Master of Science Thesis in Electrical Engineering
Department of Electrical Engineering, ISY, Linköping University, 2021
LiTH-ISY-EX–21/5373–SE

Estimation of Stability Margins for the Closed-Loop Air Charge Control of an Internal Combustion Engine Using Sinusoidal Disturbances

Victor Sundelin & Filip Jussila

June 11, 2021

LINKÖPINGS UNIVERSITET

Project Identity

Customer:	Fredrik Wemmert, VCC/PCC
	E-mail: fredrik.wemmert@volvocars.com
	Phone: +46723716110
Course Responsible:	Lars Eriksson, Linköping University
	E-mail: lars.eriksson@liu.se
Supervisors:	Olov Holmer, Linköping University
	E-mail: olov.holmer@liu.se
	Marcus Rubensson, VCC/PCC
	E-mail: marcus.rubensson@volvocars.com
	Phone: +46729775165
Lab responsible:	Tobias Lindell, Linköping University
	E-mail: tobias.lindell@liu.se
	Phone: +46709306929

Group Members

Initials	Name	Phone	E-mail
FJ	Filip Jussila	+46768015124	filip.jussila@gmail.com
VS	Victor Sundelin	+46707915902	victor.sundelin@gmail.com

Abstract

The vehicle industry have for many years improved the design of car motors and iterated the control systems associated with it. The systems have become very complex and hard to understand because of this work process. It is today very difficult to perform evaluations of the engine's performance or components theoretically at Powertrain Engineering Sweden (PES). This thesis proposes a test method to estimate the robustness, in terms of stability margins, of the air charge throttle control loop using measurement data. Alternative test methods are also presented, for example system identification performed with MATLAB's SITB.

The proposed test method superimposes a sine wave upon the control signal in a closed loop system. The control signal is measured after it is superimposed and after it have made one round trip around the loop. These two signals is regarded as sine in and sine out. The phase shift and relation in amplitude are estimated from the measurements and the robustness is presented by Bode plots. The method finds the phase shift from the time difference between the zero-crossings of the input- and output signal. The relation in amplitude is found by looking at the total sum of the absolute value sine wave.

Extensive testing with different tunings of the P-part of the air charge controller shows that the proposed method correctly identifies if the systems stability margins have become larger or smaller. For nine measurements with different P-tunings it is seen that the magnitude curves stay separate throughout the whole Bode plot. It is also shown that the gain margins are decreasing for every increase in P-value. The overall results and findings in this thesis are promising and can act as a foundation for future thesis' work to come.

Acknowledgements

We would like to thank our supervisor Olov Holmer at Linköping University for his help and guidance throughout this thesis. We would also like to thank Lars Eriksson for connecting us with Volvo and his words of encouragement.

Thanks to Marcus Rubensson and Fredrik Wemmert at PES for this opportunity and every fruitful discussion and support.

As a closing remark we would like to praise Tobias Lindell who made sure that the practical work in this thesis was possible every week and for sharing his knowledge and experience.

Linköping, May 2021
Victor Sundelin and Filip Jussila

Contents

List of definitions

- **PES** - Powertrain Engineering Sweden, an affiliate of Volvo Cars Göteborg.
- **LiU** - Linköping University.
- **ICE** - Internal combustion engine.
- **SITB** - MATLAB's System Identification Toolbox.
- **MIMO** - Multiple inputs multiple outputs.
- **SISO** - Single input single output.
- **IMC** - Internal model control.
- **RGA** - Relative gain array.
- **PRBS** - Pseudorandom Binary Sequence.
- **Plant** - The physical system which is being controlled.
- **PID** - Proportional, Integral, Derivative. The most common used controller in industry.

Signal abbreviations

- **Hw** - sVcAesHw_X_Thr_Tar
- **Ch** - sVcAesCh_X_ThrPosnTar
- **PCmp** - tVcAesCt_fac_ThrCtrlPCmp
- **ThrTar** - sVcAesCt_m_ThrTar
- **CylTar** - rVcAesCt_m_CylTarDyn

1 Introduction

This master thesis was a cooperation between Linköping University and Powertrain Engineering Sweden which is an affiliate of Volvo Cars. The thesis is written by two students at LiU during the course of 20 weeks with the help of one supervisor from both PES and LiU.

1.1 Background

Engine tests are used to evaluate performance of components, amount of emissions, fuel consumption and other requirements that a modern engine has to fulfil. These tests are also used to evaluate design changes in terms of new components or new operational requirements and determine how these changes affects the overall performance of the engine. Engine tests are therefore a crucial part of an engine's development.

To optimise the performance of the engine, calibration and tuning of control parameters is necessary. This is a continually ongoing process during the development of a new engine, where adjustments and refinements are made to achieve a mature and robust performance for the given engine hardware. When changing the control parameters, the robustness of the controller most often changes and the stability of the system can no longer be guaranteed [1].

An engine consists of complex systems and components, often with associated control systems. Engine development is therefore typically distributed into different sub-systems, each being optimised separately. Because of this, different teams are often working on different parts of the same engine and it is important that the control system remains stable and robust when these parts are combined. Because of this and the numerous interactions between the different sub-systems and different control loops, it is difficult to guarantee robustness of the overall engine control.

Currently there are no efficient tests methods used at PES, to validate the complete system's robustness. It also takes a lot of effort to investigate an issue that arises in the engine control, for example a stability problem. The cause of the problem can be difficult to identify, because a single control loop will often affect many other control loops. Therefore the question,

- Is it possible to determine the robustness of a control loop efficiently?

has been brought to light. To be able to determine robustness after each parameter change would both save a lot of time and assure test engineers that the system will not become unstable. It could also create the possibility to be able to run real engine tests without supervision if they could ensure that the system would be robust and remains stable for different parameter changes which can not be done today.

1.1.1 Purpose

An engine is a complex structure which contains numerous control loops and it is not possible to describe all of them in this thesis. This is therefore, a broad thesis that in the future will act as a foundation for future work at PES or by other master's thesis students. To narrow down the scope of this thesis, the focus will be put on the air flow control loop, more precisely the throttle servo control loop. Apart from creating a foundation for future work the purpose is also to create a method for determining stability margins of the throttle servo control loop. This method should be able to produce gain- and phase margins from measurement data collected from a real engine. The final method will be the product of iterative testing to prove it's validity.

The majority of the work of this thesis will be carried out in the engine laboratory with practical tests on the real engine. Measurement data will be collected and used for determination of the stability margins for the throttle control loop and distinguish if a parameter change will increase or decrease the robustness. The purpose of all the practical measurements is to find good tunings and stationary work points where the developed method is able to produce the stability margins. This, would give the teams at PES a way of knowing if their change in their control loop affects the robustness of the feedback system and could therefore prevent future complications with the work of other teams.

1.2 Problem formulation

This thesis' main goal is to develop a method to estimate the robustness of a throttle servo control loop using experimentally collected data. The thesis' goal is also to determine the change in robustness when changing control parameters. The type of measured data, how it is collected and how it is treated to produce results are all important topics.

- What is a good method to estimate robustness using experimentally measured sine waves?
  - How can the measurement noise be treated?
  - How can the stability margins be validated?

It would also be interesting to compare the estimated robustness from experimental data with an estimate made from system identification methods using a PRBS-signal. This, to determine which of the two methods that is the best choice in terms of simplicity and accurate results.

- Can system identification be used to estimate robustness of the throttle servo control loop?

1.3 Related work

This section functioned as the foundation of ideas which were investigated through out the work of this thesis. The different articles and papers it contains relates to the purpose and problem formulation of this thesis and were chosen because of their relevance to the subject. The different topics it contains are system identification and robustness analysis. The general idea of the articles is presented briefly and compared to each other for similarities and how they would relate to this thesis.

1.3.1 System identification

An important part when analysing plants is system identification since a precise model of the real system yields more reliable data when for example the gain- and phase margins are calculated during a robustness analysis. It is also a useful tool for obtaining a model of an examined system where the governing equations are unknown. A common tool which was used for the system identification was MATLAB's SITB where it is easy to fit models to measurement data efficiently. SITB can be used for both open-loop identification [2] and closed-loop identification [3]. It is a useful tool for determining dynamic equations which can describe both linear and nonlinear systems [4]. In this thesis only linear system identification was considered.

A common method to produce gain- and phase margins of a system to describe its robustness is by using sine waves either at static frequencies or by sweeping them over numerous frequencies with a ramp. Sine waves are often used in electrical measurements where the amplitude and phase are interesting parameters to estimate [5], [6]. It exists methods for estimating these parameters, for example with frequency domain least-square approaches for linear and nonlinear systems which was used in [5] for complex sine waves. In [7], a sine wave was superimposed on a signal where it was considered as noise. In this thesis a sine wave was superimposed as well where the method for determining amplitude and phase for the input- and output signal was developed. In [8], a sine signal was used to determine a Bode plot for a linear controller where the output's frequency and amplitude was measured. Since it was a linear system that was being analysed in [8] they used the relationship in equation (1).

$$\begin{cases} u(t) = A\sin(\omega t) \\ y(t) = |G(i\omega)|A\sin(\omega t + \varphi) \end{cases} \tag{1}$$

When analysing sine waves the peaks or zero-crossings can be used for calculating the phase lag between the input- and output signal. The zero-crossings of a sine wave can be deemed more robust than the peaks when there exist unknown nonlinear disturbances [9]. The zero-crossings were therefore investigated in this thesis since the investigated system contained nonlinearities.

When performing the identification experiments, white noise or a PRBS-signal is often used as input signal when measuring the output signal [2]. The PRBS-signal is easier to use compared to the white noise signal in practise because of its simple implementation and reproducibility [10]. When choosing the input prior a collection of measurement data, amplitude and sampling frequency are especially important parameters to specify. Both the amplitude and sampling frequency of the PRBS-signal can be changed and it is also possible to include the bias and variance of this signal [11]. When performing system identification, the input signal should be chosen so that the output signal of the system is larger than the sensor noise for good identification properties [11]. This was accounted for in this thesis where the approach for determining the characteristics of the PRBS-signal followed the proposed method in [11]. This was an interesting approach for determining robustness parameters and was used as a comparison to the results produced by using a sine wave as input with the proposed method in this thesis for determining the stability margins. In an ideal world the gain- and phase margins should be the same within a small margin of error and thus it was interesting to investigate if that was the case for the examined system.

1.3.2 Robustness analysis

To determine if a system is on the verge of instability a robustness analysis is useful. It is commonly performed when designing a system but can also be important when control parameters have been changed online. Gain- and phase margin are two common parameters to identify when performing a robustness analysis since they are fundamental measures of robustness [12]. The gain- and phase margins serves as maximum uncertainties. The gain margin for a system with proportional control is the maximum value which the proportional controller can be multiplied with and the phase margin corresponds to the amount of time delay the system can withhold without risking instability [13]. In [13], a method for determining the gain- and phase margin was proposed for a proportionally controlled SISO-system. The system investigated in this thesis however, could not be seen as a proportionally controlled system even when the D-part of the PD-controller was turned off. This thesis' control system was more complex than the system investigated in [13] and consist of two parallel controllers, one PD-controller and one separate I-controller. However, it was possible to determine the effect of a change of the proportional control parameter in the investigated system once the D-part of the PD-controller was turned off because of the sensors located on the engine. The proposed method for determining the maximum gain margin in [13] could be applied on this closed-loop system once the region of stable gains was determined.

In both [14] and [15], the throttle of an ICE was investigated more thoroughly compared to [12] and [13]. The throttle's nonlinearities and gear backlash gave rise to a difficult control task to solve and thus, [14] chose to use a simplified model of the throttle before they used the Robust Control Toolbox from MATLAB. A model of the throttle was not used in this thesis and therefore the same simplification as in [14] could not be made. In [15], a PD-controller was used similar to the setup investigated in this thesis. A method to guarantee robust control was presented for the single PD-controller and would have been of interest if it would have been possible to turn off the I-controller. The proposed robust control designs in both [14] and [15] were defined robust based on their ability to be exposed to disturbances and still produce stable results. In this thesis, the stability margins were investigated deeper than in [14] and [15] since the purpose of this robustness analysis was to determine if the robustness had decreased or increased given a parameter change.

1.4 Approach

In this thesis a test procedure was developed for collecting proper measurement data which can be used for determining the stability margins of the air charge control loop of an ICE. This was done by first establishing a background theory and doing a thorough literature study where similar work and inspiration for new ideas was found. This thesis' aim was to make findings and conclusions with practical results and data collection from real engine tests. The tests were made with a Volvo engine provided by LiU using both Volvo's control system in INCA and LiU's control system in ControlDesk where the throttle servo control loop was of the essence.

The initial tests were performed early in the work process and were done with only the ignition on and in some cases only on a throttle which was hanging freely in the air. The tests revolved around getting to know the testing facility and how to use the hardware and software. This was done by simple tests containing steps on the throttle's opening angle. After doing the measurements they were stored into data files for analysis and processing in MATLAB. Before it was possible to use the data in MATLAB it had to be converted correctly into files that MATLAB could read. Processing was also necessary in some cases, for example to remove unwanted measurement noise.

Later in the work process tests with the engine running were performed. A lot of guidance and help were needed from the supervisors at PES since the control system of the engine is complex and would take years to learn. The majority of the hypothesis, possible methods and engine setups came from a collaboration with the supervisors at PES and this thesis' supervisor from LiU. Information was provided about which signals to record and which parameters to change to create different stationary work points that were investigated. This was done in order to set up tests and produce measurements that contained meaningful information.

The input signal for the real tests was disturbed with an oscillator which superimposed a sine wave on the static signal. The tests were made with different static frequencies which were able to capture the 0dB and $-180°$ point. Before the interesting static frequencies could be decided upon a frequency sweep had to be made to roughly find where the cross-over frequencies and boundaries for the robustness margins were for a specific stationary point. This was done to gain the most information from every test. Accordingly a set of predetermined frequencies were chosen and used to compare measurements of the engine behaviour after its parameters were changed.

With these measurements MATLAB was used to develop the method that could extract the phase and amplitude of both the input- and output signal. With this data a Bode plot could be drawn manually and the gain- and phase margin could be determined from the plot. The aim was to prove that the method to draw the Bode plot was correct and was showing a fair estimate of the margins from the experimentally measured system. Once the method was deemed accurate more tests could be performed to prove that the PID-parameters that were changed was in fact making the system less or more robust.

Finally, a documentation consisting of test procedures was made where the interesting stationary points and frequencies were highlighted. The system was excited with both induced disturbances such as sine waves and nonlinear signals such as time delays which provoked the system. The frequencies and amplitudes of the sine waves were documented for each test together with the respective time delay which was used to shift the point $-180°$ towards lower frequencies. This, to make the tests which were performed reproducible for the staff at PES but also for future master's thesis students.

1.5 Limitations

The system could not be deemed totally linear even if the stationary work point was far from the limp-home position where nonlinear spring effects existed. The nonlinear effects was seen when studying the opening and closing of the throttle where the throttle opened faster than it closed. These nonlinear effects were not accountable for since it was too difficult to determine their magnitude. There was only a small operating region considered when the robustness analysis was made. In this region the engine dynamics behaved almost linear. Linear analysis methods such as the relation between a sine as input- and output signal of a system was used for this. Nonlinear system identification was therefore not considered because it was too difficult to estimate a model of the system that would be precise in the whole operating region of the system during the scope of this thesis.

Another limitation was that the oscillator that was superimposed on the control signal could not be moved to another part of the loop because of limitations in the software. Therefore, ideas about moving the oscillator to improve the results could not be implemented.

The amplitude of the output was too small for any usable data to be analysed when collecting measurements for some tunings at high frequencies. This made it impossible to determine the stability margins since the sine wave method depended on knowing the difference in amplitude and phase between the input- and output signal.

2 System Description

The different systems and subsystems investigated in this thesis are presented in this section.

2.1 Hardware

The tests were performed on a real engine at the engine laboratory at LiU. The laboratory is equipped with two test engines with asynchronous machines that can drive and break the engines. The controller hardware that are used is dSPACE together with a control system from Simulink which is built in ControlDesk or a control system that Volvo uses called INCA. The measurement system is built around an HP VXi mainframe but can also be used with Linux or built in capabilities of dSPACE.

2.1.1 Test engine

The test rig is equipped with a four cylinder 2.0L (1.989cc) petrol engine with single turbocharger and dual variable cam phasing. The compression ratio is 10.3 : 1 and the inter cooler is using direct injection. The engine has an electric water pump and balance shafts. The bore length is 82mm and stroke length 93.2mm.

2.1.2 Test engine settings

The constant settings of the test engine had to be determined in order to perform tests that would generate comparable and reproducible data. The following settings were used for all tests.

- Engine speed of 1750 rpm.
- Fully open waste gate, no turbocharging was used.
- The throttle was 19% open.

The values of the engine speed and static throttle angle which the sine wave was superimposed upon was decided to keep the relation between the pressure ratio and compressible flow linear. This was done by having a throttle angle larger than 10% whilst keeping the relation between intake manifold pressure and the pressure before the throttle lower than 0.52. The theory which this was based on is displayed in Figure 1. The nonlinear effects from the sub-sonic velocity region was therefore not considered in this thesis.

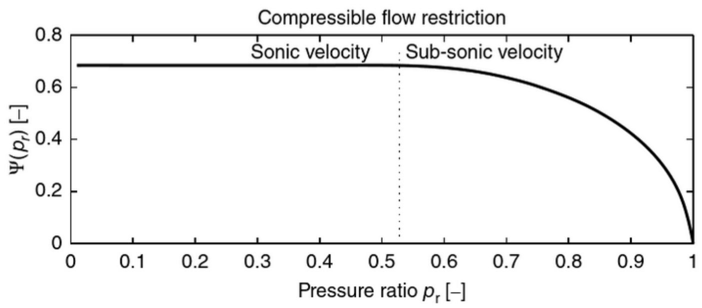


Figure 1: The compressible flow restriction graph from [16].

2.2 Software

The various software used in the engine laboratory at LiU are described in the following sections.

2.2.1 INCA

INCA is a software used by VOLVO where the whole control system can be modified before and during engine tests. Any control signals of interest can be measured in a preferred sampling rate and it is easy during tests to visualise the signals of interest in plot scopes and see the effects of parameter changes instantly. Signals which were of interest during the engine tests are presented and described in Table 1. Calibration variables which were modified between the different tests are presented and described in Table 2.

Table 1: Signals used in INCA.

Signal name	**Description**
sVcAesHw_X_ThrTar	Superimposed control signal to the throttle servo loop.
sVcAesCh_X_ThrPosnTar	The original control signal to the throttle servo loop.
Sae_TgtThr_Pst	Input signal to the throttle servo loop, equivalent to Hw.
Scm_EfThrAngl	The measured output signal from the throttle servo loop.
Svt_ActAng_A	The measured output signal from the VVT control loop.
Svt_TrgAng	The reference signal to the VVT control loop.
sVcAesCt_m_ThrCtrl	The control signal in [mg/stk] to the throttle servo loop.
sVcAesCt_m_ThrTar	The contribution on ThrCtrl from the PD-controller.
rVcAesCt_m_ImcCorrnCylSm	The contribution on ThrCtrl from the I-part.

Table 2: Calibration variables used in INCA.

Variable name	Description
cVcAesCt_B_ThrCtrlFbSel	Activation variable.
cVcAesCt_tc_ImcCorrnCyl	Time delay of I-part's contribution on the control signal.
cVcAesMo_t_CylPresPredAdj	Time delay of the system's reference model.
tVcAesCt_fac_ThrCtrlPCmp	The P-part of the air charge PD-controller.
tVcAesCt_Z_ThrCtrlDEst	The D-part of the air charge PD-controller.
tVcAesMo_rt_MafCorrnDynLvl	Activation variable.

2.2.2 ControlDesk

ControlDesk is a similar software as INCA, its the software used by employees at LiU and is used to conduct experiments and collect measurement data. The difference between INCA and ControlDesk is that INCA has less freedom in terms of changing the core of the software. That is because the control system used in ControlDesk is built in Simulink and can be modified endlessly.

2.2.3 MATLAB

MATLAB is a matrix based programming software used for analysing and designing control systems. It contains numerous toolboxes which can be used to solve a variety of problems. In this thesis it will function as the main software for processing and analysing the collected measurement data.

2.2.4 Simulink

Simulink is a MATLAB based graphical simulation software used for modelling and simulation of control systems. It can be used to evaluate transfer functions which have been estimated in SITB. It is also a intuitive tool for creating simple controllers and control loops which can be used for estimating ideas and new methods for processing data since Simulink produces measurement data that is free from measurement noise.

2.3 Air charge control

The throttle control loop consists of a PID-controller, F, and a plant that is separated into two systems. The error signal is fed through the controller with the PD-part and I-part in parallel. The signal from the separated parts are then added together as the output. The signal is then fed through G1 and G2 in series before fed back to the reference signal as seen in Figure 2.

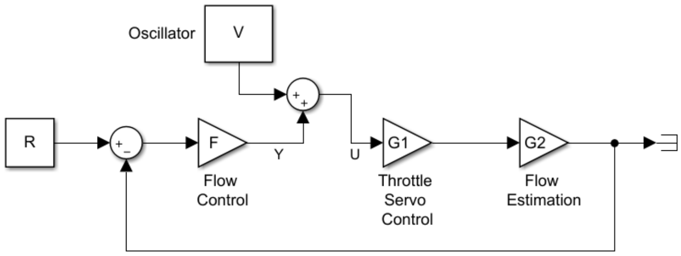


Figure 2: Simplified model of throttle angle demand and control loop. U is the input to the servo control and Y is the output from the controller.

G1 was a cascade control loop and a more elaborate model is seen in Figure 3. $U1$ corresponds to the control signal U and a feedback was introduced to create $G1$ with components $F1$ and $G11$.

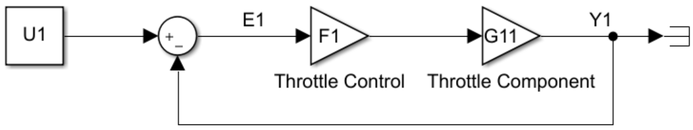


Figure 3: The cascade control loop of the throttle servo.

The control loop is described more in detail in Figure 4 where all components and correct signal names used in INCA are displayed. The Simulink model in Figure 4 was only used to display how the signals were connected to each other in INCA, it was not possible to use the model for simulations in Simulink.

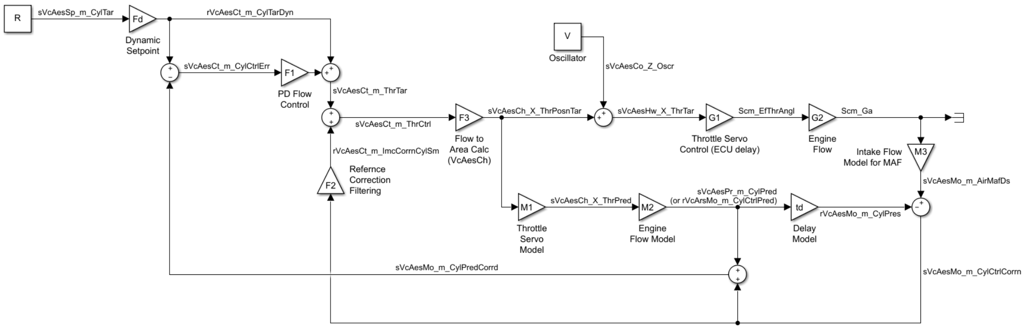


Figure 4: A model of the air charge control loop.

The control loop where the gain- and phase margin were investigated which was previously displayed in Figure 2 is displayed more in detail in Figure 5. The signal Hw (sVcAesHw_X_ThrTar) was used as input and Ch (sVcAesCh_X_ThrPosnTar) was used as output to capture the whole control loop.

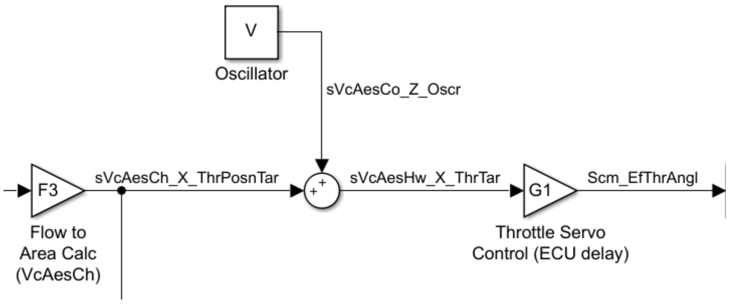


Figure 5: The input- and output signal of the investigated part of the air charge control loop.

There was not a parameter of the control loop from Hw to Ch, seen in Figure 5 which could be considered as the loop gain. Therefore, an alternative loop gain had to be identified. The air charge controller consisted of a PD-controller (PD Flow Control) connected in parallel with an I-part (Reference Correction Filtering), seen in Figure 6. The contribution from the I-part on the control signal could not be affected since it was not possible to change or turn off the I-part from INCA. Both the P- and D-part of the PD-controller could be changed and the decision was made to turn off the D-part. By only changing the P-part of the now considered P-controller, once the D-part was turned off, the contribution on the control signal could be seen as a gain that would mostly affect the amplitude curve of the Bode plot for the investigated loop.

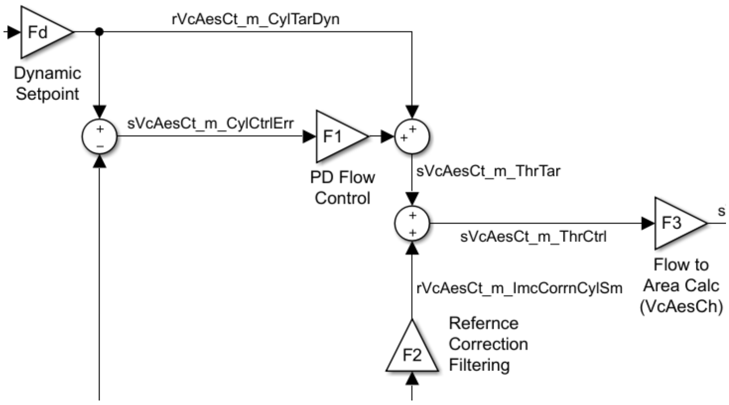


Figure 6: The summation of the control signal, sVcAesCt_m_ThrCtrl.

A change of the P-part, PCmp (tVcAesCt_fac_ThrCtrlPCmp), could have been identified proportionally on the signal ThrTar (sVcAesCt_m_ThrTar) in Figure 6. However, the signal was also influenced by CylTar (rVcAesCt_m_CylTarDyn) which was discovered late in the thesis' work process. This caused problems which are discussed later in this thesis.

2.4 Servo control

There is an electronic servo motor that controls the angle of the throttle. The electronic throttle control is called (ETC) which controls the air mass flow to the intake manifold of the engine. The current angle of the throttle is measured with a potentiometer. There are two voltage outputs measuring the throttle at the same time so sensor failure can be detected. The throttle is pre-loaded with a spring force in case the engine would lose its electricity. This prohibits the throttle from closing all the way so the engine can start even when the throttle is not working. This is called the limp-home position. The spring however exhibits both linear and nonlinear effects in the ETC system. There is also a nonlinear effect from friction force between the valve plate and the manifold. The spring constant K_s, limp home position θ_0, input motor torque T, preload torque T_p, and the angular displacement of the spring the throttle angle θ_{valve} becomes piece-wise linear as seen in Equation (2).

$$\theta_{valve} = \begin{cases} \theta_0 & if \quad |T| \leq T_p \\ \theta_0 + sgn(T) \cdot \frac{T - T_p}{K_s} & if \quad |T| > T_p \end{cases} \tag{2}$$

The linear ETC model can be described with the motor armature current, i, the motor output torque, T, the motor resistance, R, inductance, L, the torque constant, K_t and the back e.m.f., K_b. This is done with model presented in Equation (3) and the transfer function G(s) from the input voltage V_i to the angular valve displacement θ_{valve} in Equation (4) [3].

$$V_i = \frac{R}{K_t \cdot n}\Big[\frac{Jd^2\theta_{valve}}{dt^2} + \Big(D + \frac{n^2 K_b \cdot K_t}{R}\Big)\frac{d\theta_{valve}}{dt} + K\theta_{valve}\Big], \tag{3}$$

$$G(s) = \frac{\theta_{valve}(s)}{V_i(s)} = \frac{\frac{nK_t}{J \cdot R}}{s^2 + \Big(\frac{D}{J} + \frac{n^2 K_b K_t}{JR}\Big)s + \frac{K}{J}} \tag{4}$$

2.5 Air charge control

The air charge control is the major responsibility the throttle manages. It is an indirect result of the servo control where the throttle angle is controlled. For a requested engine torque from the gas pedal the throttle translates the torque into an air mass flow which corresponds to the requested engine torque. The air mass flow affects the amount of fuel which is getting injected into the cylinders since the air/fuel (A/F) mass flow ratio has to be close to 1 [16]. It is an import control loop since it has a lot of influence on the torque provided from the engine.

The throttle servo control is an inner loop which the air charge control loop contains. The two control loops works together with the common control strategy, cascade control. When using cascade control the inner loop (servo control) is supposed to have faster dynamics than the outer loop (flow control). A common design of cascade control is to have a 5 to 10 times faster inner loop than the outer loop for good performance [17].

2.6 Variable valve timing control

The variable valve timing (VVT) is an actuator which is a part of the air charge control loop and is influenced by the throttle servo control loop as well as it affects the throttle servo control loop. The timing of the valve opening and closing will affect the performance, fuel consumption and emissions of the engine. The VVT's openings and closings are relative to the rotation of the crankshaft which drives the camshaft. The VVT is beneficial since it opens up the possibility to influence the work production and exhaust temperature [16]. Without a VVT, the opening and closing of the combustion chamber would be the same for all engine speeds which would result in a less optimal engine. There are four different timing adjustments which can be used to optimise the performance of the ICE. The first is late intake valve closing which primarily is used to reduce pumping losses, the second is early intake valve closing which also reduces pumping losses during low engine speeds. The third is early intake valve opening which increases the volumetric efficiency since when it opens early exhaust gas flowing back into the intake manifold and there is less exhaust gas to be expelled by the exhaust stroke. The final adjustment is early or late exhaust valve closing which is used to reduce the amount of emissions.

3 Theory

The relevant theory used in this thesis is presented in this section.

3.1 Control theory terms

A definition of terms is presented in Table 3 and the following sections to better understand what is meant by an input/control signal, stability, robustness and other terms that are present in an ordinary control loop.

Table 3: Description of signals used in Figure 7.

Signal name	**Designation**	**Description**
Reference signal	r(t)	The reference signal is the desired value of the output signal.
Input signal	u(t)	The input signal is the signal that is sent to the transfer function of the system.
Output signal	y(t)	The output signal is the actual value which the system sends to the next loop or actuator.
Control error signal	e(t)	The error signal is the difference between the reference- and output signal, $e(t) = r(t) - y(t)$, which only exists if there is a feedback loop.

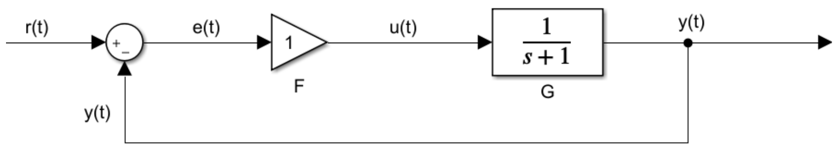


Figure 7: A simple feedback control loop where all names of the signals are displayed where F is the controller and G is an arbitrary transfer function.

3.1.1 Stability

Given a limited input signal the output signal also have to be limited for the system to be stable [18]. A stable system have to reach the steady-state position of the reference signal and remain in that state even after small changes of control parameters of the system. An unstable system is therefore often characterised by signals which converges to either positive or negative infinity.

Stability of linear systems

Common methods for determining stability for linear systems are the Nyquist criterion, Bode plot, pole-zero analysis and also numerical methods such as Euler-forward. The poles of a system can be found by studying the transfer function of the system where the poles are the roots to the polynomial in the denominator of the transfer function. When studying the poles of a system in continuous time all poles need to have negative real parts for the system to be stable. If the system is written in time-discrete form then all poles have to lie within the unit circle in the real/imaginary-plane to be deemed stable.

3.1.2 Robustness

Robustness is a well established measurement for evaluating a system's sensitivity against model errors, in other terms, how large the model errors can be without risking instability [19]. It is only useful if the analysed control system is stable since robustness analysis of unstable systems does not yield any satisfying results. It differs from stability where robustness is used as an indication of how stable the system is. It gives an indication of how much a certain design parameter can be tuned without risking instability in terms of gain- and phase margin. A variety of methods exists to compute the robustness of a given SISO- or MIMO-system, commonly used methods are the Nyquist criterion, Bode analysis, and evaluation of sensitivity and complementary sensitivity function. From Nyquist and Bode both the gain- and phase margin of a given system can be determined which are used to describe the robustness of the system. A robust system characterises by a large gain- and phase margin.

Gain margin

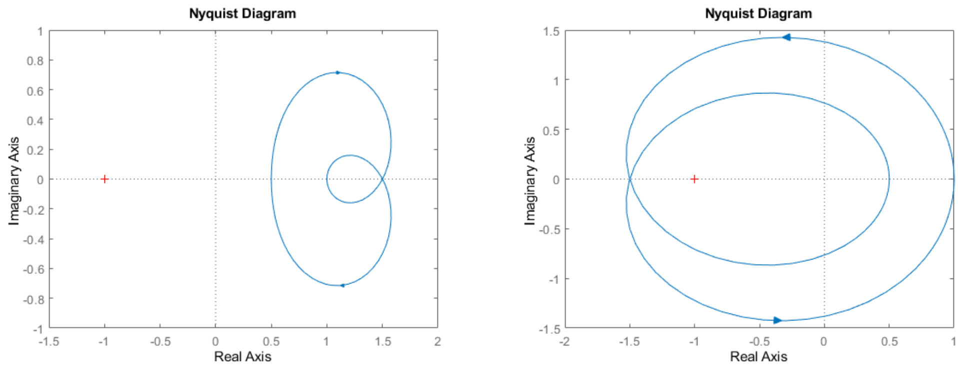
The gain margin for linear system is a measurement of how much the loop gain of a system can be increased until instability is reached. For a linear system the gain margin should be proportional to the loop gain and thus if the loop gain is increased with a factor 2 then the gain margin should decrease with a factor 2. The gain margin can for example be determined by Nyquist analysis where the loop gain is increased until instability occurs or by studying the Bode plot where the gain margin can be determined either visually or by tools in MATLAB. When the gain margin is equal to 0 it means that the system is on the verge of instability.

Phase margin

For a system to be considered stable it has to have both a positive gain- and phase margin. The phase margin can either be estimated from the Nyquist plot, a Bode plot visually or be determined with equations. The phase margin when studying the Nyquist plot is the angle between the negative real axis and the point where the Nyquist curve is crossing the unit circle. With a Bode plot, the phase margin can be determined from the frequency that yields the gain margin, $GM = 1$ (0 dB). A common design wish is to have a phase margin of at least 60 degrees for good robustness [18].

3.2 Nyquist criterion

The Nyquist criterion or Nyquist stability criterion is a method to determine if the closed-loop system is stable by studying the open-loop transfer function in the real/imaginary-plane. The criterion states that if the open-loop transfer function, G_o, has no unstable poles then the closed-loop system, G_c, is stable if the Nyquist curve does not encircles the point -1 on the real axis. If G_0 has x unstable poles then for G_c to be stable the Nyquist curve has to encircle the point -1 x times. The Nyquist curves for two different transfer functions $G_1(s)$ and $G_2(s)$ are displayed in Figure 8. In Figure 8a the Nyquist curve does not encircle the point -1, which is visualised in the figure as a red plus sign and thus, the system is considered stable. However, Figure 8b displays the Nyquist curve of another transfer function where the point -1 is encircled which indicates that the system is unstable.



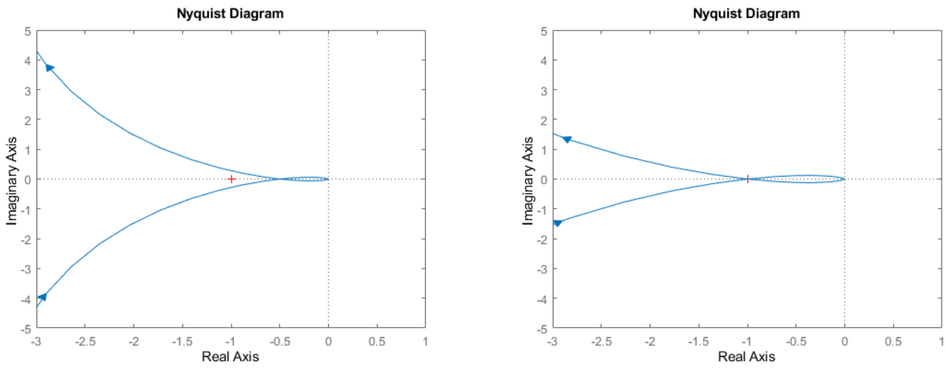
(a) $G_1(s) = \frac{s^2+3s+2}{s^2+2s+4}$.   (b) $G_2(s) = \frac{s^2+3s+2}{s^2-2s+4}$.

Figure 8: The Nyquist plot for two different open-loop transfer functions.

The Nyquist criterion can be used in both continuous- and discrete time where the criterion remains the same, thus both $G_o(i\omega)$ for $0 \leq \omega < \infty$ and $G_o(e^{i\theta})$ for $0 \leq \theta < 2\pi$ can be used. When studying the Nyquist plot in continuous- and discrete time the point where the Nyquist curve encircles -1 can be determined visually for simple models. Figure 9 demonstrates how a Nyquist plot can be used to visually see the effects of the loop gain, K, on the system's stability properties. The transfer function that is used is

$$G(s) = \frac{K}{2s^3 + 4s^2 + 1s}$$

and it is seen in Figure 9a that when $K = 1$, the Nyquist curve does not encircle the point -1 but when $K = 2$ in Figure 9b, the point -1 is precisely encircled which means that the system is stable for $0 < K < 2$.



(a) With loop gain $K = 1$.

(b) With loop gain $K = 2$.

Figure 9: The Nyquist plot for two different transfer functions where one precisely encircles the point -1 on the real axis.

For more complex models it can be difficult to determine stability by studying the Nyquist curve visually. Therefore, equations must be used to determine which loop gain, K, that will lead to either stability or instability for the investigated system. Moreover, the Nyquist curve can also be used to determine stability margins in terms of gain- and phase margin. If the analysed system is stable and a gain margin exists then the loop gain can be increased until the curve encircles the point -1 on the real axis. That amount is then the gain margin of the system. The phase margin can also be determined by studying either the Nyquist curve visually or by using Equation (5) in discrete time.

$$\varphi = \pi + \arg G_0(e^{i\omega^*}) \tag{5}$$

The gain cross-over frequency, ω^*, is determined by Equation (6).

$$|G_0(e^{i\omega^*})| = 1 \text{ (0 dB)} \tag{6}$$

3.3 Bode

The Bode plot is often used when performing frequency analysis where the magnitude curve is plotted against frequency and also the phase versus frequency are being examined. Both the sensitivity- and complementary sensitivity function can be plotted as a Bode plot to determine the gain- and phase margin when studying the robustness of a system. In Figure 10 a Bode plot is displayed for the transfer function given by (3.3).

$$G(s) = \frac{1}{2s^3 + 4s^2 + s}.$$

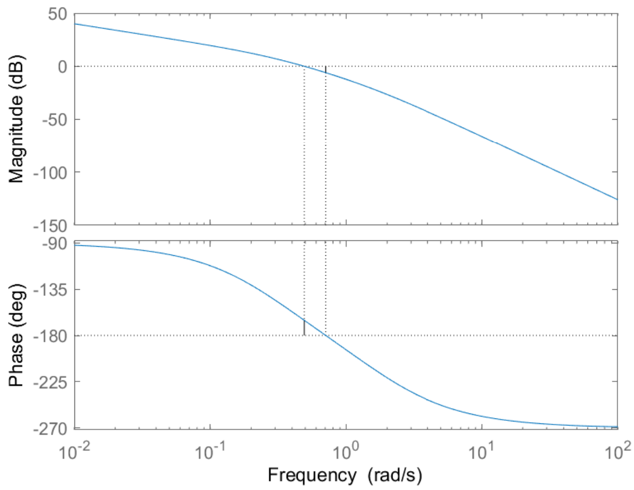


Figure 10: A Bode plot where the gain- and phase margin are displayed.

In Figure 10 both the gain- and phase margin are displayed at the top of the figure. To determine the gain margin visually the frequency where the phase crosses -180 degrees has to be determined and find which gain, G, that frequency corresponds to in decibels in the magnitude plot. Then to calculate the gain margin (GM) for the system take 0dB and subtract G, $GM = 0 - G$. To determine the phase margin visually, first determine the frequency where the magnitude, $|G(i\omega)| = 0$, then find the degree, P, which corresponds to that frequency in the frequency vs. phase curve. The phase margin (PM) can then be calculated as $PM = P + 180$.

3.4 System identification

To be able to perform the robustness analysis using the Nyquist curve and Bode plot the SITB graphical user interface (GUI) in MATLAB is used for estimating and validating transfer functions from measurement data. SITB in MATLAB is a commonly used tool for estimating models from measurement data and is used in [3]. In the following sections the procedure of how the system identification is executed in SITB are explained.

3.4.1 Processing of measurement data

The processing of the measurement data can be performed either when the input- and output signal is distinguished in MATLAB or an iddata object has been created. The measurement data has to be processed before any models can be estimated. This is easily done using a processing tool in SITB called "Quick start" which removes the means of the measurement data and divides it into an estimation data set and a validation data set. This is important to properly validate the models that are created.

3.4.2 Linear models

From measurement data it is possible to estimate both linear- and nonlinear models of the examined system. In this thesis the emphasis is put on linear models such as ARX, ARMAX and Box-Jenkins (BJ) models. The arbitrary discrete-time ARMAX model is given by

$$A(z)y(t) = B(z)u(t) + C(z)e(t).$$

The parameters which defines the ARMAX model is given by

$$[\text{na} \;\; \text{nb} \;\; \text{nc} \;\; \text{nk}],$$

where each parameter either describes the number of poles or zeros for the input to output- and disturbance to output transfer functions. Where na is the order of poles, nb is the order of zeros, nc is the order of error and nk is the input-output delay.

3.4.3 Frequency analysis

A spectral model has to be decided on which the estimated models should be compared to before any models are estimated. A good spectral model is helpful for estimating the number of poles in the true system since each resonance peak in the frequency plot corresponds to a pair of poles of the system. There are three different types of spectral models the Blackman-Tukey's (SPA), frequency dependent resolution (SPAFDR) and smoothed Fourier transform (EFTE). The number of frequencies and the frequency resolution when estimating these models can be changed to suit the examined system depending on if the analysed system has fast or slow dynamics.

3.4.4 Residual analysis

The residual analysis is important when comparing two estimated models against each other. Both the auto-correlation of residuals for the output and the cross-correlation residuals between the input and output can be plotted. These curves should be within specific margins which usually lies between -1 and 1. When the curves exceeds these limits it means that some information is lost when the models are estimated, they can not describe all dynamics of the system. If a linear model is used on measurement data collected from a known nonlinear system then there should be considerable spikes when studying the residual plots since the linear models are unable to describe the nonlinear behaviour.

3.4.5 Model output

The model output is used to compare the estimated model to the validation data set from the measurement data in percentage of how similar the estimated model is compared to the collected data. Depending on if the measurement data has been collected from a linear or nonlinear system different model output fits are acceptable.

3.4.6 Poles and zeros

The poles and zeros for every estimated model can be displayed in a plot in discrete time to see if the model has an unnecessary high order. This can be seen if a model have a lot of poles close to each other or in some cases even overlapping each other. That is an indication that several poles describe the same system dynamics and they can therefore be removed since as low order as possible for the system is desirable to minimise the risk of computational difficulties. It can be difficult to spot the poles that are overlapping or even the ones that are close to each other. The confidence intervals for the poles can then be plotted to be able to find out if two poles are too close to each other and should always be done to be able to determine if the model order is acceptable or if it can be reduced.

3.5 Control system

The engine's parts are controlled by numerous controllers which purpose are to keep a high performance and robustness even when there exists disturbances. Some controller types and control strategies which the engine consists of are described in this section.

3.5.1 PID-controller

The Proportional-Integral-Derivative controller is the most commonly used controller in industry [17] and there are a lot of PID-controllers in the modern engine's control system. There are three main parts of a PID-controller, the P-part corresponds to the proportional gain which influence on the control signal, $u(t)$, is proportional to the static control error, $e(t) = r(t) - y(t)$. The I-part's purpose is to eliminate the static control error, this is done by integrating the error. The influence of the I-part is therefore proportional to the integral of $e(t)$. The D-part is used to eliminate overshoots and oscillations, the influence from the D-part on the control signal is proportional to the derivative of $e(t)$. When using a PID-controller in a feedback system the control signal, $u(t)$, is given by

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}.$$

A simple feedback control system scheme with a PID-controller is displayed in Figure 11 to illustrate how it can be implemented in a Simulink model, where the PID-controller is inside the blue line and the controlled process is within the red line.

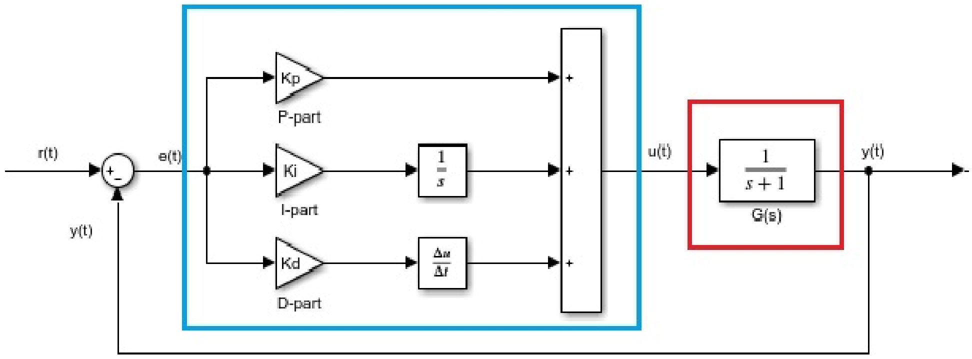


Figure 11: A PID-controller scheme with feedback control.

The purpose of increasing the P-part is to give the system a faster step response, the drawback of increasing the P-part too extensively are overshoots and in some cases residual oscillations [18]. These overshoots can be taken care of by the D-part, so with a PD-controller the step response of a system can be increased without adding any overshoots. The drawback of using the D-part is that if the D-part becomes too large it will damp the system too hard which will result in oscillations which might be large enough for the system to become unstable. The I-part's sole purpose is to eliminate the static control error for good reference following. It is done by moving the poles of the system, this could sometimes cause instability since if the I-part is too large then the poles can move into the right hand side of the imaginary axis [18].

3.5.2 Cascade control

Cascade control is most commonly used in processes where there are more than one measurable output signal which can be used when controlling the system. The control strategy is characterised by an inner- and outer loop where the process has been decomposed into two sub-processes $G_1(s)$ and $G_2(s)$ that use several measurement signals, z and z_r, to calculate the control signal, u [17]. A simple cascade control loop is illustrated in Figure 12 where the blue box contains the process and the red box contains the inner loop. This strategy is beneficial because any system disturbances that would have influenced the sub-process $G_2(s)$ is taken care of by the controller, $F_2(s)$, inside the inner loop. This will lead to that the effect the disturbance would have had on the output signal, y, is decreased. A condition for a cascade control loop to function properly is to have much faster dynamics of the inner loop, most often around 5 to 10 times faster than the outer loop [17].

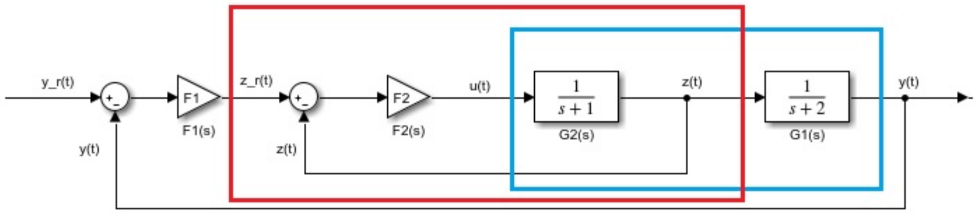


Figure 12: A simple cascade control loop from [17].

An engine consists of a lot of cascade control loops where one has already been mentioned, the throttle servo control loop. When controlling the throttle the servo control operates as the inner/secondary loop and the air flow control is seen as the outer/primary loop. When tuning or optimising a cascade loop it is beneficial to start with the inner loop and choose an appropriate $F_2(s)$. This is recognised in this thesis as the practical work will begin with the servo control loop of the throttle when the engine is at idle. When the servo loop has been examined then the air flow loop will be investigated. Even though the purpose of this thesis is not to tune controllers inside a cascade loop it is of the essence to understand how the control loops within the engine are affecting each other. For the simple design of a cascade control loop in Figure 12 proposed in [17] the transfer function for the secondary loop can be written as

$$G_{c,2}(s) = \frac{G_2(s)F_2(s)}{1 + G_2(s)F_2(s)}.$$

The transfer function for the primary loop, $G_{c,1}$, can be approximated as

$$G_{c,1}(s) = \frac{G_1(s)F_1(s)}{1 + G_1(s)F_1(s)}.$$

If the secondary loop is a lot faster than the primary loop then the inner loop can be considered as an arbitrary gain. The transfer function for the closed-loop system, from y_r to y, can be written as

$$G_c(s) = \frac{G_1(s)G_{c,2}(s)F_1(s)}{1 + G_1(s)G_{c,2}(s)F_1(s)}.$$

3.6 Filters

When analysing and performing calculations on measured signals it is sometimes hard to draw conclusions from it because of noise from the measurement equipment or the measured system itself. Filters are used to reduce the influence of measurement noise to achieve more accurate results. The filter used in this thesis is described in the following section.

3.6.1 Zero-phase filter

The main purpose of the zero-phase digital filter, for example MATLAB function filtfilt, is to take a normalised signal and reduce its noise without inducing any phase distortion. The filter reduces noise by minimising the start-up and ending transients by matching their initial conditions. In order to achieve zero-phase distortion the filter processes the input data x in the forward and the reverse direction. Additionally, it then reverses the filtered sequence and runs it back into the filter again. This results in the filter having no zero-phase distortion and a filter transfer function equal to the squared magnitude of the original filter transfer function.

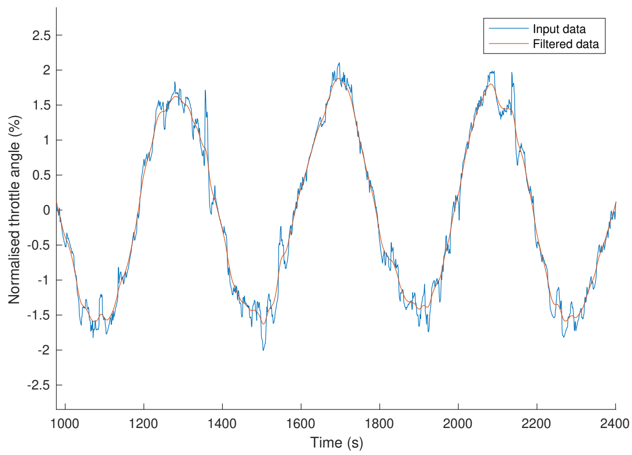


Figure 13: The filter filtfilt used on the input signal to filter out noise for the zero-crossing method.

4 Test Descriptions

The majority of the work through out of this thesis has been conducted at the engine laboratory at LiU where numerous measurements have been collected both with INCA and ControlDesk. These tests was the foundation which the results of this thesis relied upon. The tests are described in chronological order in this section.

4.1 Throttle servo identification in ControlDesk

The initial tests in ControlDesk for the system identification of the throttle control loop were made on the external throttle that hung freely in the air. First, steps with different amplitudes were executed to be able to calculate the time constant, T, and time delay, L, for the throttle's dynamics. Three steps were made where T and L were estimated and used to derive a simple three parameter model given by Equation (7).

$$G(s) = \frac{K}{sT+1}e^{-sL} \tag{7}$$

The first step was from 10% to 30% open throttle, the second was from 10% to 50% and the final step was from 10% to 90% throttle angle. For all the three measurements were $L = 0.01s$, for the first two smaller steps were $T = 0.03s$ and for the large step was $T = 0.034s$. That the time constant differed for the three measurements was not unreasonable since there was a big difference of amplitude between the steps. These initial tests gave a lot of knowledge about how to use ControlDesk and how to change the control system to create desirable input signals. The results from the three parameter model were not satisfactory since the models were not accurate enough and could therefore not be used in future work of this thesis. The purpose of this test was to learn how the software functioned and how to process the measurement data in MATLAB efficiently which was deemed fulfilled.

4.1.1 PRBS-identification

A PRBS-signal was created in Simulink with a random number generator and a switch which determined the lower and upper limit of the signal. The sample time which determined how often the random number generator would generate a number between 0 and 1 was set to 0.01s. This PRBS-signal was then used as the control signal to the throttle servo control loop where the actual throttle angle could be measured in ControlDesk. Several tests were executed with different amplitudes of the PRBS-signal. A measurement was collected when the PRBS-signal switched between 30% and 50% open throttle and is analysed in Section 5.1. Further on in the project the decision was made together with Volvo that ControlDesk would not suffice for future work of this thesis. ControlDesk's code was built differently and had different features compared to Volvo's code and thus, the results from the tests performed in ControlDesk were not of interest. INCA is built on the same code as Volvo uses and was deemed sufficient for this project and was used for the remaining part of this thesis.

4.2 Throttle servo identification in INCA

The throttle servo had been examined with ControlDesk where the measurement data had been processed by SITB. SITB was used to estimate and validate transfer functions for stationary work points of the throttle both with large and small amplitudes of the PRBS-signal. For the initial tests and measurements when running INCA instead of ControlDesk an oscillator was created that was able to overlay either a PRBS-signal, sine wave or a square wave on the original control signal which is seen in Figure 2. The signal Sae_TgtThr_Pst was chosen as input and Scm_EfThrAngl was chosen as output for these initial tests. Both the amplitude and frequency of the sine wave could be adjusted. By increasing the frequency of the oscillator the output experienced increased phase shift and loss of amplitude and from that a Bode plot could be made instead of a transfer function. For a linear system the sine waves were a convenient input signal for determining system properties. The throttle was set in a stationary work point where the dynamics of the throttle could be deemed linear, which was when the throttle's opening was larger than $9-10\%$ since that was where the limp home position was located for this throttle. Then, for a linear system the following relationship between input and output holds

$$\begin{cases} u(t) = A\sin(\omega t) \\ y(t) = |G(i\omega)|A\sin(\omega t + \varphi) \end{cases}$$

and therefore it was easy to determine the amplitude and phase curve of the Bode plot from the measurement data. The purpose of using the PRBS-signal was to produce stability margins which could be compared to the stability margins produced from the sine waves. The PRBS-signal and the system identification made from measurements collected in INCA is discussed further in Section 5.2.

4.2.1 Selection of input

The superimposed sine waves on the static control signal could either be ramped or static, both are described in this section.

Ramped sine waves

The identification of stability margins for the throttle was made with two different approaches. The first approach was to ramp the sine wave from a low frequency to a high frequency by continuously decreasing the period time of the sine wave. The reason for this was to capture the 0dB crossing where the input- and output signal had the same amplitude and the $-180°$ crossing point where the output signal was shifted 180 degrees in terms of the input signal. The initial tests were made with only one ramp which meant that the frequency was ramped with the same increment from lowest to highest frequency. This was later deemed insufficient which was discovered when trying to analyse the measurement data. The period time shifted too fast for the low frequencies and the phase lag could therefore not be identified properly. This was only an issue for the low frequencies and not for the high frequencies where the periods were a lot faster. The ramp time for the initial tests were 120s and the frequencies were ramped from a period time of 10s to 0.01s. A solution to this issue was to ramp between several frequencies. This allowed the ramp to have different time between low and high frequencies. This was a better take on this approach since the periods were captured before the period time shifted. The original simplicity of this approach disappeared when only one ramp was deemed insufficient. The several ramp approach did not yield any satisfying results and because of that together with the loss of simplicity a new approach had to be investigated. The decision to use static frequencies for the analysis was therefore taken together with assistance from the supervisor at PES.

Static sine waves

When the different approaches of using the ramped sine wave had been fully explored, the static sine waves were investigated instead. Studying static frequencies were not interesting at first due to the manual labour involved. That had to be compromised to be able to achieve measurement data that could be used for the robustness analysis. The advantage of using static frequencies was that the mean of the phase lag could be determined over several periods with the same period time. The mean value was desirable since the measurement data contained a lot of noise which gave rise to a significant variance. The frequencies which were decided upon between each measurement was identified by using the ramped signal. It was easy to determine the region of interesting frequencies using the ramped signal to find where the input- and output signal almost had the same amplitude and where the output signal's phase had been shifted $-180°$. With those two regions could several frequencies be decided upon which captured those two properties.

4.2.2 Static waves 1

When the use of static sine waves had been decided upon and investigated how to be used efficiently, a test was designed. This test's purpose was to collect measurement data where the stability margins could be recreated with similar measurements using the developed method for determining the gain- and phase margin of the throttle servo control loop. The parameter values of the engine setup is presented in Table 4.

Table 4: The engine setup which were the same for all measurements collected in this test.

Parameter	Value
cVcAesCt_B_ThrCtrlFbSel	1
cVcAesCt_tc_ImcCorrnCyl	0.05
cVcAesMo_t_CylPresPredAdj	0.03
tVcAesCt_Z_ThrCtrlDEst	0
tVcAesMo_rt_MafCorrnDynLvl	1
Engine speed	1750 rpm
Throttle angle	19%

The period times that were used for this test were 5s, 0.5s, 0.4s, 0.3s, 0.25s, 0.23s and 0.2s. The controller settings for the measurements differed between the measurements where the D-part was set to 0 for all of them but the P-part was set to 0.2 for the first measurement, 1.0 for the second and 1.2 for the final measurement. Another interesting tuning which had to be done for this test and the following tests was a time delay that was induced on I-part's contribution on the control signal and another time delay on the system's reference model. The time delays acted as a disturbances that made the system less robust. The reason for this change was because the point $-180°$ was previously reached at a high frequency. The induced time delay moved that point towards a region of lower frequencies where the method for analysing the data functioned. For the purpose of this particular test the settings where the P-part was equal to 1 were collected twice to compare the gain- and phase margin to be able to determine its reproducibility. From intuition the gain margin was expected to be larger for the test with $P = 0.2$ than for $P = 1.2$ compared to the reference with $P = 1$.

4.2.3 Static waves 2

The final set of measurements was collected with the same parameter settings as the previously described test displayed in Table 4. This test was designed to validate that the proposed method for producing the stability margins functioned properly. Nine different sets of measurement data were collected where the P-part of the air charge controller was the only part that differed between the measurements. Each measurement was collected with eight static sine waves which were decided based on a frequency sweep created by the ramped signal. It was important to capture the 0dB- and $-180°$ point when choosing the period time of each sine wave. The period times used were 5s, 4s, 0.5s, 0.4s, 0.3s, 0.25s, 0.23s and 0.2s. The relation between loop gain and gain margin of the system was not interesting with this test. The P-part could not be seen as proportional to the gain margin since it was connected in parallel with the I-part. The purpose of this test was therefore to see if the developed method was able to distinguish the change in gain margin correctly between all nine measurements. The P-parts which were used in this test were 0.2, 0.4, 0.6, 0.8, 1, 1.2, 1.4, 2.0 and 2.1. The expected result was that the amplitude curve of the Bode plot should have increased for each measurement. The P-value which would cause instability was also explored experimentally so that the calculated gain margins had something to be validated against.

5 Throttle servo identification

The transfer function of the throttle servo control loop was unknown and it was therefore deemed interesting to see if an approximate transfer function could be estimated with system identification. The purpose of creating a transfer function of the throttle servo loop was to use the Nyquist curve to determine the gain- and phase margin of the same loop which was investigated with sine waves in INCA. The whole air flow control loop containing the throttle was identified and investigated with sine waves in INCA where sVcAesHw_X_ThrTar was used as input and sVcAesCh_X_ThrPosnTar as output which has been mentioned in previous sections. For ControlDesk the PRBS-signal was used as input and the measured throttle angle was used as output. For the system identification a PRBS-signal was used instead of the sine wave as the superimposed disturbance on the static control signal. The PRBS-signal was used both early in the work process in ControlDesk but also later in INCA. The advantage of using ControlDesk for this particular part of the thesis was that the PRBS-signal which was the reference signal could be kept constant. The PRBS-signal was superimposed on the control signal in INCA which caused complications once the output signal was fed back and the signals were therefore difficult to process.

5.1 ControlDesk

A PRBS-signal was used as input with a frequency of 125rad/s for a step range of 30% to 50% open throttle. The throttle's opening angle was measured and stored in MATLAB for the throttle which was hanging freely in the air. This test was therefore not collected with the engine running. The input- and output signal which were collected and used for the system identification are displayed in Figure 14.

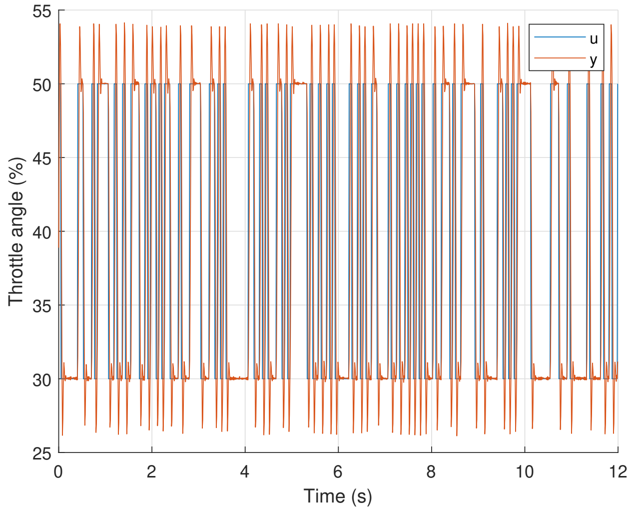


Figure 14: The input- and output signal which was used for the system identification.

The PRBS-signal together with the measured throttle angle were sent to SITB where they were analysed. Before any models were created the input- and output signal had to be processed which is displayed in Figure 15. The processing of the measurement data was done with quick start in SITB to remove the means and to separate the data set into an estimation and a validation data set.

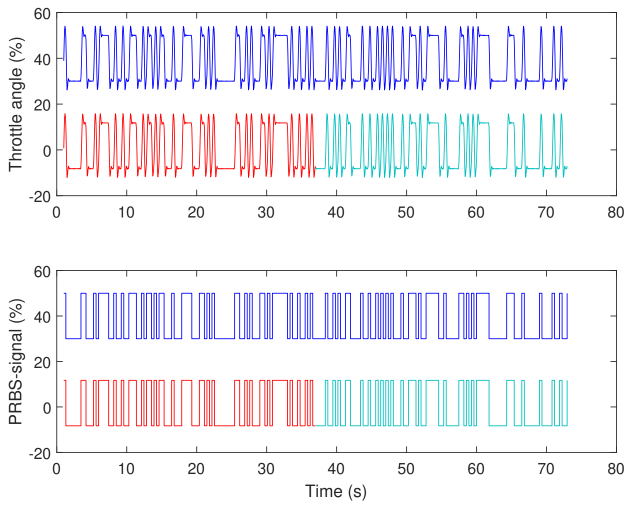


Figure 15: The processed input- and output signal in SITB.

A spectral model was decided upon and displayed in Figure 16. A tiny peak could be identified from the spectral model which indicated that the investigated system should contain at least two poles.

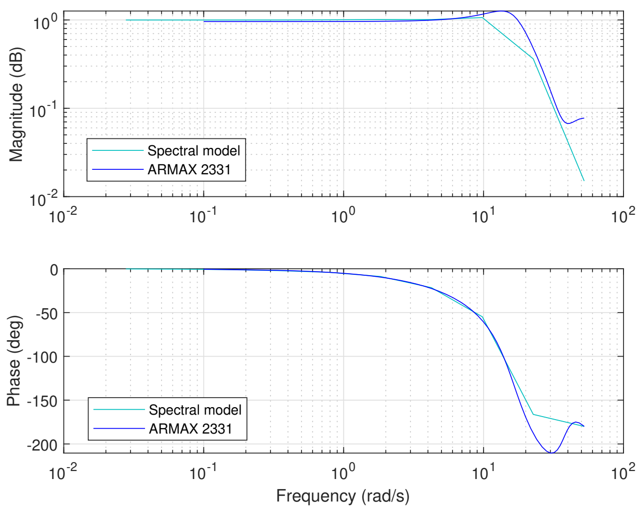


Figure 16: The frequency response for the estimated Bode plot made with Blackman-Tukey, SPA, in SITB.

Several ARMAX models were examined given the information from the spectral analysis which indicated that a model of order 2 should be sufficient enough. An ARMAX model that was deemed sufficient with parameters, [2 3 3 1], was chosen seen in Figure 16 together with the spectral model. The model was decided upon by comparing different constellations of the parameters against each other and studying both the model residuals and the model output. The residual analysis for the proposed model is seen in Figure 17. It was apparent that the system could not be deemed linear when the PRBS-signal was ranging from 30% to 50%. The cross-correlation plot in Figure 17 had a considerable spike which indicated that the proposed model was unable to describe all of the system's dynamics. However, that was a reoccurring property of all examined models with reasonable model order, an ARMAX of model order 6 was able to describe the nonlinear dynamics but was deemed unreasonable due to the spectral analysis.

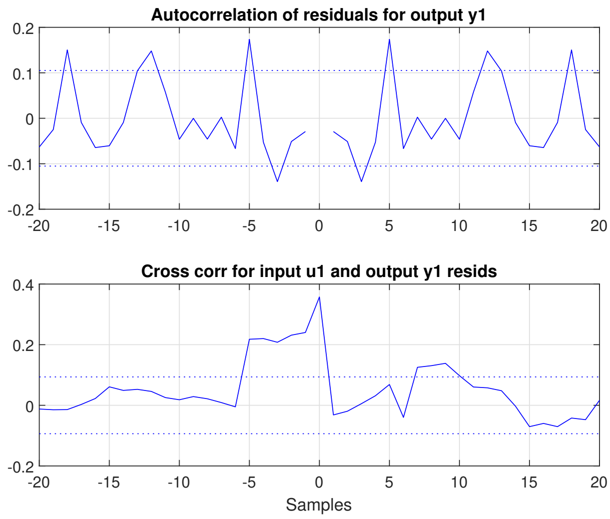


Figure 17: Residual analysis with max and min values for auto correlation in the top and cross correlation in the bottom.

The model output fit of the estimated model compared to the validation data was 88.58%, seen in Figure 18. Given the spike in the cross-correlation where the residuals between input and output exceeded its boundary a model fit of 88.58% was considered good.

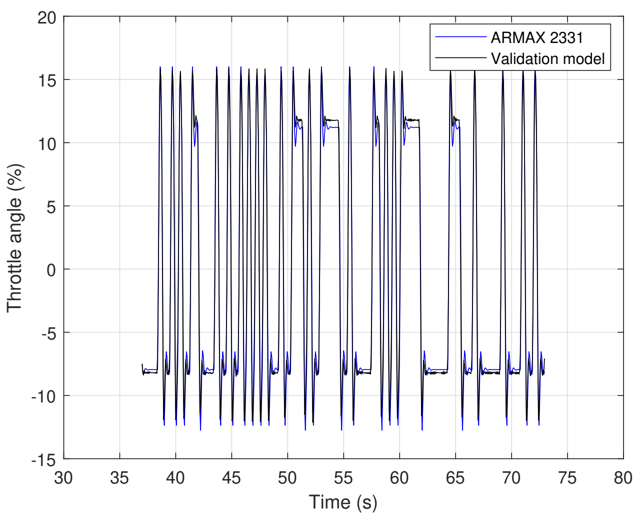


Figure 18: The ARMAX model compared to the validation data set.

The estimated ARMAX model's transfer function could be translated in SITB and analysed in MATLAB. From SITB the transfer function was described as

$$G(s) = \frac{0.1815s^2 + 10.81s + 694.6}{s^2 - 66.67s + 1111},$$

where the confidence interval for each parameter was printed in MATLAB's command window. All estimated parameters was within their respective confidence interval which meant that there was small parameter uncertainty. The transfer function was estimated for the open-loop system and the Nyquist curve could therefore be applied on the transfer function to determine the stability margins. The Nyquist curve of the estimated ARMAX model is displayed in Figure 19 where it was seen that the point -1 on the x-axis was not encircled.

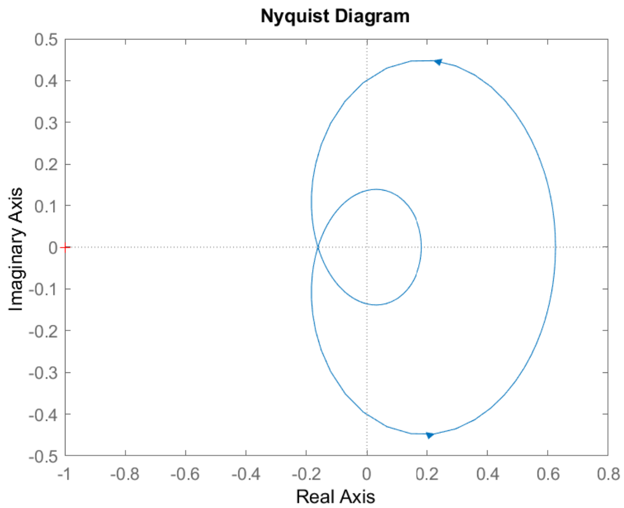


Figure 19: The Nyquist curve of the estimated ARMAX model.

5.2 INCA

The system identification performed from measurements collected in INCA would have been ideal to use for a comparison with the results produced using the static sine waves. The same engine setup was used when collecting these measurement data as for the test, static sine waves 2, previously displayed in Table 4. The PRBS-signal used in INCA was not equal to the one used in ControlDesk. In ControlDesk the signal was switching between two identical values for every switch. The PRBS-signal used in INCA functioned in a different manner. The signal switched amplitude for every sample time with different size of each switch instead of having the possibility to switch for each sample time which is seen when comparing Figure 20 with Figure 14. That caused the signal in INCA to never stagnate at any given amplitude. The feedback loop was another dominating factor which caused the signal to converge to the static control signal which the PRBS-signal was superimposed upon.

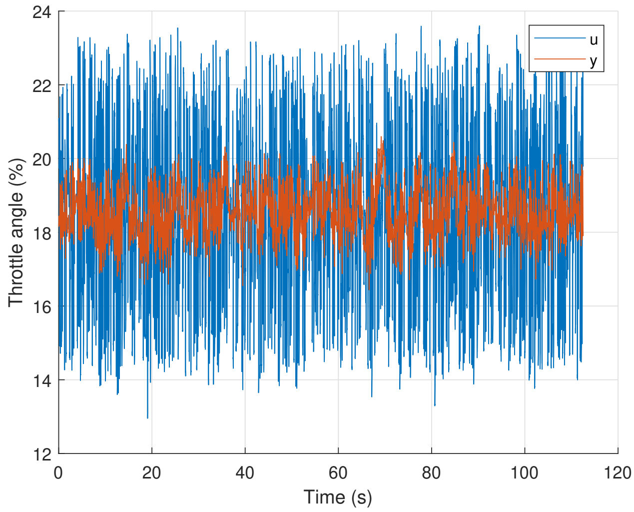


Figure 20: The input- vs. output signal when the PRBS-signal was superimposed on the control signal.

A spectral analysis was made and is displayed in Figure 21. It was apparent that it would be difficult to estimate a good model given the spectral model with the signals collected from the PRBS-test. The signal sVcAesHw_X_ThrTar was used as input and sVcAesCh_X_ThrPosnTar as output in order to estimate the same loop which was estimated with the static sine waves. The output should follow the input fairly when performing system identification to achieve good models. It was clear from the start when studying Figure 20 that it would be difficult to estimate a good model for this part of the throttle loop when using a PRBS-signal.

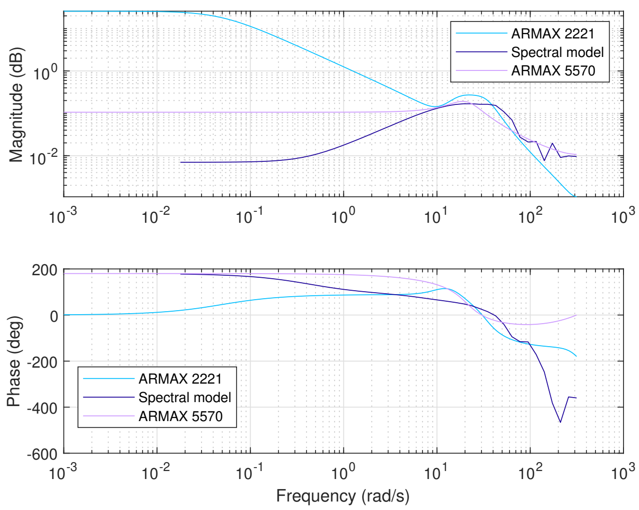


Figure 21: The spectral analysis for two ARMAX models.

Two ARMAX models were estimated with different model orders. The first AR-

MAX model had order 2 and could almost follow the spectral model from Figure 21 for low frequencies but differed a lot for higher frequencies. It became apparent when studying the model residuals in Figure 22 that a model of order 2 was unable to describe the system's dynamics. Both the auto-correlation and cross-correlation curve exceeded its boundaries which was an indication of that information about the system was lost when the model was estimated. The second model that was estimated had order 5, this model differed a lot from the spectral model, seen in Figure 21. The model residuals represented by the turquoise curve in Figure 22 were within its boundaries which meant that the model was able to describe the true system's dynamics.

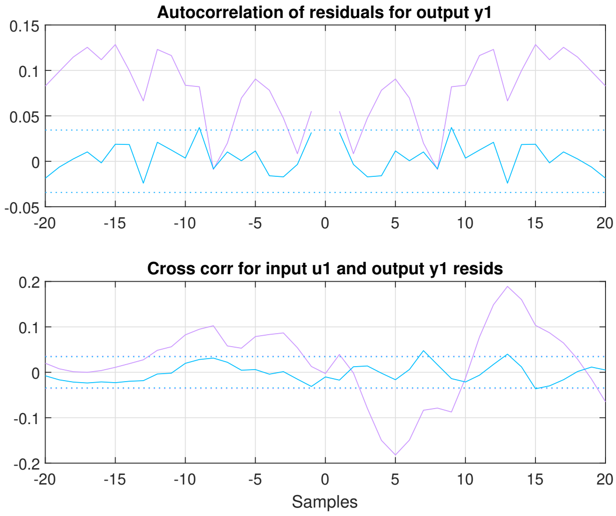


Figure 22: The residuals for the two ARMAX models.

The model output fit for ARMAX 5570 was 45.36% and 15.12% for ARMAX 2221. The estimated models' outputs together with the validation data set are displayed in Figure 23. Not much could be said about the ARMAX model of order 2 given the poor agreement of 15.12%. The ARMAX model of order 5 however, had a fairly good model output fit given that the signals used in SITB were not ideal. A model output fit of 45.36% for a system which had to be considered nonlinear given the poor agreement for the model of order 2 was deemed good. However, none of the estimated models could be deemed sufficient enough to produce credible stability margins due to the spectral analysis.

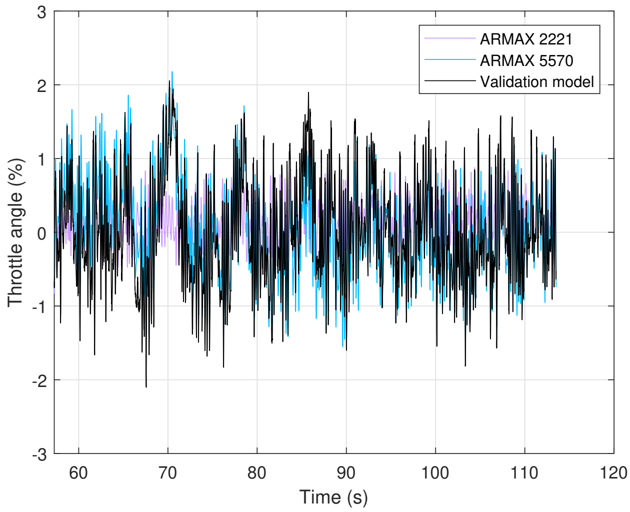


Figure 23: The two ARMAX models compared to the validation data.

6 Methods to determine stability margins

Several methods for determining the stability margins using measurement data have been investigated in this thesis. It was a difficult work process since both the gain- and phase margin depends on both the amplitude- and phase curve of the Bode plot. The methods were tested together and apart to see which ones that gave the most reliable results from several similar measurement data sets.

The purpose of all the investigated methods was to determine the relation in amplitude and phase shift between two sine wave signals in the most reliable way. The relation in amplitude was either determined by comparing the peaks of the input- and output signal or by comparing the summed normalised absolute values of the input- and output signal. The time delay was determined by subtracting the time for a specific position of the output with the corresponding position of the input. The positions was determined either by looking at the peaks of the sine wave or the zero-crossings. The time delay was then converted into a phase lag by multiplying it with the frequency.

In this section methods to determine the gain- and phase margin without knowledge of the transfer function is presented together with a validation of the chosen method.

6.1 Processing of data

A method for processing the data had to be developed before any methods for the stability margins were investigated. When a test was completed the data was collected by saving it as a .dat-file in INCA. The data had to be processed before any results or conclusions could be drawn from it. First the file format had to be changed in order for MATLAB to read the data. Then the arrangement of the data had to be altered from being placed in cells to matrices for the sake of making it easy to access and to write functions around. The function data_INCA was written to efficiently do all of this, seen in Appendix A. Since the signals were analysed as sine waves it was necessary to normalise the signals. This was done by subtracting the mean value of the data points within the same frequency from the original values within that frequency. The equations used to normalise the input- and output signal was the following.

First the mean values were calculated as

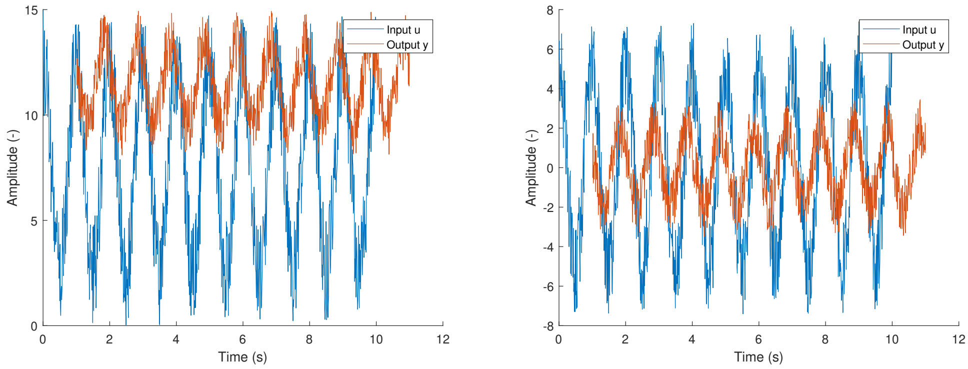
$$\bar{u} = \frac{\sum_{n=1}^{N} u}{N}, \quad \bar{y} = \frac{\sum_{n=1}^{N} y}{N},$$

then they they were subtracted from the signal as

$$u_{norm} = u - \bar{u},$$

$$y_{norm} = y - \bar{y}.$$

Two test signals, before and after the processing, is displayed in Figure 24.



(a) Before the normalisation. (b) After the normalisation.

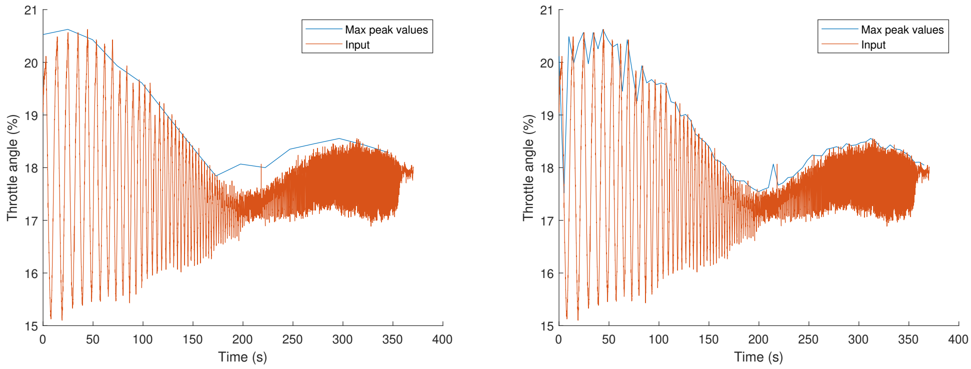
Figure 24: The input- and output of test signal before and after they were normalised.

6.2 Amplitude curve

The following methods were developed to calculate the amplitude curve of the Bode plot.

6.2.1 Mean peak value method

The fundamental idea with the mean peak value method was to take the highest peak within a set of periods from any given signal. The relation in amplitude could then be calculated by simply dividing the output with the input for the same set of periods. This method separated the measurement data into smaller sets of data where the peak in each data set was identified and stored into a vector. The size of each set could be changed to either include more or fewer values for each data set. This increased or decreased the resolution of the method since it affected the amount of amplitude values within the same time frame. This method was able to bypass noise, i terms of small variations in amplitude, by only looking at the highest peak value and not possible extra peaks from noise. In Figure 25a the method is displayed where the resolution was considered low, in other terms the data sets were large and each mean value contained many data points. In Figure 25b the method is displayed where the resolution was considered high and each mean value contained fewer data points. The precision was recognised by studying how the blue curve followed the signal.



(a) Peak values with low accuracy.

(b) Peak values with high accuracy.

Figure 25: The mean peak value method for amplitude.

6.2.2 Findpeaks method for amplitude

This method was developed simultaneously as the mean peak value method to enable comparisons between the two methods. The idea behind the findpeaks method was to find the local peak value and the time of the local peak value of each period of the sine wave. The values were placed into two separate vectors for each signal, one for peak values and one for the time values. This made it possible to calculate the loop gain by taking the output's amplitude values divided by the input's amplitude values. This also enabled the phase lag to be calculated which is mentioned in section 6.3.1.

The findpeaks function used two help functions called MinPeakProminence and MinPeakDistance where they have been given arbitrary values below as 1 and 10 respectively.

```
[y_max, y_loc]  = ...
    findpeaks(y,'MinPeakProminence',1,'MinPeakDistance',10);
[u_max, u_loc]  = ...
    findpeaks(u,'MinPeakProminence',1,'MinPeakDistance',10);
```

Those two values had to be changed for every measurement data set which was collected and the output signal had to be divided into several findpeaks functions because of a large decrease in amplitude. The reason for this was that MinPeakProminence determined the least prominence of the signal which meant that for a peak to be recognised it had to have a larger prominence than the one that was preset in the findpeaks function. MinPeakDistance made it possible to decide the smallest distance between two local peaks, this was necessary when the measurement data contained a lot of noise which generated faulty additional peaks for the same period.

6.2.3 Absolute value method

The idea with the absolute value method was to find an easy way of calculating the relation in amplitude between the output and input without having to find each peak value of the sine wave. This method could calculate the total sum of each period of the sine wave and use that to find the loop gain. The amplitude curve could then be drawn using this method when analysing signals with measurements that are constant over each frequency measured. This was done by first normalising the input- and output signal and then taking the absolute value of the respective signal. The absolute values was summarised and the relation in amplitude was calculated by dividing the sum of the output with the sum of the input.

The relation in amplitude for each period between the input- and output signal with the same frequency was described as

$$\frac{\sum_{x=0}^{2\pi} |A_1 \cdot sin(x)|}{\sum_{x=0}^{2\pi} |A_2 \cdot sin(x)|}.$$

The amplitude A_1 and A_2 was written outside the sum and the absolute value was removed and resulting in two positive sine parts from $x = [0, \pi]$

$$\frac{A_1}{A_2} \cdot \frac{\sum_{x=0}^{\pi} 2 \cdot sin(x)}{\sum_{x=0}^{\pi} 2 \cdot sin(x)}$$

The sine waves cancelled out each other and the relation in amplitude remained the same as

$$\frac{A_1}{A_2}.$$

The filter filtfilt was used on the signals for the absolute value method since the collected measurements contained a lot of noise. The method was still working correctly together with the filter which is shown in the following section.

6.3 Phase curve

The following methods were developed and investigated to calculate the phase curve of the Bode plot.

6.3.1 Findpeaks method for phase

The same method was used as in 6.2.2. The function findpeaks not only stored the peak value but also the time for each peak. This method made it possible to find the phase lag between the input- and output signal that was calculated by subtracting the output's peak time with the input's peak time. This method could give the phase lag for each period of a perfect sine wave and was in that sense an accurate method.

6.3.2 Zero-crossing method

This method differed from the previous investigated method for determining the phase curve of the Bode plot since the previous method used the peaks from the sine wave data and this method used the zero-crossing points of each period. The fundamental idea was to normalise the sine wave and use the zero-crossings to calculate the period times and phase lags of each signal.

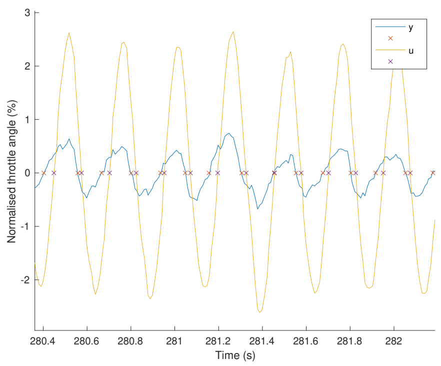
The time difference between the zero-crossings had to be determined as

$$\Delta t = t_{input} - t_{output}.$$

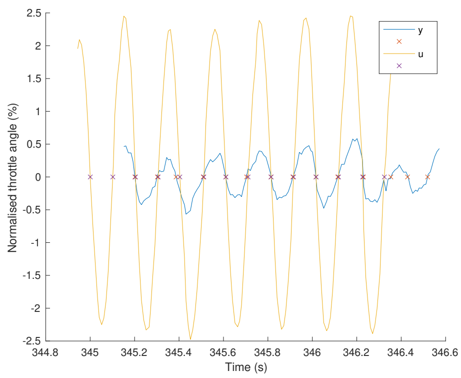
Then, the phase lag was calculated in degrees by multiplying the time difference with the frequency as

$$\varphi_{lag} = \Delta t \cdot f \cdot \frac{180}{\pi}.$$

An example of when this method finds each zero-crossing is displayed in Figure 26a. Since it was important that each crossing from the input was compared to the correct crossing of the output the method also skips the initial unwanted crossings from the output. This can be seen in Figure 26b.



(a) Zero-crossing points for input and output.



(b) Zero-crossing points for input and output from a signal with with -360 of phase shift.

Figure 26: The zero-crossing method for determining phase shift.

7 Results and Discussion

This section presents the results this thesis produced while investigating the problem formulation. The results from comparing the different methods with each other are presented together with the results from the throttle control loop with different tunings of the air charge controller.

7.1 Methods

The methods investigated and used in this thesis to determine stability margins is discussed in this section.

7.1.1 Mean peak value method

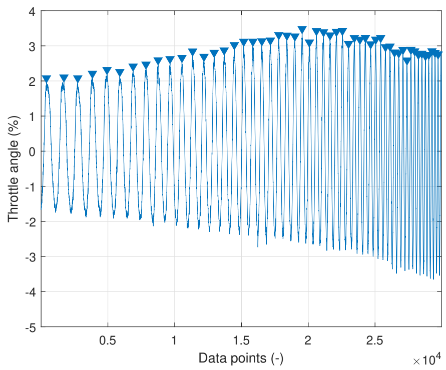
This method needed some manual work to be done before the measurement data could be processed. This involved choosing an appropriate setting for the number of values for each section of data in order for the resolution to be good. The resolution was deemed good when the data was split into sections small enough for the amplitude trend to be captured and big enough to avoid measurement noise. This work did not required as much time as the findpeaks method did, which is discussed in section 7.1.2. This method was considered good because of its simplicity and ability to be changed to match different types of measurement data. It was less affected by the measurement noise compared to the findpeaks method. It also produced a magnitude curve with straight lines between each value therefore the curve was not quickly shifting up and down. This avoided the risk of registering multiple points for the gain cross-over frequency.

The downside of this method was that it could solely store the max peak values. Within each section of the data set it would always take the highest peak and not consider any other peaks. In that sense it interpreted one value as the same value for the whole section and did not identify a mean value. This also made it possible to get results of faulty peaks for when a noise spike was added together with the sinus wave peak. This was however not an issue with small noise. The change in amplitude was a lot larger between each max peak value because of the increase in frequency.

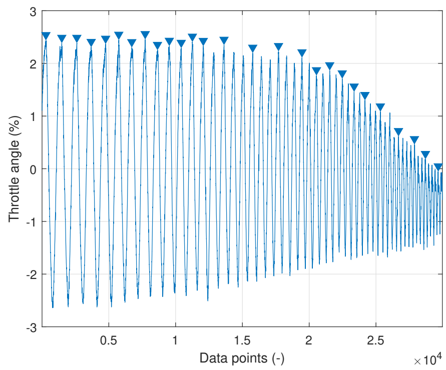
Another downside was that this method could not determine the phase lag since it did not have information about the individual periods like the findpeaks method. This method needed to be complemented with a function that calculated the phase lag to produce the complete Bode plot and could therefore not calculate the stability margins by itself.

7.1.2 Findpeaks method

This method offered an accurate way of finding the correct peak value and peak time for a sine wave with constant frequency and low noise and could generate a complete Bode plot for those cases. However, since the signals for the measurements with ramped sine waves and the static sine waves contained a lot of noise and had a frequency that changed over time there were a lot of problems. One problem was that the registered peak was not always at the centre of the peak because of the measurement noise. This was not an issue when calculating the amplitude but when calculating the phase it produced a lot of variance since it would not compare every period at the same spot each time. This method's accuracy also differed between input- and output signal. The input signal was considerably easier to process with this method compared to the output signal which is seen when comparing the two processed signals in Figure 27. For a too large MinPeakDistance the findpeaks function was unable to capture every peak for the high frequency periods since they came too close to each other which is displayed in Figure 27b.



(a) The normalised input signal.



(b) The normalised output signal.

Figure 27: The findpeaks method for both the normalised input- and output signal from the same measurement data set.

This had to be addressed and fixed and therefore the measurement data was divided into a low frequency data set and a high frequency data set. Then the peaks were able to be identified using two findpeaks functions which could be put together once all peaks had been identified. This method was promising in the beginning but later it turned out to be a time consuming approach to process measurement data with. But most importantly it had a hard time to reproduce accurate and consistent results from measurement data, even when processing two identical measurements. The noise created too much variance in the results and it was therefore deemed unusable.

7.1.3 Absolute value method

The absolute value method was the last amplitude method that was developed and was one of the two proposed methods from this thesis. This method turned out to be a reliant and accurate way to find the relation in amplitude between the input- and output signal. An advantage with this method was that it did not need to find the individual amplitude for each peak. This allowed the method to function even when there was a lot of noise. This made the method more reliable since the other methods had a hard time finding the correct peaks because of the measurement noise. This method did however require a measurement with more than one period for each frequency to function properly since it was taking the relation of the total sum for each signal. This method was therefore only used on the static sine wave tests. This method, like the mean peak value method, could only create the magnitude curve and required the phase curve from another method in order to create the complete Bode plot.

Validation of absolute value method

To validate that this method gave the correct amplitude curve of the Bode plot a validation model was created in Simulink and is presented in Figure 28. The validation was done with several static frequencies similar to how the measurement data looked like from the engine tests. Then the frequencies and amplitudes were interpolated exactly in the same manner as they were going to be for the real tests and compared to the actual Bode plot of the system in Figure 28.

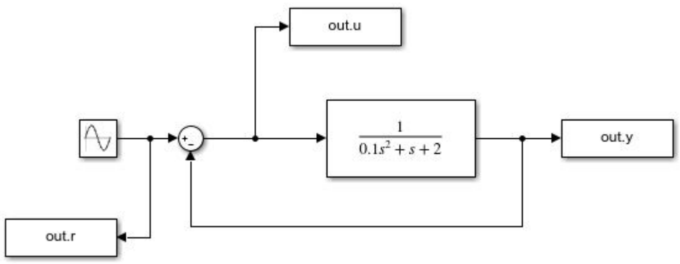


Figure 28: The Simulink model used to collect data for validation of the absolute value method.

With the collected measurement data from Simulink where the Bode plot of the system was known the amplitude curve could be created using the absolute value method. The result is displayed in Figure 29 and as seen was the agreement between the two curves evident.

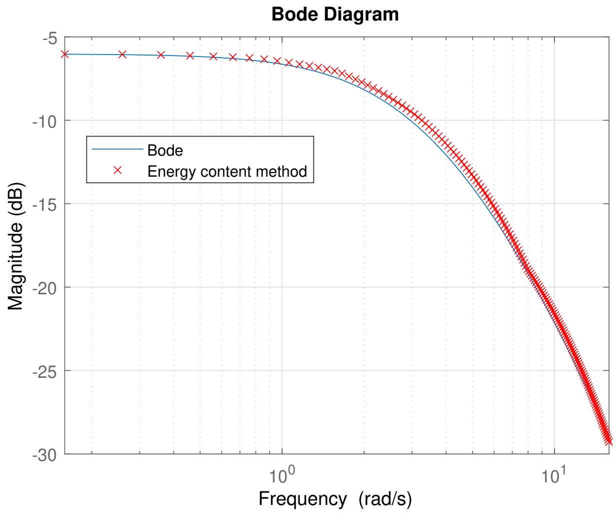


Figure 29: The amplitude curve of the system compared to the absolute value method.

This was a quick method that did not need any manual labour to compute which was a desirable property when this method was designed. The swiftness of this method was a huge advantage compared to the previous two methods which had been investigated, especially compared against the findpeaks method which was a time consuming method.

Considering that the measured data contained a lot of noise the method was tested on a perfect sine wave with induced noise in MATLAB to investigate how well it would perform but also to see if the filter would affect the results. The signal was made with the following code.

```
%%Time specifications:
Fs = 100;                        % samples per second
dt = 1/Fs;                       % seconds per sample
StopTime = 10;                   % seconds
t = (0:dt:StopTime-dt)';                   % milliseconds
t2 = (101*dt:dt:StopTime-dt+101*dt)';      % milliseconds

%%Sine wave:
hz = 1;                         % frequency
A1 = 5;                         % Amplitude 1
A2 = 2;                         % Amplitude 2
noise1 = 2*rand(size(linspace(-2*pi, 2*pi,length(t2))));     % noise1
noise2 = 1.2*rand(size(linspace(-2*pi, 2*pi,length(t))));    % ...
    noise 2
x1 = A1*cos(2*pi*hz*t)+noise1';                               % input
x2 = A2*cos(2*pi*hz*t2 + pi/3)+noise2';                       % output
```

The amplitude of the input and output was $A_1 = 5$ and $A_2 = 2$ respectively. The relation in amplitude between the signals was 0.4 and the absolute value method produced the following results

```
Absolute_value_method_with_filter =

    0.4010

Absolute_value_method_no_filter =

    0.4180
```

where the signals that was tested can be seen in Figure 30

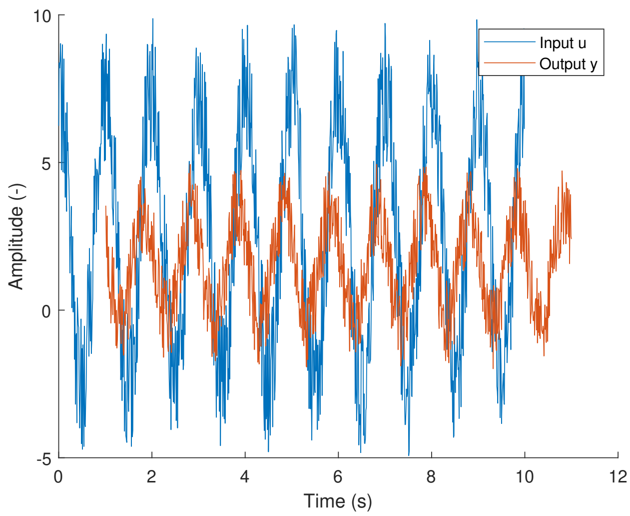


Figure 30: The input and output of a perfect sine wave with induced noise to test the absolute value method.

7.1.4 Zero-crossing method

The zero-crossing method was the other proposed method and was validated with the results displayed in Figure 32. After the validation the method was tested with real measurement data using static sine waves. In the previously mentioned phase method a challenge was to find a consistent way to measure the same spot of the sine wave for each period. The zero-crossing method solved this in the best way by looking at the zero-crossing of the normalised sine wave. Compared to finding the maximum value by looking at the peak of the sine wave, it was easier to find the point where the signal was close to zero. Finding the approximate point where the sine wave was equal to zero was done with interpolation.

The measured signals contained a lot of noise which gave rise to several zero-crossings on each period. Because of that, the decision was made to filter the measurement data with the filter filtfilt in MATLAB. How the signal was filtered can be seen in Figure 13. The mean peak value- and findpeaks method could not be used together with the filter since the amplitude was decreased for the filtered sine wave. This was not a problem for the absolute value method since the relation in amplitude was kept when using the sum of the whole sine wave, see Figure 30

Validation of zero-crossing method

When this method was developed the emphasis was on reliability and thus, the method was tested with perfect measurement data created in Simulink to get rid of measurement noise. To create the perfect measurement data with Simulink a chirp signal was used to represent the sine wave which was created by the oscillator in INCA. The Simulink model which was used is presented in Figure 31. An ordinary stable open-loop transfer function with a feedback was used as the system. From the Simulink model both the input signal, u, and output signal, y, were sent to MATLAB's Workspace where the method was executed.

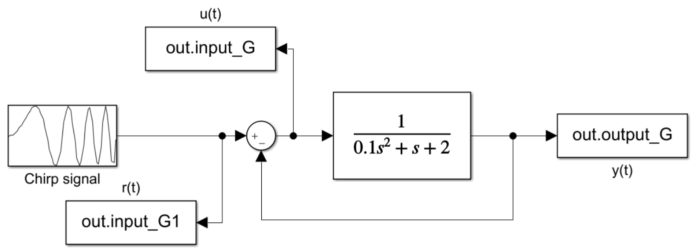
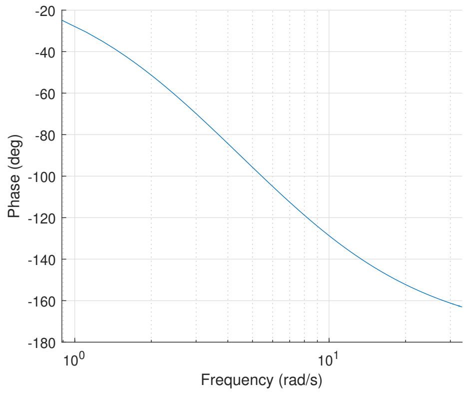
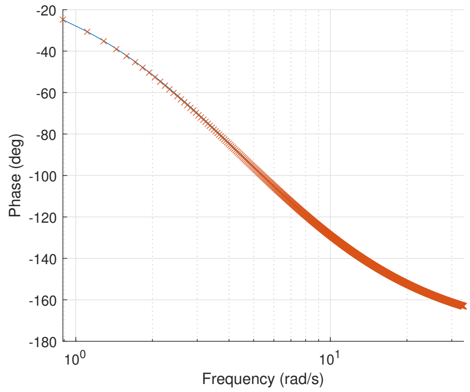


Figure 31: The Simulink model which was used to produce perfect measurement data.

The convenient thing about this method was that the transfer function that was used to produce the signals was known. That, opened up the opportunity to create the phase curve of the Bode plot using the "bode" command in MATLAB where the actual Bode plot for the investigated system could be presented. With the developed method both the frequency and phase of each period could be determined and thus, the Bode plot could be created with those data points from each zero-crossing of every period. When both the actual Bode plot and every frequency and corresponding phase was determined the method was evaluated against the actual Bode plot. The result from this evaluation is displayed in Figure 32 and as seen the agreement of the zero-crossing method with the actual Bode plot was evident.



(a) Bode plot of open loop transfer function.



(b) Bode plot compared with zero-crossing method.

Figure 32: The zero-crossing method compared to the actual Bode plot of the feedback.

7.2 Tests

In Figure 33 the Bode plot is presented, estimated with the absolute value- and zero-crossing method using data from static waves 1. In this test sVcAesHw_X_ThrTar was used as input signal and sVcAesCh_X_ThrPosnTar as output signal. For every measurement it was important to capture the frequencies where the magnitude curve crossed 0dB and where the phase curve crossed $-180°$ to be able to calculate the gain- and phase margin. Both the magnitude curve and phase curve in Figure 33 had to be interpolated to find the approximate frequencies that corresponded to the 0dB- and $-180°$ point. MATLAB's function interp1 was used to perform the linear interpolation. The mean values were chosen as end points and the interpolation was made for every 0.1rad/s increment in frequency.

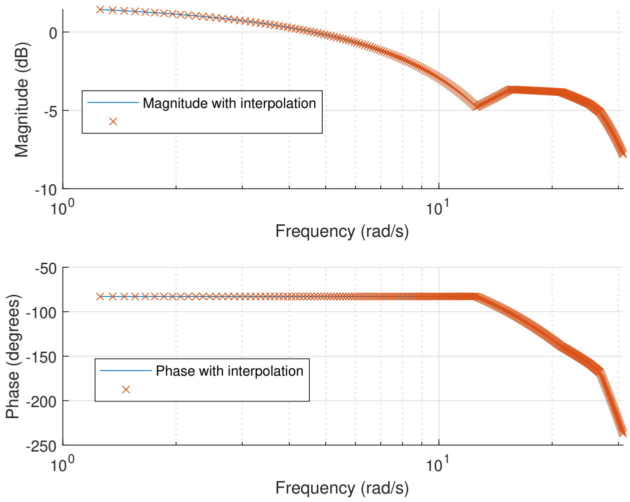
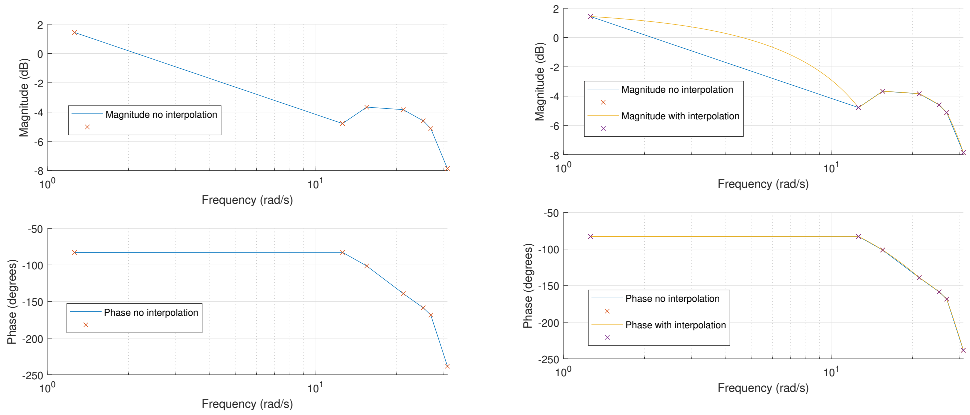


Figure 33: The interpolated Bode plot where the absolute value- and zero-crossing method were used for a poorly calibrated throttle servo control loop with $P = 1$.

In Figure 34a the mean value Bode plot is presented which was before the interpolation of the amplitude and phase had been made. The comparison between the Bode plot before and after the interpolation is displayed in Figure 34b. The phase curve was almost unchanged where the magnitude curve differed more for lower frequencies. This was an issue which caused the 0dB to be unknown since the interpolation between point 1 and 2 was too far away from each other. The phase margin was therefore affected by this uncertainty and the phase margins calculated from the test static waves 1 could not be trusted. The agreement between the interpolated curve and the original curve was a lot better for high frequencies which meant that the gain margin was not affected by the interpolation. The phase margin became 97.18° and the gain margin became 5.55dB for this measurement where $P = 1$.



(a) The mean value Bode plot.

(b) The mean value Bode plot vs. interpolated Bode plot.

Figure 34: The mean value Bode plot and the comparison between mean value Bode and interpolated Bode plot.

7.2.1 Static waves 1

When performing the calibration and tuning of the engine it would be helpful to have a way of knowing if the system became less or more robust after a parameter change. Therefore, three measurements were taken where the first served as a reference to the other two cases. The purpose of using a reference was to see if the robustness had increased and decreased for the other two cases. The hypotheses was that the gain margin should have increased for a less aggressive P-tuning and decreased for the more aggressive case. This would prove that the method was capable of measuring how the robustness had changed after a certain tuning. As seen in Figure 35 when the P-part was increased the magnitude curve increased and when the P-part was decreased the magnitude curve decreased. The same behaviour was observed for the phase curve.

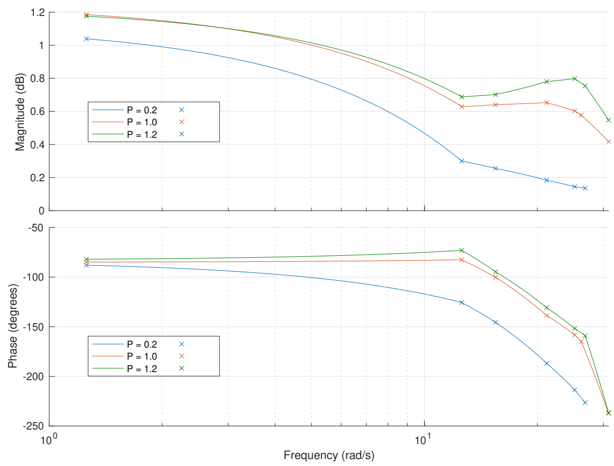


Figure 35: Bode plot compared with three different tuning.

The stability margins for the three measurements are presented in Table 5. The phase margins of this test could not be trusted as mentioned previously in Section 7.2. However, the correct trend in phase margins was observed. The gain margins were not affected by the interpolation since there were several interpolations made around the $-180°$ point.

Table 5: Gain- and phase margins for measurements with different P-parameters.

P-value	Gain margin [dB]	Gain margin [-]	Phase margin [degrees]
0.2	14.25	5.16	90.99
1.0	5.36	1.85	95.74
1.2	3.23	1.45	100.69

Since the measurements contained a lot of noise and a variance within the mean values for each frequency it was necessary to determine if the method and its measurements were reproducible. To confirm if the tests were reproducible, the proposed methods was used on two different measurements of the previously mentioned reference case where they were collected with the same setup and repeated after each other. This allowed for the data to be analysed for reliability since both measurements should produce the same gain- and phase margin. Seen in Figure 36 the magnitude curve was the same for both measurements except at the second data point where there was a slight difference. This was however within the boundaries for the gain margin and did not affect the result. The same result was found for the phase curve which was even more precise and had almost no deviation. The difference in gain margin between the two tests was 3.4% and 1.5% for the phase margin.

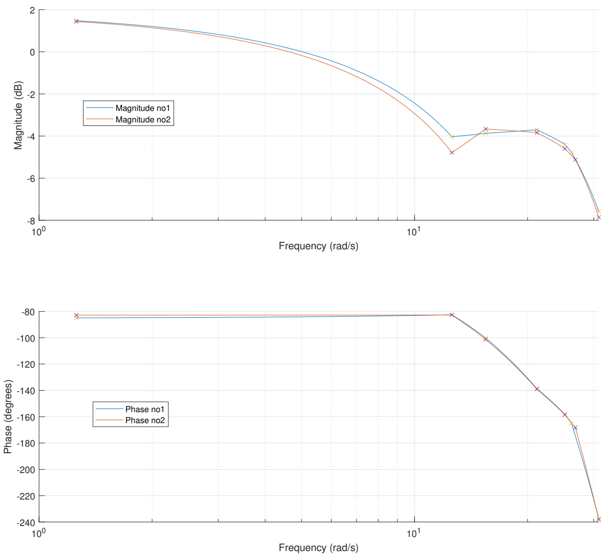


Figure 36: Bode plot compared with two measurements of same setup.

7.2.2 Static waves 2

The Bode plots for all nine measurements are displayed in Figure 37 where it is evident to spot the correct relation between the magnitude of the magnitude curve and the value of the P-part. None of the nine magnitude curves crossed each other where the magnitude curve of $P = 0.2$ was the smallest and $P = 2.1$ was the largest. The static sine waves with period time 0.5s and 0.4s could not be used for any measurement due to large amount of measurement noise.

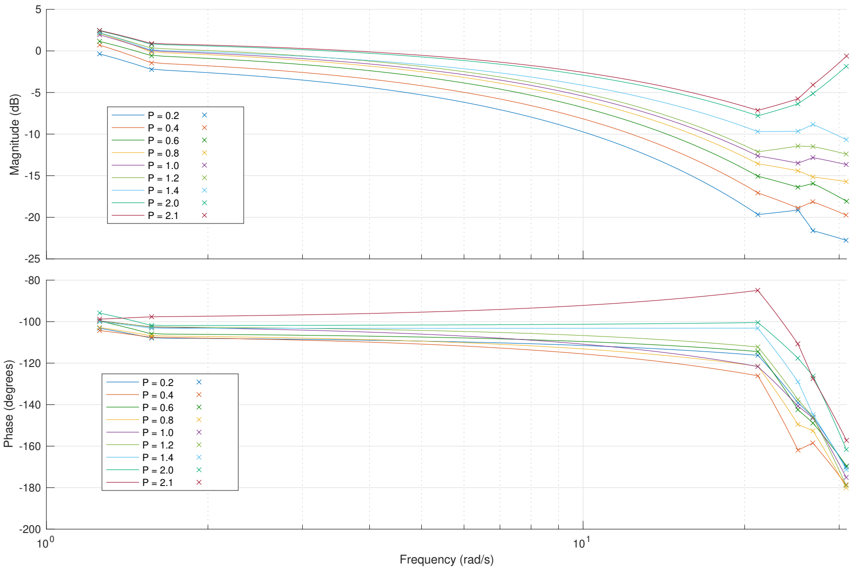


Figure 37: Nine different tunings of the P-part.

The gain- and phase margins of the whole throttle servo control loop was calculated for each test and are displayed in Table 6. The effects of the increased P-part on the gain margin can be seen in Table 6 and as expected, the gain margin was decreased for every P-value which was increased.

Table 6: Trend in gain- and phase margins for measurements with different P-parts.

P-value	Gain margin [dB]	Gain margin [-]	Phase margin [degrees]
0.2	22.75	13.72	76.82
0.4	20.58	10.69	74.66
0.6	18.07	8.01	76.31
0.8	15.70	6.10	73.33
1.0	13.64	4.80	77.21
1.2	12.38	4.16	77.86
1.4	10.64	3.40	76.78
2.0	1.85	1.2374	78.25
2.1	0.62	1.07	83.77

7.2.3 Nonlinearities in throttle

It was also important to determine if the gain margin was correct for each measurement in addition to the correct trend of the gain margin and the P-part of the air charge controller which was displayed in Table 6. The gain margin should have been halved when the P-part was doubled if the P-part had been proportional to the gain margin. This was not the case and that was established even before the gain margins had been calculated. The loop gain which would have been proportional to the investigated loop's gain margin could not be distinguished in INCA. However, it was possible to cause instability by tuning the P-part. The loop was not robust for both $P = 2.0$ and $P = 2.1$ where their gain margins were close to 1, seen in Table 6. Given the first assumption that the P-part was proportional to the gain margin those two measurements indicated that $P \geq 2.25$ would cause instability. This was tested for $P = 2.3$ where the system became unstable at once. This was a validation that the calculated stability margins were close to reality even when there had been simplifications. This was an important validation since it would have become difficult to calculate the correct stability margins theoretically for the control system used in INCA.

The validation was important not only because the margins could not be calculated theoretically but because of the nonlinearities that existed. The stability margins were calculated from a Bode plot which is a linear analysis method. The engine was set in a linear stationary work point since the throttle angle was larger than 10% to get rid of the nonlinear effects from the limp-home position. It was also small enough to be kept in the linear region of the compressible flow restriction of sonic speed since the pressure ratio was smaller than 0.52. There was also dynamically built in nonlinearities in the servo motor. The process of building up flow in the engine was a faster process than slowing it down. That meant that the throttle opened up faster than it closed which was a contributor to the nonlinear behaviour of the servo motor. This was also observed visually on each collected measurement data where the sine waves had different inclination upwards and downwards.

7.3 System identification

The results produced from the measurements collected with ControlDesk were bland since they could not be confirmed nor compared to any other results. However, it was satisfactory to see that the Nyquist curve could be used to estimate the stability margins for that particular loop even if they could not be used further in this thesis. The signals which were used in ControlDesk did not exist in INCA and a comparison between the static sine waves was not possible. The results from INCA were also not satisfactory even when the exact same signals were used for the static sine waves tests and the PRBS-test. Better models could have been estimated from measurements collected in INCA if other signals had been used in SITB for the system identification. The reason why that path was not investigated was because the results it would have produced could not have been compared with the results from the static sine wave test. With this insight on the possibilities of using the PRBS-signal in INCA the decision was made to not continue with this approach and instead focus on the static sine waves which produced credible results.

8 Conclusion and Future Work

In this thesis different methods for estimating stability margins have been developed for the air charge control loop. Additionally, estimations of the stability margins of the same control loop but with different tunings has been made. The main contribution of this work was to prove the possibility to estimate the stability margins from experimental measurements along with the increased or decreased robustness when changing the gain of the control loop. This chapter presents the conclusions of this thesis together with suggested future work.

8.1 Conclusions

This thesis has looked into the amplitude- and gain margins for the air charge control loop. The focus was to develop a reliable test method, investigate different tunings and shape the foundation for future work of this subject for PES and LiU.

A variety of test methods and possible solutions have been presented. The methods differ from each other in both approach and ability to estimate the stability margins. It was established that the methods that was utilising the peaks to find amplitude and phase that was struggling the most, therefore alternative methods was proposed to solve these problems. The two methods that solved this problem in the best way was the zero-crossing method and absolute value method which could estimate a complete Bode plot once they were combined. The zero-crossing method estimated the phase shift. The absolute value method could not estimate the individual amplitude of the input and output but that was not a problem since the purpose was only to find the relation in amplitude.

System identification was also considered as a possible method for determining the stability margins of the investigated system along with the test methods developed for the sine waves. This approach was deemed insufficient for the throttle servo control loop of the engine's control system. The purpose of investigating system identification was to compare the computed stability margins with the time consuming method that used sine waves. This could not be done with the signal Hw as input and Ch as output, because of that the decision to stop investigating system identification methods was taken.

Tests when varying the tuning of the P-part of the air charge controller were conducted and the stability margins were calculated and presented. From the two tests static waves 1 and 2 a multitude of static points were measured and compared. In static waves 1 the hypotheses that the amplitude would linearly follow the change in of the P-part was disproved. The amplitude did however change with a similar amount. Therefore, the more extensive test static waves 2 was done. With the measurements from static waves 2 it was concluded that the stability margins were for all tested frequencies getting worse for a larger P-part and better for a lower P-part. The proposed method was consequently presenting a decrease or increase of the stability margins with good accuracy. After the tests were conducted additional knowledge arose regarding the control system and coupling of the signals which was contributing with nonlinearities. With this in mind the results presented was promising and can further be improved upon.

8.2 Future work

The long term intention of the sine wave method is to be able to identify the real stability margins of a control loop of a complex system. Further experiments and tests are required to get better results and a wider understanding of the control loop. First, a more comprehensive mapping of the control loop has to be made. Secondly, all possible parts that can contribute to change in the gain needs to be accounted for.

The air charge control loop is a complex loop where some signals are multiplied with gains while others are summed together or fed back into the loop. This increases the difficulty to find a linear relation between how a single variable influences the system and how far the system is from instability. Therefore, knowledge of how these couplings affect each other and how the individual signals behave would enable a more true linear comparison and analysis of the control loop.

When the ability to determine the stability margins is achieved a lot of possibilities exists. One interesting topic to investigate is if the robustness can be increased with a different PID-tuning method than Ziegler-Nichols which is currently used today. Another is to see if any cross-coupling effects can be distinguished between loops within the air flow control loop to see if the robustness of intermediate loops increases or decreases when changes have been made inside the throttle servo control loop.

Appendix

A MATLAB code and functions

To be able to use the stored data from a test in INCA in MATLAB it is first necessary to convert it. First it is needed to manipulate the file format of the data whilst keeping its information intact. This is done by putting the data into a '*data*' format and then into a MATLAB time table. Next the header names (names for each signal measured) is saved and then the table is converted into a cell array. Then the new header positions are located and stored. And lastly the cell is made into a MATLAB matrix with all signals intact with its corresponding headers. The headers in '*Vrbls*' is used in the *find* function to locate the data that is needed. Because INCA often automatically changes the positions of the signals in the data file it was necessary to search for the respective signals by name.

```
function [Matrix_sorted] = data_INCA(data)
% Convert data to time table
tt_data      = data{6};
t_data       = timetable2table(tt_data);
% Save headers
headers      = t_data.Properties.VariableNames;
% Convert To Cell Array
Cell         = table2cell(t_data);
NumberIdx       = cellfun(@isnumeric,Cell);
% Get Corresponding Headers
Vrbls        = headers(NumberIdx(1,:));
Values         = reshape(Cell(NumberIdx), size(Cell,1), []);
% Reshape Vector To Matrix
Matrix          = cell2mat(Values);
% Sorting and finding data
u = find(ismember(Vrbls,'sVcAesHw_X_ThrTar_ETKC_1'));
y = find(ismember(Vrbls,'sVcAesCh_X_ThrPosnTar_ETKC_1'));
t = find(ismember(Vrbls,'time'));
Oscr_in = find(ismember(Vrbls,'sVcAesCo_Z_Oscr_ETKC_1'));
Oscr_p = find(ismember(Vrbls,'rVcAesCo_t_OscrPerd_ETKC_1'));
Matrix_sorted = Matrix(:,[u,y,t,Oscr_in,Oscr_p]);
end
```

The function *phase_freq* returns output data for phase, frequency and amplitude for a given set of input data u, y, ref and tau where u is the input into the system, y is the output, ref is the reference signal of the disturbance, in our case the sinus wave and tau is the value for how soft or hard the filtfilt function filters the input signal in *zero_location*.

```
function [phase,freq,amp] = phase_freq(u,y,ref,tau)
u(:,1)         = u(:,1) - mean(u(:,1));
y(:,1)         = y(:,1) - mean(y(:,1));
y_zerotime     = zero_location(y(:,1),y(:,2),tau);
```

```
u_zerotime   = zero_location(u(:,1),u(:,2),tau);
ref_zerotime = zero_location(ref(:,1),ref(:,2),tau);

u_zerotime   = ...
    u_zerotime(1:min([length(y_zerotime),length(u_zerotime)...
                ,length(ref_zerotime)]));
y_zerotime   = ...
    y_zerotime(1:min([length(y_zerotime),length(u_zerotime)...
                ,length(ref_zerotime)]));

t_lag        = (u_zerotime-y_zerotime);
period       = (zeros(length(t_lag)-2,1));

for i = 3:length(t_lag)
    period(i-2) = ref_zerotime(i) - ref_zerotime(i-2);
end

freq         = (2*pi) ./ period;
phase        = ((180/pi) .* freq .* t_lag(1:length(period)) + 180);
amp          = sum(abs(y(:,1))) / sum(abs(u(:,1)));
end
```

The function *zero_location* return all times stamps for when a the input signal does zero-crossings. The input to the function is *in_data* which is a normalized signal, *t* is the time vector for the input signal and *tau* is the value for how soft or hard the filtfilt function filters in signal.

```
function [t_zero] = zero_location(in_data,t,tau)
Ts           = 0.01;
K            = exp(-Ts / tau);
k_filtfilt   = 1 - K;
data_ff      = filtfilt(k_filtfilt,[1 -(1-k_filtfilt)],in_data);

bef          = zeros(1,length(t))';
aft          = zeros(1,length(t))';
t_bef        = zeros(1,length(t))';
t_aft        = zeros(1,length(t))';

for i = 2:length(data_ff)
    if data_ff(i-1) * data_ff(i) < 0
        bef(i-1)   = data_ff(i-1);
        aft(i)     = data_ff(i);
        t_bef(i-1) = t(i-1);
        t_aft(i)   = t(i);
    end
end

bef_abs     = bef(bef≠0);
aft_abs     = aft(aft≠0);
t_bef_abs   = t(bef≠0);
t_aft_abs   = t(aft≠0);
t_zero      = zeros(length(bef_abs),1);

for j = 1:length(bef_abs)
    time        = [t_bef_abs(j);t_aft_abs(j)];
```

```
    values      = [bef_abs(j);aft_abs(j)];
    ti          = (time(1):0.00001:time(end))';
    yi          = interp1q(time,values,ti);
    [¬,nearest] = (min(abs(yi)));
    t_zero(j)   = ti(nearest);
end
end
```

The function *bode_interp_plot* is used to plot bode plots from experimental data. The inputs are *phase* (Phase), *freq* (frequency) and *amp* (amplitude) and should be of same length. The frequency relates to each value by matching the index position of each vector. For example is $phase(3)$ the phase at frequency $freq(3)$. This thesis measured many periods at the same frequency to get a proper estimate at each stationary point. The values are more accurate but the data may only contain as low as 7 values. Therefore interpolation is utilized with the function $interp1$. This allows for a smoother bode plot and helps finding a more suitable cross-over frequencies and boundaries for stability margins.

```
function [] = bode_interp_plot(phase,freq,amp)
xq = [freq:0.1:freq(end)];
amp_int = interp1(freq,amp,xq);
phase_int = interp1(freq,phase,xq);

subplot(2,1,1);
hold on
    plot(xq,20*log10(amp_int))
    plot(freq,20*log10(amp),'x')
hold off
grid on
set(gca, 'XScale', 'log')
set(gca,'xticklabel',{[]})
ylabel('Magnitude (dB)')

subplot(2,1,2);
hold on
    plot(xq,phase_int)
    plot(freq,phase,'x')
hold off
set(gca, 'XScale', 'log')
xlabel('Frequency (rad/s)')
ylabel('Phase (degrees)')
grid on
end
```

References

[1] B. Verma and P. K. Padhy. "Robust Fine Tuning of Optimal PID Controller With Guaranteed Robustness". In: *IEEE Transactions on Industrial Electronics* 67.6 (2020), pp. 4911–4920. DOI: 10.1109/TIE.2019.2924603.

[2] E. H. V. Díaz and M. A. C. Presentación. "Closed loop identification in a four coupled tanks system". In: *2016 IEEE International Conference on Automatica (ICA-ACCA)*. 2016, pp. 1–4. DOI: 10.1109/ICA-ACCA.2016.7778437.

[3] C. Chen, H. Tsai, and Y. Lin. "Servo control design for electronic throttle valve with nonlinear spring effect". In: *2010 11th IEEE International Workshop on Advanced Motion Control (AMC)*. 2010, pp. 88–93. DOI: 10.1109/AMC.2010.5464020.

[4] Y. Naung et al. "Implementation of data driven control system of DC motor by using system identification process". In: *2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*. 2018, pp. 1801–1804. DOI: 10.1109/EIConRus.2018.8317455.

[5] Daniel Belega, Dario Petri, and Dominique Dallet. "A frequency-domain linear least-squares approach for complex sine-wave amplitude and phase estimation". In: *2017 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*. 2017, pp. 1–6. DOI: 10.1109/I2MTC.2017.7969665.

[6] Daniel Belega and Dario Petri. "Effect of Frequency Uncertainty on the Sine-wave Amplitude Estimator Returned by Linear Least-Squares Fitting". In: *2020 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*. 2020, pp. 1–5. DOI: 10.1109/I2MTC43012.2020.9129349.

[7] Shanglin Ye and Elias Aboutanios. "An algorithm for the parameter estimation of multiple superimposed exponentials in noise". In: *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2015, pp. 3457–3461. DOI: 10.1109/ICASSP.2015.7178613.

[8] M. Özkan and Ç. Kasnakoğlu. "Active Fault Detection in Linear Controller Hardware with Sine Signal". In: *2020 7th International Conference on Electrical and Electronics Engineering (ICEEE)*. 2020, pp. 90–94. DOI: 10.1109/ICEEE49618.2020.9102484.

[9] Holger Boche and Ullrich J. Mönich. "Signal reconstruction from sine wave crossings". In: *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2011, pp. 4196–4199. DOI: 10.1109/ICASSP.2011.5947278.

[10] B. Safarinejadian, N. Kianpour, and M. Asad. "A novel identification method for fractional-order wiener systems with PRBS input". In: *2016 4th International Conference on Control, Instrumentation, and Automation (ICCIA)*. 2016, pp. 290–295. DOI: 10.1109/ICCIAutom.2016.7483176.

[11] E. S. Yadav and T. Indiran. “PRBS Based Identification and Conditional Control for an Optimal Operation of a Pilot Plant Binary Distillation Column”. In: *2019 8th International Conference on Modeling Simulation and Applied Optimization (ICMSAO)*. 2019, pp. 1–5. DOI: 10.1109/ICMSAO.2019.8880422.

[12] Sunil Kumar Singh and Prabin Kumar Padhy. “Design of PID controller for delayed systems using gain margin and phase margin criteria”. In: *2017 4th International Conference on Power, Control Embedded Systems (ICPCES)*. 2017, pp. 1–6. DOI: 10.1109/ICPCES.2017.8117658.

[13] Nevra Bayhan and Mehmet Turan Soylemez. “A new technique for calculation of maximum achievable gain and phase margins with proportional control”. In: *2007 Mediterranean Conference on Control Automation*. 2007, pp. 1–6. DOI: 10.1109/MED.2007.4433865.

[14] Mircea Dulua and Stelian-Emilian Oltean. “Simulations of Robust Control of the Throttle Valve Position”. In: *2020 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*. 2020, pp. 1–5. DOI: 10.1109/AQTR49680.2020.9129912.

[15] Paolo Mercorelli. “An Optimal Minimum Phase Approximating PD Regulator for Robust Control of a Throttle Plate”. In: *Proceedings of the 45th IEEE Conference on Decision and Control*. 2006, pp. 3572–3576. DOI: 10.1109/CDC.2006.376842.

[16] Lars Eriksson and Lars Nielsen. *Modeling and control of engines and drivelines*. West Sussex, United Kingdom: John Wiley and Sons Ltd, 2014.

[17] Martin Enqvist et al. *Industriell reglerteknik - Kurskompendium*. Linköping: Linköpings universitet, 2014.

[18] Torkel Glad and Lennart Ljung. *Reglerteknik - Grundläggande teori*. Lund: Studentlitteratur, 1981.

[19] Torkel Glad and Lennart Ljung. *Reglerteori*. Lund: Studentlitteratur, 1997.