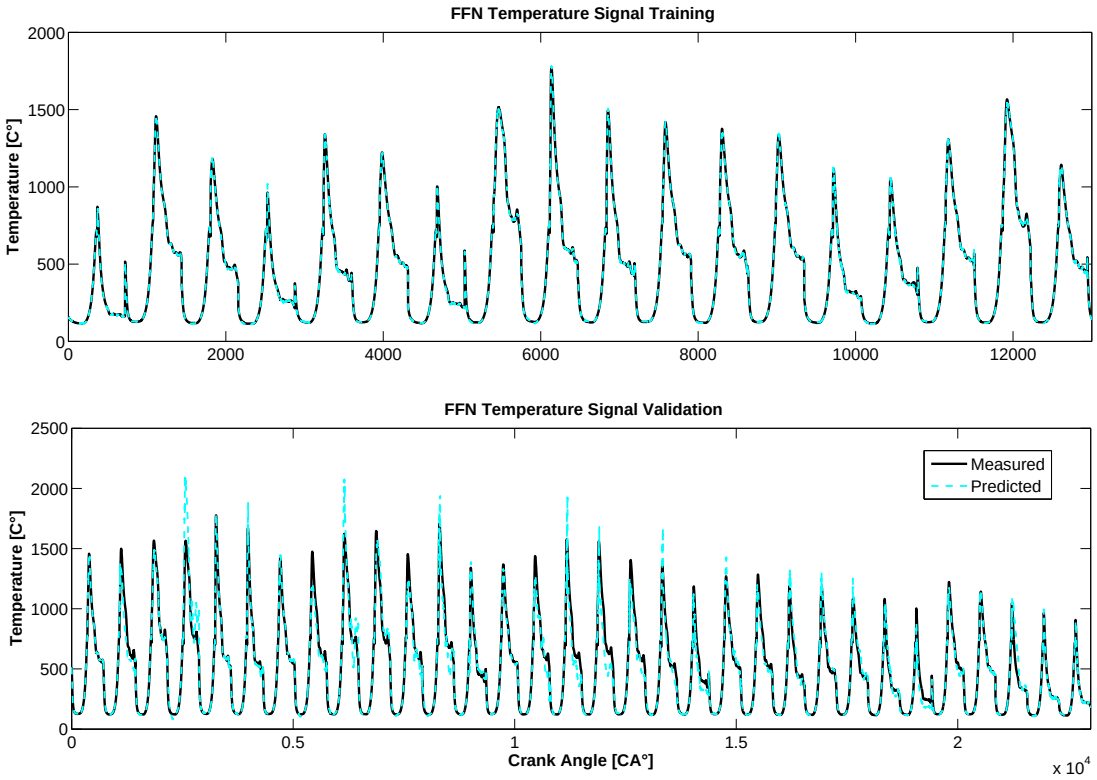


***Figure 7.10:*** *Comparison result for a FFN two-layer network [10 10] with eight inputs: training* $\mathrm{R}^2$ *= 0.99 and validation* $\mathrm{R}^2$ *= 0.98*

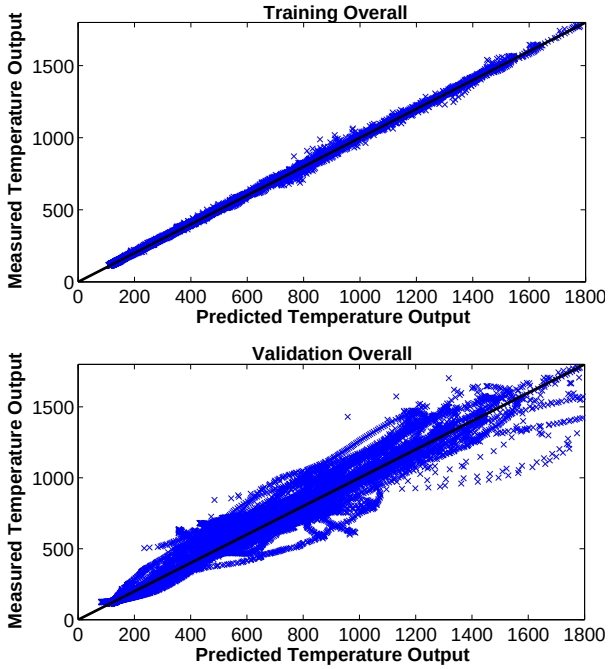


***Figure 7.11:*** *Value-to-value comparison along a linear plot line: training and validation for the temperature prediction*

Similar to the in-cylinder pressure data the FFN structure shows a close comparison in predicting the training data. In addition, the comparison coefficient of the validation set is sufficient. However, the value-to-value visualisation discloses that the comparison value mainly shows that the network can predict the trend of the temperature characteristic. The network is not trained for an exact value prediction.

## 7.4 Conclusion on the Investigation of Real Engine Data Modelling

This chapter presented the applicability of the findings of chapter 6. The network structures FFN and NLARX are first used to predict the in-cylinder pressure and then the in-cylinder temperature.

The training and validation data is recorded from the Caterpillar C6.6 engine presented in chapter 5. The training set contains seven initial inputs that are recorded in the crank angle domain. The list of signals differs slightly from the initial input set chosen in the GT-Power modelling chapter. This is due to restrictions in sensor resolution or a lack of equipment.

Nevertheless, the chosen inputs provide the information required for successful input-output mapping. In the case of the FFN structure, the crank angle signal is required as an additional input in order to find a mapping capability. The input enables the network to relate crank angle degrees with other input information.

The approach for in-cylinder pressure modelling can be concluded with the NLARX structure being the best choice for this modelling problem. A sufficient topology is found that shows close comparison in terms of training and validation for the comparison coefficient and the linear regression value-to-value fit. It can generalise over a variety of states that contain steady-state scenarios as well as torque transients generated through the design of experiment.

The FFN structure also achieves a sufficient training comparison and meets requirements such as prediction of peak values and correct timings. However, in some unseen cases within the validation set, the network fails to find the correct relation. A variety of network topologies have been tested and it is found that the FFN cannot sufficiently generalise over the chosen validation set. The network is able to predict the signal trend of in-cylinder pressure and temperature and predict the peak value. However, the value-to-value comparison discloses the weakness of exact value prediction with this structure.

From this investigation it can be concluded that the NLARX structure is the network with the best capability for in-cylinder modelling. It can be trained quickly, it shows good capability of generalisation, and can achieve good results with a minimum topology size. The main advantage of this network structure is the dynamics information provided through delayed input information and feedback of resulting outputs.

# 8 Summary and Conclusions

This thesis has outlined the challenging field of monitoring in-cylinder conditions. In particular, cylinder pressure and temperature characteristics during the combustion process would have great utility in combustion control. These parameters are fundamental in the description of the combustion process.

The detection and measurement of combustion parameters are currently both equipment and cost intensive and therefore pose difficulties for real-time application and mass-production. Temperature detection has been developed principally for laboratory applications and is quite unsuited to mass-production engines. Its measurement requires optical access and is therefore impractical. Cylinder pressure has been used for a long time in larger diesel engines and was recently introduced for the first time in light-duty engines. Its drawback includes high initial costs of implementation and high maintenance requirements.

Consequently, several other approaches have been developed for real-time monitoring of cylinder conditions to meet the needs of on-board diagnostics and controller design. Chapter 2 describes the key approaches for temperature and pressure reconstruction and prediction. However, the approaches are ill-posed by certain characteristics of engine parts or due to the difficulties in extracting information from structure-transferred noise that incorporates the in-cylinder information. The approaches presented for real-time application reconstruct the signal from previously obtained information. Hence, there is no predictive capability. Other modelling approaches are often accurate and require good system knowledge. Physical and mathematical models have a drawback due to their computational cost, which makes them inappropriate for real-time application.

This work investigates the applicability of neural networks to the field of combustion engine parameter prediction. In Chapter 4, 5, 6 and 7 several contributions in the perspective of

network structure and arrangement, their input identification, data acquisition and training data definition are made.

## 8.1 Main Findings and Contributions

**Neural Network Structure** The investigation of applicable neural network structures has shown that for engine parameter modelling, the NLARX structure is a reliable predictor. This structure showed its capability in a variety of investigations such as for predicting $\mathrm{NO_x}$, particulate matter or engine exhaust parameters. For this reason it is also chosen for the application of in-cylinder condition prediction where it showed its quality with regard of generalisation, accuracy and least-complex topology choice in comparison to other network structures. Others have shown sufficient predictive capability in view of response characteristics and overall correlation. However, the value-to-value correlation showed considerable differences in comparison to the NLARX structure. Feed-forward approaches can predict signals closely and can map the trend of non-linear parameter behaviour. However, the resolution and response of the NLARX structure shows superior accuracy. Here, the main finding is the applicability of the NLARX structure to the modeling task of in-cylinder conditions which shows apparent non-linear highly dynamic system behaviour. The application of a NLARX structure to model the combustion process is the novelty in this case.

**Input Identification** The modeling of systems behaviour requires an information source for state changes to the system that is incorporated within the input set. Hence, identification of the correct input set is crucial for defining a network that can represent the underlying functionality of the system. This work shows that the choice of inputs can drastically increase performance as in the case of the FFN structure implemented for the in-cylinder pressure and temperature prediction. By adding sample or time based information such as the crank position, the network can relate other events to certain crank positions. In particular, recurring events can be exploited as they occur within the in-cylinder domain. In general, input identification can be achieved with different approaches:

- Trial and error

- Systems knowledge
- Principal Component Analysis (PCA)

The first trial and error approach is most efficient if knowledge of the system is minor but a number of measurements are available. The second approach requires initial knowledge of the system which should allow the definition of inputs and their effect on the output. However, this approach can be misleading because black-box models are often used where there is a lack of system knowledge. In the last approach, the PCA characteristics of an input are analysed and ranked in comparison to the other inputs. The investigation of soot-rate prediction in section 4.2.4 showed a successful reduction of the input set and generation of an ANN prediction capability. An initial set-up of inputs was reduced using a PCA. Main findings are identification of key parameters as inputs for a successful neural network modeling approach. The analysis of inputs is approached with systems understanding rather than with statistical analysis as shown previously with a PCA. The inputs impact is confirmed by the networks modeling accuracy.

**Data Acquisition and Processing** The data acquisition and processing of data is crucial. The data available for systems training ideally has to completely cover the system's behaviour. Hence, knowledge of system boundaries is required in order to derive the output scope and the corresponding input range. In order to define additional cases, a random signal can be used to simulate the input characteristics and broaden the system's response and, with it, the data range covered by the network. The data acquired can either be recorded by a real-engine system, a software simulation, or a hybrid approach. In this work, a novel hybrid approach leads to the solution of developing a model that can predict the in-cylinder temperature of the real engine. Since measuring in-cylinder temperature is not possible on the real engine, a GT-Power simulation model is used to generate the missing signal. The model is validated against the real engine by using a Dynasty model, a Caterpillar Inc. simulation software. A predictive combustion model is then used to generate the in-cylinder temperature trace that is subsequently associated with the corresponding scenario measured on the real engine. This finding is confirmed by the networks performance with real engine input data predicting simulated in-cylinder temperature data over a broad engine operating range. In addition, the

acquisition approach also showed its validity in the in-cylinder pressure prediction where the network could be trained to predict simulated and measured pressure traces across the engine operating range.

Various methods can be used in order to capture a broad variety of system responses:

- Predefined tests
- Pseudo-random input signals
- Design of experiment

Another novelty in the perspective of data acquisition is the choice of predefined engine test scenarios are presented in the work for $\mathrm{NO_x}$, smoke and soot-rate prediction where an NRTC test is used for defining the system's boundaries. Another approach is the use of pseudo-random control signals that might be applied to start-of-injection, fuel-rail-pressure, or the ratio between injections as presented in the work about fuel path control in section 4.3. For the work presented in chapters 6 and 7, a design of experiment is applied where key feature points of the engine operation map are chosen.

The definition of training and validation sets is another part of this work that has been investigated closely. The initial work on the prediction of $\mathrm{NO_x}$ showed the importance of a sufficient and comprehensive representation of system behaviour in the training data used to find an optimum network topology. A further investigation on this is presented in sections 4.2.3 and 4.2.4. In addition, it is important to ensure that the maximum output signal amplitudes are represented in the training signal as shown in section 4.2.3 where the peaks in smoke output are part of the modelling task. Here, another contribution is presented. A novel approach of slicing a signal into separate amplitude zones in the time domain rather than investigating the frequency domain is shown. The resulting modular network combination is applied from literature but shows the validity of slicing approach.

## 8.2 Conclusion on In Cylinder Condition Modelling Opportunities and Limitations

A new approach has been presented for in-cylinder modelling. An NLARX network is found to be the best-performing network structure on in-cylinder pressure and temperature prediction. The network structure shows fast training characteristics. The training algorithm finds a good generalising and predictive network structure after less than 50 iterations. The correlation of training and validation data is sufficient with an $\mathrm{R}^2$ =0.99 for both sets. The network topology found generalises over a variety of engine states. The training and validation set was chosen randomly from the data generated with a validated GT-Power model and a real C6.6 engine.

The results show that the feed-forward structures can predict the trend of the desired signal. However, the accuracy of the found network topologies is not sufficient. The network's output response is slightly delayed and causes the loss of characteristics of the desired signal such as start of combustion or inlet and outlet valve impacts. Despite this uncertainty, the peak pressure and temperature are predicted for most of the test scenarios. Hence, for monitoring of peak pressure conditions, the feed-forward structures seem applicable. Their ability of pressure trace prediction appears limited and hence an application for in-cylinder closed-loop control is in question. Key parameters such as start-of-combustion or 50$\%$ Mass-Fraction-Burned cannot be determined exactly in the current training state. The limitations in computational performance restrict the training process and hence a better prediction.

The presented NLARX network structure can be either used for monitoring purposes or for the support of controller design. It could be either implemented as an engine plant model that enables the simulation of engine cycles during tuning and optimization of a closed-loop controller. In addition, the network could be operated as a virtual sensor within the engine management environment. Also, the application as part of a model- predictive controller is a possibility. The drawback of this approach is the wide range of channels required for successful modelling. Injector current, needle lift or exhaust port pressure are not currently standard sensors within a production engine. Another issue is the possible shift of engine behaviour and the consequent change in input signals and hence system output. A network implemented for engine monitoring requires regular retraining in order to ensure a reliable

output.

Some of the current limitations shall be part of future work on this topic.

## 8.3 Outlook and Future Work

The presented approach is promising for controller design as well as on-board diagnostics and combustion control. The on-board diagnostic capability is currently restricted through the required high sampling rates and some of the chosen inputs. One future step would be the implementation of the network structure on a test engine such as the C6.6 engine used for data generation in order to test the real-time applicability across the entire engine operation range. The current design of experiment incorporates a variety of key engine speeds and loads. In addition to the steady-state cases, transients between load stages are incorporated. However, transients between speeds and transients such as idle speed and no load to full load require further investigation. This area of additional training cycles also raises the point of how much data a network could cover. For this scenario, a further investigation could look into the application of a network map that covers this distinct engine operating range. These operating ranges can either be on the speed-load curve or defined by the transients that occur during engine operation. An initial test of the network structure can be applied with the validated GT-Power model. This work showed that in-cylinder temperature output data generated from the GT-Power simulation can be allocated with input data from the real engine and then predicted. The next step would be the validation of the network's ability to operate in real - time on the engine and predict the in-cylinder temperature based on the simulated training data. A final step for proving the network's capability of temperature prediction would be the generation of real in-cylinder temperature data following the training and validation of a network topology.

Another field of interest would be the implementation of an in-cylinder condition plant model for controller design. Combustion control is challenging and requires fast and reliable monitoring information. The presented approach can provide a reliable model plant for designing control structures aiming at emissions formation reduction based on temperature and pressure information. In addition, parameters such as heat - release, peak cylinder pressure, burn rate

or start-of-combustion can be derived from a virtual sensor-based approach. These parameters can help controlling combustion conditions that reduce emissions output such as lower temperatures in view of $\mathrm{NO_x}$ formation.

# Bibliography

[1] W. Knecht. Diesel engine development in view of reduced emission standards. *Energy*, 33:264–271, 2008.

[2] K. P. Schindler. Accelerating light-duty diesel sales in the u.s. market. In *12th Diesel Engine-Efficiency and Emissions Research (DEER) Conference*, August 20-24 2006.

[3] E.F. Obert. *Internal Combustion Engines and Air Pollution*. Harper & Row, New York, 1973.

[4] T. Bae. An interferometric fiber optic sensor embedded in a spark plug for in-cylinder pressure measurement in engines, 2002. ID: 1.

[5] S. Neumann. High temperature pressure sensor based on thin film strain gauges on stainless steel for continous cylinder pressure control. In *CIMAC World Congress on Combustion Engine Technology*, volume 25. International Council on Combustion Engines, 21-24 May 2007.

[6] H. Zhao and N. Ladommatos. *Engine Combustion Instrumentation And Diagnostics*, volume 1. Society of Automotive Engineers, Warrendale, 2001. ISBN 0768006651.

[7] J. C. Livengood, T. P. Rona, and J. J. Baruch. Ultrasonic temperature measurement in internal combustion engine chamber. *The Journal of the Acoustical Society of America*, 26(5):824–830, 1954.

[8] W. T. Lyn. Diesel combustion study by infrared emission spectroscopy. *Journal of the Institution of Petroleum*, 43(398):25–46, 1954.

[9] N. Kawahara, E. Tomita, and H. Kamakura. Unburned gas temperature measurement in a spark-ignition engine using fibre-optic heterodyne interferometry. *Measurement Science and Technology*, 13:125–131, 2002.

[10] D. Prokhorov. Virtual sensors and their automotive applications, 2005.

[11] E. Hanzevack. Virtual sensors for spark ignition engines using neural networks, 1997.

[12] C. Atkinson, M. Traver, T. W. Long, and E. Hanzevack. Predicting smoke, 2002.

[13] C. A. Finol and K. Robinson. Thermal profile of a modern passenger car diesel engine. *SAE Technical Paper*, 2006-01-3409, 2006.

[14] R. Hickling, D. A. Feldmaier, F. H. K. Chen, and J. S. Morel. Cavity resonances in engine combustion chambers and some applications. *Journal of the Acoustical Society of America*, 73(4):1170–1178, 1983.

[15] J. Carryer, R. H. Roy, and J. D. Powell. Estimating in-cylinder precombustion mixture temperature using accoustic resonances. *Transactions of the American Society of Mechanical Engineers*, 118:106–112, 1996.

[16] Y. Huang, L. Zhang, and W. Liu. The optimum of heat release patterns in high speed d.i. diesel engine. *SAE Technical Paper*, 941694:57–62, 1994.

[17] G. Stiesch and G. P. Merker. A phenomenological model for accurate and time efficient prediction of heat release and exhaust emissions in direct-injection diesel engines. *SAE Technical Paper*, 1999-01-1535, 1999.

[18] D. T. Hountalas, D. A. Kouremenos, and S. B. Fiveland. Some considerations on the estimation of the heat release of di diesel engines using modelling techniques. *SAE Technical Paper*, 2004-01-1405, 2004.

[19] P. Tamilporai, N. Baluswamy, P. Mannar Jawahar, S. Subramaniyam, S. Chandrasekaran, K. Vijayan, S. Jaichandar, and K. Janci Rani, J. Arunchalam. Simulation and analysis of combustion and heat transfer in low heat rejection diesel engine using two zone combustion model and different heat transfer models. *SAE Technical Paper*, 2003-01-1067, 2003.

[20] F. G. Chmela and G. C. Orthaber. Rate of heat release prediction for direct injection diesel engines based on purely mixing controlled combustion. *SAE Technical Paper*, 1999-01-0186, 1999.

[21] P. A. Lakshminarayanan, Y. V. Ahgav, A. D. Dani, and P. S. Mehta. Accurate prediction of the rate of heat release in a modern direct injection diesel engine. *Proceedings of the I MECH E Part D Journal of Automobile Engineering*, 216:663–675, 2002.

[22] H. Ogawa and N. Miyamoto. Characteristics of low temperature and low oxygen diesel combustion with ultra-high exhaust gas recirculation. *International Journal of Engine Research*, 8:365–378, 2007.

[23] R. Villarino and J. F. Böhme. Fast in-cylinder pressure reconstruction from structure-borne sound using the em algorithm, 2003.

[24] J. Antoni, J. Daniere, and F. Guillet. Effective vibration analysis of ic engines using cyclostationarity. part i - a methodology for condition monitoring. *Journal of Sound and Vibration,*, 257(5):815–837, 11/7 2002.

[25] R. G. DeJong, R. E. Powell, and J. E. Manning. Engine monitoring using vibration signals. *SAE Technical Paper*, 861246, 1986.

[26] J. Antoni, J. Daniere, F. Guillet, and R. B. Randall. Effective vibration analysis of ic engines using cyclostanarity. partii - new results on the reconstruction of the cylinder pressure. *Journal of Sound and Vibration,*, 257(5):839–856, 11/7 2002.

[27] D. J. McCarthy and R. H. Lyon. Recovery of impact signatures in machine structures. *Mechanical Systems and Signal Processing,*, 9(5):465–483, 9 1995.

[28] J. T. Kim and R. H. Lyon. Cepstral analysis as a tool for robust processing, deverberation and detection of transients. *Mechanical Systems and Signal Processing*, 6(1):1–15, 1992.

[29] M. El-Ghamry, J. A. Steel, R. L. Reuben, and T. L. Fog. Indirect measurement of cylinder pressure from diesel engine using acoustic emission. *Mechanical Systems and Signal Processing*, 19:751–765, 2005.

[30] Y. Gao and R. B. Randall. Reconstruction of diesel engine cylinder pressure using a time domain smoothing technique. *Mechanical Systems and Signal Processing*, 12(5): 709–722, 1999.

[31] G. Zurita, D. Haupt, and A. Anders. Reconstruction of cylinder pressure through multivariate data analysis for prediction of noise and exhaust emissions. *Noise Control Engineering Journal*, 52(4):154–163, 2004.

[32] H. Du, L. Zhang, and X. Shi. Reconstructing cylinder pressure from vibration signals based on radial basis function networks. In *Proceedings of the I MECH E Part D Journal of Automobile Engineering*, volume 215, pages 761–767, 2001.

[33] S. Vulli. *Engine Cylinder Pressure Reconstruction Using Neural Networks and Knock Sensor Measurement*. Phd-thesis, School of Design and Technology, University of Sussex, 2006.

[34] J. Williams. An overview of misfiring cylinder engine diagnostic techniques based on crankshaft angular velocity measurements. *SAE Technical Paper*, 960039:31–37, 1996.

[35] S. Ginoux and J. C. Champoussin. Engine torque determination by crankangle measurements: State of the art, future prospects. *SAE Technical Paper*, 970532:17–22, 1997.

[36] G. Rizzoni. Estimate of indicated torque from crankshaft speed fluctuations: a model for the dynamics of the ic engine, 1989.

[37] G. Rizzoni and Y. Zhang. Identification of a non-linear internal combustion engine model for on-line indicated torque estimation. *Mechanical Systems and Signal Processing,*, 8 (3):275–287, 5 1994.

[38] F. T. Connolly. Direct estimation of cyclic combustion pressure variability using engine speed fluctuations in an internal combustion engine. *SAE Technical Paper*, 940143:1–11, 1994.

[39] F. T. Connolly and A. E. Yagle. Modeling and identification of the combustion pressure process in internal combustion engines. In *Proceedings of the 36th Midwest Symposium on Circuits and Systems*, volume 1, page 204, 1993.

[40] D. Moro, N. Caviana, and F. Ponti. In-cylinder pressure reconstruction based on instantaneous engine speed signal. *ASME Journal of Engineering for Gas Turbines and Power*, 124:220–225, January 2002.

[41] B. Lee, G. Rizzoni, Y. Guezennec, A. Soliman, M. Cavalletti, and J. Water. Engine control using torque estimation. *SAE Technical Paper*, 2001-1-0995, 2001.

[42] D. Taraza, N. A. Henein, and W. Bryzik. The frequency analysis of the crankshaft's speed variation: A reliable tool for diese engine diagnosis. *ASME Journal of Engineering for Gas Turbines and Power*, 123:428–432, 2001.

[43] D. Taraza. Statistical correlation between the crankshaft's speed variation and engine performance - parti: Theoretical model. *ASME Journal of Engineering for Gas Turbines and Power*, 125:791–796, 2003.

[44] D. Taraza. Statistical correlation between the crankshaft's speed variation and engine performance-partii: Detection of deficient cylinders and mean indicated pressure calculation. *ASME Journal of Engineering for Gas Turbines and Power*, 125:797–803, 2003.

[45] P. Zeng and D. N. Assanis. Cylinder pressure reconstruction and its application to heat transfer analysis. *SAE Technical Paper*, 2004-1-0922, 2004.

[46] P. J. Jacob, F. Gu, and A. D. Ball. Non-parametric models in the monitoring of engine performance and condition part1: modelling of non-linear engine process. *Proceedings of the I MECH E Part D Journal of Automobile Engineering*, 213(D):73–81, 1999.

[47] F. Gu, P. J. Jacob, and A. D. Ball. Non-parametric models in the monitoring of engine performance and condition part2: non-intrusive estimation of diesel engine cylinder pressure and its use in fault detection. *Proceedings of the I MECH E Part D Journal of Automobile Engineering*, 213(D):135–143, 1999.

[48] R. Potenza, J. F. Dunne, S. Vulli, D. Richardson, and P. King. Multicylinder engine pressure reconstruction using narx neural networks and crank kinematics. *International Journal of Engine Research*, 8:499–518, 2007.

[49] R. Johnsson. Cylinder pressure reconstruction based on complex radial basis function networks from vibration and speed signals. *Mechanical Systems and Signal Processing,*, 20(8):1923–1940, 11 2006.

[50] S. Singh and R. D. Reitz. Comparison of characteristic time (ctc), representative interactive flamelet (rif) and direct integration with detailed chemistry combustion models

agains multi-mode combustion in a heavy-duty di diesel engine. *SAE Technical Paper*, 2006-01-0055, 2006.

[51] S. Singh, R. D. Reitz, M. P. B. Musculus, and T. Lachaux. Validation of engine combustion models against detailed in-cylinder optical diagnostics data for a heavy-duty compression-ignition engine. *International Journal of Engine Research*, 8:97–126, 2007.

[52] E. G. Pariotis and D. T. Hountalas. Validation of a newly developed quasi-dimensional combustion model - application on a heavy duty di diesel engine. *SAE Technical Paper*, 2004-01-0923, 2004.

[53] O. Grondin, J. Maquet, R. Stobart, and H. Chafouk. Compression ignition engine simulator for instantaneous pressure and torque generation. 6-8 October 2004.

[54] K. Allmendinger, L. Guzzella, A. Seiler, and O. Loffeld. A method to reduce the calculation time for a internal combustion engine model. *SAE Technical Paper*, 2001-01-0574, 2001.

[55] L. Eriksson and I. Andersson. An analytical model for cylinder pressure in a four stroke si engine. *SAE Technical Paper*, 2002-01-0371, 2002.

[56] G. Chen. Prediction of peak cylinder pressure variations over varying inlet air condition of compression ignition engine. *Journal of Engineering for Gas Turbines and Power*, 129: 589–595, 2007.

[57] Y. H. Zweiri, J. F. Whidborne, and L. D. Seneviratne. Detailed analytical model of a single-cylinder diesel engine in the crank angle domain. *Proceedings of the I MECH E Part D Journal of Automobile Engineering*, 215(D):1197–1215, 2001.

[58] S. M. A. Burney, T. A. Jilani, and C. Ardil. A comparison of first and second order training algorithms for artificial neural networks. *Engineering and Technology*, 1:9–15, 2005.

[59] S. Haykin. *Neural Networks - A Comprehensive Foundation*. Prentice Hall, Upper Saddle River, 2 edition, 1999. ISBN 0132733501.

[60] S. A. Kalogirou. Artificial intelligence for the modeling and control of combustion processes: a review. *Progress in Energy and Combustion Science*, 29:515–566, 2003.

[61] M. S. Ozerdem and S. Kolukisa. Artificial neural network approach to predict the mechanical properties of cu sn pb zn ni cast alloys. *Materials and Design*, 30(3):764–769, 2009.

[62] M. T. Hagan, H. B. Demuth, and M. Beale. *Neural Network Design*. PWS Publishing Company, 1999.

[63] B. Maass, R. K. Stobart, and J. Deng. Diesel engine emissions prediction using parallel neural networks. *Proceedings of 2009 American Control Conference*, 2009.

[64] A.J.C. Sharkey. *Combining Artificial Neural Nets: ensemble and modular multi net systems*, volume 1. Springer, London, 1999.

[65] M. I. Soumelidis and J. F. Dunne. Modular neural network modelling of simple powertrain systems. In *Int. Conf. on Total Vehicle Technology*, volume 3, pages 297–308, 2004.

[66] W. Guoyin. Parallel neural network architectures, 1995. ID: 1.

[67] A. J. C. Sharkey and N. E. Sharkey. Combining diverse neural nets. *The Knowledge Engineering Review*, 12(3):231–247, 1997.

[68] A. J. C. Sharkey, N. E. Sharkey, and G. O. Chandroth. Diverse neural net solutions to a fault diagnosis problem. *Neural Computing & Applications*, 4:218–227, 1996.

[69] A. J. C. Sharkey, G. O. Chandroth, and N. E. Sharkey. A multi-net system for the fault diagnosis of a diesel engine. *Neural Computing & Applications*, 9:152–160, 2000.

[70] B. Lee. Parallel neural networks for speech recognition. In *Proc. of Intern. Joint Conf. NN*, pages 2093–2097. IEEE, 1997.

[71] P. J. Werbos. Backpropagation through time: What it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560, October 1990.

[72] M. T. Hagan and M. B. Menhaj. Training feedforward networks with the marquardt algorithm. *IEEE Transactions on Neural Networks*, 5:989–993, 1994.

[73] D.R. Jones, C.D. Perttunen, and B.E. Stuckman. Lipschitzian optimization without the lipschitz constant. *Journal of Optimization Theory and Application*, 79(1):157–181, 1993.

[74] D. Whitley. A genetic algorithm tutorial, 1993.

[75] M. Hafner, M. Schüler, O. Nelles, and R. Isermann. Fast neural networks for diesel engine control design. *Control Engineering Practice*, 8:1211–1221, 2000.

[76] M. Ouladsine, G. Bloch, and X. Dovifaaz. Neural modelling and control of a diesel engine with pollution constraints. *Journal of Intelligent and Robotic Systems*, 41:157–171, 2004.

[77] G. Thompson, C. Atkinson, N. Clark, T. Long, and E. Hanzevack. Neural network modelling of the emissions and performance of a heavy-duty diesel engine. In *Proc. Instn. Mech. Engrs*, volume 214, 2000.

[78] T. Winsel, M. Ayeb, D. Lichtenthäler, and H. J. Theuerkauf. A neural estimator for cylinder pressure and engine torque. *SAE Technical Paper*, 1999-01-1165, 1999.

[79] Y. He and C.J. Rutland. Applications of artificial neural networks in engine modelling. *Int. J. Engine Res.*, 5(4):281–296, 2004.

[80] B. Maass, R. K. Stobart, and J. Deng. Prediction of nox emissions of a heavy duty diesel engine with a nlarx model. *SAE Powertrains, Fuels and Lubricants Meeting*, 2009.

[81] J. B. Heywood. *Internal Combustion Engine Fundamentals*, volume 1. McGraw-Hill, Singapore, 1988. ISBN 0071004998.

[82] J. Deng, E. Winward, R. Stobart, and P.R. Desai. Modeling techniques, to support fuel path control in medium duty diesel engines. *SAE Technical Paper*, 2010-01-0332, 2010.

[83] K. K. Kuo. *Principles of Combustion*, volume 2. John Wiley & Sons, Inc., 2005.

[84] J. Warnatz, U. Maas, and R. W. Dibble. *Combustion - Physical and Chemical Fundamentals, Modeling and Simulation, Experiment, Pollutant Formation*, volume 3. Springer, 2001.

[85] Gamma Technologies Inc. Gt-suite. Internet, August 2010. URL `http://www.gtisoft.com/`.

[86] K. Funahashi. On the approximate realization of continuous mappings by neural networks. *Neural Networks*, 2:183–192, 1989.

# Appendices

# A Continuous Speed-Load Acceptance

***Table A.1:*** *Constant speed-load acceptance cycle with ramp times, speed and torque values*

| Speed [RPM] | Start Torque [Nm] | Target Torque [Nm] | Comment |
|---|---|---|---|
| 1000 | 50 | 700 | Ramp up 10s - hold 1min |
| 1000 | 700 | 50 | Ramp down 10s |
| 1200 | 50 | 50 | Ramp to next speed point with 50 Nm |
| 1200 | 50 | 900 | Ramp up 6s - hold 1min |
| 1200 | 900 | 50 | Ramp down 6s |
| 1400 | 50 | 50 | Ramp to next speed point with 50 Nm |
| 1400 | 50 | 900 | Ramp up 4s - hold 1min |
| 1400 | 900 | 50 | Ramp down 4s |
| 1600 | 50 | 50 | Ramp to next speed point with 50 Nm |
| 1600 | 50 | 885 | Ramp up 3s - hold 1min |
| 1600 | 885 | 50 | Ramp down 3s |
| 1800 | 50 | 50 | Ramp to next speed point with 50 Nm |
| 1800 | 50 | 805 | Ramp up 2s - hold 1min |
| 1800 | 805 | 50 | Ramp down 2s |
| 2000 | 50 | 50 | Ramp to next speed point with 50 Nm |
| 2000 | 50 | 715 | Ramp up 2s - hold 1min |
| 2000 | 715 | 50 | Ramp down 2s |
| 2200 | 50 | 50 | Ramp to next speed point with 50 Nm |
| 2200 | 50 | 620 | Ramp up 2s - hold 1min |
| 2200 | 620 | 50 | Ramp down 2s |
| 2200 | 50 | 620 | No speed change - ramp torque up in 2s |
| 2200 | 620 | 50 | Ramp down 2s |
| 2000 | 50 | 50 | Ramp to next speed point with 50 Nm |
| 2000 | 50 | 715 | Ramp up 2s - hold 1min |
| 2000 | 715 | 50 | Ramp down 2s |
| 1800 | 50 | 50 | Ramp to next speed point with 50 Nm |
| 1800 | 50 | 805 | Ramp up 2s - hold 1min |
| 1800 | 805 | 50 | Ramp down 2s |
| 1600 | 50 | 50 | Ramp to next speed point with 50 Nm |
| 1600 | 50 | 885 | Ramp up 3s - hold 1min |
| 1600 | 885 | 50 | Ramp down 3s |
| 1400 | 50 | 50 | Ramp to next speed point with 50 Nm |
| 1400 | 50 | 900 | Ramp up 4s - hold 1min |
| 1400 | 900 | 50 | Ramp down 4s |
| 1200 | 50 | 50 | Ramp to next speed point with 50 Nm |
| 1200 | 50 | 900 | Ramp up 6s - hold 1min |
| 1200 | 900 | 50 | Ramp down 6s |
| 1000 | 50 | 50 | Ramp to next speed point with 50 Nm |
| 1000 | 50 | 700 | Ramp up 10s - hold 1min |
| 1000 | 700 | 50 | Ramp down 10s |

# B Results for GT-Power Modelling

## B.1 Training and Validation Sets for GT-Power Modelling - Cycle Arrangement

***Table B.1:*** *Arrangement of training set for GT-Power Modelling - Load and Speed cases and their arrangement within the training set shown in figure 6.1*

| Torque/Speed | 800 RPM | 1400 RPM | 2200 RPM |
|---|---|---|---|
| 0 % | Cycle 1 | 7 | 9 |
| 10 % | | | |
| 20 % | Cycle 4 | 15 | 6 |
| 30 % | | | |
| 40 % | Cycle 8 | 5 | 3 |
| 50 % | | | |
| 60 % | Cycle 10 | 14 | 11 |
| 70 % | Cycle 12 | 2 | 13 |

***Table B.2:*** *Arrangement of validation set for GT-Power Modelling - Load and Speed cases and their arrangement within the validation set shown in figure 6.2*

| Torque/Speed | 800 RPM | 1200 RPM | 1400 RPM | 1800 RPM | 2200 RPM |
|---|---|---|---|---|---|
| 0 % | Cycle 25 | 24 | 23 | 22 | 21 |
| 10 % | Cycle 18 | | 19 | | 20 |
| 20 % | Cycle | 17 | | 16 | |
| 30 % | Cycle 13 | | 14 | | 15 |
| 40 % | Cycle | 12 | | 11 | |
| 50 % | Cycle 8 | | 9 | | 10 |
| 60 % | Cycle | 7 | | 6 | |
| 70 % | Cycle 1 | 2 | 3 | 4 | 5 |

## B.2 Training and Validation Results for GT-Power Modelling - Network Topologies

***Table B.3:*** *Results for GT-Power Modelling - Results for all tested network topologies for pressure and temperature prediction with the FFN*

Results for GT-Power Pressure Modelling with FFN

| FFN with 6 inputs | | | | FFN with 7 inputs | | |
|---|---|---|---|---|---|---|
| | | Results $\mathrm{R}^2$ | | | Results $\mathrm{R}^2$ | |
| Layer | Neurons | Training | Validation | Neurons | Training | Validation |
| 3 | [4 4 4] | 0.94 | 0.75 | [4 4 4] | 0.97 | 0.97 |
| | [10 10 10] | 0.95 | 0.6 | [10 10 10] | 0.98 | 0.58 |
| | [15 15 15] | 0.91 | 0.37 | [15 15 15] | 0.99 | 0.98 |
| | [20 20 20] | 0.92 | 0.65 | [20 20 20] | 0.99 | 0.98 |
| | [25 25 25] | 0.96 | 0.32 | [25 25 25] | 0.99 | 0.99 |
| 2 | [4 4] | 0.78 | 0.75 | [4 4] | 0.97 | 0.97 |
| | [10 10] | 0.93 | 0.54 | [10 10] | 0.99 | 0.99 |
| | [20 20] | 0.95 | 0.0426 | [20 20] | 0.99 | 0.99 |
| | [25 25] | 0.93 | 0.48 | [25 25] | 0.99 | 0.99 |

Results for GT-Power Temperature Modelling with FFN

| FFN with 6 inputs | | | | FFN with 7 inputs | | |
|---|---|---|---|---|---|---|
| | | Results $\mathrm{R}^2$ | | | Results $\mathrm{R}^2$ | |
| Layer | Neurons | Training | Validation | Neurons | Training | Validation |
| 3 | [4 4 4] | 0.95 | 0.88 | [4 4 4] | 0.99 | 0.98 |
| | [10 10 10] | 0.98 | 0.86 | [10 10 10] | 0.99 | 0.98 |
| | [15 15 15] | 0.98 | 0.8 | [15 15 15] | 0.99 | 0.99 |
| | [20 20 20] | 0.98 | 0.75 | [20 20 20] | 0.99 | 0.99 |
| | [25 25 25] | 0.99 | 0.57 | [25 25 25] | 0.99 | 0.97 |
| 2 | [4 4] | 0.63 | 0.62 | [4 4] | 0.52 | 0.52 |
| | [10 10] | 0.96 | 0.85 | [10 10] | 0.99 | 0.99 |
| | [20 20] | 0.92 | 0.83 | [20 20] | 0.99 | 0.98 |
| | [25 25] | 0.98 | 0.73 | [25 25] | 0.99 | 0.97 |

***Table B.4:*** *Results for GT-Power Modelling - Results for all tested network topologies for pressure and temperature prediction with the FFNTD*

Results for GT-Power Pressure Modelling with FFNTD

| FFNTD with 6 inputs | | | | FFNTD with 7 inputs | | |
|---|---|---|---|---|---|---|
| | | Results $\mathrm{R}^2$ | | | Results $\mathrm{R}^2$ | |
| Layer | Neurons | Training | Validation | Neurons | Training | Validation |
| 3 | [4 4 4] | 0.72 | 0.7 | [4 4 4] | 0.99 | 0.94 |
| | [10 10 10] | 0.86 | 0.73 | [10 10 10] | 0.98 | 0.96 |
| | [15 15 15] | 0.92 | 0.62 | [15 15 15] | 0.99 | 0.99 |
| | [20 20 20] | 0.93 | 0.69 | [20 20 20] | 0.99 | 0.97 |
| | [25 25 25] | 0.9 | 0.65 | [25 25 25] | 0.99 | 0.94 |
| 2 | [4 4] | 0.8 | 0.77 | [4 4] | 0.99 | 0.99 |
| | [10 10] | 0.87 | 0.69 | [10 10] | 0.98 | 0.97 |
| | [15 15] | 0.85 | 0.71 | [15 15] | 0.96 | 0.88 |
| | [20 20] | 0.91 | 0.71 | [20 20] | 0.99 | 0.99 |
| | [25 25] | 0.91 | 0.59 | [25 25] | 0.99 | 0.94 |

Results for GT-Power Temperature Modelling with FFNTD

| FFNTD with 6 inputs | | | | FFNTD with 7 inputs | | |
|---|---|---|---|---|---|---|
| | | Results $\mathrm{R}^2$ | | | Results $\mathrm{R}^2$ | |
| Layer | Neurons | Training | Validation | Neurons | Training | Validation |
| 3 | [4 4 4] | 0.86 | 0.82 | [4 4 4] | 0.99 | 0.98 |
| | [10 10 10] | 0.95 | 0.85 | [10 10 10] | 0.99 | 0.98 |
| | [15 15 15] | 0.94 | 0.82 | [15 15 15] | 0.99 | 0.99 |
| | [20 20 20] | 0.96 | 0.85 | [20 20 20] | 0.99 | 0.99 |
| | [25 25 25] | 0.97 | 0.83 | [25 25 25] | 0.99 | 0.97 |
| 2 | [4 4] | 0.87 | 0.81 | [4 4] | 0.52 | 0.52 |
| | [10 10] | 0.94 | 0.89 | [10 10] | 0.99 | 0.99 |
| | [15 15] | 0.95 | 0.88 | [15 15] | 0.99 | 0.98 |
| | [20 20] | 0.98 | 0.84 | [20 20] | 0.99 | 0.98 |
| | [25 25] | 0.98 | 0.73 | [25 25] | 0.99 | 0.99 |

***Table B.5:*** *Results for GT-Power Modelling - Results for all tested network topologies for pressure and temperature prediction with the NLARX*

Results for GT-Power Pressure Modelling with NLARX

| NLARX with 6 inputs | | | | NLARX with 7 inputs | | |
|---|---|---|---|---|---|---|
| | | Results $\mathrm{R}^2$ | | | Results $\mathrm{R}^2$ | |
| Layer | Neurons | Training | Validation | Neurons | Training | Validation |
| 3 | [4 4 4] | 0.98 | 0.98 | [4 4 4] | 0.99 | 0.99 |
| | [10 10 10] | 0.99 | 0.99 | [10 10 10] | 0.97 | 0.97 |
| | [15 15 15] | 0.99 | 0.99 | [15 15 15] | 0.83 | 0.83 |
| | [20 20 20] | 0.89 | 0.89 | [20 20 20] | 0.99 | 0.99 |
| | [25 25 25] | 0.99 | 0.99 | [25 25 25] | 0.9 | 0.9 |
| 2 | [4 4] | 0.99 | 0.99 | [4 4] | 0.99 | 0.99 |
| | [10 10] | 0.99 | 0.99 | [10 10] | 0.99 | 0.99 |
| | [15 15] | 0.99 | 0.99 | [15 15] | 0.99 | 0.99 |
| | [20 20] | 1 | 1 | [20 20] | 1 | 1 |
| | [25 25] | 1 | 1 | [25 25] | 1 | 0.99 |

Results for GT-Power Temperature Modelling with NLARX

| NLARX with 6 inputs | | | | NLARX with 7 inputs | | |
|---|---|---|---|---|---|---|
| | | Results $\mathrm{R}^2$ | | | Results $\mathrm{R}^2$ | |
| Layer | Neurons | Training | Validation | Neurons | Training | Validation |
| 3 | [4 4 4] | 0.99 | 0.99 | [4 4 4] | 0.99 | 0.99 |
| | [10 10 10] | 0.99 | 0.99 | [10 10 10] | 0.99 | 0.99 |
| | [15 15 15] | 1 | 0.99 | [15 15 15] | 1 | 0.99 |
| | [20 20 20] | 0.99 | 0.99 | [20 20 20] | 1 | 0.99 |
| | [25 25 25] | 1 | 0.99 | [25 25 25] | 1 | 0.97 |
| 2 | [4 4] | 0.99 | 0.99 | [4 4] | 0.99 | 0.99 |
| | [10 10] | 0.99 | 0.99 | [10 10] | 1 | 0.99 |
| | [15 15] | 0.99 | 0.99 | [15 15] | 1 | 0.99 |
| | [20 20] | 0.99 | 0.99 | [20 20] | 1 | 0.99 |
| | [25 25] | 0.99 | 0.99 | [25 25] | 1 | 0.99 |

# C Results for Real Engine Modelling

## C.1 Training and Validation Sets for Real Engine Modelling-Cycle Arrangement

***Table C.1:*** *Arrangement of training set for Real Engine Modelling - Load and Speed cases and their arrangement within the training set shown in figure 7.2*

| Torque/Speed | 800 RPM | 1400 RPM | 2200 RPM |
|---|---|---|---|
| 0 % | Cycle 1 | 4 | 18 |
| 0-10 % | Cycle | 15 | |
| 10 % | Cycle 7 | | |
| 10-20 % | Cycle | | 6 |
| 20 % | Cycle | 3 | |
| 20-30 % | Cycle 14 | | |
| 30 % | Cycle | | 16 |
| 30-40 % | Cycle | 12 | |
| 40 % | Cycle 8 | 5 | 3 |
| 40-50 % | Cycle | | 13 |
| 50 % | Cycle | 19 | |
| 50-60 % | Cycle 10 | | |
| 60 % | Cycle | | 11 |
| 60-70 % | Cycle | 8 | |
| 70 % | Cycle 9 | 17 | 2 |

***Table C.2:*** *Arrangement of validation set for real engine modelling - Load and Speed cases and their arrangement within the validation set shown in figure 7.3*

| Torque/Speed | 800 RPM | 1200 RPM | 1400 RPM | 1800 RPM | 2200 RPM |
|---|---|---|---|---|---|
| 0 % | Cycle 33 | 32 | 31 | 30 | 19 |
| 0-10 % | Cycle 27 | | | | 28 |
| 10 % | Cycle | 26 | | 25 | |
| 10-20 % | Cycle | | 24 | | |
| 20 % | Cycle | 23 | | 22 | |
| 20-30 % | Cycle 20 | | | | 21 |
| 30 % | Cycle | 19 | | 18 | |
| 30-40 % | Cycle | | 17 | | |
| 40 % | Cycle | 16 | | 15 | |
| 40-50 % | Cycle 13 | | | | 14 |
| 50 % | Cycle | 12 | | 11 | |
| 50-60 % | Cycle | | 10 | | |
| 60 % | Cycle | 9 | | 8 | |
| 60-70 % | Cycle 6 | | | | 7 |
| 70 % | Cycle 5 | 4 | 3 | 2 | 1 |

## C.2 Training and Validation Results for Real Engine Modelling - Network Topologies

***Table C.3:*** *Results for real engine modelling - Results for all tested network topologies for pressure and temperature prediction with the NLARX*

| Results for Real Engine Pressure Modelling with FFN | | | |
|---|---|---|---|
| FFN with 8 inputs | | | |
| | | Results $\mathrm{R}^2$ | |
| Layer | Neurons | Training | Validation |
| 2 | [4 4] | 0.99 | 0.99 |
| | [10 10] | 0.99 | 0.99 |
| | [15 15] | 0.99 | 0.99 |
| | [20 20] | 0.99 | 0.99 |

| Results for Real Engine Temperature Modelling with FFN | | | |
|---|---|---|---|
| NLARX with 8 inputs | | | |
| | | Results $\mathrm{R}^2$ | |
| Layer | Neurons | Training | Validation |
| 2 | [4 4] | 0.99 | 0.99 |
| | [10 10] | 0.99 | 0.99 |
| | [15 15] | 0.99 | 0.99 |
| | [20 20] | 0.99 | 0.99 |

***Table C.4:*** *Results for Real Engine Modelling - Results for all tested network topologies for pressure and temperature prediction with the NLARX*

Results for Real Engine Pressure Modelling with NLARX

NLARX with 7 inputs

| | | Results $\mathrm{R}^2$ | |
|---|---|---|---|
| Layer | Neurons | Training | Validation |
| 2 | [4 4] | 0.99 | 0.99 |
| | [10 10] | 0.99 | 0.99 |
| | [15 15] | 0.99 | 0.99 |
| | [20 20] | 0.99 | 0.99 |

Results for Real Engine Temperature Modelling with NLARX

NLARX with 7 inputs

| | | Results $\mathrm{R}^2$ | |
|---|---|---|---|
| Layer | Neurons | Training | Validation |
| 2 | [4 4] | 0.99 | 0.99 |
| | [10 10] | 0.99 | 0.99 |
| | [15 15] | 0.99 | 0.99 |
| | [20 20] | 0.99 | 0.99 |