Diploma thesis

Exhaust Gas Recirculation and Air Flow Virtual Sensors for Internal Combustion Turbocharged Engines

Jaroslav Tabaček

May 2016

supervisors: Ing. Daniel Pachner, PhD.
Ing. Ondřej Šantin

Czech Technical University in Prague
Faculty of Electrical Engineering, Department of Control Engineering

Czech Technical University in Prague
Faculty of Electrical Engineering

Department of Control Engineering

DIPLOMA THESIS ASSIGNMENT

Student: **Bc. Jaroslav Tabaček**

Study programme: Cybernetics and Robotics
Specialisation: Systems and Control

Title of Diploma Thesis: **Exhaust Gas Recirculation and Air Flow virtual sensors for internal combustion turbocharged engines**

Guidelines:

1. Study the literature on turbocharged engine gas exchange system modeling. Review various modeling approaches.
2. Design a nonlinear observer for the gas exchange model of an example engine.
3. Study the impact of various component models accuracies on the virtual sensor accuracy. Discuss the ideal compromise between accuracy and model complexity.
4. Demonstrate a real-time ready virtual sensor prototype. Evaluate the computational time and memory requirements. Study the algorithm applicability considering typical Engine Control Unit hardware parameters.

Bibliography/Sources:

[1] Guzzella, L., and Onder, C., Introduction to Modeling and Control of Internal Combustion Engine Systems, Berlin: Springer, 2004.
[2] Jensen, J., Kristensen, A., Spencer, C. Sorenson, S., Houbak, N. et al., Mean Value Modeling of a Small Turbocharged Diesel Engine, SAE Technical Paper 910070, 1991.
[3] Julier, J. S., and Uhlman, K., A New Extension of the Kalman Filter to Nonlinear Systems, 1997.
[4] Relevant journal and conference papers

Diploma Thesis Supervisor: Ing. Ondřej Šantin

Valid until till the summer semester 2016/2017

L.S.

prof. Ing. Michael Šebek, DrSc.
Head of Department

prof. Ing. Pavel Ripka, CSc.
Dean

Prague, February 29, 2016

Acknowledgement

Especially I am grateful to my supervisors, Ing. Daniel Pachner, Ph.D. and Ing. Ondřej Šantin, whose guidance enabled me to develop an understanding of the subject.

I would also like to thank my family who inspired, encouraged and fully supported me in every way. Lastly, I offer my regards to all of those who supported me in any respect during the completion of this thesis.

Last but not least, I am thankful to *Honeywell International, Inc.* for providing the data.

Declaration

I declare that I worked out the presented thesis independently and I quoted all used sources of information in accordance with Methodical instructions about ethical principles for writing academic thesis.

Prague, 27th May 2016

Jaroslav Tabaček

Abstrakt

V dnešnej dobe sa emisné normy pravidelne sprísňujú, čo núti automobilový priemysel vytvárať pokročilé riadenia motorov, ktoré vyžadujú presné merania. Taktiež výmena fyzických senzorov za virtuálne bez straty presnosti meraní je veľmi žiadaná kvôli zníženiu výrobných nákladov. Informácia o hodnote hmotnostného toku recirkulácie výfukových plynov (EGR) je veľmi dôležitá pre riadenie emisií vznikajúcich pri spaľovaní paliva a meranie tejto hodnoty je zložité. Tieto vlastnosti robia z hmotnostného toku EGR ideálneho kandidáta pre meranie virtuálnym senzorom.

Táto diplomová práca popisuje proces vytvorenia virtuálneho senzoru pre hmotnostný tok EGR aplikovaného v preplňovanom naftovom motore pre ťažké vozidlá, ktorý by mohol byť alternatívou pre súčasne používaný Venturi senzor. Hmotnostný tok vzduchu úzko súvisí s hmotnostným tokom EGR, preto je virtuálny senzor pre hmotnostný tok vzduchu vytvorený ako vedľajší produkt.

Základ dobrého virtuálneho senzoru je spoľahlivý model, preto je v tejto práci preskúmaných niekoľko modelovacích prístupov pre vytvorenie modelu vzduchovej cesty motoru. Každý z týchto prístupov je založený na vytvorení modelu pre každý komponent vzduchovej cesty. Samostatné komponenty sú bližšie predstavené, pre každý z nich je vytvorený polynomiálny model a reálne dáta ustálených stavov sú použité pre ich identifikáciu. Kompletné štruktúry modelov vzduchovej cesty ktoré združujú modely komponentov sú vysvetlené a zhodnotené z hľadiska robustnosti, presnosti a zložitosti. Vhodné modely sú potom simulované a zhodnotené sú možnosti ďalšieho použitia.

Na základe týchto zistení je vybratý najvhodnejší model na vytvorenie virtuálneho senzoru. Návrh virtuálneho senzoru bere do úvahy všetky vhodné merania, ktoré sa normálne vyskytujú v bežných motoroch, a ktoré by mohli zjednodušiť štruktúru modelu motoru. Finálny návrh pozorovateľa je kombináciou redukovaného modelu motora, meraní lambda senzoru a lineárneho Kalmanovho filtra s plánovaným zosilnením. Taktiež sú zvážené možnosti implementácie senzoru na riadiacu jednotku motora.

Kľúčové slová

preplňovaný naftový motor; kompaktný model motoru; polynomiálny model; virtuálny sensor; hmotnostný tok EGR; hmotnostný tok vzduchu

Abstract

The emission standards are regularly made more stringent these days which put pressure on automotive industry to create advanced engine control that require the accurate measurements. Also, replacing hardware sensors with virtual sensors without loss of accuracy is highly desirable due to their lower production cost. The exhaust gas recirculation (EGR) mass flow value is very important information for controlling combustion emissions and its physical measuring is difficult, which makes it an ideal candidate for virtual sensing.

This thesis describes a process of a creation of EGR mass flow virtual sensor for heavy-duty turbocharged diesel engine with EGR, which could be an alternative for Venturi sensor. The air mass flow is closely related to the EGR mass flow therefore the air mass flow virtual sensor is created as a by-product.

A basis of good virtual sensor is reliable model, so few modelling approaches for design of engine's air path model are reviewed. Each of them requires a model creation for each air path component. The particular components are introduced, a polynomial model is created for each one and real steady-state data from a diesel engine are used for their identification. The complete air path model structures which merge the component models are developed and discussed from robustness, accuracy and complexity point of view. Eligible models are simulated and their applicability for further use is also discussed.

Based on these findings, the best model is used to create virtual sensor. The virtual sensor design respects all eligible measurements available in a common production engine, which results in simplifying the engine model structure. The final observer design is a combination of a reduced engine model, lambda sensor measurements and a linear Kalman filter with scheduled gain. The possibility of the implementation on Engine Control Unit is discussed.

Keywords

turbocharged diesel engine, compact engine model, polynomial model, virtual sensor, EGR mass flow, air mass flow

Contents

Symbols

p_i	Pressure at a component inlet (Pa)
p_o	Pressure at a component outlet (Pa)
T_i	Temperature at a component inlet (K)
T_o	Temperature at a component outlet (K)
p_0	Ambient air pressure (Pa)
p_1	Pressure after compressor (Pa)
p_3	Pressure before turbine (Pa)
T_0	Ambient air temperature (K)
T_1	Temperature after compressor (K)
T_{1b}	Temperature after intercooler (K)
T_2	Intake manifold temperature (K)
T_3	Temperature before turbine (K)
T_c	Temperature of coolant (K)
T_{3b}	Temperature after EGR cooler (K)
$\dot{m}_a$	Fresh air mass flow ($\mathrm{kg\,s^{-1}}$)
$\dot{m}_{ch}$	Charge mass flow ($\mathrm{kg\,s^{-1}}$)
$\dot{m}_e$	Exhaust gas recirculation mass flow ($\mathrm{kg\,s^{-1}}$)
$\dot{m}_f$	Fuel mass flow ($\mathrm{kg\,s^{-1}}$)
$\dot{m}_t$	Turbine mass flow ($\mathrm{kg\,s^{-1}}$)
$\dot{m}_{in}$	Engine intake mass flow ($\mathrm{kg\,s^{-1}}$)
$\dot{m}_{exh}$	Engine exhaust mass flow ($\mathrm{kg\,s^{-1}}$)
N_{tc}	Turbocharger speed (rpm)
N_t	Turbine rotational speed (rpm)
N_c	Compressor rotational speed (rpm)
P_t	Turbine power (W)
P_c	Compressor power (W)
I_{tc}	Turbocharger moment of inertia ($\mathrm{kg\,m^2}$)
N_e	Engine's rotational speed (rpm)
η_c	Compressor efficiency (-)
η_t	Turbine efficiency (-)
η_{vol}	Volumetric efficiency (-)
R	Ideal gas constant ($\mathrm{J\,K^{-1}\,mol^{-1}}$)
Π	Outlet to inlet pressure ratio (-)
c_d	Discharge coefficient (-)
c_{pa}	Specific heat capacity at constant pressure of air ($\mathrm{J\,kg^{-1}\,K^{-1}}$)
c_{pe}	Specific heat capacity at constant pressure of exhaust gas ($\mathrm{J\,kg^{-1}\,K^{-1}}$)
c_{va}	Specific heat capacity at constant volume of air ($\mathrm{J\,kg^{-1}\,K^{-1}}$)

c_{ve}	Specific heat capacity at constant volume of exhaust gas ($\mathrm{J\,kg^{-1}\,K^{-1}}$)
r_{egr}	EGR rate (-)
u_{egr}	EGR valve position (-)
u_{VGT}	Turbine nozzles position (-)
γ	Adiabatic index (-)
Φ	Head parameter (-)
Ψ	Flow rate parameter (-)
M	Mach number (-)
d_c	Compressor diameter (m)
V_d	Displaced volume ($\mathrm{m^3}$)

1 Introduction

Maximizing efficiency of engines, minimizing production cost and fulfilling various emission standards, these are some of the goals of the automotive industry. There are many ways to achieve such goals from the control engineering point of view. Advanced control and estimation algorithms are one of them. They can be used to control the engine more effectively or to replace expensive hardware sensors by virtual ones. However, such algorithms are usually complex and need to be computed in real-time which means there has to be available adequate computing power. Most of currently made vehicles do not have ECU (Engine Control Unit) that is able to run such algorithms. However, some engines already use virtual sensors (turbo-speed, NO_x) but they are usually very simplified (e.g. based on the static maps). Even if the computational power will increase in the future, computational efficiency and robustness will remain important. For these reasons, the estimation and control algorithms are usually transformed to numerically convenient forms which are then implemented in ECU's.

The configuration of the EGR (exhaust gas recirculation) systems differ in heavy-duty engines and light-duty engines [1]. Also the approach for measuring the EGR mass flow is different. The light-duty engines use to have MAF (mass air flow) sensor placed before a compressor. Then the EGR flow is obtained based on that measurement and a model of engine's charge flow [2] [3]. However, this thesis is focused on the heavy-duty diesel engines. The most common method is to use a Venturi sensor [4] which calculates EGR flow from differential pressure and temperature. Then it can be combined with intake manifold pressure, temperature and engine speed [5] to determine the charge flow. This method is accurate, but difficult to calibrate. Actually, the exact methods for measuring the EGR mass flow used in production engines are difficult to obtain because automotive companies are keeping their technology confidential. One of the reasons for this is the legislation for the emission standards that is repetitively updated. Therefore, the companies have to regularly innovate their strategies to fulfil these strict standards which makes the know-how highly valuable. One of the solutions for meeting the emission standard is using SCR (selective catalytic reduction) system [6]. It is a system for reduction of NO_x in exhaust gas using urea injection and catalysts. In order to correctly control this system, NO_x measurements before and after the SCR has to be available. The oxygen concentration has to be measured in order to achieve correct NO_x measurements, hence lambda sensors are used together with NO_x sensors. The lambda sensors can also be used as mass flow sensors, but their measurement is delayed. This raised the question in the automotive industry if it is possible to remove the Venturi sensor and use the lambda sensor for measuring the EGR mass flow and the air mass flow instead. It seems to be possible, since few automotive companies already released new versions of engines without the Venturi sensor.

In order to design an accurate virtual sensor, the state of the system has to be reconstructed by state estimation algorithm. The engine system is nonlinear and very complex therefore the nonlinear estimation algorithms (e.g. Extended Kalman Filter [7] [8] [9]) are usual choice. The precision and simplicity of the engine model are essential goals of creating a reliable virtual sensor.

The first step of a virtual sensor development process is to create a reliable mathematical model that simulates the behaviour of a real system with sufficient accuracy. The usual form of control oriented systems is a set of ordinary differential equations (ODE). Such description of an air system of a turbocharged engine is introduced in [10] and [11]. However, the engine air path system model generally has fast and slow dynamics which differ substantially, i.e. its mathematical description includes stiff equations. This introduces unwanted complexity into the design, because solving such equations is numerically difficult. It brings problems with running it in real-time and designing an observer for such system is also challenging. Differential Algebraic Equations (DAE) model is much more convenient for these purposes. It removes stiffness by replacing fast dynamics with algebraic equations [10] [12]. It includes a set of implicit equations and overall has better numerical properties than the ODE model. Even more efficient is model structure which connects particular components expressed by explicit functions with simple iterative methods [13] [14].

This thesis's goal is to design the virtual sensors for the EGR mass flow and the air mass flow with reasonable trade off between the accuracy and the computational complexity. The accuracy should be comparable to the accuracy of the Venturi sensor which would make the Venturi sensor unnecessary.

Since a good engine model is a key to successful development of a virtual sensor, the thesis focuses thoroughly on finding the best available option. Several engine model structures are discussed throughout the thesis, but only one is chosen as a basis for the virtual sensor development and its properties are described in detail. All of the reviewed model structures consist of the models of particular components of engine's air path. For this reason, the components of the engine are introduced one by one and their properties are discussed due to their impact on the overall model's accuracy and complexity.

The virtual sensor design should also encounter generally available physical measurements that are relevant for improving its accuracy. Therefore, the virtual sensor designed in this thesis uses measurements that are available in common heavy-duty turbocharged diesel engine with EGR. This allows the proposed engine model to be reduced, so instead of the complete model, only its subset is used as the basis of the virtual sensor. Also, using the lambda sensor measurements enables us to use a linear observer rather than nonlinear. These improvements results in the eligibility of the virtual sensor for the implementation on ECU.

The text is organized as follows: A basic description of an air path subsystem of a turbocharged diesel engine is introduced in the beginning of the thesis. Next, the overview of structures of desired mathematical models of engine are described. The next chapter is dedicated to individual components of the modelled subsystem of the turbocharged engine. The purpose of the engine elements are shortly outlined in each chapter. Then mathematical equations in convenient forms that describe components behaviour are derived based on physical principles and empirical relationships between dependent variables. After that, the text includes the chapter devoted to the engine model implementation. This chapter merges the engine model structures with models of the engine components. First the model structures are described in detail. Then the developed models are simulated and compared to each other and real data. To conclude, the last chapter is devoted to the design of virtual sensors for the EGR mass flow and the air mass flow. The structures of the sensors are described and the implementation in a real-time prototype is discussed.

2 Engine Model Structure

The studied system model represents an air path system of a turbocharged diesel engine with an exhaust gas recirculation system. It is schematically described in Figure 1.

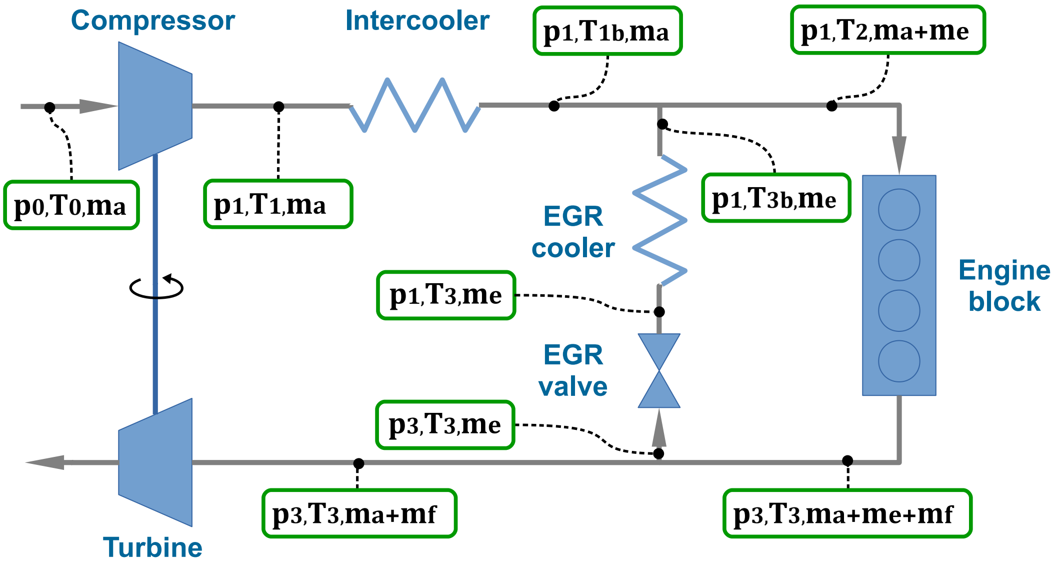

Figure 1 Engine model. The modelled system with all components and states. The states are further described in Table 1.

First the inlet air is compressed in a compressor and cooled in an intercooler in order to increase its density for the combustion. Then it is mixed with exhaust gas in order to minimize the NO_x (Nitrogen oxides) emissions that the engine produces. After that the mixture is flown into engine cylinders where fuel is added and mechanical energy is produced as a result of the combustion process. The part of the exhaust gas goes through an exhaust gas recirculation (EGR) feedback back to the engine intake and the rest is used to power a turbine that is connected to the compressor by a shaft. The turbine has variable geometry, which means that an angle of nozzles can be controlled.

The model described in this report is calibrated on data from 4-cylinder turbocharged 2.2 Litre diesel engine.

p_0	ambient air pressure
p_1	pressure after compressor
p_3	pressure before turbine
T_0	ambient air temperature
T_1	temperature after compressor
T_{1b}	temperature after intercooler
T_2	intake manifold temperature
T_3	temperature before turbine
T_{3b}	temperature after EGR cooler
$\dot{m}_a$	fresh air mass flow
$\dot{m}_f$	fuel mass flow
$\dot{m}_e$	exhaust gas recirculation mass flow
N_{tc}	rotational speed of a turbocharger shaft (turbo-speed)

Table 1 Engine variables. The table clarify denoting of the engine variables.

Since the model is to be used in state estimation algorithms, it is advantageous to maintain reasonable precision and keep the complexity as low as possible at the same time. The model serves as a basis of a virtual sensor, therefore the more accurate model gives the better chance to create an accurate virtual sensor. However, the more complex the structure of the model gets the more computational power will be needed. The goal is to found the trade off between these two criteria, that also fulfil requirements. These models are called control oriented models and such models are described in this thesis. The next sections introduce models that are eligible options for the basis of the virtual sensor.

2.1 ODE Model

The most common model approach is to create a set of ordinary differential equations (ODE) in time domain as follows

$$\dot{x}(t) = f\left(t, x(t), u\right), \tag{1}$$

where t is time, $x(t)$ are system states and u represents a vector of system inputs. Design of such model can be based on so called receivers [10]. They are assumed to be fixed volumes with the same thermodynamic states over the entire volume. Such receivers are the intake and the exhaust manifold. The model is completed by adding the differential equation of a turbo-speed. The state variables vector is defined as

$$x(t) = \left[p_1,\, p_3,\, N_{tc},\, T_3\right], \tag{2}$$

where notation of the particular states is according Table 1. Other model variables can be described by algebraic equations. This model structure is further discussed in [10].

The issue is that the system of interest is stiff, i.e. it has very fast as well as slow dynamics. This phenomenon is a result of high nonlinearity and non-differentiability of the right hand side of the differential equation at some points $x(t)$ that cause difficulties in a numerical solution.

2.2 DAE Model

The proposed solution is to use a set of differential algebraic equations (DAE) in a form of a system of implicit equations written as

$$\mathbf{F}\left(\dot{x}(t), t, x(t)\right) = 0. \tag{3}$$

The first advantage is that a system order can be reduced compared to ODE (2.1). The fast dynamics is replaced with static effects which are represented by algebraic equations. This removes stiffness from a model, hence the model its evaluation can be computed cheaply on ECU. Furthermore, the time derivatives can be isolated which forms the model into semi-explicit structure.

In order to transform a stiff ODE system into a DAE system the ODEs can be approximated by polynomials or rational functions in an appropriate form using key variables. The particular forms differ case-by-case and have to be chosen individually for each component. By combining the separate components equations into one system, the pursued model is designed.

The model designed in this report is a mean-value model (MVM) which means it neglects discrete cycles of the engine and assumes that processes and effects are spread out over engine cycle [10]. In this case the mean value model is also lumped parameter model, i.e. it has no spatially varying parameters and the only independent variable is time.

The variables describing the state of the ODE engine model (2.1) can be traced to two different types in the DAE model. Firstly the static variables which achieve their equilibria immediately at any time and secondly the true dynamic variables which change with respect to time according to their own history. Hence they represent a memory (accumulation) of the system. If we simplify the model by omitting the energy and mass accumulation in the gas system, a single dynamic state variable remains in the model - the turbo-speed.

The final model consists of 10 implicit equations (See Table 2). Each equation interprets phenomenon that bounds to separate part of engine. The equations are defined in Chapter 3.

F_{Cm}	Compressor mass flow eq.
F_{Ct}	Compressor outlet temperature eq.
F_{Tm}	Turbine mass flow eq.
F_P	Turbocharger power balance eq.
F_{EGRm}	Exhaust gas recirculation valve mass flow eq.
F_{Hegr}	Heat exchanger eq. (Exhaust gas recirculation cooler)
F_{Hcac}	Heat exchanger eq. (Charge air cooler)
F_M	Mixing plenum eq.
F_V	Volumetric efficiency eq.
F_{Et}	Exhaust temperature eq.

Table 2 Name list of equations. List summarizes all equations included in final model.

We are able to derive any of the state variables by solving the set of implicit nonlinear equations what can be done using Newton-Raphson method. The algorithm is depicted on Figure 2.

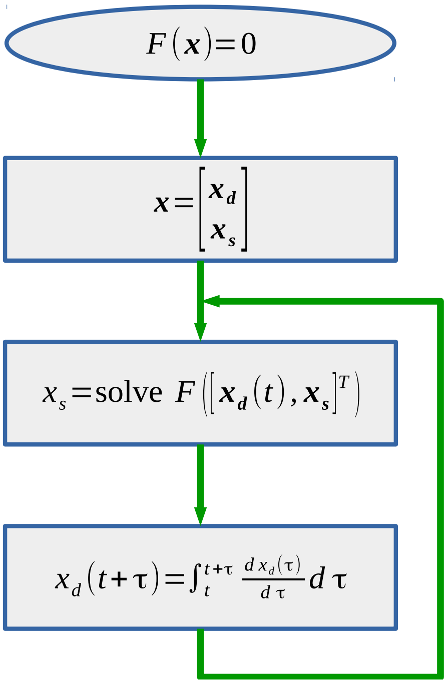
$F(\boldsymbol{x})=0$
$\boldsymbol{x}=\begin{bmatrix}\boldsymbol{x_d}\\ \boldsymbol{x_s}\end{bmatrix}$
$x_s=\text{solve } F\left(\left[\boldsymbol{x_d}(t),\boldsymbol{x_s}\right]^T\right)$
$x_d(t+\tau)=\int_t^{t+\tau}\frac{d x_d(\tau)}{d\tau}d\tau$

Figure 2 Algorithm. The figure illustrates the algorithm for determining solutions of a set of DAE. First the states are distinguished as dynamic (x_d) or static (x_s). Then the solution is split into solution of the algebraic (static) part and dynamic part as it can be seen in the figure.

In the real-time implementation the set of equations can be solved offline over a grid and the particular solutions can be then found very efficiently using an interpolation table. However, since there can be multiple solutions for the same equations, the interpolation could be invalid. When preparing the interpolation table, it is necessary to ensure that the mapping of external input values to the selected solutions of the model equations represent a homotopy [15]. To achieve that, it is important not to select solutions which do not have physical sense although they are formally solving the equations.

Such approach results in the creation of the nonlinear model suitable for use in advanced state estimation or control algorithms. However the algorithm solving the interpolation issue is beyond the scope of this thesis.

2.3 Quasi-Explicit Model

The solution proposed above has satisfactory numerical properties and accuracy requirements. However, the conditions for the convergence of the Newton-Raphson method have to be carefully defined or some globally convergent (or convergent on interval) alternative to the Newton-Raphson method should be applied which would require more computational power. To overcome this issues the more convenient model is proposed. The solution is based on the creation of a steady-state model which output is then used in turbo-speed differential equation. The turbo-speed is propagated into another steady-state model output of which capture transient behaviour. The particular parts are discussed in the following subsections.

2.3.1 Steady state

The steady state model is designed by splitting a model into two separate parts: turbocharger part and EGR loop part. The turbocharger part includes the compressor, the intercooler and the turbine. The EGR loop part includes the EGR cooler, the EGR valve and the engine block (See Figure 3). In each part, all pressures and temperatures are expressed explicitly as the functions of the input parameters, the air mass flow ($\dot{m}_a$), the EGR mass flow ($\dot{m}_e$) and the exhaust gas temperature (T_3). The key element of this approach is to iteratively adjust $\dot{m}_a$, $\dot{m}_e$, T_3 from an initial guess so that common variables (p_1, p_3, T_3) are equal for each part.

The explicit functions are derived from implicit equations of the engine components, which are defined in Chapter 3. They are in the form of a polynomial or a rational function and the exact forms of the functions are presented in Chapter 4.

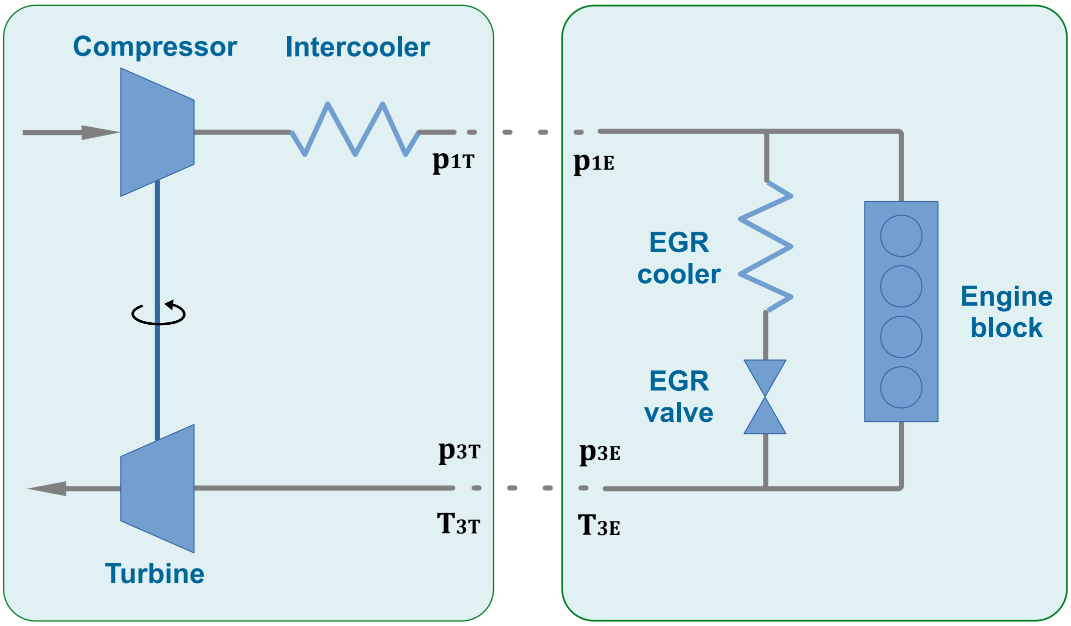

Figure 3 Model Parts. The figure shows splitting the model into two parts. The turbocharger part in on the left-hand side of the picture and the right-hand part is noted the egr-loop part.

The algorithm for solving the model states is depicted on Figure 4. Firstly, the initial values of $\dot{m}_a$, $\dot{m}_e$ and T_3 are chosen reasonably (e.g. zeros or values from the interval of feasibility).

Then the explicit functions in the turbocharger part is evaluated as follows

$$\begin{aligned} \mathbf{x}_T &= f_T\left(\dot{m}_a, \dot{m}_e, T_3, u_T\right), \\ \mathbf{x}_T &= \begin{bmatrix} p_{3T}, & p_{1T}, & T_{1b}, & N_{tc} \end{bmatrix}^T, \\ u_T &= \begin{bmatrix} p_0, & T_0, & \dot{m}_f, & u_{VGT} \end{bmatrix}^T. \end{aligned} \tag{4}$$

The EGR loop part has to be calculated after the turbocharger part due to required value of the temperature after intercooler (T_{1b}). The EGR loop part includes two sets of explicit functions. They represent two different situations. The usual behaviour is when the intake manifold pressure (p_1) is lower than the inlet turbine (p_3) pressure. The following set of equations are evaluated in these scenarios

$$\begin{aligned} \mathbf{x}_E &= f_E\left(\dot{m}_a, \dot{m}_e, T_3, T_{1b}, u_E\right), \\ \mathbf{x}_E &= \begin{bmatrix} p_{3E}, & p_{1E}, & T_{3E}, & T_{3b}, & T_2 \end{bmatrix}^T, \\ u_E &= \begin{bmatrix} N_e, & \dot{m}_f, & u_{egr} \end{bmatrix}^T. \end{aligned} \tag{5}$$

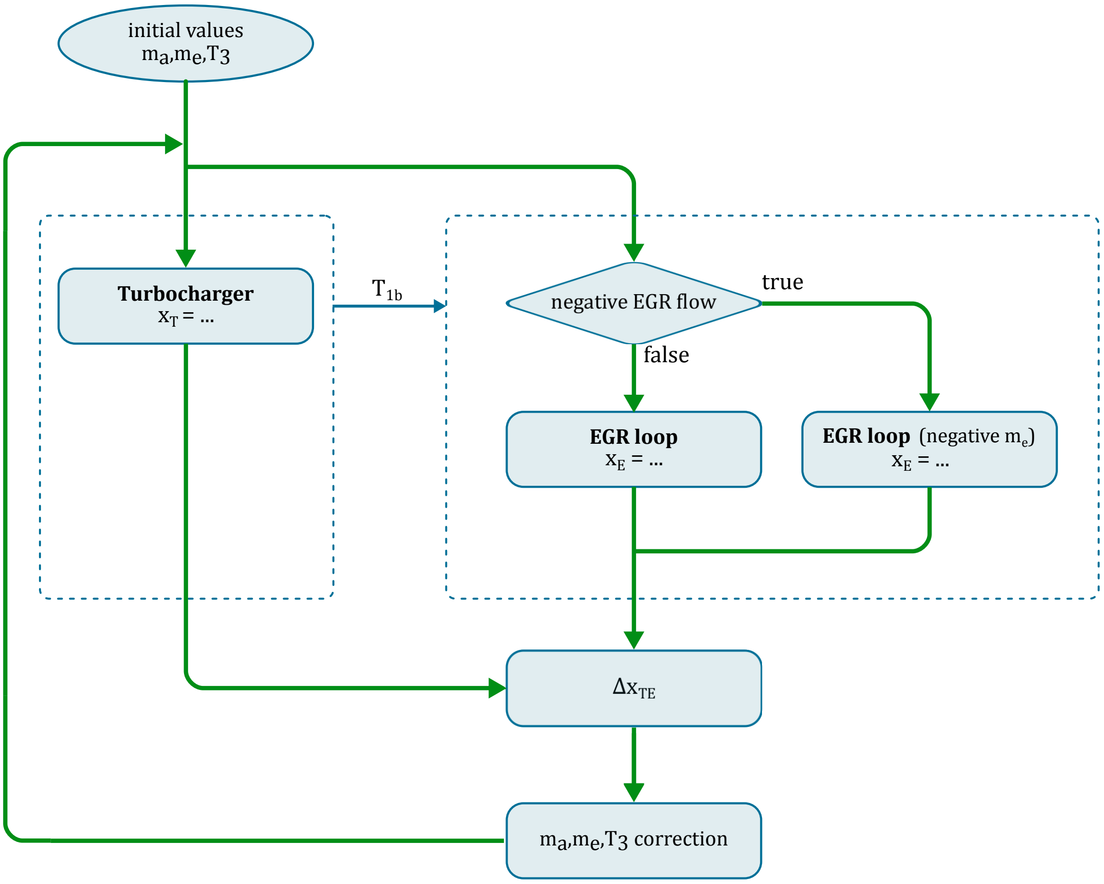

Figure 4 **Steady-state model algorithm**.

However, occasionally the reverse can occur. For example when EGR valve is closed it can happen that p_1 is larger that p_3. Ideally in this situation the EGR mass flow should be zero because of closed valve. But due to the imperfect valve sealing and large pressure difference, the air flows through the EGR feedback in the opposite direction (negative $\dot{m}_e$). The particular situations and their state variables are depicted in Figure 5.

When both parts are evaluated, the difference of the common variables ($x_{TE} = [p_1,\ p_3,\ T_3]^T$) is computed

$$\Delta x_{TE} = \begin{bmatrix} p_{1T} - p_{1E} \\ p_{3T} - p_{3E} \\ T_{3T} - T_{3E} \end{bmatrix}. \tag{6}$$

After that the input parameters are corrected using Newton-Raphson method

$$\begin{bmatrix} \dot{m}_a^* \\ \dot{m}_e^* \\ T_3^* \end{bmatrix} = \begin{bmatrix} \dot{m}_a \\ \dot{m}_e \\ T_3 \end{bmatrix} - S^{-1}\,\Delta x_{TE}, \tag{7}$$

where S is the sensitivity matrix defined as Jacobian of Δx_{TE}:

$$S = \begin{bmatrix} \frac{\partial\,(p_{1T} - p_{1E})}{\partial \dot{m}_a} & \frac{\partial\,(p_{1T} - p_{1E})}{\partial \dot{m}_e} & \frac{\partial\,(p_{1T} - p_{1E})}{\partial T_3} \\ \frac{\partial\,(p_{3T} - p_{3E})}{\partial \dot{m}_a} & \frac{\partial\,(p_{3T} - p_{3E})}{\partial \dot{m}_e} & \frac{\partial\,(p_{3T} - p_{3E})}{\partial T_3} \\ \frac{\partial\,(T_{3T} - T_{3E})}{\partial \dot{m}_a} & \frac{\partial\,(T_{3T} - T_{3E})}{\partial \dot{m}_e} & \frac{\partial\,(T_{3T} - T_{3E})}{\partial T_3} \end{bmatrix}. \tag{8}$$

The corrected input parameters are set as default and whole process is recalculated. The experiments were done where the algorithm was running from different initial values and it turned out that Δx_{TE} converges close to zero in 5 iterations. However, adding more iterations results in better accuracy. The number of loop iteration is fixed to 10 as a trade off between the accuracy and the real-time system requirements.

If Δx_{TE} converge to zero, the solution exists. The prove that the proposed model has always solution is beyond the scope of this thesis. However, the model was thoroughly tested and solution has always been found.

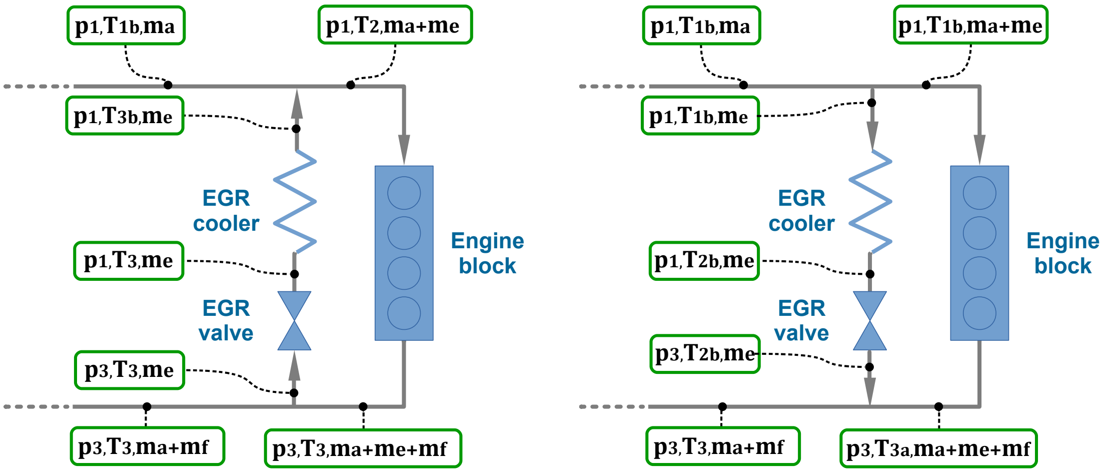

Figure 5 Notation of the states valid for the particular situation. The right diagram shows the EGR loop with negative $\dot{m}_e$. The left one shows the positive EGR flow.

2.3.2 Transient state

Similarly to the DAE model, we omit the energy and mass accumulation in the gas system, so we have only single dynamic variable left - the turbo-speed. The modelling approach of the transient model is based on the requirement for the turbo-speed measurement. Nowadays, the turbo-speed sensor is a common part of the most turbocharged engines. It is usually very accurate therefore eligible for the suggested application.

The proposed approach is first to compute the turbo-speed from the steady-state model ($N_{tc}(\infty)$) as described in the previous subsection. Then use this value to express the turbo-speed as follows

$$\dot{N}_{tc}(t) = \frac{1}{\tau}\left(N_{tc}(\infty) - N_{meas}(t)\right), \tag{9}$$

where N_{meas} is measured turbo-speed and τ is a time constant, that needs to be identified or which can be derived from turbocharger power balance. The next step is to use the current turbo-speed value to evaluate the system states. To do this, the modified steady-state model has to be developed.

The EGR loop part is left unchanged, but the turbocharger part is modified in a way that it uses the turbo-speed as the input variable rather than the state. The explicit functions of the modified turbocharger part can be formally written as follows

$$\begin{aligned} \mathbf{x}_T &= f_T\left(\dot{m}_a, \dot{m}_e, T_3, u_T\right), \\ \mathbf{x}_T &= \begin{bmatrix} p_{3T}, & p_{1T}, & T_{1b} \end{bmatrix}^T, \\ u_T &= \begin{bmatrix} p_0, & T_0, & \dot{m}_f, & u_{VGT}, & N_{tc} \end{bmatrix}^T. \end{aligned} \tag{10}$$

The structure of the algorithm described above is depicted in Figure 6.

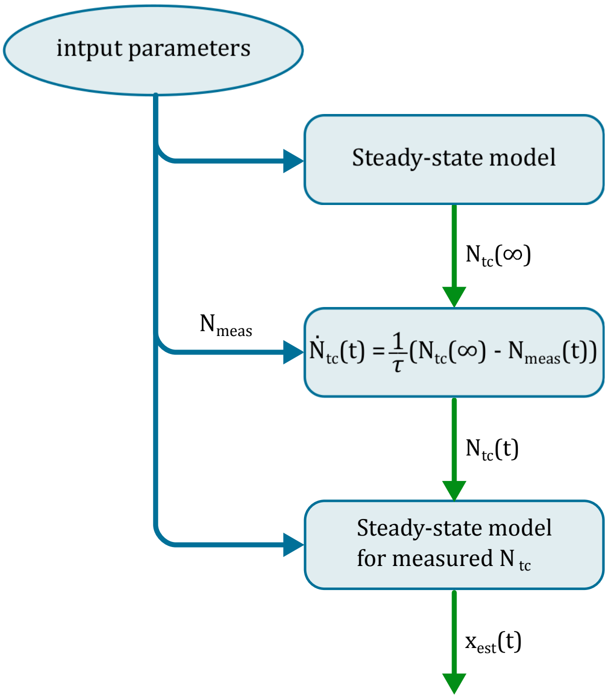

Figure 6 Transient model algorithm.

3 Models of Engine Components

The important physical phenomena of the components are introduced in this chapter. Each one of them is discussed in a separate section. Their approximations are described by implicit equations in a polynomial form [10] [3] [13] [14].

3.1 Compressor mass flow

The compressor compresses the inlet air, in order to deliver the required oxygen mass to the combustion. Compressors' behaviour is typically described by compressor maps. They usually include information about pressures at a compressor inlet and outlet port, compressor rotational speed, compressor efficiency and mass flow.

The model developed in this chapter is inspired by model introduced in [16]. The essential part of these models is introducing dimensionless parameters which can be fitted more accurate. The dimensionless head parameter (Ψ) represents the compressor work and it is defined as follows

$$\Psi = \frac{c_{pa}\, T_i \left[\left(\frac{p_o}{p_i} \right)^{\frac{1-\gamma}{\gamma}} - 1 \right]}{0.5\, U_c}, \tag{11}$$

where p_o, p_i represent the output/input pressure respectively, c_{pa} is the specific heat capacity of air, T_i is the inlet temperature and γ represents the adiabatic index. The compressor tip speed (U_c) is obtained as a function of the compressor diameter (d_c) and its corrected rotational speed (N^*_{tc})

$$U_c = \frac{\pi}{60} d_c\, N^*_{tc}. \tag{12}$$

The corrected rotational speed is obtained as

$$N^*_{tc} = \sqrt{\frac{T_{\text{ref}}}{T_i}}\, N_{tc}. \tag{13}$$

The expression with the exponential function is approximated as a rational function for the control oriented model [10]

$$\left(\frac{p_o}{p_i} \right)^{\frac{1-\gamma}{\gamma}} - 1 \approx \frac{d_0\, p_i + d_1\, p_o}{d_2\, p_i + d_3\, p_o}. \tag{14}$$

The dimensionless flow parameter (Φ) expresses the mass flow rate of the compressor:

$$\Phi = \frac{\dot{m}^*}{\rho_a \, \frac{\pi}{4} \, d_c^2 \, U_c}, \tag{15}$$

where ρ_a is the density of the air and $\dot{m}^*$ is the corrected compressor mass flow that means it is the compressor mass flow corrected with the input pressure and the input temperature

$$\dot{m}^* = \dot{m} \, \frac{p_i}{p_{\text{ref}}} \sqrt{\frac{T_i}{T_{\text{ref}}}}. \tag{16}$$

The square root can also be approximated in interval of plausible temperature by rational function

$$\sqrt{\frac{T_i}{T_{\text{ref}}}} \approx \frac{e_0 \, T_i + e_1}{e_2 \, T_i + 1}, \text{ or } \sqrt{\frac{T_{\text{ref}}}{T_i}} \approx \frac{e_4 \, T_i + e_5}{e_6 \, T_i + 1}, \tag{17}$$

where e_x are identified using nonlinear regression. The dimensionless head parameter is then approximated by the following rational function

$$\Psi = \frac{b_0 \, \Phi + b_1 \, \Phi \, M + b_2 \, M + b_3 \, M^2}{b_4 \, \Phi + b_5 \, \Phi \, M + b_6 \, M}, \tag{18}$$

where b_x are the unknown coefficients and M represents inlet Mach number defined as follows

$$M = \frac{U_c}{\sqrt{\gamma \, R \, T_i}}, \tag{19}$$

where R is the ideal gas constant. The coefficients b_x are identified by using the nonlinear regression identification based on the the compressor-map data.

To complete model, (11) and (18) are put equal The expressions (15) (12) (19) together with corrections (16) (13) and approximations (14) (17) are substituted into that equation. Then the model of mass flow can be expressed by implicit equation (or a rational function that explicitly defines the mass flow):

$$F_{Cm_m} \left(\dot{m}, \, N_{tc}, \, p_i, \, p_o, \, T_i \right) = 0. \tag{20}$$

This process is eligible for evaluating compressor mass flow. However, formulas for evaluating the turbo-speed or the outlet pressure will also be required. Deriving these formulas from (20) leads to unwanted singularities which is not suitable for the final model. But except the singularities, the model is fitted well. This allows us to use the model to create the grid of valid points and use this data to develop a simpler model without the problem with singularities.

The new model structures are used to derive an explicit formula for the turbo-speed and the compressor output pressure. Therefore, the first order dependency on this variables would be the best choice in the model. However, such model structure that would also coincide with data were not successfully designed. Therefore, two models have been created. They are described in the following subsections.

3.1.1 Turbo-speed equation

In the first model, the turbo-speed is modelled as an explicit polynomial function of the mass flow and the pressure ratio. The model was derived empirically and it is given as follows:

$$\begin{aligned} N_{tc}^{*} = & a_0 + a_1\,\dot{m}^{*} + a_2\,\Pi_c + a_3\,\dot{m}^{*\,2} + a_4\,\dot{m}^{*}\,\Pi_c + a_5\,\Pi_c^2 \\ & + a_6\,\dot{m}^{*\,3} + a_7\,\dot{m}^{*\,2}\,\Pi_c + a_8\,\dot{m}^{*}\,\Pi_c^2 + a_9\,\Pi_c^3, \end{aligned} \tag{21}$$

where $\Pi_c = \frac{p_o}{p_i}$. Model described by (21) is identified by nonlinear regression method on the compressor map data. After substituting the expressions in (13) and (16) into (21), the implicit equation expressing the compressor mass flow is formed as

$$F_{Cm_N}\left(\dot{m},\, N_{tc},\, p_i,\, p_o,\, T_i\right) = 0. \tag{22}$$

The comparison of the model with the compressor map is shown in Figure 7.

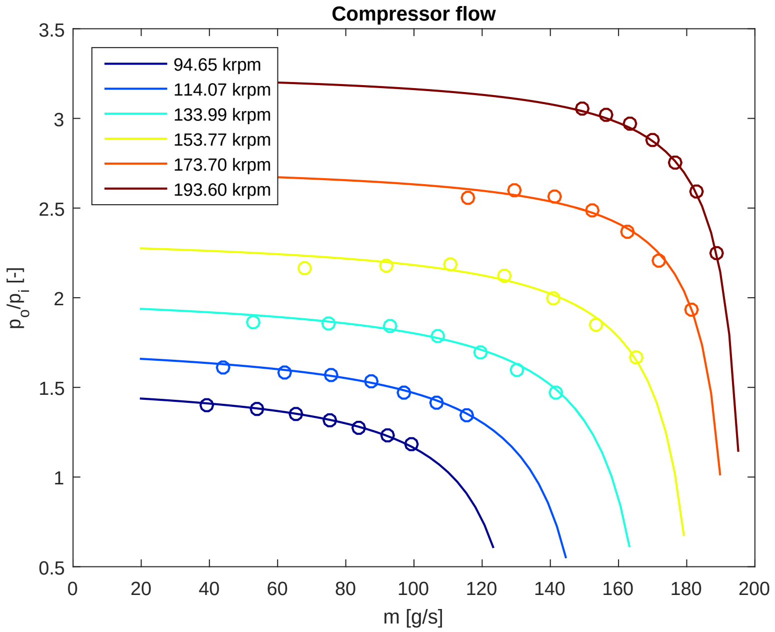

Figure 7 Compressor mass flow. The figure shows comparison of the model (lines) described by (22) and the compressor map (circles). The colors distinguish the particular rotational speeds of the compressor.

The model sensitivity to the perturbation of the model variables is tested. The testing process is as follows. First the solution of the implicit equation that represent feasible steady state of the engine is chosen. Then all model variables are perturbed one by one and deviations from the solution are observed. This gives information about how is the accuracy of the model effected by accuracy of the variables or in other words how much the model deviates from its solution when the input variables are affected by some error. The deviations are represented by changes of the right-hand side of the implicit equation (or zero).

This test was done for various steady-state points and one of them is shown in Figure 8. The figure includes the variable notation according to Figure 1.

The ambient pressure p_0 and the ambient temperature T_0 have usually small error, therefore their perturbation analysis by ±50% is not very relevant. It can be seen that the perturbations of most variables cause only small deviations which is a desirable result. ±50% perturbation region is shown to be sure that no unexpected behaviour of the model will occur.

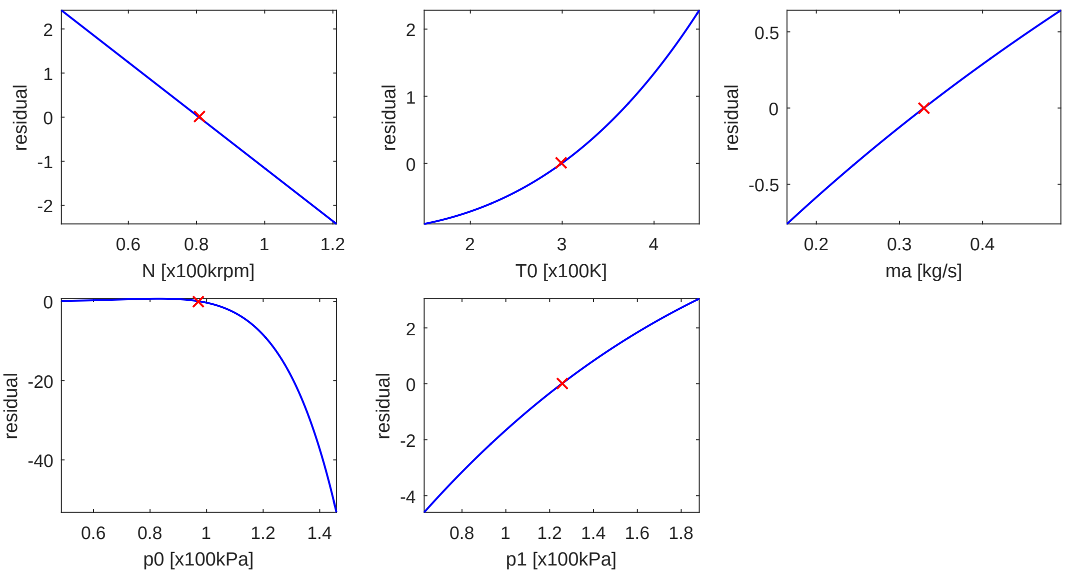

Figure 8 Perturbation of variables. The figure shows perturbations of variables in the particular solution of (22). The y axis represents right-hand side of the implicit equation. The variables are perturbed by ±50%. The red cross represents solution.

3.1.2 Outlet pressure equation

The compressor pressure ratio is modelled as an explicit rational function of the mass flow and the turbo-speed. The model was derived empirically and it is given as follows:

$$\Pi_c = \frac{d_{10} + d_{11}\, N_{tc}^{*} + d_{12}\, \dot{m}^{*} + d_{13}\, N_{tc}^{*}\, \dot{m}^{*} + d_{14}\, N_{tc}^{*2} + d_{15}\, \dot{m}^{*2}}{d_{20} + d_{21}\, N_{tc}^{*} + d_{22}\, \dot{m}^{*} + d_{23}\, N_{tc}^{*}\, \dot{m}^{*} + d_{24}\, N_{tc}^{*2} + d_{25}\, \dot{m}^{*2}}, \tag{23}$$

Model described by (23) is identified by nonlinear regression method on the compressor map data. After substituting the expressions in (13) and (16) into (23), the implicit equation expressing the compressor mass flow is formed as

$$F_{Cm_p}\,(\dot{m},\, N_{tc},\, p_i,\, p_o,\, T_i) = 0. \tag{24}$$

Perturbations of the particular solution of this equation are shown in Figure 8.The figure includes the variable notation according to Figure 1. As mentioned before the ambient pressure p_0 and the ambient temperature T_0 perturbation analysis by ±50% is not very relevant. The perturbations of other variables cause only small deviations which is a desirable result.

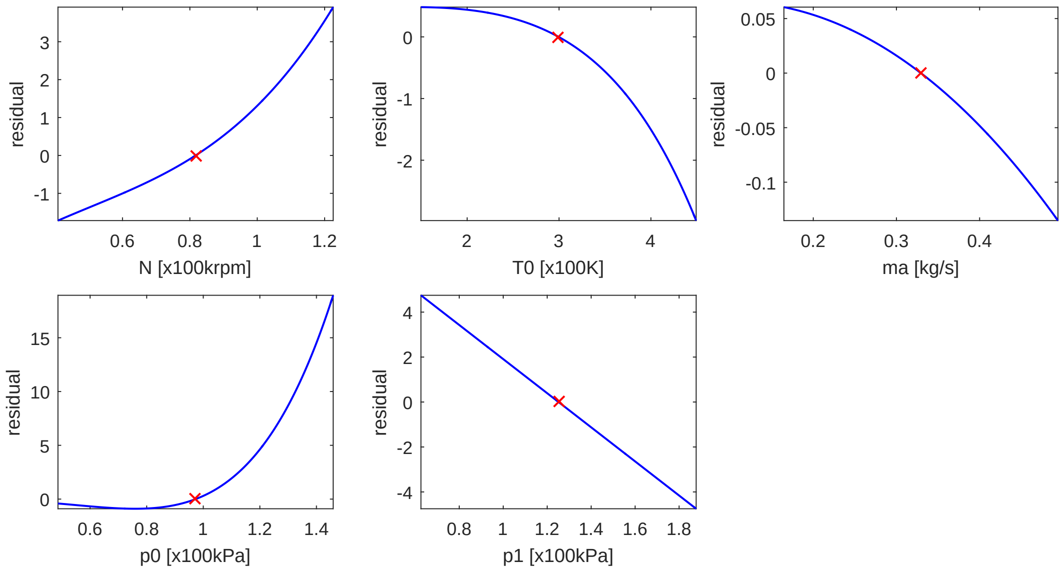

Figure 9 Perturbation of variables. The figure shows perturbations of variables in the particular solution of (24). The y axis represents right-hand side of the implicit equation. The variables are perturbed by ±50%. The red cross represents solution.

3.1.3 Calibration

As mentioned before, the models in subsections above are identified on compressor map data or data generated from a model which was identified on compressor map data. In order to achieve even better accuracy in real implementation, we need to calibrate the model on real data. The calibration of the model represented by (22) is done by replacing p_o in already identified model with following substitution

$$p_o \leftarrow c_0 + c_1\,\dot{m} + c_2\,p_o. \tag{25}$$

After substituting this expression into model are coefficients c_x identified on real data. Analogical calibration is done with model in (24). In this case is the calibration polynomial given as substitution

$$N_{tc} \leftarrow c_3 + c_4\,\dot{m} + c_5\,N_{tc}. \tag{26}$$

The accuracy of the models after the calibration is shown in Figure 10.

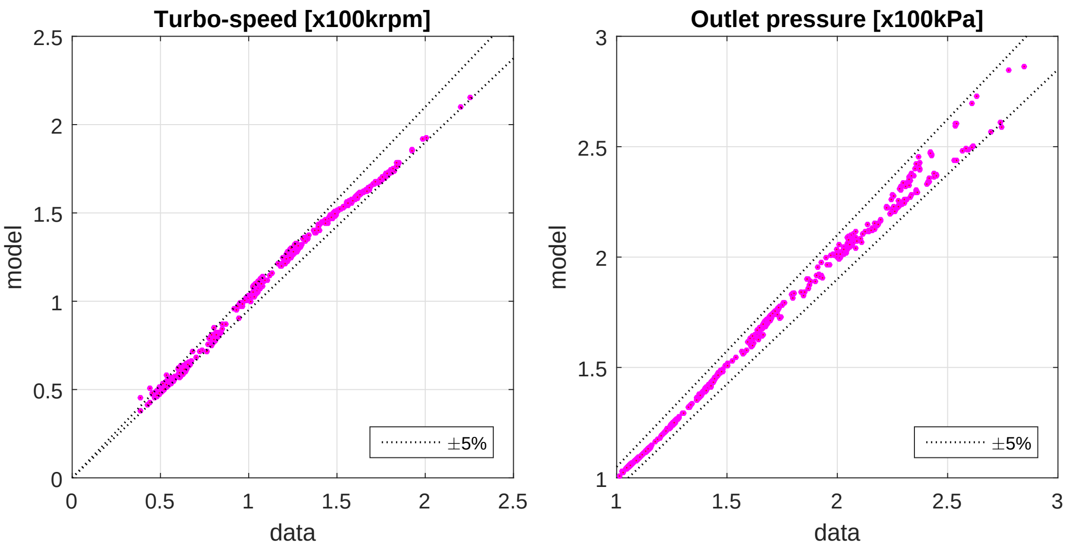

Figure 10 Model calibration. The figure shows the accuracy of the models compared to real data in steady state. On the left is shown the calibrated model of turbo-speed (22) and on the right is the calibrated model of compressor outlet pressure (24). The points of the figure represent steady state data points. 5% error threshold is also shown.

3.2 Compressor outlet temperature

The increase of the temperature at the compressor outlet is bounded by the temperature achieved by the isentropic compression. A device with such behaviour would be an ideal compressor. The temperature in a real physical compressor is additionally increased. Then according to [10] the value of the temperature at the compressor outlet is as follows

$$T_o = \left(1 + \left[\left(\frac{p_o}{p_i}\right)^{\frac{1-\gamma}{\gamma}} - 1\right]\frac{1}{\eta_c}\right) T_i, \tag{27}$$

where γ is the heat capacity ratio.

The expression with the exponential function is approximated as a rational function for the control oriented model [10]

$$\left(\frac{p_o}{p_i}\right)^{\frac{1-\gamma}{\gamma}} - 1 \approx \frac{d_0\, p_i + d_1\, p_o}{d_2\, p_i + d_3\, p_o}. \tag{28}$$

To make the model as simple as possible the efficiency is approximated by a constant in this case ($\eta_c \approx 0.7$). Then the implicit equation for the compressor outlet temperature is obtained by substituting (28) into (29) and can be formally rewritten into following form

$$F_{Ct}\left(p_i, p_o, T_i, T_o\right) = 0. \tag{29}$$

Perturbations of the particular solution of this equation are shown in Figure 11. The figure includes the variable notation according to Figure 1. It can be seen that the perturbations of the variables cause only small deviations which is a desirable result.

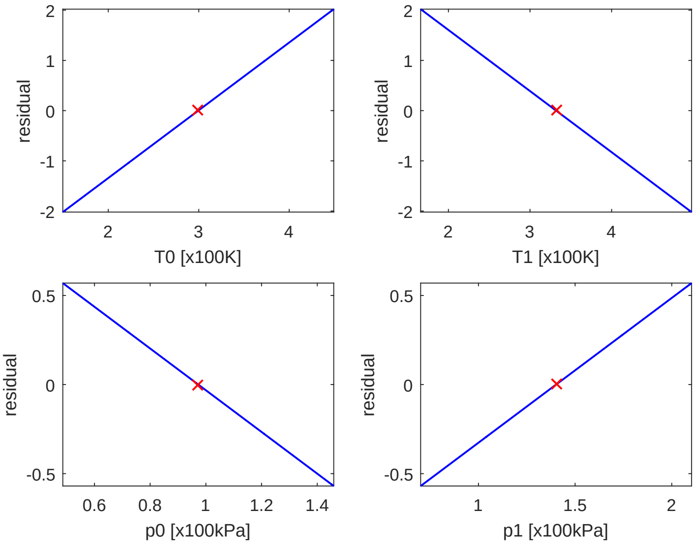

Figure 11 Perturbation of variables. The figure shows perturbations of variables in the particular solution of (29). The y axis represents right-hand side of the implicit equation. The variables are perturbed by ±50%. The red cross represents solution.

3.3 Turbine mass flow

Similarly to the compressor, the turbine behaviour can also be represented by a turbo map. However, the modelling approach differs from the compressor's one. The turbine used in the modelled engine is variable-geometry (or variable-nozzle) turbine, that means an angle of nozzles can be controlled.

The mass flow through the turbine is described by the compressible fluid orifice flow equation. Initial steps described by equations are the same for the turbine and the valve or flow restriction models.

Then the mass flow of the gas through the turbine can be expressed as follows [10]

$$\dot{m}_t = \frac{c_d}{\sqrt{R}} \, \Psi\left(\Pi_t\right) \frac{p_i}{p_{\mathrm{ref}}} \sqrt{\frac{T_{\mathrm{ref}}}{T_i}}, \tag{30}$$

where $\Pi_t = \dfrac{p_i}{p_o}$, c_d is the discharge coefficient, R is the gas constant, p_i, p_o represent the intake and the outtake pressure and T_{ref}, p_{ref} are reference values of the temperature and the pressure. Function ψ is the flow function that can be formulated as ([10])

$$\psi = \Pi_t^{1/\gamma} \sqrt{\frac{2\gamma}{\gamma - 1}\left(1 - \Pi_t^{\frac{1-\gamma}{\gamma}}\right)}, \quad \text{for } \Pi_t \leq \left(\frac{2}{\gamma - 1}\right)^{\frac{1-\gamma}{\gamma}}. \tag{31}$$

The discharge coefficient is defined as follows

$$c_d = \frac{\dot{m}\sqrt{R}\, p_{\mathrm{ref}}}{\psi\, p_i} \sqrt{\frac{T_i}{T_{\mathrm{ref}}}}. \tag{32}$$

The physics-based model described above can be implemented by approximating nonlinearities by rational functions. However, the model introduced here is less complex and furthermore it turned out to be more accurate as well. The proposed model of the turbine mass flow is in the form of the following rational function

$$\dot{m}^* = \left(1 + a_1\, u_{VGT} + a_2\, u_{VGT}^2\right) \frac{d\,\left(\Pi_t - 1\right)}{c\,\Pi_t - 1}. \tag{33}$$

where u_{VGT} represents the controlled position of the turbine nozzles and it ranges from 0 to 1. The function (33) is identified by nonlinear regression on the turbine map data. Furthermore, the model is calibrated by replacing u_{VGT} (map values) with linear or affine function $a\, u_{VGT} + b$ (control system value), where parameters a, b are identified by linear regression on real steady-state data.

$$u_{VGT} \leftarrow a\, u_{VGT} + b. \tag{34}$$

This calibration is necessary because the correspondence between the nozzle position (from the map) and the position of the nozzle actuator used by the engine manufacturer has to be determined. Typically, the turbine map is obtained for a turbine without the actuator and the actuator characteristics represent an additional mapping (in this case a linear gain).

Then the final implicit equation is obtained by substituting corrections (16) (34) into (33) and simple rearranging. It can by formally written as follows

$$F_{Tm}\left(\dot{m}, p_i, p_o, T_i, u_{VGT}\right) = 0. \tag{35}$$

Perturbations of the particular solution of this equation are shown in Figure 12. The figure includes the variable notation according to Figure 1. It can be seen that the perturbation of p_3 causes relatively high deviation from the solution. However in the final model, p_3 will be derived from the implicit equation, in other words it will not be used as an input variable into this implicit equation, therefore this perturbations does not matter. The perturbations of the other variables cause only small deviations which is a desirable result.

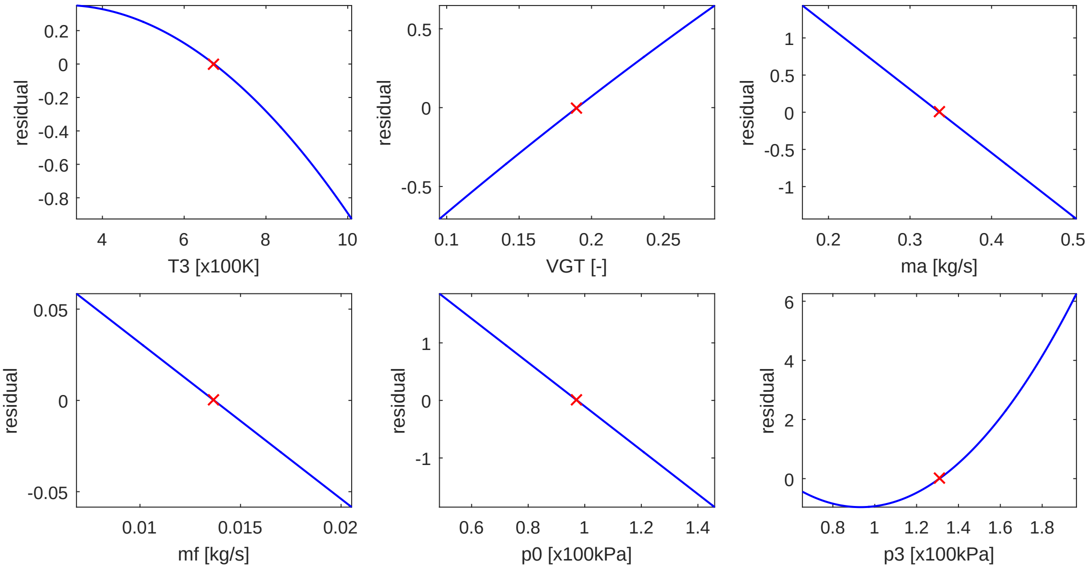

Figure 12 Perturbation of variables. The figure shows perturbations of variables in the particular solution of (35). The y axis represents right-hand side of the implicit equation. The variables are perturbed by ±50%. The red cross represents solution. *VGT* in the figure is identical with u_{VGT} mentioned in text.

3.4 Turbocharger power balance

Exhaust gas causes time derivative of the rotational speed of the turbine which is connected to the compressor by a shaft. The simplest case is taken into account in which no energy accumulation is assumed.

3.4.1 Powers

The change of the turbocharger speed can be found to be proportional to the difference between power produced by the turbine and the power consumed by the compressor. This relation can be expressed by the following formula [17]

$$\frac{dN_{tc}}{dt} = \frac{60}{2\,\pi\, I_{tc}} \left(\frac{P_t}{N_t} - \frac{P_c}{N_c} - M_{fr} \right), \tag{36}$$

where P_c, P_t are the power of the compressor and turbine respectively and M_{fr} is torque that models additional friction losses, but for the purpose of this model it is assumed to be zero.

Furthermore, we focus only on designing the steady state model, because the transient model proposed in this thesis works on different principle as described in (2.3). Therefore the power balance equation (in steady state) can be simplified into the following equation

$$P_c = P_t. \tag{37}$$

The power of the compressor is defined as the ideal (isentropic) compressor power needed for the compression of the mass flow to the given pressure divided by the efficiency (η_c) of the real compressor and it is given as follows [10]

$$P_c = c_p\, \dot{m}_c\, T_{i\,c} \left[\left(\frac{p_{o\,c}}{p_{i\,c}} \right)^{\frac{\gamma_a - 1}{\gamma_a}} - 1 \right] \frac{1}{\eta_c}. \tag{38}$$

The immeasurable variable - compressor efficiency is modelled in next section. The turbine power is defined analogously to the compressor. The power produced by the turbine is defined as [10]

$$P_t = c_p\, \dot{m}_t\, T_{i\,t} \left[1 - \left(\frac{p_{i\,t}}{p_{o\,t}} \right)^{\frac{1-\gamma_e}{\gamma_e}} \right] \eta_t, \tag{39}$$

where turbine efficiency (η_t) is modelled in next section. The exponential function expressions need to be removed since they are not fulfilling the real time model requirements. The following approximation is made for the expression in (38)

$$\left(\frac{p_{oc}}{p_{ic}} \right)^{\frac{1-\gamma_a}{\gamma_a}} - 1 \approx \frac{d_0\, p_{ic} + d_1\, p_{oc}}{d_2\, p_{ic} + d_3\, p_{oc}}. \tag{40}$$

And for (39) is approximation similar

$$1 - \left(\frac{p_{ot}}{p_{it}} \right)^{\frac{1-\gamma_e}{\gamma_e}} \approx \frac{d_4\, p_{it} + d_5\, p_{ot}}{d_6\, p_{it} + d_7\, p_{ot}}. \tag{41}$$

3.4.2 Efficiencies

The following subsections describes approaches to the compressor and turbine efficiency in (38) and (39).

3.4.2.1 Compressor

The difference between an ideal compressor and real realization is expressed by a compressor efficiency. The introduced model is second order polynomial with dependency on the rotational speed and the pressure ratio. The formulation of proposed model is following

$$\eta_c = b_0 + b_1\, N_{tc} + b_2\, \Pi_c + b_3\, N_{tc}\, \Pi_c + b_4\, N_{tc}^2 + b_5\, \Pi_c^2 + b_6\, N_{tc}^2\, \Pi_c + b_7\, N_{tc}\, \Pi_c^2. \tag{42}$$

Parameters b_x are identified on compressor map data using nonlinear regression. The model is inspired by [18] and changed empirically by adding elements with coefficients b_6 and b_7.

3.4.2.2 Turbine

The difference between an ideal turbine and real realization is expressed by compressor efficiency. It mostly depends on u_{VGT}. The introduced model is second order polynomial that also depends on the rotational speed and the pressure ratio. The formulation of the proposed model is following

$$\eta_c = \left(b_0 + b_1\, N_{tc} + b_2\, \Pi_t + b_3\, N_{tc}\, \Pi_t + b_4\, N_{tc}^2 + b_5\, \Pi_t^2\right)\left(c_0 + c_1\, u_{VGT} + c_2\, u_{VGT}^2\right). \tag{43}$$

The model is created empirically taking model in by [18] and adding dependency on u_{VGT}.

3.4.2.3 Combined

It is important to notice that in the final equation of the turbocharger power balance there appears product of compressor efficiency and turbine efficiency. The product of the efficiencies introduced above causes high order of the final equation and the error also increases. Therefore we can model directly their product instead. This approach can lower the error and complexity simultaneously.

The solution for this problem lies in approximation of the product of the efficiencies at once as one polynomial. It proved to be sufficient to model it as a function of u_{VGT} and $\dot{m}_a$. The model that achieved the best results with respect to the complexity has the following formulation

$$\begin{aligned}\eta_c\, \eta_t \approx & a_0 + a_1\, \dot{m}_c + a_2\, u_{VGT} + a_3\, \dot{m}_c^2 + a_4\, \dot{m}_c\, u_{VGT} + a_5\, u_{VGT}^2 \\ & + a_6\, \dot{m}_c^3 + a_7\, \dot{m}_c^2\, u_{VGT} + a_8\, \dot{m}_c\, u_{VGT}^2 + a_9\, u_{VGT}^3,\end{aligned} \tag{44}$$

The polynomial parameters are identified using nonlinear regression. The data for the identification was created by deriving the product from the (37) and evaluating it using real data. The explicit analytic formulation for the product of the efficiencies is given as

$$\eta_c\,\eta_t = \frac{c_{p\,a}\,\dot{m}_c\,T_{i\,c}}{c_{p\,e}\,\dot{m}_t\,T_{i\,t}}\left[\left(\frac{p_{o\,c}}{p_{i\,c}}\right)^{\frac{\gamma_a-1}{\gamma_a}} - 1\right]\left[1 - \left(\frac{p_{i\,t}}{p_{o\,t}}\right)^{\frac{1-\gamma_e}{\gamma_e}}\right]^{-1}. \tag{45}$$

This modelling process is depicted in Figure 13 where the effect of combined η is shown.

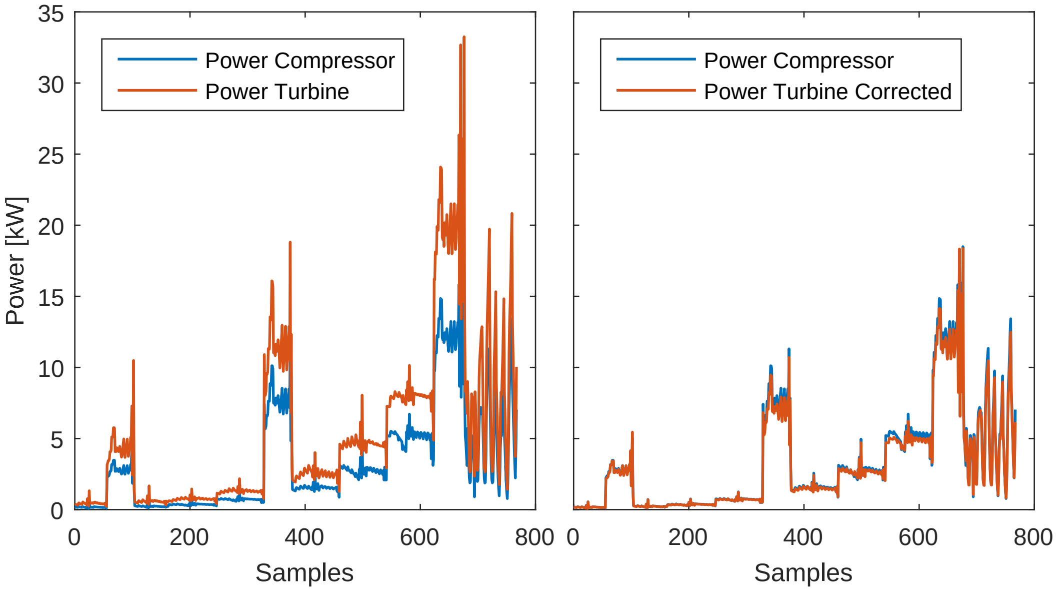

Figure 13 Combined efficiencies. The left plot shows the power of compressor and turbine without taking into account efficiencies (or with the efficiencies equal to 1). The powers with applied combined efficiency is shown in the right plot. It can be seen that (37) is satisfied correctly. The plots show points in steady state.

3.4.3 Conclusion

The final implicit equation modelling the power balance is then written as follows

$$c_{p\,a}\,\dot{m}_c\,T_{i\,c}\,\alpha_1(\Pi_c) - c_{p\,e}\,\dot{m}_t\,T_{i\,t}\,\alpha_2(\Pi_t)\,\alpha_3(\dot{m}_c, u_{VGT}) = 0, \tag{46}$$

where α_1, α_2, are rational functions in (40), (41) respectively and α_3 is approximation polynomial (44).

When the approximations are substituted, the polynomial implicit equation is formally formulated as

$$F_P\left(\dot{m}_c, \dot{m}_t, p_{i\,c}, p_{i\,t}, p_{o\,c}, p_{o\,t}, T_{i\,c}, T_{i\,t}, u_{VGT}\right) = 0. \tag{47}$$

Perturbations of the particular solution of this equation are shown in Figure 14. The figure includes the variable notation according to Figure 1. It can be seen that the perturbations of the variables cause only very small deviations which is a desirable result.

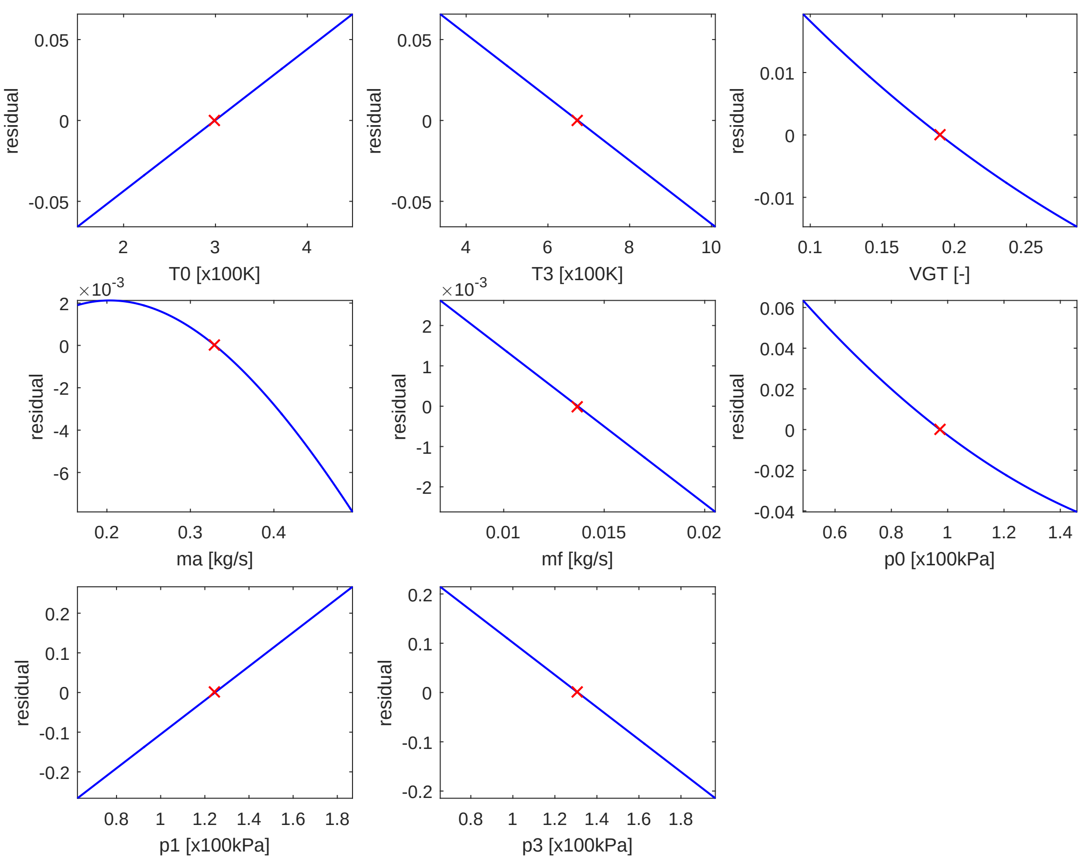

Figure 14 Perturbation of variables. The figure shows perturbations of variables in the particular solution of (47). The y axis represents right-hand side of the implicit equation. The variables are perturbed by ±50%. The red cross represents solution.

3.5 EGR valve mass flow

The valve can be described as a restriction between two reservoirs. The mass flow between these reservoirs is a result of a pressure difference. Only the simple physical formulation is presented due to the goal to develop the control oriented model. We will assume no friction and no inertial effects in the mass flow, also all phenomena are zero dimensional and the system is isolated from any external energy exchange. Furthermore, the pressure in the narrowest point of the valve is equal to the downstream pressure and the temperature of the gas at the outlet matches the inlet gas temperature.

Then the mass flow of the gas through the valve can be expressed as follows [10]

$$\dot{m} = \frac{c_d}{\sqrt{R}}\, \Psi\left(\frac{p_i}{p_o}\right) \frac{p_i}{p_{\mathrm{ref}}} \sqrt{\frac{T_{\mathrm{ref}}}{T_i}}, \tag{48}$$

where c_d is the discharge coefficient, R is the gas constant, p_i, p_o represent the intake and the outtake pressure and T_{ref}, p_{ref} are reference values of the temperature and the pressure. Function Ψ is the flow function that can be formulated as [10]

$$\psi = \left(\frac{p_i}{p_o}\right)^{1/\gamma} \sqrt{\frac{2\gamma}{\gamma-1}\left(1-\left(\frac{p_i}{p_o}\right)^{\frac{1-\gamma}{\gamma}}\right)}, \quad \text{for} \quad \frac{p_i}{p_o} \leq \left(\frac{2}{\gamma-1}\right)^{\frac{1-\gamma}{\gamma}}. \tag{49}$$

To obtain the polynomial of the mass flow the partial models are modelled by rational functions. First the square root of the temperature inlet is approximated by the rational function in the following form

$$\sqrt{\frac{T_{\mathrm{ref}}}{T_i}} \approx \frac{b_0\, T_i + b_1}{b_2\, T_i + b_3}. \tag{50}$$

The flow function is also approximated by the first degree rational function given as follows

$$\psi = \left(\frac{p_i}{p_o}\right)^{1/\gamma} \sqrt{\frac{2\gamma}{\gamma-1}\left(1-\left(\frac{p_i}{p_o}\right)^{\frac{1-\gamma}{\gamma}}\right)} \approx \frac{c_0\, p_i + c_1\, p_o}{c_2\, p_i + c_3\, p_o}. \tag{51}$$

The discharge coefficient depends on the valve opening. It is equal to zero when fully closed and it is entirely opened when its value is maximum. Initially, it is derived from (48) and approximated by the first degree monotonic polynomial defined as

$$c_d = \frac{\dot{m}\, \sqrt{R}\, p_{\mathrm{ref}}}{\psi\, p_i} \sqrt{\frac{T_i}{T_{\mathrm{ref}}}} \approx d_0\, u_{egr}, \tag{52}$$

where u_{egr} lies in interval $\langle 0, 1\rangle$ and express the proportion of the valve throughput.

When the approximations in (50), (51) and (52) are substituted into (48), the rational function empirical model for the valve mass flow is created. The simulations of this formula showed very low dependence on the inlet temperature, hence it was removed. The final formula for approximation of (48) is given as follows

$$\dot{m} = d_1\, u_{egr}\, \frac{c_0\, p_i + c_1\, p_o}{c_2\, p_i + c_3\, p_o}\, p_i, \tag{53}$$

where d_1 aggregates the constants. The model is fit on real data using nonlinear regression. With several basic math operations is this model transformed into model described by the implicit polynomial function with the following form

$$F_{\mathrm{EGRm}}(\dot{m}, p_i, p_o, u_{egr}) = 0. \tag{54}$$

Perturbations of the particular solution of this equation are shown in Figure 15. The figure includes the variable notation according to Figure 1. It can be seen that the perturbations of the variables cause only very small deviations which is a desirable result.

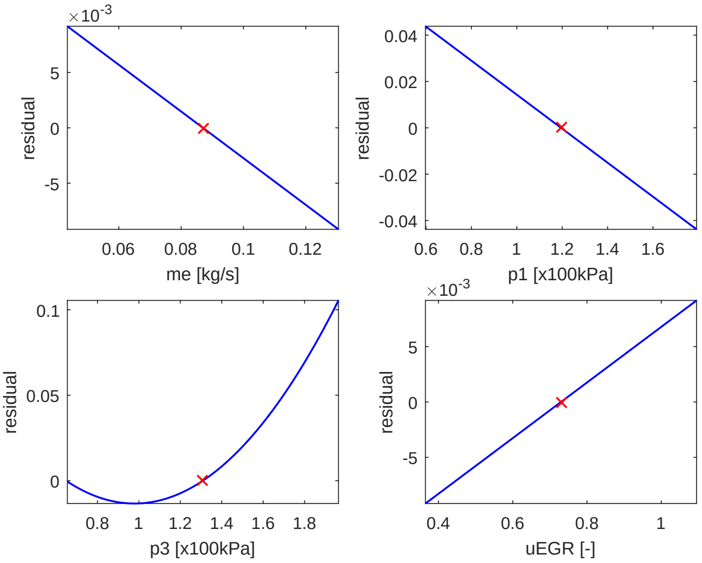

Figure 15 Perturbation of variables. The figure shows perturbations of variables in the particular solution of (54). The y axis represents right-hand side of the implicit equation. The variables are perturbed by ±50%. The red cross represents solution.

3.6 Heat exchanger

There are two heat exchangers in our model. The charge air cooler (CAC) is positioned right after the compressor and the EGR cooler is at the end of the EGR feedback.

Both the CAC and the EGR cooler are modelled as a simple heat exchanger. The pressure restriction of the cooler is not taken into account in neither case, that means the pressures before and after the cooler are considered to be equal.

The energy of the flowing gas decreases with efficiency η [17]. The Efficiency of energy exchange depends on the mass flow of the gas. The output temperature T_o is then given as

$$T_o = T_i + \eta(\dot{m})\ (T_i - T_c)\,, \tag{55}$$

where T_c is the temperature of coolant that surrounds the tube with the flowing gas and it is assumed to be a constant.

The efficiency is approximated by the following rational function

$$\eta \approx \frac{1}{a_0 + a_1\,\dot{m}}, \tag{56}$$

then the model can be rewritten as follows

$$T_o - T_i = \frac{a_2 + a_3\,T_i}{a_0 + a_1\,\dot{m}}, \tag{57}$$

where parameters a_x can be identified on real data using linear regression. The equation can be easily transformed into the following form

$$F_H\left(\dot{m}, T_i, T_o\right) = 0. \tag{58}$$

Perturbations of the particular solution of this equation are shown in Figure 16. The figure includes the variable notation according to Figure 1. It can be seen that the perturbations of the variables cause only very small deviations which is a desirable result.

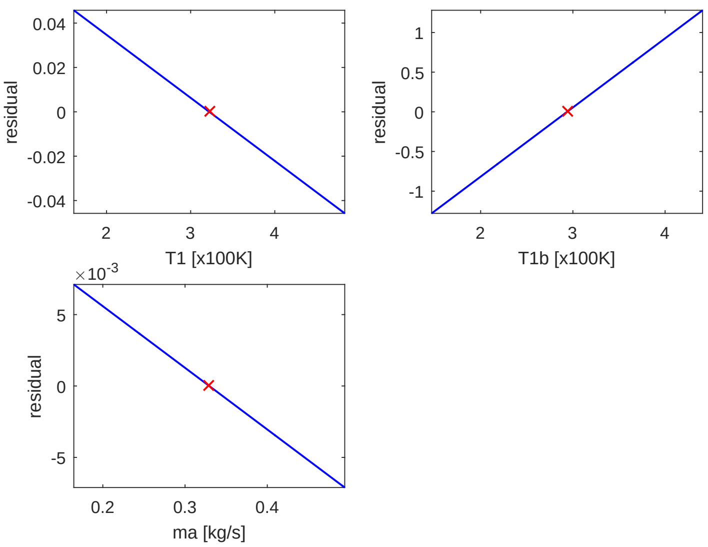

Figure 16 Perturbation of variables. The figure shows perturbations of variables in the particular solution of (58) for CAC. The y axis represents right-hand side of the implicit equation. The variables are perturbed by ±50%. The red cross represents solution.

3.7 Mixing plenum

This chapter is dedicated to the model of mixing of the exhaust gas mass flow from the exhaust gas recirculation ($\dot{m}_e$) with the air mass flow ($\dot{m}_a$) from the CAC cooler. The mix is then flown into the intake manifold. This relation can be expressed by fluid mixing equation [10]

$$c_{p\,exh}\,\dot{m}_e\,T_e + c_{p\,a}\,\dot{m}_a\,T_a = c_{p\,exh}\,\dot{m}_e\,T_o + c_{p\,a}\,\dot{m}_a\,T_o, \tag{59}$$

where T_o is the temperature of the gas in the intake manifold, T_e is exhaust gas temperature and T_a is the temperature of the air from the CAC cooler. The exhaust gas mass flow is expressed by using an EGR rate (r_{egr}). The EGR rate represents relative amount of recirculated exhaust gas to total mass flow rate at the intake manifold

$$\dot{m}_e = \dot{m}_a \frac{r_{egr}}{1 - r_{egr}}, \tag{60}$$

where $r_{egr} \in \langle 0, 1 \rangle$.

By substituting (60) into (59) is created the implicit equation for the intake manifold mass flow

$$F_M\left(\dot{m}_a, r_{egr}, T_a, T_e\right) = 0. \tag{61}$$

Perturbations of the particular solution of this equation are shown in Figure 17. The figure includes the variable notation according to Figure 1. It can be seen that the perturbations of the variables cause only very small deviations which is a desirable result.

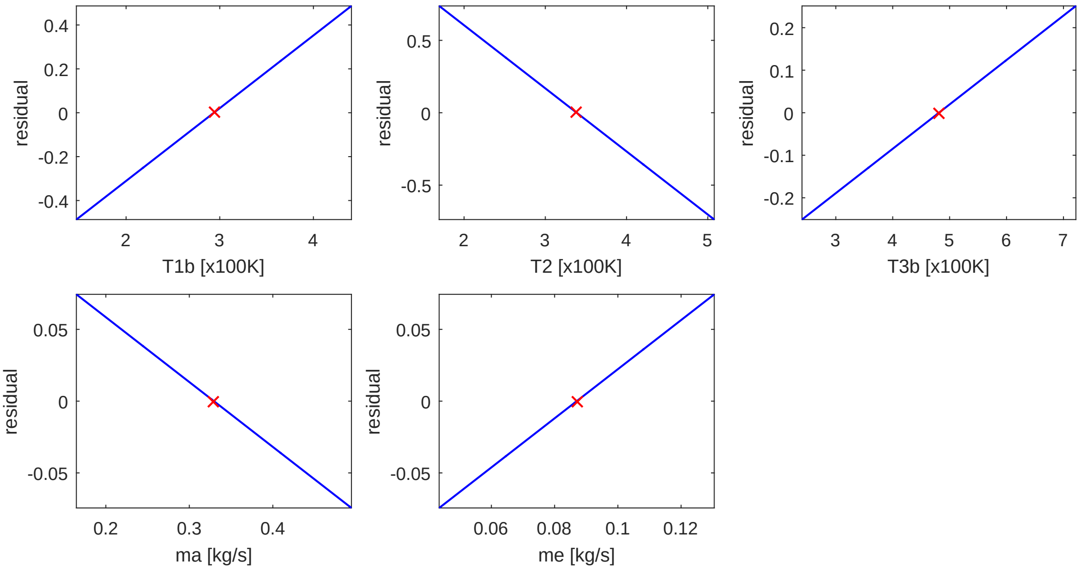

Figure 17 Perturbation of variables. The figure shows perturbations of variables in the particular solution of (61). The y axis represents right-hand side of the implicit equation. The variables are perturbed by ±50%. The red cross represents solution.

3.8 Volumetric efficiency

The in-cylinder process itself is very sophisticated part of the model due to numerous phenomena that occurs there. However, in this text we focus only on its functionality as a part of the air system. Since its volume flow is approximately proportional to rotational speed, it can be approximated as the volumetric pump. The mass flow of the combustion process (also called charge mass flow) can be expressed in form [17]

$$\dot{m} = \rho_i \, \eta_{vol} \, V_d \, \frac{N_e}{120}, \tag{62}$$

where ρ is density of the gas at the intake, V_d is the displaced volume, N_e is engine's rotational speed and η_{vol} is volumetric efficiency that defines difference between the engine and a perfect volumetric device. The density can be substituted with the expression derived from ideal gas law. Then the formula for the mass flow is rewritten into form

$$\dot{m} = \eta_{vol} \, V_d \, \frac{p_i}{R \, T_i} \, \frac{N_e}{120}, \tag{63}$$

where R is ideal gas constant.

In order to include (63) in system's overall model, the volumetric efficiency needs to be modelled first. Precise models of the volumetric efficiency are difficult to obtain, since it is influenced by many effects that are difficult to predict or measure.

However, simple models can be created with the consideration that it mostly depends on engine's rotational speed and the pressure at the intake and the exhaust. The first model considers only the former and it is approximated by linear function

$$\eta_{vol} = a_0 + a_1 \, N_e. \tag{64}$$

The second one is higher order polynomial introduced in [19]. It additionally depends on the intake pressure and its formulation is following

$$\eta_{vol} = a_0 + a_1 \, N_e + a_2 \, N_e^2 + a_3 \, N_e^3 + a_4 \, N_e^4 + b_0 \, p_i. \tag{65}$$

The final simulations showed that the former is sufficient to reach our goals. The final model is created by substituting (64) into (63). The implicit polynomial equation can be developed by few simple operations and it can be formulated as

$$F_V \left(\dot{m}, N_e, p_i, T_i \right) = 0. \tag{66}$$

Perturbations of the particular solution of this equation are shown in Figure 18. The figure includes the variable notation according to Figure 1. It can be seen that the perturbations of the variables cause only very small deviations which is a desirable result.

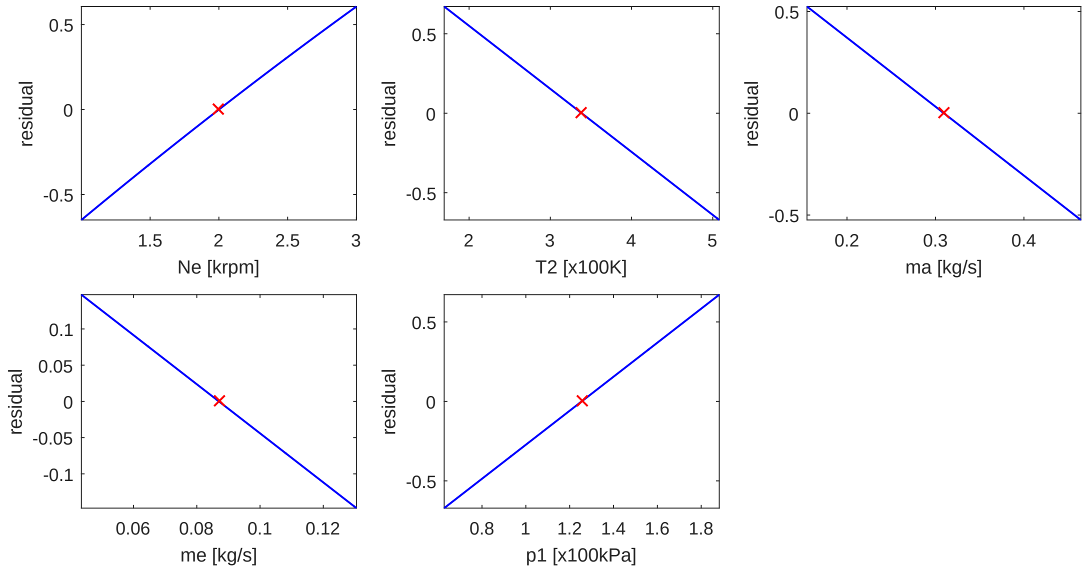

Figure 18 Perturbation of variables. The figure shows perturbations of variables in the particular solution of (66). The y axis represents right-hand side of the implicit equation. The variables are perturbed by ±50%. The red cross represents solution.

3.9 Exhaust temperature

The last equation needed for the engine modelling is the equation that expresses thermodynamic processes in the engine cylinders which are related to the conversion of the fuel energy to the mechanical work. As a modelling variable the ratio between real exhaust temperature (T_o) and exhaust temperature if the process was an isenthalpic process (T_{ise}) is chosen:

$$\eta_{el} = \frac{T_o}{T_{\text{ise}}}. \tag{67}$$

Isenthalpic process can be described by an enthalpy balance equation written as follows

$$P + c_{p\,exh}\,(\dot{m}_a + \dot{m}_e + \dot{m}_f)\,T_{ise} = c_{p\,a}\,(\dot{m}_a + \dot{m}_e)\,T_i + c_{p\,f}\,\dot{m}_f\,T_i + H_l\,\dot{m}_f, \tag{68}$$

where $c_{p\,x}$ are the specific heat capacities (exhaust - exh, air - a, fuel - f), P represents engine's power output and H_l is the lower heat value of the fuel. If T_{ise} is derived from (68) and substituted into (67) then the following model is obtained

$$\eta_{el} = \frac{c_{p\,exh}\,(\dot{m}_a + \dot{m}_e + \dot{m}_f)\,T_o}{c_{p\,a}\,(\dot{m}_a + \dot{m}_e)\,T_i + c_{p\,f}\,\dot{m}_f\,T_i + H_l\,\dot{m}_f - P}. \tag{69}$$

The engine's power is approximated by a multilinear function in form

$$P = a_1\,\dot{m}_f\;\left(a_2 + a_3\,(\dot{m}_a + \dot{m}_e) + a_4\,N_e\right), \tag{70}$$

where the parameters a_x are identified on real dataset using the nonlinear regression.

The ratio in (67) depends on the rotational speed of the engine and the mass flow of the air and fuel. An empirical model that describes this dependence can be created. The best results were achieved with the rational function in the following form

$$\eta_{el} = \frac{b_0 + b_1\,(\dot{m}_a + \dot{m}_e) + b_2\,\dot{m}_f + b_3\,N_e}{b_4 + b_5\,(\dot{m}_a + \dot{m}_e) + b_6\,\dot{m}_f + b_7\,N_e}. \tag{71}$$

where the parameters b_x are identified on data created by evaluating (69).

The final model is developed by comparing (69) and (71) and substituting (70) into this comparison. The model can be generally written as follows

$$F_{Et}\,(\dot{m}_a, \dot{m}_e, \dot{m}_f, N_e, T_i, T_o) = 0. \tag{72}$$

Perturbations of the particular solution of this equation are shown in Figure 19. The figure includes the variable notation according to Figure 1. It can be seen that perturbing T_3 causes high deviations. However in the final model, this variable will be derived from this implicit equations, therefore it will not be an input variable. Perturbing $\dot{m}_f$ also causes substantial inaccuracy, but it is considered to be precisely measured variable since it plays important role in the engine combustion. The perturbations of the other variables cause small deviations which is a desirable result.

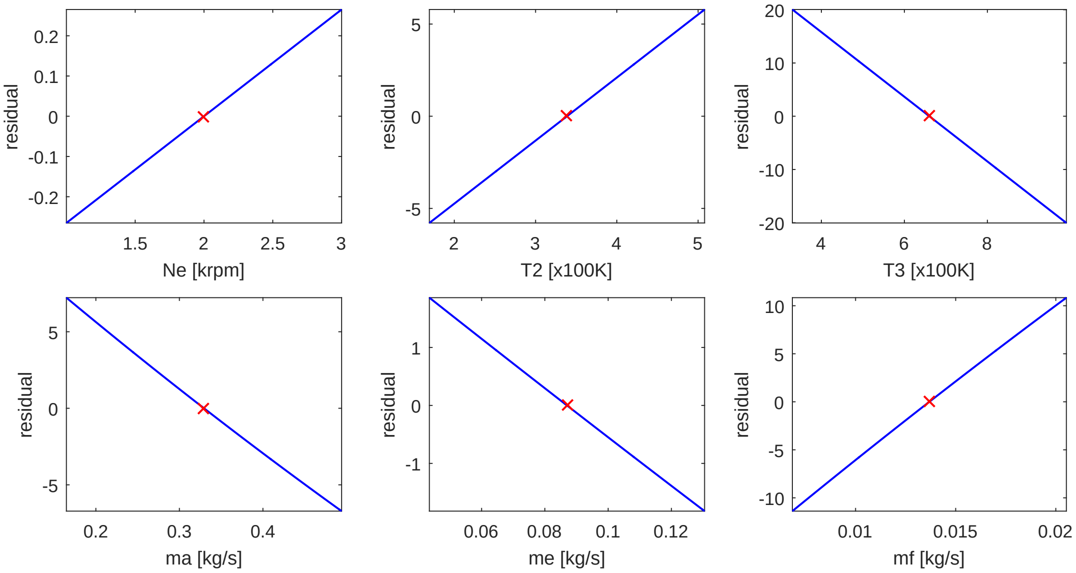

Figure 19 Perturbation of variables. The figure shows perturbations of variables in the particular solution of (72). The y axis represents right-hand side of the implicit equation. The variables are perturbed by ±50%. The red cross represents solution.

4 Engine Model Implementation

This chapter is focused on the detailed description of the Quasi-Explicit model, since it is the one that satisfies the requirements the most. The transient data are not available for this engine, therefore the ODE model is also described in order to create physical-based comparison for the transient data.

The chapter creates a link between previous chapters as it introduces the process of transforming components' implicit equations into Quasi-Explicit model introduced in (2.3). The particular models are defined. After that they are simulated and compared to data.

4.1 ODE Model

The model is introduced in (2.1). The intake manifold receiver is described by the following differential equation for its pressure

$$\dot{p}_1(t) = \frac{R\,T_2}{V_i}\left(\dot{m}_a + \dot{m}_e - \dot{m}_{in}\right), \tag{73}$$

where $\dot{m}_{in}$ represents the mass flow at the engine intake and V_i is the intake manifold volume.

The intake manifold temperature (T_2) is derived from the implicit equation for the mixing plenum (61). The EGR cooler outlet temperature (T_{3b}) which is needed to explicitly evaluate T_2 is derived from the heat-exchanger implicit equation (58). The air mass flow ($\dot{m}_a$) is derived from the compressor mass flow implicit equation (20) and the EGR flow ($\dot{m}_e$) is derived from the EGR valve implicit equation (54). The formula for $\dot{m}_{in}$ is derived from the volumetric efficiency implicit equation (66).

The next differential equation represents the pressure in the exhaust manifold:

$$\dot{p}_3(t) = \frac{R\,\gamma_e}{V_e}\left(T_{3,in}\,\dot{m}_{exh} - T_3(t)\left(\dot{m}_t + \dot{m}_e\right)\right), \tag{74}$$

where $\dot{m}_{exh} = \dot{m}_{in}+\dot{m}_f$ represents the mass flow at the engine exhaust, V_e is the exhaust manifold volume and $T_{3,in}$ represent the temperature and the exhaust manifold inlet.

The formula for $T_{3,in}$ is derived from the exhaust temperature implicit equation (72). The turbine mass flow $\dot{m}_t$ is derived from the implicit equation for the turbine (35).

The exhaust manifold is also described by the differential equation for exhaust manifold temperature T_3:

$$\dot{T}_3(t) = \frac{R\,T_3(t)}{V_e\,c_{ve}\,p_3(t)}\left(c_{pe}\,T_{3,in}\,\dot{m}_{exh} - c_{pe}\,T_3(t)\,(\dot{m}_t + \dot{m}_e) - c_{ve}\,T_3(t)\,(\dot{m}_{exh} - \dot{m}_t - \dot{m}_e)\right). \tag{75}$$

The last differential equation expresses dynamics of turbo-speed and it is a simplified version of (36):

$$\dot{N}_{tc}(t) = \frac{30}{\pi\,I_{tc}}\left(\frac{P_t - P_c}{N_{tc}(t)}\right), \tag{76}$$

where powers P_t, P_c are expressed analogically to (38) and (39), and the efficiencies are chosen to be constants to keep the complexity low.

These differential equations are then solved using *Mathworks MATLAB ode23s* solver. The model is stiff, however none of the ODE solvers for stiff systems in *MATLAB* are able to robustly solve this model. The solvers give good results only when the initial conditions and the inputs are chosen wisely. The speed of solver computation highly depends on the change of the state variables due to a variable sampling step. The accuracy of the successful simulations cannot be evaluated, because real transient data are not available. Because of all these issues, the model is not suitable for real-time implementation.

4.2 Quasi-Explicit Model

As mentioned in (2.3) the algorithm of the Quasi-Explicit model evaluates steady-state then compute dynamics of the turbo-speed and use this value to reevaluate steady-state model, hence propagate the dynamics into other states. This section describes the steady-state models included in the transient model algorithm in detail. The model is also simulated and the accuracy is discussed.

4.2.1 Steady state

The steady-state model is divided into two parts: the turbocharger part defined by equations

$$\begin{aligned}
\mathbf{x}_T &= f_T\left(\dot{m}_a, \dot{m}_e, T_3, u_T\right), \\
\mathbf{x}_T &= \begin{bmatrix} p_{3T}, & p_{1T}, & T_{1b}, & N_{tc} \end{bmatrix}^T, \\
u_T &= \begin{bmatrix} p_0, & T_0, & \dot{m}_f, & u_{VGT} \end{bmatrix}^T,
\end{aligned} \tag{77}$$

and the EGR loop part:

$$
\begin{aligned}
\mathbf{x}_E &= f_E\left(\dot{m}_a, \dot{m}_e, T_3, T_{1b}, u_E\right), \\
\mathbf{x}_E &= \begin{bmatrix} p_{3E}, & p_{1E}, & T_{3E}, & T_{3b}, & T_2 \end{bmatrix}^T, \\
u_E &= \begin{bmatrix} N_e, & \dot{m}_f, & u_{egr} \end{bmatrix}^T. \qquad (78)
\end{aligned}
$$

These expressions are obtained as follows.

4.2.1.1 Turbocharger

The first explicit equation expresses the pressure before turbine (p_{3T}). It is derived from the implicit equation for the turbine mass flow (35). However, the pressure in this implicit equation is in the second order, therefore the quadratic equation have to be solved and the maximum solution has to be chosen.

The function for the compressor outlet pressure (p_{1T}) is derived from the turbocharger power balance implicit equation (47).

The formula for the CAC outlet temperature (T_{1b}) is derived from the heat-exchanger implicit equation (58).

The last function is for the turbo-speed (N_{tc}) and it is derived from the compressor mass flow implicit equation (20).

The overview of these derivations is shown in Table 3.

Derived variable	Source equation name	Source equation reference
p_{3T}	turbine mass flow	(35)
p_{1T}	turbocharger power balance	(47)
T_{1b}	heat-exchanger	(58)
N_{tc}	compressor mass flow	(20)

Table 3 Explicit functions overview. The table shows which variables (Derived variable) are derived from which implicit functions (Source equation) in order to get explicit formulas for that variables.

The formal definitions of the mentioned functions:

$$p_{3T} = f_{T,p_3}\left(\dot{m}_a, \dot{m}_f, p_0, T_3, u_{VGT}\right), \qquad (79)$$

$$p_{1T} = f_{T,p_1}\left(\dot{m}_a, \dot{m}_f, p_0, p_{3T}, T_0, T_3, u_{VGT}\right), \qquad (80)$$

$$T_{1b} = f_{T,T_{1b}}\left(\dot{m}_a, p_0, p_{1T}, T_0\right), \qquad (81)$$

$$N_{tc} = f_{T,N_{tc}}\left(\dot{m}_a, p_0, p_{1T}, T_0\right). \qquad (82)$$

4.2.1.2 EGR loop

There are two sets of equations for the EGR loop part. The sets are switched according to the EGR mass flow direction.

Positive EGR mass flow

The first explicit equation expresses the intake manifold temperature (T_2). It is derived from the implicit equation for the mixing plenum (61).

The function for the turbine inlet temperature (T_{3E}) is derived from the exhaust temperature implicit equation (72).

The formula for the EGR cooler outlet temperature (T_{3b}) is derived from the heat-exchanger implicit equation (58).

The function for the compressor outlet pressure (p_{1E}) is derived from the volumetric efficiency implicit equation (66).

The last function is for the turbine inlet pressure (p_{3E}) and it is derived from the EGR valve implicit equation (54).

The overview of these derivations is shown in Table 4.

Derived variable	Source equation name	Source equation reference
T_2	mixing plenum	(61)
T_{3E}	exhaust temperature	(72)
T_{3b}	heat-exchanger	(58)
p_{1E}	volumetric efficiency	(66)
p_{3E}	EGR valve mass flow	(54)

Table 4 Explicit functions overview. The table shows which variables (Derived variable) are derived from which implicit functions (Source equation) in order to get explicit formulas for that variables.

The formal definitions:

$$T_2 = f_{E,T_2}\left(\dot{m}_a, \dot{m}_e, \dot{m}_f, T_{1b}, N_e\right), \tag{83}$$

$$T_{3E} = f_{E,T_3}\left(\dot{m}_a, \dot{m}_e, \dot{m}_f, T_2, N_e\right), \tag{84}$$

$$T_{3b} = f_{E,T_{3b}}\left(\dot{m}_e, T_{3E}\right), \tag{85}$$

$$p_{1E} = f_{E,p_1}\left(\dot{m}_a, \dot{m}_e, T_2, N_e\right), \tag{86}$$

$$p_{3E} = f_{E,p_3}\left(\dot{m}_e, p_{1E}, u_{egr}\right). \tag{87}$$

Negative EGR mass flow

The first function expresses the compressor outlet pressure (p_{1E}). It is derived from the volumetric efficiency implicit equation (66).

The explicit equation for the intake manifold temperature (T_{2b}) is derived from the heat-exchanger implicit equation (58).

The function for the turbine inlet pressure (p_{3E}) is derived from the EGR valve implicit equation (54).

The formula for the EGR cooler outlet temperature (T_{3a}) is derived from the exhaust temperature implicit equation (72).

The last function is for the turbine inlet temperature (T_{3E}) and it is derived from the implicit equation for the mixing plenum (61).

The overview of these derivations is shown in Table 5.

Derived variable	Source equation name	Source equation reference
T_{3a}	exhaust temperature	(72)
T_{3E}	mixing plenum	(61)
T_{2b}	heat-exchanger	(58)
p_{1E}	volumetric efficiency	(66)
p_{3E}	EGR valve mass flow	(54)

Table 5 Explicit functions overview. The table shows which variables (Derived variable) are derived from which implicit functions (Source equation) in order to get explicit formulas for that variables.

The formal definitions:

$$p_{1E} = f_{E,p_1}\left(\dot{m}_a, \dot{m}_e, T_2, N_e\right), \tag{88}$$

$$T_{2b} = f_{E,T_{2b}}\left(\dot{m}_a, \dot{m}_e, \dot{m}_f, T_{1b}, N_e\right), \tag{89}$$

$$p_{3E} = f_{E,p_3}\left(\dot{m}_e, p_{1E}, u_{egr}\right), \tag{90}$$

$$T_{3b} = f_{E,T_{3b}}\left(\dot{m}_e, T_{3E}\right), \tag{91}$$

$$T_{3E} = f_{E,T_3}\left(\dot{m}_a, \dot{m}_e, \dot{m}_f, T_2, N_e\right). \tag{92}$$

The comparison of the complete model to the steady-state real data is shown on Figure 20.

4.2.2 Steady state with measured turbo-speed

The structure of the transient model was introduced in (2.3.2). This section describes steady-state part used in last part of transient model algorithm. The EGR loop part of this steady-state model is the same as described in refsubsec:ss.

The turbocharger part for measured turbo-speed is similar to the model defined by eqs. (79) to (82). The difference is that the equations for the turbo-speed is omitted and the formula for the compressor outlet pressure (p_{1T}) is derived from the compressor mass flow implicit equation (24).

The overview of these derivations is shown in Table 6.

Derived variable	**Source equation name**	**Source equation reference**
p_{3T}	turbine mass flow	(35)
p_{1T}	compressor mass flow	(24)
T_{1b}	heat-exchanger	(58)

Table 6 Explicit functions overview. The table shows which variables (Derived variable) are derived from which implicit functions (Source equation) in order to get explicit formulas for that variables.

Then the turbocharger part for measured turbo-speed is given as follows

$$p_{3T} = f_{T,p_3}\left(\dot{m}_a, \dot{m}_f, p_0, T_3, u_{VGT}\right), \tag{93}$$

$$p_{1T} = f_{T,p_1}\left(\dot{m}_a, N_{tc}, p_0, T_0\right), \tag{94}$$

$$T_{1b} = f_{T,T_{1b}}\left(\dot{m}_a, p_0, p_{1T}, T_0\right). \tag{95}$$

4.3 Model fit

This section is devoted to simulating and comparing the models described earlier in this chapter. Since only the steady-state data are available, the transient experiments can not be compared to real data. However, the results are shown and compared between models.

4.3.1 Steady state

The steady-state simulations were done for both steady-state models included in the Quasi-Explicit model algorithm (with and without measured turbo-speed where the latter determines the accuracy of the transient-state model in steady-state). The results are shown in Figure 20. The figure also shows evaluations of the accuracy of the models using Normalized Root Mean Squared Error (NRMSE). NRSME is frequently used to measure the differences between prediction and real data.

It can be seen that model with the measured turbo-speed is more accurate. Accuracy of the steady-state models is further described in Figure 21 and Figure 22, which show the errors explicitly together with ±10% deviation region.

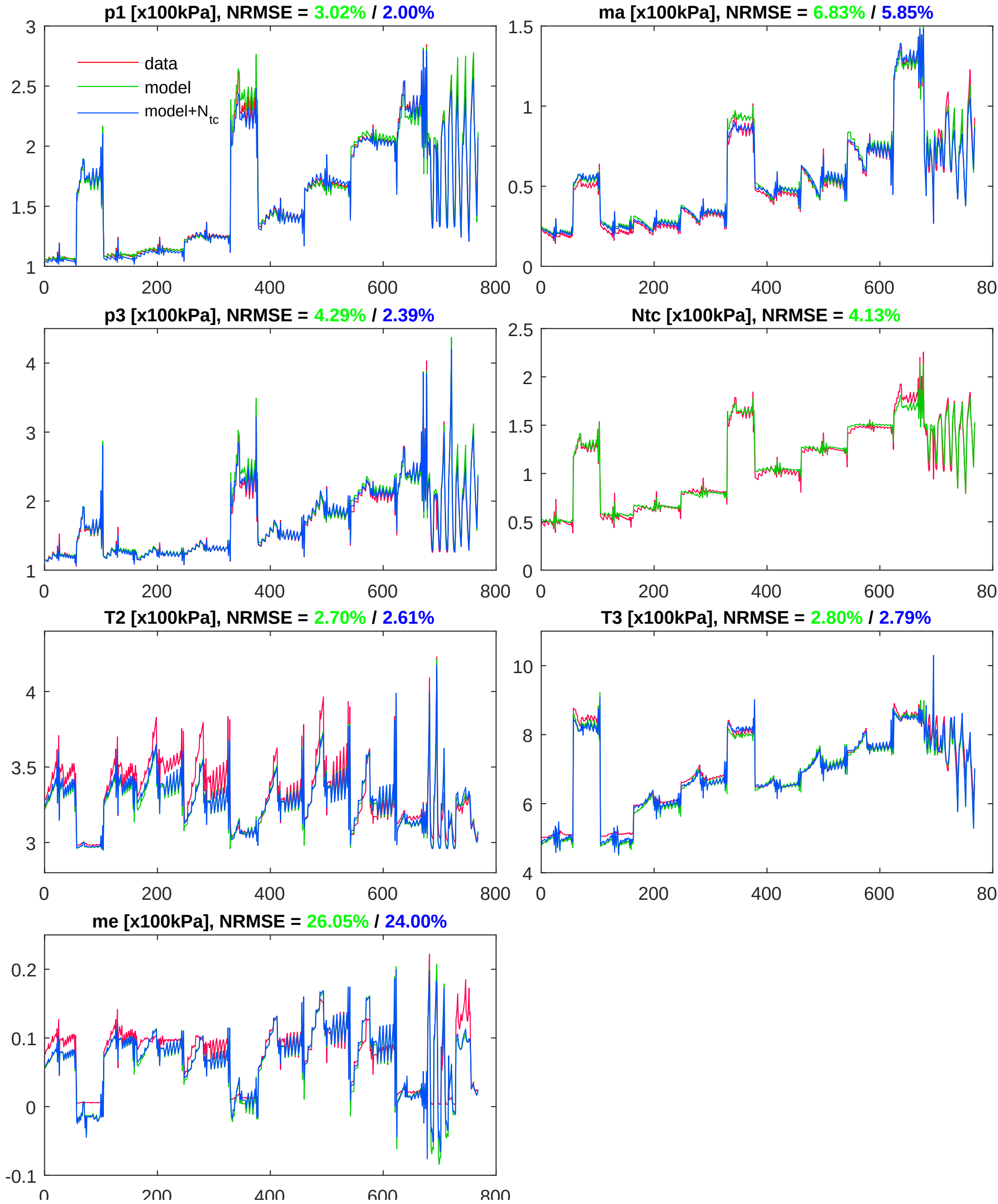

Figure 20 Steady state simulation. Steady states of the real system compared to the model are displayed on the figure. Two steady-state models are simulated: with (4.2.2) and without (4.2.1) measured turbo-speed (N_{tc}). The latter determines accuracy of the transient model in steady-state. Each variable is shown in separate plot window and its NRMSE values appears in its title.

There can be seen inaccuracies in the estimation of the EGR mass flow. However, this is not the error of the model, but the imprecision of the measured data. The provided EGR mass flow data were obtained by measuring CO_2 concentration in the intake manifold. The problem is that the intake manifold has relevant volume, so the distribution of CO_2 is not homogeneous. Therefore the placement of the sensor is very important and the measured values highly depends on it. The provided data seem to have this issue since the model was successfully tested for different engines. The second problem with this measurement process is that the negative EGR flow cannot be measured. It is always evaluated as zero.

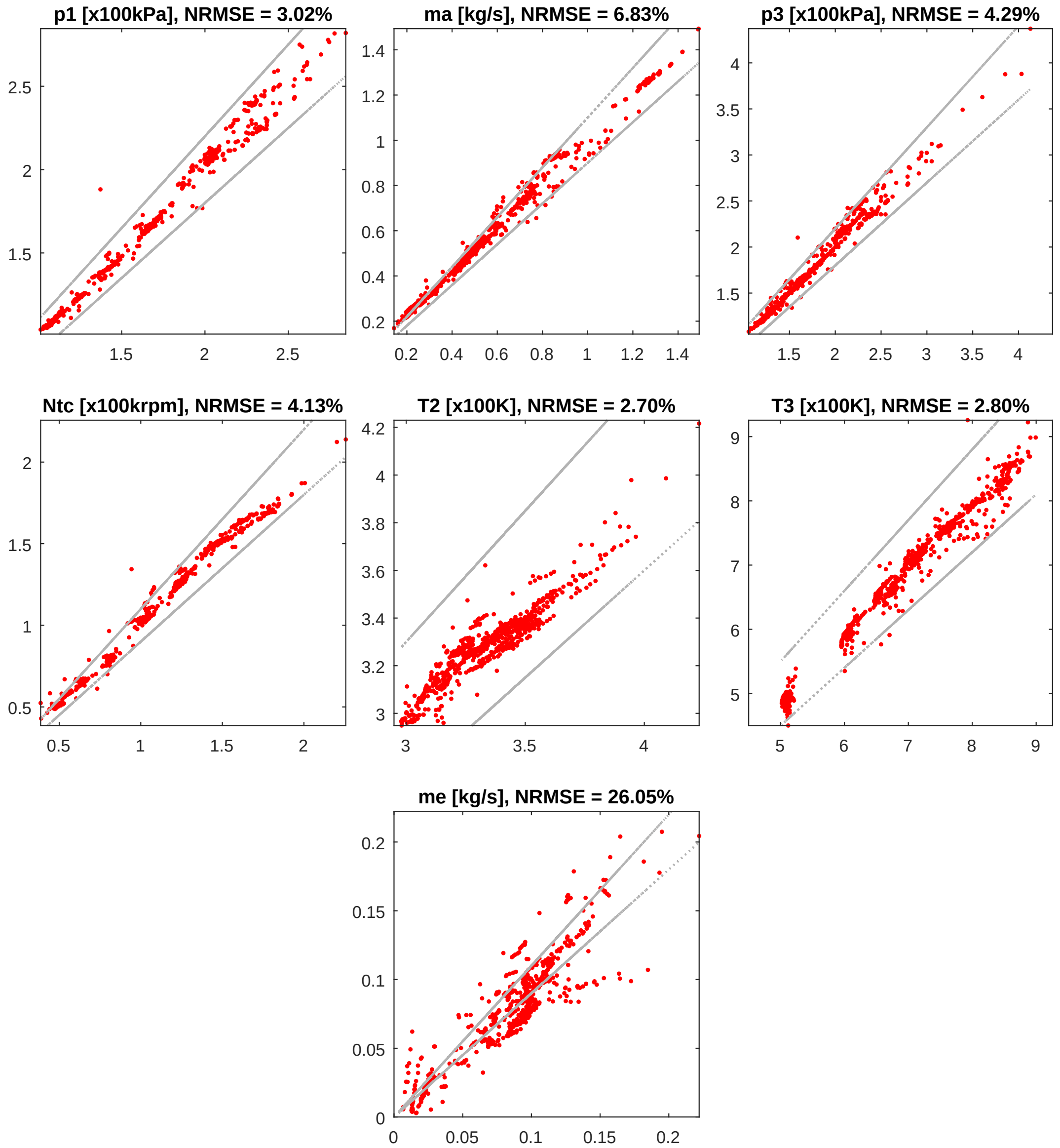

Figure 21 Steady-state solution errors. The figure shows deviations of the results from real data. On the x axis are real data and y axis represents steady-state solutions of model. The points express relation between real and solved data for particular time samples. Dotted lines represent boundaries of ±10% deviation region.

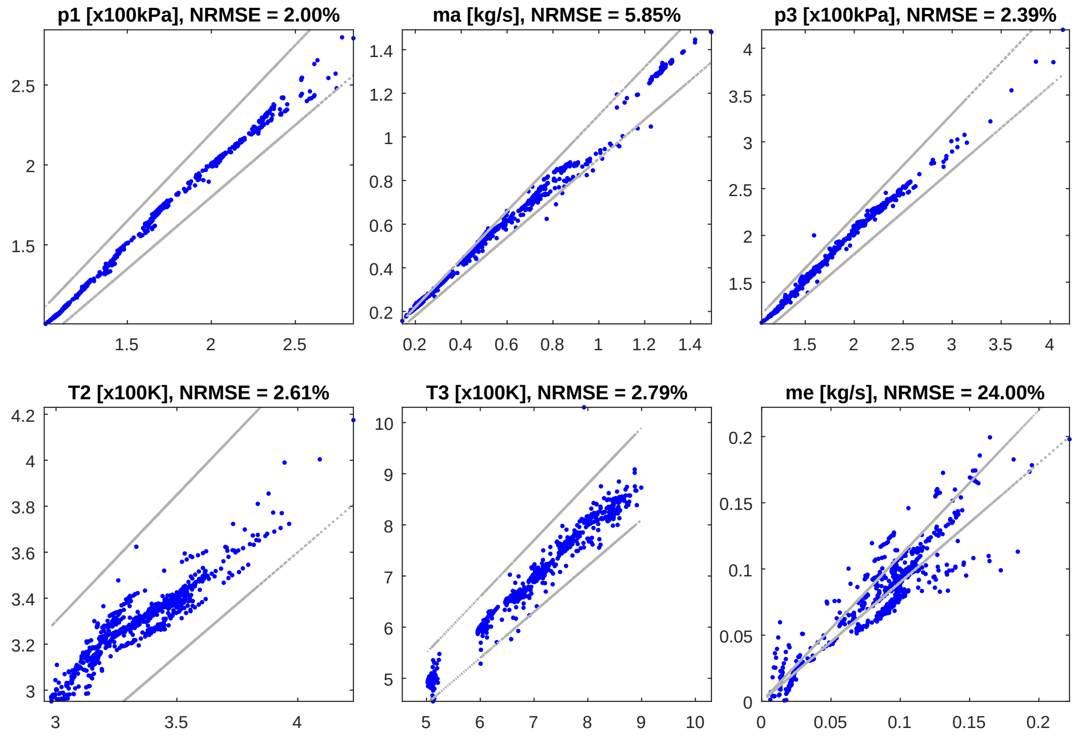

Figure 22 Steady-state solution errors for model with measured turbo-speed. The model showed in the figure determines the accuracy of the transient-state model in steadstate (4.2.2). The figure shows deviations of the results from real data. On the x axis are real data and y axis represents steady-state solutions of model. The points express relation between real and solved data for particular time samples. Dotted lines represent boundaries of ±10% deviation region.

4.3.2 Transient state

The accuracy of the transient-state model in steady-state can be determined by the accuracy of the steady-state model with measured N_{tc}, which error compared to real data is shown in Figure 22.

The demonstration of the transient-state model algorithm functioning is shown in Figure 23. The figure shows responses of all steps of the transient model algorithm: steady-state estimation (red), transient turbo-speed (green) and transient-state estimation (blue). The simulations show bias between steady-state and transient-state which is caused by different accuracy of the models.

The experiment is done by choosing the steady-state point and then perturbing the inputs in time. The models is initiated from zero initial conditions. u_{VGT} is increased by 20% in time interval $t_{VGT} = \langle 1s, 1.6s \rangle$ and u_{egr} is decreased by 20% in time interval $t_{uEGR} = \langle 2s, 3s \rangle$. The model's sampling time is $T_s = 0.05$s.

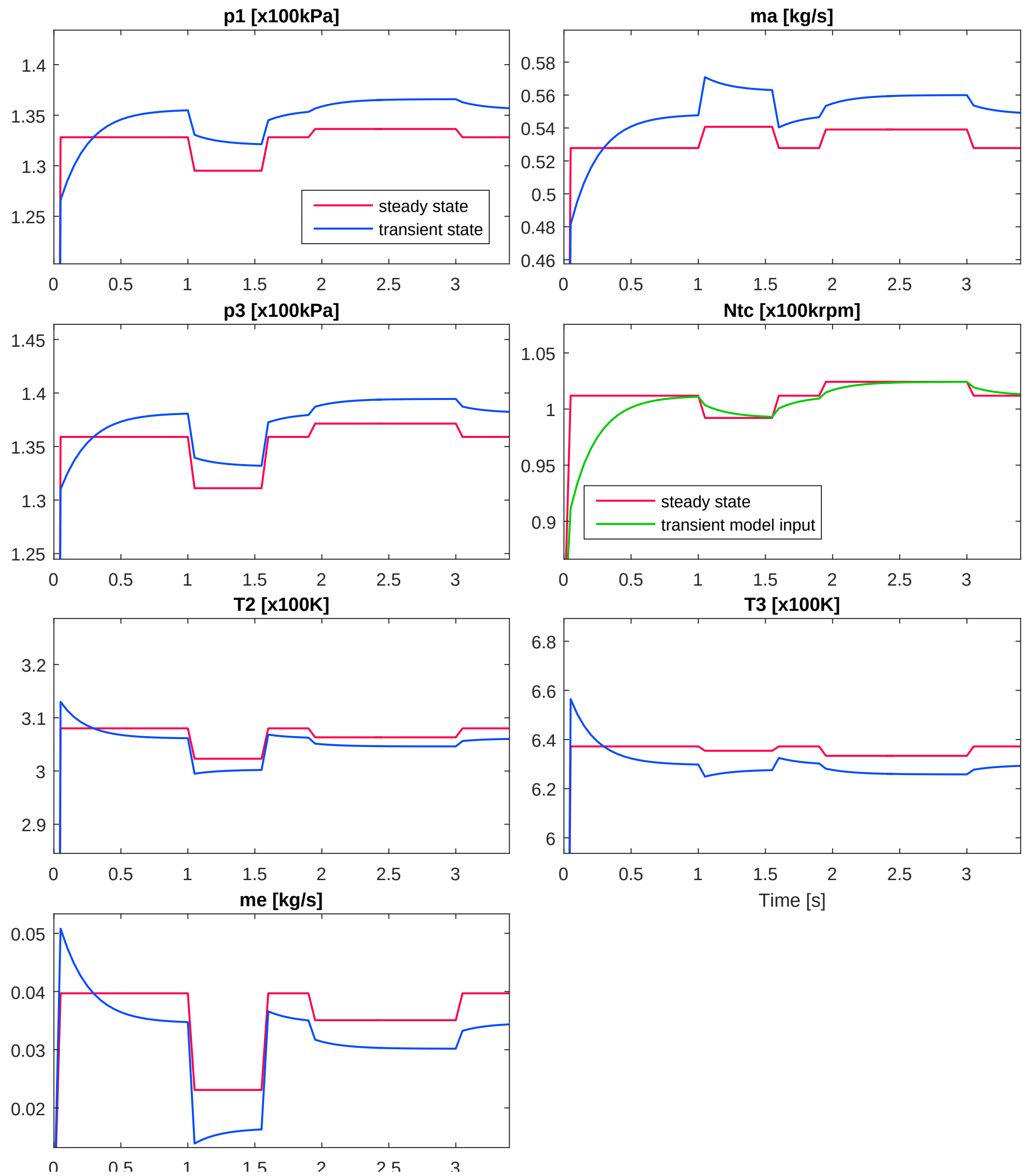

Figure 23 Transient model response. The figure shows response of all main parts of transient model algorithm: steady-state estimation (red), transient turbo-speed (green) and transient-state estimation (blue). u_{VGT} is perturbed at time $\langle 1s, 1.6s\rangle$ and u_{egr} at time $\langle 2s, 3s\rangle$. The plots are zoomed. The initial values are set to zero.

In order to determine speed of the proposed model, the above experiment was run 20 times and each algorithm's sample evaluation time was measured (See Figure 24 for the results). The algorithm code is written and run in *MATLAB* and it has not been optimized yet. The results show that a mean value of the sample evaluation is $\hat{T}_{eval} = 0.0208s$ and the largest value is $max(T_{eval}) = 0.0395s$. The experiments were done in *MathWorks MATLAB R2015b* software on Dell Precision M4600 computer with Intel Core i7-2720QM processor and 8GB RAM.

The algorithm has 4445 parameters that have to be stored in memory. These parameters are coefficients of the rational functions and the polynomials that are included in algorithm. That means it needs 35.56kB of available memory when the coefficients are stored in double-precision floating-point format or 17.78kB in case of the single-precision floating-point format. The size of flash memory in modern ECUs ranges usually from 1MB to 10MB, therefore the algorithm's memory requirements can be considered to be eligible.

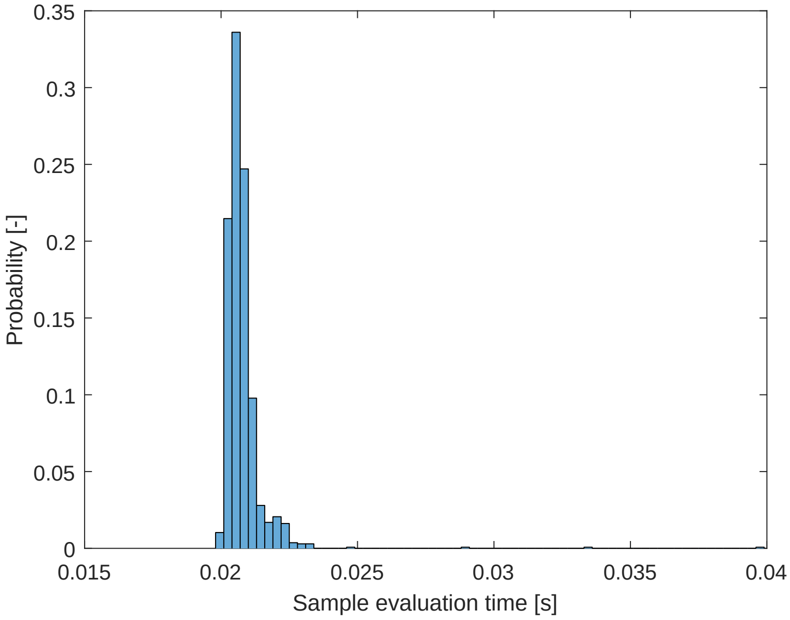

Figure 24 Probability distribution. The figure shows the distribution of the sample evaluation times.

In order to examine if the behaviour of the model corresponds to reality, the Quasi-Explicit model is compared the ODE model (4.1). The ODE model dynamics is derived directly from physics phenomena, therefore its dynamics is considered to be a reliable benchmark. The accuracy of the ODE model is determined by the polynomial models that it includes. The Quasi-Explicit model is simplified in order to make the accuracy of the ODE model as similar as possible. The model of power-shaft efficiency is substituted with the constant and the same compressor mass flow model is used as in the ODE model. However, all component models cannot be the same. This means that there still is an accuracy bias.

The experiments can only be done for the setups for which the ODE model is able to converge to a solution. The first experiment includes response to the initial conditions and continuous perturbation (sine function) of u_{egr} (see Figure 25). In the second one, u_{VGT} is perturbed by decreasing the value by 20% (see Figure 26).

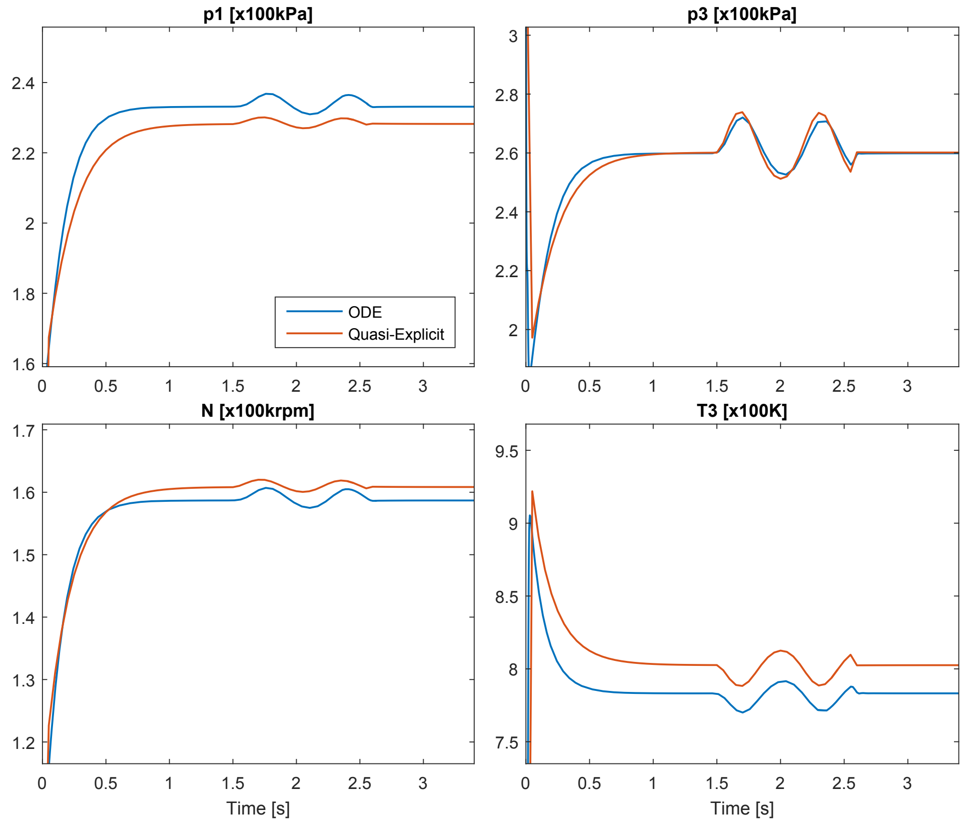

Figure 25 Model comparison. The figure shows comparison of the ODE model and the Quasi-Explicit model. The experiment shows perturbation of the EGR valve input by sine function.

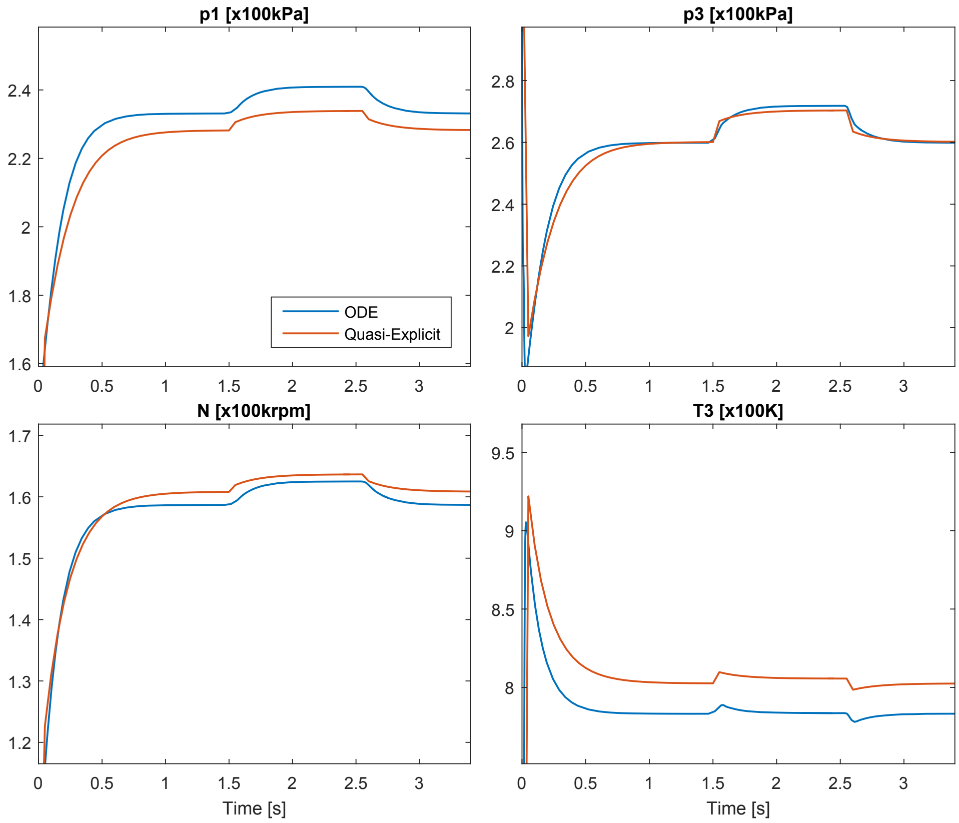

Figure 26 **Model comparison.** The figure shows comparison of the ODE model and the Quasi-Explicit model. The experiment shows perturbation of u_{VGT} decresing it by 20%.

5 Virtual Sensors

This chapter is focused on the design of the virtual sensors for the air mass flow and the EGR mass flow.

There are few options for measuring the mass flow by physical sensor. The air mass flow and EGR mass flow are closely related and often only one of them is directly measured. In the light-duty engines, the air mass flow is usually measured by hot-wire sensor (MAF) [2] and the EGR flow is obtained using model of the engine's charge flow. On the other hand in heavy-duty engines, the most common is to use Venturi sensor to measure the EGR mass flow and use the intake manifold pressure measurement to obtain the air mass flow [5].

The virtual sensor proposed in this chapter uses measurements from the lambda (oxygen) sensor which measures oxygen concentration in the exhaust gas. It is positioned after turbine and it is used in SCR (Selective Catalytic Reduction) system together with $\mathrm{NO_x}$ sensors [6] but its primary function is to measure air-to-fuel ratio. It can be used to measure the air fuel but that is usually not the case because its measurement is delayed. Lambda sensors used together with $\mathrm{NO_x}$ sensors have typically larger delay than high quality lambda sensors used for air-to-fuel ratio measurement. The delay depends on the flow velocity and can be from 1s up to 4s. The virtual sensor design merges the Quasi-Explicit model described in (4.2), delayed lambda measurements and Kalman filter algorithm.

In order to create the virtual sensor with properties more convenient for real application, the measurements that are available in common production heavy-duty turbocharged diesel engine are used. This allows the Quasi-Explicit model to be reduced into explicit formulation.

The first section in this chapter describes general structure of Kalman filter that removes physical sensor delay. The next section introduce the application of this structure to the particular problem of estimating the EGR mass flow and the air mass flow. To conclude the last section introduce the virtual sensor modification that would be capable to run on ECU.

5.1 General structure

This section proposes the idea for the observer structure that is able to remove sensor's measurement delay. It is assumed that the measurement is accurate after the delay and that the less accurate model is available for measured variable. The goal is to use the model estimation when the delay occurs and follow the sensor measurements for the rest of the time.

The system schematically shown in Figure 27 is used as the basis of such estimator.

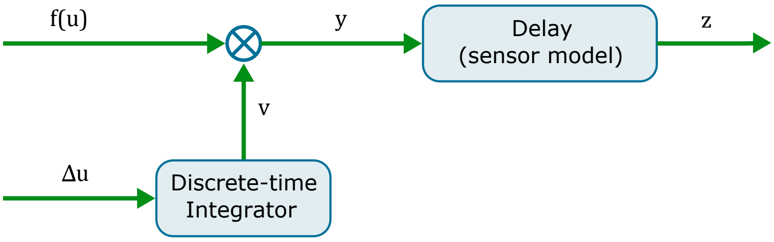

Figure 27 Observed system structure. Kalman filter is applied to the system depicted on the figure. y represents the state of the interest.

The inputs to the system are evaluation of measured model ($f(u)$) and changes in model inputs ($\Delta\, u$) which acts as disturbance. The output is delayed with the model of the real delay. This system can be written as follows

$$\begin{aligned} z(k+1) &= \frac{1}{\tau}\left(k_y\, y(k) + z(k)\right), \\ y(k+1) &= f(u(k)) + \sigma\, v(k), \\ v(k+1) &= v(k) - k_v\, \left(u(k) - u(k-1)\right). \end{aligned} \tag{96}$$

The system would also work if state v is omitted, but the responses are faster when the input differences are introduced to the system. Observing the state y in this model gives required results. In order to correctly estimate this state, the Kalman filter algorithm is applied. Since the observed system is linear and simple, the basic linear Kalman filter algorithm is chosen to be used. The optimal Kalman gain is evaluated offline by solving algebraic Riccati equation.

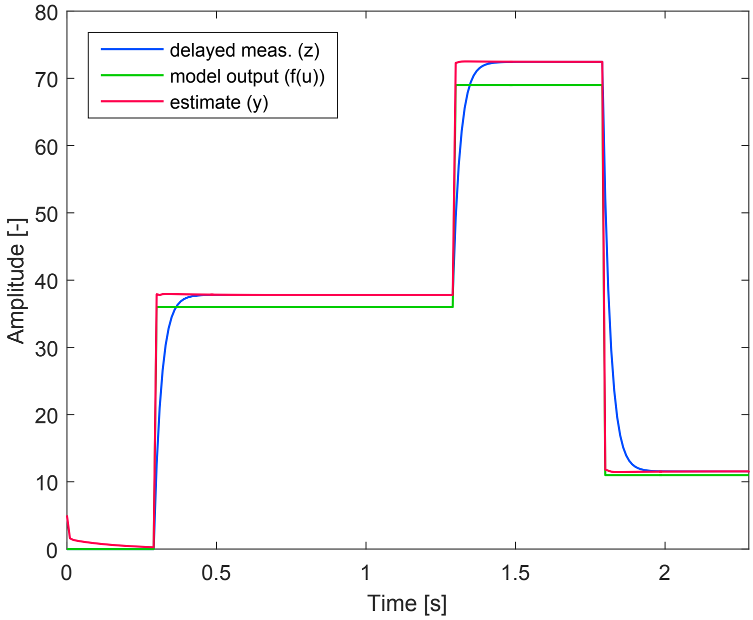

Figure 28 Estimation test. The figure shows response to few input steps. Model output (green) shows evaluation of the function $f(u)$, 5% error of the model is applied in simulation.

The described estimation system can be used to various nonlinear polynomial models with available delayed measurement. As an example, estimating of the single variable nonlinear polynomial function with the delayed value measurement is chosen. The function is given as

$$f(u) = -u^3 + 10\,u^2 + 2\,u. \tag{97}$$

The first-order delay is used to simulate delayed measurement and its parameters are same for the delay model:

$$\tau = 1.5,\;\; k_y = 0.5. \tag{98}$$

The following system parameters were chosen empirically:

$$\sigma = 0.45,\;\; k_v = 0.5. \tag{99}$$

The simulation of the system with the state estimation is shown in Figure 28. The test consists of applying few step to the system input. It can be seen that the estimation works as it is required.

Also the measurement noise can be suppressed since the Kalman filter algorithm is used.

5.2 Air mass flow

The air mass flow virtual sensor is based on the structure introduced in (5.1). The lambda sensor is used for this purpose. The lambda sensor measures an amount of oxygen in the gas and it is usually used to measure "λ" which is a ratio defined as air-to-fuel ratio (AFR) to stoichiometric ratio [10]:

$$\lambda = \frac{AFR}{AFR_{stoich}}, \quad AFR = \frac{\dot{m}_a}{\dot{m}_f}. \tag{100}$$

Stoichiometric ratio is the ratio of the exact amount of air needed to completely burn all fuel and the amount of that fuel. The value of this ratio depends on the fuel composition.

Lambda sensors are accurate but their measurement is delayed. Since the model of the air mass flow is part of the model discussed in the previous chapters, the proposed structure (5.1) fits to this situation. The structure of the observed system is shown in Figure 29.

The observed system is given as follows

$$\begin{aligned}
\dot{m}_\lambda(k+1) &= \frac{1}{\tau}\left(k_y\, y(k) + \dot{m}_\lambda(k)\right) \\
y(k+1) &= \dot{m}_{a,est} + \sigma\, v(k) \\
v(k+1) &= v(k) - \mathbf{k}_v\, \left(\mathbf{u}(k) - \mathbf{u}(k-1)\right)^T
\end{aligned} \tag{101}$$

where $\dot{m}_\lambda$ is the modelled output of the lambda sensor which is measured, $\dot{m}_{a,est}$ is computed estimation from Quasi-explicit model and the observed state in interest is y. In this case, the input $\mathbf{u}$ and $\mathbf{k}_v$ are vectors. The composition of the input variables in input vector $\mathbf{u}$ is chosen, so the input changes correspond with changes of the air mass flow. The most convenient input is u_{VGT}, other inputs showed not to be eligible for use in this vector. The use of a constant instead of Δu gave good result as well, which lowers the complexity of the sensor.

The simulation experiment (See Figure 30) is the same that was used to demonstrate the transient model functioning in Figure 23. Data of lambda sensor are generated by delaying the model data and adding 5% error. The algorithm's sampling time is set to model's sampling time (0.05s).

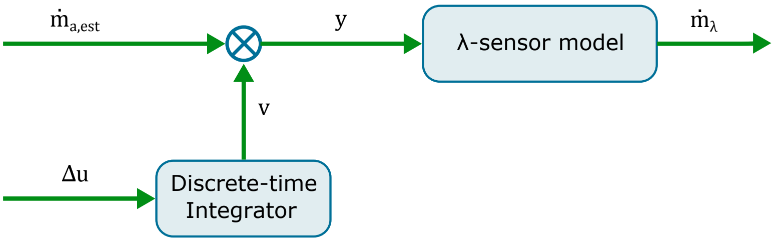

Figure 29 Virtual sensor structure. The structure depicted on the figure is the model used in kalman filter algorithm. The estimated state of the interest is y.

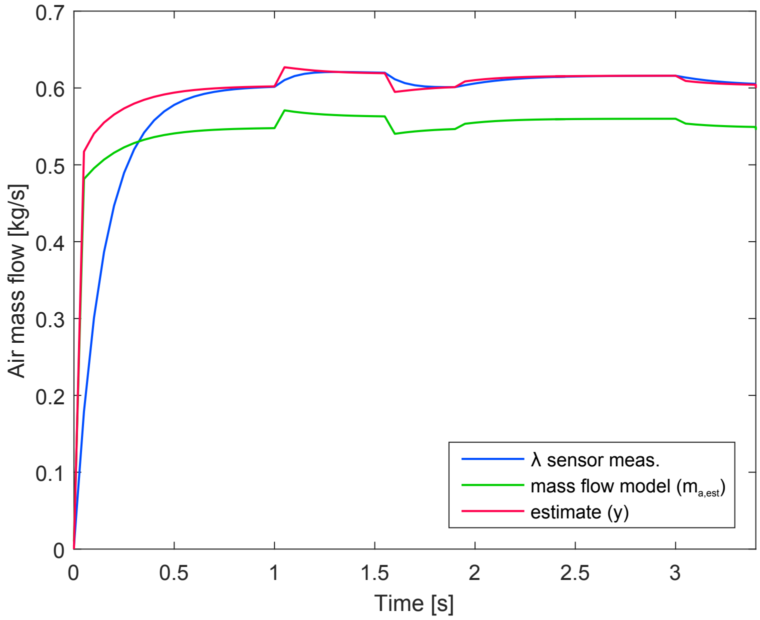

Figure 30 Estimation test. The figure shows the response to the initial conditions and to the perturbations. u_{VGT} is perturbed at time $\langle 1s, 1.6s\rangle$ and u_{egr} at time $\langle 2s, 3s\rangle$. The initial values are set to zero.

5.2.1 Accuracy

As mentioned before, the accuracy of this design cannot be tested on real data. However, there is an aspect of this approach that needs to be investigate more: the effect of humidity on the lambda sensor. The oxygen sensors are build to measure the oxygen concentration in the dry gas. The more wet gas is, the less accurate measurement is. The example of measurement formula in the actual lambda sensor is as follows

$$\lambda = \frac{0.21}{0.21 - [O_2]}, \tag{102}$$

where $[O_2]$ represents the mole fraction which is defined as

$$[O_2] = \frac{n_{O2}}{n}, \tag{103}$$

where n_{O2} is the number of moles of the oxygen in the mixture and n is the total number of moles in the mixture.

When we consider the following chemical reactions in the combustion process [10]

$$4\,\mathrm{C_{12}H_{23}} + 71\,\mathrm{O_2} \rightarrow 46\,\mathrm{H_2O} + 48\,\mathrm{CO_2} \tag{104}$$

$$\mathrm{N_2} \rightarrow \mathrm{N_2} \tag{105}$$

$$\mathrm{H_2O} \rightarrow \mathrm{H_2O} \tag{106}$$

and the fuel mass and the air percent composition by mass are known then the lambda can be computed from the the air mass, and vice versa. The error of the air mass flow measurement can be determined using this process. That means we calculate the lambda from the air mass by considering the humid air and then we calculate air mass from the lambda considering the dry air. Then the error between calculated air masses is determined. This process was recalculated for several values of relative humidity and air temperature. The results are shown in Figure 31. This error can be compensated by adding the humidity measurement.

The humidity can be added to the model by creating the virtual sensor for λ instead of $\dot{m}_a$. This means that first the air mass flow from the model is recalculated into λ using the process mentioned above. Then the lambda delay is removed using the observation of the system with introduced structure, which is following

$$\begin{aligned}\lambda(k+1) &= \frac{1}{\tau}\left(k_y\, y(k) + \lambda(k)\right)\\ y(k+1) &= \lambda_{model} + \sigma\, v(k)\\ v(k+1) &= v(k) - \mathbf{k}_v\, \left(\mathbf{u}(k) - \mathbf{u}(k-1)\right)^T .\end{aligned} \tag{107}$$

After that the estimated λ is recalculated back to the air mass flow.

To recapitulate, the difference between model (101) and model (107) is that (107) uses modelled λ (which is λ value recalculated from modelled $\dot{m}_a$) which enables us to include information about the relative humidity into the virtual sensor.

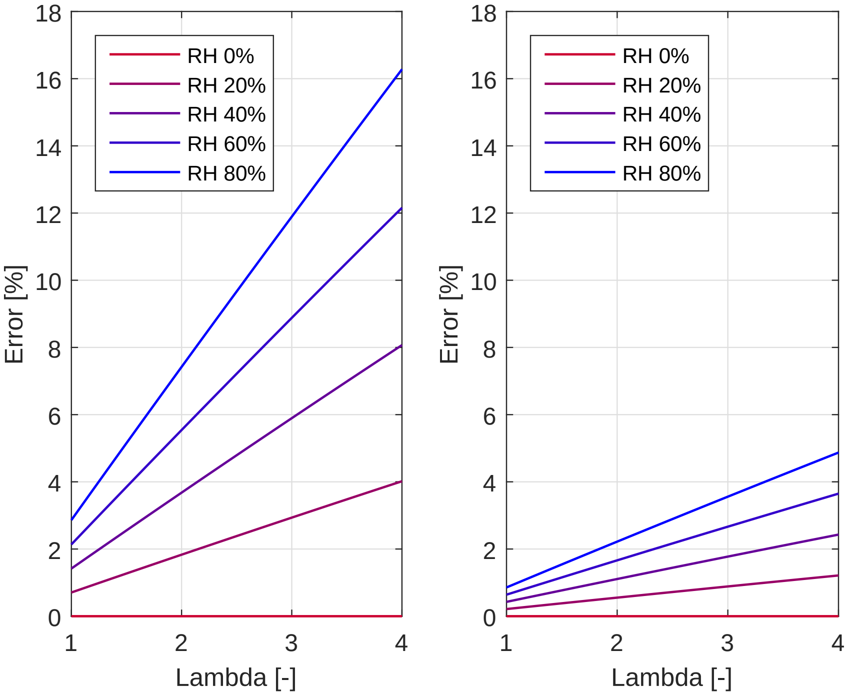

Figure 31 Lambda measurement error. The left-hand side figure show error in measurements at 35°C and the right-hand side figure represents experiments at 15°C. RH represents relative humidity in the air.

5.2.2 Gain-scheduling

The delay of the real lambda sensors is actually not a constant first order delay. However, it has been empirically observed that it is similar to the first order delay, but its delay constant differs. In order to achieve better results, the Kalman filter should change its configuration. Again, it has been observed that the delay constant is correlated with the amplitude of λ, since the lambda sensor is generally inaccurate for large λ values. This observation is used to design gain-scheduling Kalman filter.

The proposed idea is to change τ, k_y parameters in (107) according to the amplitude of λ, or in other words create the linear parameter-varying model (LPV) [20]. These changes in parameters mean that the Kalman gain (L) has to be also changed. The Kalman filter computed online in real-time is one option, but it would unnecessarily higher the computational time of the virtual sensor. The more convenient way is to evaluate τ, k_y, L offline in a plausible interval of values of λ. Then use the acquired values of τ, k_y, L to approximate them by a polynomial or a rational function that depends on λ:

$$\tau = f_\tau(\lambda), \tag{108}$$

$$k_y = f_{k_y}(\lambda), \tag{109}$$

$$L_x = f_{L_x}(\lambda), \tag{110}$$

where L_x are the elements of L vector. $f_\tau(\lambda)$ and $f_{k_y}(\lambda)$ are assumed to be affine polynomial functions, which transforms (107) into LPV system

$$\begin{aligned} \lambda(k+1) &= \frac{1}{\tau(\lambda_{meas})}\left(k_y(\lambda_{meas})\, y(k) + \lambda(k)\right) \\ y(k+1) &= \lambda_{model} + \sigma\, v(k) \\ v(k+1) &= v(k) - \mathbf{k}_v\, \left(\mathbf{u}(k) - \mathbf{u}(k-1)\right)^T, \end{aligned} \tag{111}$$

where λ_{meas} represents measurements from the lambda sensor. The approximation function for the elements of Kalman gain have been empirically chosen to be in following form

$$L_x = \frac{b_{2x}\,\lambda_{meas}^2 + b_{1x}\,\lambda_{meas} + b_{0x}}{a_{2x}\,\lambda_{meas}^2 + a_{1x}\,\lambda_{meas} + a_{0x}}, \tag{112}$$

where coefficients are identified using the nonlinear regression method. The gains were approximated in feasible lambda interval of $\langle 1.5,\ 3\rangle$. The values of lambda that are out of this interval are rounded to the nearest value of the interval boundaries. The gain approximation results are shown in Figure 32.

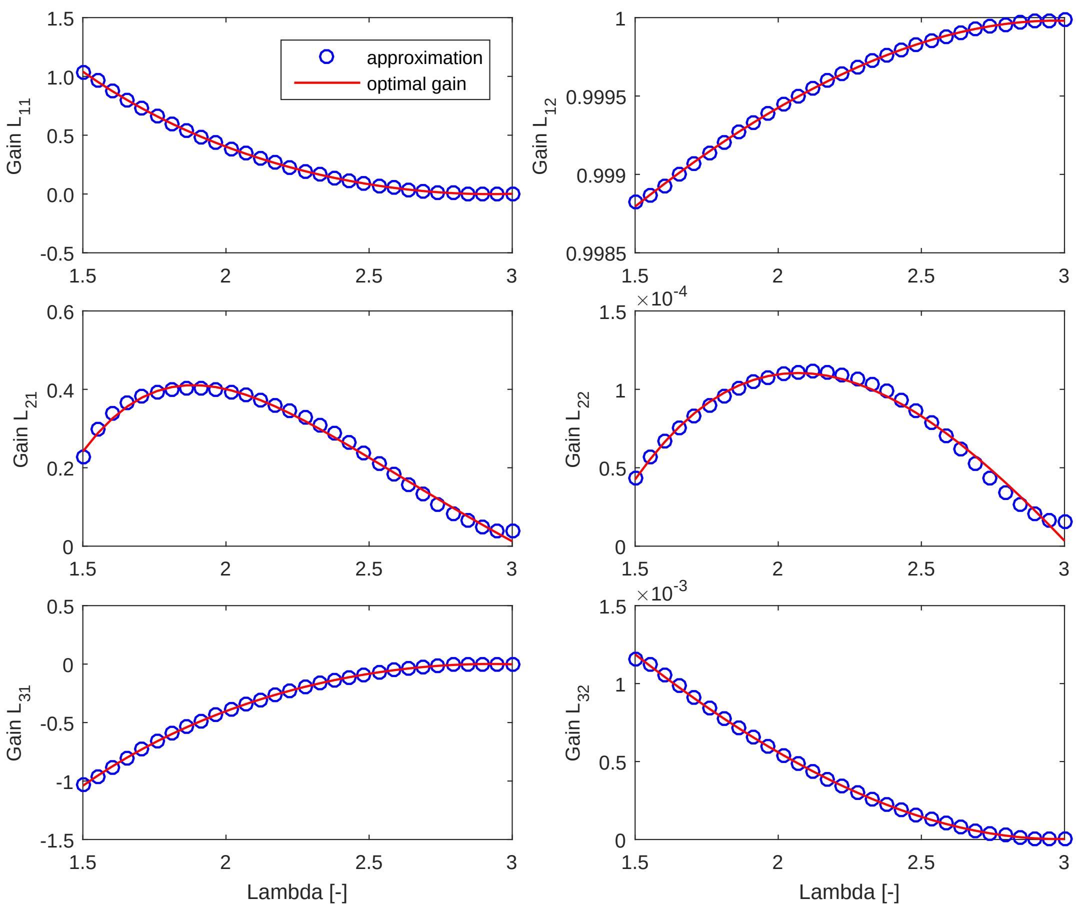

Figure 32 Scheduled gains. The figure shows approximations of the particular elements of the optimal Kalman gain matrix depending on λ by rational functions (112).

5.3 EGR mass flow

The proposed EGR mass flow virtual sensor is based on the idea that the air mass flow is known. For example, it can be estimated by the virtual sensor described in the previous section.

The most effective way to create the EGR mass flow sensor is to derive $\dot{m}_e$ from the volumetric efficiency implicit equation (66). The mass flow in this equation represents charge flow which is the sum of the air mass flow and the EGR mass flow. Then the formula of the EGR mass flow is given as follows

$$\dot{m}_e = \frac{a_0 \, N_e \, p_1 + a_1 \, N_e^2 \, p_1 - T_2 \, \dot{m}_a}{T_2}. \tag{113}$$

Accuracy of this method depends on the accuracy of the air mass flow estimation and the the accuracy of the model in (113). The latter cannot be evaluated or simulated because we cannot rely on the steady-state data for the EGR mass flow (for details see (4.3.1)).

5.4 Implementation

The implementation of virtual sensor into ECU that would be ready for production is complicated process which involves a lot of testing, optimizing and various failure analysis. Most of it is beyond the scope of this thesis.

However, the step forward to the final implementation is to create the system designed in *Simulink*. The main motivation is that C code can be easily generated from Simulink model using *MathWorks Embedded Coder*. Such code is optimized and certified for automotive ECUs.

Nowadays, the common production turbocharged engines includes many sensors which can be used to radically reduce the complexity of the Quasi-Explicit model and to improve accuracy at the same time. At some configuration, the iterative part of the model is no longer needed, therefore the model becomes explicit. Such virtual sensor model is shown in Figure 33. The engine configuration include sensors for the intake manifold pressure (p_1) The model includes function for expressing the charge mass flow which is expressed as follows

$$\dot{m}_{ch} = \dot{m}_a + \dot{m}_e, \tag{114}$$

and it is derived from volumetric efficiency model introduced in (3.8). Exhaust temperature model is the model introduced in (3.9). The block for turbine inlet pressure include function derived from turbocharger power balance model in (3.4) and the EGR mass flow block consists of model of EGR valve mass flow in (3.5).

The charge flow and the EGR mass flow values are then recalculated into λ which is then corrected using the lambda sensor. Finally the filtered λ is recalculated back to the resulting air mass flow and the EGR mass flow is obtained using the charge mass flow.

The designed model has no more than 200 parameters including the coefficients of the engine model rational functions and polynomials, the coefficients of the Kalman filter matrices and the coefficients of the gain-scheduling rational functions. That means it needs 1.6kB of available memory when the coefficients are stored in a double-precision floating-point format or 0.8kB in case of a single-precision floating-point format. The size of flash memory in modern ECUs ranges usually from 1MB to 10MB, therefore the memory requirements can be considered to be satisfied.

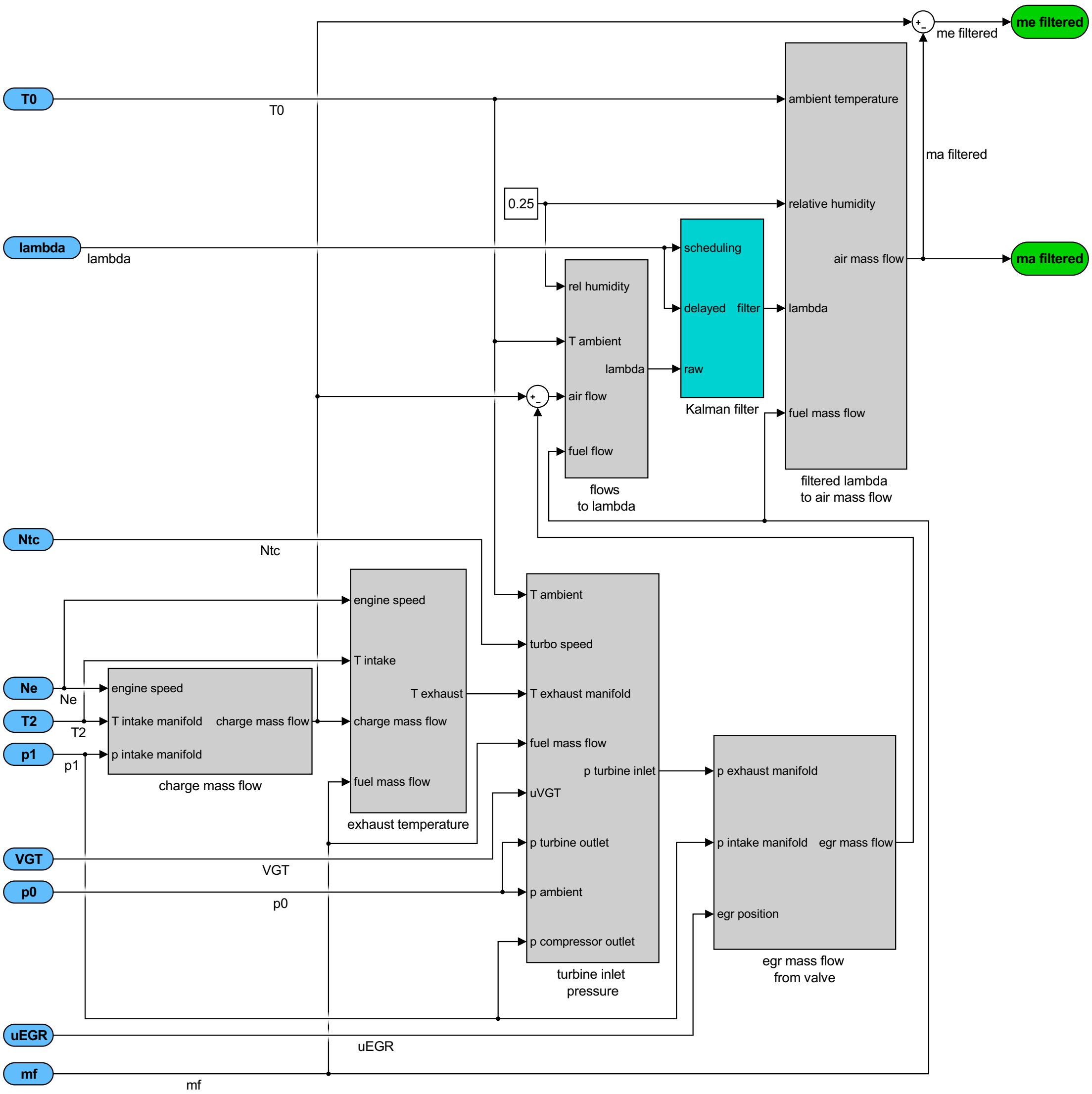

Figure 33 Simulink model.

From the engine control point-of-view, the accuracy of a CO_2 molar fraction at the intake manifold is the one that determines the quality of the virtual sensor. It can be directly evaluated from the air mass flow and the EGR mass flow.

The transient data are not available for the particular engine. However, in order to show the virtual sensor performance, the model was refit on data from different engine with mounted Venturi sensor and it was tested on real transient data. The simulation of the virtual sensor comparison with the CO_2 molar fraction computed using Venturi sensor is shown in Figure 34 and in detail in Figure 35.

The first tests of the generated code showed that it is able to run on ECU together with other ECU tasks with 0.02s period.

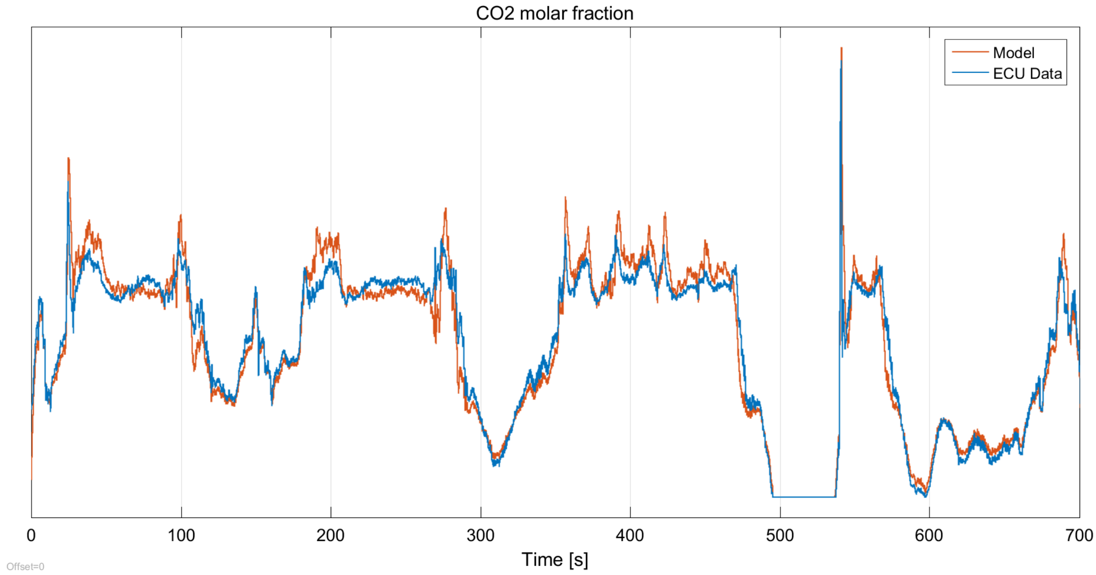

Figure 34 Comparison. The figure shows the comparison of the CO_2 molar fraction computed by proposed virtual sensor and Venturi sensor. Y-axis is not described due to confidentiality of data.

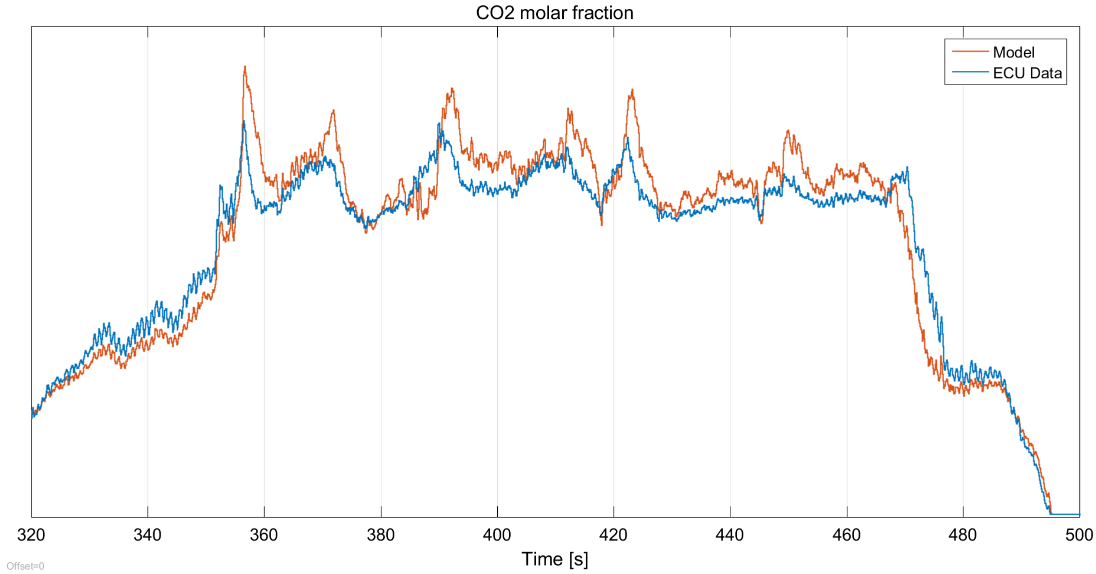

Figure 35 Comparison in a shorter time interval. The figure shows the comparison of the CO_2 molar fraction computed by proposed virtual sensor and Venturi sensor. Y-axis is not described due to confidentiality of data.

6 Conclusion

The thesis goal of creating the virtual sensor for the EGR mass flow and the air mass flow required thoroughly designed engine model with good trade off between complexity and accuracy.

The model of the turbocharged diesel engine with EGR was briefly introduced in the beginning. Each submodel of the engine was individually analysed and an implicit equation or set of implicit equations that describes its behaviour were designed. All these equations were polynomial in order to achieve low complexity and explicit differentiation, hence high usability in further use in estimation algorithms. The equations were obtained by data approximation with polynomials or rational functions whose analytical form was physics based. The submodels were designed in order to achieve good trade off between structure complexity and accuracy. The quality of the particular submodels was shown by perturbation of the individual state variables in the single point of the equation solution and observing an amplitude of deviations from that solution.

The conventional modelling approach using ODE equations was implemented and tested. Also a few other models were introduced. The DAE model algorithm was described and the Quasi-Explicit model was derived in detail and tested. The accuracy of the latter was evaluated by the comparison to the real turbocharged engine steady-state data. The accuracy of the Quasi-Explicit model was evaluated in steady-state and compared to available data. Only the steady state data were available, therefore the transient model dynamics was only compared to the ODE model. The ODE model has physics-based dynamics hence it could have been used as the benchmark. The robustness of the ODE model turned out to be very low, so the experiments were chosen wisely. In order to approach the accuracy of the ODE model, the Quasi-Explicit model was simplified. Its response character showed to be very similar to the ODE model, which demonstrates the correctness of the dynamics in the proposed model.

The simulations demonstrated that the Quasi-Explicit model evaluates all system variables with satisfying accuracy. The errors in steady-state were explicitly illustrated and evaluated using NRMSE. The sample evaluation times were statistically tested and the results showed that the evaluation of the model is close to its sampling period. However, the Matlab code used to test the model was not optimized. The memory requirements of the model were evaluated as well and they fulfilled the requirements for the implementation in ECU.

The Quasi-Explicit model uses turbo-speed measurement as input variable and it can be itself considered to be nonlinear observer for the engine. However, its accuracy of the evaluation of mass flows does not satisfy requirements for the real implementation. Therefore it has to be extended with other estimating methods.

In the last chapter, virtual sensors for the EGR mass flow and the air mass flow were designed. The goal was to replace conventional methods for measuring these mass flows and that way lower the production cost. These mass flows are closely related by the charge flow equation. Therefore, the design consists of one virtual sensor which estimates both mass flows. The designed structure is based on using accurate yet delayed measurements of lambda sensor. This structure combines modelled value together with delayed measurements and the Kalman filter algorithm. The general structure for removing the delay from the measurement was introduced. The design of the air mass flow virtual sensor was based on such structure and it was successfully simulated. Possible drawbacks of this method related to relative humidity in the air were also discussed together with available solution. The solution lies in removing the measurement delay using modelled lambda value rather than the air mass flow. The gain-scheduling Kalman filter was developed in order to accurately model the lambda sensor delay. The extension of the virtual sensor was introduced for the estimation of the EGR mass flow. It was based on the model of the volumetric efficiency. Since the Kalman filter is linear and has scheduled-gains calculated by explicit formulas, its computational time is close to zero. Therefore the computational time of the virtual sensor is basically equal to the computational time of the Quasi-Explicit model, which have been discussed. The memory requirements are similar too.

Last but not least, the real implementation of virtual sensor is discussed. The measurements available in common production engine were used in order to simplify the model and make it faster. The Quasi-Explicit model were transformed into explicit model by omitting the iteration part and several component models. The possible real implementation in *Simulink* were shown and its advantages were discussed. The memory requirements were lowered compared to full-size model. Since the transient-state data were not available, the virtual sensor were implemented for different engine and compared to the real data. The first tests of generated C code showed that the virtual sensor is able to run together with other ECU tasks on the ECU which has sampling period defined to 20ms.

In the nearest future, the performance of the proposed virtual sensor will be improved. In the further work, the proposed Quasi-Explicit model will be enhanced by optimizing *MATLAB* code and further tuning the particular component models. The mathematical proof for the existence of the Quasi-Explicit model solution should be formulated. The model will also be tested for other turbocharged diesel engines with EGR. After that more virtual sensors will be designed for sensing various model variables. The model will be also implemented in advanced control algorithms.

Bibliography

[1] MAN. *Tier III Compliance - Low Speed Engines.* MAN Diesel & Turbo. Copenhagen, Denmark, 2011.

[2] W. J. Fleming. “Overview of automotive sensors”. In: *IEEE Sensors Journal* 1.4 (Dec. 2001), pp. 296–308. ISSN: 1530-437X. DOI: 10.1109/7361.983469.

[3] Daniel Pachner and Jaroslav Beran. *Comparison of Sensor Sets for Real-Time EGR Flow Estimation.* SAE Technical Paper 2016-01-1064. 2016. DOI: 10.4271/2016-01-1064.

[4] Fredrik Laurantzon et al. *Response of common flowmeters to unsteady flow.* Technical Report. KTH Mechanics, Stockholm, Sweden.

[5] J.D. Russell, A.J. Kotwicki, and F.C. Gates. *System and method for air flow and EGR flow estimation.* US Patent 6,609,058. Aug. 2003.

[6] C.D. Telford. *Method and apparatus for selective catalytic reduction of NOx.* US Patent 7,776,280. Aug. 2010.

[7] Tomas Poloni et al. “Comparison of Sensor Configurations for Mass Flow Estimation of Turbocharged Diesel Engines”. In: *Identification for Automotive Systems.* Ed. by Manfred Morari et al. Vol. 418. London: Springer London, 2012, pp. 303–326.

[8] Erik Höckerdal. “Model Error Compensation in ODE and DAE Estimators with Automotive Engine Applications”. PhD thesis. Linköping University, 2011.

[9] Fredrik Swartling. “Gas flow observer for Diesel Engines with EGR”. MA thesis. Linköping University, 2005.

[10] Lino Guzzella and Christopher Onder. *Introduction to Modeling and Control of Internal Combustion Engine Systems.* 2009, p. 368. ISBN: 3642107753. DOI: 10.1007/978-3-642-10775-7.

[11] Johan Wahlström and Lars Eriksson. “Modelling diesel engines with a variable-geometry turbocharger and exhaust gas recirculation by optimization of model parameters for capturing non-linear system dynamics”. In: *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* 225.7 (2011), pp. 960–986. ISSN: 0954-4070. DOI: 10.1177/0954407011398177.

[12] R. Sharma, D. Nesic, and C. Manzie. “Model Reduction of Turbocharged (TC) Spark Ignition (SI) Engines”. In: *IEEE Transactions on Control Systems Technology* 19.2 (2011), pp. 297–310. ISSN: 1063-6536. DOI: `10.1109/TCST.2010.2043735`.

[13] Daniel Pachner, Jaroslav Beran, and Jans Tigelaar. *Uncertainty Analysis of a Virtual Turbo Speed Sensor*. SAE Technical Paper 2016-01-0096. 2016. DOI: `10.4271/2016-01-0096`.

[14] Dean Kihas et al. *Concept Analysis and Initial Results of Engine-Out NOx Estimator Suitable for on ECM Implementation*. SAE Technical Paper 2016-01-0611. 2016. DOI: `10.4271/2016-01-0611`.

[15] F. Soltanian, Mehdi Dehghan, and S. M. Karbassi. “Solution of the differential algebraic equations via homotopy perturbation method and their engineering applications”. en. In: *International Journal of Computer Mathematics* 87.9 (2010), pp. 1950–1974. ISSN: 0020-7160, 1029-0265. DOI: `10.1080/00207160802545908`.

[16] J.-P. Jensen et al. *Mean Value Modeling of a Small Turbocharged Diesel Engine*. Tech. rep. 1991. DOI: `10.4271/910070`.

[17] Jonas Biteus. “Mean Value Engine Model of a Heavy Duty Diesel Engine”. In: *Sensors (Peterborough, NH)* December (2004).

[18] Xiande Fang and Yu Xu. “Development of an empirical model of turbine efficiency using the Taylor expansion and regression analysis”. In: *Energy* 36.5 (2011), pp. 2937–2942. ISSN: 03605442. DOI: `10.1016/j.energy.2011.02.036`.

[19] G. De Nicolao, R. Scattolini, and C. Siviero. “Modelling the volumetric efficiency of IC engines: Parametric, non-parametric and neural techniques”. In: *Control Eng. Pract.* 4.10 (1996), pp. 1405–1415. ISSN: 09670661. DOI: `10.1016/0967-0661(96)00150-5`.

[20] Wilson J. Rugh and Jeff S. Shamma. “Research on gain scheduling”. In: *Automatica* 36.10 (2000), pp. 1401–1425. ISSN: 0005-1098. DOI: `10.1016/S0005-1098(00)00058-3`.