CHALMERS

Real-time Estimation of Pressure in Diesel Intake Manifolds

Mattias Eriksson

Department of Signals and Systems
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden

SSYX04/2011

Abstract

This master thesis work deals with the task of finding a simple mathematical model to describe the pressure dynamics of the intake manifold in diesel engines. The model takes into account the intake throttles for air and exhaust gas recirculation (EGR).

A good model is essential for the engine control, especially in transient conditions. The control is performed in real-time, why the computational burden of the calculations must be low.

In this thesis work, a physical model of the dynamics is presented. Numerical optimization methods are used to find a solution-set from the model. The solution-set acts as a reference for grey-box modelling, with the aim of creating a closed parameterized function describing the pressure of the intake manifold.

The proposed grey-box model suffers from a high maximum error of 43%.

Other models are presented with the aim of being used in collaboration with simulation of the system. Different models are chosen for working points selected to cover the most difficult conditions to control. For those models, the accuracy is acceptable.

KEYWORDS: pressure estimation, diesel engine, intake manifold, EGR, optimization, parameterization

Sammanfattning

Detta examensarbete handlar om att hitta en enkel matematisk modell för att beskriva trycket i insugsröret på en dieselmotor. Modellen tar hänsyn till spjällen för ingångsluft och avgasåterföring (EGR).

En bra modell är viktig för motorns reglersystem, speciellt under transienta förlopp. Regleringen sker i realtid, och därför får inte beräkningarna vara krävande i fråga om datorkraft.

I detta arbete presenteras en fysikalisk modell av flödesdynamiken. Numerisk optimering används för att generera en lösningsmängd utifrån modellen. Lösningsmängden fungerar som en referens för gråboxmodellering, med syftet att skapa en sluten parametriserad function som beskriver trycket i insugsröret.

Den föreslagna gråboxmodellen har tyvärr ett alltför högt maximalt fel (43%).

Andra modeller presenteras för att användas tillsammans med simulering av systemet. Olika modeller avpassass för olika arbetspunkter, valda för att täcka de mest svårstyrda intervallen. För dessa intervall är noggrannheten acceptabel.

NYCKELORD: tryckestimering, dieselmotor, insugsrör, EGR, optimering, parametrisering

Preface

This is the report of a master thesis project at Chalmers University of Technology. The work was conducted at Combine AB, with supervision from the Department of Signals and Systems at Chalmers.

I would like to thank the engineers at Combine AB, and especially CEO Anders Movert, for the regular supervision of the work and for giving intelligent inputs when needed. Thanks also for the high-performance hardware that shortened many days of heavy simulations.

I would also like to thank my supervisor at Chalmers, docent Torsten Wik, for guidelines in the work process and creative comments and discussions on the specific problems to solve.

Ljungskile, September 15, 2011

MATTIAS ERIKSSON

Contents

7   DISCUSSION   31

8   CONCLUSION   32

1 INTRODUCTION

Electronic control units are an important part in the design and construction of a modern car. The benefits of control are many, e.g. better performance, handling and safety. Modern engine control in cars is very much focused on reducing toxic emissions, but has other objectives as well. The output torque in a diesel-engine is controlled by the air/fuel-ratio, for example. (Guzzella and Onder 2010)

Among the unwanted pollutants, there are nitrogen oxides (NO_x). NO_x is created when the temperature in the combustion chamber rises too much. To reduce the combustion temperature, Exhaust Gas Recirculation (EGR) is used. The idea is simple; reduce the combustible components in the combustion by recirculating exhaust gas.

Amstutz and Del Re (1995) proposes a linear sensor-based control for the EGR-flow, and concludes that the transient operations are not well handled. The sensors are not good enough to handle fast nonlinear dynamics. A better alternative is model-based control, which means the controller is based on a model of the system that is controlled. A good model is the basis for a good control design. A difficult problem in diesel engines of today is to model the pressure in the intake manifold during transient conditions. To model the air/fuel-ratio, the flows into the intake manifold must be known. In the case when the pressure differences around the intake- and EGR-throttles are small, a small change in pressure may lead to a significant change in the EGR-flow.

The goal for this thesis is to create an algorithm that can be used to estimate the pressure in the intake-manifold. The estimation is carried out in real-time, so the computational burden must be small for the control unit.

This thesis is limited to a specific number of connections to the manifold, that is an intake-throttle, an EGR-throttle and a flow from the manifold to the combustion chambers. The pressures before the throttles are assumed to be known, as well as the physical properties of the throttles and the gas-flow to the combustion chambers. It is sometimes meaningful to let the flow go backwards through the EGR-throttle, and this case is also covered by this work.

It is generally possible to simulate the system in real-time, by integrating flows and pressure in discrete time. Problems occur, however, when the difference in pressure directly before and after one throttle is small. The integration of the flow is then unstable, because the integration function has an infinite derivative and is not Lipschitz-continuous. Guzzella and Onder (2010) suggest some possible solutions to this problem. All of them include changing the fundamental equation from the pressure ratio α to the mass-flow $\dot{m}$ by limiting the maximum derivative of the function.

It is of interest to find a solution that includes all physical dynamics of the problem. The methods examined are classified into

- Numerical solutions. Iterative methods to find the solution in real time. Those solutions suffer from the problem of the infinite derivative, and they may be unstable for some conditions. Descent optimization methods and iterative fixpoint methods are used.

- Parameterized solutions. Given a large set of solution-points, find an analytical function matching those points.

The result of this thesis work shows that numerical methods are possibly unstable, or tend to have very slow convergence under some conditions. Among the proposed parameterized solutions, none is sufficiently good for all possible combinations of dependent variables. If limited to specific regions for which the differences in pressure around the throttles are small, there is however a combination of methods that has a maximum relative error of 2.5%, which is acceptable. The algorithm then has to choose method depending on the operating point, which may be tricky.

2 NOTATION

Capital Letters

A area (m^2)
C constant (dimensionless)
$\boldsymbol{F}$ Flow rate function (dimensionless)
J Jacobian
P pressure (Pa)
Q coefficients of 3rd grade polynomial equations
R gas constant (J kg^{-1} K^{-1})
S aggregated variable (dimensionless)
T temperature (K)
U cost-function for optimization (dimensionless)
V volume (m^3)

Small Letters

a aggregated variable connected to the manifold pressure (dimensionless)
b aggregated variable connected to the intake throttle (dimensionless)
c aggregated variable connected to the EGR-throttle (dimensionless)
d pressure ratio (dimensionless)
e relative error (dimensionless)
c_p specific heat at constant pressure (J kg^{-1} K^{-1})
c_v specific heat at constant volume (J kg^{-1} K^{-1})
f mass flow rate function (dimensionless)
$f_{\max}$ maximum mass flow rate at a throttle (kg s^{-1})
g aggregated function (dimensionless)
k parameterization constant (dimensionless)
m mass (kg)
n iteration number
$\dot{m}$ mass flow rate (kg s^{-1})
$\boldsymbol{p}$ descent direction
p parameterization constant (dimensionless)
r aggregated function (dimensionless)
t time (s)
x_1 flow rate at intake throttle (dimensionless)
x_2 flow rate at egr throttle (dimensionless)
$\boldsymbol{x}$ vector aggregation of x_1 and x_2

Greek Letters

α pressure ratio
β step-length
γ ratio of specific heats, c_p/c_v
Δ slope in b-c plot
κ filter factor for fixed-point iteration
λ factor for Gauss-Newton modification
ξ temporary variable in calculation of limits

Subscripts

0   fixed
k iteration
T throttle
s sample (time)
S stagnation (applies to P, T)
c critical (applies to the pressure ratio limit for choked flows)
uc unchoked (flow)
in   before or directly referring to intake throttle
egr   before or directly referring to the throttle controlling egr
bound   boundary of the bc-plane
level   level-curves in the bc-plane
comb   combined

Superscripts

(n) sample number (connected to time)

Diacritical marks

ˆ   approximative

3 MODELING

3.1 Gas flow through a nozzle
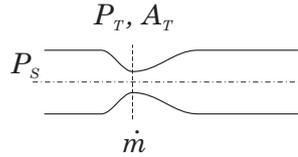
Figure 3.1. Flow through a nozzle.

Consider a gas flow through a nozzle as in Figure 3.1. The gas is suppposed to be ideal. P_T and A_T is the pressure and cross-section area, respectively, in the throat of the nozzle, P_S and T_S are the stagnation pressure and temperature.

The flow velocity is maximized when the throttle flow equals the velocity of sound (Heywood 1988). This is called choked flow, and occurs for a pressure ratio lower than the critical pressure ratio α_c, defined by

$$\alpha_c = \left(\frac{P_T}{P_S}\right)_{\text{critical}} = \left(\frac{2}{\gamma+1}\right)^{\gamma/(\gamma-1)} , \tag{3.1}$$

where $\gamma = c_p/c_v$ is the specific heat ratio. The critical pressure ratio corresponds to the maximum flow rate

$$\dot{m} = f_{\text{max}} = \frac{A_T P_S}{\sqrt{RT_S}} \sqrt{\gamma} \left(\frac{2}{\gamma+1}\right)^{(\gamma+1)/2(\gamma-1)} .$$

For unchoked flows, the mass rate is described by

$$\dot{m} = \frac{A_T P_S}{\sqrt{RT_S}} \left(\frac{P_T}{P_S}\right)^{1/\gamma} \left\{\frac{2\gamma}{\gamma-1}\left(1-\left(\frac{P_T}{P_S}\right)^{(\gamma-1)/\gamma}\right)\right\}^{1/2} , \tag{3.2}$$

where R is the gas constant. Equation (3.2) can be reformulated to include choked flows by

$$\dot{m} = f_{\text{max}} f\left(\frac{P_T}{P_S}\right) \tag{3.3}$$

5

where $f(\alpha)$ takes different form depending on the value of the pressure ratio α. For ratios of α less than the critical ratio α_c, f is equal to 1. In the unchoked region $\alpha_c < \alpha \le 1$, f follows the analytical expression hidden in Equation (3.2):

$$f_{\mathrm{uc}} = C\,\alpha^{1/\gamma}\left(1 - \alpha^{(\gamma-1)/\gamma}\right)^{1/2}, \tag{3.4}$$

where C is a constant, depending on γ alone:

$$C = \sqrt{\frac{2\gamma}{\gamma-1}}\left(\sqrt{\gamma}\left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{2(\gamma-1)}}\right)^{-1} = \sqrt{\frac{2}{\gamma-1}}\left(\frac{2}{\gamma+1}\right)^{-\frac{\gamma+1}{2(\gamma-1)}}.$$

On the other hand, if $\alpha > 1$, meaning that $P_T > P_S$, the problem is symmetric, and the flow will behave the same but flow in the opposite direction. Collecting these pieces of information gives us the following definition of f:

$$f(\alpha) = \begin{cases} 1 & , 0 < \alpha \le \alpha_c \\ f_{\mathrm{uc}}(\alpha) & , \alpha_c < \alpha \le 1 \\ -f_{\mathrm{uc}}(1/\alpha) & , 1 < \alpha < 1/\alpha_c \\ -1 & , \alpha \ge 1/\alpha_c \end{cases} \tag{3.5}$$

f is not defined on the negative real axis, since that reflects a negative pressure, which is physically impossible. The derivative of f will be needed further on. Differenting f_{uc} results in

$$\frac{df_{\mathrm{uc}}}{d\alpha} = C\frac{2\alpha^{-\frac{\gamma-1}{\gamma}} - \gamma - 1}{2\gamma\sqrt{1 - \alpha^{\frac{\gamma-1}{\gamma}}}}. \tag{3.6}$$

When $\alpha \to 1$, $\frac{df_{\mathrm{uc}}}{d\alpha} \to -\infty$ independent of whether it is limit from left or right. Including choked flows on the derivatives gives

$$\frac{df}{d\alpha}(\alpha) = \begin{cases} 0 & , 0 < \alpha \le \alpha_c \\ \frac{df_{\mathrm{uc}}}{d\alpha}(\alpha) & , \alpha_c < \alpha < 1 \\ -\infty & , \alpha = 1 \\ \frac{df_{\mathrm{uc}}}{d\alpha}(1/\alpha) & , 1 < \alpha < 1/\alpha_c \\ 0 & , \alpha \ge 1/\alpha_c \end{cases}$$

A plot of f together with its derivative is shown in Figure 3.2.

3.2   Model of the intake manifold

Figure 3.3 shows the variable definitions for the relevant quantities in the manifold model. Assume that the manifold pressure P equals that in the throttle throats. Assume also that the stagnation pressure is constant in the throttle.
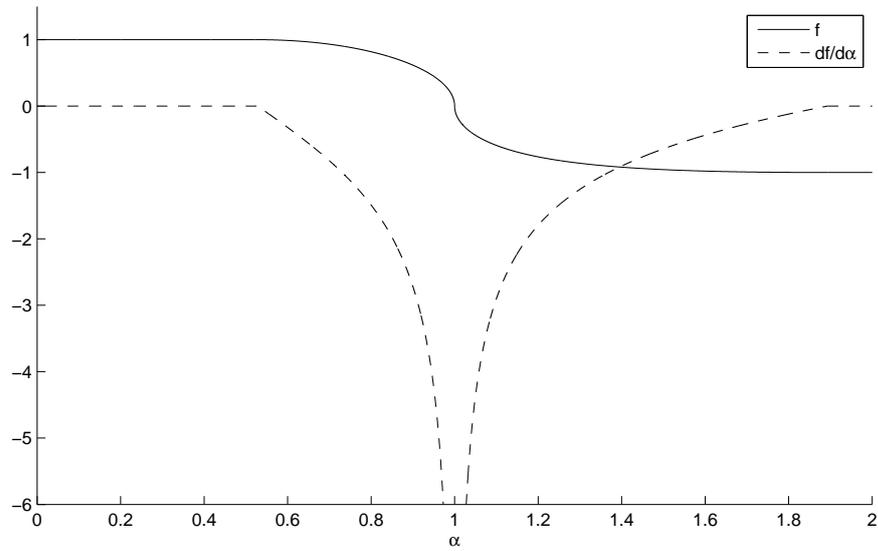
Figure 3.2. The function $f(\alpha)$ (describing the flow through a nozzle) and its derivative.
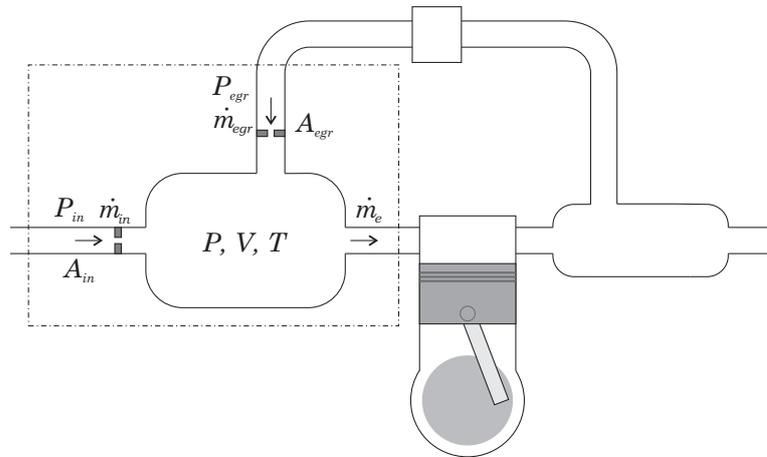
Figure 3.3. A simplified view of the intake manifold.

The resulting equations are then

$$\dot{m}_{\mathrm{in}} = f_{\mathrm{max,in}} f\left(\frac{P}{P_{\mathrm{in}}}\right)$$

$$\dot{m}_{\mathrm{egr}} = f_{\mathrm{max,egr}} f\left(\frac{P}{P_{\mathrm{egr}}}\right)$$

Introducing $x_1 = f(P/P_{\text{in}})$ and $x_2 = f(P/P_{\text{egr}})$ gives

$$\dot{m}_{\text{in}} \quad = \quad f_{\text{max,in}} x_1 \tag{3.7}$$

$$\dot{m}_{\text{egr}} \quad = \quad f_{\text{max,egr}} x_2 \tag{3.8}$$

Applying the ideal gas law in the intake manifold gives

$$P = \frac{RT}{V} m \ .$$

Assume that we have adiabatic conditions and that the system is sampled with a sample time t_s. Integrating the pressure using backward Euler from time $t_s(n-1)$ to $t_s n$ gives

$$P^{(n)} = P^{(n-1)} + t_s \frac{RT}{V} (\dot{m}_{\text{in}} + \dot{m}_{\text{egr}} - \dot{m}_e) \,, \tag{3.9}$$

assuming that the mass flows and the temperature are constant on the sampling interval.

Insert (3.7) and (3.8) into the equation above, and apply to x_1 and x_2 at time-step n:

$$x_1 \quad = \quad f\left(\frac{P}{P_{\text{in}}}\right) = f\left(\frac{P^{(n)}}{P_{\text{in}}}\right) = f\left(a + bx_1 + cx_2\right) \tag{3.10}$$

$$x_2 \quad = \quad f\left(\frac{P}{P_{\text{egr}}}\right) = f\left(\frac{P^{(n)}}{P_{\text{egr}}}\right) = f\left(d(a + bx_1 + cx_2)\right) \,, \tag{3.11}$$

where

$$a \quad = \quad \frac{P^{(n-1)}}{P_{\text{in}}} - \frac{t_s RT \dot{m}_e}{V P_{\text{in}}}$$

$$b \quad = \quad \frac{t_s RT f_{\text{max,in}}}{V P_{\text{in}}}$$

$$c \quad = \quad \frac{t_s RT f_{\text{max,egr}}}{V P_{\text{egr}}}$$

$$d \quad = \quad \frac{P_{\text{in}}}{P_{\text{egr}}} \ .$$

Equation (3.10) and (3.11) form a system of implicit equations, and the objective of the remaining part of this report is to find a computationally cheap solution to this specific problem. Introducing

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

and

$$\boldsymbol{F}(\boldsymbol{x}) = \begin{bmatrix} F_1(\boldsymbol{x}) \\ F_2(\boldsymbol{x}) \end{bmatrix} = \begin{bmatrix} f(a + bx_1 + cx_2) \\ f(d(a + bx_1 + cx_2)) \end{bmatrix} \ .$$

gives a shorter notation to the problem described by (3.10) and (3.11):

$$\boldsymbol{x} = \boldsymbol{F}(\boldsymbol{x}) \ . \tag{3.12}$$

3.3   Boundaries of the dependent variables

$a, b, c,$ and d all represent combinations of common physical properties. Thus, they are all real-valued. Table 3.1 shows the possible limits in which the variables are assumed to stay within.

Table 3.1. Possible intervals for the dependent variables

Variable	Interval
a	(0,2)
b	(0,20)
c	(0,20)
d	(0.5,2)

4 NUMERICAL SOLUTIONS

Our task is to make the solutions to Equation (3.12) available in real-time. This chapter describes how to solve the problem with numerical methods. The aim is twofold. Firstly, the generated data will be used as reference for parameterized solutions. Secondly, if a numerical solution is sufficiently fast, it may be possible to use it in real-time as an alternative to a parameterized solution.

4.1 Optimization

Introduce the cost-function

$$U(\boldsymbol{x}) = \frac{1}{2} \left(\boldsymbol{x} - \boldsymbol{F}(\boldsymbol{x}) \right)^2 = \frac{1}{2} \left((x_1 - F_1(\boldsymbol{x}))^2 + (x_2 - F_2(\boldsymbol{x}))^2 \right) . \qquad (4.1)$$

If a solution to (3.12) exists, it is equivalent to the solution to the following optimization problem:

$$\underset{\boldsymbol{x} \in \mathbb{R}^2}{\text{minimize}} \, U(\boldsymbol{x}) \qquad (4.2)$$

We also know when a solution is found, when $U = 0$.

4.1.1 Analysis of the optimization problem

Every parameter choice of a, b, c and d results in a unique cost-function $U(x_1, x_2)$ to minimize. Figure 4.1 shows a typical cost-function for a specific parameter set, $a = 0$, $b = 3$, $c = 8$ and $d = 0.5$. The function is obviously not convex, but seems to be quasiconvex. If a function is quasiconvex, it has a unique minimum value (Frenk and Kassay 2001). U is not analytically proved to be quasiconvex for all choices of parameters, but no exceptions have been found in this thesis work.

A few different optimization methods are presented. For this specific problem, the Gauss-Newton method is the most successful one, but there are problems with convergence for some choices of parameters. A variation of fix-point iteration was implemented with slower but more consistent convergence to the solution.

The optimization terminates when

$$U \le 10^{-6} .$$
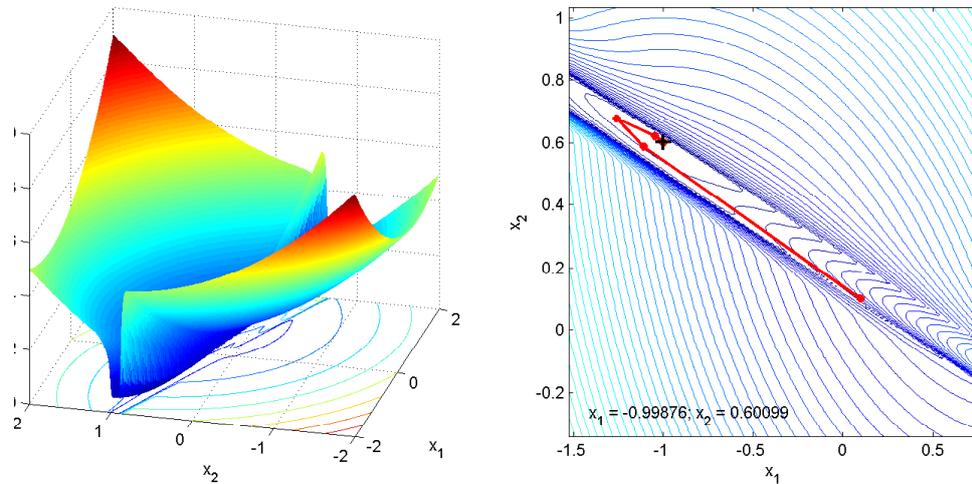
Figure 4.1.   The cost function U for the parameter set $a = 0$, $b = 3$, $c = 8$ and $d = 0.5$. A numerical solution with the Gauss-Newton method is shown to the right.

4.1.2   Descent algorithms

This is a large class of search algorithms consisting of a number of steps (Andreasson *et al.* 2005).

1. Let $k = 0$. Choose a starting point $\boldsymbol{x}_0$.

2. Choose a descent direction $\boldsymbol{p}_k$.

3. Choose a step length β_k, i.e. $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \beta_k \boldsymbol{p}_k$, such that $U(\boldsymbol{x}_{k+1}) < U(\boldsymbol{x}_k)$.

4. If a termination criterion is fulfilled, then stop. Otherwise, let $k = k + 1$ and go to step 2.

For step 2, gradient descent, Newton's method and the Gauss-Newton method was implemented to find the descent direction $\boldsymbol{p}_k$. Among those, the Gauss-Newton method performed the best.

Gradient descent

Gradient descent is probably the most intuitive of the descent methods. The search direction is simply chosen to be the negative gradient of the cost objective function.

$$\boldsymbol{p}_k = -\nabla U(x_1, x_2)$$

The gradient is known analytically, which is good, but the changing topology of U makes gradient descent a bad choice with a very slow convergence.

Newton's method

Newton's method uses

$$\boldsymbol{p}_k = -(\nabla^2 U(x_1, x_2))^{-1}\nabla U(x_1, x_2)$$

The Hessian $\nabla^2 U(x_1, x_2)$ is known analytically just like the gradient. However, for the specific problem defined in (4.2), the Gauss-Newton method is experimentally proved to have faster convergence, so it is used instead.

Gauss-Newton method

The Gauss-Newton method uses an approximation of the Hessian on functions of the quadratic form in (4.1). For reasons of simplicity, denote $r_1(\boldsymbol{x}) = x_1 - F_1(\boldsymbol{x})$ and $r_2(\boldsymbol{x}) = x_2 - F_2(\boldsymbol{x})$. Let J be the Jacobian matrix defined by

$$J = \begin{bmatrix} \frac{dr_1}{dx_1} & \frac{dr_1}{dx_2} \\ \frac{dr_2}{dx_1} & \frac{dr_2}{dx_2} \end{bmatrix} = \begin{bmatrix} 1 - b\frac{df}{d\alpha}(a + bx_1 + cx_2) & -c\frac{df}{d\alpha}(a + bx_1 + cx_2) \\ -bd\frac{df}{d\alpha}(d(a + bx_1 + cx_2)) & 1 - cd\frac{df}{d\alpha}(d(a + bx_1 + cx_2)) \end{bmatrix}.$$

$\nabla^2 U \approx J^T J$, giving

$$\boldsymbol{p}_k = -(J^T J)^{-1}\nabla U(x_1, x_2).$$

In practice, the descent direction $\boldsymbol{p}_k$ is generated by solving the equation system

$$-J^T J \boldsymbol{p}_k = \nabla U. \tag{4.3}$$

In the case when $J^T J$ is not positive definite, steepest descent is used. If the solution is not found after 120 iterations, a modification to the descent direction is used. If a solution is not found after another 300 iterations, fixed-point iteration is performed (section 4.2).

The initial point $\boldsymbol{x}_0 = [0.1\,,\,0.1]^T$ was used.

Modification by adding diagonal matrix

The objective function U has something similar to a canyon in the topology close to the optimal point (Figure 4.1). For some parameter settings, the Gauss-Newton method creates successive points that alternate between the sides, without finding the low parts of the objective. If the termination criterion has not been fulfilled after 120 iterations, a factor λ is introduced and Equation (4.3) is modified according to

$$\lambda_k = \begin{cases} 0.5\lambda_{k-1} & ,\ U(\boldsymbol{x}_{k-1}) \leq U(\boldsymbol{x}_{k-2}) \\ 200\lambda_{k-1} & ,\ U(\boldsymbol{x}_{k-1}) > U(\boldsymbol{x}_{k-2}) \\ 10^{-3} & ,\ k = 300 \end{cases}$$

$$(J^T J + \boldsymbol{I} \lambda_k \operatorname{diag}(J^T J)) \boldsymbol{p}_k = \nabla U \,.$$

This means that the diagonal elements in the $J^T J$-matrix are enforced in each step and λ decides the amount of enforcement. λ is increased if the objective function is increased from one step to another and is decreased otherwise.

An intuitive explanation for the modification is to avoid steps in the direction where the partial second derivative of the objective is large. The new iteration will tend to follow the level curves.

Golden section search

The line search in step 3 is implemented with golden section search, a direct search method that works for unimodal functions.(Andreasson *et al.* 2005) This holds if our original multi-dimensional minimization problem is quasi-convex, which we assume it is.

The method searches successively smaller intervals, where the remaining proportion of the original interval for evaluation in each step is $\frac{\sqrt{5}-1}{2} \approx 0.618$.

The termination is based on a criterion on the step-size, see Press *et al.* (2007).

Figure 4.1 shows an example of the minimization problem for a specific parameter set using Gauss-Newton for the descent direction $\boldsymbol{p}_k$ and Golden section search for the step-size β_k.

4.2   Fixed-point iteration

This method is experimentally proved to converge for all possible parameter sets, even if it normally has a slow convergence. The idea is to simply iterate the implicit equation by

$$\boldsymbol{x}_{n+1} = (1 - \kappa_n)\boldsymbol{x}_n + \kappa_n \boldsymbol{F}(\boldsymbol{x}_n) \,, \tag{4.4}$$

where κ_n is a filter parameter to prevent instability, $\boldsymbol{F}(\boldsymbol{x})$ is defined by (3.12). κ_n is updated in each iteration by

$$\kappa_n = \left(1 - \frac{3n}{2n_{\max}}\right)\kappa_0 \,,$$

where the parameters were chosen to be $\kappa_0 = 10^{-3}$, $n_{\max} = 10^5$, $\boldsymbol{x}_0 = [0.1,\, 0.1]^T$.

4.3   Analysis of the algorithms

The twofold aim of the numerical algorithms was introduced in the beginning of this chapter. The first aim was to generate the values of the solutions to Equation (3.12), the second to find a computationally cheap numerical algorithm to run in real-time. This section is about the algorithms in terms of accuracy and ability to find the solution.

Gauss-Newton was run around $14 \cdot 10^6$ times for different parameter settings. It succeeded to find the solution in maximum 300 iterations in 99.88 % of the cases. The resulting number of iterations in the case when a solution was found is shown in Figure 4.2, and has a mean of 11.5 iterations.
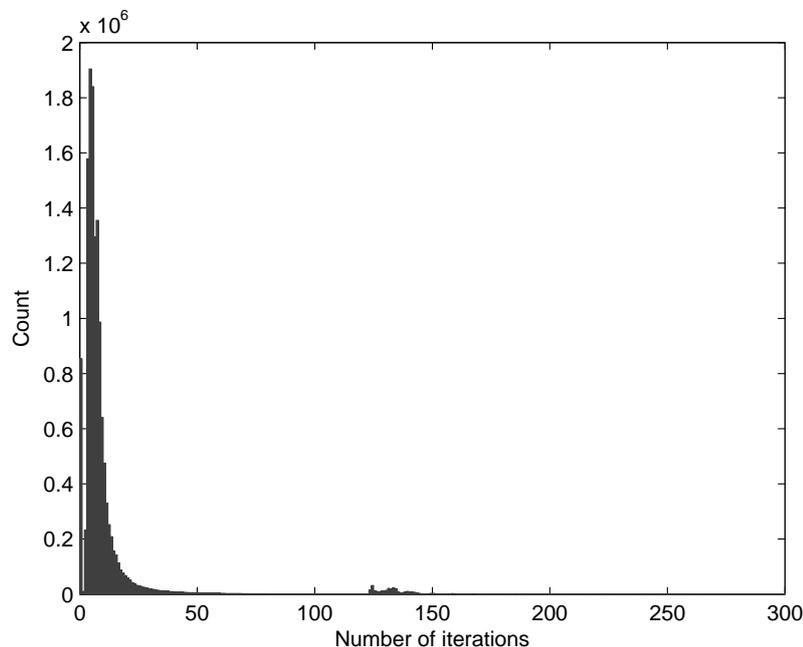
Figure 4.2.   Histogram of the number of iterations to run the Gauss-Newton optimization algorithm. Mean is 11.5 iterations.

For the rouhgly 16 000 remaining parameter-settings, the fixed-point algorithm was used. The mean is around $33 \cdot 10^3$ iterations and maximum is around $67 \cdot 10^3$ iterations. The result is shown as a histogram in Figure 4.3.

Now to the question; is any of these algorithms good enough to run in real-time? With a mean of more than 30 000 iterations, it is obvious that fixed-point iteration is not good enough. The Gauss-Newton algorithm is not so bad, though, with a mean of 11.5 iterations. There is no sharp limit of how many iterations that can be carried out in real-time, but since it usually takes more than 10 iterations, the solution is considered not good enough. A faster and more accurate solution may be to use off line generated solution values and find an analytical function that approximates
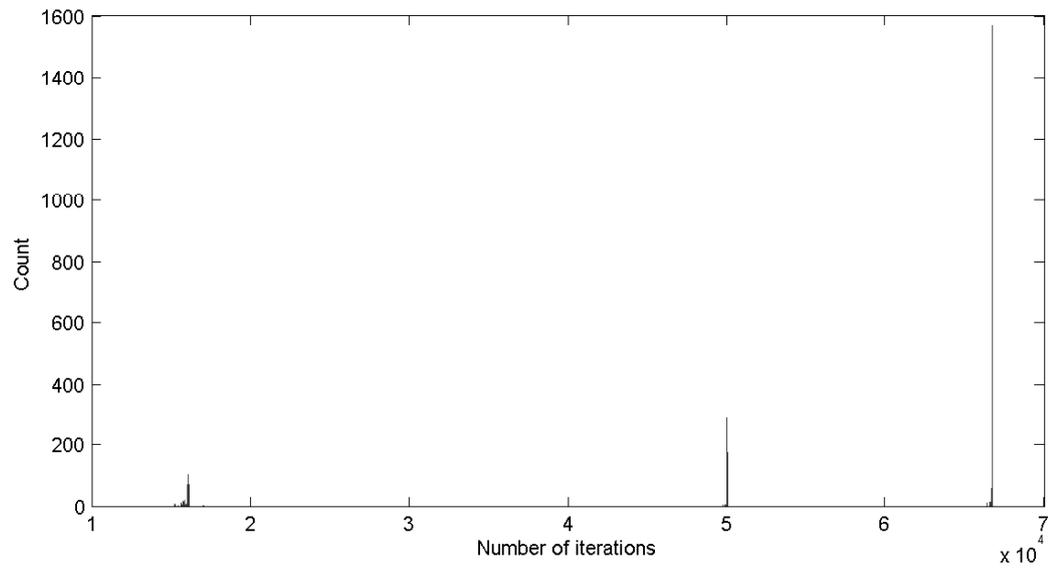
Figure 4.3. Histogram of the number of iterations to run the fixed-point iteration algorithm. Mean is 33 177 iterations. The reason for the low mean value is that the spikes to the right are very narrow.

these solutions as accurately as possible.

5 PARAMETERIZED SOLUTIONS

To reduce the complexity of the parameterization, the function to parameterize is defined as

$$g(a, b, c, d) = a + bx_1(a, b, c, d) + cx_2(a, b, c, d),$$ (5.1)

where x_1 and x_2 are solutions to (3.12) for specific choices of a, b, c and d. This is acceptable since x_1 and x_2 are easily calculated by using Equations (3.10) and (3.11).

5.1 Properties of the solution

The result of the numerical calculations is a large number of data-points, specifying the value of g in those points. One suitable way to present the data is to lock the values of a and d and plot the surface defined by g as a function of b and c. Figure 5.1 shows the surface plots for some combinations of a and d (see Figure 5.2 for the corresponding level-curves). Notice that the level curves are straight lines, which is possible to verify analytically:

Choose a fix level g_0, and use Equations (3.10) and (3.11):

$$g_0 = a + bx_1 + cx_2 = a + bf(g_0) + cf(dg_0).$$

Rearrange to get

$$c = -\frac{f(g_0)}{f(dg_0)}b + \frac{1}{f(dg_0)}(g_0 - a), f(dg_0) \neq 0$$ (5.2)

which shows the linear dependence between b and c for a fix level g_0 and constant a and d. $f(dg_0) = 0 \implies g_0 = 1/d$ for which the level-curves are straight lines along the c-axis.

5.2 Limits of g

It is suitable to lock $a = a_0$ and $d = d_0$ to watch the behaviour of g as a function of b and c. By using the implicit equations (3.10) and (3.11) together with the shape of f some conclusions are drawn.

$b = 0$, $c = 0$:
$$g(a_0, 0, 0, d_0) = a_0 + 0 \cdot x_1 + 0 \cdot x_2 = a_0$$
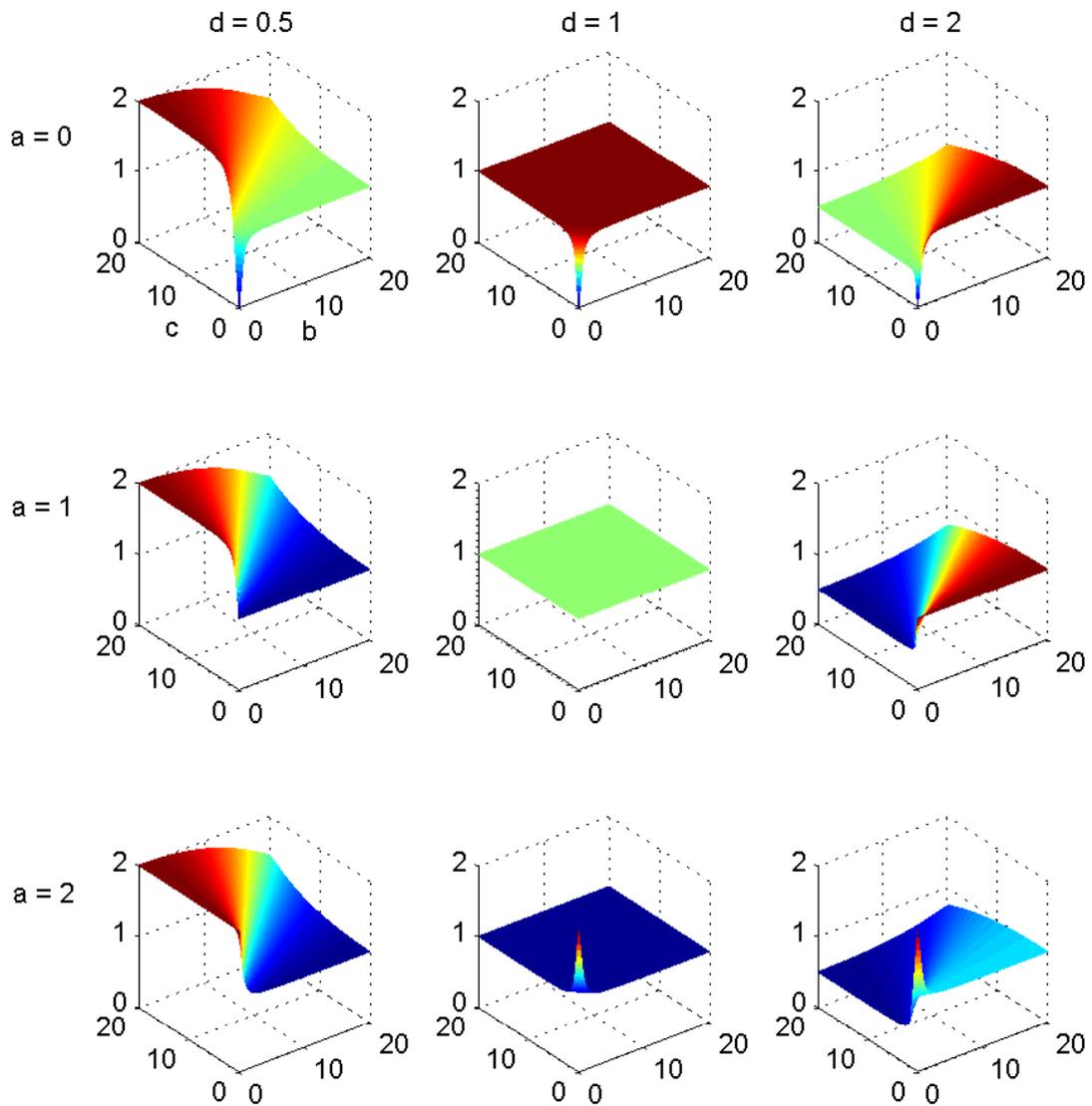
Figure 5.1.   The function g as a function of b and c, locked for some values of a and d. Note that $g = a$ in every origo.

$b \to \infty$, $c = 0$:

$$g(a_0, b, 0, d_0) = a_0 + bx_1 = a_0 + bf(a_0 + bx_1)$$

Let $\xi = a_0 + bx_1$, leading to

$$\xi = a_0 + bf(\xi).$$

Suppose there is a solution ξ_{lim} at the limit when $b \to \infty$. So $bf(\xi) \to \xi_{\text{lim}} - a_0$, $b \to \infty$, implying that $f(\xi) \to 0$ and thus $\xi \to 1$.
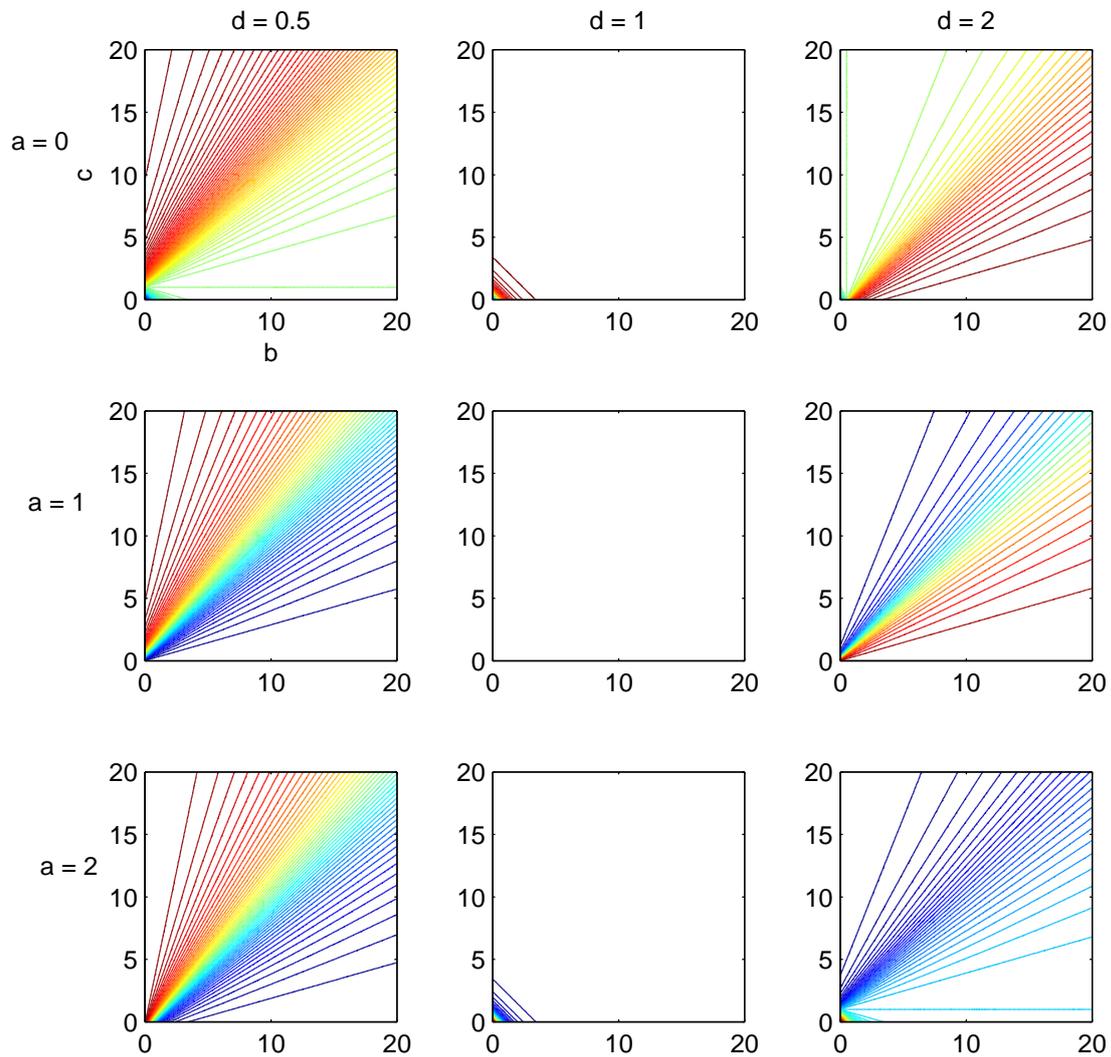
Figure 5.2.   Level plots of g as a function of b and c, locked for some values of a and d. The level curves are straight lines, which is analytically verified by Equation (5.2).

$b = 0$, $c \to \infty$:

$$g(a_0, 0, c, d_0) = a_0 + cx_2 = a_0 + cf(d_0(a_0 + cx_2))$$

Let $\xi = a_0 + cx_2$,

$$\xi = a_0 + cf(d_0\xi)$$

Using the same argument as above gives $d_0\xi \to 1$, $c \to \infty$, concluding that $\xi \to 1/d_0$.

$b = c$, $b \to \infty$:

$$g(a_0, b, b, d_0) = a_0 + bx_1 + bx_2 = a_0 + bf(a_0 + bx_1 + bx_1) + bf(d_0(a_0 + bx_1 + cx_2))$$

Let $\xi = a_0 + bx_1 + bx_2$ to get

$$\xi = a_0 + b(f(\xi) + f(d_0\xi))$$

As $b \to \infty$, $f(\xi) + f(d_0\xi) \to 0$. Using Equation (3.5), $f(\xi) \to -f(d_0\xi) = f\left(\frac{1}{d_0\xi}\right)$. For $\alpha_c \leq \xi \leq 1/\alpha_c$, f is monotonic, leading to the conclusion that the arguments are identical: $\xi \to 1/(d_0\xi)$, so $\xi \to \pm 1/\sqrt{d_0}$. Since $\xi > 0$, $\xi \to 1/\sqrt{d_0}$ as $b \to \infty$. Other solutions may exist in the region $\{\xi : \xi < \alpha_c \text{ or } \xi > 1/\alpha_c\}$. Examine both possibilities, first $f(\xi) = f(1/(d_0\xi)) = 1 \implies d_0 < \alpha_c$ and $1/(d_0\xi) < \alpha_c$, so $d_0 > 1/(\xi\alpha_c) > 1/\alpha_c^2 > 2$, which is the largest value we allow for d_0. In the same way, examine the possibility for $f(\xi) = f(1/(d_0\xi)) = -1 \implies d_0 < \alpha_c^2 < 0.5$, which is the lowest possible value for d_0.

The limits are concluded for the fixed values a_0 and d_0:

- $g = a_0$, $b = 0$, $c = 0$

- $g \to 1$, $b \to \infty$, $c = 0$

- $g \to 1/d_0$, $b = 0$, $c \to \infty$

- $g \to 1/\sqrt{d_0}$, $b = c$, $b \to \infty$

Those limits are important help when the task is to find a parameterized solution. Only solutions that behave correctly in the limits need to be taken into consideration.

5.3   Parameterization of the boundaries

Given a choice of a and d, the boundaries of g are the points on the b-axis and the c-axis. For choked flows, the flow is known, and f is either -1 or plus 1. Let us now examine each axis separately.

5.3.1   Parameterization of flow only through the intake-throttle

Since

$$c = \frac{t_s R T f_{\text{max,egr}}}{V P_{\text{egr}}}$$

there will be no flow through the EGR-throttle if $c = 0$. Given that $c = 0$, there is a known parameterization. The suggested parameterization is consistent with the

limits discussed in Section 5.2:

$$g_{\text{in}} = g(a, b, c, d)|_{c=0} =$$

$$1 - \frac{1}{1 + p_1 b^{p_2}} (1 - a)^{2 - e^{-p_3 b}}, \quad a \leq 1$$

$$1 + \frac{1}{1 + p_1 b^{p_2}} (a - 1)^{2 - e^{-p_3 b}}, \quad a > 1$$

A simple analysis for the choked flows gives the following conditions on g:

$$g_{\text{in}} = g(a, b, c, d)|_{c=0} =$$

$$a + b, \quad a + b \leq \alpha_c$$

$$a - b, \quad a - b \geq \frac{1}{\alpha_c}$$

The parameters p_1, p_2 and p_3 are optimized based on minimizing the relative error of the objective function, since it is very connected to the maximum error of the pressure in the manifold. Using `fminsearch` in Matlab for optimizing the parameters p_1, p_2 and p_3 with respect to minimum of the relative error gives $p_1 = 3.755$, $p_2 = 1.851$ and $p_3 = 2.399$. The resulting maximum relative error is 2.2 %. See Figure 5.3, where g_{in} is plotted together with the reference points for some different values of a.
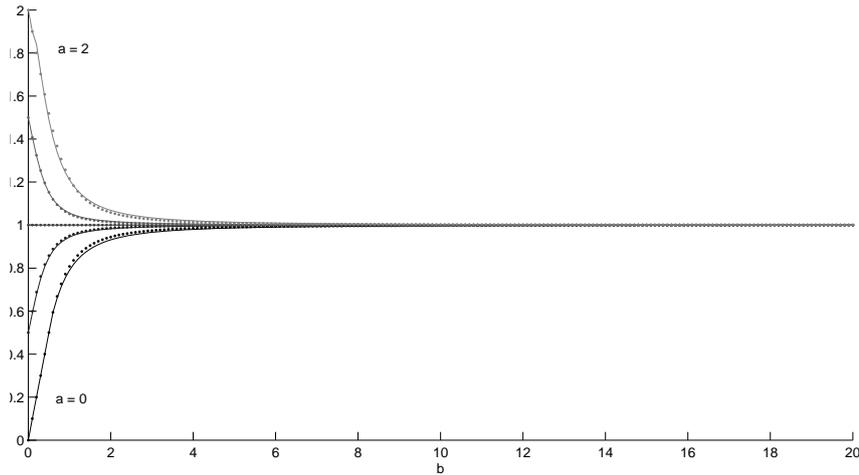
Figure 5.3.   The optimized g_{in} (solid line) for $a = 0$, 0.5, 1, 1.5, and 2 together with corresponding reference points (dashed line).

5.3.2 Parameterization of flow only through the EGR-throttle

Assume that g behaves similarily on the c-axis. Using the knowledge of the limits for $b = 0$ a proposal for parameterization is

$$g_{\text{egr}} = g(a, b, c, d)|_{b=0} =$$
$$\frac{1}{d} - \frac{1}{1 - p_4 c^{p_5}} \left(\frac{1}{d} - a \right)^{2 - e^{-p_6 c}} , \quad a \leq \frac{1}{d}$$
$$\frac{1}{d} + \frac{1}{1 + p_4 c^{p_5}} \left(a - \frac{1}{d} \right)^{2 - e^{-p_6 c}} , \quad a > \frac{1}{d}$$

and the conditions for choked flows are in a similar way

$$g_{\text{egr}} = g(a, b, c, d)|_{b=0} =$$
$$a + c, \quad a + c \leq \frac{\alpha_c}{d}$$
$$a - c, \quad a - c \geq \frac{1}{\alpha_c d}$$

As it turns out, it does not work very well (maximum error of 12%). That is because the function does not fit well in connection with the choked flows. As an attempt to adapt to the choked regions, the variables are biased according to the following condition to make the function smooth with respect to the conditions for choked flows:

$$c_0 = \begin{cases} \max(\alpha_c/d - a, \, 0), & a \leq 1/d \\ \max(a - 1/(\alpha_c d), \, 0), & a \geq 1/d \end{cases}$$

and

$$g_0 = \begin{cases} \max(\alpha_c/d, \, a), & a \leq 1/d \\ \min(\frac{1}{\alpha_c d}, \, a), & a \geq 1/d \end{cases}$$

c is then substituted by $c - c_0$ and a is substituted by g_0. An optimization results in a maximum error of 7.2% with the parameter settings $p_4 = 2.546$, $p_5 = 1.628$ and $p_6 = 3.918$. Figure 5.4 shows the function for some different parameter values.

5.4 Parameterization of the flow through the intake- and EGR-throttles

The idea is to connect the boundaries in a way that gives an approximation of g for all possible parameter combinations, meaning that there is flow through both the intake- and the EGR-throttle simultaneously. One idea is to do a weighted mean of the function values on the boundaries.
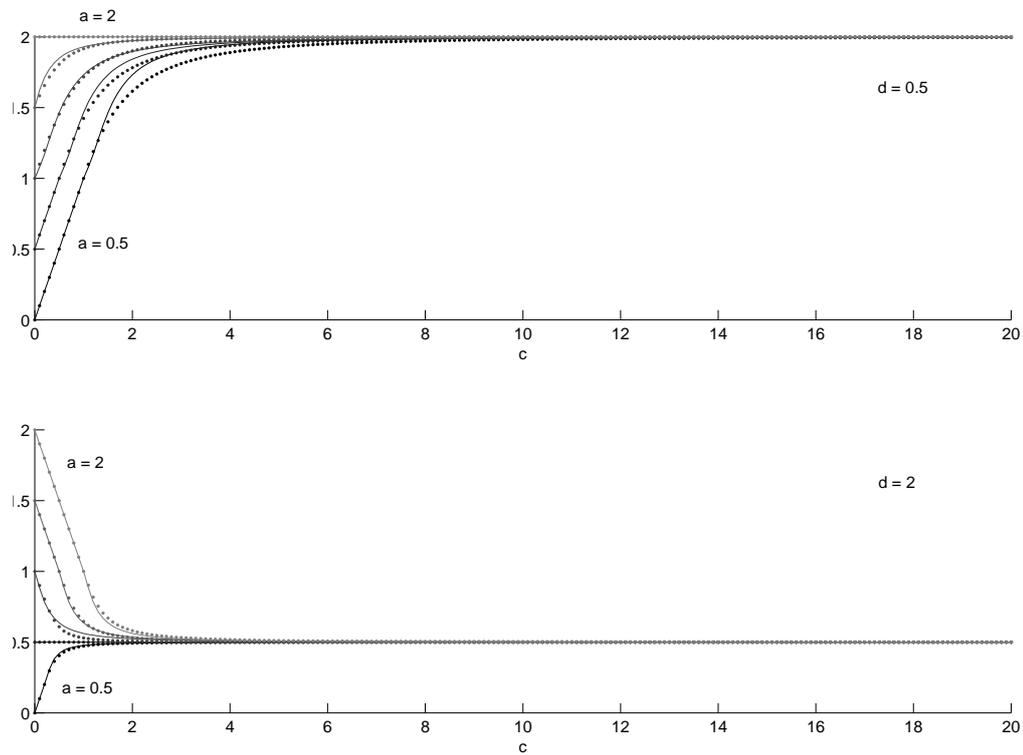
Figure 5.4.   The optimized g_{egr} (solid line) for $d = 0.5$ and 2 in two different plots together with corresponding reference points (dotted line). $a = 0, 0.5, 1, 1.5,$ and 2 in each plot.

5.4.1   Weighted mean of the boundaries

Suppose that at least one of b and c are non-zero. Then the function g is

$$g_{\mathrm{bound}}(a,b,c,d) = \frac{b\, g_{\mathrm{in}}(a,b,c,d) + c\, g_{\mathrm{egr}}(a,b,c,d)}{b+c} \,, \tag{5.3}$$

where g_{in} and g_{bound} are defined in Section 5.3. Testing on all reference points results in a maximal error of 43.5% and a mean error of 5.55%. The distribution of the error is shown in Figure 5.5.
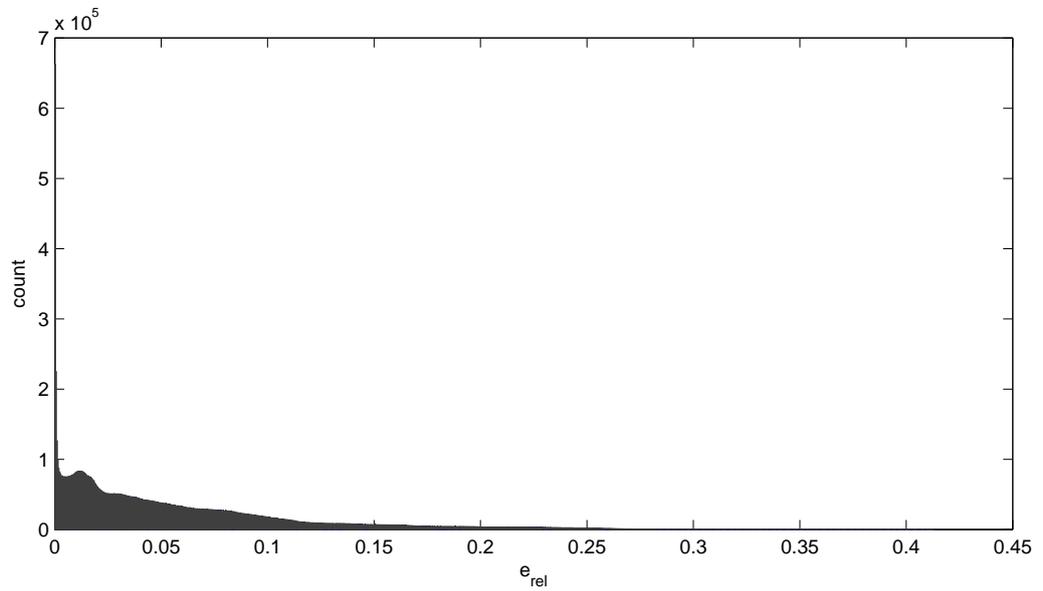
Figure 5.5.   The distribution of the relative error for g_{bound}. Maximum error is 43.5% and mean error is 5.55%.

6 SIMULATION AND PARAMETERIZA-TION

None of the proposed methods;

- Numerical algorithm running in real-time

- Parameterized function

works very well. Another alternative might be to simulate the pressure in real-time, using the measured states of earlier time-samples. This method is used by some already existing commercial softwares on the market. The problem is that current solutions do not handle the situation around $\alpha = 1$ well, where another strategy has to be implemented.

Since methods based on simulation are already in use, a possible algorithm will be discussed only briefly. Focus will be on the sensitive areas, more specifically when at least one of the pressure ratios around the nozzles are between 0.95 and 1/0.95. In those areas, a function approximation directly on f will be used. In the case when both pressure coefficients are close to 1, the parameterization of f is difficult to apply, and a parameterized solution based on the boundaries will be used.

We will also reduce the problem according to the fact that back-flow is never allowed in the intake-throttle. That means that P/P_{in} is never larger than 1.

6.1 Pressure simulation

Assume that the pressure ratios P/P_{in} and P/P_{egr} are far from 1. The pressure in the manifold is then simulated using Equation (3.9):

$$P^{(n)} = P^{(n-1)} + t_s \frac{RT}{V} (\dot{m}_{\text{in}} + \dot{m}_{\text{egr}} - \dot{m}_e).$$

The flows $\dot{m}_{\text{in}}$, $\dot{m}_{\text{egr}}$ and $\dot{m}_e$ must therefore be simulated as well. To do this correctly, we need to model not only the parts described in this thesis, but parts in the cylinder and exhaust system as well. Since this is outside the scope of this study, no details will be presented. It is worth mentioning, though, that this is a known problem and that there are solutions implemented in commercial software.

In each sample the pressure ratios are estimated to be able to detect if a pressure ratio for one of the nozzles approaches 1. If so, the corresponding flow in (3.9) has to be calculated more carefully. At first, we assume that only one of the ratios is close to 1, say P/P_{in}, while P/P_{egr} is in a region where simulation is possible.

6.2 Parameterization of f

Recall the definition of $f(\alpha)$, see Equation (3.5). Let $\gamma = 1.4$. Then, for $0.95 < \alpha \leq 1$ the function behaves very much like the function $k\sqrt{1-\alpha}$. Extending to negative flow as well, gives

$$f(\alpha) \approx \hat{f}(\alpha) = \begin{cases} k\sqrt{1-\alpha} & , 0.95 < \alpha \leq 1 \\ -k\sqrt{1-\frac{1}{\alpha}} & , 1 < \alpha < 1/0.95 \end{cases} \qquad (6.1)$$

Optimizing according to mean-squared regression leads to $k = 2.0278$, with a maximum relative error of 1.8%. Figure 6.1 shows f and $\hat{f}$ in the specific interval.
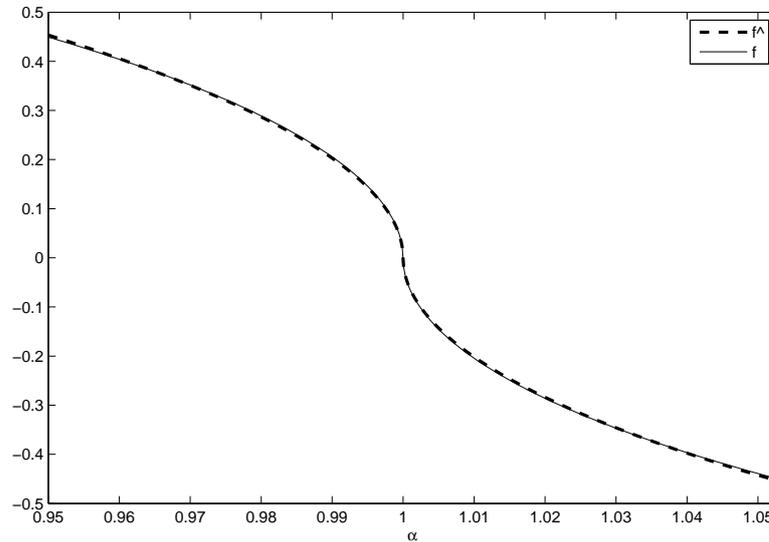
Figure 6.1. The function f, see Equation (3.5), and $\hat{f}$, see (6.1), in the interval $0.95 < \alpha < 1/0.95$.

6.3 Simulation combined with parameterization of f

Let $0.95 < P/P_{\text{in}} \leq 1$ and P/P_{egr} be far from 1. x_2 is then possible to simulate. The parameterization of f from (6.1) is used to obtain the equation

$$x_1 = f(a + bx_1 + cx_2) = k\sqrt{1 - (a + bx_1 + cx_2)}.$$

Collect terms to get a second-order polynomial equation in x_1:

$$x_1^2 + bk^2 x_1 + k^2(cx_2 + a - 1) = 0.$$

The solutions are then

$$x_1 = -\frac{bk^2}{2} \pm \sqrt{\frac{b^2 k^4}{4} - k^2(cx_2 + a - 1)}\,.$$

The flow $\dot{m}_{\text{in}}$ is assumed to be positive, and so must x_1 be according to (3.7), so the positive solution must be chosen. If, for simplicity, x_2 is supposed to be correct by simulation, the relative error in pressure is maximum 0.10% and has a mean of 0.020%.

Now, let P/P_{in} be far from 1, and thus x_1 is simulated. Assume instead that $0.95 < P/P_{\text{egr}} \le 1$. A similar analysis shows that

$$x_2 = -\frac{cdk^2}{2} + \sqrt{\frac{c^2 d^2 k^4}{4} - k^2(bdx_1 + ad - 1)}\,.$$

A slightly more complicated case is when there is a flow backwards through the EGR, meaning that $1 < P/P_{\text{egr}} < 1/0.95$. Continue to assume that $P/P_{\text{in}} < 0.95$. Our polynomial equation will then be one degree higher:

$$cdx_2^3 + (ad + bdx_1)x_2^2 - cdk^2 x_2 + k^2(1 - ad - bdx_1) = 0\,.$$

Polynomial identification gives

$$Q_1 = cd\,,\ Q_2 = ad + bdx_1\,,\ Q_3 = -cdk^2\,,\ Q_4 = k^2(1 - ad - bdx_1)\,.$$

The three solutions to the cubic equations are known (Weisstein 2011). Let $Q_1 = cd \ne 0$ and

$$\begin{aligned}
S_1 &= 2Q_2^3 - 9Q_1 Q_2 Q_3 + 27Q_1^2 Q_4 \\
S_2 &= S_1^2 - 4(Q_2^2 - 3Q_1 Q_3)^3
\end{aligned}$$

The possible solutions are then

$$x_2^{(1)} = -\frac{Q_2}{3Q_1} - \frac{1}{3Q_1}\left(\frac{1}{2}\left(S_1 + \sqrt{S_2}\right)\right)^{\frac{1}{3}} - \frac{1}{3Q_1}\left(\frac{1}{2}\left(S_1 - \sqrt{S_2}\right)\right)^{\frac{1}{3}} \quad (6.2)$$

$$x_2^{(2)} = -\frac{Q_2}{3Q_1} + \frac{1 + i\sqrt{3}}{6Q_1}\left(\frac{1}{2}\left(S_1 + \sqrt{S_2}\right)\right)^{\frac{1}{3}} - \frac{1 - i\sqrt{3}}{6Q_1}\left(\frac{1}{2}\left(S_1 - \sqrt{S_2}\right)\right)^{\frac{1}{3}} \quad (6.3)$$

$$x_2^{(3)} = -\frac{Q_2}{3Q_1} + \frac{1 - i\sqrt{3}}{6Q_1}\left(\frac{1}{2}\left(S_1 + \sqrt{S_2}\right)\right)^{\frac{1}{3}} - \frac{1 + i\sqrt{3}}{6Q_1}\left(\frac{1}{2}\left(S_1 - \sqrt{S_2}\right)\right)^{\frac{1}{3}} \quad (6.4)$$

To know which solution to choose, compare with the solutions calculated by the numerical algorithms in Chapter 4. Figure 6.2 contains the error with respect to the reference values. Evidently, (6.4) is the solution to our problem. The maximum relative error is 0.67% and the mean error is very small, 0.016%.

Now examine the case when $Q_1 = cd = 0$, meaning $c = 0$ or $d = 0$. c and d represent physical properties that in practice never are zero. We may also note that we consider the case where $1 < d < 1/0.95$, so d is never close to zero.
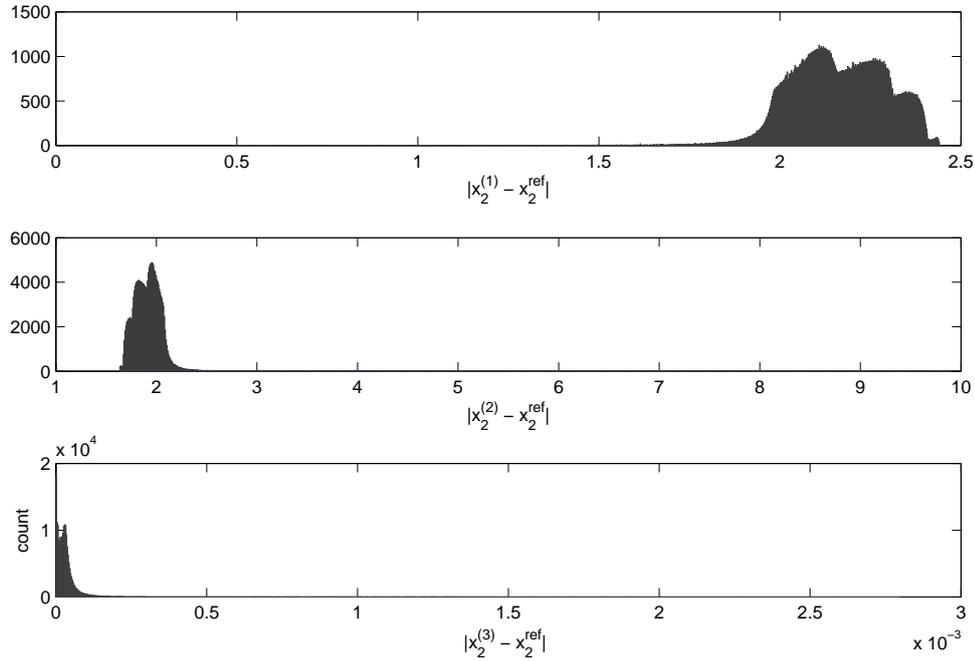
Figure 6.2. Errors for the three possible solutions of the cubic equation. $x_2^{(3)}$ is obviously the solution.

6.4 Parameterization for small pressure differences

When both pressure ratios are close to 1, the simplification with $\hat{f}$ is not enough to solve the problem analytically. Then a parameterization is used, as was discussed in chapter 5.

We are now limited to the region where

$$0.95 < P/P_{\text{in}} \le 1 \text{ and } 0.95 < P/P_{\text{egr}} < 1/0.95 \,.$$

It turns out that only a small part of the generated reference points satisfy the condition. Especially, the values of d only take values very close to 1, why the stepsize between the reference points is decreased to 0.2 in d.

6.4.1 Parameterization outgoing from the boundaries

g_{bound} defined by (5.3) is used in this region, optimized with the result of a new parameter-setting: $p_1 = 2.949$, $p_2 = 1.984$, $p_3 = 1.190$, $p_4 = 3.089$, $p_5 = 1.806$ and $p_6 = 1.192$. The maximum error in this region is 3.2%, and the mean error is 1.1% (see Figure 6.3).
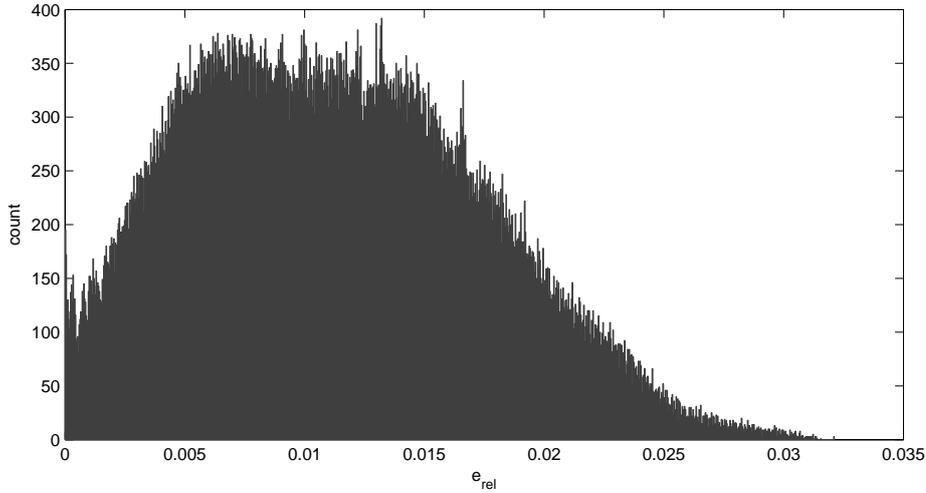
Figure 6.3.   The distribution of the error of g_{bound} in the region where $0.95 < P/P_{\text{in}} \le 1$ and $0.95 < P/P_{\text{eqr}} < 1/0.95$.

6.4.2   Parameterization according to level-curves

Equation (5.2) describes the equation for the level-curves in the bc-plots:

$$c = -\frac{f(g_0)}{f(dg_0)}b + \frac{1}{f(dg_0)}(g_0 - a)\,, f(dg_0) \ne 0$$

The idea behind this parameterization is to include the information about the level-curves to get an alternative approximation of g. Given values of a, b, c, and d, the idea is as follows:

- Calculate the slope Δ in the $b - c$-plot (see Figure 5.2).

- Get the level g by

$$\Delta = -\frac{f(g)}{f(dg)}\,.$$

Getting the Δ-value out of a, b, c and d is the tricky part. Actually, making it correct means that our problem is solved in general. The easiest approximation is to assume that all level-curves go through a common point $[b_0, c_0]^T$. Then,

$$\Delta = \frac{c - c_0}{b - b_0}\,.$$

One idea is to use the origin, but in order to avoid numerical problems arising when dividing by zero our choice is $b_0 = c_0 = -10^{-2}$. We have that

$$\Delta = -\frac{f(g)}{f(dg)} > 0\,,$$

and together with the approximation of f defined by (6.1):

$$\Delta = \sqrt{\frac{1-g}{1-\frac{1}{dg}}} \; .$$

Note that a result of the approximation above is that we have to assume back-flow in the EGR, which is not always true. Solve for g and choose the positive solution,

$$g_{\text{level}} = -\frac{\Delta^2 - 1}{2} + \sqrt{\frac{(\Delta^2 - 1)^2}{4} + \frac{\Delta^2}{d}} \; . \tag{6.5}$$

6.4.3 Combination of g_{bound} and g_{level}

The solution g_{level} is in general bad, but used in combination with g_{bound}, it turns out to decrease the error a lot. On the boundaries, g_{bound} is superior compared to g_{level}, but in the interior, g_{level} does help. Let g_{comb} be a combined solution defined by

$$g_{\text{comb}} = p_{\text{level}} \, g_{\text{level}} + (1 - p_{\text{level}}) \, g_{\text{bound}} \; , \tag{6.6}$$

where p_{level} is a ratio determining how much we trust in g_{level}, defined by

$$p_{\text{level}} = \frac{\ln\left(1 + \sqrt{b^2 + c^2}\right)}{\ln\left(p_7\right)} \; .$$

Here, p_7 is a trust radius, and should be adjusted to achieve the best fit.

Optimizing g_{comb} results in a new set of parameters p_1 - p_7: $p_1 = 3.450$, $p_2 = 1.221$, $p_3 = 0.1639$, $p_4 = 5.913$, $p_5 = 1.446$, $p_6 = 2.532$ and $p_7 = 76.94$. The maximum relative error of g_{comb} is 2.51% and mean error is 0.69%, which is a noticeable improvement. The distribution of the error is shown in Figure 6.4.

6.5 Real-time determination of region

In the proposed solution, the pressure is estimated differently depending on the value of the ratios P/P_{in} and P/P_{egr}. It is crucial that the algorithm running in real-time has the ability to detect when a region is changed.
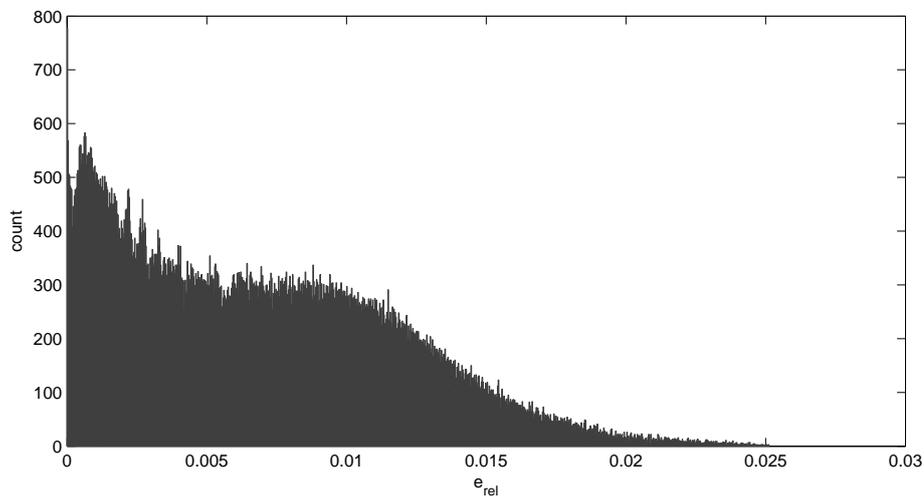
Figure 6.4.   The distribution of the error of g_{comb} in the region where $0.95 < P/P_{\mathrm{in}} \le 1$ and $0.95 < P/P_{\mathrm{eqr}} < 1/0.95$.

7 DISCUSSION

The results in this thesis work are compared to an ideal reality, that is the result given by the basic equations of flows and pressures. Even if the model and the reference perfectly agree, a comparison to reality may display a difference.

In all proposed models, the integration method used is backward Euler, which is known to cause stability issues in some conditions. Another integration method will, if the methods of this work are used, probably result in algebraic equations that are more difficult to solve than the equations resulting from Euler integration.

An interesting approach for a future work is to design a nonlinear observer.

In this work, no details about the simulations have been examined. Simulations involve the cylinder parts, the exhaust system and possibly a turbocompressor or heat-exchangers. These simulations may have other complications apart from the manifold, affecting the flows and the pressures. This has not been taken into account, and in the error calculations the simulation parts are assumed to result in theoretically correct values.

Concerning the control, is model-based control really necessary, or may other approaches be suitable as well? In Amstutz and Del Re (1995), a linear sensor-based control method is implemented. One conclusion in the article is that for transient operations, feedforward controllers tend to yield better results if the plant-model mismatch is small. Sensors are not fast enough to catch the fast nonlinear dynamics of the system. So a good model matters.

8   CONCLUSION

The goal of this thesis work was to design a model or an algorithm for real-time estimation of the pressure P in the intake manifold in diesel engines. The pressure P_{in} just before the intake-throttle and the pressure P_{egr} just before the EGR-throttle are assumed to be known.

Two approaches to fulfill that goal have been presented; numerical methods and parameterization methods.

The numerical methods are designed using different techniques. The most successful method implemented is the Gauss-Newton method for optimization (with a few modifications), which succeeded in 99.9% of the cases tried with a mean of 11.5 iterations. In 88% of the cases, the algorithm used less than 15 iterations. Even 15 iterations are, however, probably too many to run in real-time. The real trouble is that the nonlinear dynamics makes the solutions very sensitive to small changes in pressure. The character of the problem is such that it is difficult to design a numerical method that is guaranteed to converge to a solution in finite time. For future work, other optimization methods may be tried, for example direct search methods. The challenge with direct search methods is to try to limit the computation time, which may be hard. It is also difficult to formally verify the behaviour of such methods.

Among the parameterizations, none of the proposed solutions is good enough for all possible conditions. Limited to different pressure regions, different methods succeed relatively well, however. The problem is to determine in which region the engine is operating. Table 8.1 shows the proposed solutions and in which regions they work. The table also shows that the solution g_{bound} based on parameterization of the boundaries for the whole definition set does not succeed.

Table 8.1.   Summary of the results for the proposed parameterized solutions. $e_{rel,mean}$ and $e_{rel,max}$ refer to the absolute relative error in the resulting pressure P of the intake manifold.

Function	P/P_{in}	P/P_{egr}	$e_{rel,mean}$	$e_{rel,max}$
g_{bound}	free	free	5.55%	43.5%
$\hat{f}$	(0.95,1]	≤ 0.95 and $\geq 1/0.95$	0.020%	0.10%
$\hat{f}$	≤ 0.95	(0.95,1]	0.0074%	0.082%
$\hat{f}$	≤ 0.95	(1,1/0.95)	0.019%	0.10%
g_{comb}	(0.95,1]	(0.95,1/0.95)	0.69%	2.51%

The overall conclusion is that no general function or algorithm has been found that satisfies the conditions on accuracy and complexity. If divided into sub-regions of the pressure-ratios, the relative error is small. Further investigations have to be made to find a solution that could be implemented in real-time with good performance.

Hopefully, some of these conclusions may serve as inpiration and help for future work, aiming to find a method that works in real-time.

Bibliography

Amstutz, A. and Del Re, L. R. (1995). Ego sensor based robust output control of egr in diesel engines. *IEEE Transactions on Control Systems Technology, vol. 3, no. 1*.

Andreasson, N., Evgrafov, A. and Patriksson, M. (2005). *An Introduction to Continuous Optimization*. Studentlitteratur. Lund.

Frenk, J. B. G. and Kassay, G. (2001). Introduction to convex and quasiconvex analysis. *Econometric Institute Report*.

Guzzella, L. and Onder, C. H. (2010). *Introduction to Modeling and Control of Internal Combustion Engine Systems*. Springer-Verlag. Berlin.

Heywood, J. B. (1988). *Internal Combustion Engine Fundamentals*. McGraw-Hill. Publisher. USA.

Press, W. H., Teukolsky, S. A., Vetterling, W. T. and Flannery, B. P. (2007). *Numerical Recipes: The Art of Scientific Computing, pp. 492-496*. Cambridge University Press. New York.

Weisstein, E. W. (2011). Cubic formula. `http://mathworld.wolfram.com/CubicFormula.html`.