Predictive Control of a Hybrid Diesel-Electric Marine Propulsion Plant

Nikolaos Planakis

Diploma Thesis

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

Supervisor: Assistant Prof. George Papalambrou

Committee Member : Prof. N. Kyrtatos
Committee Member : Prof. K. Kyriakopoulos

October 2016

Aknowledgements

This work has been carried out at the Laboratory of Marine Engineering (LME) at the School of Naval Architecture and Marine Engineering of the National Technical University of Athens, under the supervision of Assistant Professor George Papalambrou.

I would like thank Professor Nikolaos Kyrtatos for providing the opportunity to work with the full-scale hybrid diesel-electric marine propulsion powertrain of LME. Access to LME experimental facilities was essential in order to verify theoretical concepts from a practical perspective and evaluate experimentally the designed control system. I also thank him for being a member of my supervisors committee.

I owe my greatest appreciation to my thesis supervisor Assistant Professor George Papalambrou, for giving me the chance and motivation to work on this topic. I would also like to thank him for his patience, continuous support and immense knowledge. His guidance helped me in all the time working on this thesis.

I would also like to thank Professor Konstantinos Kyriakopoulos for evaluating my work and being a member of my supervisors committee.

I would like to express my sincere gratitude to Mr. Sotirios Topaloglou, researcher of LME for his contribution to the experimental validation of control system performance. I am extremely thankful to him for sharing expertise, and sincere and valuable guidance and encouragement extended to me.

I take this opportunity to express gratitude to all LME fellow members for their help and support.

I am also sincerely grateful to my family, for the unceasing encouragement, support, motivation and attention throughout my studies.

I also place on record, my sense of gratitude to one and all, who directly or indirectly, have lemt their hand in this venture. This accomplishment would not have been possible without them. Thank you.

Abstract

In this thesis, the feasibility of using model predictive control in a hybrid diesel-electric marine powerplant is investigated. Initially, the modeling procedure is divided into two parts. The aim of first part is to find linear models using datasets with experimental data at the nominal operation point of the hybrid powertrain and investigate the adoption of different signals considered as system disturbances, as the controller is capable to cope with multi-variable systems. Second part aims to the development of a non-linear multivariable model which can describe the hybrid powertrain behavior over a wider range of operation. Finally, a output feedback manipulation is suggested in order to enable a better prediction of the future model outputs.

Model Predictive Controller (MPC) is a model-based controller which tries to compute the optimal sequence of the control moves in order to succeed the optimal control performance of a plant over a finite prediction horizon. MPC is capable of tackling multi-variable processes, satisfy input and output system constraints, deal with long time delays and utilize plant response to measured and unmeasured disturbances knowledge. Performance evaluation of the designed MPC controllers was firstly conducted through step response simulation against each controller's internal model, in order to fine-tune its parameters and ensure the system stability. As a next step, the controllers were evaluated through simulation using open-loop experimental data and the non-linear diesel engine model, using also the electric motor system model for the total power split determination. Last, the performance of the various MPC controllers was experimentally verified on the hybrid propulsion powertrain at LME. MPC response was tested at various load profiles, including alternating and propeller load, against static and dynamic reference tracking, evaluating disturbance rejection and efficiency in operating the plant within the desired exhaust emission and fuel consumption limits during closed-loop control.

Contents

Chapter 1

Introduction

Framework

Diesel engines are present to 99 % of the engine installations of the ships. During recent years, facts as the stringent legal standards and the increasing environmental concern, as well as the continuous desire for better engine performance, have forced manufacturers of marine engines to implement new technologies related to the propulsion power units of ships, aiming at lower exhaust emission and enhanced power management. This was particularly made possible due new engine components and technologies that were introduced in marine powertrains, such as multiple- stage turbo charging, variable geometry turbochargers (VGT), exhaust gas recirculation (EGR), etc. One attractive solution is the diesel-electric hybrid powertrain, where the internal combustion engine is assisted by an electric motor. Hybrid electric solutions have been used massively in automotive installations; however, they have limited application in shipping industry so far. Commercially, one such solution has been developed by CATERPILLAR and was recently launched in the market[1]. CATERPILLAR hybrid diedel-electric propulsion powertrain is shown in image 1.1.

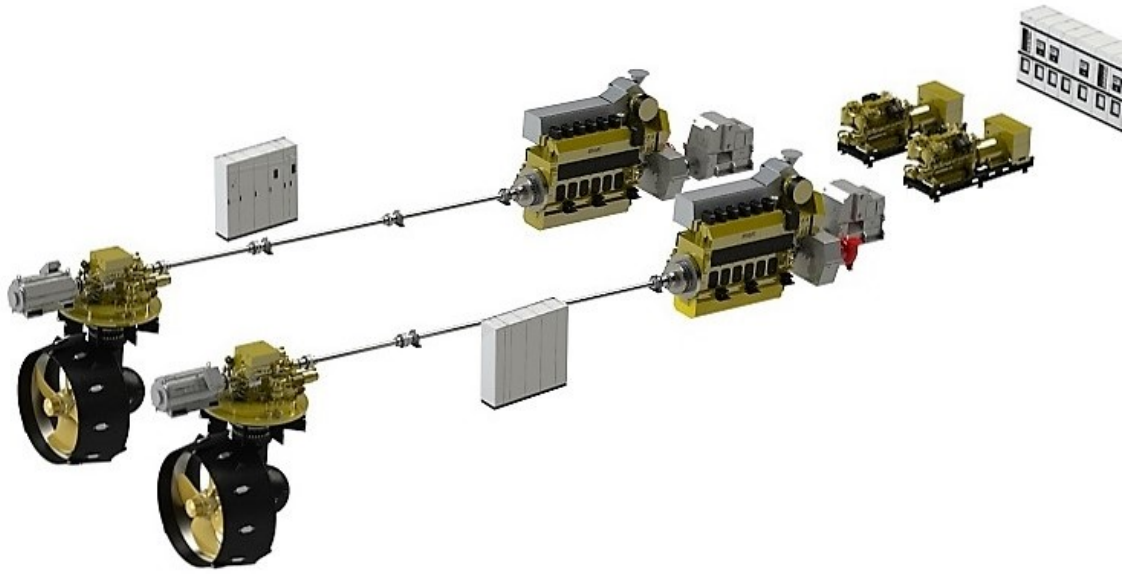

Figure 1.1: CATERPILLAR commercial hybrid diesel-electric propulsion powerplant.

[1] http://www.cat.com/en_MX/news/engine-press-releases

Different types of vessels operate in various conditions. For example, marine powerplants in cargo vessels usually operate in steady state sea going condition. Thus, some ship types operate in coastal areas and within port range, such as passenger ships, yachts, tugs, special purpose vessels, etc. The power units of these ships operate at various load points, which may be out of their high performance range. Consequently, they maintain a lower overall operational performance index.

The most aggressive operational circumstance of a diesel, in terms of exhaust emission and fuel consumption, is the transient load operation. During load transients, all the engine variables change continuously, deterring the engine from its equilibrium point. During this phenomenon, the exhaust gas quality is drastically affected, mainly due to the delayed response of the turbo-charger to provide the sufficient amount of air on time. As a result of the lean combustion, λ value, which represents the air-to-fuel ratio in the cylinder to stoichiometric air-to-fuel ratio drops with a consequent rise of the pollutant emissions of the engine. A hybrid diesel-electric configuration could be used so as to assist the engine operation during transient loading conditions and enhance the total performance index of the powerplant.

Literature Review

In such hybrid installations, the main area of interest is their control strategy, ie the decision for the power split between the powertrain components as a function of many variables. Usually, the engine control units contain a certain amount of single closed-loops, with many look up tables in order to achieve closed-loop control of the multi-parametric and strongly non-linear engine behavior [19, 24]. To maximize fuel economy, control strategies are required to estimate the amount of energy to be produced. Power management of hybrid powerplants decides how much power shall be produced by the internal combustion engine and how much should be released by the electric motor so as to achieve the total power demand at the driving shaft of the propeller. In [14] it is shown how the use of multiple input- multiple output (MIMO) models in control procedure can enhance the engine operation of marine diesel powerplants. A multivariable control scheme is designed that reduces smoke generation on an marine diesel engine. Today, a more sophisticated and complicated control method is needed. One that continuously decides the operation point of the plant, while enforcing the operating constraints and optimizing the energy consumption, in terms of fuel and electric energy consumption.

Model based control design can emphatically provide a solution, which is truly multi-variable, more flexible and easier to adapt when engine configuration change [16]. Several strategies for power management have been so far applied, including dynamic programming (DP), stochastic dynamic programming (SDP), equivalent fuel consumption minimization (ECMS) and model predictive control (MPC). Of the many advanced control design methodologies, MPC is the most capable to handle multi-variable processes, satisfy constraints, deal with long time delays and utilize plant response to measured and unmeasured disturbances knowledge [19]. MPC embodies both receding horizon optimization and feedback adjustment. MPC has been used in a broad range of applications, such as diesel engine control, [1, 15], catalyst control [16, 23], Hybrid Electric Vehicles (HEV) [24, 25], Plug-in Hybrid Electric Vehicles (PHEV) [19, 22, 26], etc.

Thesis Structure

In this thesis, MPC configuration for the closed-loop control of the Hybrid-electric powertrain of Laboratory of Marine Engineering (LME) is investigated. Two control schemes were followed: The closed-loop control behavior is based in tracking a reference λ trajec-

tory - provided by look-up tables which utilize the engine measurements - under certain constraints of the electric motor actuator, affecting indirectly the fuel injection in the cylinders of the diesel engines. It has to be noted that the fueling system of the diesel engine is controlled by the engine electronic control unit separately.

λ value control is an alternative solution of emission quality control for two main reasons:

- The behavior of *lambda* dynamics can be relatively easy to be modeled and provides better information about the diesel engine performance [13] and
- emission content can be expressed as a function of λ, as shown in Fig. 1.2 [20]. λ was proven suitable for indicating nitrogen oxides (NOx) and Particulate Matter (PM) formation.

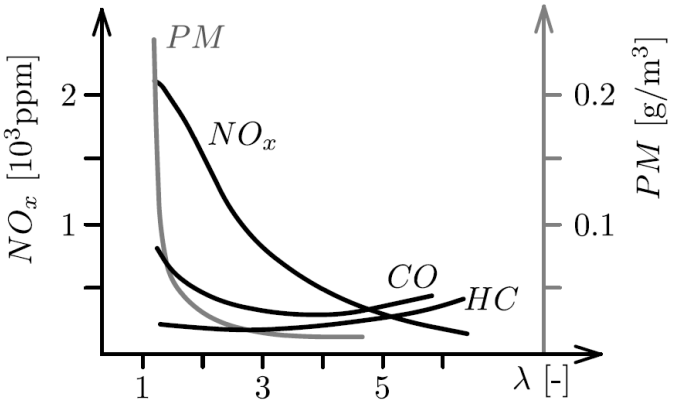

Figure 1.2: Emissions content as a function of λ. From [20]

.

In certain cases MPC has to handle output constrains on parameters like NOx emissions and fuel oil consumption in his optimization procedure. These are modeled as a function of λ and boost pressure values.

Finally, the λ reference is removed and MPC has to decide the power split in cosed-loop operation only under output constraints and access the operating of the plant without any reference trajectory to track.

MPC is capable of making a self-assessment of its performance, as a matter of a cost function optimization, which not only ensures the proper operation of the powerplant, but also succeeds the optimal use of the electric power consumption, which leads to a total enhancement of the performance index of the whole installation.

The structure of the thesis is as follows: in chapter 2 the experimental facility is presented, along with the installed engine sensors and the data acquisition system. chapter 3 describes the identification process; the derivation of models which describe the behavior of the powertain, based on experimental data. In chapter 4 the theoretical backround of model predictive controller is presented, while chapter 5 is dedicated to controller design and closed-loop performance verification. The experimental results are presented in chapter 6. Finally, the conclusions of this work are presented in chapter 7.

Chapter 2

Experimental Facility

The HIPPO-1 hybrid diesel-electric power plant consists of a internal combustion engine (ICE) in parallel connection to an electric motor (EM). In this configuration the rotational speed of the ICE and the EM are identical and the supplied torques add together to maintain the total torque demand applied by a hydrodynamic water brake (WB). In Fig. 2.1 and 2.2 the experimental hybrib powertrain of LME is presented, along with a schematic representation in AUTOCAD.

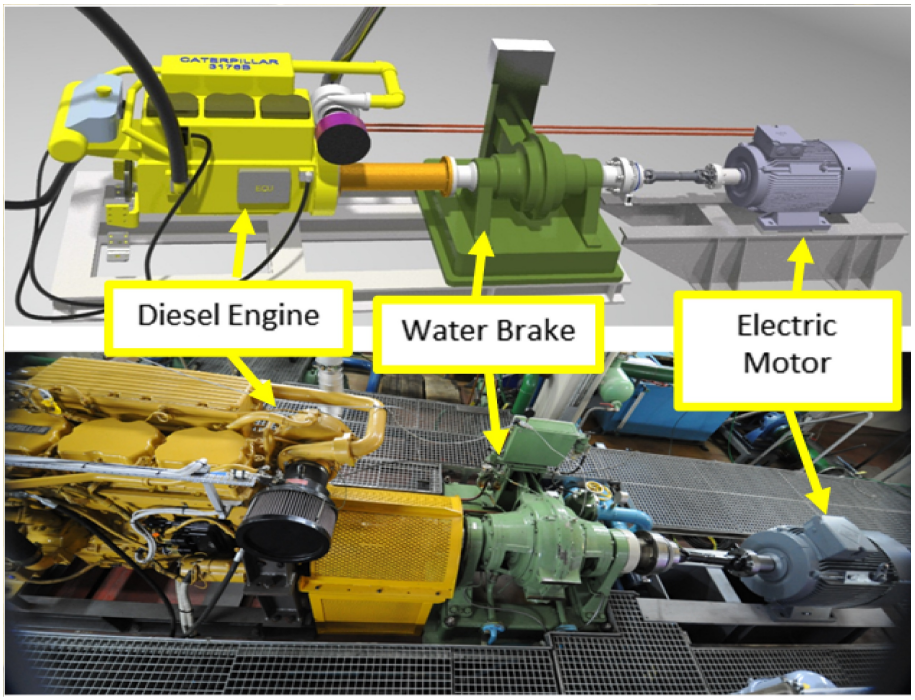

Figure 2.1: The HIPPO-1 hybrid diesel-electric testbed of LME.

2.1 Mechanical Componets

The ICE is a turbocharged CATERPILLAR 6-cylinder 10.3-liter 4-stroke marine diesel engine, model 3671B, producing 425 kW at 2300 rpm. According to the speed reference and the deviation of the speed measurement, the electronic control unit (ECU) of the ICE controls the fuel injection in the cylinders in closed loop control, using controller in the form of look-up tables. The installed sensors in the diesel engine are presented in Fig. 2.3

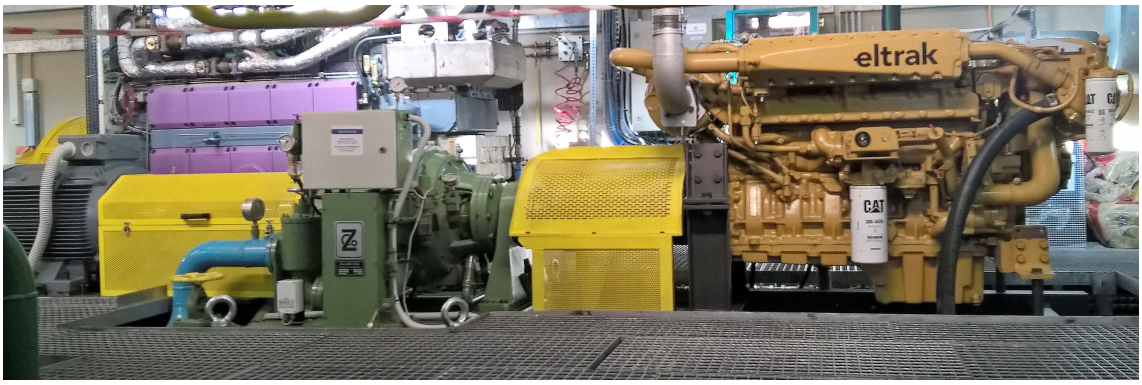

Figure 2.2: The HIPPO-1 hybrid diesel-electric testbed of LME. Between the internal combustion engine (right) and the electric motor (left) stands the water brake next to its controller board

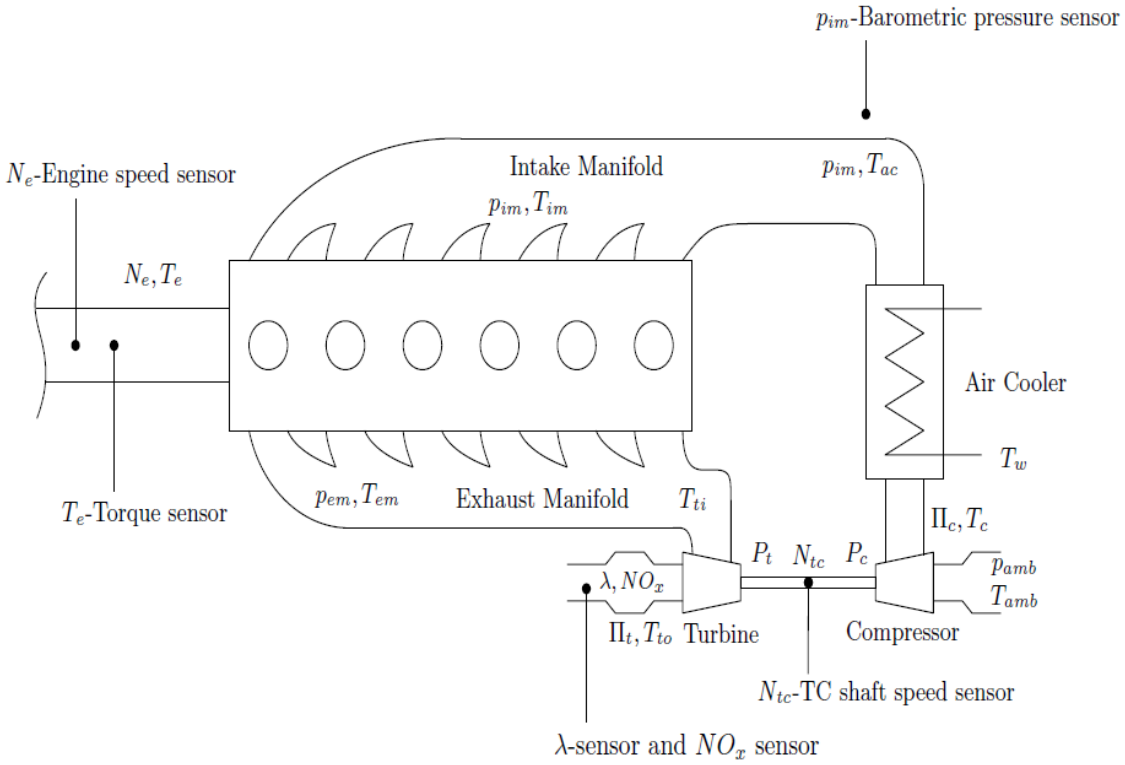

Figure 2.3: The HIPPO-1 diesel engined installed sensors.

The EM is a standard AC asychronous-induction 3-phase motor, with a rated power of 112 kW, type IE1-K21R 315 S4, manufactured by VEM. The electric torque output is regulated by a frequency inverter (Fr Inv) under closed loop control. The board of the frequency inverter is shown in picture 2.4.

The water brake of HIPPO-1 installation is manufactured by AVL Zoellner Gmbh, type 9n 38F, with 1200 kW load capacity, operating up to 4000 rpm. The water brake consists of two parts, the stator and the rotor, which is driven by the engine shaft. Between the two WB parts, the water level is regulated in order to produce the requested torque demand. The WB is controlled by a H_∞ controller designed at LME[1].

[1]C. Gkerekos. 2015. Experimental Modeling and Robust Controller Design for the Transient Loading

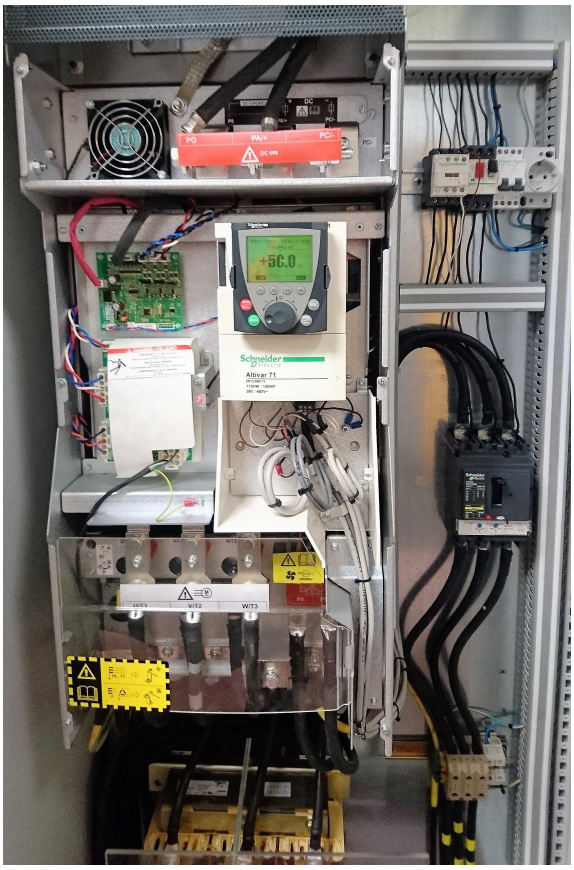

Figure 2.4: The EM frequency Inverter of HIPPO-1 EM at LME.

2.2 Sensors and Data Acquisition System

The installed sensors in the diesel engine are presented in Fig. 2.3. The NOx and λ values are provided by a *SmartNOx* sensor in the manifold downstream of the turbocharger (TC), manufactured by NGK. Exhaust gas opacity is measured by a AVL 439 opacimeter in the exhaust duct of the CAT engine. Fuel mass flow measurements are provided by two ABB Coriolis flow-meters, one at supply and one at return fuel lines. TC speed and intake manifold pressured are also measured.

The platform for the Data Acquisition and control of the powertrain is based on the dSpace DS1103 (Fig. 2.6) controller board, with rapid control prototyping capability, programmed under the MATLAB/Simulink environment.

A picture of the powertrain monitoring screen in the control room (Fig. 2.5) at LME is shown in Fig. 2.7, where all the utilities of the monitoring and the control board are presented.

of a Marine Diesel Engine. *Diploma Thesis*

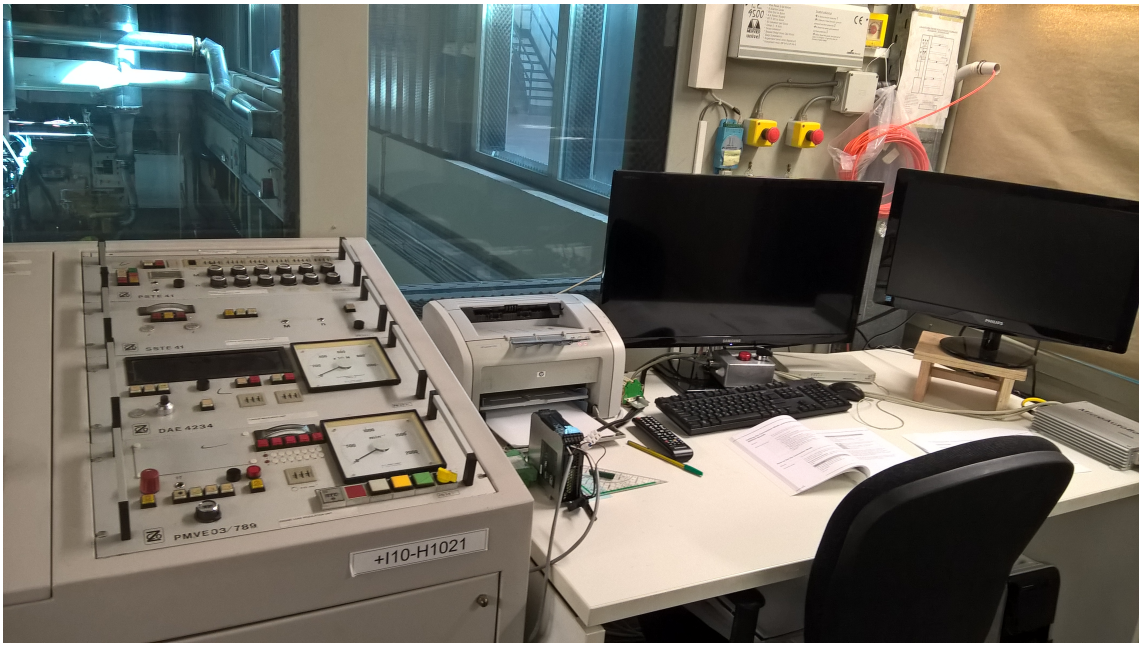

Figure 2.5: The engine control room. WB control board in front and the monitoring system of the hybrid plant on the right. Behind the safety glass, HIPPO-1 powertrain can be distinguished

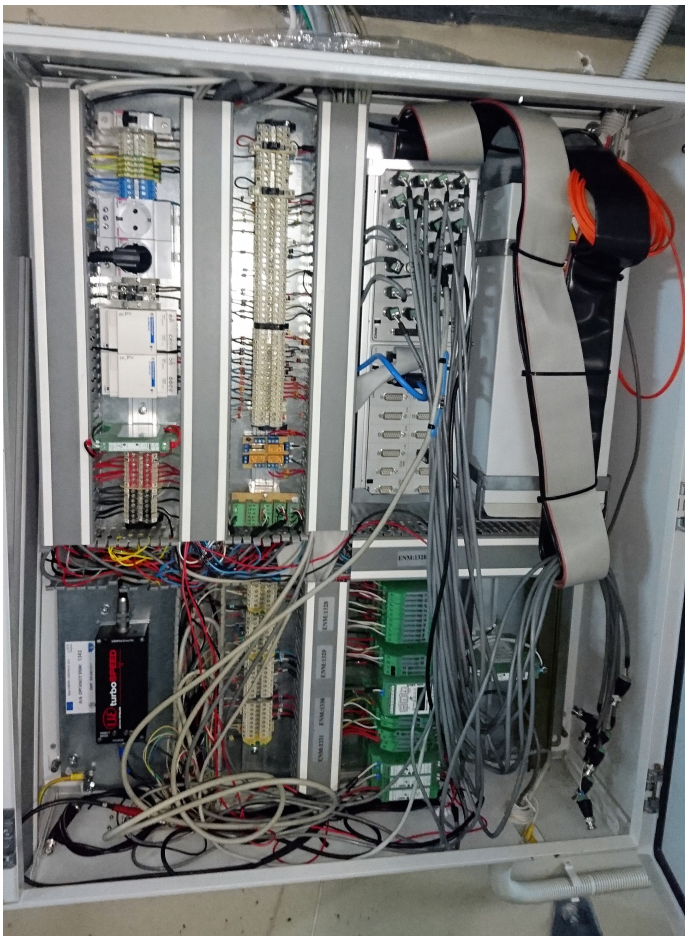

Figure 2.6: The HIPPO-1 dSpace monitoring and control board.

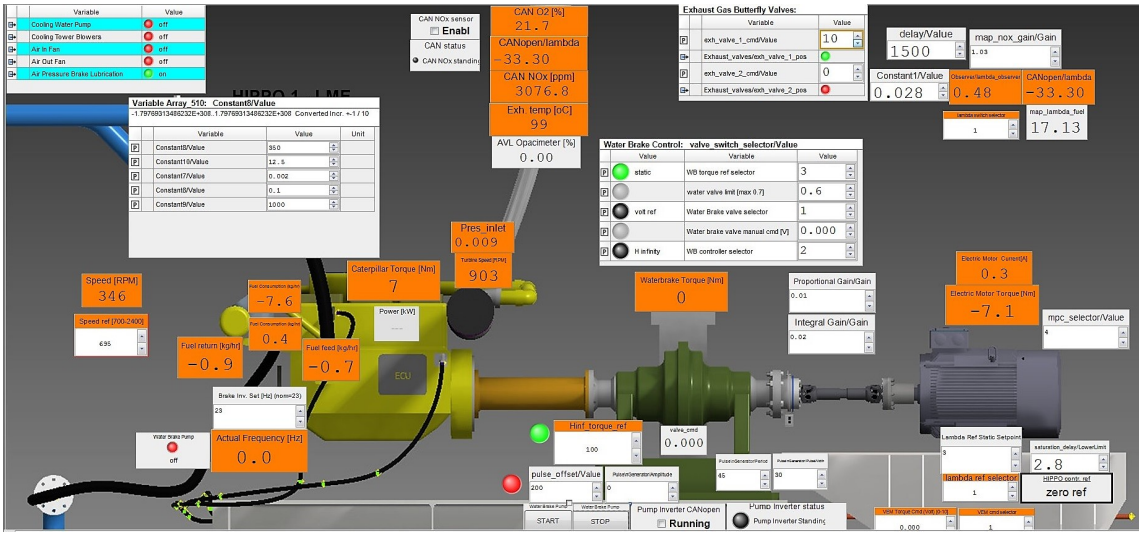

Figure 2.7: The HIPPO-1 monitoring and control screen.

Chapter 3

Modeling

3.1 Introduction

The main aim of the modeling procedure is two sided: to construct a mathematical description of the hybrid power plant behavior in such a way that the extracted models can be used in controller design and also develop a reliable simulation procedure, covering a wider operational range of the plant, in order to evaluate the controllers' design.

As the controller in use is a linear linear one, the modeling procedure was divided into two parts. The aim of part I is to find linear models using datasets with experimental data at the nominal operation point of the hybrid powertrain and investigate the adoption of different signals considered as system disturbances, as the controller is capable to cope with multi-variable systems. Part II aims to the development of a non-linear (nl) model which can describe the hybrid powertrain behavior over a wider range of operation in order to evaluate through simulation the performance of different closed-loop control scenarios.

Finally, a output feedback manipulation is suggested in order to enable a better prediction of the future model outputs.

The signal variables abbreviations are presented in Fig. 3.1 in tabular form.

ABBREVIATIONS	
VARIABLE	**SIGNAL**
λ	$\frac{\text{Air-to Fuel Ratio}}{\text{Stoichiometric Air-to Fuel Ratio}}$
FrInvCmd	Frequency Inverter Command
SE	Engine Speed
Tload	Water Brake Load Torque
Tdiesel	Diesel Engine Load Torque
Tel	Electric Motor Load Torque
MAP	Intake Manifold Pressure
NOx	Nitrogen Oxides
OPA	Exhaust Gas Opacity
FOC	Fuel Oil Consumption

Figure 3.1: Signal Abbreviations

3.2 Models used in Identification Process

Multiple kinds of models can describe the behavior of a plant. The main types of models which were used in this work are discrete time models time-invariant as follows [2]

- Linear State Space models (ss).
- Linear AutoRegressive models with eXogenous input (ARX).
- Non-linear AutoreGressive models with eXogenous input (NARX).

Linear State-Space Models

The most general state-space representation of a linear system with p inputs, q outputs and n state variables can be written in the following form

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) + Du(k) \end{aligned} \tag{3.1}$$

where

- $x(\cdot)$ - State vector.
- $y(\cdot)$ - Output vector.
- $u(\cdot)$ - Input vector.
- $A(\cdot)$ - State $n \times n$ matrix.
- $B(\cdot)$ - Input $n \times p$ matrix.
- $C(\cdot)$ - Output $q \times n$ matrix.
- $D(\cdot)$ - Feedthrough $q \times p$ matrix, which is zero in systems without a direct feedthrough.

Linear Autoregressive Model with Exogenous Input

An ARX model is a linear autoregressive model which additionally has exogenous inputs. This means that the model relates the current value of a time series in a way so that it is possible to

- explain or predict past values of the same series
- explain or predict current and past values of the driving (exogenous) series

In addition, the model contains an error term which relates to the fact that knowledge of the other terms will not allow the current value of the time series to be predicted exactly.

For a SISO system, the ARX model structure is

$$\begin{aligned} y(k) + a_1 y(k-1) + ... + a_{na} y(k-n_a) = b_1 u(k-n_k) + \\ + ... + b_{nb} u(k-n_b-n_k+1) + e(k) \end{aligned} \tag{3.2}$$

where

- $y(k)$ - Output at time t.

- n_a - Number of Poles.
- n_b - Number of zeros plus 1.
- n_k - Number of input samples that occur before the input affects the output, also called the dead time in the system.
- $y(k-1), ..., y(k-n_a)$ - Previous outputs on which the current output depends.
- $u(k-n_k), ..., u(k-n_b-n_k+1)$ - Previous and delayed inputs on which the current output depends.
- $e(k)$ - White-noise disturbance value.

The above consideration can be easily extended for MIMO systems.

Non-Linear Autoregressive Model with Exogenous Input

A NARX model is a non-linear autoregressive model which has exogenous inputs.

Such a model can be stated algebraically as

$$y(k) = F(y(k-1), y(k-2), y(k-3), \ldots,$$

$$u(k-n_k), u(k-n_k-1), u(k-n_k-2), u(k-n_k-3), \ldots) + e(k) \tag{3.3}$$

The function F is some nonlinear function, such as a sigmoid network, as used in this work [3]. The sigmoid network function is based on the following expansion

$$F(x) = (x-r)PL + a_1 f((x-r)Qb_1 + c_1) +$$
$$+ \ldots + a_n f((x-r)Qb_n + c_n) + d$$

where f is the sigmoid function, given by the following equation

$$f(z) = \frac{1}{e^{-z}+1}$$

where P and Q are $m \times p$ and $m \times q$ projection matrices. The projection matrices P and Q are determined by principal component analysis of estimation data. If the components of x in the estimation data are linearly dependent, then $p < m$. The number of columns of Q, q, corresponds to the number of components of x used in the sigmoid function. When used in a NARX model, the size of the non-linear regressors property of the object is equal to q . Moreover,

- r - $1 \times m$ vector and represents the mean value of the regressor vector computed from estimation data.
- d, a, and c - Scalars.
- L - $p \times 1$ vector.
- b - $q \times 1$ vectors.

3.3 Identification Methods

In this section the identification methods that were used during the identification process of the models describing the hybrid plant are presented. The ARX models parameters estimation was done with the Linear Least Squares Method , while for the State Space models, Numerical algorithm for Subspace State Space System IDentification (N4SID) was used [3].

Least Squares Method

Assume that $\theta = [a(1) \ \dots \ a(n_a) \ \ b(1) \ \dots \ b(n_b)]$ and $\phi(t) = [-y(t-1) \ \dots \ -y(t-n_a) \ \ u(t-n_k) \ \dots \ u(t-n_k-n_b+1)]$ are the vectors with the parametrs of the ARX model and the input-output pairs respectively. Also suppose that the parameters of θ vector are unknown, but the inputs and outputs over a time interval $1 \leq t \leq N$ are recorded.

$$Z^N = [u(1), y(1) \ \ \dots \ \ u(N), y(N)] \tag{3.4}$$

The least squares method selects θ so as to fit the calculates values $\hat{y}(t|\theta) = \phi^T(t)\theta$ to the measured outputs so as

$$\underset{\theta}{min} V_N(\theta, Z^N) \tag{3.5}$$

where

$$V_N(\theta, Z^N) = \frac{1}{N}\sum_{t=1}^{N}(y(t) - \hat{y}(t|\theta))^2 = \frac{1}{N}\sum_{t=1}^{N}(y(t) - \phi^T(t)\theta)^2 \tag{3.6}$$

The value of θ that minimizes Eq. 3.5 is denoted as

$$\hat{\theta_N} = arg \ \underset{\theta}{min} V_N(\theta, Z^N) \tag{3.7}$$

Since V_N is quadratic in θ, the minimum value can be found by setting the derivative to zero

$$0 = \frac{d}{d\theta}V_N(\theta, Z^N) = \frac{2}{N}\sum_{t=1}^{N}(\phi(t)y(t) - \phi^T(t)\theta)$$

which gives

$$\sum_{t=1}^{N}\phi(t)y(t) = \sum_{t=1}^{N}\phi(t)\phi^T(t)\theta \tag{3.8}$$

or

$$\hat{\theta_N} = \left[\sum_{t=1}^{N}\phi(t)\phi^T(t)\right]^{-1}\sum_{t=1}^{N}\phi(t)y(t) \tag{3.9}$$

N4SID Algorithm

N4SID is an algorithm for solving the subspace identification problem [5]. Details from that source are provided below. Linear subspace identification methods are concerned with systems and models of the form

$$x_{k+1} = Ax_k + Bu_k + w_k$$
$$y_k = Cx_k + Du_k + v_k \tag{3.10}$$

with

$$\mathbf{E}\left[\begin{pmatrix} w_p \\ v_p \end{pmatrix}\begin{pmatrix} w_q^T & v_q^T \end{pmatrix}\right] = \begin{pmatrix} Q & S \\ S^T & R \end{pmatrix}\delta_{pq} \geq 0. \tag{3.11}$$

The vectors $u_k \in \mathbb{R}^{m\times 1}$ and $y_k \in \mathbb{R}^{l\times 1}$ are the measurements at time instant k of the m inputs and l outputs of the process, whereas $\mathbf{E}$ shows the expectation. The vector x_k is the state of the process at discrete instant k, $v_k \in \mathbb{R}^{l\times 1}$ and $w_k \in \mathbb{R}^{n\times 1}$ are unobserved vector signals, v_k is called measurement noise and w_k process noise accordingly. It is assumed that they are zero mean, stationary white noise vector sequences and uncorrelated with the inputs u_k. $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{l\times n}$, $D \in \mathbb{R}^{l\times m}$ are the matrices as described above, while $Q \in \mathbb{R}^{n\times n}$, $S \in \mathbb{R}^{n\times l}$ and $R \in \mathbb{R}^{l\times l}$ are the covariance matrices of the noise vectors w_k and v_k.

In subspace identification it is typically assumed that the number of available data points goes to infinity and that the data is ergodic[1]. It is assumed a large number of measurements of the input u_k and the output y_k generated by the unknown system the task is to determine the order n of the unknown system, the system matrices A, B, C, D up to within a similarity tranformation and an estimate of the matrices Q, S, R.

Subspace identification algorithms always consist of two steps. The first step makes a projection of certain subspaces generated from the data, to find an estimate of the extended observability matrix and/or an estimate of the states of the unknown system. The second step then retrieves the system matrices from either this extended observability matrix or the estimated states.

The following input-output matrix equation, played a very important role in the development of subspace identification, as described also in Appendix A, Eq. (A.1):

$$Y_f = \Gamma_i X_i + H_i^d M_f + N_f \tag{3.12}$$

Also it is denoted that

$$U_p \stackrel{def}{=} U_{0|i-1}, \quad U_f \stackrel{def}{=} U_{i|2i-1}, \quad Y_p \stackrel{def}{=} Y_{0|i-1}, \quad Y_f \stackrel{def}{=} Y_{i|2i-1}$$

where the subscript p and f denote the past and the future respectively. The matrix containing the inputs U_p and the outputs Y_p is noted

$$W_p \stackrel{def}{=} \begin{pmatrix} Y_p \\ U_p \end{pmatrix}$$

The block Hankel matrix (Eq. (A.5) and (A.6)) formed with the process noise w_k and the measurement noise v_k are defined, respectively, as $M_{0|i-1}$ and $N_{0|i-1}$ in the same way. Once again, short hand notation is defines

$$M_p \stackrel{def}{=} M_{0|i-1}, \quad M_f \stackrel{def}{=} M_{i|2i-1}, \quad N_p \stackrel{def}{=} N_{0|i-1}, \quad N_f \stackrel{def}{=} N_{i|2i-1}$$

Finally the state sequence X_i is denoted as

$$X_i \stackrel{def}{=} \begin{pmatrix} x_i & x_{i+1} & x_{i+2} & \dots & x_{i+j-1} \end{pmatrix} \tag{3.13}$$

[1] a stochastic process is said to be ergodic if its statistical properties can be deduced from a single, sufficiently long, random sample of the process. The reasoning is that any collection of random samples from a process must represent the average statistical properties of the entire process.

In the following, the matrices $\mathcal{A} \in \mathbb{R}^{p \times j}$ and $\mathcal{B} \in \mathbb{R}^{q \times j}$ will be used.
The orthogonal projection of the row space of $\mathcal{A}$ into the row space ob $\mathcal{B}$ is denoted by $\mathcal{A}/\mathcal{B}$ and defined as $\mathcal{A}\mathcal{B}^{\dagger}\mathcal{B}$, where † denotes the Moore-Penrose pseudo-inverse.

$\mathcal{A}/\mathcal{B}^{\perp}$ is the projection of the row space of $\mathcal{A}$ into $\mathcal{B}^{\perp}$, the orthogonal complement of the row space of $\mathcal{B}$, for which $\mathcal{A}/\mathcal{B}^{\perp} = \mathcal{A} - \mathcal{A}/\mathcal{B}$.

The first step of the subspace algorithms performs weighted projection of the row space of the previously defined data Hankel matrices. From this projection, the observability matrix Γ_i and/or an estimate X_i of the sequence X_i can be retrived. In the second step, the system matrices $A,\ B,\ C,\ D$ and $Q,\ S,\ R$ are determined.

The subspace methods start from the previously presented matrix input-output Eq. (3.12). It states that the block Hankel matrix containing the future outputs Y_f is related in a linear way to the future input block Hankel matrix U_f and the future state sequence X_i. The basic idea of subspace identification now is to recover the $\Gamma_i X_i$-term of this equation. This is a particularly interesting term since either the knowledge of Γ_i or X_i leads to the system parameters. Moreover $\Gamma_i X_i$ is a rank deficient term (of rank n, i.e. the system order) which means that once $\Gamma_i X_i$ is known, Γ_i, X_i and the order n can be simply found from a Singular Value Decomposition (SVD). SVD procedure is described in Appendix A

3.4 Design Procedure of the Identification Datasets

An internal combustion engine can be considered as a MIMO system. The main control object in this Thesis is the λ value, so λ is the most important system output. At the same time, emissions and fuel consumption are also significant system outputs, but their behavior is affected by λ variation.

For the design of the identification experiment, the parameters that individually affect the λ value and that are independent from each other have been considered, in order to find the exact relation between each parameter and the λ value during the modeling procedure. Eventually three experiments were designed and performed, as illustrated in Fig. 3.2: each time changing the torque load ($Tload$), the engine speed (SE) or the electric motor's frequency inverter command ($FrInvCmd$). For every parameter a Pseudo-Random Binary Sequence (PRBS) signal of proper bias and amplitude, according to the expected loading conditions of the hybrid power plant during open-loop experiments. Data from each experiment was splitted for the identification and validation process, as shown in Fig. 3.2.

The Identification data were merged in order to find the model with the best average fit on the three data series shown in Fig. 3.2.

Multiple kinds of models were identified, as described in Fig.3.3 according to their Input-Output attributes [4].

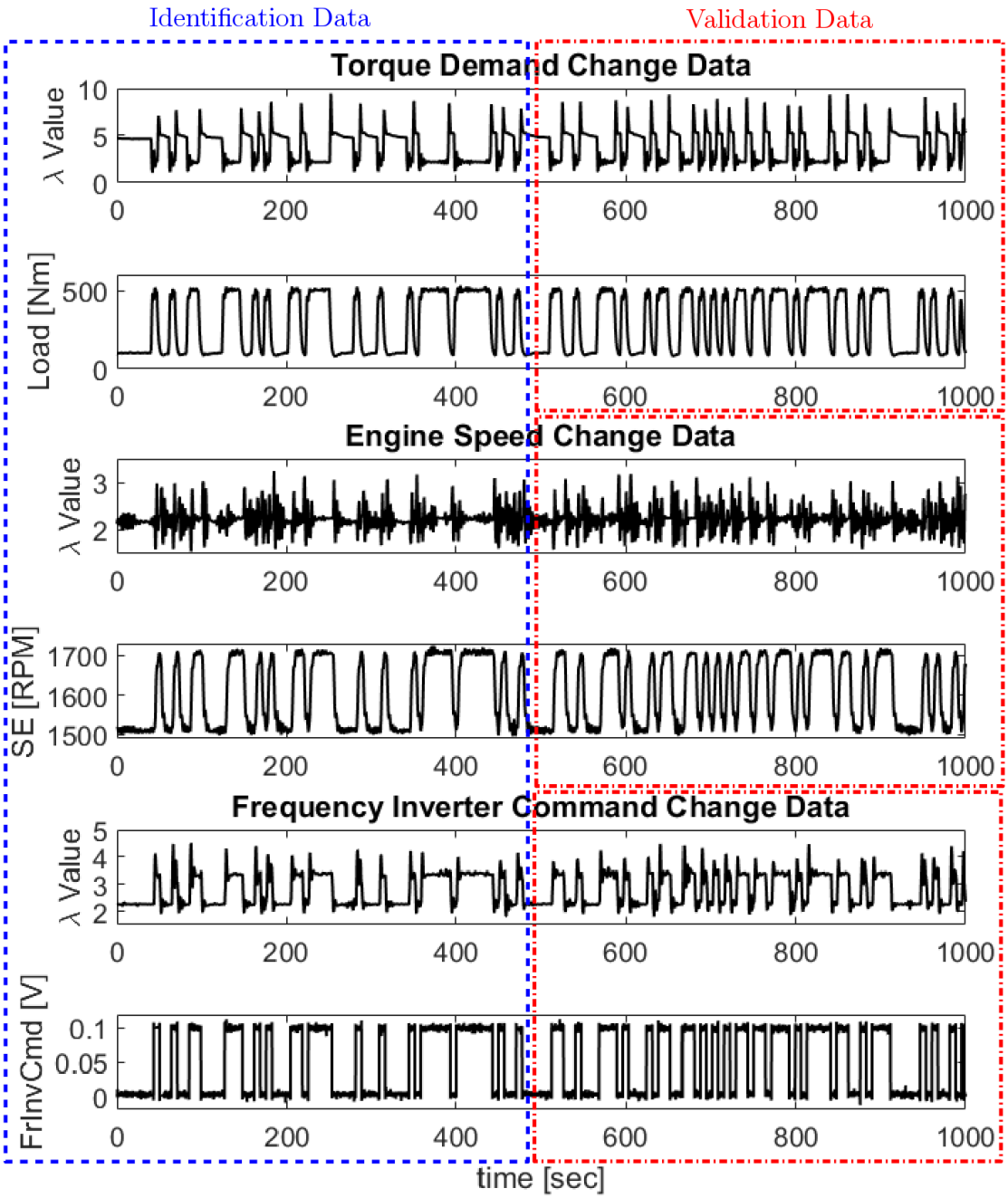

Figure 3.2: Identification and Design Process

CONTROL ORIENTED MODELS											
NAME	TYPE	INPUT					OUTPUT				
		FrInvCmd	MAP	SPEED	dSPEED	λ	λ	MAP	NOx	Opacity	FOC
m1XX	LINEAR MODEL	x					x				
m2XX	LINEAR MODEL	x	x	x			x				
m3XX	LINEAR MODEL	x			x		x				
mMAP	LINEAR MODEL	x		x				x			
mNOx	LINEAR MODEL		x			x			x		
mOPA	ARX MODEL		x			x				x	
mCons	LINEAR MODEL		x	x							x
m4XX	LIN / ARX MODEL	x	x	x	x	x	x		x	x	x
SIMULATION ORIENTED MODELS											
NAME	TYPE	INPUT					OUTPUT				
		FrInvCmd	Tdiesel	SPEED			λ	MAP	Telectric		
mTelu	LINEAR MODEL	x							x		
m0XX	NARX MODEL		x	x			x	x			

Figure 3.3: Groups of the identified models

3.5 Control-Oriented Modeling

In this section the procedure of extracting and validating the models, as these will be used for the Controller Design procedure is described. Taking into account that a the controller needs a linear internal model and considering the availability and the relative timing advance of each measurement needed, Model Identification Toolbox [2] was deployed.

For the linear model identification, method N4SID was prefered over Prediction Error Method (PEM) [2] in order to extract properties from the data linear models for multi-variable systems, as it gives models with better fit to the validation data. Emphasis was given on low-order models which, in combination with input signals, give a good description of the oscillating dynamics of the engine.

3.5.1 Linear λ Output Model

SISO linear model

This SISO model Group 1 (m1XX) with input the $FrInvCmd$ and output the λ change, as described by Eq. (3.14), has been identified using PEM algorithm by LME [13] as the best fit to $FrInvCmd$ change data. SISO models are mainly being used at the design of controllers which cannot manage multiple-variable systems. Model m101, whose fitting characteristics are presented in Fig. 3.4, is a high-order model, which was used for robust controller design in previous work of LME. In this thesis, it is compared to the MISO and MIMO models performance in Controller Design procedure.

$$\lambda = f(FrInvCmd) \tag{3.14}$$

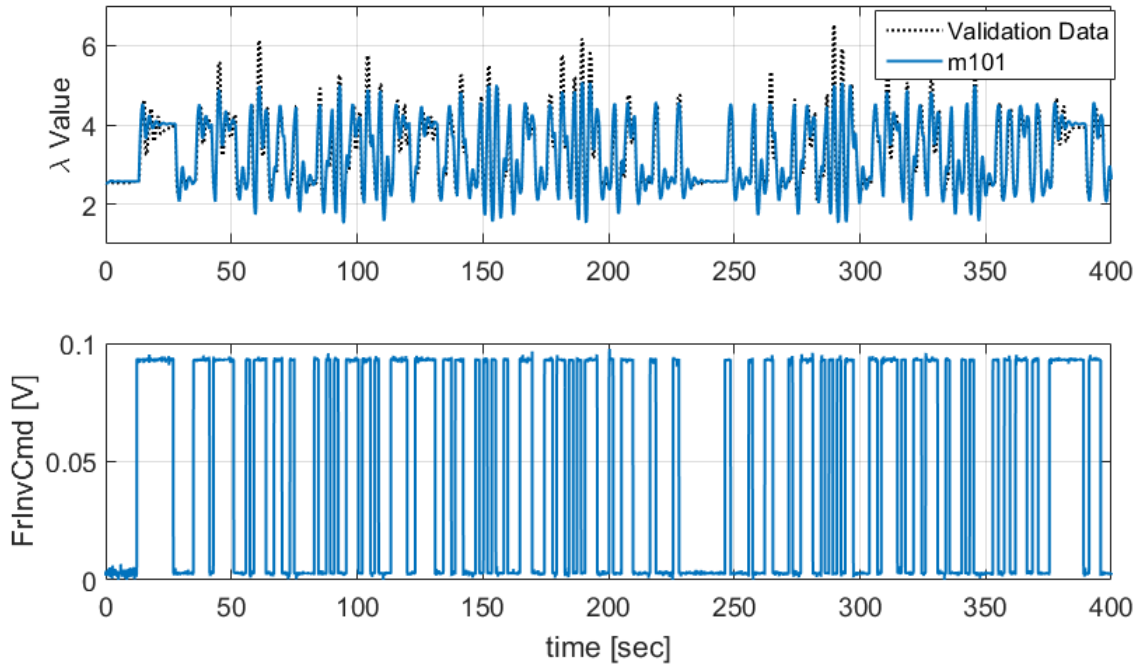

Figure 3.4: Fit of model m101 to evaluation data

MISO linear model with SE and MAP as disturbance

Models of Group 2 (m2XX) are MISO models with λ output being a result of $FrInvCmd$, MAP and SE change, as described be Eq. (3.15). MAP is used as disturbance input, as an alternative to the $Tload$ signal, because in a production engine the torque applied is not a common measurement in contrast to the boost pressure which is standard. With the use of MISO model, a lower order model can be achieved, because the system dynamics can be decomposed to a linear combination of the model input signals.

$$\lambda = f(FrInvCmd, MAP, SE) \tag{3.15}$$

The fitting results of group 2 models ($m2XX$), are presented in Fig. 3.5, 3.6, 3.7. It can be concluded that except from the very close description they give for $FrInvCmd$ change (Fig. 3.7), they can also describe the oscillating dynamics of λ value during step loading (Fig. 3.5) and during engine speed change (Fig. 3.6).

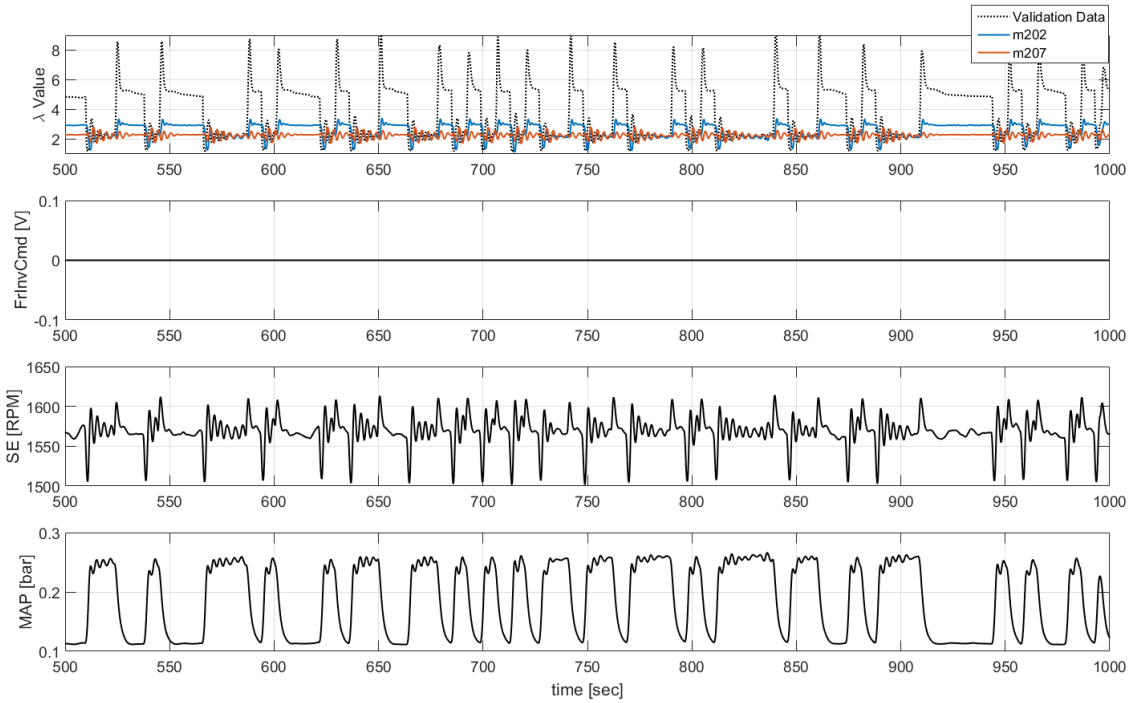

Figure 3.5: Fit of models m202 and m207 to Load Torque change evaluation data

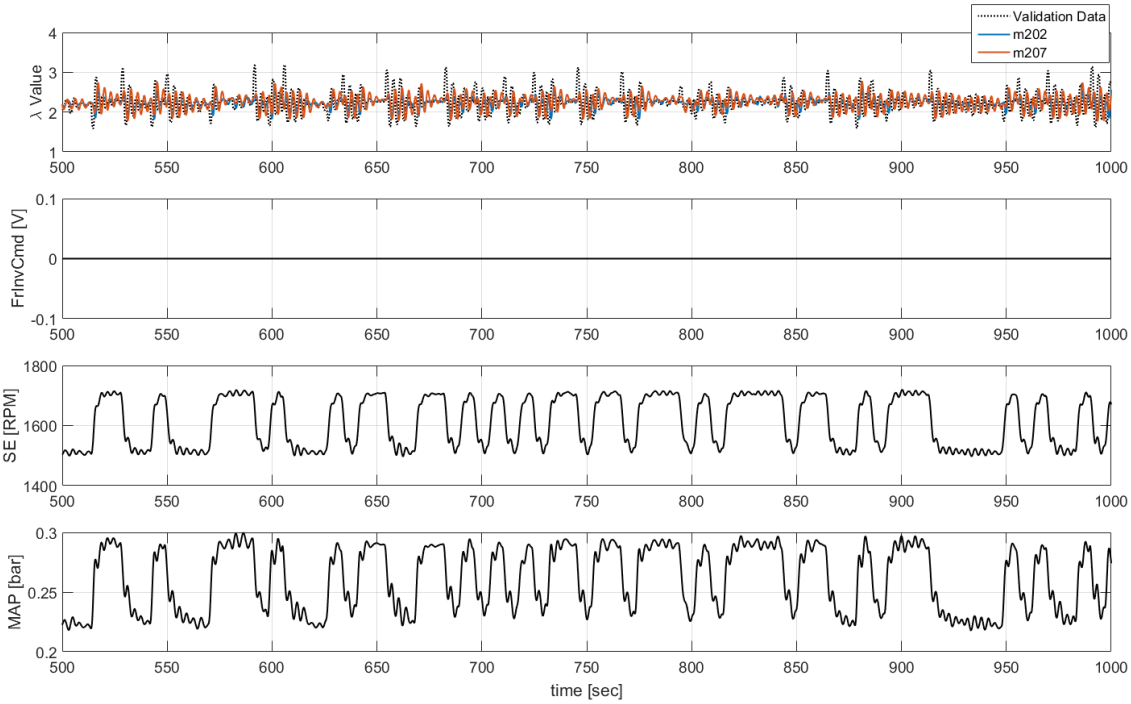

Figure 3.6: Fit of models m202 and m207 to Engine Speed change evaluation data

MISO linear model with ΔSpeed disturbance

The fuel injection of the ICE in the hybrid powertrain is independently controlled by the ECU. The ECU uses the speed deviation from the speed reference, as described by Eq. (3.16) in order to inject more or less fuel in the cylinders in order to maintain the engine speed, accelerate or slow down, according to the speed reference (SE_{Ref}). So the dSE

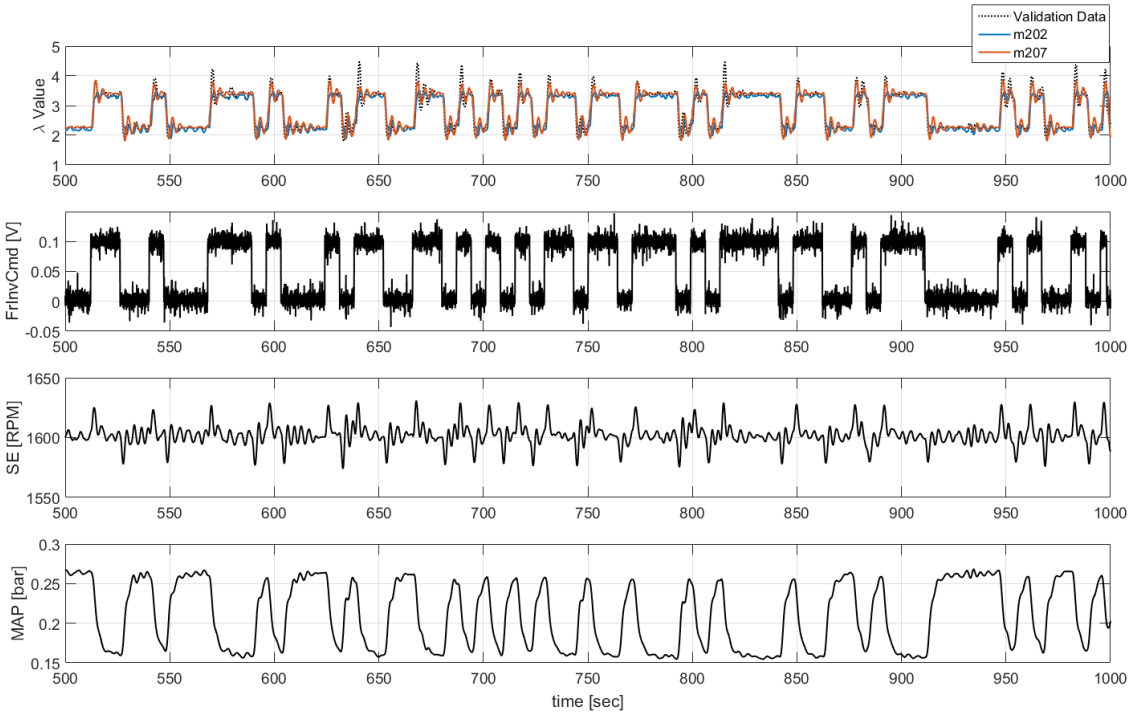

Figure 3.7: Fit of models m202 and m207 Frequency Inverter Command change evaluation data

measurement could be a precursor of the λ value change, which could help a model based controller predict a change in the output trajectory[2] and act preventively.

$$dSE = SE - SE_{Ref} \tag{3.16}$$

Another advantage of the dSE measurement compared to MAP is that speed changes directly as the load is imposed, in contrast with the boost pressure which changes almost the same time with λ.

The models of group 3 (m3XX) have the structure described by Eq. (3.17). Due to the non-linear relation between the dSE and λ value, model m301 was identified using SE and $FrInvCmd$ change identification data series, while model m303 was fitted to $Tload$ and $FrInvCmd$ change identification data. The fitting results of models m301 and m303 are presented in Fig: 3.8, 3.9, 3.10.

As shown in Fig. 3.8, 3.9, model m301 underestimates the λ value oscillation during torque load change while model m303 overestimates the λ value oscillatory behavior during changes in speed reference. The performance of these two models is to be further evaluated during experimental application of the controllers which utilize these models.

$$\lambda = f(FrInvCmd, dSE) \tag{3.17}$$

[2]trajectory here denotes the sequence of a signal, as this evolves in time.

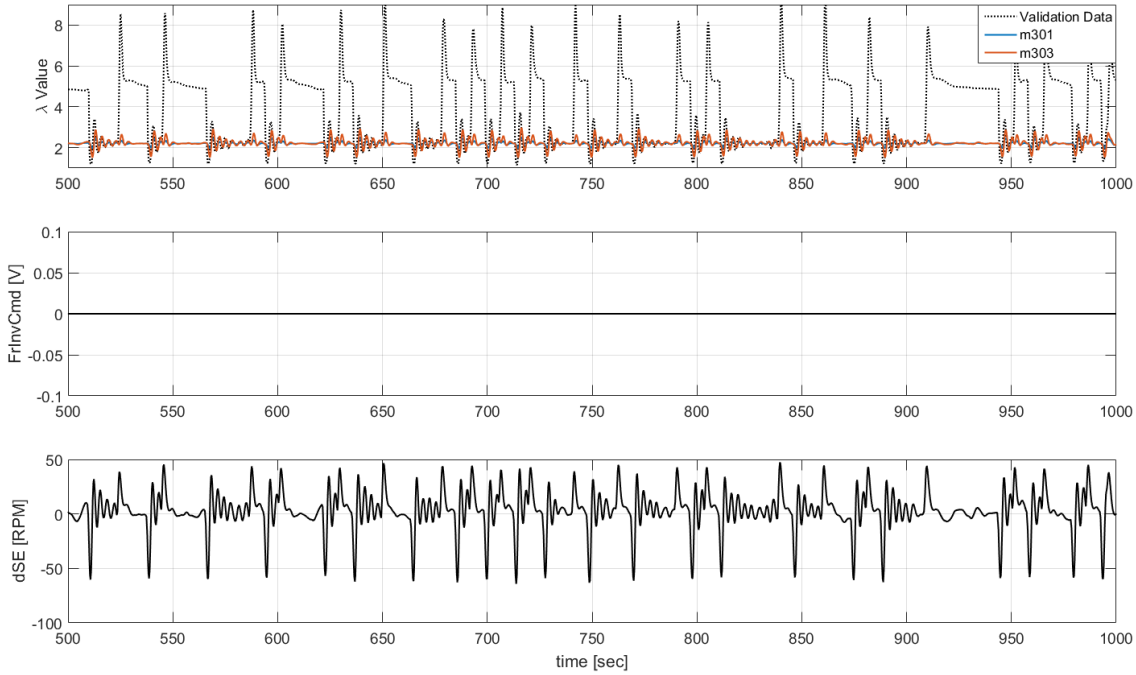

Figure 3.8: Fit of models m202 and m207 to Load Torque change evaluation data

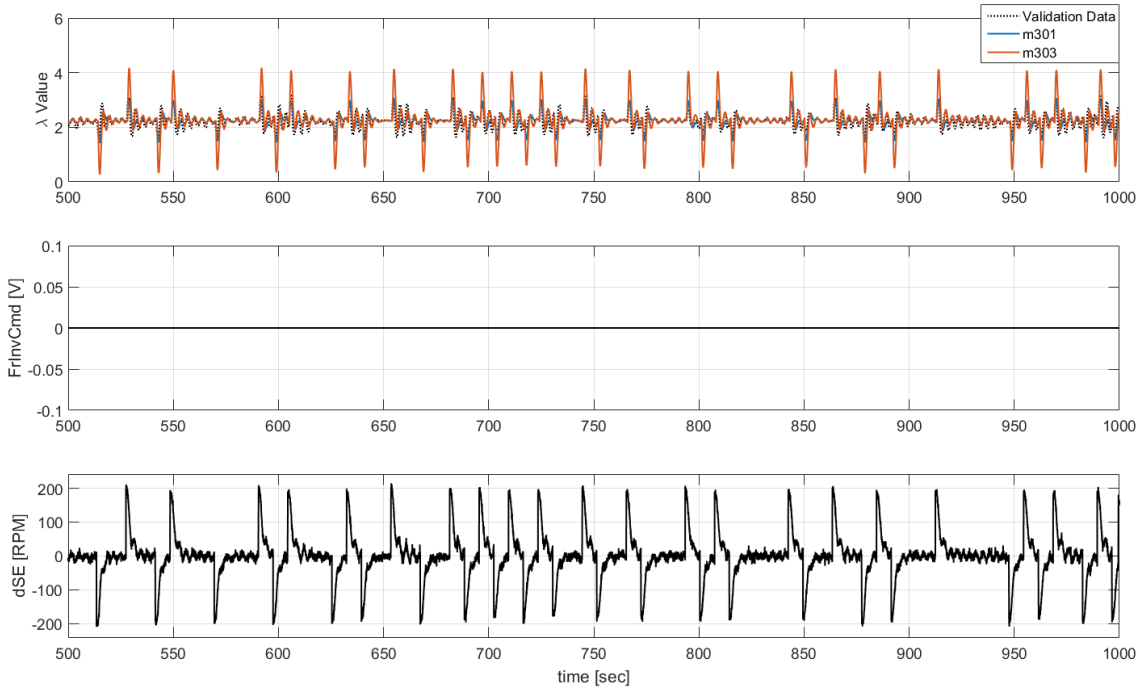

Figure 3.9: Fit of models m202 and m207 to Engine Speed change evaluation data

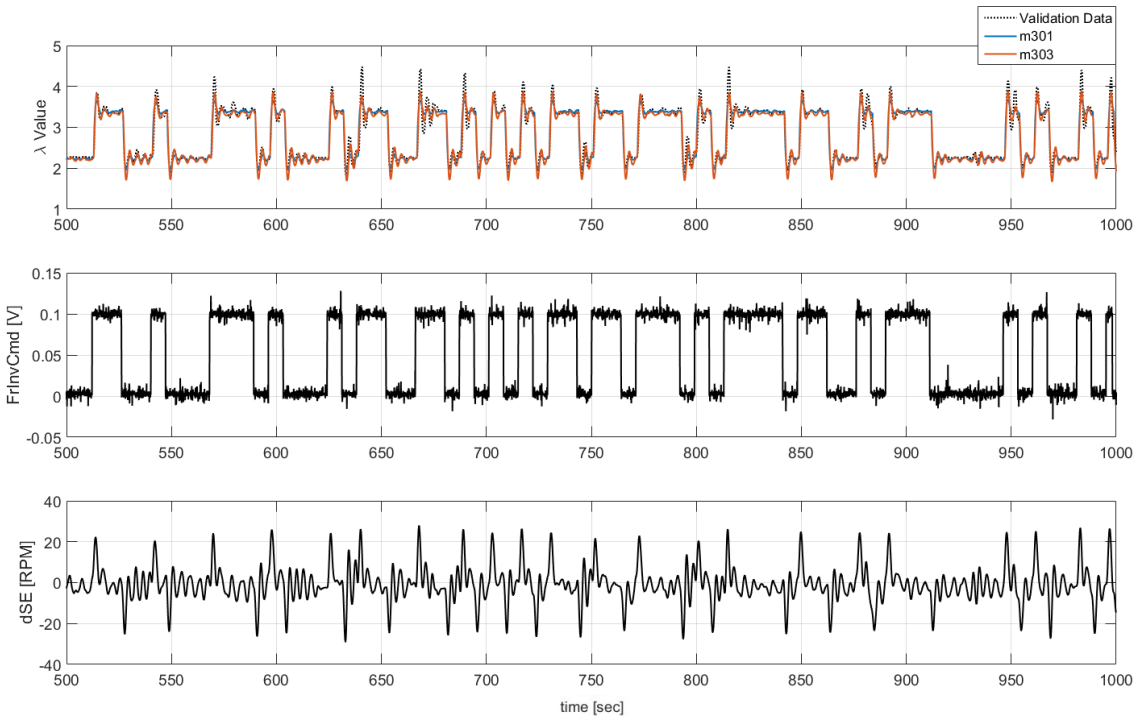

Figure 3.10: Fit of models m202 and m207 to Frequency Inverter Command change evaluation data

3.5.2 Linear MAP Output Model

Except from the λ value, another meaningful ICE output is the intake pressure of the ICE: it is essential to know what change of MAP will bring a deviation of engine speed or $FrInvCmd$. Intake pressure is used by models of group 2 (m2XX) and also by ICE emissions and fuel oil consumption models.

MAP linear model, as described by Eq. (3.18), was fitted to SE and $FrInvCmd$ change data. Its fitting is shown in Fig. 3.11 and 3.12.

$$MAP = f(FrInvCmd, SE) \tag{3.18}$$

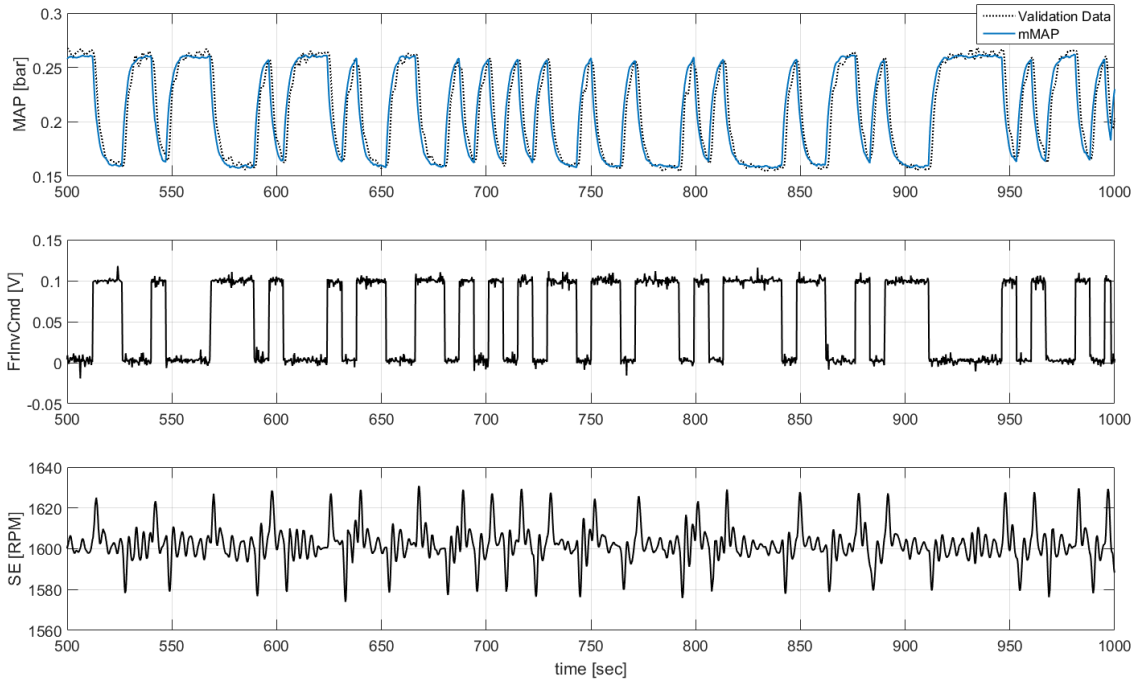

Figure 3.11: Fit of MAP model to Frequency Inverter Command change evaluation data

3.5.3 Diesel Engine Emission and Fuel Consumption Models

Engine emissions and consumption models were modeled at the operating regime $SE = 1600\ [RPM]$. Low-order NOx (NOx), Opacity (OPA) and Fuel Consumption ($F.O.C$) models depending on the fundamental engine measurements (SE, MAP, λ) were extracted from the available measured data.

Linear NOx Emissions Model

The lean-burning nature of diesel engines and the high temperatures and pressures of the combustion process result in significant production of gaseous nitrogen oxides (NOx). NOx formation is mainly depended on the cylinder's combustion maximum pressure and temperature, two measurements that are not available on HIPPO 1 testbed. As an alternative, ICE's NOx emissions have been modeled using λ and MAP values, as described by Eq. (3.19).

$$NOx = f(\lambda, MAP) \tag{3.19}$$

The fitting results of NOx model is presented in Fig. 3.13, 3.14 and 3.15. As it can be seen, model $mNOx$ is able to provide acceptable NOx emissions behavior.

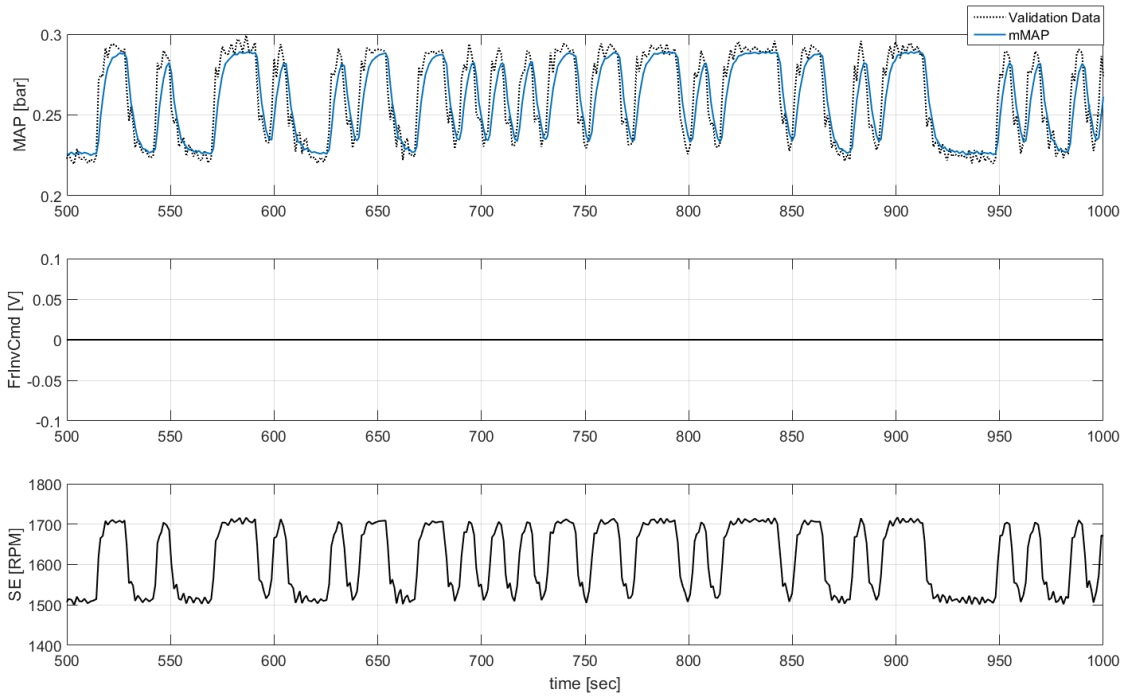

Figure 3.12: Fit of MAP model to Engine Speed change evaluation data

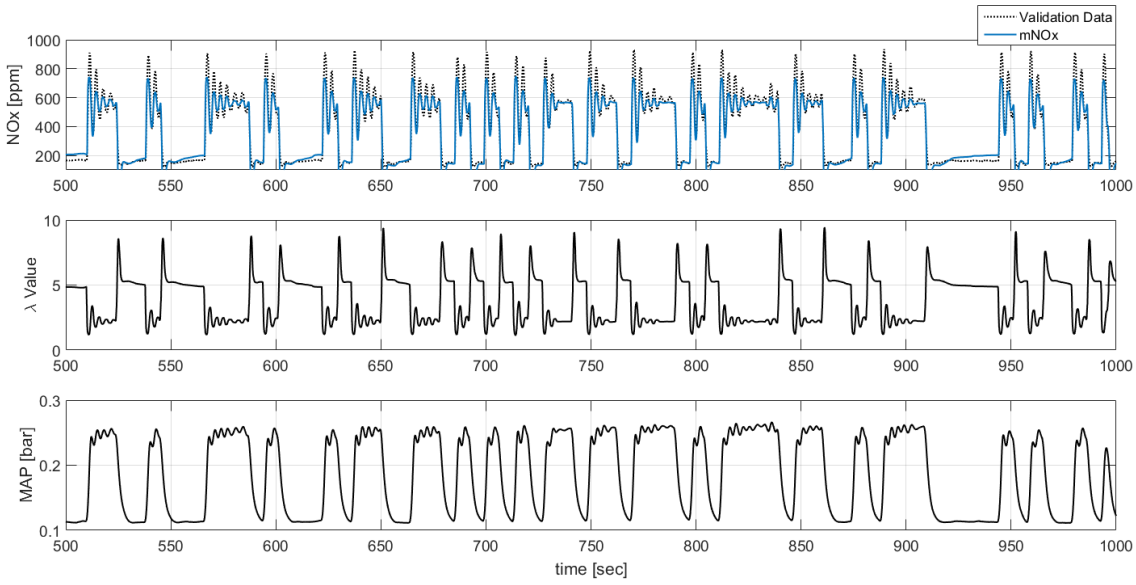

Figure 3.13: Fit of NOx model to Load Torque change evaluation data

ARX Model of Exhaust Opacity

The fine particulate matter (PM) in diesel exhaust (e.g., soot), which affect the exhaust gases' opacity, has traditionally been of greater concern. Diesel engines produce significant amounts of particulate contaminants when running without enough oxygen, as required by combustion. The amount of PM that is formed in the cylinder under the same conditions is not standard. It depends on the amount of the oxygen in the cylinder, on the loading profile of the engine and the in-cylinder temperature, where in high temperatures a amount of PM is oxidized decreasing the emitted PM amount [7].

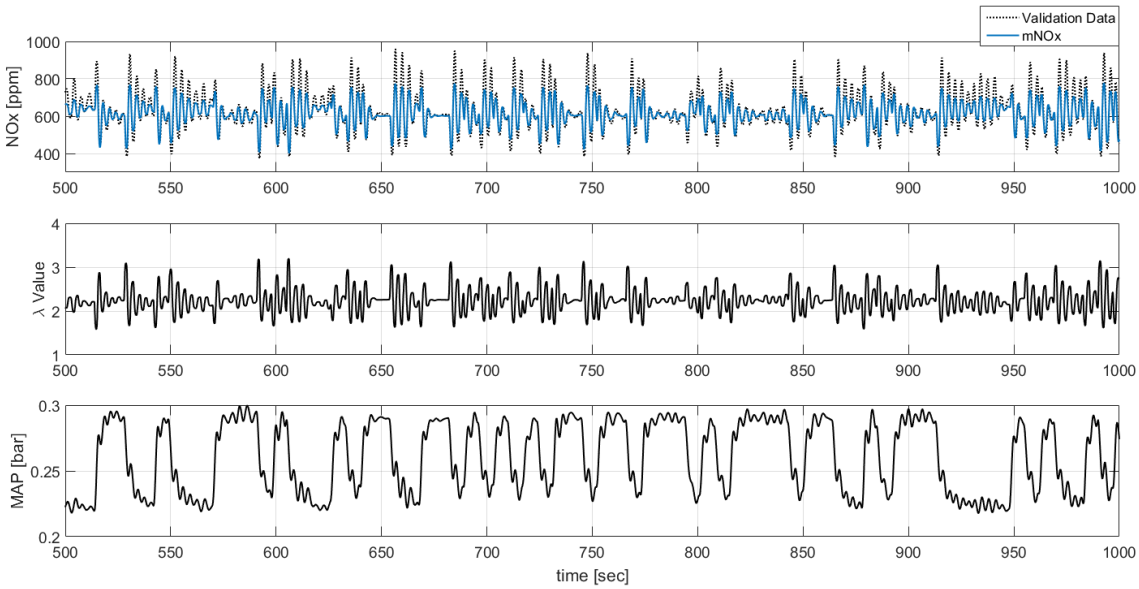

Figure 3.14: Fit of NOx model to Engine Speed change evaluation data

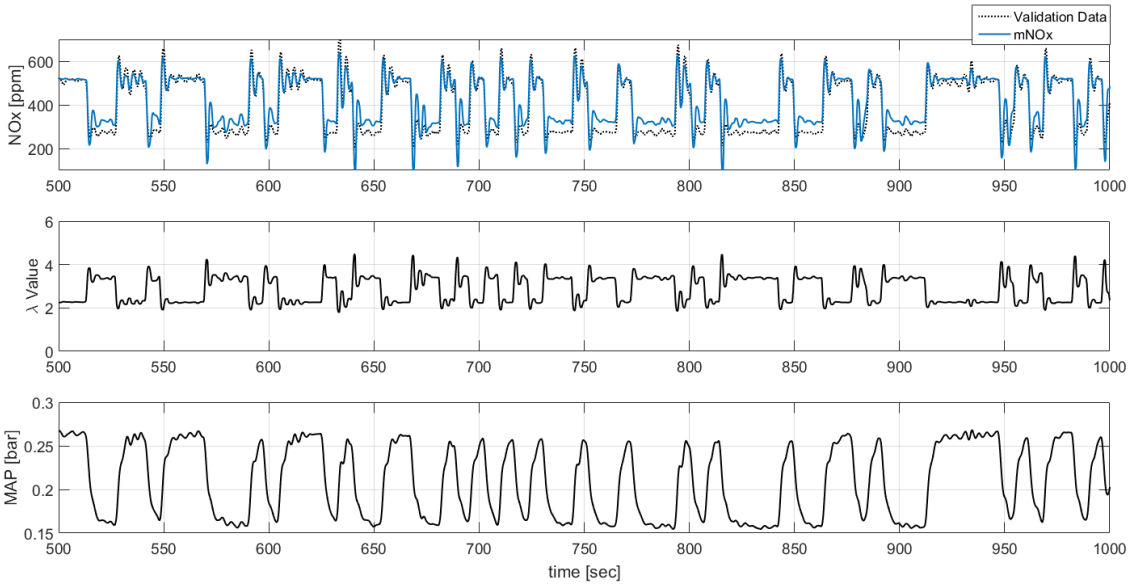

Figure 3.15: Fit of NOx model Frequency Inverter Command change evaluation data

As the engine environment is strongly non-linear, in certain occasions, ARX models describe better the dynamic behavior of the diesel engine compared to the classic linear models [1]. For this reason ARX models may be preferred over the linear state space models.

Exhaust gas opacity (OPA) was modeled as a MISO ARX model, as describen by Eq. (3.20), using $Tload$ and $FrInvCmd$ change identification data series. The fitting results of OPA model (mOPA) are demonstrated in Fig. 3.16 and 3.17.

In general, the fit of the opacity model is acceptable. Both the initial "spike" and the steady state parts are captured.

$$OPA = f(\lambda, MAP) \tag{3.20}$$

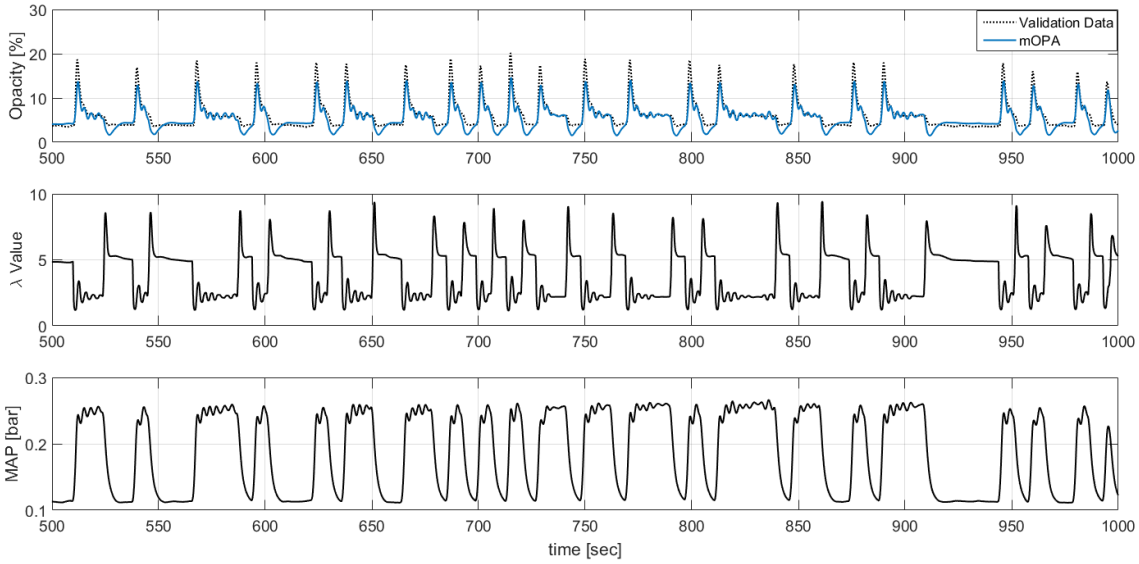

Figure 3.16: Fit of Opacity model to Load Torque change evaluation data

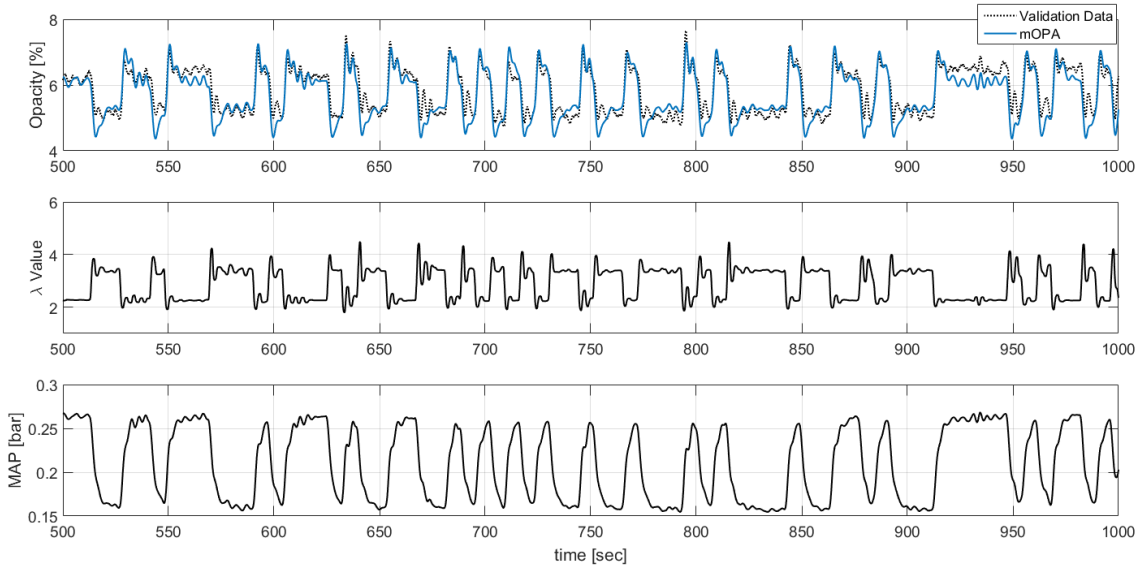

Figure 3.17: Fit of Opacity model Frequency Inverter Command change evaluation data

Linear Fuel Oil Consumption Model

Finally, the fuel oil consumption (FOC) was modeled according to engine speed and boost pressure values, as shown in Eq. (3.21). FOC model (mCons) has an accurate fit to evaluation data, as shown in Fig. 3.18, 3.19 and 3.20.

$$F.O.C. = f(MAP, SE) \tag{3.21}$$

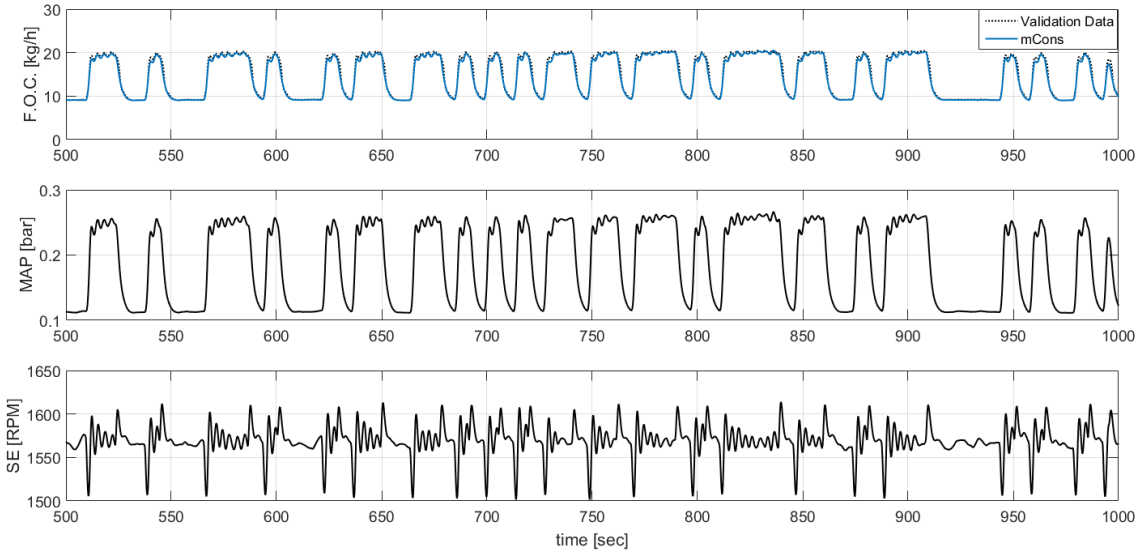

Figure 3.18: Fit of FOC model to Load Torque change evaluation data

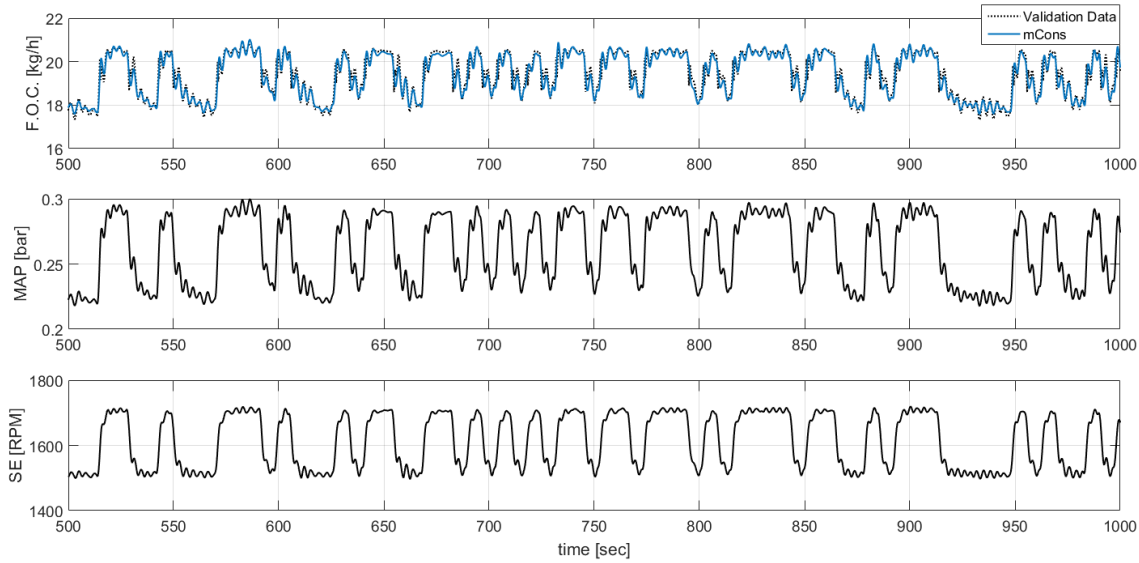

Figure 3.19: Fit of FOC model to Engine Speed change evaluation data

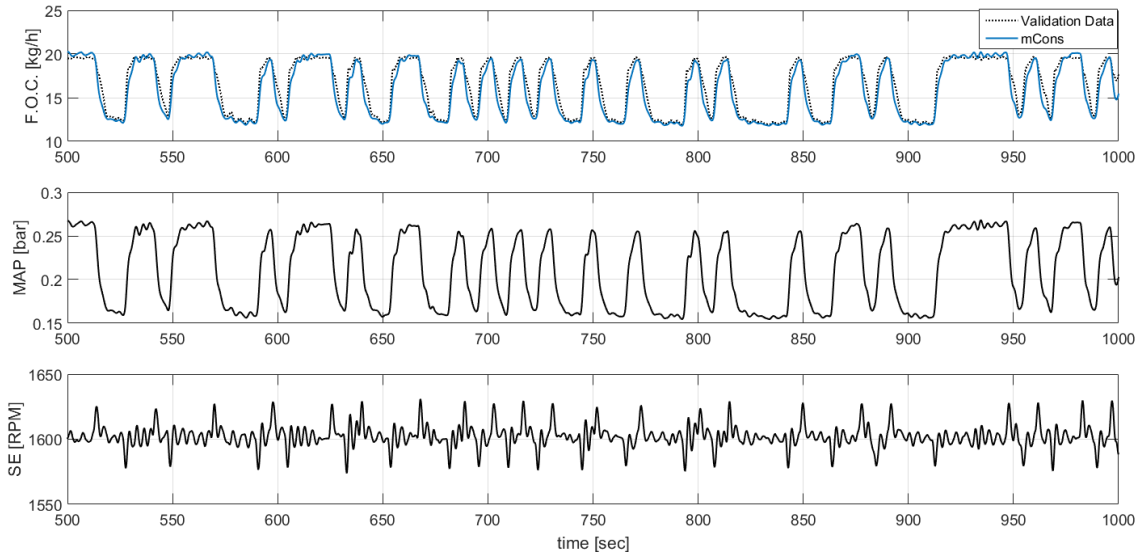

Figure 3.20: Fit of NOx model Frequency Inverter Command change evaluation data

3.5.4 Linear MIMO Engine Model

MPC can handle multi-variable systems. With the adoption of an appropriate process model, engine operating scenario may include, beside optimal λ control, compliance with environmental restrictions such as lower NOx emissions and limited hydrocarbons (HC) imprint through reduced fuel oil consumption.

In order to combine MPC's advanced features with operational needs of the hybrid plant, a MIMO model was built, as descibed by Eq. (3.22), by combining models (3.17), (3.19), (3.20), and (3.21).

$$\begin{bmatrix} \lambda \\ NOx \\ OPA \\ F.O.C. \end{bmatrix} = f(FrInvCmd, dSE, SE, MAP, \lambda) \tag{3.22}$$

The MIMO model (named m4XX) has the same fitting characteristics as each MISO model separately. A snap shoot of the four outputs of the MIMO model, based on evaluation data is being shown in Fig. 3.21.

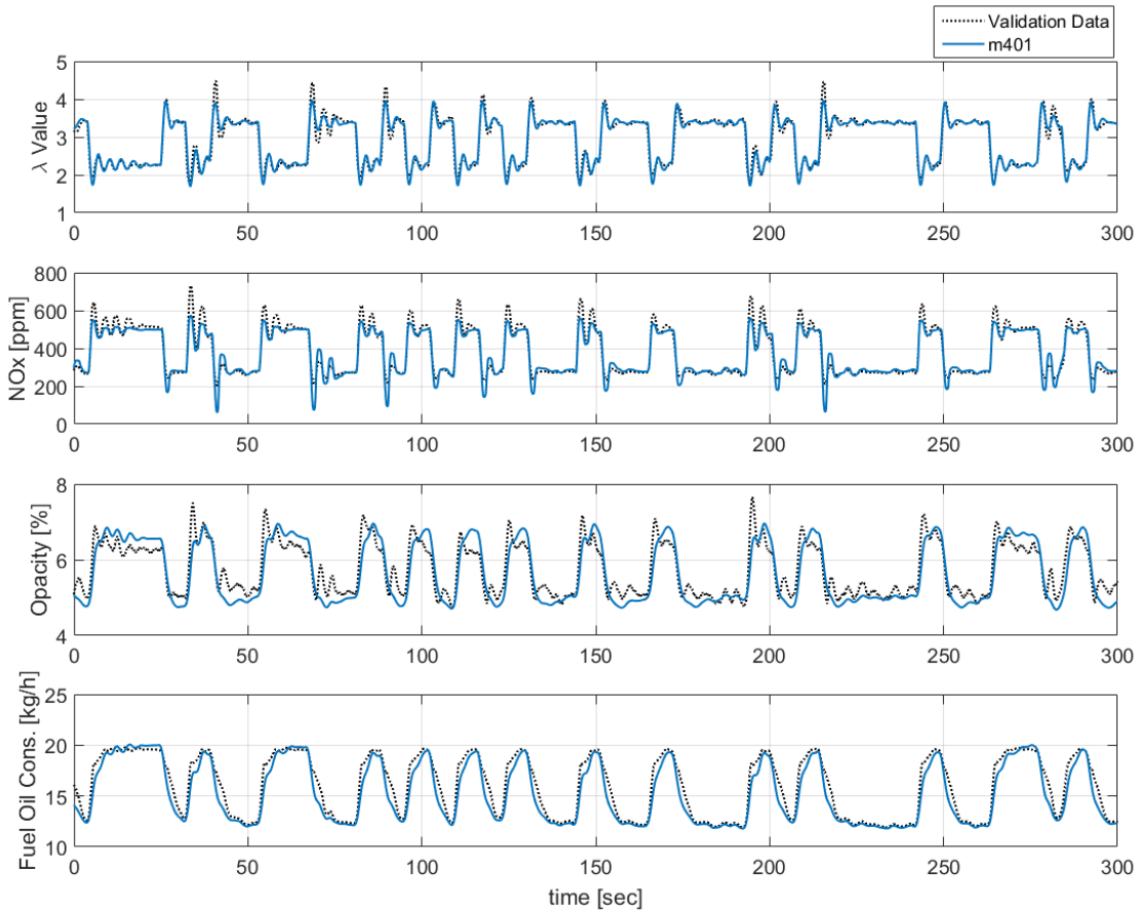

Figure 3.21: Fitting of MIMO model Frequency Inverter Command change evaluation data

3.6 Simulation-Oriented Modeling

For models that are going to be used in simulation describing the engine behavior, non-linear model identification was performed in order to cover a wider operating range of the engine. Initially high-order ARX models were extracted from identification data with Diesel Engine Torque ($Tdiesel$) and SE as inputs and λ and MAP as outputs, as shown in Eq. (3.23). The ARX model with the best fit to the validation data was modified to a non-linear ARX (NARX) model, using *sigmoidnet* as non-linear estimator and 10 terms in the sigmoid expansion as non-linear options [2].

$$\begin{bmatrix} \lambda \\ MAP \end{bmatrix} = f_{nl}(Tdiesel, SE) \tag{3.23}$$

The fitting of the nl engine model to the evaluation data is presented in Fig. 3.22, 3.23 and 3.24, while the model's fitting to open loop experimental data is shown in Fig. 3.25 and 3.26. Results confirm that the above NARX model can be used for a wider operational area of the ICE.

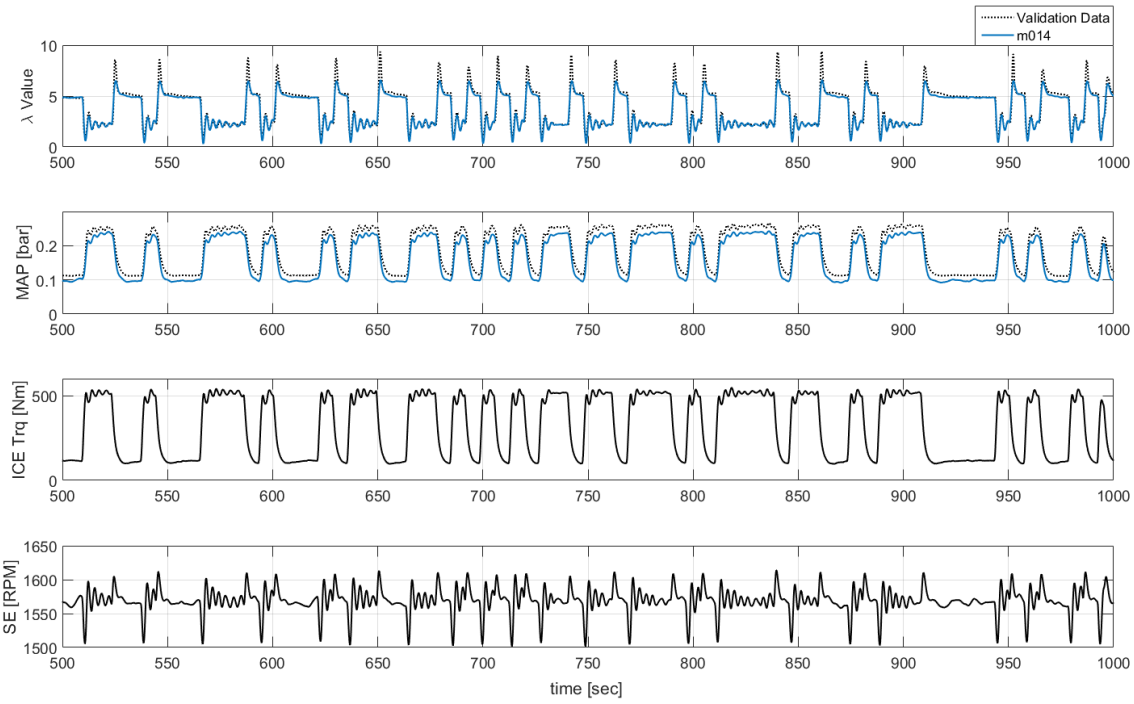

Figure 3.22: Fit of NARX engine model to Load Torque change evaluation data

Also a 1st order linear system (Eq. (3.24)) was identified in order to simulate the Torque output of the Electric Motor (Tel) according to the $FrInvCmd$ input. Electric torque model according to $FrInvCmd$ change fitting to evaluation data is presented in Fig. 3.27.

$$Tel = f(FrInvCmd) \tag{3.24}$$

The Total Torque Demand ($Tload$) as imposed by the water brake dynamometer is described by the following equation, which also determines the total power split between the two components of the powertrain, namely the thermal engine and the electric motor

$$Tload = Tdiesel + Tel \tag{3.25}$$

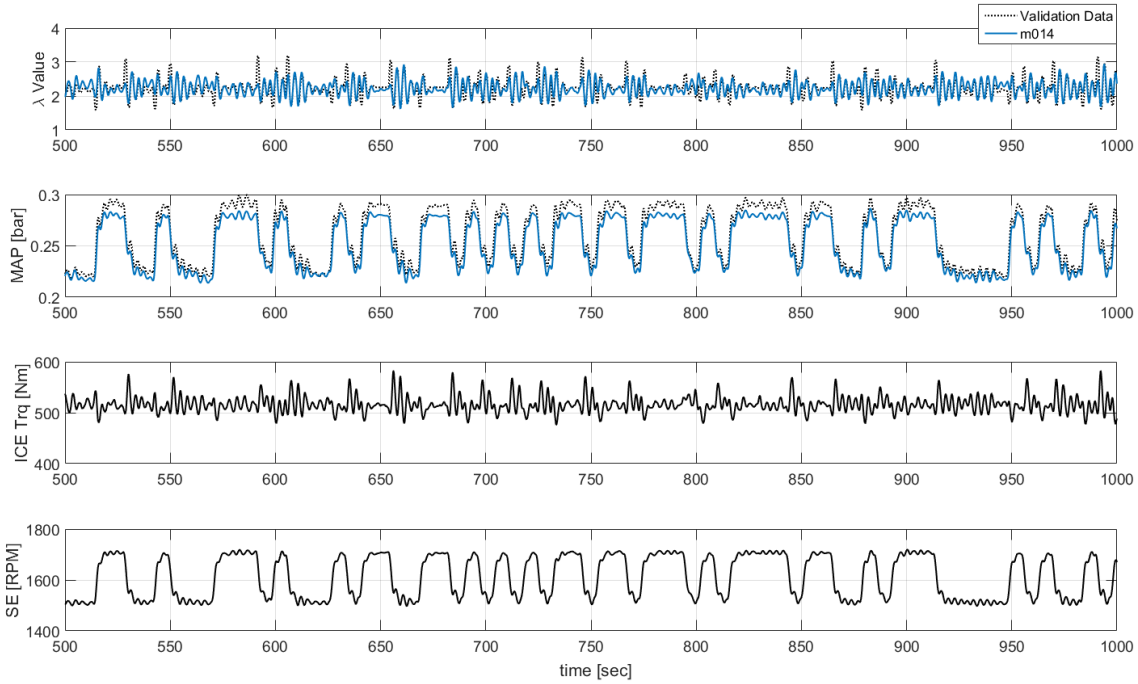

Figure 3.23: Fit of NARX engine model to Engine Speed change evaluation data

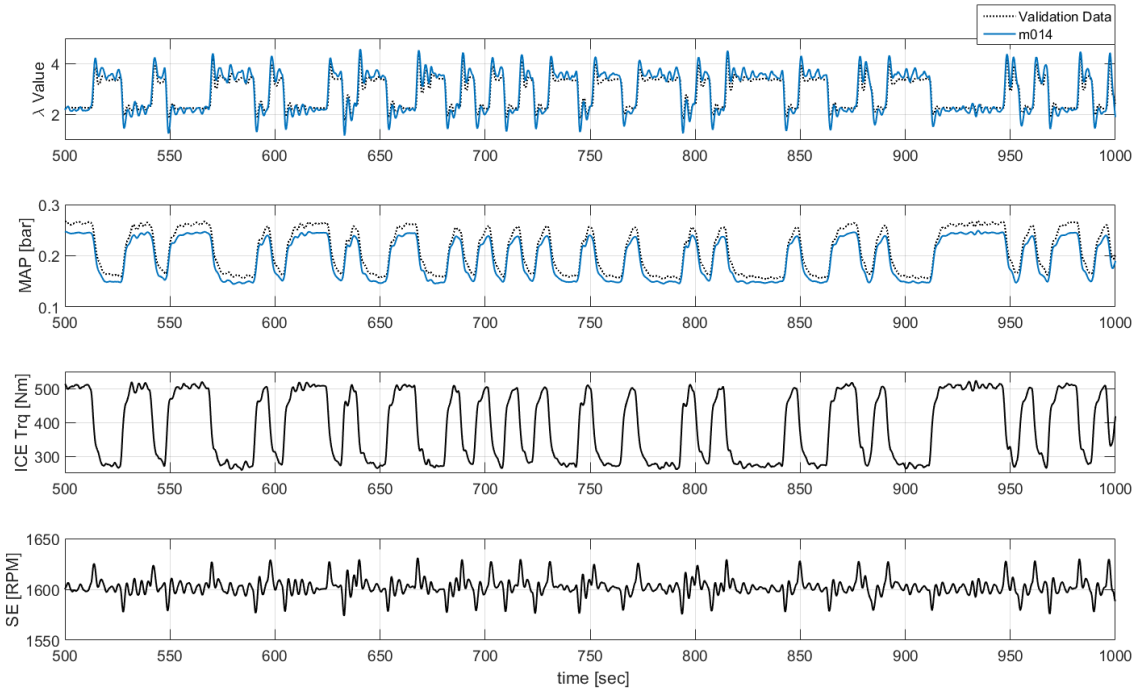

Figure 3.24: Fit of NARX engine model Frequency Inverter Command change evaluation data

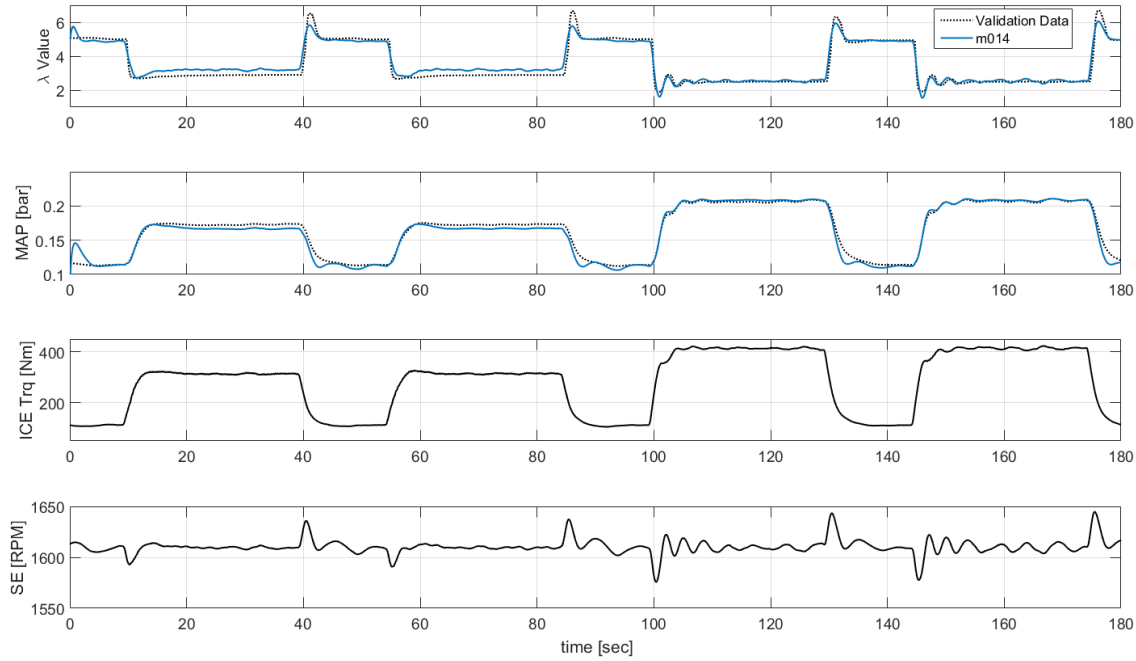

Figure 3.25: Fit of NARX engine model to open-loop experimental data

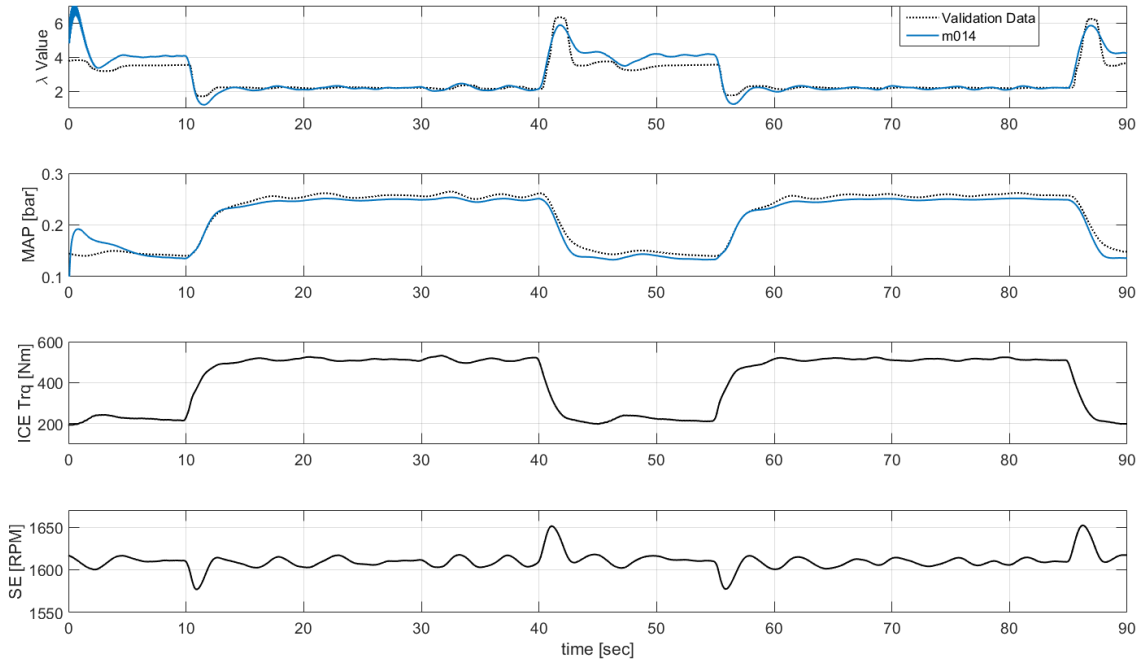

Figure 3.26: Fit of NARX engine model to open-loop experimental data

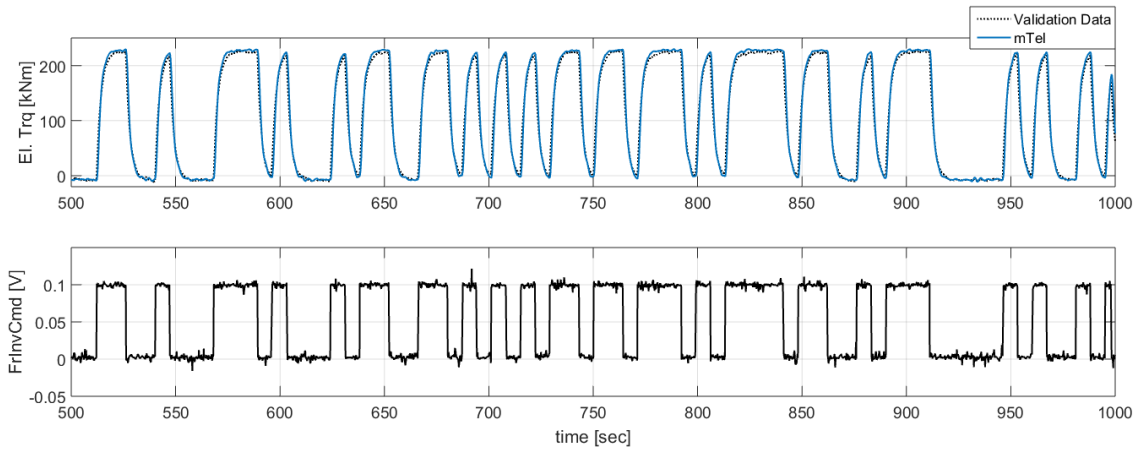

Figure 3.27: Fit of Electric Torque model Frequency Inverter Command change evaluation data

3.7 State Feedback Formulation

The models presented in section 3.5 are used for controller design, as internal models, with the task to predict the output trajectories within the prediction horizon. In some of these models, the input trajectories are not linearly independent from each other. The $FrInvCmd$ change of models (3.15) will result - besides λ value change - into a MAP value change, which also affect the λ output value of the model. So (3.15) models need a internal MAP model which feeds back its output value to the λ model, so that the future λ value could be estimated more accurately. Also the MIMO model developed in the field 3.5.4, except from the MAP input which changes with $FrInvCmd$ change, also uses the λ output as input for the exhaust emissions estimation.

A state feedback signal structure assumes that a system output value is treated also as input for the system, as shown in Fig. 3.28 and described by Eq. (3.26)-(3.31). The need of the state feedback manipulation is shown in Fig. 3.29, which shows the controller efficiency, with model m202 as internal model, with and without the use of state feedback.

$$x_i(t+1) = A_i x_i(t) + B_i u_i(t),$$
$$y_i(t) = C_i x_i(t), \tag{3.26}$$

and

$$x_j(t+1) = A_j x_j(t) + B_j u_j(t), \tag{3.27}$$
$$y_j(t) = C_j x_j(t), \tag{3.28}$$

but

$$u_j(t) = \begin{Bmatrix} y_i(t) & v_j(t) \end{Bmatrix}, \tag{3.29}$$

v_j :external Inputs/Disturbances

$$x_j(t+1) = \begin{bmatrix} A_j \\ B_j C_i \end{bmatrix} x_j(t) + B_j v_j(t) \tag{3.30}$$
$$y_j(t) = C_j x_j(t), \tag{3.31}$$

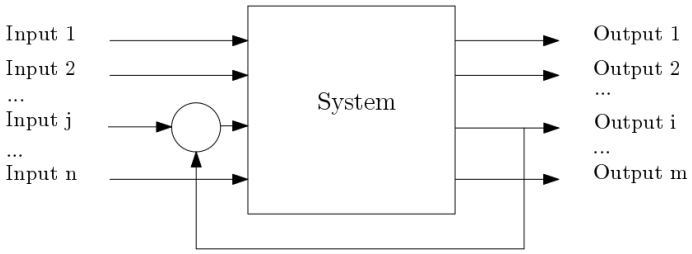

Figure 3.28: State Feedback Formulation

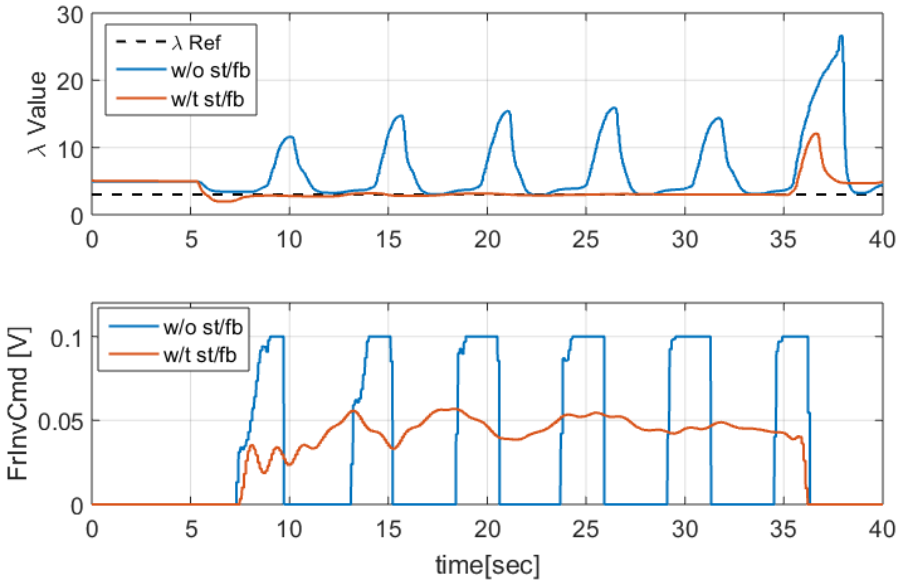

Figure 3.29: Experimental evaluation of State Feedback formulation

3.8 Analysis of Control-Oriented Models

This section is presents the spectral characteristics of the models that have been identified in section 3.5 with the State Feedback manipulation. The evaluation of the models has also been conducted, apart from the fitting results, on the time and frequency domain, as recomended by [3].

At first, mMAP model, wich feeds back the MAP change where needed, is presented in Fig. 3.30 and 3.31. The step and frequency response characteristics of the MISO models, as well as their poles-zeros maps are presented in Fig. 3.32, 3.33 and 3.34. Similar information is provided for the MIMO linear engine model of Eq. (3.22) in Fig. 3.35, 3.36 and 3.37.

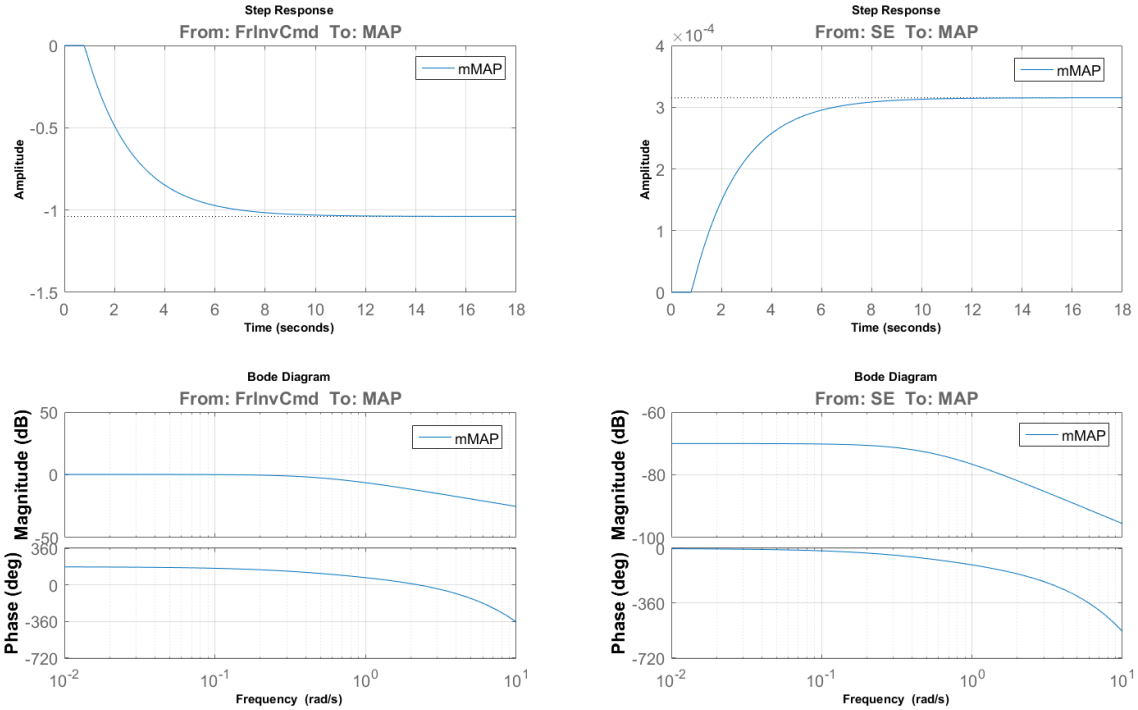

Figure 3.30: Step Response plots and Bode diagrams of model mMAP

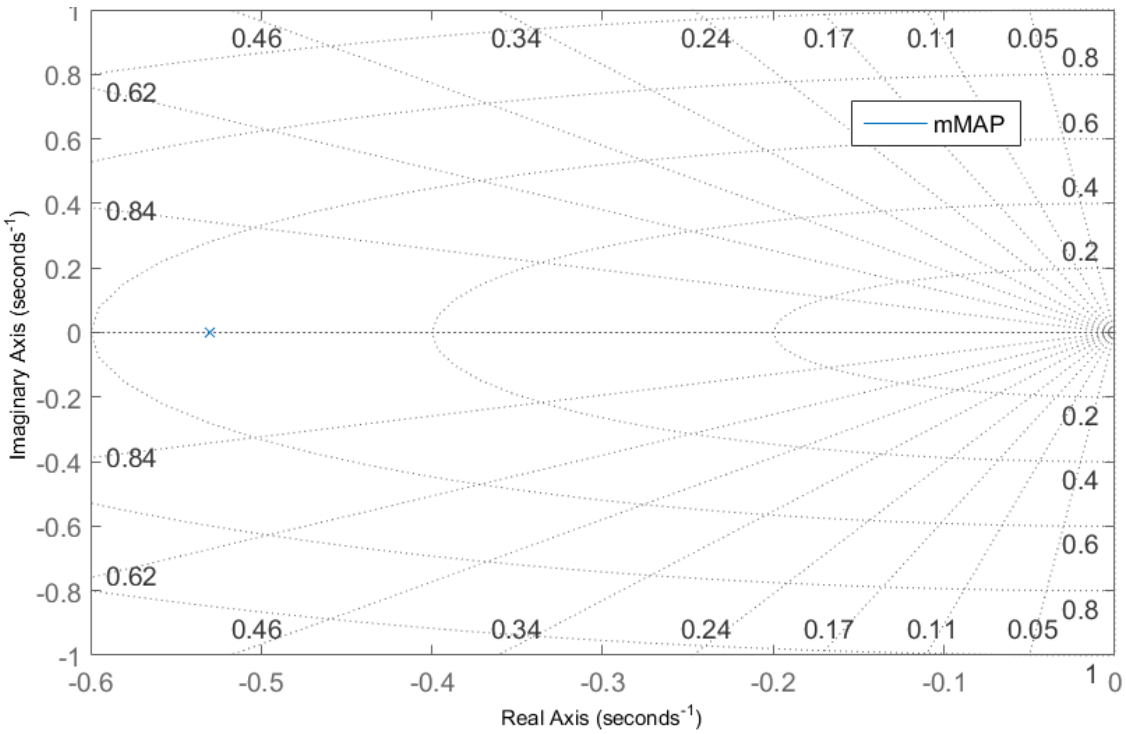

Figure 3.31: Poles-Zeros Map of model mMAP

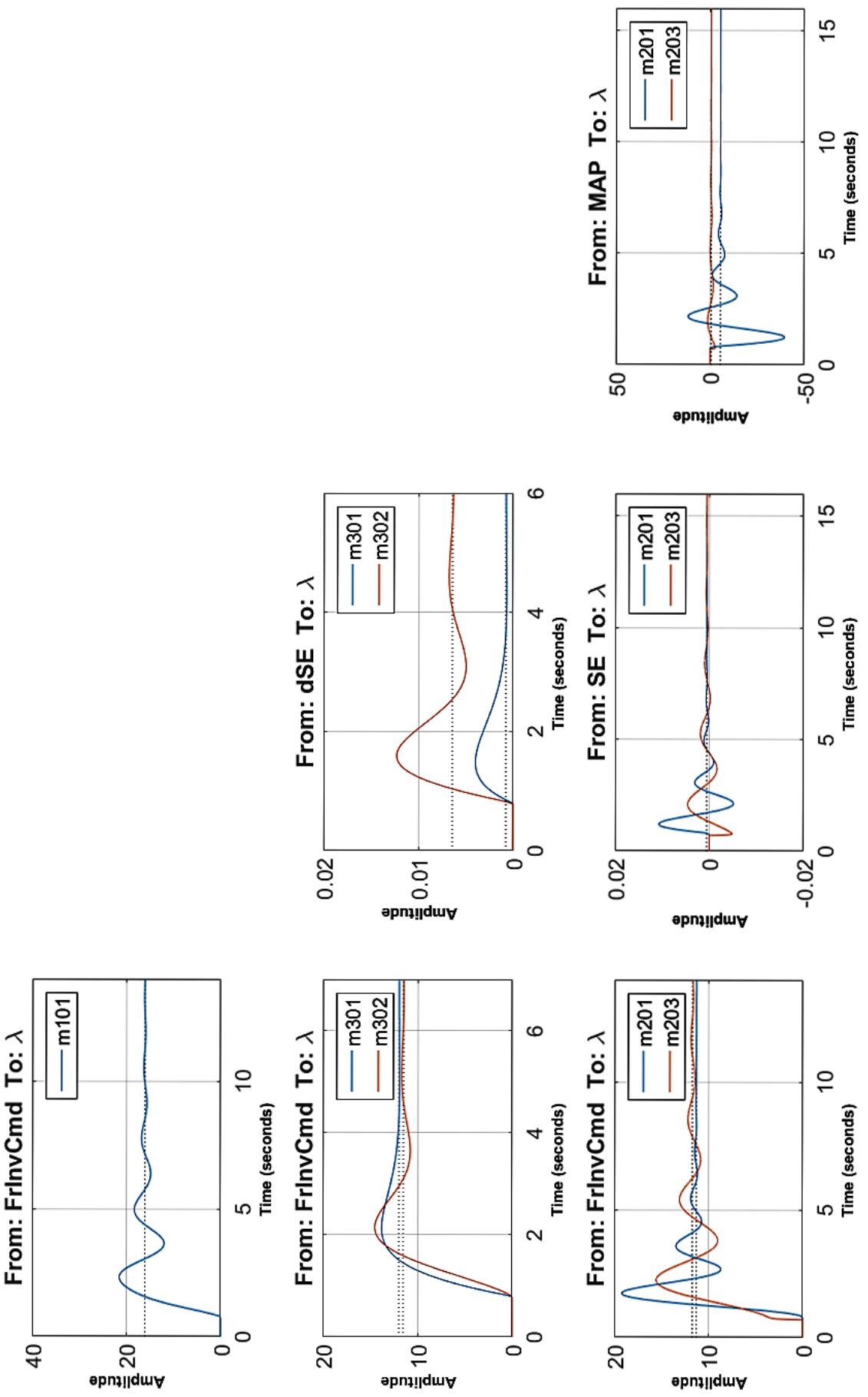

Figure 3.32: Step Response of λ output models with state feedback formulation

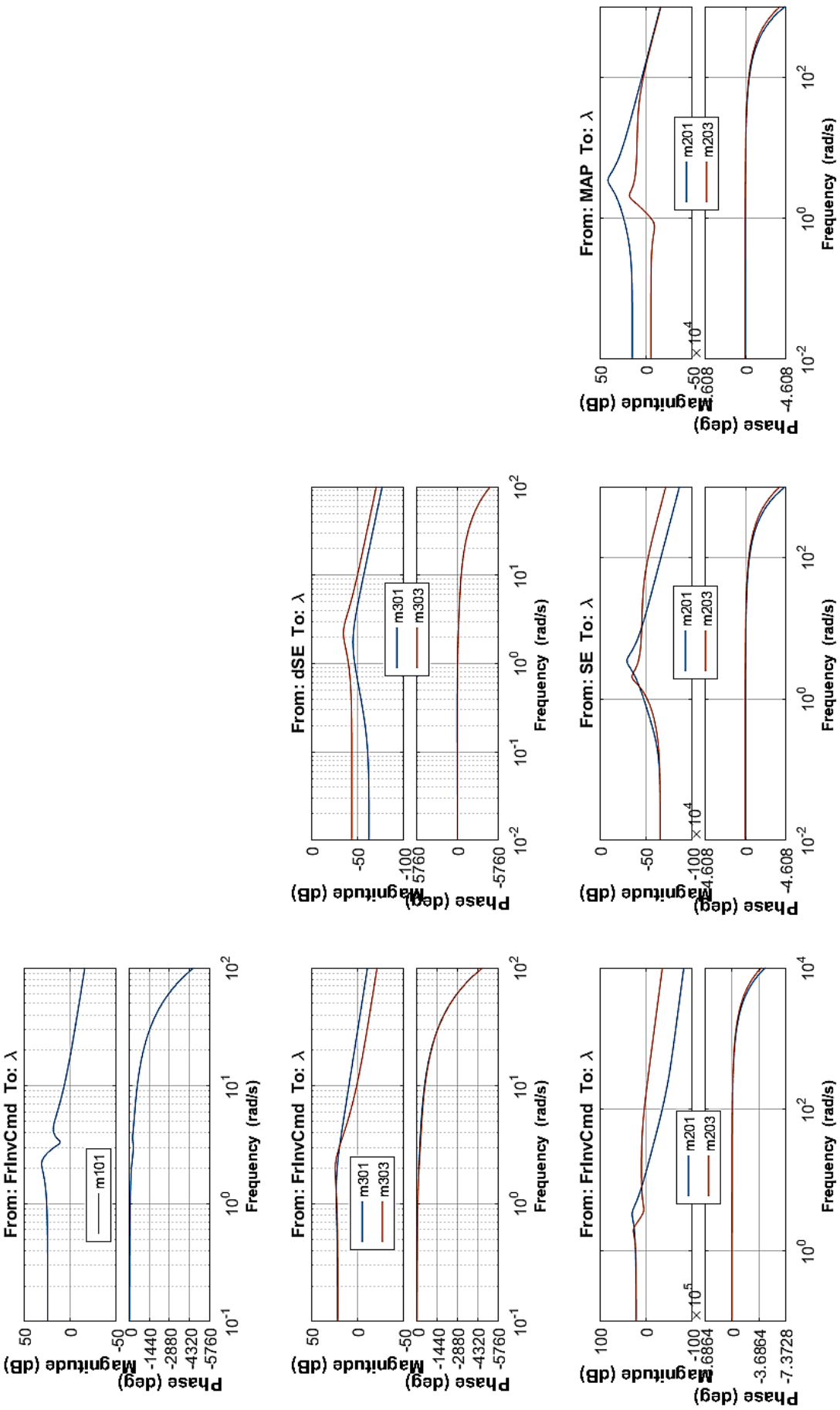

Figure 3.33: Frequency Response of λ output models with state feedback formulation

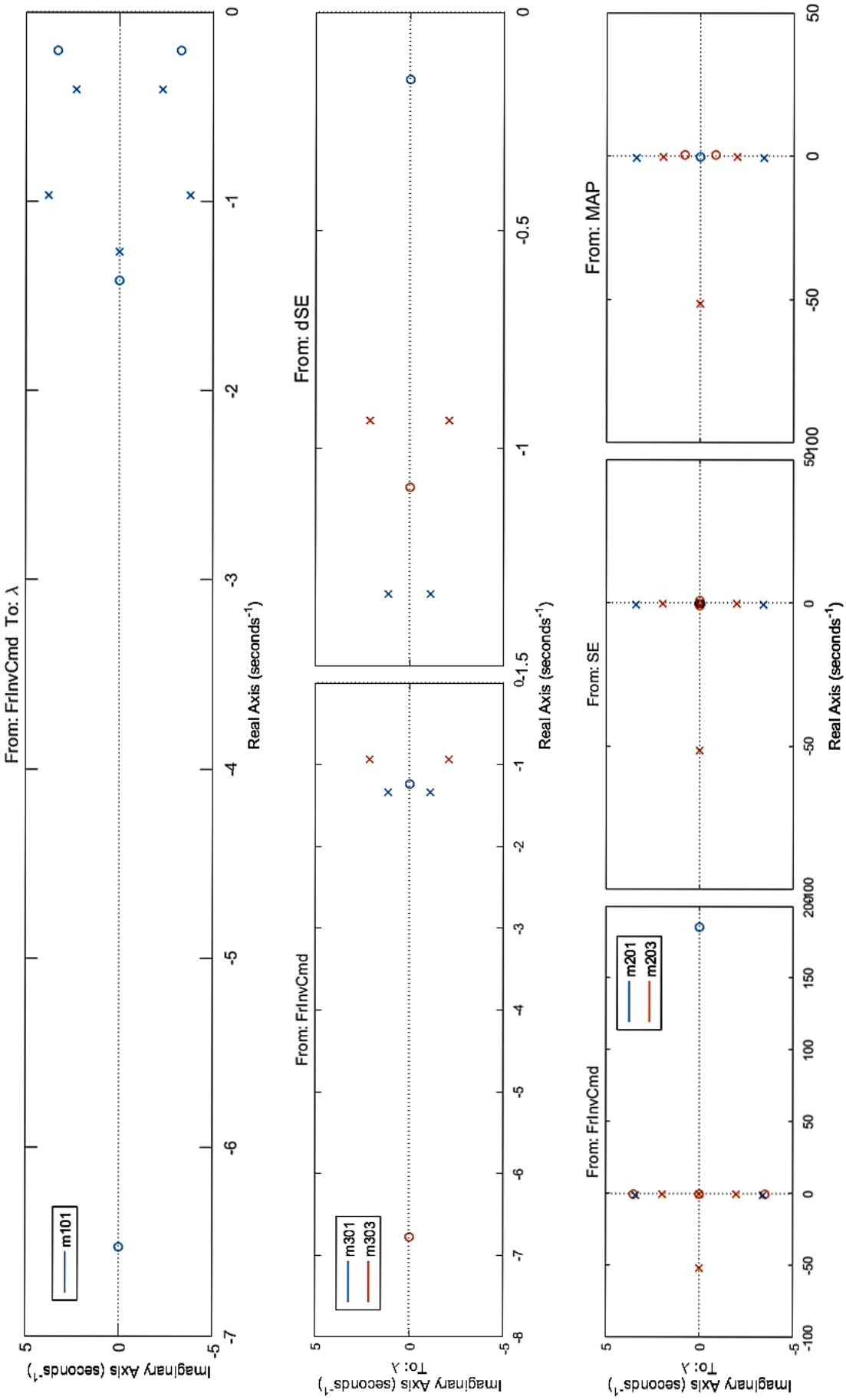

Figure 3.34: Poles-Zeros Map of λ output models with state feedback formulation

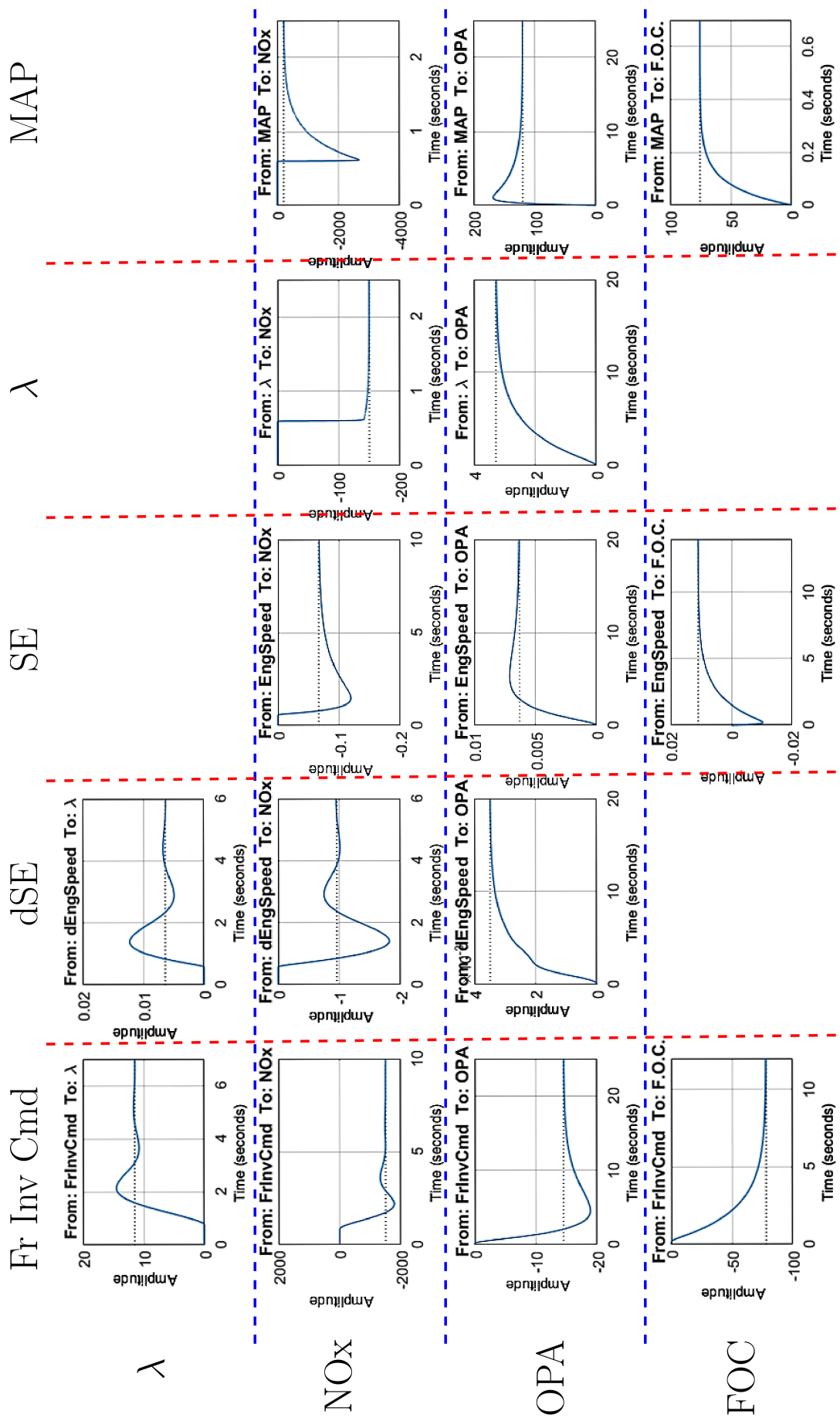

Figure 3.35: Step Response of MIMO engine model with state feedback formulation

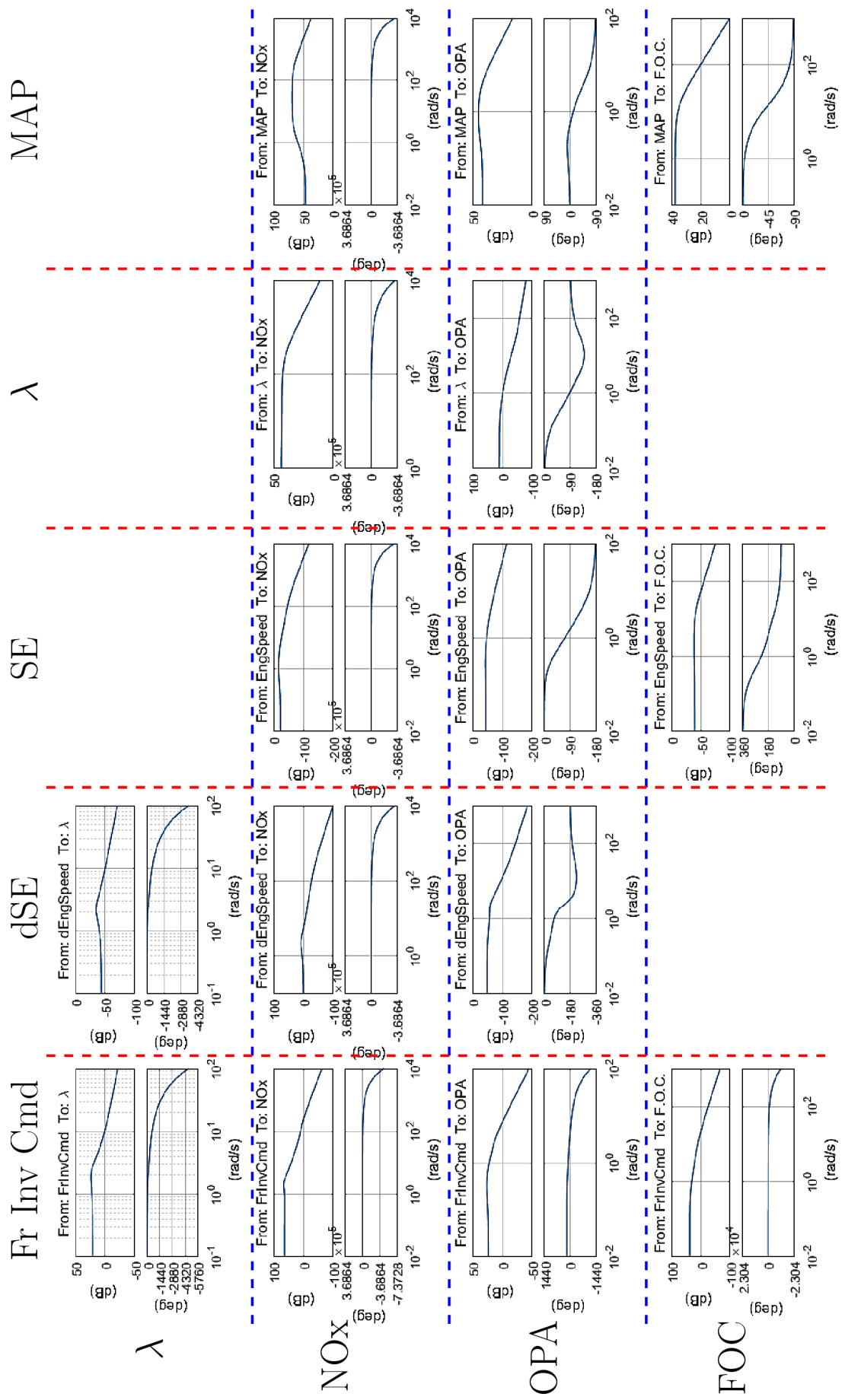

Figure 3.36: Bode Diagram of MIMO engine model with state feedback formulation

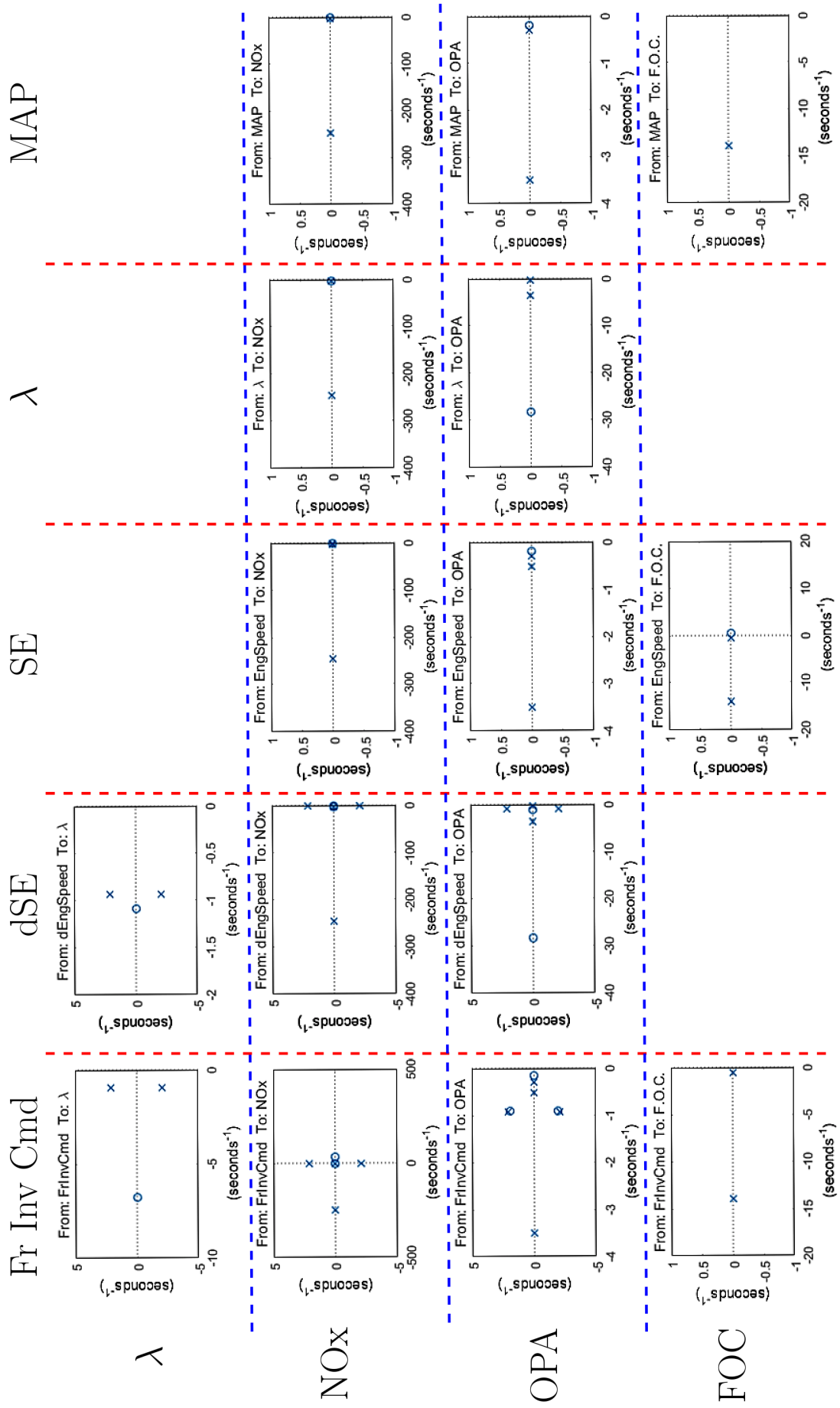

Figure 3.37: Poles-Zeros Map of MIMO engine model with state feedback formulation

Chapter 4

MPC Theory

4.1 MPC Problem Set-Up

4.1.1 Concept

Model Predictive Controller is a model-based controller which tries to compute the optimal sequence of the control moves in order to succeed the optimal control performance of a plant over a finite prediction horizon. As MPC can predict the future behavior of a system and plan an optimal control strategy, at the same time it can be aware of the limits and the constraints of the plant, hence reacting very differently to a disturbance which pushes the output towards the constraint compared to what it would do in response to a disturbance which pushes he plant away from it. Consequently, it is possible by adopting MPC strategy to operate a plant very close to its limits [8].
The basic idea of MPC regarding a SISO plant is presented in Fig. 4.1 [1].

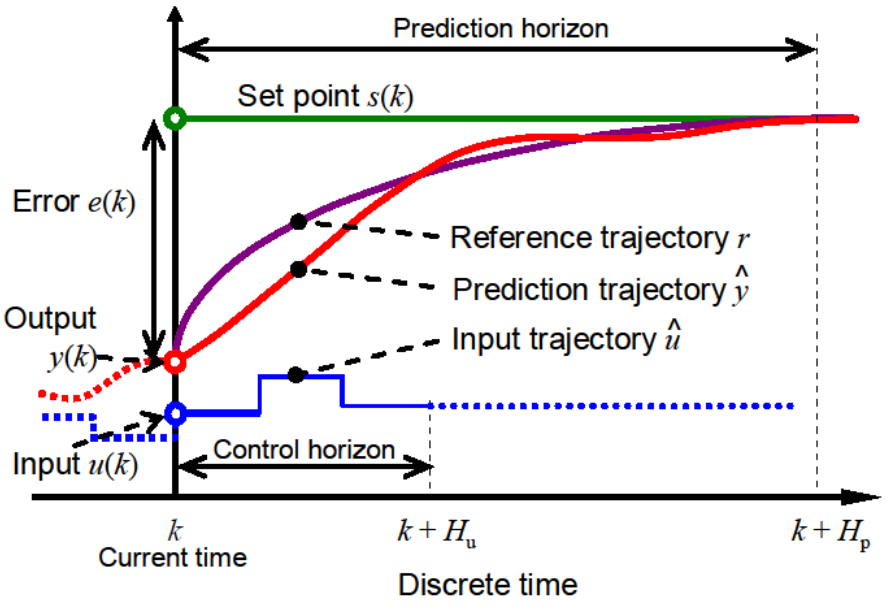

Figure 4.1: Concept of Model Predictive Controller, from [1]

The current time interval is labeled as k. At the current time the measured plant output is $y(k)$; the previous history of the output trajectory is also shown. By $s(k)$ the set-point trajectory is denoted, which should ideally be followed by the output. The

reference trajectory r defines the ideal trajectory that the output should follow in order to return to the set-point trajectory. This reference trajectory is a function of the controller tuning parameters. The current error value is defined as $e(k) = s(k) - y(k)$ and is formed over the prediction horizon H_p as

$$e(k+i|k) = s(k) - \hat{y}(k+i|k), \quad i = 1 : H_p \tag{4.1}$$

The notation $(k+i|k)$ indicates that the future value of a signal is depended on the conditions at time k. The trajectory $\hat{y}(k+i|k)$ is the controller prediction of the output value according to its internal model responding to the future sequence of the inputs $\hat{u}(k+i|k)$, $i = 1 : H_u$ of the controller, where $H_u \leq H_p$ is the control horizon and defines the acceptable control moves within the prediction horizon. The indication $\hat{u}$ means that the estimated value of $u(k+i|k)$ may be different from the actual input value $u(k+i)$ that will be applied at time interval $k+i$.

The concept of the MPC is to find the most promising control strategy of the input trajectory, that fits as well as possible the output trajectory to its reference, according to the conditions at time k.

Once the optimal input trajectory has been selected, the first control move $u(k) = u(k|k)$ is applied to the plant, until the new measurement $y(k+i)$ of the output is available in order to devise the new optimal control strategy at time $(k+1)$ over the new horizon $i = 2 : (H_p + 1)$. This strategy, where the H_p-length horizon slides by one sample interval at each step, is called *receding horizon strategy* [8]. If we suppose that, according to the internal model of MPC, the free response of the system is $\hat{y}_f(k+i|k)$ and S_u is the step input response of the system, then the estimation of the future output trajectory values at the end of prediction horizon can be expressed as

$$\hat{y}(k+H_p|k) = \hat{y}_f(k+H_p|k) + \sum_{i=0}^{H_u} S_u \Delta\hat{u}(k+i|k) \tag{4.2}$$

4.1.2 Definition of the Cost Function

The solution of finding the optimal sequence control moves is not explicitly known, as it is almost impossible for the output trajectory to coincide at each point of the reference trajectory within the prediction horizon. MPC tries to minimize a cost function (Eq. (4.3)); the solution this determines the behavior and performance of the controller.

$$\begin{aligned} J(z_k) = &\sum_{j=1}^{n_y}\sum_{i=1}^{p}\left\{\frac{w_{i,j}^{y}}{s_j^{y}}[r_j(k+i|k) - \hat{y}_j(k+i|k)]\right\}^2 + \\ &+\sum_{j=1}^{n_u}\sum_{i=0}^{p-1}\left\{\frac{w_{i,j}^{u}}{s_j^{u}}[u_j(k+i|k) - u_{j,target}(k+i|k)]\right\}^2 + \\ &+\sum_{j=1}^{n_u}\sum_{i=0}^{p-1}\left\{\frac{w_{i,j}^{\Delta u}}{s_j^{u}}[u_j(k+i|k) - u_j(k+i-1|k)]\right\}^2 + \\ &+ \rho_\epsilon \varepsilon_k^2 \end{aligned} \tag{4.3}$$

where

- z_k - Optimization process decision, given by

$$z_k^T = \begin{bmatrix} u(k|k)^T & u(k+1|k)^T & \dots & u(k+p-1|k)^T & {}_k \end{bmatrix}$$

.

- k - Current control interval.
- p - Number of prediction intervals.
- $r_j(k+i|k)$ - Reference value for jth output at ith prediction horizon step.
- $\hat{y}_j(k+i|k)$ - Predicted value of jth output at ith prediction horizon step.
- $u_j(k+i|k)$ - The control command of the jth system input at $(k+i)$th control interval, given by z_k function.
- $u_{j,target}(k+i|k)$ - Target value for jth manipulated variable at $(k+i)$th control interval, given by z_k function.
- $w^k_{i,j}$ - Tuning weight of jth variable of the kth controller item at ith prediction horizon step.
- s^k_j - Scale factor for jth variable of the kth controller item of the system, in engineering units.
- ε_k - Slack variable at control interval k.
- ρ_ϵ - Constraint violation penalty weight.

4.1.3 Constraints

Certain constraints are not defined explicitly, but instead are implied. For example if $H_u < H_p$ then certain control moves are constrained to zero. Explicit constraints are defined as described below

$$\frac{y_{j,min}(i)}{s^y_j} - \epsilon_k V^y_{j,min} \leq \frac{y_j(k+i|k)}{s^y_j} \leq \frac{y_{j,max}(i)}{s^y_j} + \epsilon_k V^y_{j,max}, \quad i = i:p, \quad j = 1:n_y \tag{4.4}$$

$$\frac{u_{j,min}(i)}{s^u_j} - \epsilon_k V^u_{j,min} \leq \frac{u_j(k+i|k)}{s^u_j} \leq \frac{u_{j,max}(i)}{s^u_j} + \epsilon_k V^u_{j,max}, \quad i = i:p, \quad j = 1:n_u \tag{4.5}$$

$$\frac{\Delta u_{j,min}(i)}{s^u_j} - \epsilon_k V^{\Delta u}_{j,min} \leq \frac{\Delta u_j(k+i|k)}{s^u_j} \leq \frac{\Delta u_{j,max}(i)}{s^u_j} + \epsilon_k V^{\Delta u}_{j,max}, \quad i = i:p, \quad j = 1:n_u \tag{4.6}$$

where

- $y_{j,min}(i), y_{j,max}(i)$ - Lower and upper bounds for jth plant output at ith prediction horizon step.
- $u_{j,min}(i), u_{j,max}(i)$ - Lower and upper bounds for jth plant input at ith prediction horizon step.
- $\Delta u_{j,min}(i), \Delta u_{j,max}(i)$ - Lower and upper bounds for jth plant input increment at ith prediction horizon step.
- ϵ_k - Scalar QP slack variable[1] used for constraint softening.
- $V^k_{j,min}, V^k_{j,max}$ - Dimensionless controller constants (ECR values) used for constraint softening and expresses the softened limits of the specific system's variable.

[1] slack variable is a variable that is added to an inequality constraint to transform it to an equality. Introducing a slack variable replaces an inequality constraint with an equality constraint and a nonnegativity constraint

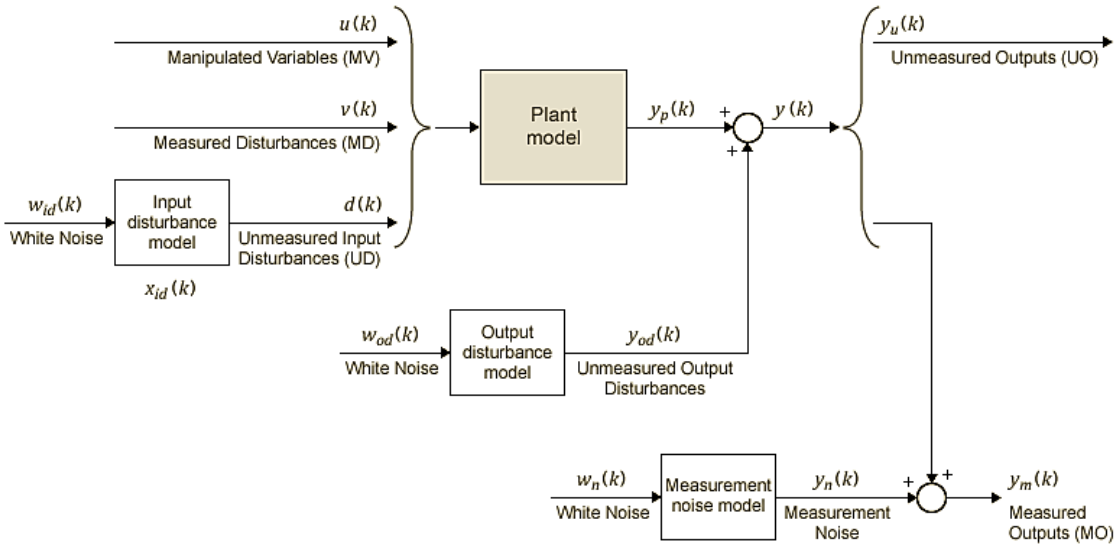

Figure 4.2: MPC state vectors representation, from [9]

4.2 State Estimation

At each control interval, MPC controller uses its current state x_c as initial basis for predictions [9].

$$x_c^T(k) = \begin{bmatrix} x_p^T(k) & x_{id}^T(k) & x_{od}^T(k) & x_n^T(k) \end{bmatrix} \tag{4.7}$$

where

- x_c - the controller state, comprising $n_{xp} + n_{id} + n_{od} + n_{xn}$ state variables
- x_p - the state vector of the plant, of length n_{xp}
- x_{id} - the state vector of the input disturbance, of length n_{xid}
- x_{od} - the state vector of the output disturbance, of length n_{xod}
- x_n - the state vector of the noise model, of length n_{xn}

The state vectors are shown in Fig. 4.2, from [9]. The controller updates its state using the latest plant measurements.

State Observer

The state observer is generated by a combination of models, as shown in Fig. 4.2

$$\begin{aligned} x_c(k+1) &= Ax_c(k) + Bu_o(k) \\ y(k) &= Cx_c(k) + Du_o(k) \end{aligned} \tag{4.8}$$

The state observer estimates the values of unmeasured states and predict how the proposed sequence of the MV changes will affect future plant output values.

The input signals of the observer is the dimensionless vector with plant manipulated and measured disturbance inputs and the white noise inputs to the disturbance and noise models.

$$u_o^T(k) = \begin{bmatrix} u^T(k) & v^T(k) & w_{id}^T(k) & w_{od}^T(k) & w_n^T(k) \end{bmatrix} \tag{4.9}$$

The disturbance and noise models are zero mean value Gaussian white noise.
The observer outputs are the n_y dimensionless plant outputs.

The parameters of the observer defining the models shown in Fig. 4.2 are

$$A = \begin{bmatrix} A_p & B_{pd}Cid & 0 & 0 \\ 0 & A_{id} & 0 & 0 \\ 0 & 0 & A_{od} & 0 \\ 0 & 0 & 0 & A_n \end{bmatrix}, B = \begin{bmatrix} B_{pu} & B_{pv} & BpdDid & 0 & 0 \\ 0 & 0 & B_{id} & 0 & 0 \\ 0 & 0 & 0 & B_{od} & 0 \\ 0 & 0 & 0 & 0 & B_n \end{bmatrix},$$
$$C = \begin{bmatrix} C_p & D_{pd}C_{id} & C_{od} & \begin{bmatrix} C_n \\ 0 \end{bmatrix} \end{bmatrix}, D = \begin{bmatrix} 0 & D_{pv} & D_{pd}D_{id} & D_{od} & \begin{bmatrix} D_n \\ 0 \end{bmatrix} \end{bmatrix} \tag{4.10}$$

State Estimation

MPC uses a steady state Kalman filter that yields from the state observer. At the beginning of the k^{th} control interval the controller state is estimated with the following steps

1. obtain the following data and scale plant input and output signals in order top be used dimensionless in calculations
   - $x_c(k|k-1)$ - Controller state estimate from previous control interval, $k-1$.
   - $u_{act}(k-1)$ - The actually used MV in the plant from $k-1$ to k.
   - $u_{opt}(k-1)$ - The optimal MV recomended by MPC and supposed to have been used in the plant from $k-1$ to k.
   - $v(k)$ - Current measured disturbances.
   - $y_m(k)$ - Current measured outputs of the plant.
   - B_u, B_v - Columns of observer parameter B corresponding to u_k and v_k inputs of the plant.
   - C_m - Rows of observer parameter C corresponding to measured outputs of the plant.
   - D_m - Rows and columns of observer parameter D corresponding to measured outputs and measured disturbance inputs of the plant.
   - L, M - Constant Kalman gain matrices, as defined in the next field.
2. revise $x_c(k|k-1)$ when $u_{act}(k-1) \neq u_{opt}(k-1)$ as follows
$$x_c^{rev}(k|k-1) = x_c(k|k-1) + B_u[u_{act}(k-1) - u_{opt}(k-1)] \tag{4.11}$$
3. compute the innovation
$$e(k) = y_m(k) - [C_m x_c^{rev}(k|k-1) + D_{mv}v(k)] \tag{4.12}$$
4. Update the controller state estimate to account for the latest measurements
$$x_c(k|k) = x_c^{rev}(k|k-1) + Me(k)] \tag{4.13}$$

The current state estimete $x_c(k|k)$ is used to solve the QP problem at control interval k. The optimal V move $u^{opt}(k)$ is to be used in the plant between k and $k+1$.

Finally, the controller prepares for the next control interval, assuming that the unknown inputs $w_{id}(k)$, $w_{od}(k)$, $w_n(k)$ have zero mean value between k and $k+1$. The impact of the known inputs and the innovation is predicted as follows

$$x_c(k+1|k) = Ax_c^{rev}(k|k-1) + B_u u^{opt}(k) + B_v v(k) + Le(k) \tag{4.14}$$

Steady-State Kalman Gains Calculation

MPC uses the L and M Kalman estimator gains, assuming that state observer parametrs are time-invariant, the controller states x_c are detectable, stohastic inputs $w_{id}(k)$, $w_{od}(k)$, $w_n(k)$ are independent white noise of zero mean and identity covariance. Additionally, white noise $w_u(k)$ and $w_v(k)$ is added to the dimensionless $u(k)$ and $v(k)$ inputs respectively. Without loss of generality, the $u(k)$ and $v(k)$ inputs are set zero. So the effect of the stohastic inputs to the controller states and measured plant outputs is

$$\begin{aligned} x_c(k+1) &= Ax_c(k) + Bw(k) \\ y_m(k) &= C_m x_c(k) + D_m w(k) \end{aligned} \tag{4.15}$$

where

$$w^T(k) = \begin{bmatrix} w_u^T(k) & w_v^T(k) & w_{id}^T(k) & w_{od}^T(k) & w_n^T(k) \end{bmatrix} \tag{4.16}$$

Inputs to the Kalman estimator are the state observer matrices A, C_m, as well as the following covariance matrices

$$Q = E\left\{Bww^TB^T\right\} = BB^T \tag{4.17}$$

$$R = E\left\{D_m ww^T D_m^T\right\} = D_m D_m^T \tag{4.18}$$

$$N = E\left\{Bww^T D_m^T\right\} = BD_m^T \tag{4.19}$$

where $E\{\ldots\}$ denotes the expectation.

Output Variable Prediction

The key application of the state observer is the prediction of noise-free future plant outputs that are required by the MPC to be used in the optimization process.

At k^{th} control interval, the required data is the following

- $p \geq 1$ - prediction horizon
- $x_c(k|k)$ - Controller state estimates
- $v(k+i|k) = v(k)$ - Current and future measured disturbances inputs, where $i = 1 : (p-1)$
- A, B_u, B_v, C, D_v - tate observer constants, where B_u, B_v and D_v are the columns of B and D matrices of the inputs u and v.

The expectation of the unknown white noise is assumed to be zero, as well as the plant outputs to be noise free. As a result, all terms describing the measurement noise are not included in the state observer equations. This is equivalent to zeroing the last $n \times n$ elements of $x_c(k|k)$. State observer predicts for the first step

$$x_c(k+1|k) = Ax_c(k|k) + B_u u(k|k) + B_v v(k) \tag{4.20}$$

Generalising for the rest steps $i = 2 : p$

$$x_c(k+i|k) = Ax_c(k+i-1|k) + B_u u(k+i-1|k) + B_v v(k+i-1|k) \tag{4.21}$$

The predicted noise-free plant outputs at any step $i = 1 : p$ are

$$x_c(k+i|k) = Ax_c(k+i|k) + D_v v(k+i|k) \tag{4.22}$$

For faster computations, the MPC controller uses an alternative form of the above equations, which are described in the following section.

4.3 Solution of the Optimization Problem

MPC solves at each control interval an optimization problem, with the use of quadratic programming (QP). QP solver determines the MV moves over the control horizon. From the QP decision, only the first move is applied until the next control interval. The QP problem deals with the following issues

- the cost function, which is a scalar non-negative measurement of controller performance to be minimized,
- constraints, which are the operational conditions of the plant that the solution must satisfy, such as te physical bounds of MV and the plant output variables.

Finally, the decision of QP solver includes the MV adjustments that minimize the cost function over the prediction horizon and, at the same time, satisfy the constraints.

4.3.1 QP Matrices

Prediction

The prediction model has the following form, assuming that there is a unit white Gaussian noise as input disturbance, such that $d(k) = n_d(k)$. Then the problem is denoted as follows

$$x \leftarrow \begin{bmatrix} x \\ x_d \end{bmatrix}, A \leftarrow \begin{bmatrix} A & B_d\bar{C} \\ 0 & \bar{A} \end{bmatrix}, B_u \leftarrow \begin{bmatrix} B_u \\ 0 \end{bmatrix}, B_v \leftarrow \begin{bmatrix} B_v \\ 0 \end{bmatrix}, B_d \leftarrow \begin{bmatrix} B_d\bar{D} \\ \bar{B} \end{bmatrix}, C \leftarrow \begin{bmatrix} C & D_dC \end{bmatrix}$$

Then the prediction model is

$$x(k+1) = Ax(k) + B_u u(k) + B_v v(k) + B_d n_d(k)$$

$$y(k) = Cx(k) + D_v v(k) + D_d n_d(k)$$

Next, the problem of predicting the future trajectories of the model performed at time k=0 is being considered. Setting $n_d(i) = 0$ for all prediction instants, it is obtained

$$y(i|0) = C\left[A^i x(0) + \sum_{h=0}^{i-1} A^{i-1}\left(B_u\left(u(-1) + \sum_{j=0}^{h}\Delta u(j)\right) + B_v v(h)\right)\right] + D_v v(i)$$

This equation gives the solution

$$\begin{bmatrix} y(1) \\ \vdots \\ y(p) \end{bmatrix} = S_x x(0) + S_{u1} u(-1) + S_u \begin{bmatrix} \Delta u(0) \\ \vdots \\ \Delta u(p-1) \end{bmatrix} + H_v \begin{bmatrix} v(0) \\ \vdots \\ v(p) \end{bmatrix}$$

where

$$S_x = \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^p \end{bmatrix} \in \mathbb{R}^{pn_y \times n_x}, S_{u1} = \begin{bmatrix} CB_u \\ CB_u + CAB_u \\ \vdots \\ \sum_{h=0}^{p-1} CA^h B_u \end{bmatrix} \in \mathbb{R}^{pn_y \times n_u}$$

$$S_u = \begin{bmatrix} CB_u & 0 & \dots & 0 \\ CB_u + CAB_u & CB_u & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{h=0}^{p-1} CA^h B_u & \sum_{h=0}^{p-2} CA^h B_u & \dots & CB_u \end{bmatrix} \in \mathbb{R}^{pn_y \times pn_u}$$

$$H_v = \begin{bmatrix} CB_v & D_v & 0 & \dots & 0 \\ CAB_v & CB_v & D_v & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{p-1}B_v & CA^{p-2}B_v & CA^{p-3}B_v & \dots & D_v \end{bmatrix} \in \mathbb{R}^{pn_y \times (p+1)n_v}$$

Now the optimization variables will be denoted. Let m be the number of free control moves, then

$$\begin{bmatrix} \Delta u(0) \\ \vdots \\ \Delta u(p-1) \end{bmatrix} = J_M \begin{bmatrix} z_0 \\ \vdots \\ z_{m-1}) \end{bmatrix} = J_M z$$

where the $p \times m$ table J_M depends on the choice of blocking moves. Together with the slack variable ε, vectors $z_0, ..., z_{m-1}$ constitute the free optimization variables of the optimization problem. In this work, where only one MV exists, $z_0, ..., z_{m-1}$ are scalars.

Cost function

The function to be optimized is

$$J(z,\varepsilon) = \begin{bmatrix} u(0) \\ \vdots \\ u(p-1) \end{bmatrix}^T W_u^2 \begin{bmatrix} u(0) \\ \vdots \\ u(p-1) \end{bmatrix} + \begin{bmatrix} \Delta u(0) \\ \vdots \\ \Delta u(p-1) \end{bmatrix}^T W_{\Delta u}^2 \begin{bmatrix} \Delta u(0) \\ \vdots \\ \Delta u(p-1) \end{bmatrix} +$$

$$+ \left(\begin{bmatrix} \hat{y}(1) \\ \vdots \\ \hat{y}(p) \end{bmatrix} - \begin{bmatrix} r(1) \\ \vdots \\ r(p) \end{bmatrix} \right)^T W_y^2 \left(\begin{bmatrix} \hat{y}(1) \\ \vdots \\ \hat{y}(p) \end{bmatrix} - \begin{bmatrix} r(1) \\ \vdots \\ r(p) \end{bmatrix} \right) + \rho_\varepsilon \varepsilon^2$$

where

$$W_u = w_u I^{p \times n_u}$$

$$W_{\Delta u} = w_{\Delta u} I^{p \times n_u}$$

$$W_y = w_y I^{p \times n_y}$$

Finally, after substituting $u(k),\ \Delta u(k),\ y(k),\ J(z,\varepsilon)$ can be rewritten as

$$J(z,\varepsilon) = \rho_\varepsilon \varepsilon^2 + z^T K_{\Delta u} z +$$

$$+2 \left(\begin{bmatrix} r(1) \\ \vdots \\ r(p) \end{bmatrix}^T K_r + \begin{bmatrix} v(0) \\ \vdots \\ v(p) \end{bmatrix}^T K_v + u(-1)^T K_u + x(0)^T K_x \right) z +$$

$$+ constant$$

Constraints

In this fiels, the limits on inputs, input increments and outputs along with the constraint $\varepsilon \geq 0$ are being considered.

$$\begin{bmatrix} y_{min}(1) - \varepsilon V^{y}_{min}(1) \\ \vdots \\ y_{min}(p) - \varepsilon V^{y}_{min}(p) \\ u_{min}(0) - \varepsilon V^{u}_{min}(0) \\ \vdots \\ u_{min}(p-1) - \varepsilon V^{u}_{min}(p-1) \\ \Delta u_{min}(0) - \varepsilon V^{\Delta u}_{min}(0) \\ \vdots \\ \Delta u_{min}(p-1) - \varepsilon V^{\Delta u}_{min}(p-1) \end{bmatrix} \leq \begin{bmatrix} y(1) \\ \vdots \\ y(p)u(0) \\ \vdots \\ u(p-1) \\ \Delta u(0) \\ \vdots \\ \Delta u(p-1) \end{bmatrix} \leq \begin{bmatrix} y_{max}(1) - \varepsilon V^{y}_{max}(1) \\ \vdots \\ y_{max}(p) - \varepsilon V^{y}_{max}(p) \\ u_{max}(0) - \varepsilon V^{u}_{max}(0) \\ \vdots \\ u_{max}(p-1) - \varepsilon V^{u}_{max}(p-1) \\ \Delta u_{max}(0) - \varepsilon V^{\Delta u}_{max}(0) \\ \vdots \\ \Delta u_{max}(p-1) - \varepsilon V^{\Delta u}_{max}(p-1) \end{bmatrix}$$

and similar to what was done to the cost function, substituting $u(k),\ \Delta u(k),\ y(k)$, it is obtained

$$M_z z + M_\varepsilon \varepsilon \leq M_{lim} + M_v \begin{bmatrix} v(0) \\ \vdots \\ v(p) \end{bmatrix} + M_u u(-1) + M_x x(0)$$

matrices $Mz,\ M_\varepsilon,\ M_{lim},\ M_v,\ M_u,\ M_x$ are obtained from the upper and lower bounds and ECR values.

4.3.2 QP Solver

Unconstrained Case

The optimal solution is explicitly computed by

$$z^* = -K^{-1}_{\Delta u} \left(\begin{bmatrix} r(1) \\ \vdots \\ r(p) \end{bmatrix}^T K_r + \begin{bmatrix} v(0) \\ \vdots \\ v(p) \end{bmatrix}^T K_v + u(-1)^T K_u + x(0)^T K_x \right)^T$$

Constrained Case

The model predictive controller QP solver converts an MPC optimization problem to the general QP form [9]

$$\min_x (\frac{1}{2} x^{\text{œ}} H x + f^{\text{œ}} c)$$

such that

$$Ax \leq b$$

where

- x is the solution vector.
- H is the Hessian matrix.
- A is a matrix of linear constraint coefficients.
- b and f are vectors.

H and A matrices are constants. The controller computes these constant matrices during initialization and retrieves them from computer memory when needed. It computes the time-varying b and f vectors at the beginning of each control instant.

KWIK (Knows What it Knows) algorithm [11] is used to solve the QP problem, which requires the Hessian to be positive definite. In the first control step, KWIK uses as initial guess is the unconstrained solution. If x satisfies the constraints, it is the optimal QP solution, x*, and the algorithm terminates. Otherwise at least one of the linear inequality constraints must be satisfied as an equality. In this case, KWIK uses an efficient, numerically robust strategy to determine the active constraint set satisfying the standard optimization conditions.

A KWIK algorithm begins with an input set X and output set Y. The hypothesis class H consists of aset of functions from X to Y : $H \subseteq (X \rightarrow Y)$ [12]. The target function $h^* \in H$ is unknown to the learner. The hypothesis class H and parameters ε and δ are known to both the learner and environment. The environment selects a target function $h* \in H$ adversarially.
The agent then repeats the following

1. The environment selects an input $x \in X$ adversarially and informs the learner
2. The learner predicts an output $\hat{y} \in Y \cup \perp$ where $\perp$ means "I don't know".
3. If $\hat{y} \neq \perp$, it should be accurate: $|\hat{y} - y| \leq \varepsilon$ where $y = h^*(x)$. Otherwise the entire run is considered a failure. The probability of a failed run must be bounded by δ
4. If $\hat{y} = \perp$, the learner makes an observation $z \in Z$ of the output, where $z = y$ in the deterministic case, $z = 1$ with probability y and 0 with probability $1 - y$ in the Bernoulli case, or $z = y + n$ for the zero-mean random variable n in the additive noise case.

In the following control steps, the active constraint set determined at the previous control step becomes the initial guess for the next.

Chapter 5

Controller Design

5.1 Controller Synthesis and Tuning

The aims of the Controller Design procedure are to ensure

- stability of the system,
- reference tracking of the control object (in this work's case λ Value),
- disturbance rejection during loading, unloading or during the application of a constand load on the diesel engine and
- that the system under closed loop control operation maintains performance within the desired operational limits.

In this work Model Predictive Controller (MPC) was used, as it is capable of handling complex systems with many parameters, such as the Hybrid Power Plant Installation of LME. The use of multiple input models will help MPC to predict more accurately the future values of the output trajectory.

The λ value is controlled by changing the $FrInvCmd$, so $FrInvCmd$ is the only Manipulated Variable (MV). The other system inputs are treated as Measured Disturbances (MD) and are used so as to provide a better output trajectory prediction over the prediction horizon (N_p).

For the solution of the optimization problem, quadratic programming (QP) is used in order to minimize the cost function (Eq. (4.3)) by computing the N_u optimal sequence of the MV ($FrInvCmd$) moves over the Control Horizon (N_u) [9, 10]. The simplified cost function which is used in this work is the following

$$J(z_k) = \underbrace{\sum_{i=1}^{p} \left\{ \frac{w_\lambda}{s_\lambda} \left[\lambda_{Ref}(k) - \hat{\lambda}(k+i|k) \right] \right\}^2}_{J_y} + \underbrace{\sum_{i=0}^{p-1} \left\{ \frac{w_u}{s_u} u(k+i|k) \right\}^2}_{J_u} + \\ + \underbrace{\sum_{i=0}^{p-1} \left\{ \frac{w_{\Delta u}}{s_u} \Delta u(k+i|k) \right\}^2}_{J_{\Delta u}} + \underbrace{\rho_\epsilon \varepsilon_k^2}_{J_{cst}} \quad , \tag{5.1}$$

where

- z_k - Optimization process decision, given by

$$z_k^T = \begin{bmatrix} u(k|k)^T & u(k+1|k)^T & \dots & u(k+p-1|k)^T & {}_k \end{bmatrix}$$

- k - Current control interval.
- p - Number of prediction intervals.
- $\lambda_{Ref}(k)$ - Reference value for λ value at current control interval.
- $\hat{\lambda}(k+i|k)$ - Predicted value of λ at ith prediction horizon step.
- $u(k+i|k)$ - Optimal $FrInvCmd$ value predicted for $(k+i)$th control interval, given by z_k function.
- $\Delta u = u(k+i|k) - u(k+i-1|k)$ - Move of the $FrInvCmd$ value between the Horizon intervals used for manipulated variable move suppression. As control horizon is smaller than prediction horizon, Δu is constrained to zero for certain steps in the prediction horizon.
- w_j - Tuning weight of jth variable of the controller
- s_j - Scale factor for jth variable of the system, in engineering units.
- ε_k - Slack variable at control interval k.
- ρ_ϵ - Constraint violation penalty weight.

The controller tuning parameters are selected so that the controller's response has the desirable characteristics. Sample time in the working platform of MATLAB was set to 0.1 *sec*. In general, the Prediction Horizon (N_p) was selected about as 20% of the time duration of the transient phenomenon and Control Horizon (N_u) was selected between $5-30$ % of N_p. Finally, the relative weighting between the reference tracking (w_λ), the command value (w_u) and the rate of the command change ($w_{\Delta u}$) was selected so that the controller response is relative to the dynamics of the Electric Motor (Eq. (3.24)).

The problem is hard constrained by the physical limits of the electric motor system, with the Electric Motor Torque Command constrained as follows

$$\frac{u_{min}}{s_u} \leq \frac{u(k+i|k)}{s_u} \leq \frac{u_{max}}{s_u},\ i = 0 : p-1 \tag{5.2}$$

Frequency Inverter Command must remain within $[0, 0.1]$ $[V]$ The designed controllers and their tuning parameters are illustrated in Fig. 5.1, in tabular form.

MPC 101

MPC 101 has the SISO model (m101) as internal model and was designed in order to evaluate the advantage of measured disturbances against controllers based on SISO models. MPC 101 computes the optimal MV change sequence, according to its internal model, trying to cope with the control command's consequence on λ value, without taking into account plant's disturbances as speed change, e.t.c, which could also influence the λ value within the finite prediction horizon.

MPC 201

MPC 201 uses model m202 which handles SE and MAP inputs as MD and also uses the State Feedback Formulation in order to predict the future MAP change according to the control command.

MPC 401

MPC 401 has m303 as internal model with dSE as MD signal. As speed changes immediately with the load change, MPC 401 can predict the upcoming λ value change after about 0.8 seconds; enough amount time to act preventively in a system with input and output delays.

MPC 900, 901 and 902

In MPC 900, MPC 901 and MPC 902 scenario, NO_x emissions and Fuel Oil Consumption (FOC) are also taken into account and soft constraints are applied in order to cope with environmental and operational limitations regarding these system outputs. When exceeding these softened limits a cost is added to the cost function forcing the controller, with the purpose if it is possible, to restore the system within the limits. Soft constraints' penalty is described with "J_{cst}" term of the cost function and is applied when

$$\frac{y_{NOx}(k+i|k)}{s_{y_{NOx}}} \leq \frac{y_{NOx,max}}{s_{y_{NOx}}} + \epsilon_k V_{NOx,max}, \tag{5.3}$$

$$\frac{y_{FOC}(k+i|k)}{s_{y_{FOC}}} \leq \frac{y_{FOC,max}}{s_{y_{FOC}}} + \epsilon_k V_{FOC,max}, \tag{5.4}$$

$$i = 1 : p$$

where

- $y_{j,max}$ - Upper bounds for jth plant output
- ϵ_k - Scalar QP slack variable used for constraint softening.
- $V_{j,max}$ - Dimensionless controller constants for jth plant output used for constraint softening and expresses the softened limits of the specific system's variable.

MPC 900 controls the λ value and also tries to cope with NOx and FOC upper limits. In MPC 901 the "J_y" term of the cost function becomes zero by setting $w_\lambda = 0$, so the controller acts only when NOx limit is exceeded or is predicted that the limit will be exceeded. MPC 902 in addition to MPC 901 has limitation also for the FOC value. When the output values are within the desired limits, setting $w_u > 0$ forces the controller to provide a zero command to the electric motor actuator as a result of the cost optimization.

		INPUTS / SCALE FACTOR	MEASURED DISTURBANCES / SCALE FACTOR				OUTPUTS / SCALE FACTOR				CONTROLLER PARAMETERS								CONSTRAINTS	
CONTROLLER NAME	MPC MODEL	FrInvCmd	MAP	SE	dSE	λ	λ	NOx	OPA	FOC	SAMPLE TIME (sec)	Np	Nu	INPUT WEIGHTS	INPUT RATE WEIGHT	OUTPUT WEIGHTS	ECR	OUTPUT ECR WEIGTHS (SOFTENING)	MANIPULATED VARIABLE CONSTRAINTS	OUTPUT CONSTRAINTS
MPC 101	m101	YES / 0,1					YES / 10				0.1	12	2	0	0.1	1	100000		[0 0,1]	-
MPC 200	m202	YES / 0,1	YES / 0,2	YES / 300			YES / 10				0.1	20	5	0	0.1	1	100000		[0 0,1]	-
MPC 201	m202	YES / 0,1	YES / 0,2	YES / 300			YES / 10				0.1	12	4	0	0.1	1	100000		[0 0,1]	-
MPC202	m202	YES / 0,1	YES / 0,2	YES / 300			YES / 10				0.1	20	5				100000		[0 0,1]	-
MPC 203	m202	YES / 0,1	YES / 0,2	YES / 300			YES / 10				0.1	12	2	0	0.1	1	100000		[0 0,1]	-
MPC 301	m301	YES / 0,1			YES / 300		YES / 10				0.1	12	2	0	0.1	1	100000		[0 0,1]	-
MPC 401	m303	YES / 0,1			YES / 300		YES / 10				0.1	12	2	0	0.1	1	100000		[0 0,1]	-
MPC 402	m303	YES / 0,1			YES / 300		YES / 10				0.1	12	2	0	0.6	10	100000		[0 0,1]	-
MPC 500	m204	YES / 0,1	YES / 0,2	YES / 300			YES / 10				0.1	12	2	0	0.1	1	100000		[0 0,1]	-
MPC 600	m207	YES / 0,1	YES / 0,2	YES / 300			YES / 10				0.1	12	2	0	0.1	1	100000		[0 0,1]	-
MPC 204	m202	YES / 0,1	YES / 0,2	YES / 100			YES / 5				0.1	12	2	0	0.1	1	100000		[0 0,1]	-
MPC 205	m202	YES / 0,1	YES / 0,2	YES /100			YES / 5				0.1	9	2	0	0.1	1	100000		[0 0,1]	-
MPC 403	m303	YES / 0,1			YES / 100		YES / 5				0.1	12	2	0	0.1	1	100000		[0 0,1]	-
MPC404	m303	YES / 0,1			YES / 100		YES / 5				0.1	20	4	0	0.1	1	100000		[0 0,1]	-
MPC 900	m303, mMAP, mNOx, m OPA, mCons	YES / 0,1	YES / 0,25	YES / 300	YES / 300	YES / 10	YES / 10	YES / 500	YES / 20	YES / 30	0.1	12	2	0	0.1	[0.4 0 0 0]	1000	NOx : 14 FOC: 10	[0 0.1]	NOx : [-Inf 400] FOC: [-Inf 14]
MPC 901	m303, mMAP, mNOx, m OPA, mCons	YES / 0,1	YES / 0,25	YES / 300	YES / 300	YES / 10	YES / 10	YES / 500	YES / 20	YES / 30	0.1	20	1	0.002	0.1	[0 0 0 0]	1000	NOx : 14 FOC: 10	[0 0.1]	NOx : [-Inf 300] FOC: [-Inf 20]
MPC 902	m303, mMAP, mNOx, m OPA, mCons	YES / 0,1	YES / 0,25	YES / 300	YES / 300	YES / 10	YES / 10	YES / 500	YES / 20	YES / 30	0.1	20	1	0.002	0.1	[0 0 0 0]	1000	NOx : 14 FOC: 10	[0 0.1]	NOx : [-Inf 350] FOC: [-Inf 12]

Figure 5.1: Summary of the designed MPC Controllers

5.2 Performance Evaluation of the Controllers

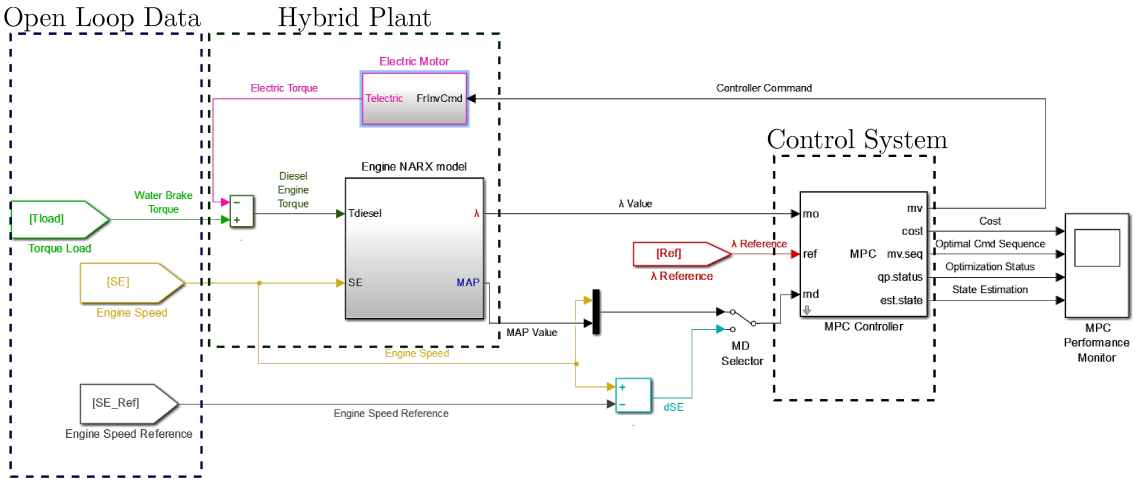

Figure 5.2: Simulation Set-up

Performance evaluation of the designed MPC controllers was firstly conducted through step response simulation against each controller's internal model, as shown in Fig. 5.3 ,in order to fine tune its parameters and ensure the system stability. After that, the controllers were evaluated through simulation using open-loop experimental data and the non-linear diesel engine model (Eq. (3.23)), using also the electric motor system model (Eq. (3.24)) and the total power split relation (Eq. (3.25)). The Simulation set-up in the Simulink platform is demonstrated in Fig. 5.2.

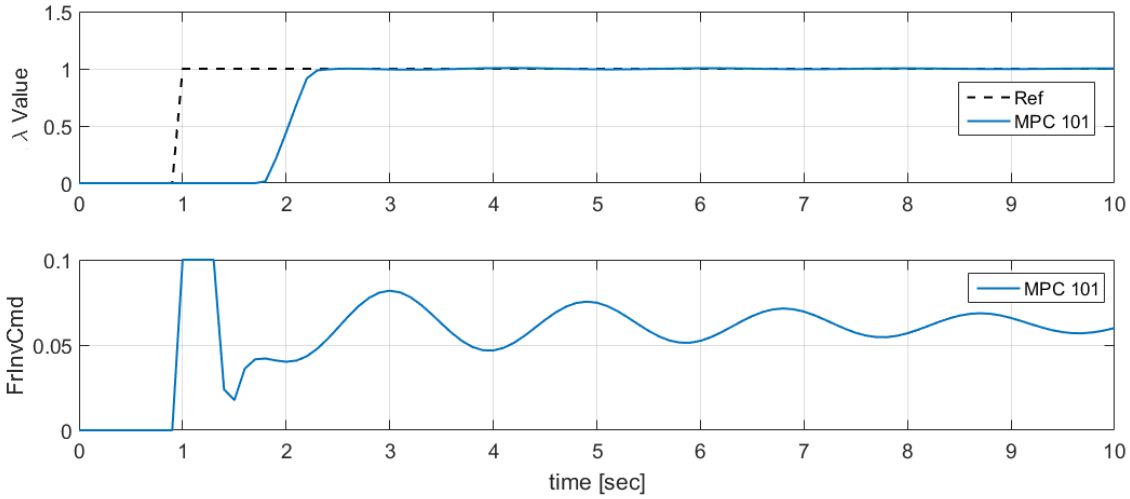

Figure 5.3: MPC 101 step response against its internal model

During simulation the controller's performance was evaluated against tasks like reference tracking, disturbance rejection and keeping the system within desired limits, which will also be applied in the real environment. The simulation results of MPC 201 are shown in Fig. 5.4 , together with their cost function value over the prediction horizon at each control interval. In Fig. 5.5 shows in detail the estimation of MPC 201 for the optimal sequence of control command over the prediction horizon at each control interval "$k(t)$", according to the output trajectory value estimation and the estimated control strategy.

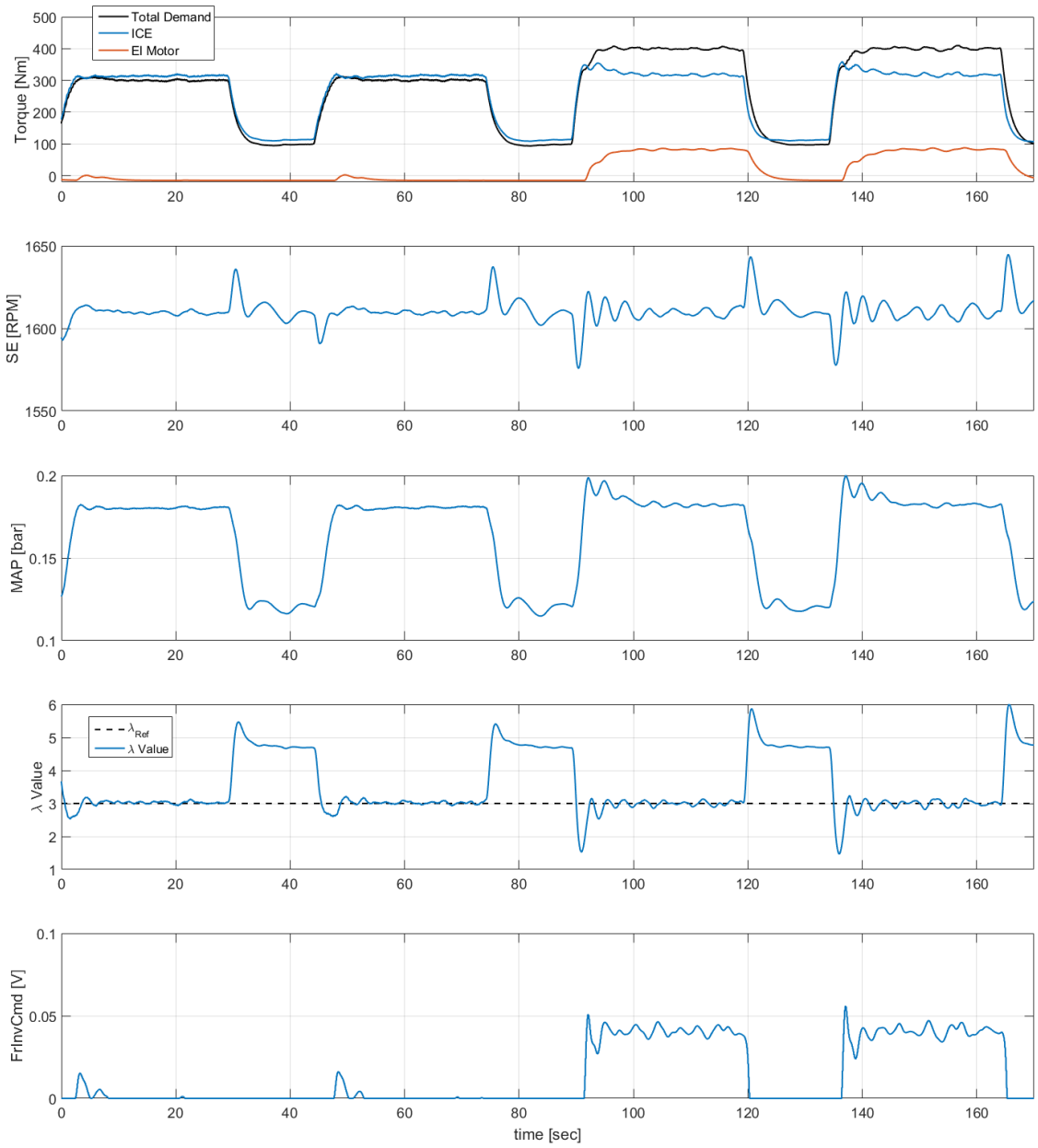

Figure 5.4: MPC 201 Simulation Results

Finally, the assessment of MPC 401 performance during simulation in relation to experimental results is shown in Fig. 5.6.

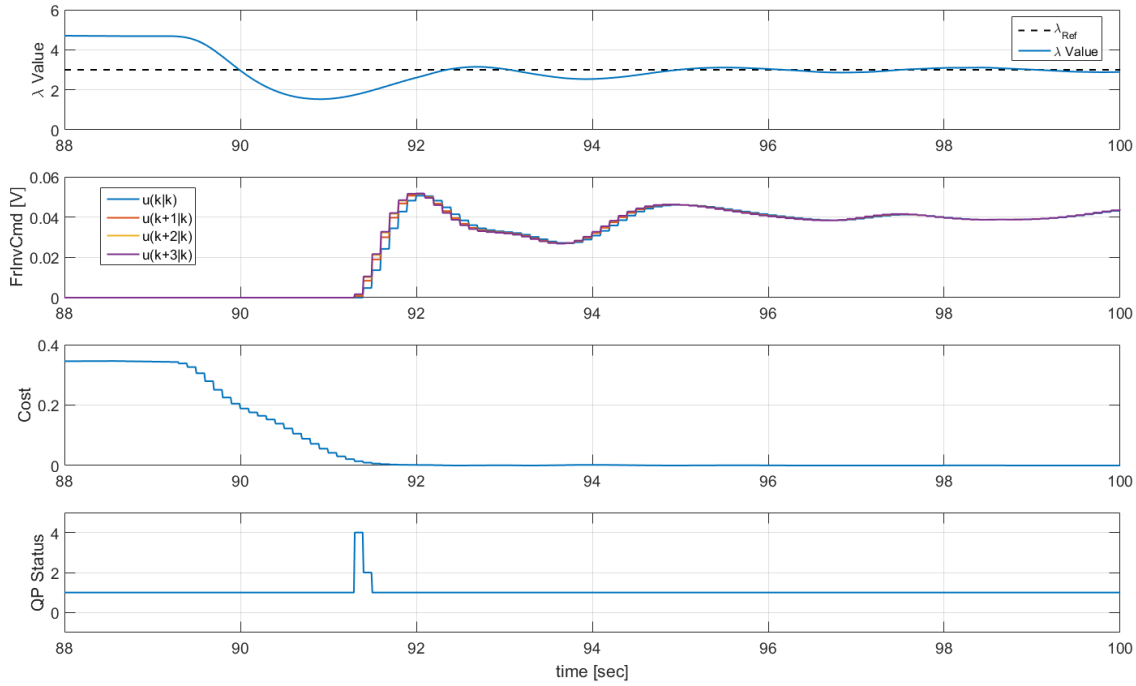

Figure 5.5: Detail of MPC 201 Simulation Cost and estimation of the Optimal Control Command Sequence at each control interval point "$k(t)$"

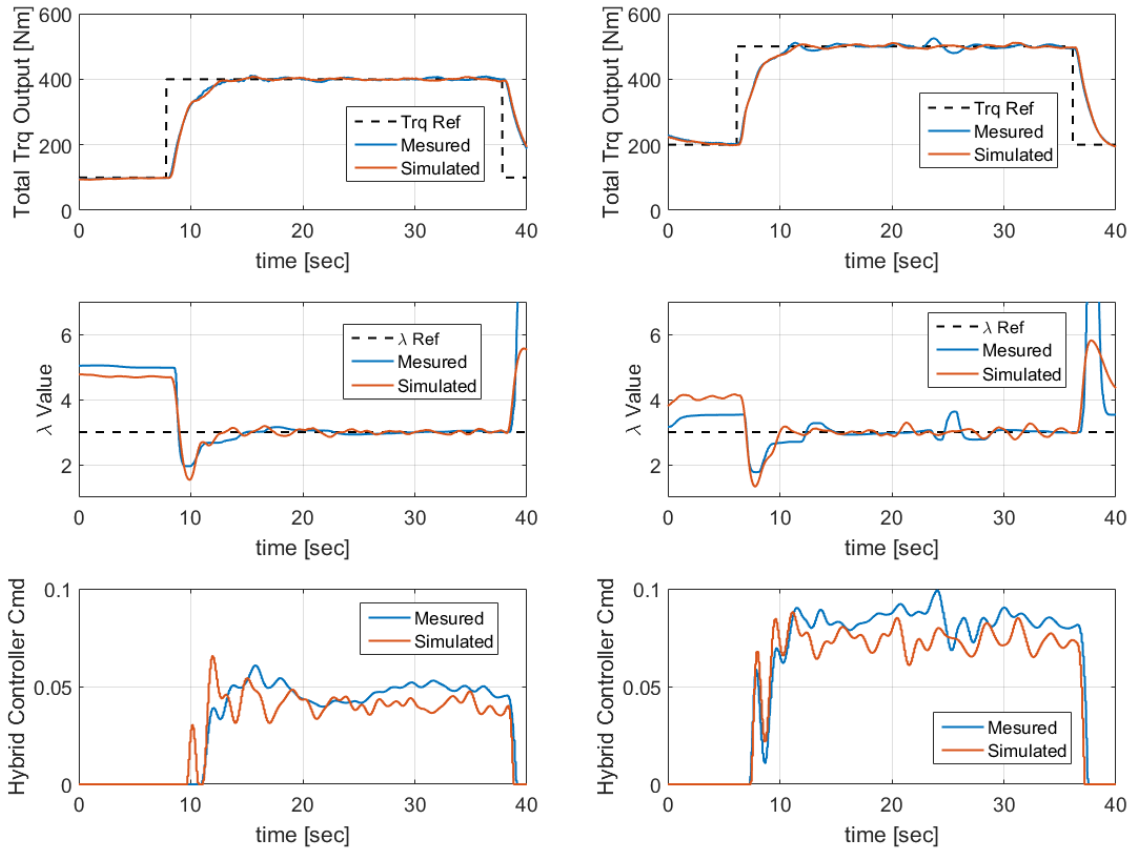

Figure 5.6: MPC 401 Simulation and Experiment comparison

Chapter 6

Experimental Results

Various experiments were conducted on the hybrid propulsion powertrain of LME, in order to evaluate the performance of the various MPC controllers. Experimental procedure included alternating loading and propeller loading. Total torque demand was initially set at the nominal operation point of the MPC internal models, but alternating loading was also tested at various load profiles of different bias and amplitude. The controller response was evaluated against static and dynamic reference tracking and its efficiency in operating the plant within the desired limits during closed loop control. Finally, torque demand and engine speed was changing simultaneously, according to the propeller loading law.

The index of MPC experiments is demonstrated, in tabular form, in Fig. 6.1. The most important experimental results, highlighted with light orange color, are demonstrated comparatively to each other in the following fields.

ALTERNATING LOAD					
STATIC λ REFERENCE					
CONTROLLER NAME	ENGINE SPEED [rpm]	λ REF	λ MIN	LOAD [Nm]	DURATION [sec]
MPC 101	1600	3	-	100-300 & 100-400 & 300-500	250
MPC 101	1600	3	-	200-500	90
MPC 200	1600	3	-	100-300 & 100-400	270
MPC 201	1600	3	-	100-300 & 100-400	270
MPC 201	1600	3	-	200-500	90
MPC 202	1600	3	-	100-300 & 100-400	270
MPC 203	1600	3	-	100-300 & 100-400 & 300-500	250
MPC 301	1600	3	-	100-300 & 100-400 & 300-500	250
MPC 301	1600	3	-	200-500	90
MPC 401	1600	3	-	100-300 & 100-400 & 300-500	250
MPC 401	1600	3	-	200-500	90
MPC 402	1600	3	-	200-500 & 300-500	220
DYNAMIC λ REFERENCE					
CONTROLLER NAME	ENGINE SPEED [rpm]	λ REF	λ MIN	LOAD [Nm]	DURATION [sec]
MPC 401	1750	dynamic	2.9	200-350 & 200-500	270
MPC 401	1600	dynamic	2.9	200-350 & 200-500	270
MPC 401	1800	dynamic	3	200-350 & 200-500	270
MPC 401	1800	dynamic	-	200-350 & 200-500	270
MPC 900	1600	dynamic	3	200-350 & 200-500 & 300-500	450
MPC 901	1600	N/A	-	100-400 & 200-500	250
MPC 902	1600	N/A	-	100-400 & 200-500	250
PROPELLER LOAD					
CONTROLLER NAME	ENGINE SPEED [rpm]	λ REF	λ MIN	LOAD [Nm]	DURATION [sec]
MPC 101p		dynamic	-	PROPELLER LOAD	260
MPC 203p		dynamic	-	PROPELLER LOAD	260
MPC 401p		dynamic	-	PROPELLER LOAD	260
MPC 900p		dynamic	-	PROPELLER LOAD	260

Figure 6.1: Index of MPC experiments

6.1 Alternating Load

The first set of experiments resembles a generator on-board a ship, where the engine operates at constant speed (1600 RPM) and with alternating load applied as step loading with various load demands. In this case the controller receives a static lambda reference value. Also the tracking of dynamic λ reference has been investigated. The dynamic reference λ values are derived from static look-up tables (maps) that utilize the measured intake manifold pressure and engine speed. These maps were created from experimental data of the ICE, under a wide range of operation [13]. In Fig. 6.2 the set-up of the MPC 401 case with dynamic reference is represented, which also shows an aspect of the experimental set-up on HIPPO 1 test-bed.

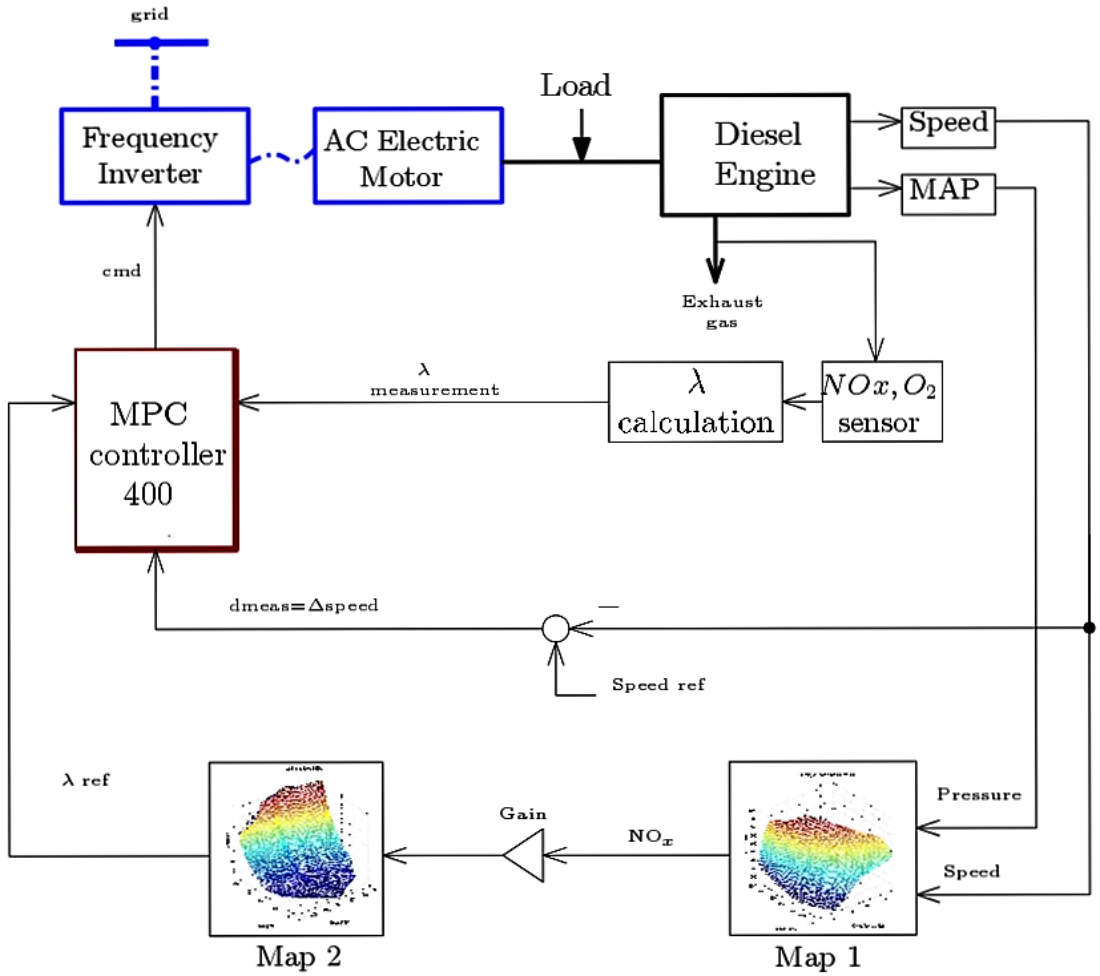

Figure 6.2: Block diagram of MPC 401 experiment set-up.

6.1.1 λ Control with Static Reference

100-300 and 100-400 Nm Step Loading

In Fig. 6.3, 6.4 and 6.5 is demonstrated the response of controllers MPC 101 and MPC 201 in relation to the conventional powertrain, re. without a hybrid setup. MPC 101 uses the SISO model in order to control λ value, while MPC 201 utilizes MAP and SE in order to estimate output trajectory value over the finite prediction horizon and act accordingly. As can be seen from Fig. 6.3, as soon as there is a rising edge in the applied

total torque the lambda drops rapidly and an error is created between the measured and reference lambda values. The controller is engaged and the EM produces torque which is added to the torque produced by the ICE in order to meet the total torque demand. MPC 101 engages the EM earlier than MPC 201, which leads to an overshoot of λ value on higher load amplitude, in contrast to the MPC 201, in which the modeled dynamics of the λ transient behavior lead to a smoother output profile.

In Fig. 6.4 the electric torque that is produced by the EM is demonstrated, where can be seen that in MPC 201 set-up λ value is less oscillating, as well as the other system outputs. Boost pressure drops as soon as the EM is engaged, as the lower torque portion which is undertaken by the ICE leads to lower exhaust gases power on the turbine side of the TC, leading to lower inlet pressure.

In Fig. 6.5, it can be noted that with the hybrid setup, both controllers offer the same gains in terms of the NOx content, which is around 25 % less during steady state than the conventional system in higher load. Exhaust gases' opacity and fuel consumption are also reduced at the same percentage levels as NOx emissions. Thus during the first spike of the transient loading, MPC 201 has an even better performance, managing to reduce NOx and opacity emissions at about 10%.

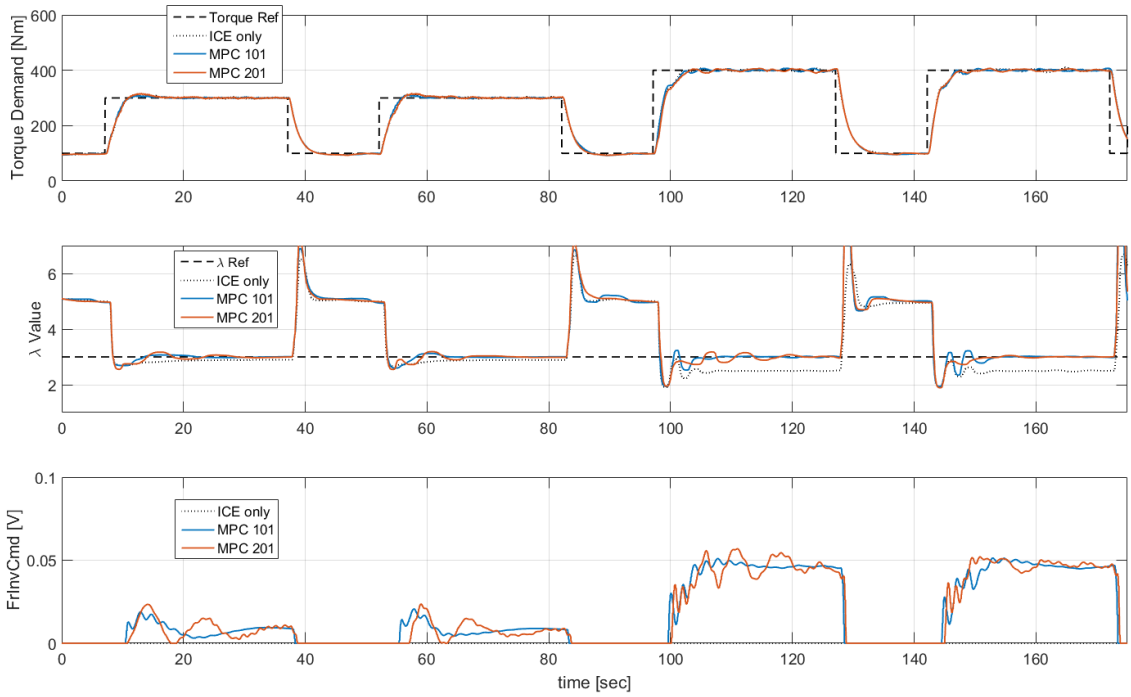

Figure 6.3: Measured total torque output, corresponding lambda values and hybrid controller command to the EM.

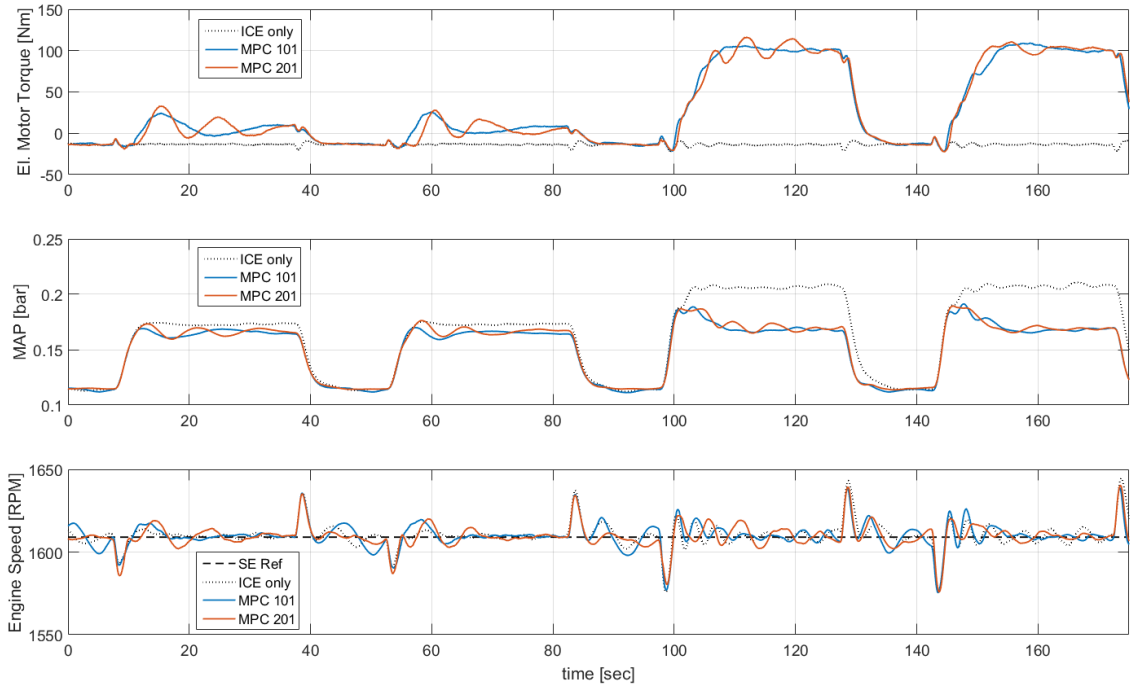

Figure 6.4: Electric torque, inlet pressure and engine speed.

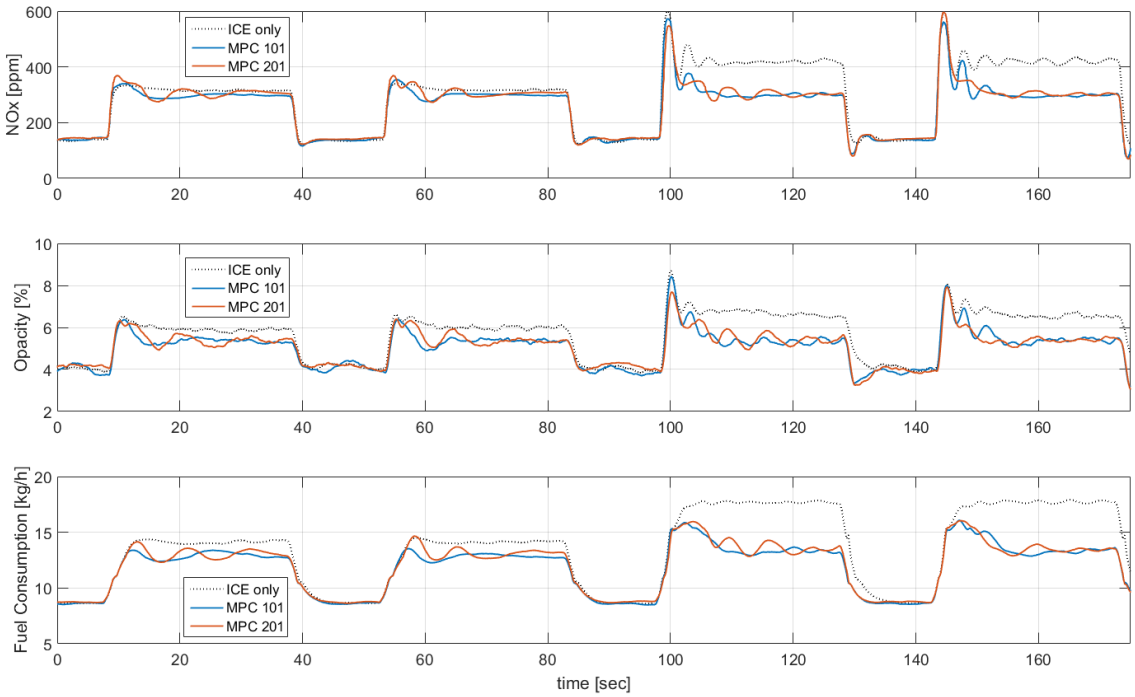

Figure 6.5: Effect of the hybrid powetrain on exhaust NO_x, opacity and fuel consumption, during step loading with static λ reference.

In Fig. 6.6, 6.7 and 6.8, the response of MPC 401 is compared to the behavior of MPC 201. MPC 401 utilizes the difference between the engine's speed and the speed reference as disturbance to estimate the λ transient dynamics. It is obvious that MPC 401 has enhanced performance compared to MPC 201, and consequently to MPC 101, managing to copy in a better way with the oscillating dynamics of the plant on lower and higher loads, as shown in Fig. 6.6, straightening emphatically the λ value profile. As it is presented in Fig. 6.8, MPC 401 has a bigger influence (5%) than MPC 201 to NOx and Opacity reduction during the transient phenomenon and in steady state condition.

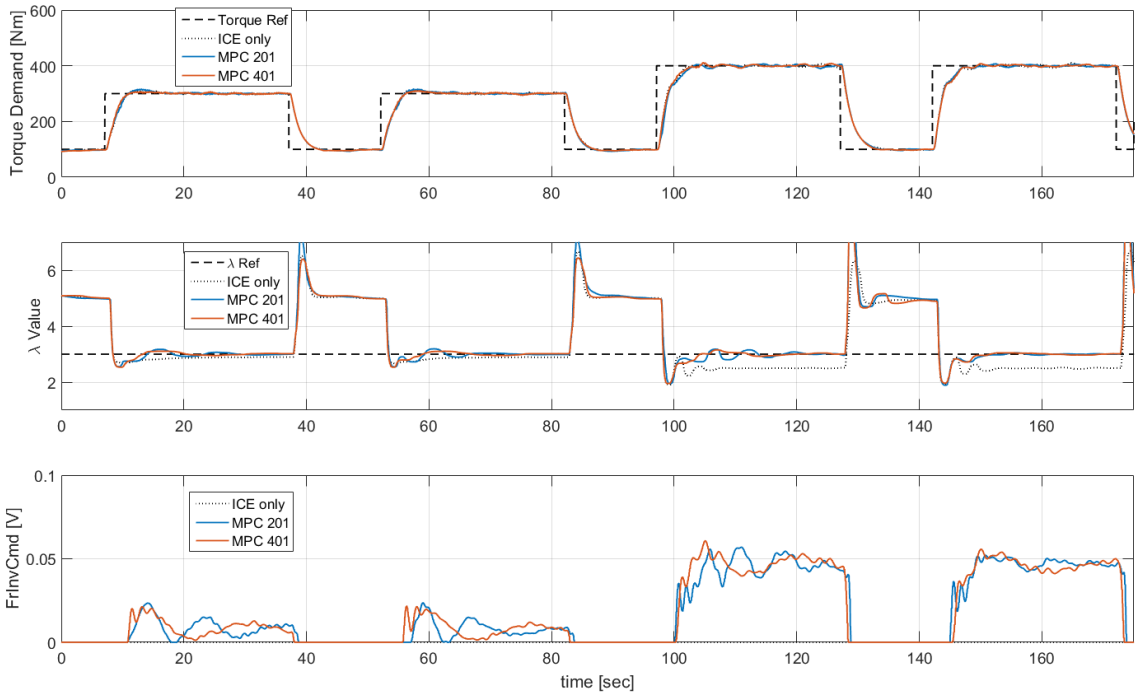

Figure 6.6: Measured total torque output, corresponding lambda values and hybrid controller command to the EM.

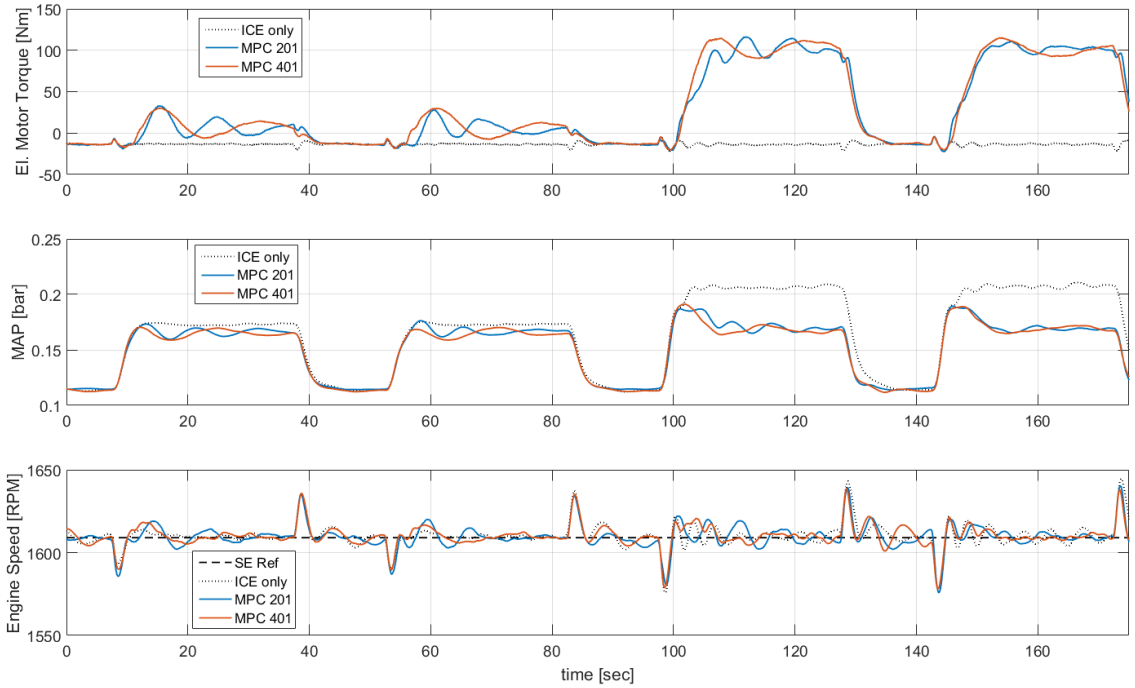

Figure 6.7: Electric torque, inlet pressure and engine speed.

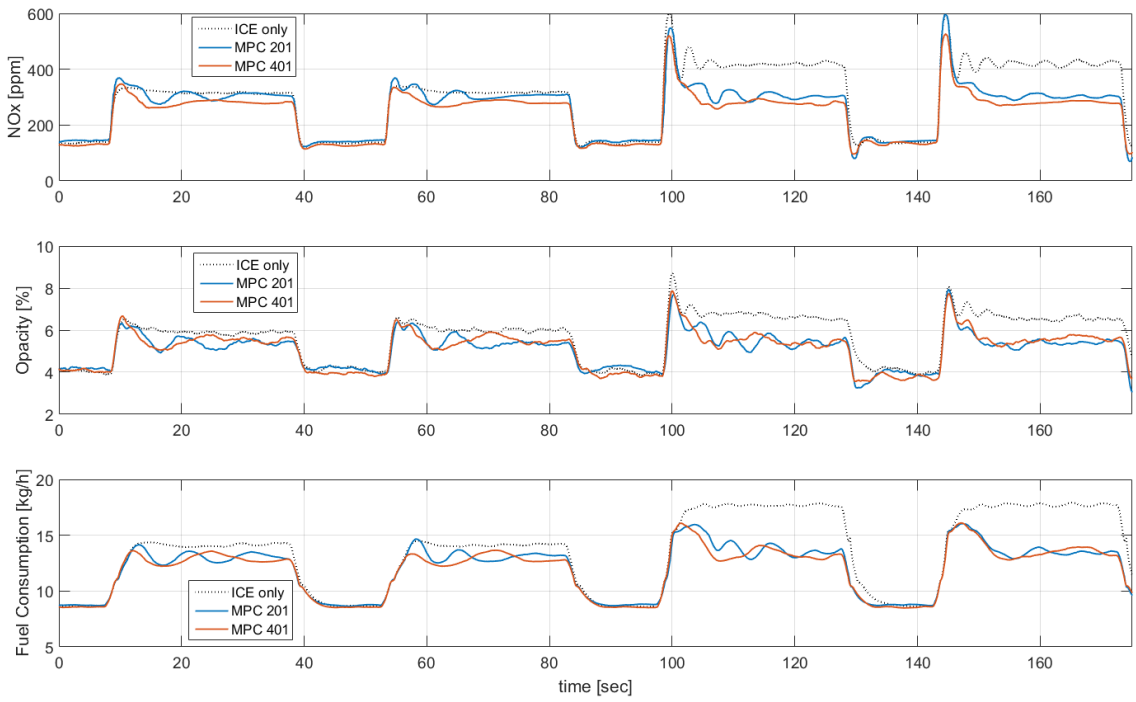

Figure 6.8: Effect of the hybrid powetrain on exhaust NO_x, opacity and fuel consumption, during step loading with static λ reference.

200-500 Nm Step Loading

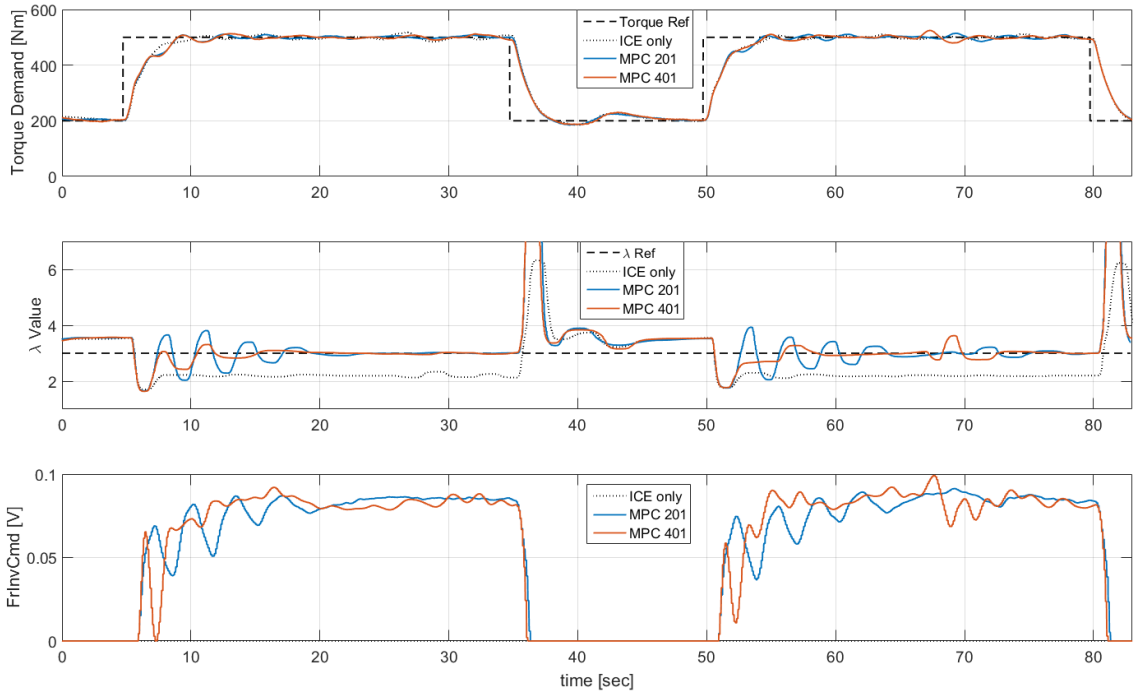

Figure 6.9: Measured total torque output, corresponding lambda values and hybrid controller command to the EM.

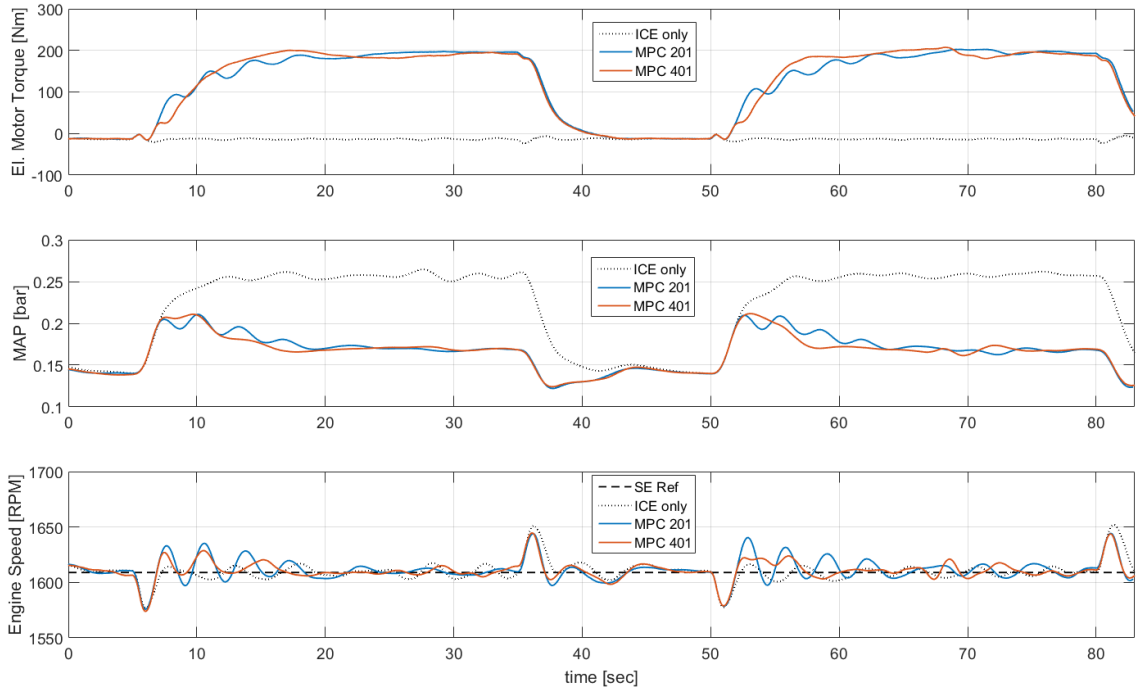

Figure 6.10: Electric torque, inlet pressure and engine speed.

In Fig. 6.9, 6.10 and 6.11, the performance of MPC 201 and MPC 401 is presented in alternating load between 200 and 500 Nm. Both controllers lead to the same steady state conditions, leading to 40% reduction of NOx, 30% reduction of fuel consumption

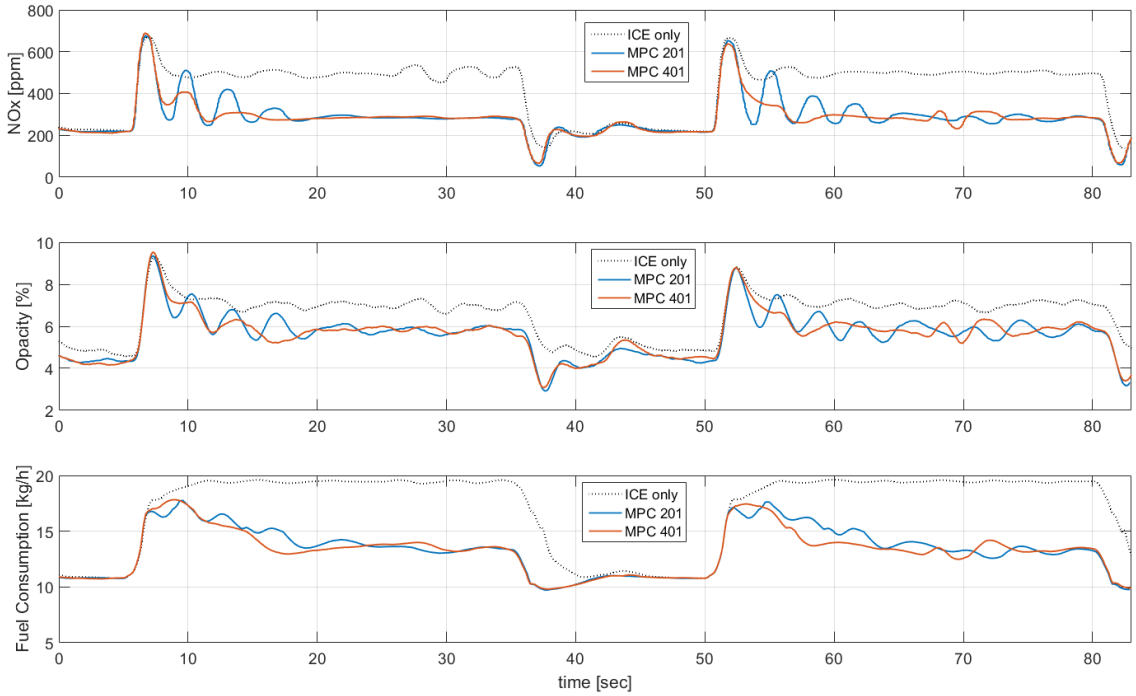

Figure 6.11: Effect of the hybrid powetrain on exhaust NO_x, opacity and fuel consumption, during step loading with static λ reference.

and 15% enhancement of the exhaust gas opacity, as shown in Fig. 6.11. However, during transients MPC 401 succeeds more stable λ value and other plant's measured outputs (MAP, EM torque). This happens because at 500 Nm, the plant operates out of the nominal operational area of the MPC 201 model (Fig. 3.2) and MAP, which is utilized as MD of MPC 201, has different gain compared to loads between 100-400 Nm at 1600 RPM. It is concluded that MPC 401 has a enhanced ability to reject disturbances compared to the other designs.

300-500 Nm Step Loading

In this scenario, where the ability of disturbance rejection of the MPC 101, 201, 401 controllers is evaluated, the EM is engaged during both higher and lower load, evaluating the performance of the hybrid plant in continuous operation. As presented in Fig. 6.12 and 6.13, as there is a rise of the load and a drop to the engine speed, MPC 201 and MPC 401 react before λ value drops, leading to slightly smaller drop of the λ value, while MPC 101 acts when λ has already dropped below its reference. After that, MPC 401, although it gives the highest command at the beginning, decreases the applied electric torque, as it estimates that the λ value is going to overshoot its reference and then reaches its steady state condition.

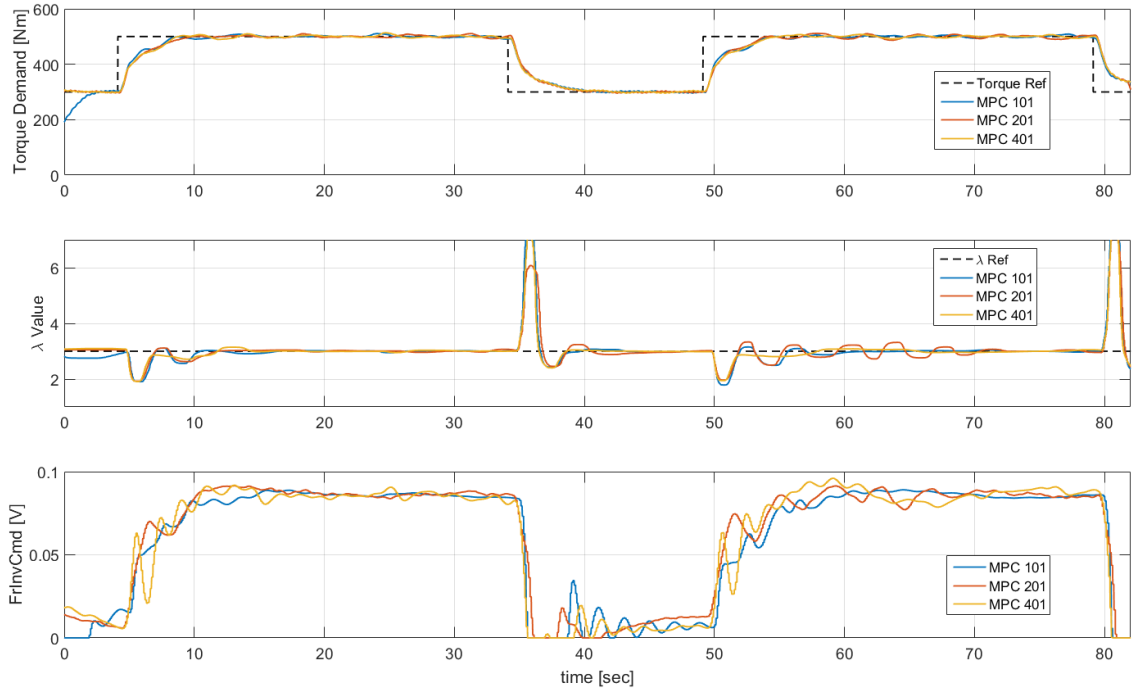

Figure 6.12: Measured total torque output, corresponding lambda values and hybrid controller command to the EM.

On the contrary, MPC 101 and 201 force λ value to overshoot, incommoding the plant to find its stability, which is shown also in the rest powertrain's measurements. MPC 401, stabilizes the λ value and then alleviates the error from the reference in contrast to the other controllers which have an oscillating behavior. In Fig. 6.14 it is noted that MPC 201 and MPC 401 lead to lower emissions $(10 - 15\%)$ during the first spike than MPC 101.

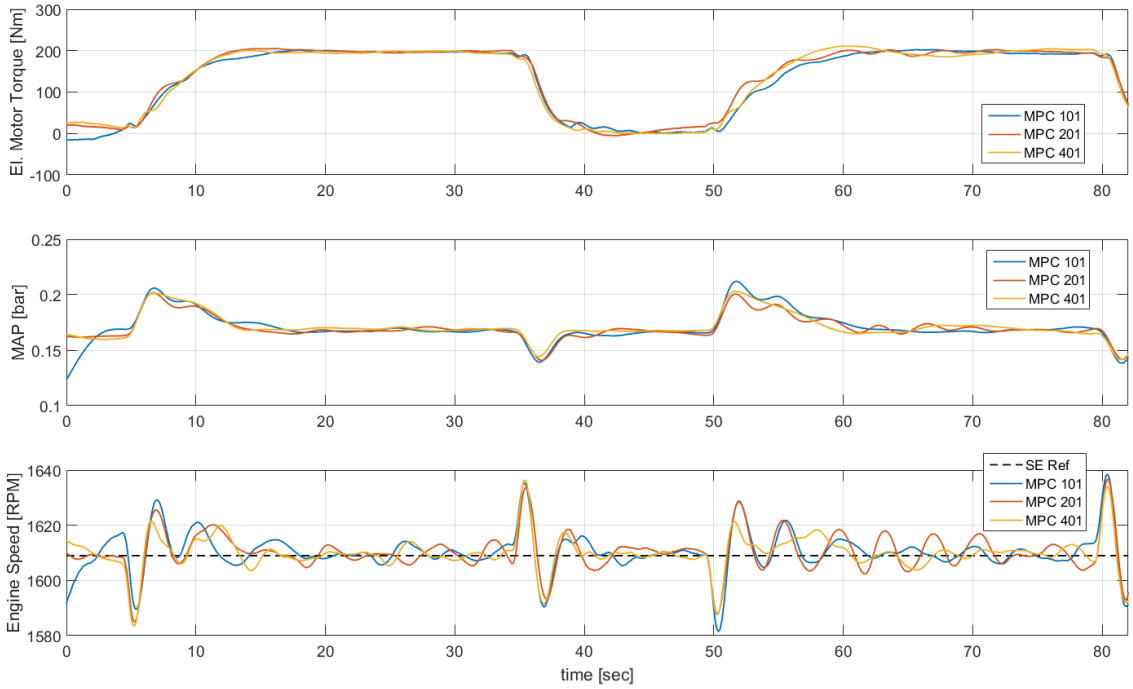

Figure 6.13: Electric torque, inlet pressure and engine speed.

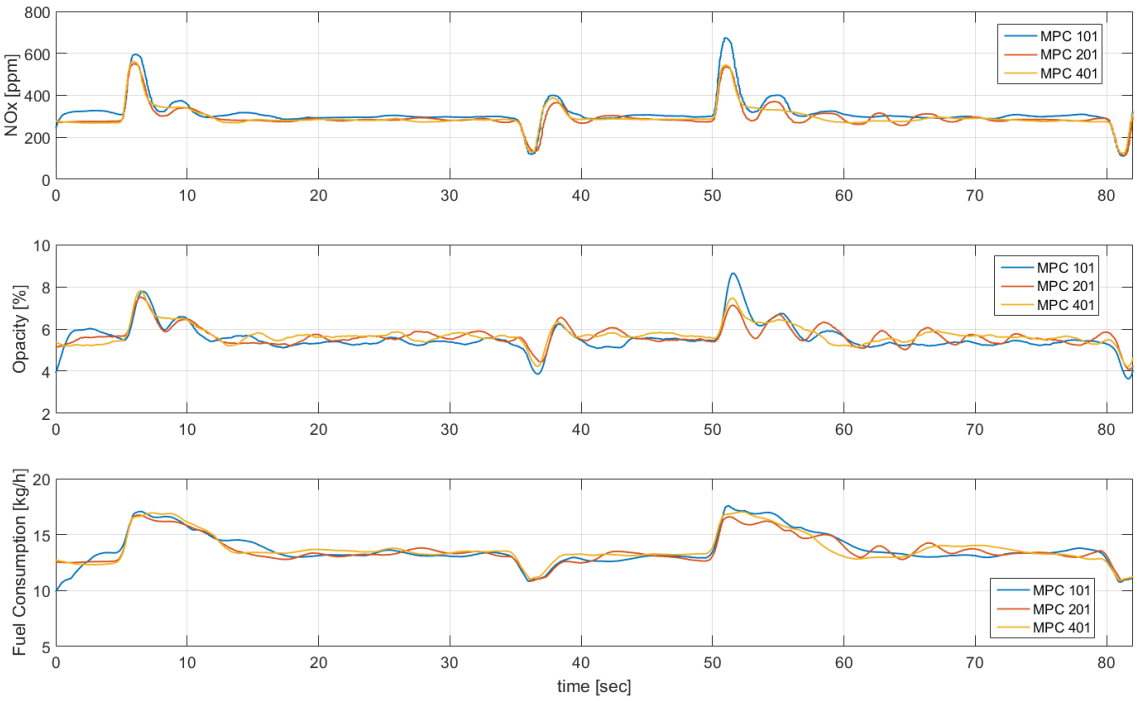

Figure 6.14: Effect of the hybrid powetrain on exhaust NO_x, opacity and fuel consumption, during step loading with static λ reference.

MPC comparison against PI Controller

In Fig. 6.15, 6.16 and 6.17, MPC 401, which is considered the most attractive and competitive design with MPC techniques, is compared to a PI controller with gains $K_p = 0.001$, $K_i = 0.002$. The PI controller was designed in LME in order to allow comparison of MPC performance against classic control solutions, as PID types of controller are considered an industry standard scheme.

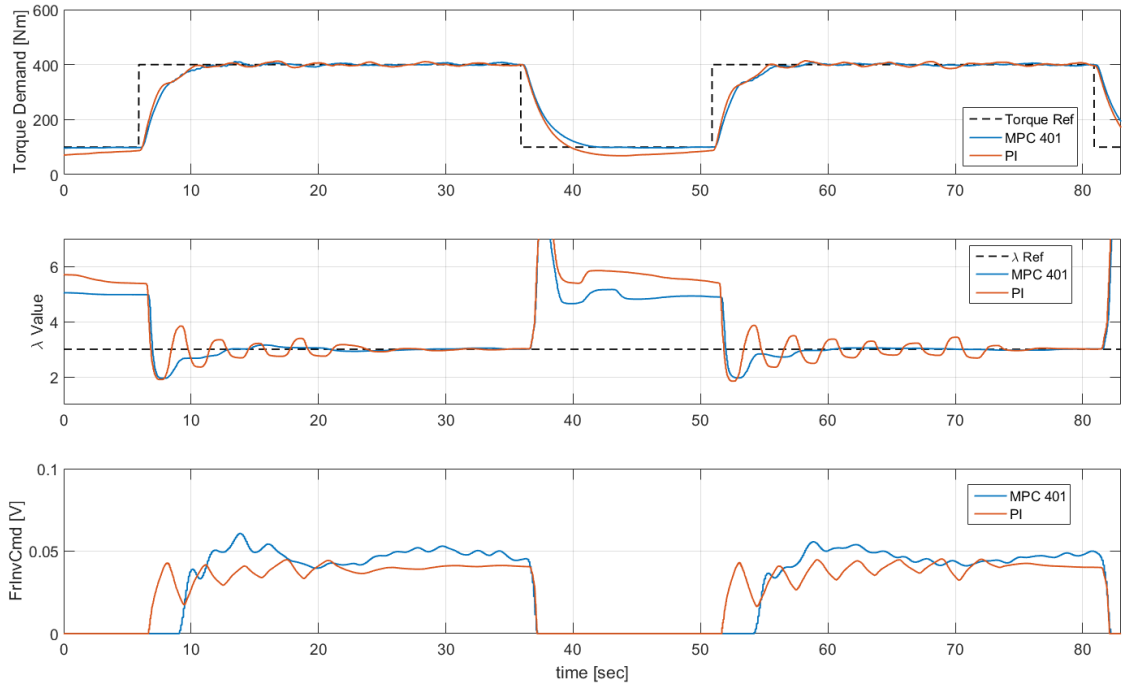

Figure 6.15: Measured total torque output, corresponding lambda values and hybrid controller command to the EM.

In addition to that, the PI controller was designed so that to provide the same dynamics with MPC at the engagement of the EM, as shown in the first subplot of Fig. 6.16. As shown in Fig. 6.15, PI reacts immediately as λ drops during step loading, while MPC 401 estimates that this is not the optimal solution, according to the controllers internal plant and the forthcoming speed disturbance. Consequently, acts at the critical point to stabilize the λ value. Consequently, MPC 401 leads to lower average emissions and fuel consumption, while PI control leads to an oscillating plant's behavior. It can be concluded that MPC is more capable of acting aggressively to a plant under control, while classic controllers tend to act rather smoother.

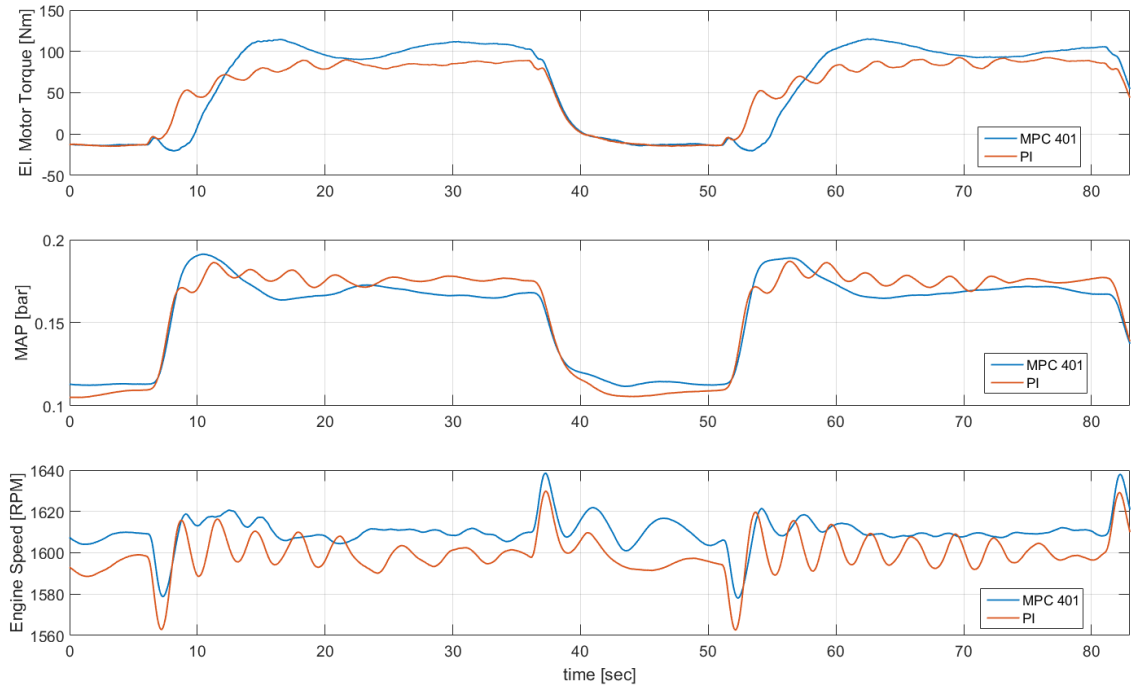

Figure 6.16: Electric torque, inlet pressure and engine speed.

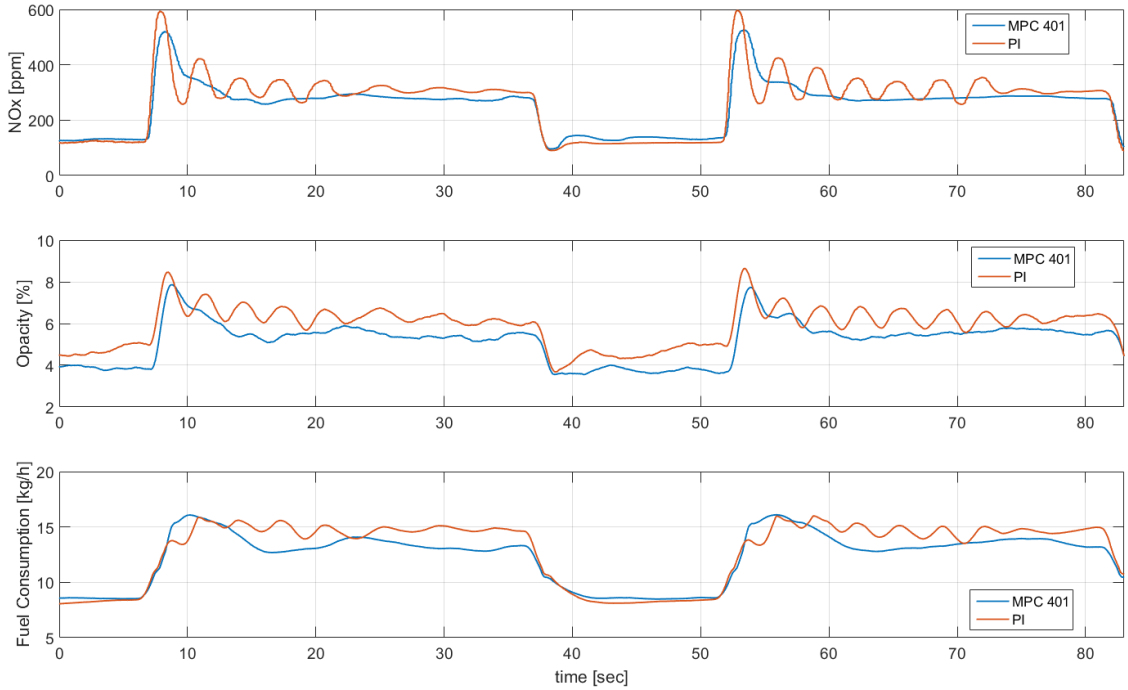

Figure 6.17: Effect of the hybrid powetrain on exhaust NO_x, opacity and fuel consumption, during step loading with static λ reference.

6.1.2 λ Control with Dynamic Reference

MPC is an attractive control method, which has the advantage to find the optimal control strategy over a finite prediction horizon according to a plant's current state. The challenge is to find the optimal reference tracking trajectories for the plant's control outputs [15]. Towards this direction are the look-up tables, created by LME, which provide a realistic λ reference point, according to the speed and boost pressure value of the engine [13]. In Fig. 6.18 a view of these maps is provided, where the output contour lines are shown.

As it can be noted from Fig. 6.19, in both reference scenarios the MPC controller has almost the same dynamic behavior, stabilizing the λ value after the first drop during loading. During the dynamic reference scenario, MPC 401 acts 0.3 sec. faster, as it is estimated that the control value will drop earlier under the reference trajectory. As far as the rest measurements of the powertrain are concerned, in the dynamic reference scenario are greater than in the static, because in steady state during loading the dynamic reference is lower than the static, as demonstrated in Fig. 6.20 and 6.21.

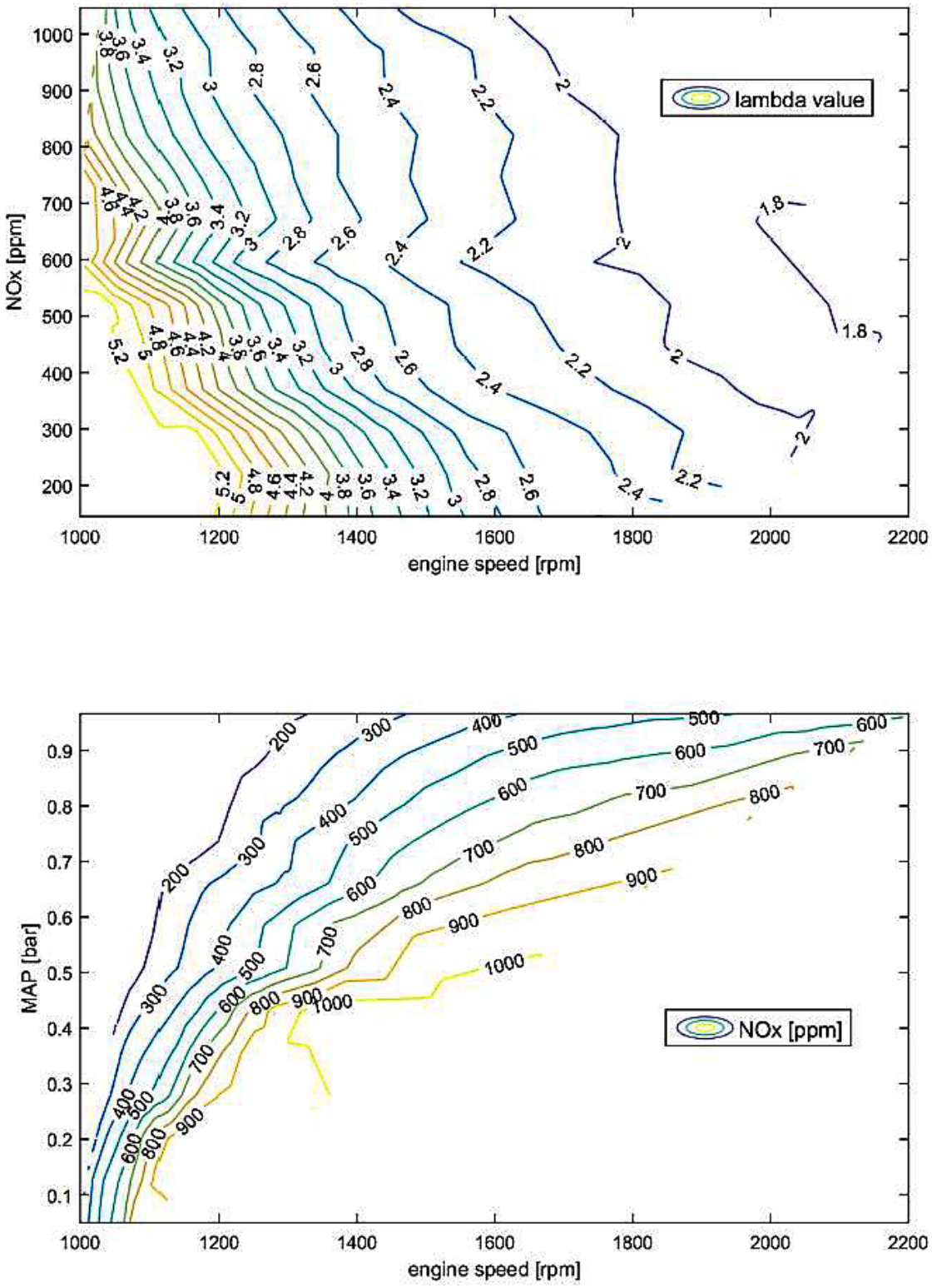

Figure 6.18: Look-up tables providing the dynamic λ reference according to engine speed, inlet pressure and NOx values.

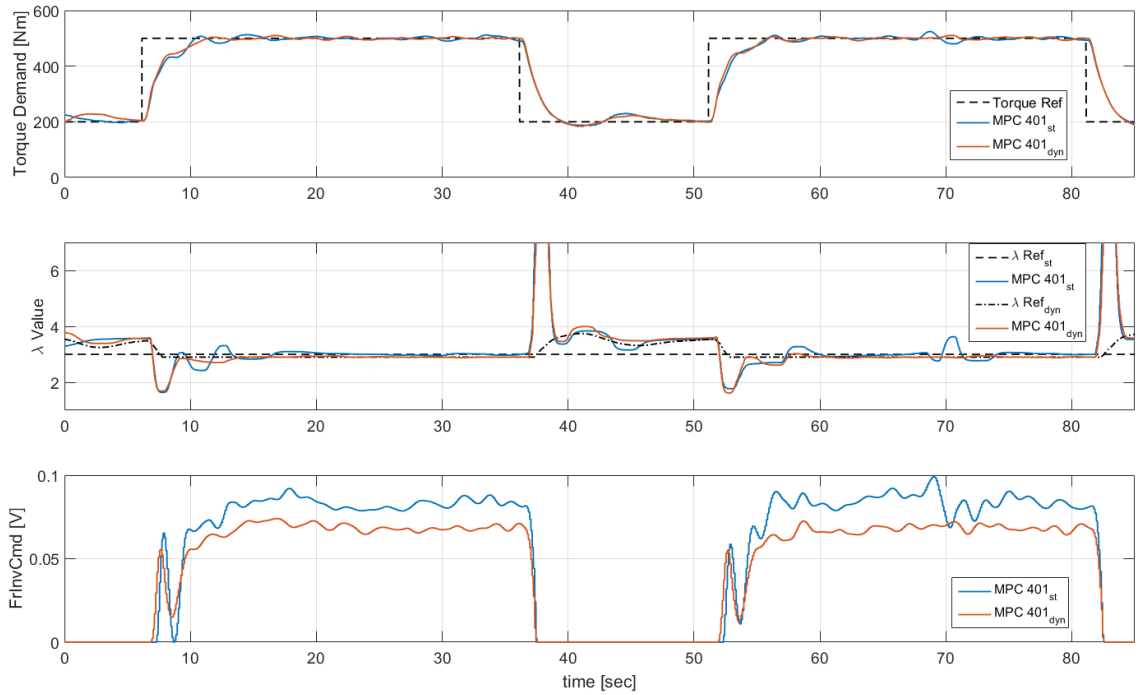

Figure 6.19: Measured total torque output, corresponding lambda values and hybrid controller command to the EM.

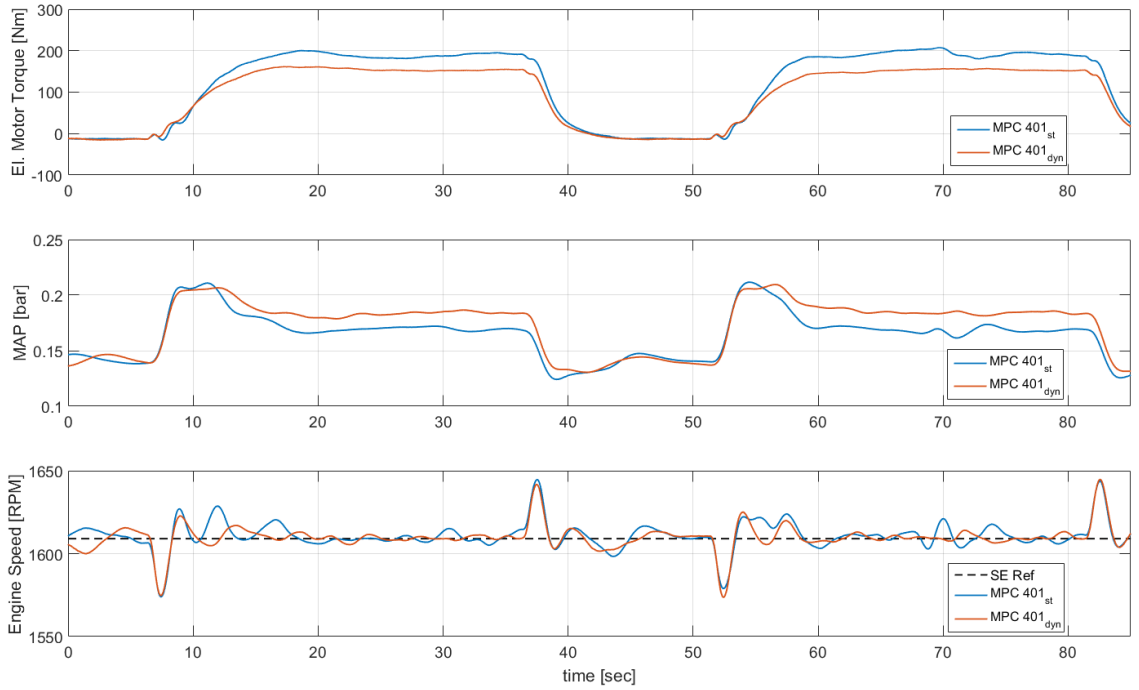

Figure 6.20: Electric torque, inlet pressure and engine speed.

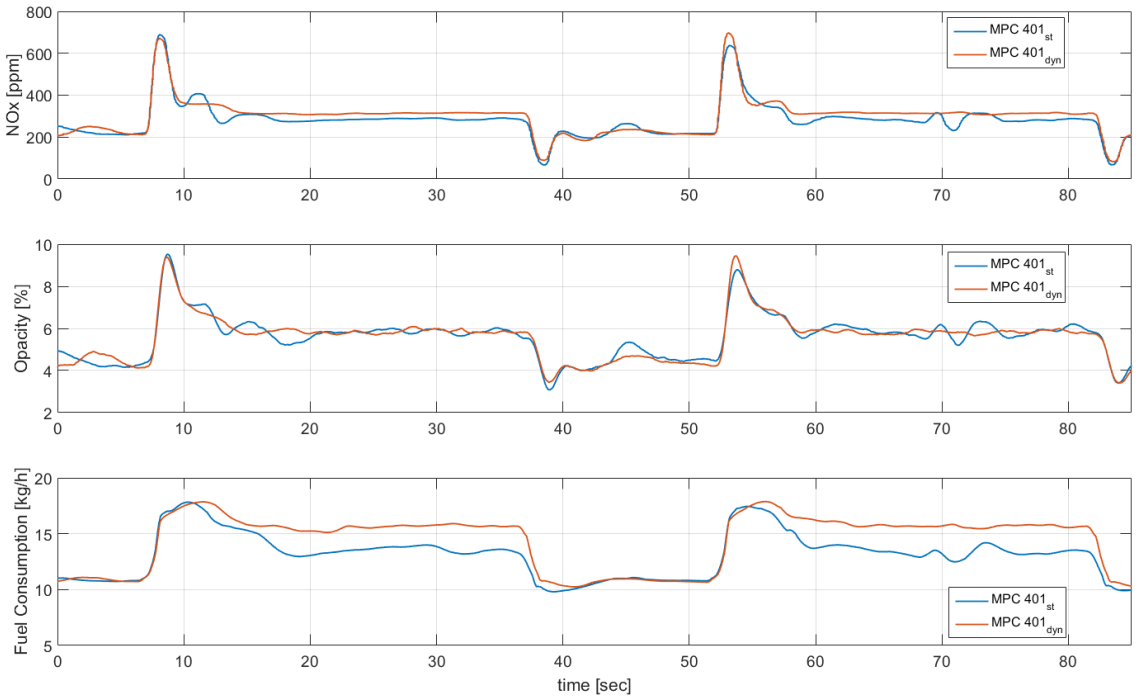

Figure 6.21: Effect of the hybrid powetrain on exhaust NO_x, opacity and fuel consumption, during step loading with static and dynamic λ reference.

6.1.3 λ Control with Dynamic Reference under Constraints in NOx and Fuel Consumption

MPC can handle plants that have operational limitations simultaneously with the reference tracking. In such case, NOx emissions and fuel consumption limits are posed at 400 ppm and 14 kg/h respectively as soft constraints and the behavior of MPC is evaluated. Fuel oil consumption is estimated through a look-up table which utilizes inlet pressure and engine speed. Soft constraints posing results to a cost addition when the system violates the limits, or the controller estimates that the limits are going to be violated within the prediction horizon. The block representation which shows the set-up of the constrained MPC 900 case is shown in fig. 6.22.

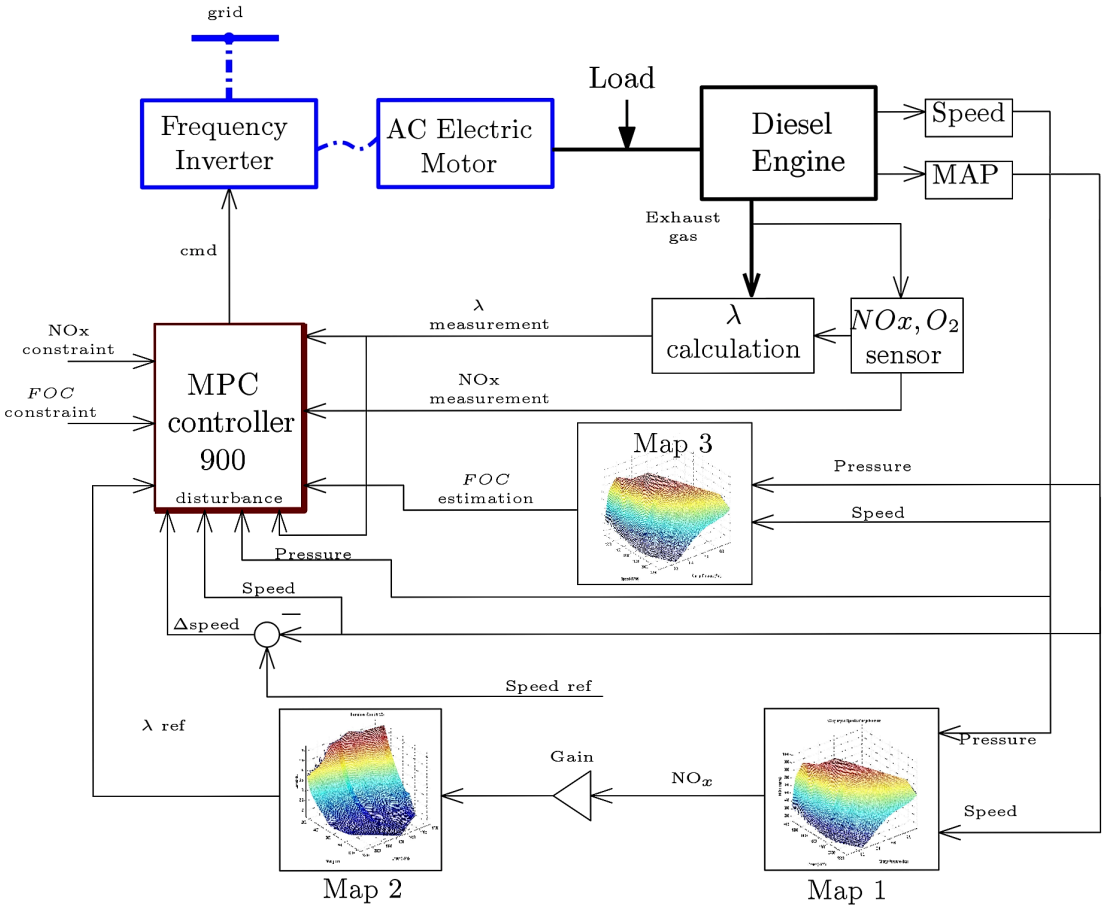

Figure 6.22: Block diagram of the constrained MPC 900 case set-up.

200-350 and 200-500 Nm Step Loading

As it can be noted in Fig. 6.23, 6.24, 6.25, when the load is applied both controllers act a few milliseconds before the drop of the λ value. MPC 401, that utilizes the speed error, engages a few milliseconds before λ drops, due to the fact that the speed of the ICE is dropping almost instantly when a load is applied. After the initial command spike, the controller can predict that this big of a command will lead to an overshoot, and the command value drops once again. It can also be seen that generally the 900 design produces higher command values than the 401, although the internal model of the plant used is the same in both, due to the addition of constraints in MPC 900 for NOx content and fuel consumption.

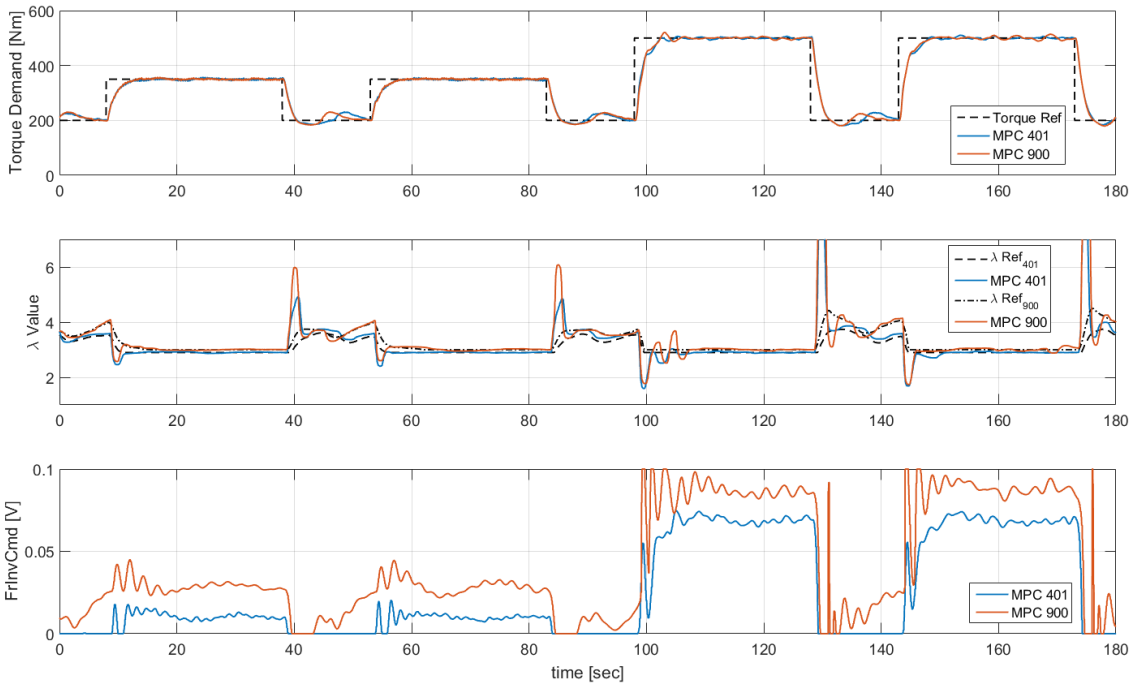

Figure 6.23: Measured total torque output, corresponding lambda values and hybrid controller command to the EM.

In Fig. 6.24, the torque output of the EM, the ICE intake manifold pressure and the rotational speed of the powertrain are presented. In the first subplot it can be seen that the shape of the EM torque is generally the same for both controller designs, with MPC 900 producing more torque due to the limitations it offers in exhaust emissions and fuel consumption. In the second subplot, it can be noted that MPC 900 leads to lower MAP values because in this setup the engine contributes less to the total power demand than in MPC 400 scenario.

In Fig. 6.25, it can be observed that both controllers offer the same gains in terms of NOx emissions and exhaust gas opacity. The first spike on the NOx content cannot be avoided due to the system dynamics. Thus MPC 900 shows a $\sim 10\%$ better performance during transients. In MPC 900 scenario, fuel consumption is being kept under the limits, resulting to more gain than in MPC 401 scenario, thus, leading to a greater electric power consumption.

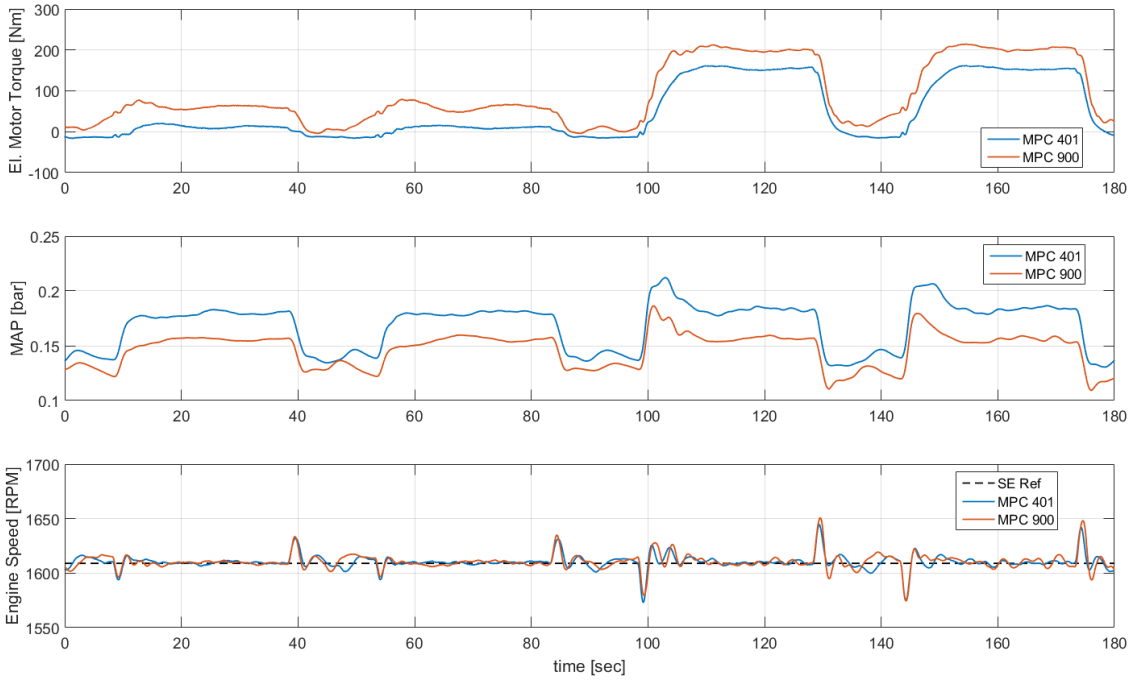

Figure 6.24: Electric torque, inlet pressure and engine speed.

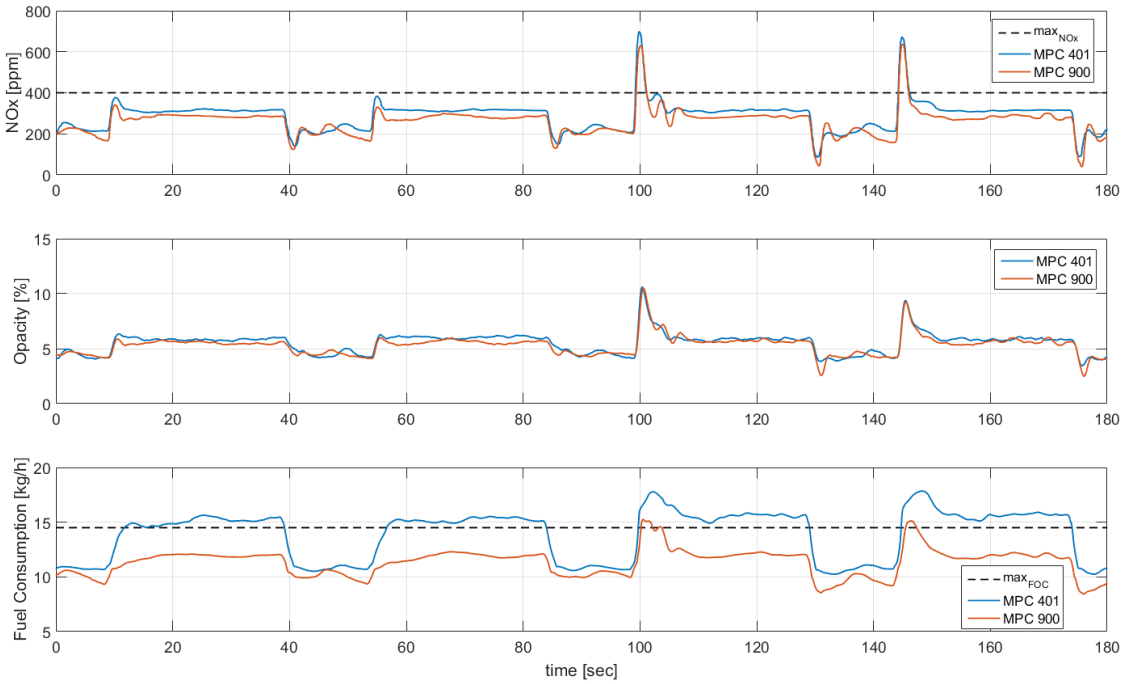

Figure 6.25: Effect of the hybrid powetrain on exhaust NO_x, opacity and fuel consumption, during step loading with dynamic λ reference.

200-500 Nm Step Loading compared to Static Reference without constraints

In Fig. 6.26, 6.27, 6.28, the constrained scenario with dynamic reference and operational limits is compared to the standard, unconstrained, static reference scenario. In general MPC 900 acts more emphatically on the plant in order to stay within the desired limits, managing to reach faster the λ reference and obtaining higher stability of the controlled object (λ value).

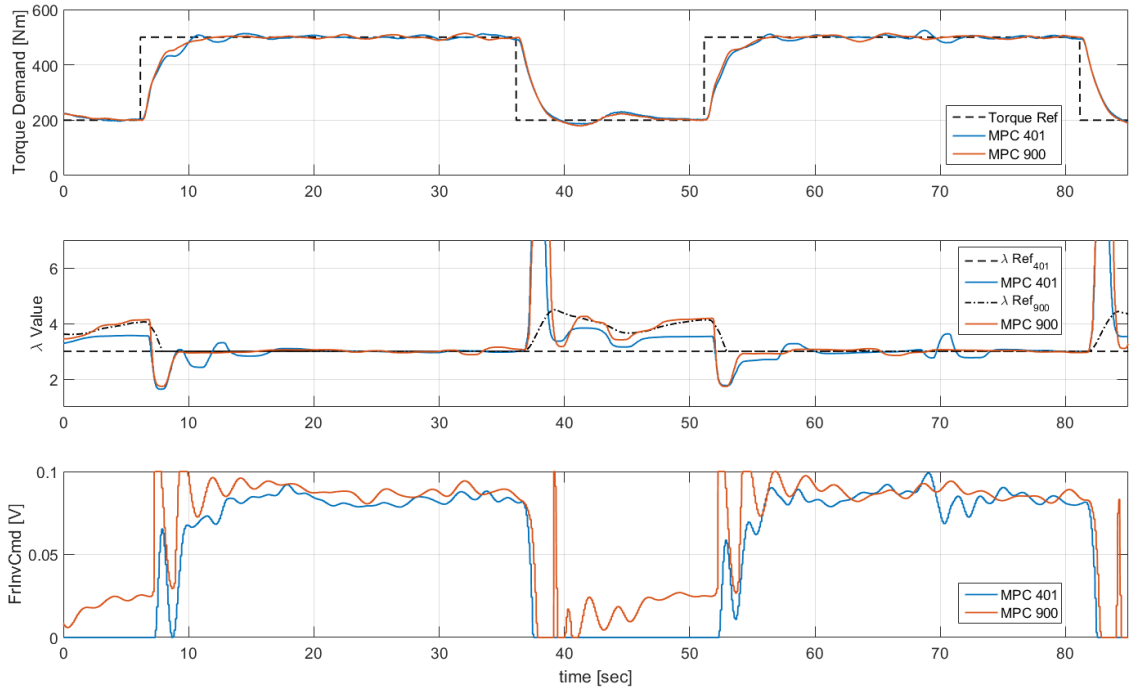

Figure 6.26: Measured total torque output, corresponding lambda values and hybrid controller command to the EM.

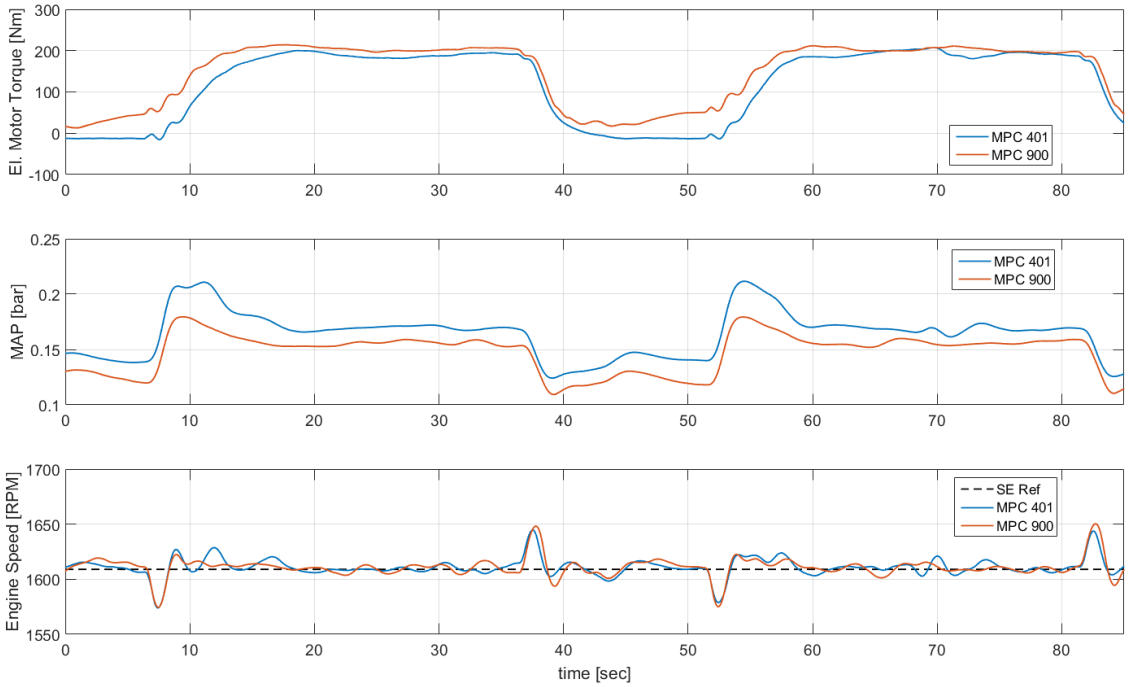

Figure 6.27: Electric torque, inlet pressure and engine speed.

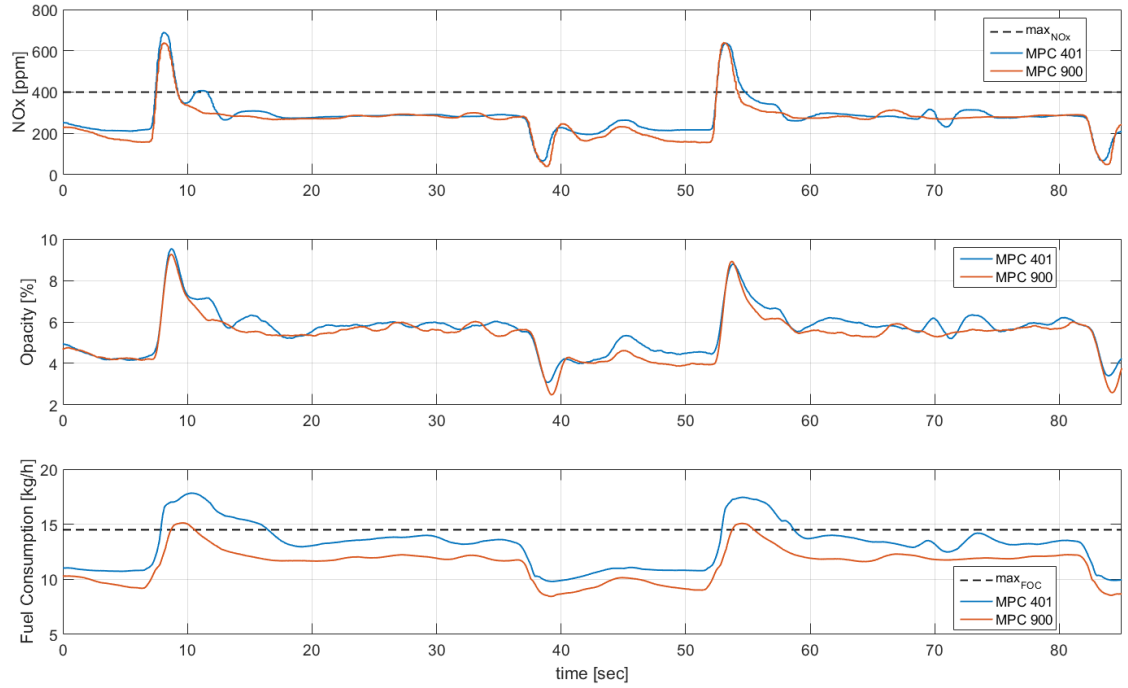

Figure 6.28: Effect of the hybrid powetrain on exhaust NO_x, opacity and fuel consumption, during step loading with dynamic λ reference compared to static reference.

6.1.4 Control under Constraints in NOx and Fuel Consumption without Reference Tracking

In this scenario, the controller tries not to exceed the posed limits regarding NOx emissions and fuel oil consumption without tracking any reference trajectory. MPC 901 set-up has only NOx content limitation at 300 ppm, while MPC 902 is constrained both in NOx emissions at 350 ppm and fuel oil consumption at 12 kg/h. Both controllers succeeded to operate the plant within the desirable limits. As the first spikes during loading can't be avoided, the controllers restore the plant's operation under the maximum limits, as shown in Fig. 6.31. In fig 6.29 can be noted that the controller has different dynamic when increasing the command in comparison to the points when the command is decreased. This is a result of the severe change of the cost function value above the limits, because as it is estimated that a constraint is going to be violated, a great cost is added, while under the limits the cost is depended on the command amplitude. As the controller command is penalized, the controller is forced to provide a zero command value, until a limit is exceeded again, where the constraints violation cost is added again to the cost function, forcing the controller to increase the command.

It can be concluded that this idea can work using only one individual output constraint, as well as in the combination of more operational constraints.

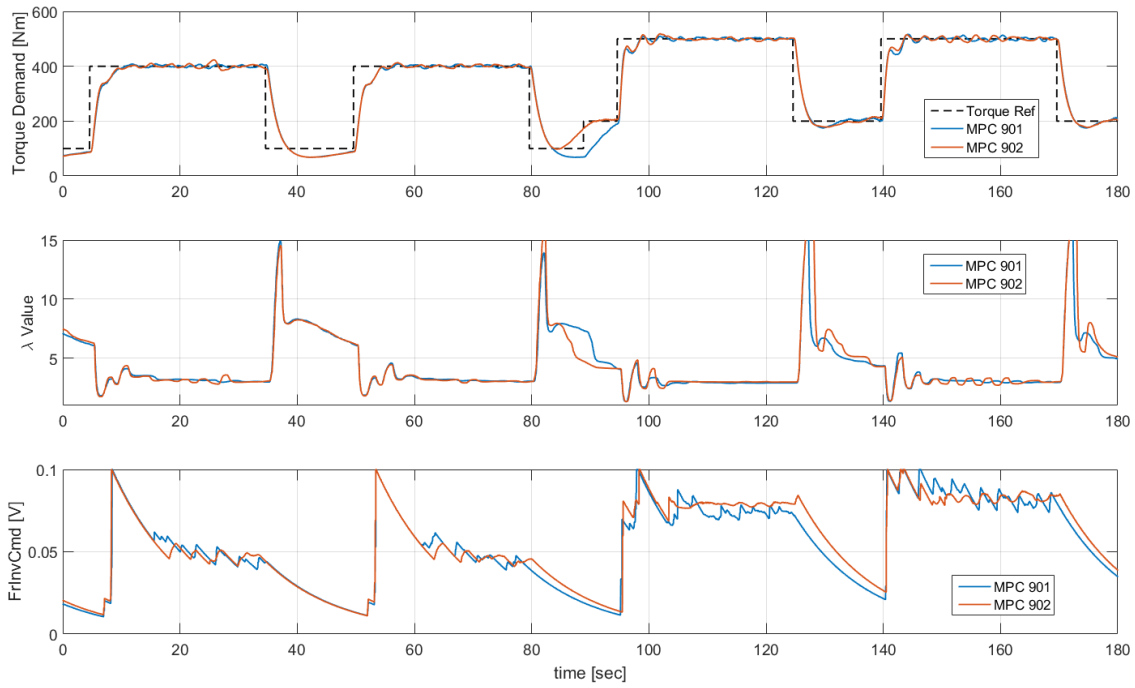

Figure 6.29: Measured total torque output, corresponding lambda values and hybrid controller command to the EM.

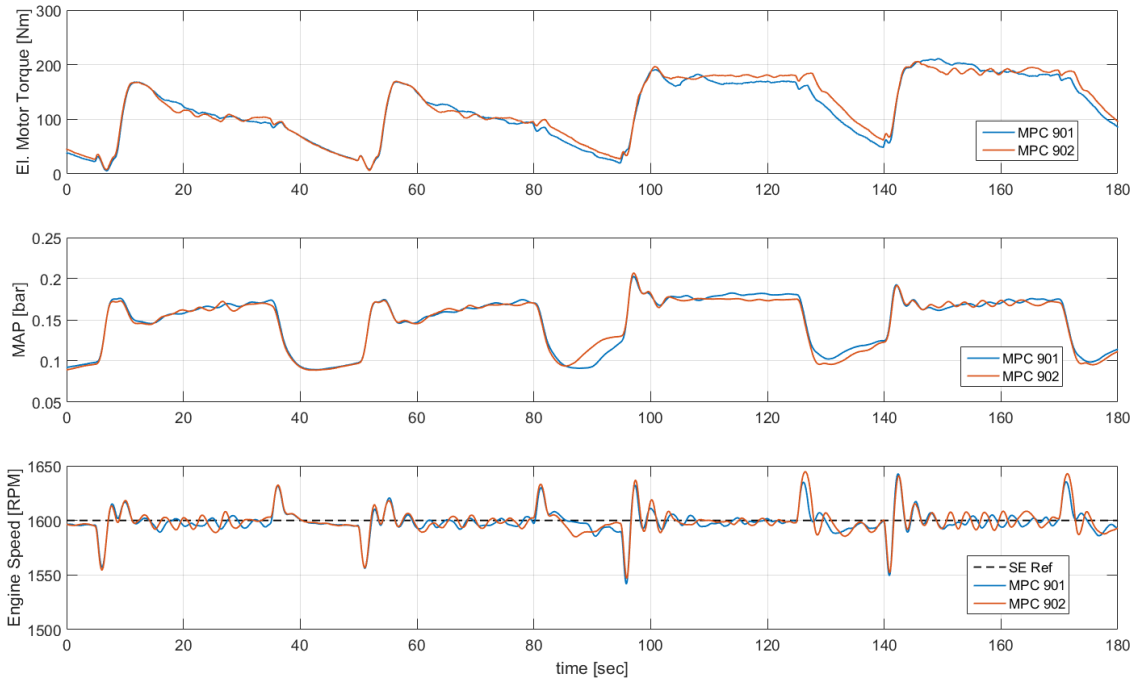

Figure 6.30: Electric torque, inlet pressure and engine speed.

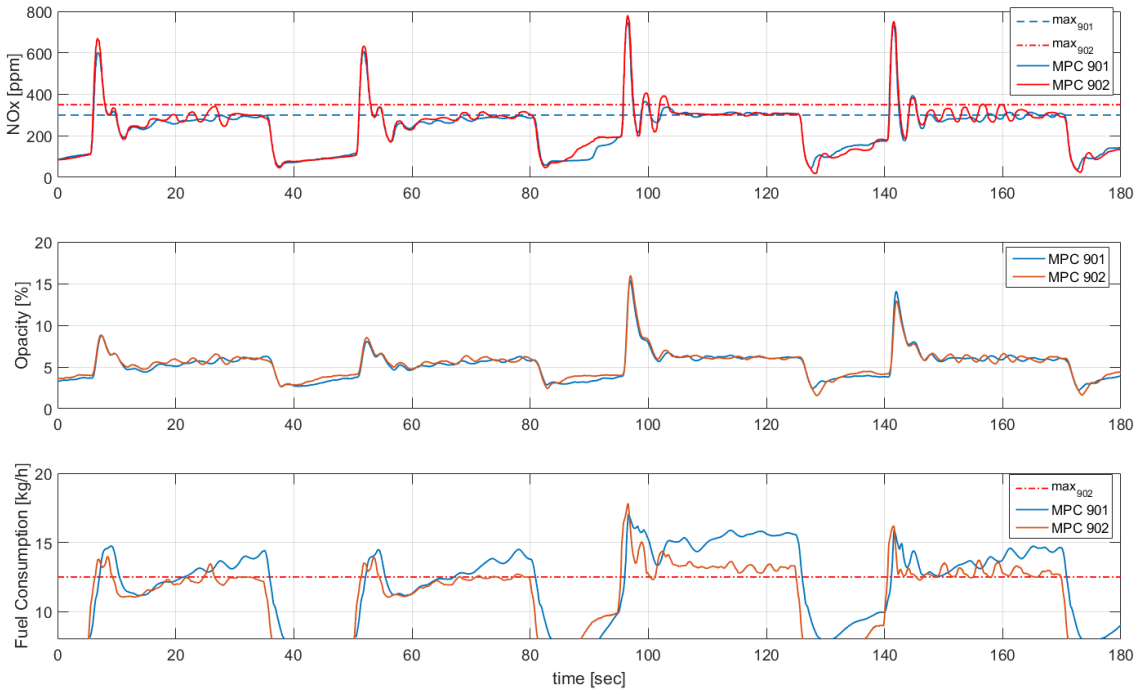

Figure 6.31: Effect of the hybrid powetrain on exhaust NO_x, opacity and fuel consumption, during step loading without λ reference.

6.2 Propeller Load

In the second set of experiments the engine simulates a propeller loading operation, with alternating speed and torque, as shown in Fig. 6.32 from measurements gathered by LME on-board a high speed vessel [13]. The power demand is a cubic function of the propeller rotational speed ($P = c \times N^3$). These data was scaled to fit properly in the range of the diesel engine output power and rotational speed limits. In the graph are clearly shown three fields of operation, the low-load area, where the engine is loaded during maneuvering, transient loading - acceleration area when the vessel departs from a port and the steady state area, which is the nominal operation point of the engine, when the vessel is maintaining its cruising speed. In this case a set of look-up tables was used, so as to provide dynamic λ reference points.

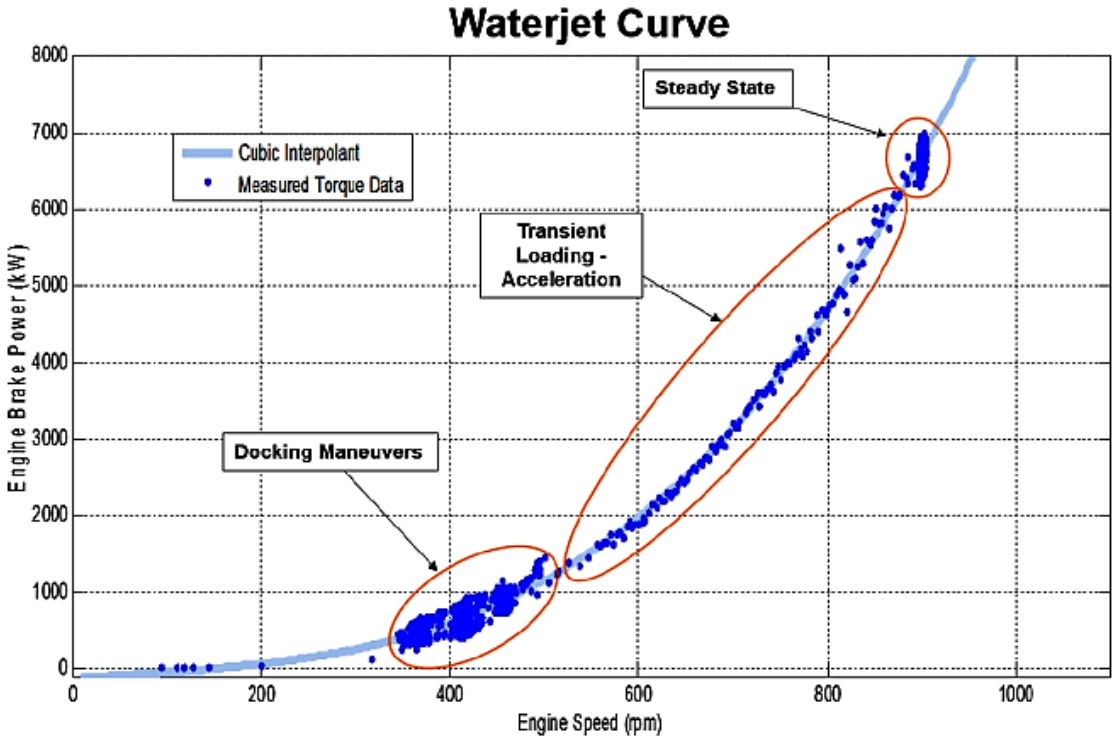

Figure 6.32: Measured power demand vs engine speed data on-board high-speed vessel [13].

In this scenario the outputs of the plant don't have such an abrupt behavior as in the step loading case seen in the previous section. As it can be noted from Fig. 6.33, 6.34 and 6.35, all controllers track successfully the λ value reference which is being provided by the look-up table.

Additionally, MPC 401 and MPC 900, which utilize the engine speed difference between the measurement and the reference, when the engine needs to speed up they increase the command value as they predict that the λ value will drop due to loading. Respectively, the command switches off, as the speed reference drops during unloading, creating an overspeeding of the engine.

As far as MPC 900 is concerned it manages to stay below the operational NOx and fuel consumption limits, by changing its dynamics, as shown in Fig. 6.33, at the end of the loading procedure.

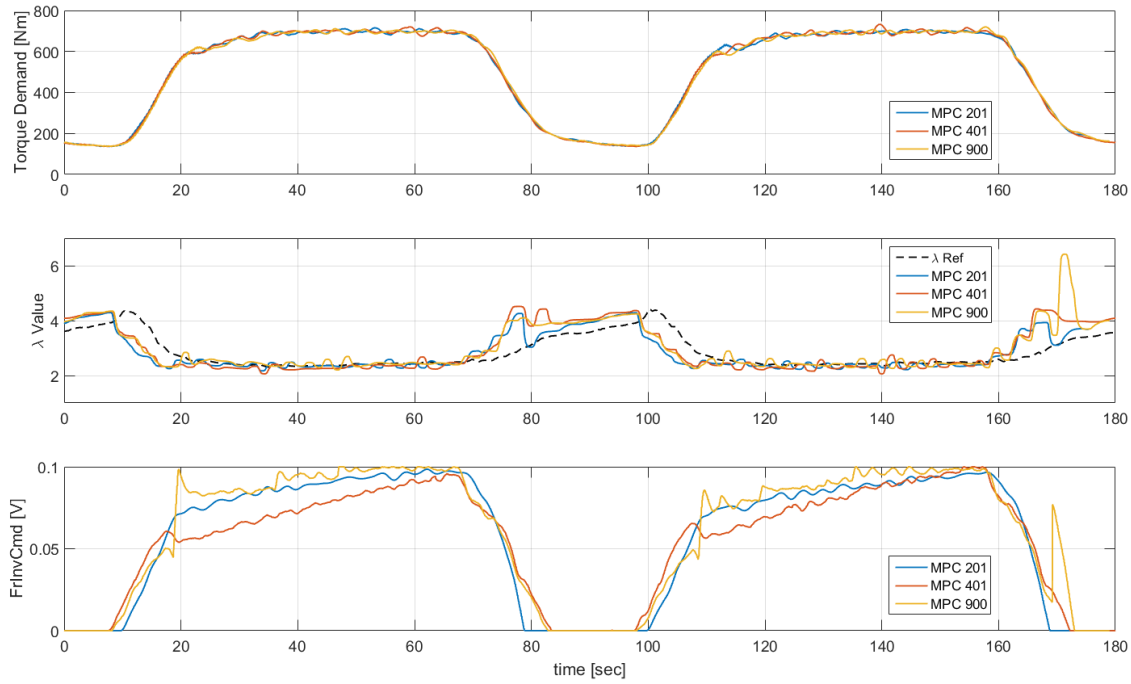

Figure 6.33: Measured total torque output, corresponding lambda values and hybrid controller command to the EM.

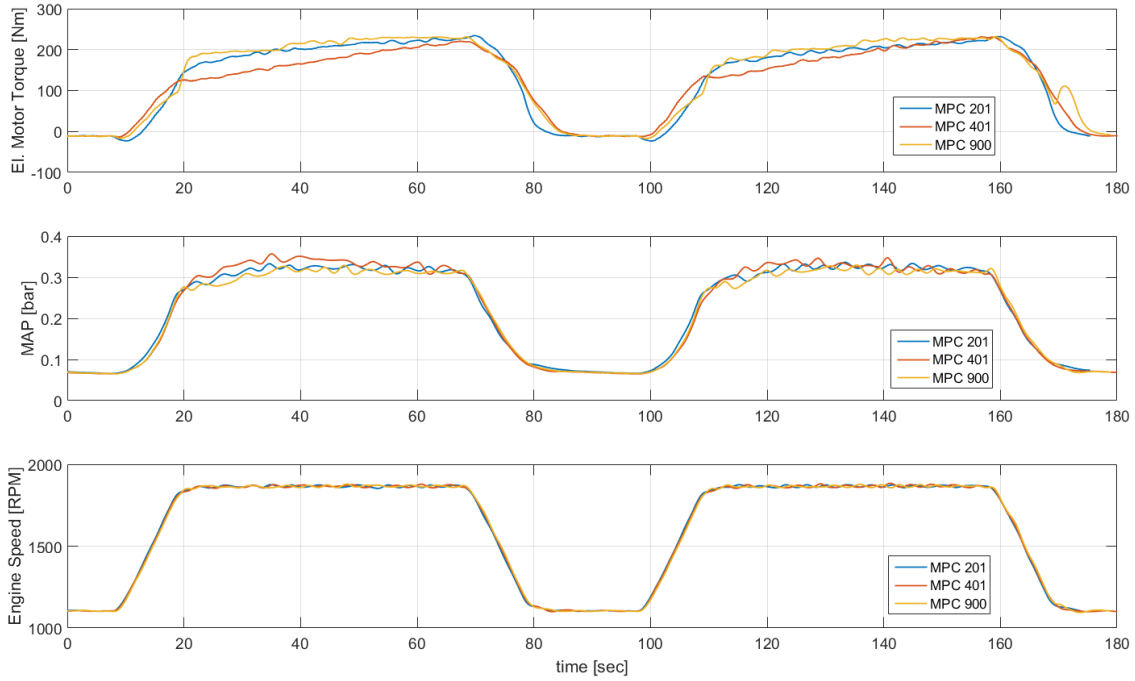

Figure 6.34: Electric torque, inlet pressure and engine speed.

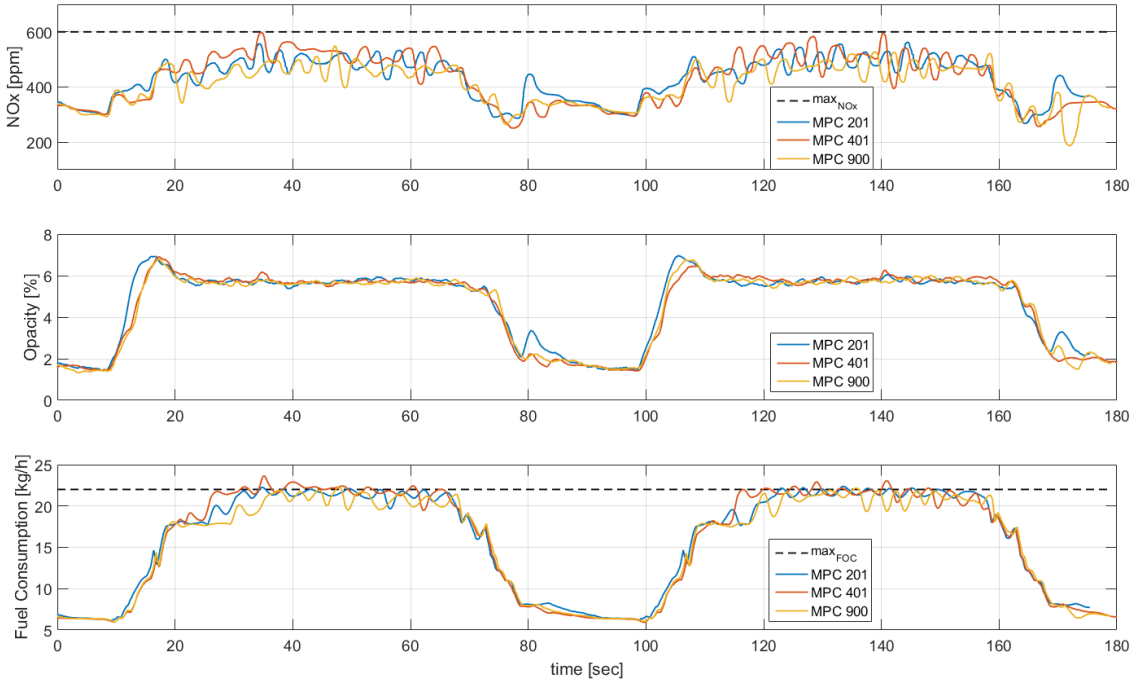

Figure 6.35: Effect of the hybrid powetrain on exhaust NO_x, opacity and fuel consumption, during propeller loading with dynamic λ reference.

Chapter 7

Conclusions and Future Work

Conclusions

In this thesis, the feasibility of using model predictive control in a hybrid diesel-electric marine powerplant has been investigated. Initially, identification experiments have been executed in order to extract models for the controller design procedure. Several MIMO models were identified and evaluated, using multiple signals as disturbance inputs. Also, a feedback manipulation was suggested, so that future disturbance input attributes can be predicted. As far as the controller design procedure is concerned, various MPC controllers were designed in order to control under closed-loop, the operation of the hybrid installation in multiple loading scenarios. The performance of controllers was tested also in simulation using open-loop experimental data.

Finally, the performance of the controllers was verified experimentally under realistic operating conditions on the hybrid diesel-electric testbed at LME. Experimental results showed the efficient control of the plant during transient operation, in terms of pollutants reduction in the emission content and improvement of fuel consumption efficiency through disturbance rejection, while delicately handling input and output constraints of the plant.

Conclusively, it can be derived that model predictive control is a sophisticated control method that can be successfully applied in marine hybrid powertrains, as it is capable of tackling multivariable systems, handle constraints, reject disturbances and achieve minimum wastage of energy.

Suggestions for Future Work

In this work, model predictive control under input and output constraints using linear MIMO internal models, with state feedback manipulation was considered.

Concerning the predictive control of the hybrid powertrain, the use of adaptive models or linear parameter-varying models can be suggested, in order to deal with diesel engine non-linearities, which could be also treated taking into account the uncertainty factor. Also, Explicit-MPC solution can be investigated, where the QP problem is solved off-line and stored in look-up tables, which provide the controller with the optimal solution at each control interval. With the explicit solution the sampling time of the can be drastically decreased, which is crucial in engine control applications. Moreover, a custom cost function could be investigated, using the desirable variables to be optimized.

Further to the field of MPC as an individual solution, the selection of optimal reference trajectory is of great concern, as if such existing, the controller can find the way of tracking it efficiently.

Appendices

Appendix A

N4SID Algorithm

A.1 Input-output matrix equation

$$Y_f = \Gamma_i X_i + H_i^d M_f + N_f \tag{A.1}$$

where

- The extended observability matrix Γ_i

$$\Gamma_i \stackrel{def}{=} \begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{i-1} \end{pmatrix} \tag{A.2}$$

- The deterministic lower block triangular Toeplitz matrix H_i^d

$$H_i^d \stackrel{def}{=} \begin{pmatrix} D & 0 & 0 & \dots & 0 \\ CB & D & 0 & \dots & 0 \\ CAB & CB & D & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{i-2}B & CA^{i-3}B & CA^{i-4}B & \dots & D \end{pmatrix} \tag{A.3}$$

- The stohastic lower block triangular Toeplitz matrix H_i^s

$$H_i^s \stackrel{def}{=} \begin{pmatrix} 0 & 0 & 0 & \dots & 0 \\ C & 0 & 0 & \dots & 0 \\ CA & C & D & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{i-2} & CA^{i-3} & CA^{i-4} & \dots & 0 \end{pmatrix} \tag{A.4}$$

- The input and output block Hankel matrices are defined as

$$U_{0|i-1} \stackrel{def}{=} \begin{pmatrix} u_0 & u_1 & \dots & u_{j-1} \\ u_1 & u_2 & \dots & u_j \\ \vdots & \vdots & \ddots & \vdots \\ u_{i-1} & u_i & \dots & u_{i+j-2} \end{pmatrix} \tag{A.5}$$

$$Y_{0|i-1} \stackrel{def}{=} \begin{pmatrix} y_0 & y_1 & \cdots & y_{j-1} \\ y_1 & y_2 & \cdots & y_j \\ \vdots & \vdots & \ddots & \vdots \\ y_{i-1} & y_i & \cdots & y_{i+j-2} \end{pmatrix} \tag{A.6}$$

where it is assumed for stohastic reasons that $j \to \infty$.

A.2 Singular Value Decomposition (SVD)

For Singular Value Decomposition, the defined notion of orthogonal projection difined in chapter 3 is needed [5]. By projecting the row space of Y_f into the orthogonal complement $U_f^\perp$ of the row space of U_f so that

$$Y_f U_f^\perp = \Gamma_i X_i / U_f^\perp + H_i^s M_f / U_f^\perp + N_f / U_f^\perp$$

and since it is assumed that the noise is uncorrelated with the inputs

$$M_f / U_f^\perp = M_f, \quad N_f / U_f^\perp = N_f.$$

Therefore

$$Y_f U_f^\perp = \Gamma_i X_i / U_f^\perp + H_i^s M_f + N_f.$$

The following step consists in weighting this projection to the left and to the right with some matrices W_1 and W_2

$$W_1 Y f / U_j^\perp W_2 = \underbrace{W_1 \Gamma_i}_{1.} \underbrace{X_i / U_f^\perp W_2}_{2.} + \underbrace{W_1 (H_i s^s M_f + N_f) W_2}_{3.}.$$

Of course, the inputs U_f and the weighting matrices W_1 and W_2 can not be chosen arbitrarily but they should satisfy the following three conditions

1.
$$rank\ (W_1 \Gamma_i) = rank\ \Gamma_i \tag{A.7}$$

2.
$$rank\ (X_i / U_f^\perp W_2) = rank\ X_i \tag{A.8}$$

3.
$$W_1 (H_i s^s M_f + N_f) W_2 = 0 \tag{A.9}$$

The first two conditions guarantee that the rank-n property of $\Gamma_i X_i$ is preserved after projection onto $U_f^\perp$ and weighting by W_1 and W_2. The third condition expresses that W_2 should be uncorrelated with the noise sequences w_k and v_k. If these three conditions are satisfied, then

$$\mathcal{O}_i \stackrel{def}{=} W_1 Y_f / U_f^\perp W_2 = W_1 \Gamma_i X_i / U_f^\perp W_2 \tag{A.10}$$

with SVD

$$\mathcal{O}_i = \begin{pmatrix} U_1 & U_2 \end{pmatrix} \begin{pmatrix} S_1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} V_1^T \\ V^2 \end{pmatrix}$$

The following important properties can now be stated

$$rank\ \ \mathcal{O}_i = n,$$

$$W_1\Gamma_i = U_1 S_1^{1/2}$$
$$X_i/Uf^{\perp}W_2 = S^{1/2}V_2^T$$

Obviously, the singular value decomposition of the matrix $W_1Y_f = U_f^{\perp}W_2$ delivers the order n of the system. Moreover, from the left singular vectors corresponding to the non-zero singular values the extended observability matrix Γ_i can be found (up to a similarity transformation) where as the right singular vectors contain information about the states X_i. For an appropriate choice of the weighting matrix W_2, the matrix

$$\tilde{X}_i \stackrel{def}{=} X_i/U_f W_2 \tag{A.11}$$

can indeed be considered as an estimate of the state sequence X_i. For a particular choice of W_2, $\tilde{X}_i$ is a Kalman filter estimate of X_i. By choosing appropriate weighting matrices W_1 and W_2, all subspace algorithms for LTI systems can be interpreted in the above framework, including N4SID algorithm.

N4SID algorithm uses the state estimates $\tilde{X}_i$ to find the state space model. The weights W_1 and W_2 at N4SID correspond to the following values

$$W_1 = I_{li} \tag{A.12}$$
$$W_2 = (W_p/U_f^{\perp})^{\dagger}W_p \tag{A.13}$$

By selecting as defined above W_1 and W_2 , the estimated state sequence $\tilde{X}_i$ can be interpreted as the solution of a bank of Kalman filters, working in parallel on each of the columns of the matrix W_p. Besides $\tilde{X}_i$, we also need the state sequence $\tilde{X}_{i+1}$. This sequence can be obtained from a projection and new weights $\overline{W}_1$, $\overline{W}_2$ in Eq. (A.10 based on $W_{0|i}$, $Y_{i+1|2i-1}$ and $U_{i+1|2i-1}$. This leads to the sequence $\mathcal{O}_{i+1}$ and the Kalman filter states $\tilde{X}_{i+1}$

$$\mathcal{O}_{i+1} = \overline{W}_1 Y_{i+1|2i-1}/U_{i+1|2i-1}^{\perp}\overline{W}_2 = \overline{W}_1\Gamma_{i-1}\tilde{X}_{i+1}\overline{W}_2$$

The state space matrices $A,\ B,\ C,\ D$ can now be found by solving a simple set of over determined equations in a least squares sense.

$$\begin{pmatrix}\tilde{X}_{i+1}\\ Y_{i|i}\end{pmatrix} = \begin{pmatrix}A & B\\ C & D\end{pmatrix}\begin{pmatrix}\tilde{X}_i\\ U_{i|i}\end{pmatrix} + \begin{pmatrix}\rho_w\\ \rho_v\end{pmatrix} \tag{A.14}$$

with obvious definitions for ρ_w and ρ_v as residual matrices. This reduces to

$$\underset{A,B,C,D}{min}\left\|\begin{pmatrix}\tilde{X}_{i+1}\\ Y_{i|i}\end{pmatrix} - \begin{pmatrix}A & B\\ C & D\end{pmatrix}\begin{pmatrix}\tilde{X}_i\\ U_{i|i}\end{pmatrix}\right\|_F^2$$

Finally, the noise covariances $Q,\ S$ and R can be estimated from the residuals ρ_w and ρ_v as

$$\begin{pmatrix}Q & S\\ S^T & R\end{pmatrix}_i = \frac{1}{j}\left[\begin{pmatrix}\rho_w\\ \rho_v\end{pmatrix}\left(\rho_w^T\rho_s^T\right)\right] \geq 0$$

where the index i denotes a bias induced for finite i, which disappears as $i \to \infty$. As is obvious by construction, this matrix is guaranteed to be positive semi-definite. This is an important feature since only positive definite covariances can lead to a physically realizable noise model.

Bibliography

[1] S. Adachi, M. Iwadare, M. Ueno. 2009. Multi-Variable Air-Path Management for a Clean Diesel Engine Using Model Predictive Control. *SAE International.*

[2] L. Ljung. 2015. System Identification Toolbox. *Mathworks.*

[3] L. Ljung. 1999. System Identification. Theory for the User. *PRENTICE HALL PTR.*

[4] D. Alberer. H. Anderson 2012. Identification for Automotive Systems. *Springer.*

[5] W. Favoreel, B. de Moor, P. V. Overschee. 2000. Subspace state space system identication for industrial processes. *Journal of Process Control.*

[6] D. B. Rodriguez. 2014. Modelling and Observation of Exhaust Gas Consentrations for Diesel Engine. *Springer.*

[7] D. B. Rodriguez. 2014. Modelling and Observation of Exhaust Gas Consentrations for Diesel Engine. *Springer.*

[8] J.M. Maciejowski. 2000. Predictive Control with constraints. *PRENTICE HALL PTR.*

[9] A. Bemporad, M. Morari, N. L. Ricker. 2015. Model Predictive Control Toolbox. User's Guige. *Mathworks.*

[10] A. Bemporad, M. Morari, N. L. Ricker. 2015. Model Predictive Control Toolbox. Getting Started Guide. *Mathworks.*

[11] C. Schmid, L. T. Biegler. 1994. Quadratic programming methods for reduced Hessian SQP. *Computers and Chemical Engineering. Vol. 18, No. 9, 1994, pp. 817-832.*

[12] L. Li and T. J. Walsh. 2008. Knows What It Knows: A Framework for Self-Aware Learning. 25^{th} *International Conference on Machine Learning (ICML-08).*

[13] S. Topaloglou, G. Papalambrou, N. Kyrtatos. 2016. Energy Management Controller Design for Hybrid Ship Propulsion During Transient Operation. *CIMAC congress, Helsinki, 6-10 June 2016, paper No. 50.*

[14] A. Stefanopoulou, R. Smith. 1998. Emissions and Performance Tradeoffs for Advanced Marine Diesel Propulsion. *IFAC Workshop on Advances in Automotive Control.*

[15] P. Ortner, L. del Re. 2007. Predictive control of a Diesel Engine Air Path. *IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY, VOL. 15, NO. 3, p. 449-456.*

[16] P. Majecki, G.M. van de Molen, M. Grimble, I. Haskara, Y. Hu, C.F.Chang. 2015. Real-Time Predictive Control for SI Engines Using Linear Parameter-Varying Models. *IFAC-PapersOnLine 48-23 (2015) p. 094-101.*

[17] Wang L. 2009. Model Predictive Control System Design and Implementation Using Matlab. *Springer.*

[18] C. Rakopoulos and E. Giakoumis. 2006. Review of thermodynamic diesel engine simulations under transient operating conditions. SAE 2006 Transactions Journal of Engines, (2006-01-0884).

[19] A. Taghavipour, N. L. Azad, J. McPhee. 2015. Real-time predictive control strategy for a plug-in hybrid electric powertrain. *Mechatronics vol. 29 p. 13-27.*

[20] L. Guzzella, C. Onder. 2004. Introduction to Modeling and Control of Internal Combustion Engine Systems. *Springer, 2nd edition.*

[21] A. Alessio, A. Bemporad. 2009. A survey on explicit model predictive control. *Lecture notes in control and information sciences, vol. 384, Springer, p. 345-369.*

[22] A. Taghavipour, N.L. Azad, J. McPhee. 2012. An optimal power management strategy for power split plug-in hybrid electric vehicles. *International Journal of Vehicle Design vol. 60 (3/4), p. 286-304.*

[23] S. Trimboli, S. Di Cairano, A. Bemporad, I. Kolmanovsky. 2009. Model predictive control for automotive time-delay processes: an application to air-to-fuel ratio. *Proceedings of 8th IFAC workshop time-delay systems, p. 1-6.*

[24] G. Ripaccioli, A. Bemporad, F. Assadian, C. Dextreit, S. Di Cairano I. Kolmanovsky. 2009.Hybrid modeling, identification, and predictive control: an application to hybrid electric vehicle energy management: hybrid systems: computation and control. *Lecture notes in computer science, vol. 5469. Springer, p. 321-335.*

[25] H. Borhan, A. Vahidi, A.M. Phillips, I. Kolmanovsky. 2009. Predictive energy management of a power-split hybrid electric vehicle. *American control conference, p. 397-406..*

[26] A. Taghavipour, M. Vajedi, N.L. Azad, J. McPhee. 2012. Predictive power management strategy for a PHEV based on different levels of trip informatio.*IFAC workshop on engine and powertrain control, simulation and modeling, vol. 3, no. 1, p. 326-333..*