INTERNSHIP REPORT
FACULTY OF ENGINEERING TECHNOLOGY

ACTIVE FEEDBACK ACOUSTIC NOISE CONTROL

L.R. Kamphuis
S1012428

Supervised by:
Prof.dr.ir. R.A. de Callafon
Dr.ir. R.G.K.M. Aarts

Enschede, June 11, 2014

UNIVERSITY OF TWENTE.

CONTEXT

This report contains a comprehensive overview of my internship at the University of California in San Diego (UCSD). During my internship, started in January 2014, I did research in the field of active noise cancellation within the Dynamic Systems & Control group at the Department of Mechanical and Aerospace Engineering (MAE) at UCSD. My supervisors were Professor Raymond de Callafon, from the dept. of MAE at UCSD, and associate professor (UHD) Ronald Aarts from the Faculty of Engineering Technology (CTW) at the University of Twente.

ABSTRACT

Active noise control is applied at an air conditioning (AC) air duct with incoming unknown noise from an input speaker, output microphone and a feedback controller. Positioning the speaker downstream to the microphone, according to the flow direction of the incoming sound, a feedback loop can be created. Sound delay will be an issue as introduced by sound traveling over a certain distance in time. In practice, high frequency sounds will be reduced passively by a damper part in the air duct, remaining lower frequencies will be important for the active noise control loop. After identifying the sound paths in the air duct, low order stable feedback controllers can be formed using loop shaping or H_2 control methods. Both controllers are created in such a way that the sensitivity function of the closed loop system is stable for all frequencies. Overall stability is required but influences the control applicability at certain frequencies controlling with a very small gain for stability. To overcome this problem a real-time Youla Updating controller is used to update the controller real-time. Shaping the controller real-time, based on minimizing the 2-norm of the microphone output, will result in a better controller for the unknown incoming sound. Especially for incoming sound with a few dominating frequencies the result is much better. Finally, an actively controller sound reduction can be achieved in the lower frequency range of $50 - 300\ Hz$.

TABLE OF CONTENTS

INTRODUCTION

In this report an AC air duct is used for active sound control. Inside the air duct a microphone and control speaker are positioned in such a way that a closed loop system between microphone and speaker can be created. Using static controllers (loop shaped or H_2) and real-time updating controllers the incoming sound will be measured and reduced as much as possible. The main question in this report will be: “Is it possible to reduce incoming sound using an active control feedback loop?”

Previous research with the same AC air duct is done using a feedforward static H_2 controller in [19]. Obtaining a H_2 controller for a standard control problem is show in [8], [4] and [7]. The updating FIR filter used in the real time Updating Youla controller is also applied on different systems with similar properties in [11], [12], [14] and [18] in which the updating algorithm is explained in [13] and [16]. To obtain overall stability in the updating algorithm a Youla parameterization is used as explained in [15] with more details about coprime factorization in [2] and [3]. Other approaches for active noise control in an air duct are shown in [17] and [6].

Following the chapters in this report will show the same sequence as the research has taken place in practice. Starting with explaining the set up and AC air duct configuration in chapter I with measuring and analyzing all the acoustic sound paths in chapter II. Identifying the sound paths in chapter III to create nominal models for computer simulations and control purposes. In chapter IV the nominal models are used to simulate the set up in practice including filters and ADC-DAC conversion. Continuing in chapter V with applying a loop shaped controller and in chapter VI an H_2 controller will be applied to the closed loop system. To improve the noise reduction even further a Youla Updating controller is introduced in chapter 0. In chapter VIII the system Set up is changed by separating the error- and control microphone instead of using one microphone for both purposes. SIMULINK models for simulations and practice are shown in chapter IX and the report ends with conclusions in chapter X and recommendations in chapter XI.

I SET UP

In this chapter the AC air duct in practice will be shown and discussed in detail. Schematic models corresponding to the sound paths in practice will be created. These models will be used to design a stable feedback loop of the system further on. A CAD model[1] of the air duct is shown in Figure 1

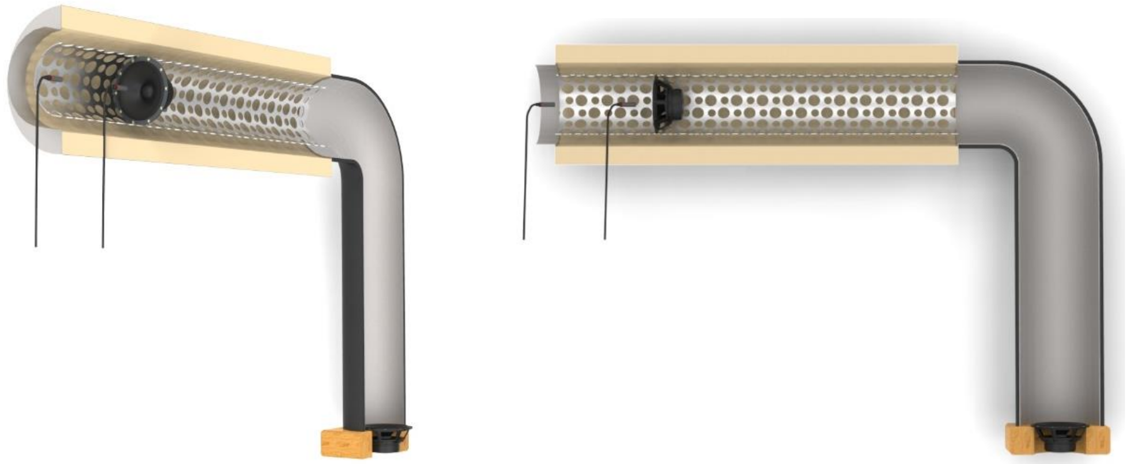

Figure 1: Section views of air conditioning air duct

A schematic representation of the air duct is presented in Figure 2, with the dimensions summarized in Table 1. One end (right) of the air duct is closed and the other end (left) is open. The main part of the air duct consists of foam to create a high frequency passive damper. This damping section has a length of l_d and an inner diameter of d_i, the approximated total length of the air duct is given by l.

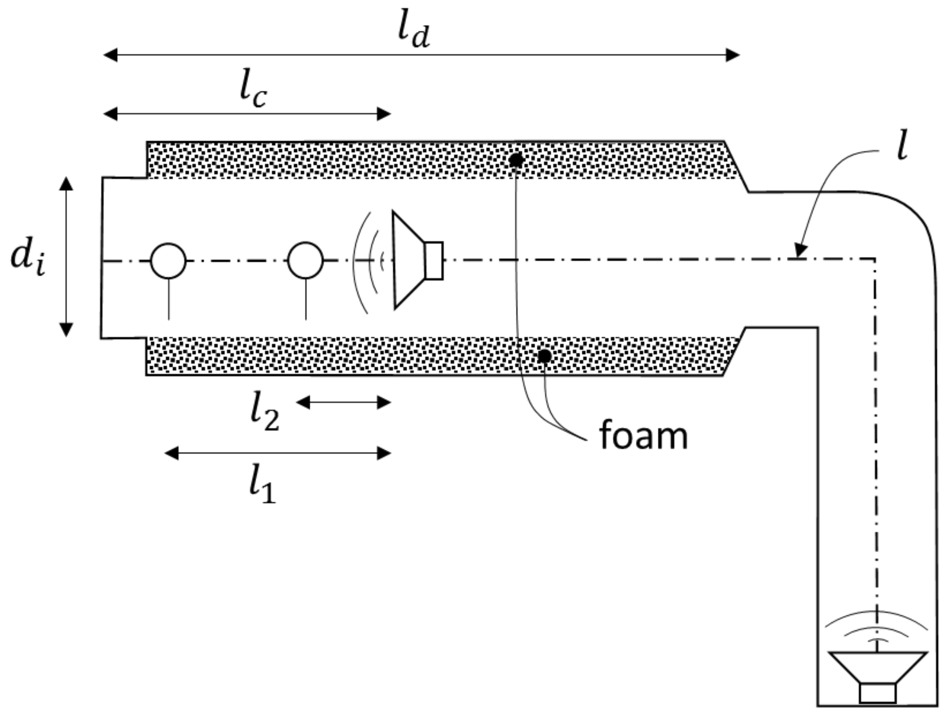

Figure 2: Schematic view air conditioning air duct

[1] SolidWorks Corp.

Two speakers and two microphones are positioned at certain positions inside the air duct. The first speaker, the so called 'Ground speaker', is positioned at the closed right end of the duct and produces an input $n(t)$. A second speaker 'Control speaker' is positioned in the middle of the duct at a distance l_c to the open end and represents input $u(t)$. Both microphones are placed downstream of the control speaker with a distance l_1 and l_2. The output of the microphones are $e_1(t)$ and $e_2(t)$, respectively.

Description	Symbol	Value
Inner diameter	d_i	$0.2\ m$
Length of air duct	l	$2.4\ m$
Length of damper	l_d	$1.4\ m$
Position microphone 1	l_1	$0.25\ m$
Position microphone 2	l_2	$0.05\ m$
Position Control speaker	l_c	$0.30\ m$

Table 1: Properties air duct

The whole set up contains two inputs $n(t)$ and $u(t)$ and two outputs $e_1(t)$ and $e_2(t)$ so in total four acoustic paths can be composed.

$$\frac{e_1(t)}{n(t)} = H_1 \qquad \text{(I-1)}$$

$$\frac{e_2(t)}{n(t)} = H_2 \qquad \text{(I-2)}$$

$$\frac{e_1(t)}{u(t)} = G_1 \qquad \text{(I-3)}$$

$$\frac{e_2(t)}{u(t)} = G_2 \qquad \text{(I-4)}$$

A schematic overview of these acoustic paths is given in Figure 3.

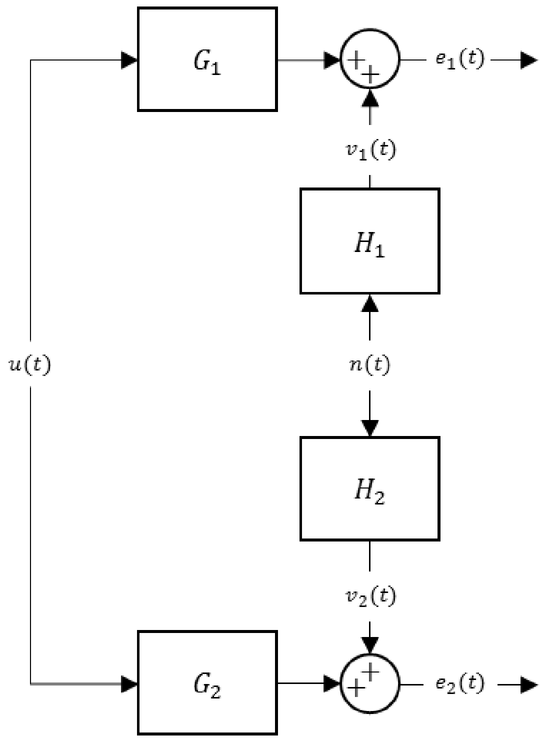

Figure 3: Schematic overview air duct

Both microphone outputs can be written in terms of the speaker inputs $u(t)$ and $n(t)$ according to.

$$e_1(t) = G_1 u(t) + H_1 n(t) \tag{I-5}$$
$$e_2(t) = G_2 u(t) + H_2 n(t) \tag{I-6}$$

In which $n(t)$ is the input of the air duct at the ground speaker and assumed to be unknown. The other input $u(t)$ is a customizable input and can be used to create a control system. For example, create a close loop between $e_2(t)$ and $u(t)$, will be named: Set up 2, or between $e_1(t)$ and $u(t)$, labeled as Set up 1. Both closed loop configuration are shown in Figure 4 and Figure 5 with unknown controller C. Based on the stability criteria explained in the next paragraph, the controller will be applied according to the Set up 2 configuration, see Figure 5. Remark: the closed loop equations for set up 1 and Set up 2 are identical for output $e_1(t)$ and $e_2(t)$ respectively.

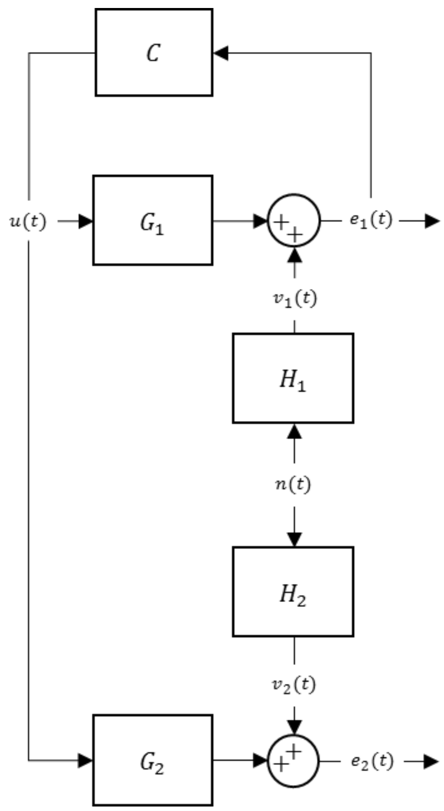

Figure 4: Set up 1

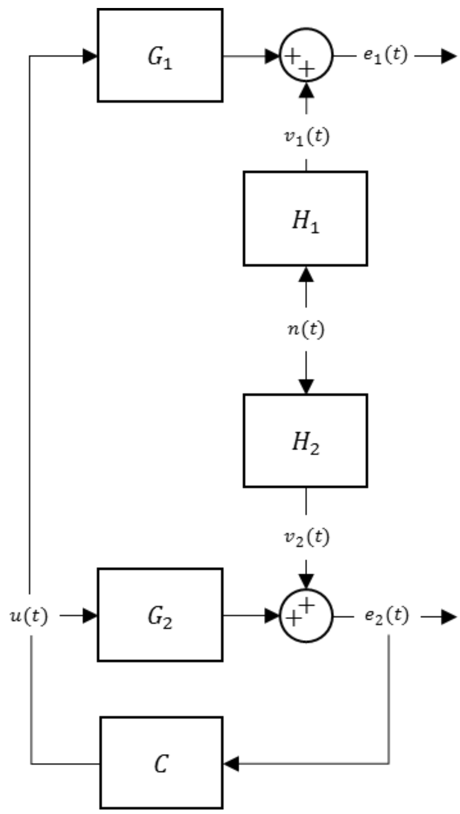

Figure 5: Set up 2

SET UP 2

The input $u(t)$ of Set up 2 can be written as.

$$u(t) = Ce_2(t) \tag{I-7}$$

Substitute (I-7) into (I-5) and (I-6).

$$e_1(t) = G_1 C e_2(t) + H_1 n(t) \tag{I-8}$$
$$e_2(t) = G_2 C e_2(t) + H_2 n(t) \tag{I-9}$$

Rewrite (I-9) to obtain the transfer function from the ground speaker to microphone 2.

$$e_2(t) = \frac{H_2}{1 - G_2 C} n(t) \tag{I-10}$$

In which the sensitivity function S and closed loop gain L are given by.

$$S = \frac{1}{1 - G_2 C} \tag{I-11}$$

$$L = -G_2 C \tag{I-12}$$

Substitute (I-10) into (I-8) to obtain the transfer function from the ground speaker to microphone 1.

$$e_1(t) = \left(H_1 + H_2 \frac{G_1 C}{1 - G_2 C}\right) n(t) \tag{I-13}$$

STABILITY

To obtain a stable feedback system the controller C must stabilize the sensitivity function S in (I-11). Assume all acoustic paths are stable, so G_2 in (I-11) is stable as well. Stability will be obtained if, and only if, the following three rules are satisfied.

I. The closed loop gain L must be stable.
II. The magnitude of the closed loop gain must be smaller than unity if the phase is $\pm 180°$.
$|L| < 1$ if $\angle(L) = \pm 180°$
III. The phase of the closed loop gain cannot be $\pm 180°$ if the magnitude is larger than unity.
$\angle(L) \neq \pm 180°$ if $|L| \geq 1$.

A stabilizing controller C will change the closed loop gain L to satisfy the three stability criteria. By looking at the closed loop gain of Set up 2 (I-12), the acoustic path G_2 introduces delay to the closed loop gain in such a way that the closed loop gain is not stable for $C = 1$ (look at Figure 7 to see this delay). An even worse result would appear if Set up 1 is considered introducing more delay by acoustic path G_1 instead of G_2 given the larger distance between the control speaker and microphone. In both situations a stable controller cannot cancel this delay by introducing something like 'inverse delay'. The only stable solution is obtained by reducing the magnitude of the closed loop gain at a phase of $\pm 180°$ according to stability criteria two. Given a smaller change in phase for acoustic path G_2 compared to G_1, see Figure 7, the restrictions on the controller will be less for Set up 2. In other words, the performance of the system is expected to be better in the situation of Set up 2 compared to Set up 1. For now Set up 1 will be neglected and Set up 2 will be used as the controllable feedback system.

IDEAL SITUATION

If H_1 equals H_2 and G_1 equals G_2, the transfer function from input $n(t)$ to output $e_1(t)$ in (I-13) must convergence to the transfer function from input $n(t)$ to output $e_2(t)$ in (I-10).

Assume $H = H_1 = H_2$ and $G = G_1 = G_2$ and substitute into (I-13).

$$e_1(t) = \left(1 + \frac{GC}{1 - GC}\right) Hn(t) \tag{I-14}$$

Substitute $1 = \frac{1-GC}{1-GC}$ and rewrite.

$$e_1(t) = \frac{H}{1 - GC} n(t) = e_2(t) \tag{I-15}$$

As expected, the error expressions (I-13) convergence to (I-10) when both microphones are positioned at the same spot $(l_1 = l_2)$.

II MEASUREMENTS

With the schematic set up of the AC air duct given, the only unknowns are the four assumed acoustic paths H_1, H_2, G_1 and G_2 as shown in (I-1) - (I-4). Those paths will be measured by applying random noise on the inputs $n(t)$ or $u(t)$ and measuring the outputs $e_1(t)$ or $e_2(t)$. A *Hewlett Packard 3563A Control systems analyzer* [25] is used to apply the random noise input, measure the output and calculate the frequency response between both.

CUT OF FREQUENCY

Before a random noise signal is applied the cut of frequency of the AC air duct will be calculated. According to Ref. [17] the cut of frequency of a circular air duct can be calculated by.

$$f_{cut-off} = 0.293\frac{c}{a} \approx 1000\ Hz \tag{II-1}$$

For our system the AC air duct radius $a = d_i/2 = 0.1\ m$ and the speed of sound $c = 343\ m/s$.

ACOUSTIC PATHS H_1 AND H_2

By applying a random noise on input $n(t)$ and assuming $u(t) = 0$ the acoustic paths H_1 and H_2 can be measured. The range of the random noise generator is set from $1 \rightarrow 2000\ Hz$, with $2000\ Hz$ the Nyquist frequency of the sample frequency used in practice (will be explained in chapter IV). For now a sample frequency of $F_s = 4000\ Hz$ will be assumed. Both frequency responses of acoustic paths H_1 and H_2 are shown in Figure 6.

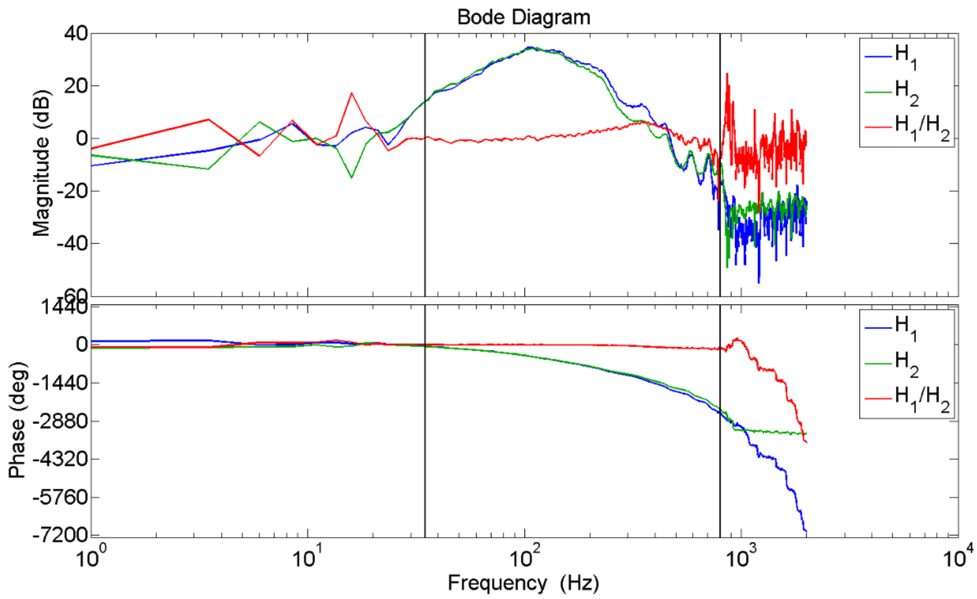

Figure 6: Frequency response H_1 and H_2

Clearly visible in the frequency response is the influence of the damping part of the acoustic air duct. The magnitude of all frequencies above 800 Hz (right black line) are almost neglectable. On the left side, below frequencies of 35 Hz (left black line), the ground speaker is not able to produce any sound and a unreal response is measured. Both restrictions will determine the useful frequency response range by $35 - 800\ Hz$, visible as the area between the two black lines.

Resonance frequencies

In Figure 6 several resonance frequencies are visible, at for example $106\ Hz$ which shows the largest magnitude of the acoustic air duct. Six clearly visible resonance frequencies can be obtained from Figure 6.

$106\ Hz \qquad 136\ Hz \qquad 346\ Hz \qquad 450\ Hz \qquad 590\ Hz \qquad 710\ Hz$

Those frequencies are the resonance frequencies of the AC air duct and can be calculated theoretically according to Ref. [20]. The resonance frequency of a tube with a closed and an open end can be calculated by.

$$f_n = \frac{nc}{4(l + 0.4d_i)} \tag{II-2}$$

Similar, the resonance frequency of a two-sided open tube, can be calculated by.

$$f_n = \frac{nc}{2(l + 0.3d_i)} \tag{II-3}$$

The first 8 resonance frequencies of the whole air duct are calculated according to (II-2).

f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
$35\ Hz$	$69\ Hz$	$\mathbf{104\ Hz}$	$\mathbf{138\ Hz}$	$173\ Hz$	$207\ Hz$	$242\ Hz$	$277\ Hz$

In which the third and fourth resonance frequency do correspond to the first two visible frequencies. Since the damping part of the air duct introduces a small volume change in the air duct, the damping part can be seen as a two ended open air duct with length l_d.

The first 8 resonance frequencies of the damping part are calculated according to (II-3).

f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
$117\ Hz$	$235\ Hz$	$\mathbf{352\ Hz}$	$\mathbf{470\ Hz}$	$\mathbf{587\ Hz}$	$\mathbf{705\ Hz}$	$822\ Hz$	$940\ Hz$

In which the third to sixth resonance frequencies are almost identical to the visible frequencies in the frequency response. Somewhere between $136 - 352\ Hz$ the frequency response changes from the whole air duct to only the damping part. This is also a region where no sufficient resonance peaks are visible. Given the background of the peaks in the measured frequency response the measurements seem to be correct and useful.

ACOUSTIC PATHS G_1 AND G_2

Similar measurements for acoustic paths G_1 and G_2 are provided. In this situation $u(t)$ will be the random noise input and $n(t) = 0$. The frequency responses are shown in Figure 7.

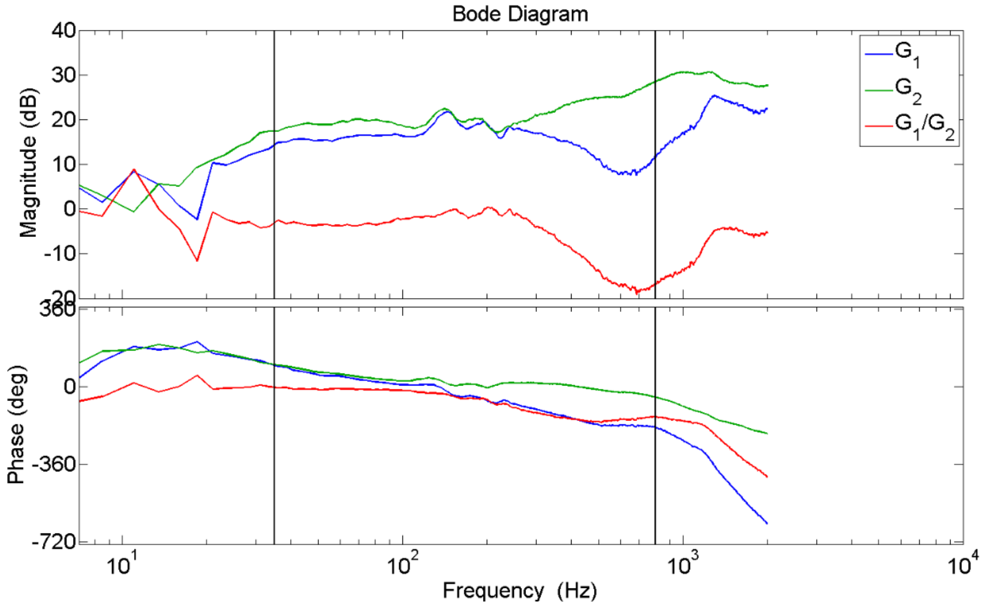

Figure 7: Frequency response G_1 and G_2

See the slightly higher magnitude of G_2 compared to G_1, introduced by the difference in distance between the control speaker and microphones. Again, the black lines indicate the useful frequency range $35 - 800\ Hz$. In practice we can say that the microphone outputs will never contain frequencies out of this frequency range introduced by the acoustic paths H_1 and H_2 and damped by the damping part of the AC air duct. Except for unknown high frequency measurement noise the acoustic paths G_1 and G_2 are only exposed to frequencies within the useful frequency range. Since microphone output 2 is directly coupled to the input $u(t)$, by a given controller C, the useful frequency range of the controller is also equal to the frequency range obtained for H_1 and H_2.

PHASE SHIFT

By looking at the phase diagram for G_1 and G_2, a simple approximation of the distance between microphone 1 and microphone 2 can be obtained. The distance in practice is given by $l_1 - l_2 = 0.2\ m$, see Table 1. At the left boundary ($35\ Hz$) the phase of both acoustic paths is the same and equals approximately $90°$. The phases at the right boundary ($2000\ Hz$) are given in Table 2.

Acoustic path	Phase $f = 35\ Hz$	Phase $f = 2000\ Hz$	Delta Phase
G_1	$90°$	$-636°$	$726°$
G_2	$90°$	$-220°$	$310°$

Table 2: Phase differences

The phase difference between both paths is $726 - 310 = 416°$ at $2000\ Hz$. Corresponding to a frequency of $1731\ Hz$ for a phase difference of $360°$. By assuming a perfect sinusoidal sound wave traveling with the speed of sound, the length of this standing wave equals $343/1731 \approx 0.2\ m$. Exactly equal to the measured distance between both microphones. Of course this is an approximation for $2000\ Hz$ and the accuracy will change by applying this approach for different frequencies introduced by the unsmooth behavior of the phase lines. Nevertheless, the result gives a practical feeling of phase delay introduced by the speed of sound travelling over a certain distance.

MEASUREMENT F_1 AND F_2

In practice the input signal $e_2(t)$ of the controller will be filtered, idem for the output signal $u(t)$ of the controller shown in Figure 5. This will be done to reduce the influence of noise in the system at frequencies out of the useful frequency range discussed before. For now, only the filters itself and the corresponding frequency responses will be discussed. Further information about the filters will be given in chapter IV.

Filter F_1 is a self-made second order Butterworth filter with a cut-off frequency of $1125\ Hz$. For details about this self-made filter see appendix A.

Filter F_2 is a *Krohn-Hite model 3200 filter*, known as a fourth order Butterworth filter, with a cut-off frequency of 1000 Hz.

Both cut-off frequencies are chosen just above the right boundary of the useful frequency range of $800\ Hz$. Measurements of the filters in practice are given in Figure 8.

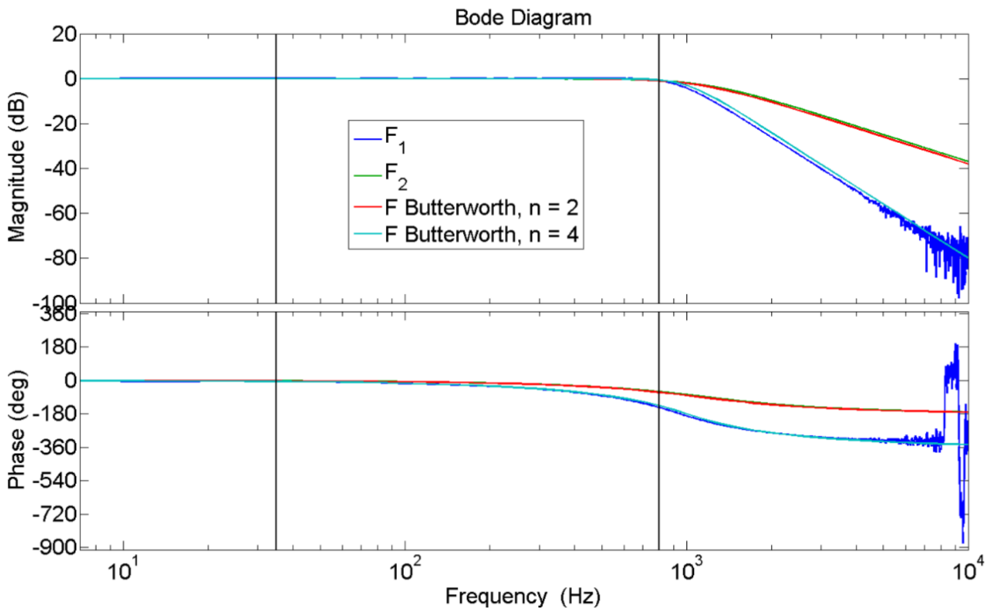

Figure 8: Frequency response F_1 and F_2

According to [22], a second- and fourth order Butterworth filter can be written as a transfer function.

$$F_{B,2nd} = \frac{\omega^2}{s^2 + \sqrt{2}\omega s + \omega^2} \quad \text{(II-4)} \qquad F_{B,4th} = \frac{\omega^4}{s^4 + 2.61\omega s^3 + 3.41\omega^2 s^2 + 2.61\omega^3 s + \omega^4} \quad \text{(II-5)}$$

Both transfer functions are added to the frequency response to see if this analytical representation is correct. As you can see in Figure 8, the analytical representation is really good and both transfer functions will be used as the nominal models of F_1 and F_2 further on.

III NOMINAL MODELS

All six frequency response models, from chapter II, will be identified to create corresponding nominal models of the acoustic paths. These nominal models will be used to design a controller and simulate Set up 2 in SIMULINK.

Two nominal models are already known and given by the Butterworth filter representations in chapter II. Both models fit well, see Figure 8. In text an additional 'roof' symbol will indicate the nominal model of a measured model in practice. In figures and illustrations an additional 'nominal' or 'measurement' term is added to distinguish between nominal and measured models.

$$\widehat{F}_1 = F_{B,2nd} \quad \text{(III-1)} \qquad \widehat{F}_2 = F_{B,4th} \quad \text{(III-2)}$$

Frequency response identifications of the four remaining models is provided with the MATLAB command *invfreqs* and the results are shown in Figure 9 - Figure 12. All models are stable models, as required, and the fitted model order is given in Table 3. By using the *invfreqs* command only high order nominal models turn out to be stable and a good fit, but the complexity of the frequency response is not always that high. To reduce the order of the nominal models and keep the stability and complexity of the frequency response as good as possible, a Hankel order reduction will be used, according to Ref. [5]. These Hankel reduced nominal models will be used as the best nominal representation of the measured models in practice and are also shown in the figures by a red line.

Model	**Order *invfreqs***	**Order Hankel**
$\widehat{H}_1$	25	16
$\widehat{H}_2$	24	16
$\widehat{G}_1$	22	9
$\widehat{G}_2$	21	9

Table 3: Nominal model orders

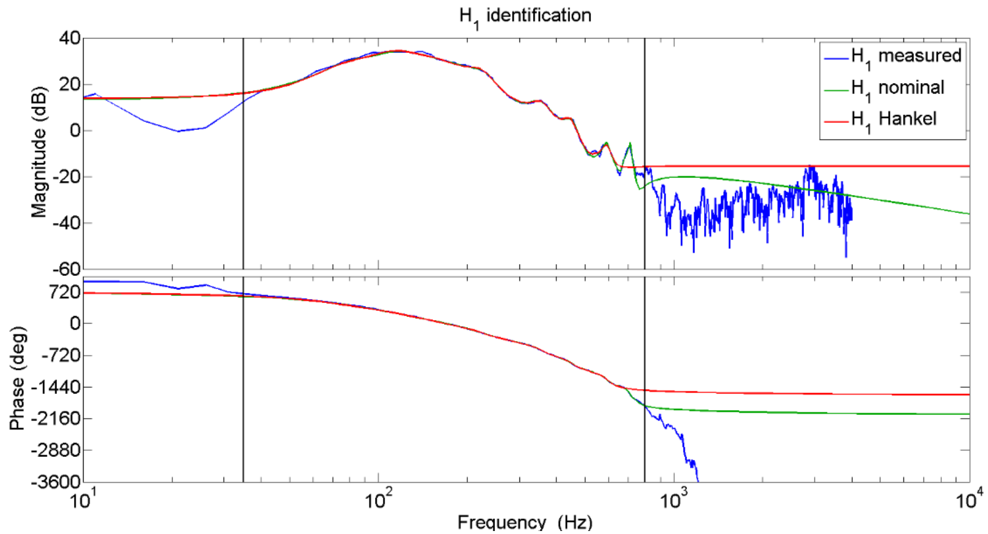

Figure 9: Identification H_1

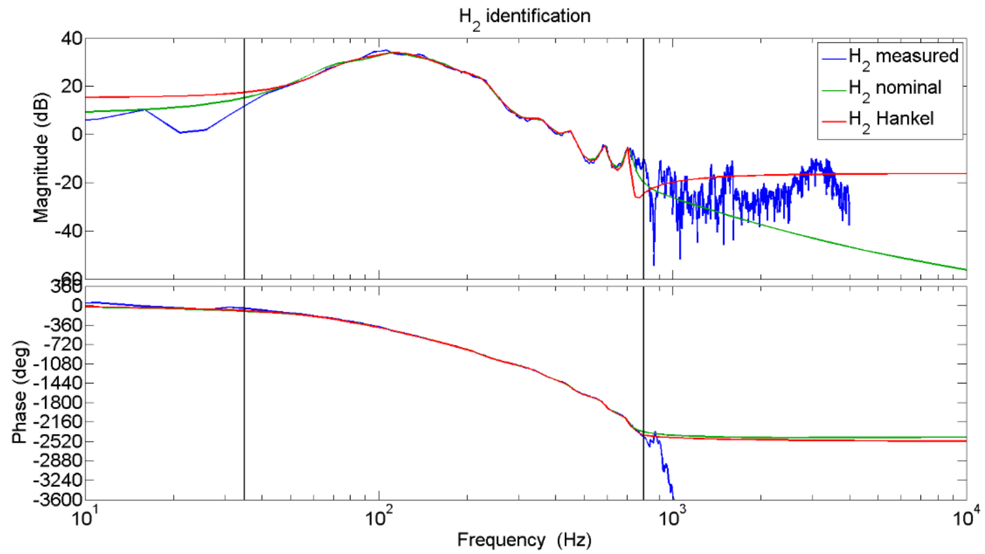

Figure 10: Identification H_2

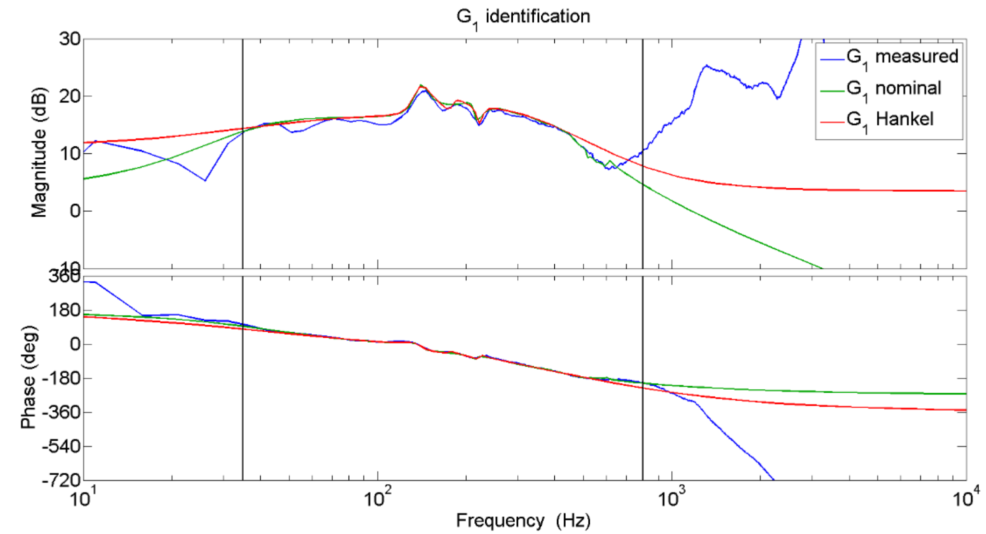

Figure 11: Identification G_1

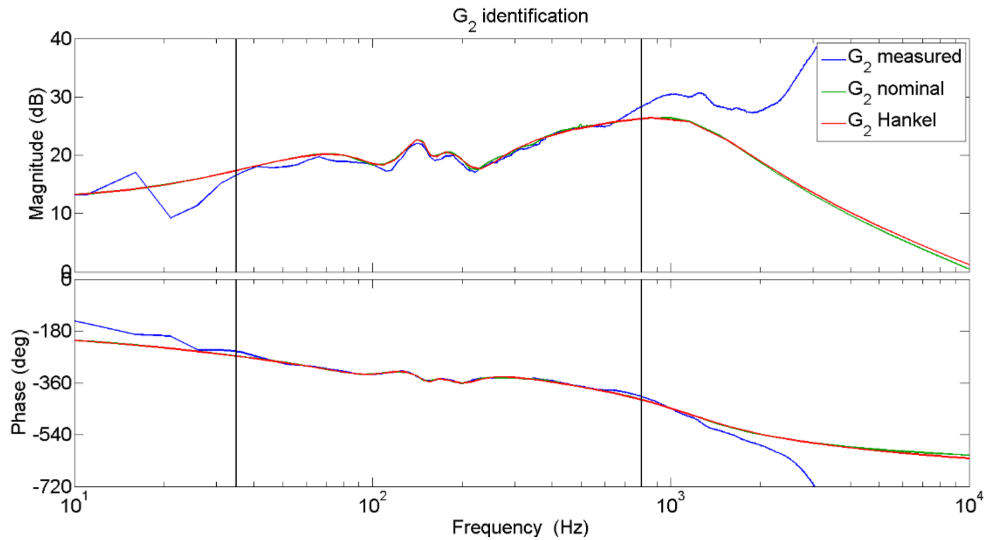

Figure 12: Identification G_2

IV PRACTICE

To design a stabilizing controller C, as shown in Set up 2 Figure 5, a practical set up will be created representing the situation in practice as good as possible. A schematic overview of the situation in practice is given in Figure 13 by adding both filters F_1 and F_2 and a computer block to the already known Set up 2 of Figure 5. Filter F_2 will be used to get rid of high frequency disturbances in the input of the controller and filter F_1 will cut-off the control signal of the control speaker. In practice the zero-order-hold sampling of the computer will introduce a sound in the control signal equal to the sample frequency. Filter F_1 is placed after the computer's DAC conversion to cancel this sample frequency in the control signal. Because of discrete time conversion by the computer (ADC), the designable controller must be a discrete time model as well, say C_D.

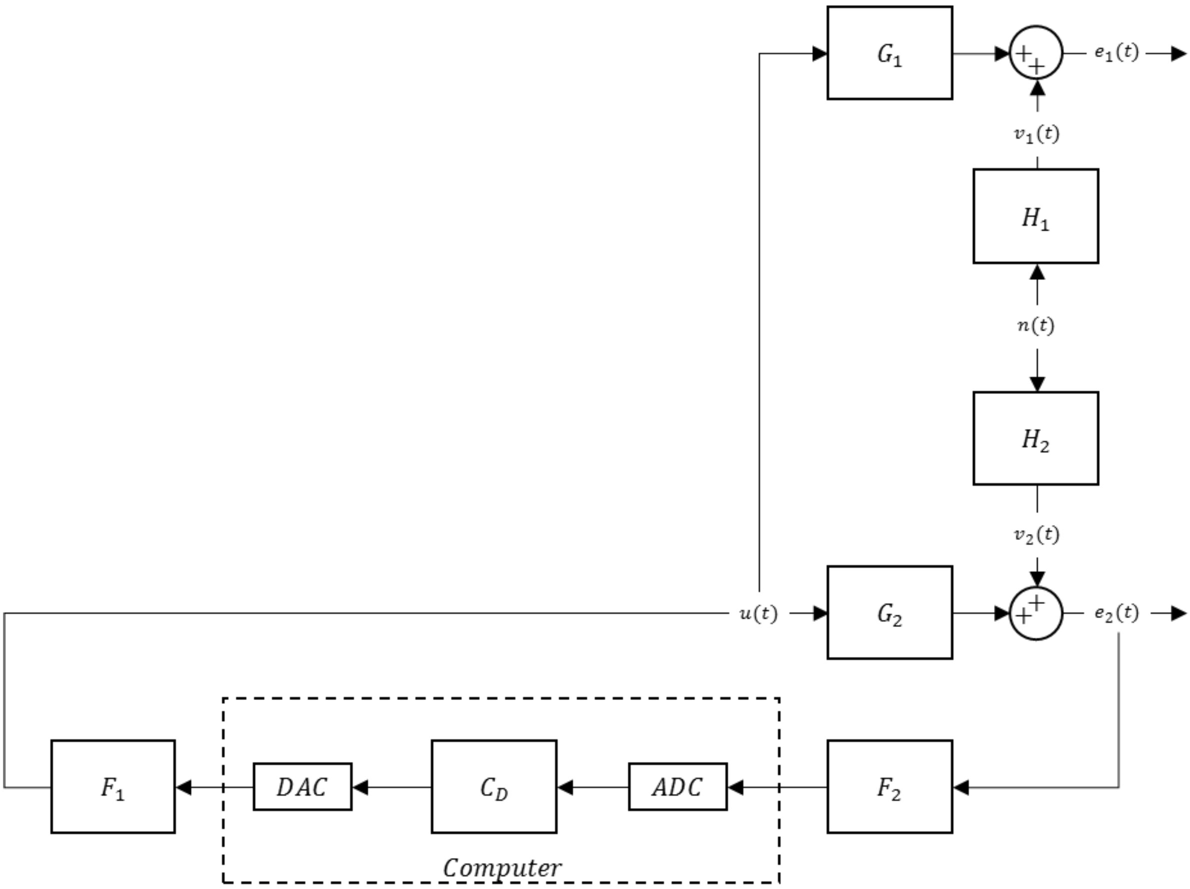

Figure 13: Practical set up

Obtain equations for $C_i, u(t), e_1(t)$ and $e_2(t)$ using the practical set up in Figure 13.

$$C_i = DAC \cdot C_D \cdot ADC \qquad \text{(IV-1)}$$

$$u(t) = F_2 C_i F_1 e_2(t) \qquad \text{(IV-2)}$$

$$e_1(t) = \left(H_1 + H_2 \frac{G_1 F_2 C_i F_1}{1 - G_2 F_2 C_i F_1}\right) n(t) \qquad \text{(IV-3)} \qquad e_2(t) = \frac{H_2}{1 - G_2 F_2 C_i F_1} n(t) \qquad \text{(IV-4)}$$

Filters

Both real-time filters are already discussed in chapter II and nominal models, using the Butterworth representations, are given by $\hat{F}_1$ and $\hat{F}_2$. By looking at Figure 8 the magnitude of both filters is unity in the useful frequency range so the filters will not influence the amplitude of the input and output signals for frequencies $< 1000\ Hz$. On the other side, both filters do introduce a small phase shift, especially in the higher frequency part of the useful frequency range. This will influence the stability of the system with the new closed loop gain in practice according to $L = -G_2 F_2 C_i F_1$, see (IV-4) . The stability criteria will stay the same but the expression changed and will include more delay at higher frequencies introduced by F_1 and F_2. This will result into a smaller effective range of the controller and worse performance of the system but both filters are necessary to get rid of unwanted high frequencies.

Sample frequency

The computer in practice must be able to measure the useful frequency range $35 - 800\ Hz$. To make sure that a frequency of $800\ Hz$ will be picked up by the computer a sample frequency of 5 times the right boundary of the useful frequency range will be used.

$$F_s = 5 * 800 = 4000\ Hz \tag{IV-5}$$

$$T_s = \frac{1}{F_s} = 0.00025\ s \tag{IV-6}$$

Computer

The computer block in Figure 13 represents the connection of input microphone 2 and output control speaker signal $u(t)$. For this connection a *National Instruments SCB-68 desktop connector block* [10] is used. Both ADC and DAC conversions of this connector block can be done within $0.01\ ms$ corresponding to a conversion frequency of $f_c = 100\ kHz$. For now the conversion time will be neglected and the ADC and DAC conversions will assumed to be small in comparison to the calculation time of the controller for each sample. To make sure that the controller output is expectable the discrete controller C_D cannot have a feedthrough term. In other words, the D-matrix of the state space representation of C_D equals zero. By using this restriction the whole ADC, DAC and discrete time conversion of the controller, say the computer block, can be replaced by a zero-order-hold in continuous time with a continuous controller C according to.

$$C_i = zoh * C \tag{IV-7}$$

NOMINAL SET UP

Using a continuous representation of the discrete computer block, the practical set up can be written as a continuous set up with only continuous nominal models, see Figure 14.

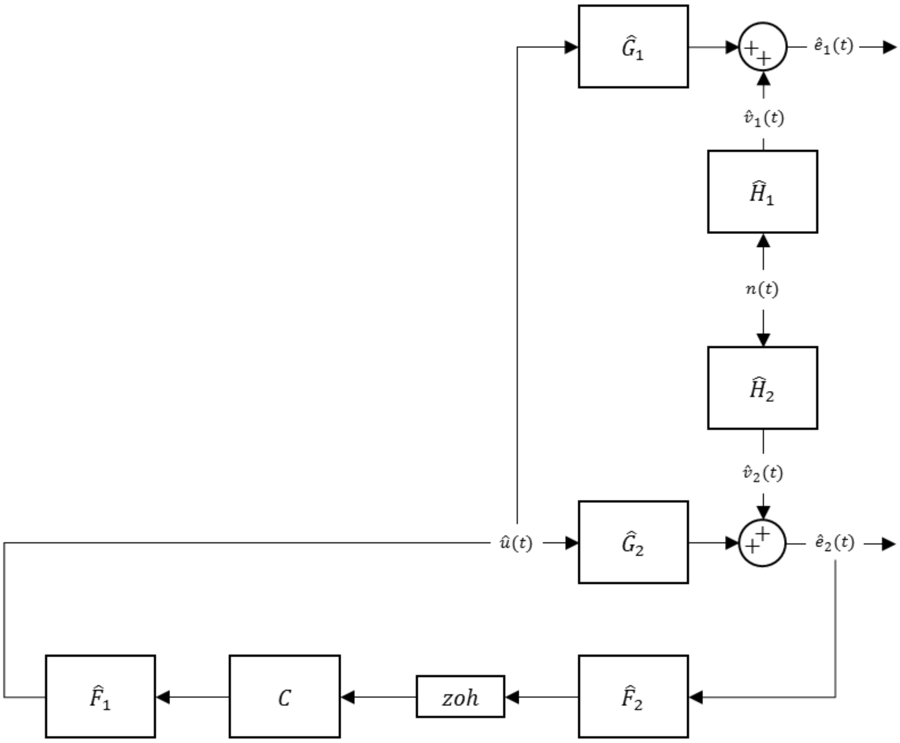

Figure 14: Nominal set up

The zero-order-hold term can be calculated using the sample time in (IV-6), according to.

$$zoh = \exp(-T_s \cdot s) \tag{IV-8}$$

Substitute (IV-7) into (IV-2) - (IV-4) to obtain the nominal equations for $u(t), e_1(t)$ and $e_2(t)$.

$$\hat{u}(t) = \hat{F}_2 \cdot zoh \cdot C\hat{F}_1 \hat{e}_2(t) \tag{IV-9}$$

$$\hat{e}_1(t) = \left(\hat{H}_1 + \hat{H}_2 \frac{\hat{G}_1 \hat{F}_2 \cdot zoh \cdot C\hat{F}_1}{1 - \hat{G}_2 \hat{F}_2 \cdot zoh \cdot C\hat{F}_1} \right) n(t) \tag{IV-10}$$

$$\hat{e}_2(t) = \frac{\hat{H}_2}{1 - \hat{G}_2 \hat{F}_2 \cdot zoh \cdot C\hat{F}_1} n(t) \tag{IV-11}$$

With $n(t)$ the unknown input signal. The nominal closed loop gain $\hat{L}$ and nominal sensitivity $\hat{S}$ are given by.

$$\hat{S} = \frac{1}{1 - \hat{G}_2 \hat{F}_2 \cdot zoh \cdot C\hat{F}_1} \tag{IV-12}$$

$$\hat{L} = -\hat{G}_2 \hat{F}_2 \cdot zoh \cdot C\hat{F}_1 \tag{IV-13}$$

With a nominal set up representing the situation in practice and all acoustic paths known as nominal models, a stable controller C can be designed according to the stability criteria in chapter I.

V LOOP SHAPED CONTROLLER

Loop shaping is the first method used to create a controller satisfying the stability criteria. To obtain noise reduction, the sensitivity function $\hat{S}$ must be as small as possible.

$$\hat{S} = \frac{1}{1+\hat{L}} \rightarrow small \tag{V-1}$$

Similar to an as large as possible closed loop gain.

$$\hat{L} \rightarrow large \tag{V-2}$$

At the same time, stability must be accomplished. Increasing the closed loop gain is possible but must fulfill the third stability criterion, known as.

$$\angle(\hat{L}) \neq \pm 180° \quad if \quad |\hat{L}| \geq 1. \tag{V-3}$$

To see how the nominal models and the zero-order-hold term in the nominal closed loop expression (IV-13) do already influence the stability criterion, a frequency response of the nominal closed loop gain is created assuming $C = 1$ and a sample frequency of $f_s = 1/T_s = 4000\ Hz$, see Figure 15.

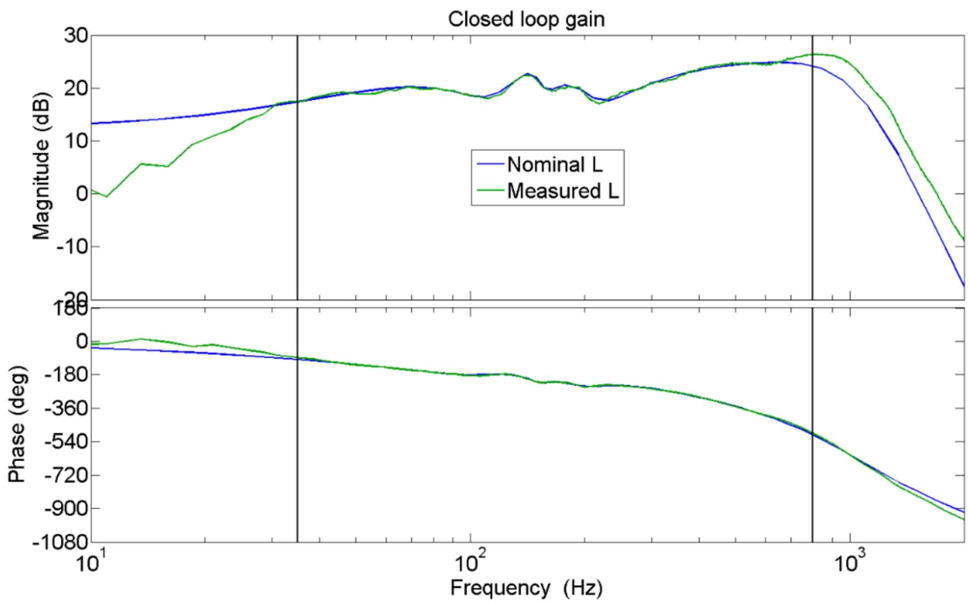

Figure 15: Closed loop gain, $C = 1$

$C = 1$ is definitely not satisfying the stability criteria and results in an unstable nominal sensitivity. The magnitude is larger than unity in the useful frequency range and passes a phase of $-180°$ in the middle of this range. In short, $C = 1$ does not satisfy the stability criteria.

A better options will be if $C = -1$, changing the phase of the nominal closed loop gain with $180°$, see Figure 16.

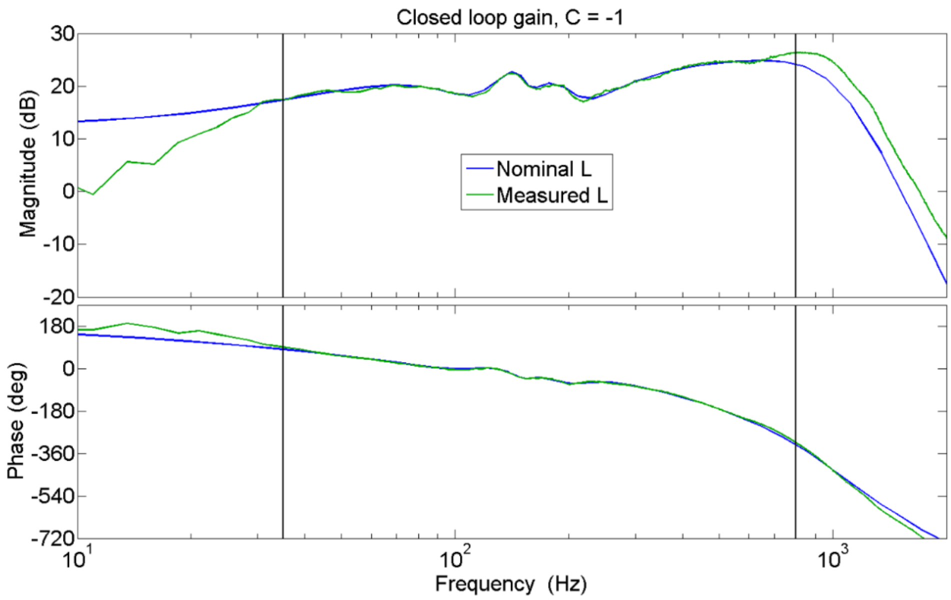

Figure 16: Closed loop gain, $C = -1$

Stable up to say $500\ Hz$. For low frequencies the stability margin is very small and for frequencies around $500\ Hz$ the nominal sensitivity function is unstable. What if a controller with a low gain in the low frequency region and a low gain in the frequency region around $500\ Hz$ is introduced? This will increase the phase margin for low frequencies and reduce the gain below unity for frequencies above $500\ Hz$. An example of such a controller is the following second order controller shown in (V-4). A zero is placed just before the useful frequency range, say $1\ Hz$, to introduce a slope of $+1$. Followed by two poles at the frequency were the phase in Figure 16 is approximately zero, say $130\ Hz$, to obtain a slope of -1. Ending with a slope of -1 introduces a zero feedthrough term and zero D-matrix in the state space representation. Which is a restriction for the controller if we use the nominal set up with a zero-order-hold term instead of a discrete computer block, all explained in chapter IV.

$$C = -k\frac{s + \omega_1}{s^2 + 2\zeta\omega_2 s + \omega_2^2} \tag{V-4}$$

$\omega_1 = 1\ Hz$ $\qquad$ $\omega_2 = 130\ Hz$ $\qquad$ $\zeta = 0.3$ $\qquad$ $k = 45$

A frequency response of the controller is shown in Figure 17 and the corresponding nominal closed loop gain in Figure 18.

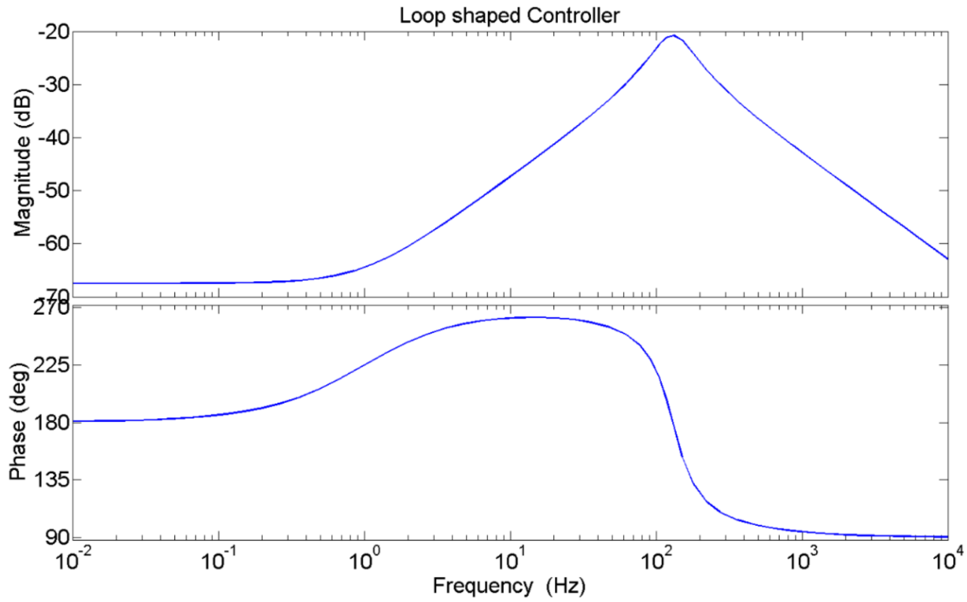

Figure 17: Loop shaped controller

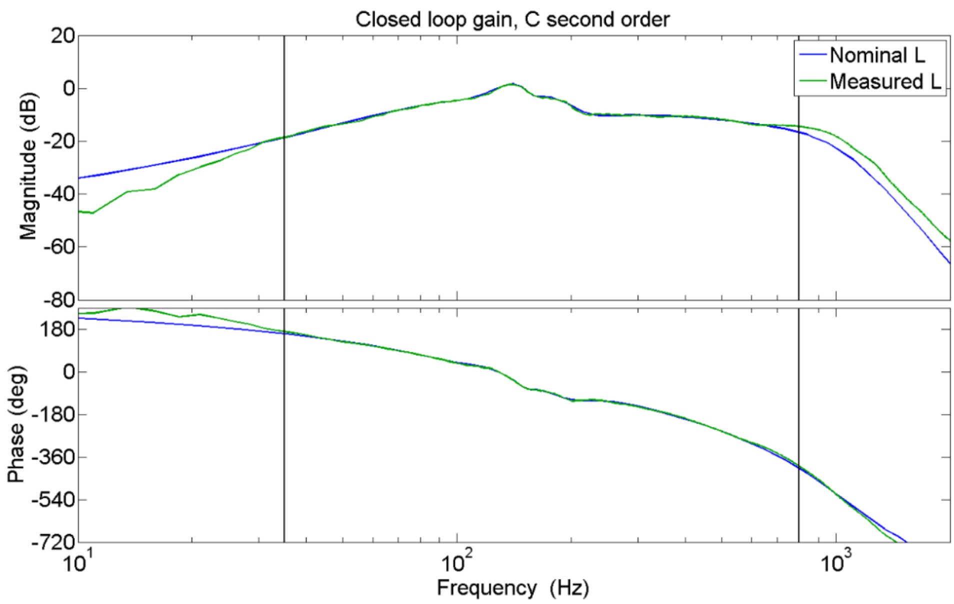

Figure 18: Closed loop gain, Loop shaped controller

The idea was to reduce the magnitude in the higher frequency part. As you can see the magnitude at higher frequencies is reduced and satisfies the stability criteria. Two intersections of phase $\pm 180°$ can be see, one at 30 Hz and a second one at approximately $360\ Hz$. For both intersections the stability criteria is satisfied because the corresponding magnitude is smaller than unity ($< 0\ dB$).

Practice

To see the performance in practice and the correctness of the nominal model $C * zoh$, the controller is discretized and used in the computer block as shown by C_D in Figure 13. Before the outputs $e_2(t)$ and $e_1(t)$ are measured, the correctness of $zoh * C$, as a model for the computer block $ADC * C_D * DAC$, will be checked. A measured model of the controller in practice and the nominal controller are shown in Figure 19.

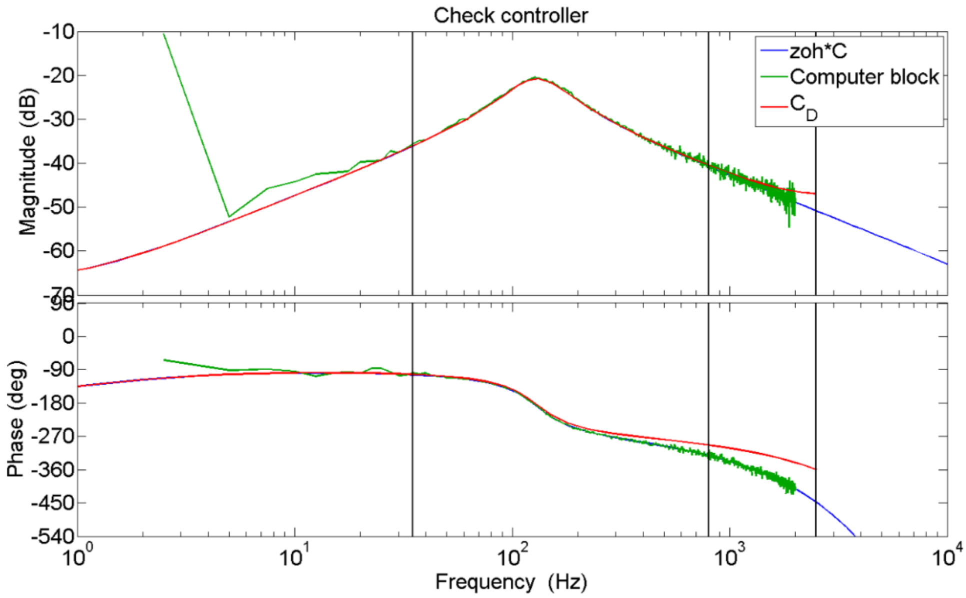

Figure 19: Check Controller

As expected the $zoh * C$ representation of the computer block $ADC * C_D * DAC$ is correct. Controller C_D is calculated with the MATLAB command $c2d(C, Ts, zoh)$ using a zero-order-hold discretization method. Applying the discretized controller on the real system in practice, the outputs $e_1(t)$ and $e_2(t)$ can be measured. A random noise signal with a frequency range of $0 \rightarrow 2\ kHz$ is applied at input $n(t)$ and the frequency response calculated by the *Control systems analyzer* is visible as the cyan lines in Figure 20 and Figure 21.

SIMULATION

A stabilizing controller (V-4) is designed by loop shaping the nominal closed loop gain. By simulating the outputs $e_1(t)$ and $e_2(t)$ using a random noise input $n(t)$, the performance of the controller can be seen. By looking at Figure 18 the best performance is expected around $130\ Hz$ were the phase is $0°$. The worst performance, or maybe even an increase in amplitude, is expected at the intersections with phase $\pm 180°$, known by the frequencies $30\ Hz$ and $360\ Hz$. The result is shown in Figure 20 and Figure 21 for output $e_1(t)$ and $e_2(t)$ respectively. How the nominal models are implemented in SIMULINK for a simulation will be discussed in chapter IX.

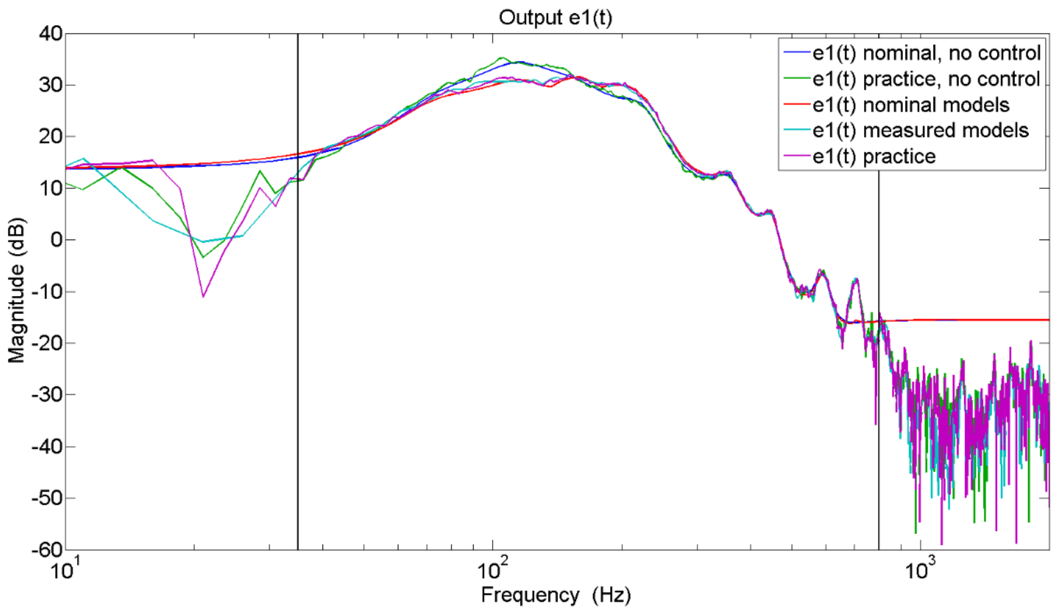

Figure 20: Output $e_1(t)$, Loop shaped controller

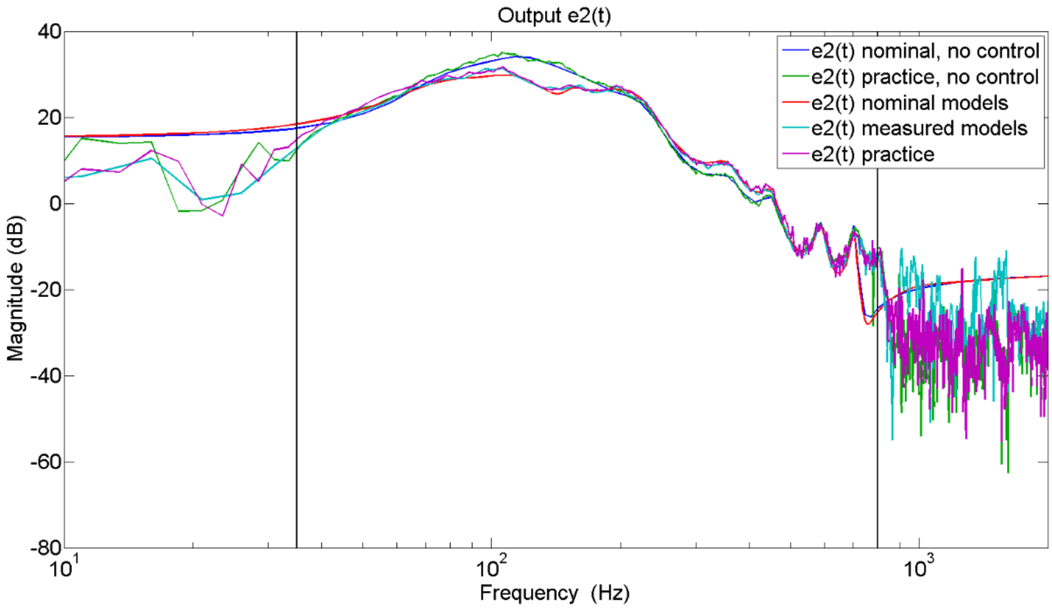

Figure 21: Output $e_2(t)$, Loop shaped controller

As you can see, the nominal setup including nominal models and a nominal controller (red line) or the light blue line, using the measured models instead of nominal models and the nominal controller, do show a good fit compared to the frequency response in practice (cyan line). Incoming sound is reduced in the frequency range $70 - 170\ Hz$ with a maximum magnitude ratio of 0.45 around $130\ Hz$ as expected. A worse performance and even an increase in magnitude as expected around $300\ Hz$ is correct as well. (control lines above no control lines in Figure 21).

In short, the amount of noise reduction in practice is similar to the nominal models used to create the loop shaped controller but the total amount of noise reduction is small. Using a loop shaped static second order controller will result into a stable feedback system with sound reduction but the amount is almost neglectable. In chapter VI an H_2 controller will be designed to see if such a controller can do a better job.

VI H_2 CONTROLLER

According to [4], [7] and [8], the H_2 control problem can be presented in the following way.

Introduce the partition of T according to.

$$\begin{bmatrix} \hat{e}_1(t) \\ \hat{e}_2(t) \end{bmatrix} = \begin{bmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{bmatrix} \begin{bmatrix} n(t) \\ \hat{u}(t) \end{bmatrix} \tag{VI-1}$$

The closed loop transfer function $F(T, C_H)$ is given as.

$$F(T, C_H) = T_{11} + T_{12}(I - C_H T_{22})^{-1} C_H T_{21} \tag{VI-2}$$

$$\hat{e}_1(t) = F(T, C_H) n(t) \tag{VI-3}$$

This H_2 control problem consists of finding a controller C_H which stabilizes the plant T and minimizes the following cost function.

$$J(C_H) = \|F(T, C_H)\|_2^2 \tag{VI-4}$$

With $\|F(T, C_H)\|_2$ the H_2-norm.

The problem is most conveniently solved in the time domain and assumed will be the state-space representation of plant T.

$$x(t) = Ax(t) + B_1 n(t) + B_2 u(t) \tag{VI-5}$$
$$\hat{e}_1(t) = C_1 x(t) + D_{11} n(t) + D_{12} u(t) \tag{VI-6}$$
$$\hat{e}_2(t) = C_2 x(t) + D_{21} n(t) + D_{22} u(t) \tag{VI-7}$$

In the state-space representation $D_{22} = 0$. The direct feedthrough from $u(t)$ to $\hat{e}_2(t)$ has assumed to be zero because physical systems always have a zero gain at infinite frequency. In our case this is not true because the nominal models $\hat{G}_1$ and $\hat{G}_2$, representing the physical system, do have a small feedthrough term as can be seen as a zero-slope in Figure 11 and Figure 12 for high frequencies.

According to Ref. [7] an optimal controller does not exist if D_{12} does not have full column rank or D_{21} does not have full row rank. In both situations solving the Ricatti equation, the method to obtain controller C_H by minimizing the cost function, is not possible because of control singularity or sensor singularity at infinite frequency, respectively $D_{12} = 0$ or $D_{21} = 0$.

IMPLEMENTATION

Implement the nominal set up as given in Figure 14, in which the closed loop transfer function is a representation of the transfer function between output $\hat{e}_1(t)$ and input $n(t)$ in (IV-10).

$$F(T, C_H) = T_{11} + T_{12}(I - C_H T_{22})^{-1} C_H T_{21} \tag{VI-8}$$

$$\hat{e}_1(t) = T_{11} n(t) + T_{12} \hat{u}(t) \tag{VI-9}$$
$$\hat{e}_2(t) = T_{21} n(t) + T_{22} \hat{u}(t) \tag{VI-10}$$
$$\hat{u}(t) = C_H \hat{e}_2(t) \tag{VI-11}$$

The following T-matrix can be assumed.

$$T_{11} = \hat{H}_1 \quad \text{(VI-12)} \qquad T_{12} = \hat{G}_1 \quad \text{(VI-13)}$$
$$T_{21} = \hat{H}_2 \quad \text{(VI-14)} \qquad T_{22} = \hat{G}_2 \quad \text{(VI-15)}$$

Rewrite (VI-9) and (VI-10) by substituting (VI-11) - (VI-15).

$$\hat{e}_1(t) = \left(\hat{H}_1 + \hat{H}_2 \frac{\hat{G}_1 C_H}{1 - \hat{G}_2 C_H}\right) n(t) \quad \text{(VI-16)} \qquad \hat{e}_2(t) = \frac{\hat{H}_2}{1 - \hat{G}_2 C_H} n(t) \quad \text{(VI-17)}$$

In which C_H will be the H_2 controller found by minimizing the cost function.

Problem 1:

The nominal expressions $\hat{u}(t)$ in the nominal set up is already given in (IV-9) and must be equal to (VI-11) above to make C_H usable as the representation of controller C in the nominal set up.

$$\hat{u}(t) = zoh \cdot C\hat{F}_1 \hat{e}_2(t) = C_H \hat{e}_2(t) \tag{VI-18}$$

(VI-18) implies that a C_H controller will correspond to the $zoh \cdot C\hat{F}_1$ term in the nominal model. So the C_H controller is not directly related to C and not implementable in the computer block.

Solution 1a:

The nominal controller can be calculated by dividing the H_2 controller C_H by the filter $\hat{F}_1$ and zero-order-hold term zoh.

$$zoh \cdot C\hat{F}_1 = C_H \rightarrow C = \frac{C_H}{zoh \cdot \hat{F}_1} \tag{VI-19}$$

In which $zoh^{-1} = \exp(T_s \cdot s)$ is a non-existing function, so the calculation will not be possible.

Solution 1b:

Implement the two filters and zero-order-hold term into the plant T instead of the assumptions for the T-matrix above in (VI-12) - (VI-15). In this way the closed loop transfer function will be equal to the nominal equation of $\hat{e}_1(t)$ in (IV-10), see (VI-20). If that is possible C_H can be discretized to obtain the C_D representation for the computer block in practice.

$$F(T,C_H) = T_{11} + T_{12}(I - C_H T_{22})^{-1} C_H T_{21} = \widehat{H}_1 + \widehat{H}_2 \frac{\widehat{G}_1 \widehat{F}_2 \cdot zoh \cdot C\widehat{F}_1}{1 - \widehat{G}_2 \widehat{F}_2 \cdot zoh \cdot C\widehat{F}_1} \qquad \text{(VI-20)}$$

Obtain the three unknown terms in (VI-20).

$T_{11} = \widehat{H}_1$ (VI-21) $T_{12}T_{21} = \widehat{H}_2 \widehat{G}_1 \widehat{F}_2 \cdot zoh \cdot \widehat{F}_1$ (VI-22) $T_{22} = \widehat{G}_2 \widehat{F}_2 \cdot zoh \cdot \widehat{F}_1$ (VI-23)

Check the output equations:

$$\hat{u}(t) = C_H \hat{e}_2(t) \qquad \text{(VI-24)}$$

$$\hat{e}_1(t) = T_{11} n(t) + T_{12} \hat{u}(t) = \widehat{H}_1 + T_{12} C_H \hat{e}_2(t) \qquad \text{(VI-25)}$$

$$\hat{e}_2(t) = T_{21} n(t) + \widehat{G}_2 \widehat{F}_2 \cdot zoh \cdot \widehat{F}_1 C_H \hat{e}_2(t) \rightarrow \hat{e}_2(t) = \frac{T_{21}}{1 - \widehat{G}_2 \widehat{F}_2 \cdot zoh \cdot \widehat{F}_1 C_H} n(t) \qquad \text{(VI-26)}$$

By looking at the nominal equation for $\hat{e}_2(t)$ in (IV-11) we can see that $T_{21} = \widehat{H}_2$.

Substitute into (VI-22) to obtain T_{12}.

$$T_{12} \widehat{H}_2 = \widehat{H}_2 \widehat{G}_1 \widehat{F}_2 \cdot zoh \cdot \widehat{F}_1 \rightarrow T_{12} = \widehat{G}_1 \widehat{F}_2 \cdot zoh \cdot \widehat{F}_1 \qquad \text{(VI-27)}$$

Substitute (VI-26) and (VI-27) into (VI-25).

$$\hat{e}_1(t) = \left(T_{11} + T_{21} \frac{T_{12} C_H}{1 - T_{22}}\right) n(t) = \widehat{H}_1 n(t) + \widehat{H}_2 \frac{\widehat{G}_1 \widehat{F}_2 \cdot zoh \cdot \widehat{F}_1 C_H}{1 - \widehat{G}_2 \widehat{F}_2 \cdot zoh \cdot \widehat{F}_1 C_H} n(t) \qquad \text{(VI-28)}$$

Which is identical to the $\hat{e}_1(t)$ in (IV-10) using the following components of plant T.

$T_{11} = \widehat{H}_1$ (VI-29) $T_{12} = \widehat{G}_1 \widehat{F}_2 \cdot zoh \cdot \widehat{F}_1$ (VI-30)

$T_{21} = \widehat{H}_2$ (VI-31) $T_{22} = \widehat{G}_2 \widehat{F}_2 \cdot zoh \cdot \widehat{F}_1$ (VI-32)

The nominal set up visualized in Figure 14 can now be shown with the partition of T, see Figure 22 and by discretizing C_H the controller can be implemented in the computer block.

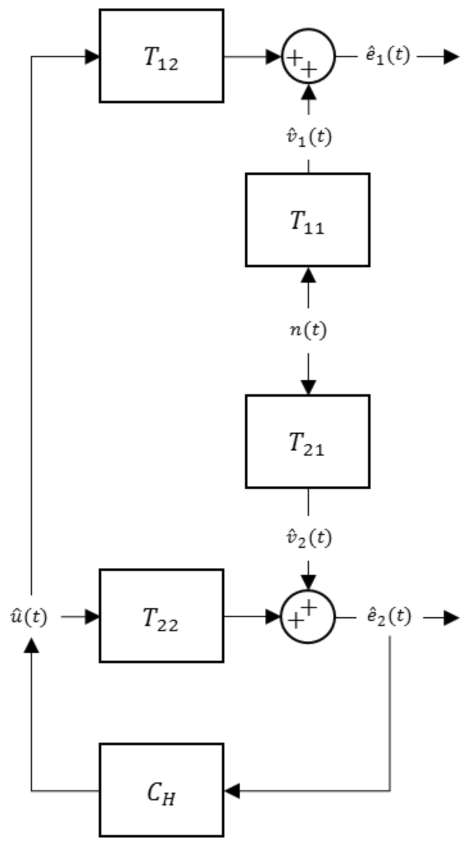

Figure 22: Nominal set up, partition *T*

Problem 2:
By substituting the filters $\hat{F}_2$ and $\hat{F}_1$ into T_{12} the feedthrough term of T_{12} becomes zero. Introduced by the fact that the feedthrough term of a Butterworth filter is always zero (ending negative slope). This results into $D_{12} = 0$ for plant T, making it impossible to solve the Ricatti equation as well as finding a controller C_H.

Solution 2:
By adding n-zeros to the filters F_1 and F_2 at a very high frequency ω_{high}, the ending negative slope can be flattened for very high frequencies. This results into a non-zero feedthrough term for $\omega \to \infty$ and will get rid of the control singularity problem. Both filters will be replaced by the following equation with n the order of the Butterworth filter and ω_{high} a selectable high frequency, say 10 times the cut-off frequency of the filters.

$$F_{new} = \frac{1}{\omega_{high}^{n}} F\left(s + \omega_{high}\right)^{n} \tag{VI-33}$$

CALCULATE C_H

Now the singularity issues are solved and the C_H controller can be discretized and implemented directly in the computer block, the C_H controller can be calculated solving the Ricatti equations with the MATLAB function *h2syn(T,1,1).*

The result of this calculation is shown in Figure 23. C_H stabilizes the nominal sensitivity function $\hat{S}$ but is unstable itself. Since the first stability criterion demands a stable closed loop gain, the controller itself must be stable too.

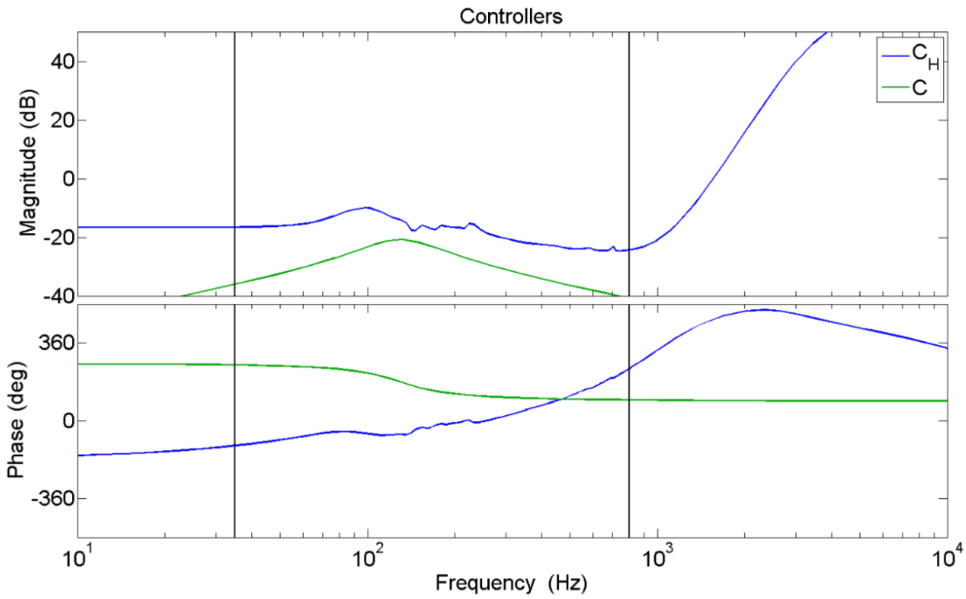

Figure 23: H_2 controller

A second figure will show the frequency response of the closed loop transfer function also known by the output $\hat{e}_1(t)$, see Figure 24.

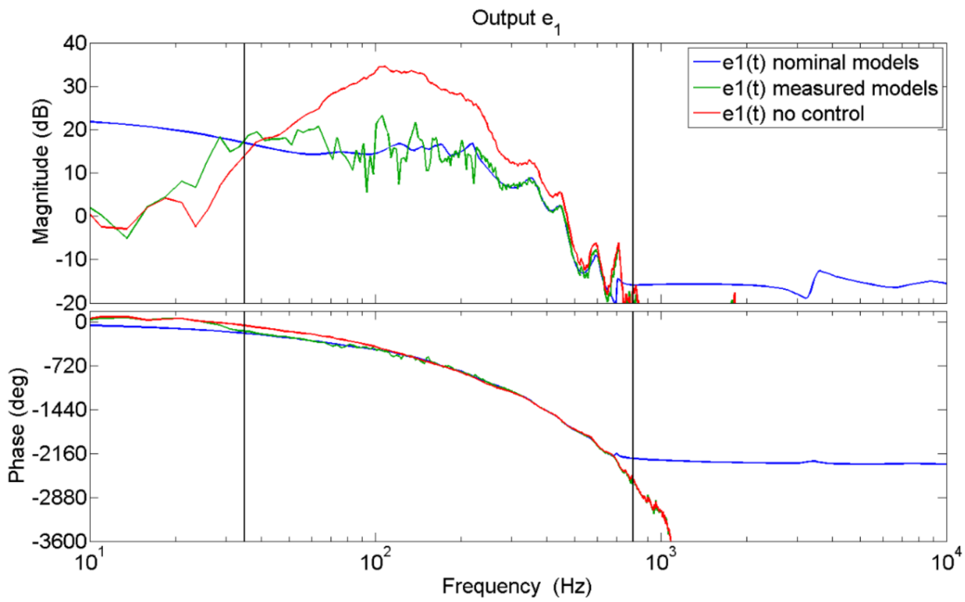

Figure 24: Output $e_1(t)$, H_2 controller

A nice result but not usable in practice because the controller itself is unstable. You can see the very nice behavior of the H_2 controller in the phase diagram of Figure 23. Phase increases by increasing the frequency as an opposite response of acoustic path G_2, see Figure 12. That way, the closed loop gain (product of both) will have a constant phase and does not reach $\pm 180°$ anymore. See Figure 25 for the perfectly flattened phase of the closed loop gain in the useful frequency range.

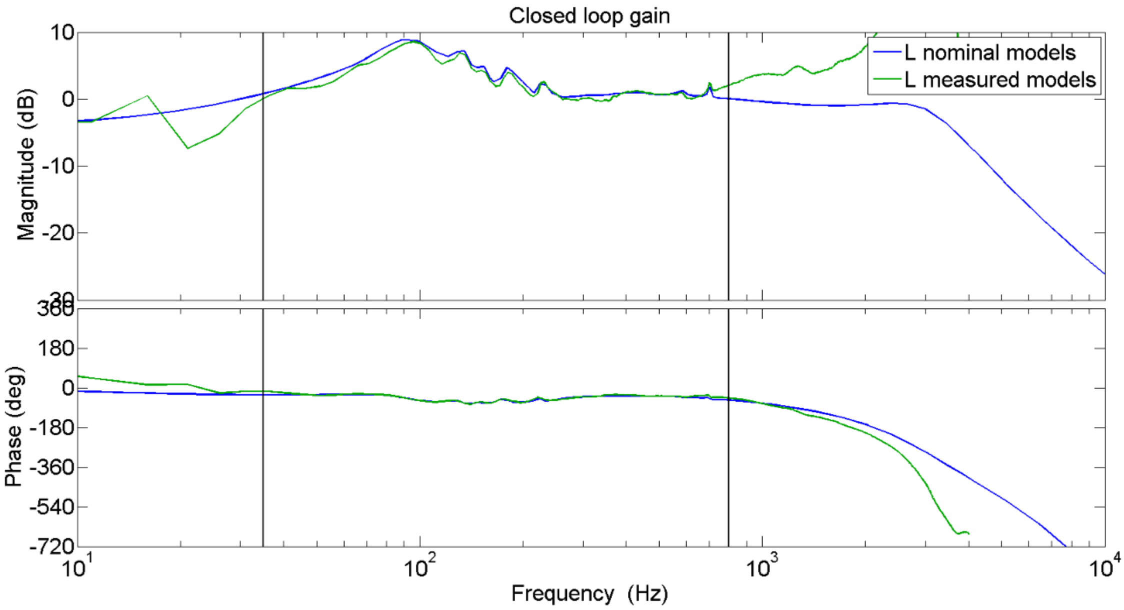

Figure 25: Closed loop gain, H_2 controller

VII YOULA CONTROLLER SET UP 2

Another way of improving the sound reduction is using a real-time updating controller. By shaping the controller, based on the characteristics of the incoming noise $n(t)$, the sound can be reduced even more. Again, the controller must satisfy the three stability criteria and according to the first stability criterion the controller must always be stable $C \in \mathcal{RH}_\infty$. This can be achieved by using a Youla parameterization according to Ref. [15].

YOULA PARAMETERIZATION

According to chapter VI the closed loop transfer function of the nominal set up can be written as.

$$\begin{bmatrix} \hat{e}_1(t) \\ \hat{e}_2(t) \end{bmatrix} = \begin{bmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{bmatrix} \begin{bmatrix} n(t) \\ \hat{u}(t) \end{bmatrix} \tag{VII-1}$$

With the T-matrix given by.

$$T_{11} = \hat{H}_1 \tag{VII-2}$$
$$T_{12} = \hat{G}_1 \hat{F}_2 \cdot zoh \cdot \hat{F}_1 \tag{VII-3}$$
$$T_{21} = \hat{H}_2 \tag{VII-4}$$
$$T_{22} = \hat{G}_2 \hat{F}_2 \cdot zoh \cdot \hat{F}_1 \tag{VII-5}$$

Consider the feedback system for nominal output $\hat{e}_2(t)$ and neglect output $\hat{e}_1(t)$ for now.

$$\hat{e}_2(t) = T_{22}\hat{u}(t) + T_{21}n(t) \tag{VII-6}$$
$$\hat{u}(t) = C\hat{e}_2(t) \tag{VII-7}$$

Assume $C = -C_i$ to obtain a negative feedback loop, which will be easier for writing down the Youla parameterization further on.

$$\hat{u}(t) = -C_i\hat{e}_2(t) \tag{VII-8}$$
$$\hat{e}_2(t) = \frac{T_{21}}{1 + T_{22}C_i} n(t) \tag{VII-9}$$

YOULA UPDATING CONTROLLER

Consider the feedback connection of a nominal model T_{22} and initial controller C_i stabilizing the closed loop transfer function $P(C_i, T_{22}) \in \mathcal{RH}_\infty$.

$$P(C_i, T_{22}) = \begin{bmatrix} C_i \\ T_{22} \end{bmatrix} (I + T_{22}C_i)^{-1} [T_{22} \quad I] \tag{VII-10}$$

All $C_Q = N_Q D_Q^{-1}$ that satisfy $P\left(C_Q, T_{22}\right) \in \mathcal{RH}_\infty$ are given by the right-coprime-factorization (rcf) and Youla parameter Q according to.

$$N_Q = N_i + DQ \tag{VII-11}$$
$$D_Q = D_i - NQ \tag{VII-12}$$

$$Q \in \mathcal{RH}_\infty \tag{VII-13}$$

With the rcf $T_{22} = ND^{-1}$ and $C_i = N_i D_i^{-1}$ according to Ref. [3]. In which the left-coprime-factorization (lcf) is given as $T_{22} = \widetilde{D}^{-1}\widetilde{N}$ and $C_i = \widetilde{D}_i^{-1}\widetilde{N}_i$. When Q is stable, stability can be obtained for the feedback of C_Q and T_{22}. Using coprime factorization to formulate stability of a feedback connection of C_i and T_{22} as $P(C_i, T_{22}) \in \mathcal{RH}_\infty$ for internal stability, is equivalent to $\Lambda^{-1} \in \mathcal{RH}_\infty$.

$$\Lambda = \widetilde{D}D_i + \widetilde{N}N_i \qquad \text{(VII-14)}$$

The Youla Controller can be obtained from the rcf $C_Q = N_Q D_Q^{-1}$.

$$C_Q = (N_i + DQ)(D_i - NQ)^{-1} \qquad \text{(VII-15)}$$

TRIVIAL CHOICE

If nominal plant T_{22} and stabilizing controller C_i are stable, a trivial choice for the rcf representations will be.

$$\widetilde{D} = D = I \qquad \widetilde{D}_i = D_i = I \qquad \widetilde{N} = N = T_{22} \qquad \widetilde{N}_i = N_i = C_i$$

Substitute into (VII-14) and (VII-15).

$$C_Q = (C_i + Q)(I - T_{22}Q)^{-1} \qquad \text{(VII-16)} \qquad \Lambda = I + T_{22}C_i \qquad \text{(VII-17)}$$

REPRESENTATION IN SET UP 2

A schematic representation of the Youla Controller is given in Figure 26 and implemented in Set up 2 as shown in Figure 27.

$$\hat{e}_2(t) = T_{22}\hat{u}(t) + T_{21}n(t) \qquad \text{(VII-18)}$$
$$\hat{u}(t) = -C_Q\hat{e}_2(t) \qquad \text{(VII-19)}$$

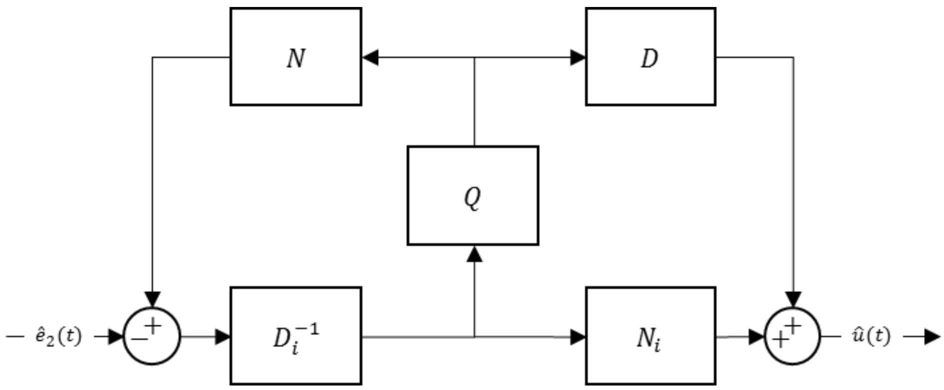

Figure 26: Youla Controller

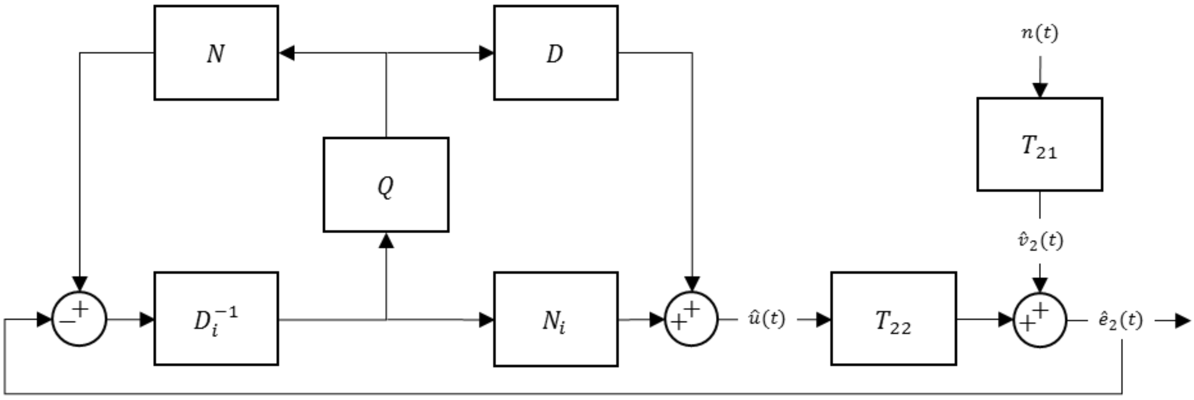

Figure 27: Youla parameterization Set up 2

Our goal is to minimize the output $\hat{e}_2(t)$ according to Set up 2 and stabilize the nominal sensitivity satisfying the stabilizing criteria. To do so, the approach of Set up 2 minimizes the following weighted two-norm performance measure.

$$\left\| \begin{matrix} \gamma W \hat{u}(t) \\ \hat{e}_2(t) \end{matrix} \right\|_2 \tag{VII-20}$$

Where W is a user-specified monic stable filter to satisfy the stability criteria, and γ is an additional scalar weighting allowed because of the monicity of W.

The variance of $\hat{e}_2(t)$ and $\hat{u}(t)$ is driven by the input $n(t)$ and nominal model T_{21} in which $n(t)$ is an unknown noise signal, see (VII-18). The Youla parameterization allows the weighted two-norm to be written as a function of the stable Youla parameter Q.

LINEAR IN Q

To write the two-norm as a function of Q the Youla controller in (VII-15) is substituted into both terms of the performance measure in (VII-20).

$$\gamma W \hat{u}(t) = -\gamma W (N_i + DQ)(D_i - NQ)^{-1} \hat{e}_2(t) \tag{VII-21}$$

$$\hat{e}_2(t) = \frac{T_{21}}{1 + ND^{-1}(N_i + DQ)(D_i - NQ)^{-1}} n(t) \tag{VII-22}$$

Rewrite (VII-22).

$$\hat{e}_2(t) = T_{21} D_i^{-1} \frac{D_i - NQ}{I + ND^{-1} N_i D_i^{-1}} n(t) \tag{VII-23}$$

Substitute (VII-23) into (VII-21).

$$\gamma W \hat{u}(t) = -\gamma W T_{21} D_i^{-1} \left(I + ND^{-1} N_i D_i^{-1}\right)^{-1} (N_i + DQ) n(t) \tag{VII-24}$$

(VII-23) and (VII-24) can be written as a vector summation linear in Q.

$$\begin{bmatrix} \gamma W\hat{u}(t) \\ \hat{e}_2(t) \end{bmatrix} = \begin{bmatrix} -\gamma W N_i \\ D_i \end{bmatrix} w(t) - \begin{bmatrix} \gamma W D \\ N \end{bmatrix} Q w(t) \tag{VII-25}$$

With $w(t)$ according to.

$$w(t) = T_{21} D_i^{-1} (I + T_{22} C_i)^{-1} n(t) \tag{VII-26}$$

$T_{21} n(t)$ can be obtained from the signals $-\hat{e}_2(t)$ and $\hat{u}(t)$. By using coprime factorizations and a stabilizing controller C_i, the signal $w(t)$ is more general than an output or equation error observer of the nominal disturbance signal $\hat{v}_2(t)$. Obtain $T_{21} n(t)$ from (VII-6) by rewriting the equation.

$$\hat{e}_2(t) = T_{22}\hat{u}(t) + T_{21} n(t) \rightarrow T_{21} n(t) = \hat{e}_2(t) - T_{22}\hat{u}(t) \tag{VII-27}$$

Can be written as a vector multiplication.

$$T_{21} n(t) = [T_{22} \quad I] \begin{bmatrix} -\hat{u}(t) \\ \hat{e}_2(t) \end{bmatrix} \tag{VII-28}$$

Substitute (VII-28) into (VII-26).

$$w(t) = D_i^{-1} (I + T_{22} C_i)^{-1} [T_{22} \quad I] \begin{bmatrix} -\hat{u}(t) \\ \hat{e}_2(t) \end{bmatrix} = \Lambda^{-1} [\widetilde{N} \quad \widetilde{D}] \begin{bmatrix} -\hat{u}(t) \\ \hat{e}_2(t) \end{bmatrix} \tag{VII-29}$$

Based on this analysis, the filtered closed loop signal $w(t)$ can be defined with $\Lambda^{-1} \in \mathcal{RH}_\infty$, $P(C_i, T_{22}) \in \mathcal{RH}_\infty$ and $T_{22} \in \mathcal{RH}_\infty$. Proof for (VII-29) is given in appendix B.

By allowing a parameterization of $Q(\theta)$, the error signal will be linear in the parameter θ, if and only if $Q(\theta) \in \mathcal{RH}_\infty$ is parameterized linearly in the parameter θ.

$$\left\| \begin{matrix} \gamma W\hat{u}(t) \\ \hat{e}_2(t) \end{matrix} \right\|_2 = \lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} \varepsilon(t,\theta)^T \varepsilon(t,\theta) \tag{VII-30}$$

$$\varepsilon(t,\theta) = \begin{bmatrix} -\gamma W N_i \\ D_i \end{bmatrix} w(t) - \begin{bmatrix} \gamma W D \\ N \end{bmatrix} Q(\theta) w(t) \tag{VII-31}$$

FIR MODEL

An obvious choice for $Q(\theta)$, according to Ref. [11], is an FIR model of order n. Given the FIR property of inherent stability, with all poles located at the origin, will introduce an always stable model for Q.

$$Q(q,\theta) = \theta_0 + \sum_{k=1}^{n} \theta_k q^{-k} \tag{VII-32}$$

Remark: For a discrete time calculation, especially when the calculation time of the Youla Controller is almost equal to the sample time, the feedthrough term θ_0 is not possible and must equal zero.

UPDATING

The error signal optimization is based on all time samples from $N = 1 \rightarrow \infty$ in which updating the controller is only possible after a finite measurement. To anticipate changes in the frequency spectrum of $n(t)$, the error signal optimization is computed over a finite number of time samples.

$$\hat{\theta}_t = \min_{\theta} \frac{1}{t} \sum_{\tau=0}^{t} \varepsilon(\tau, \theta)^T \varepsilon(\tau, \theta) \tag{VII-33}$$

This finite time computation is used to formulate a Recursive Least Squares (RLS) solution as in [13].

For a SISO system the expression for the error signal in (VII-31) can be simplified.

$$\varepsilon(t, \theta) = y_f(t) - Q(\theta) u_f(t) \tag{VII-34}$$

Where $y_f(t)$ denotes the filtered output and $u_f(t)$ the filtered input signal.

$$y_f(t) = \begin{bmatrix} -\gamma W N_i \\ D_i \end{bmatrix} w(t) \quad \text{(VII-35)} \qquad u_f(t) = \begin{bmatrix} \gamma W D \\ N \end{bmatrix} w(t) \quad \text{(VII-36)}$$

For a linearly parameterized scalar $Q(q, \theta)$, the error signal $\varepsilon(t, \theta)$ can be written in a linear regression form with a regressor $\phi(t)$. Where the regressor contains past versions of the input signal $u_f(t)$.

$$\varepsilon(t, \hat{\theta}_t) = y_f(t) - \phi(t)^T \hat{\theta}_t \tag{VII-37}$$

$$\phi(t)^T = [u_f(t) \quad u_f(t-1) \quad \dots \quad u_f(t-n)] \tag{VII-38}$$

A standard RLS update algorithm in [23] and [24] can be summarized by three iterative steps.

1) Prediction error update

$$\varepsilon(t, \hat{\theta}_{t-1}) = y_f(t) - \phi(t)^T \hat{\theta}_{t-1} \tag{VII-39}$$

2) Weighted covariance update

$$P_t = P_{t-1} - P_{t-1}\phi(t)[\phi(t)^T P_{t-1}\phi(t) + I_{2\times 2}]^{-1}\phi(t)^T P_{t-1} \tag{VII-40}$$

3) Parameter update

$$\hat{\theta}_t = \hat{\theta}_{t-1} + P_t \phi(t) \varepsilon(t, \hat{\theta}_{t-1}) \tag{VII-41}$$

CONVERGENCE

To avoid convergence of the parameters $\hat{\theta}_t$, when the covariance matrix $P_t \rightarrow 0$, a forgetting factor λ is added to the weighted covariance update in (VII-40) according to Ref. [24].

$$P_t = P_{t-1}\lambda^{-1} - P_{t-1}\lambda^{-1}\phi(t)[\phi(t)^T P_{t-1}\phi(t) + I_{2\times 2}]^{-1}\phi(t)^T P_{t-1} \tag{VII-42}$$

With $\lambda \leq 1$, usually chosen between 0.98 and 1.

Adding the forgetting factor to the weighted covariance update will introduce an exponential weighting over the regressor. By choosing λ small, the contribution of the previous samples in the regressor will be small as well. This will make the filter sensitive to the newest samples in the regressor introducing a fast response to a major frequency change in the input.

SMALL UPDATE

The parameters $\hat{\theta}_t$ are directly used in the controller perturbation $Q(q, \hat{\theta}_t)$ in (VII-32), introducing major possible changes of the controller in the control signal $\hat{u}(t) = -C_Q \hat{e}_2(t)$. To avoid large and rapid changes of the control signal, the parameters $\hat{\theta}_t$ will be time filtered before it is used in the controller perturbation. The calculation of the time filtered parameters $\tilde{\theta}_t$ used in $Q(q, \tilde{\theta}_t)$ is given by.

$$\tilde{\theta}_t = (1 - \delta)\hat{\theta}_t + \delta\tilde{\theta}_{t-1} \qquad \text{(VII-43)}$$

With $0 \leq \delta < 1$. The closer δ is to 1, the more filtering and the slower the update of the controller.

FILTER W

To avoid updating of the controller for frequencies above the useful frequency range, the user-specified filter W will be a model similar to the inverse of a low pass filter. By introducing a large gain for high frequencies in the 2-norm minimization via $\gamma W \hat{u}(t)$, the updating algorithm will try to reduce the gain for high frequencies to get rid off this massive gain in the first argument of the 2-norm, see (VII-20). In other words, It will 'reduce' the influence of high frequencies on the Youla Updating controller.

Similar for frequencies lower than the left boundary of the frequency range. By introducing a low pass filter the influence of low frequencies in the RLS algorithm will be less. Combining the high frequency inverse low pass filter and the low frequency low pass filter will result into a filter W with a high gain outside and a small gain inside the useful frequency range.

A possible fourth-order model for W, satisfying both boundaries, is shown in Figure 28 and given by.

$$W = \frac{z^4 - 2.945z^3 + 3.286z^2 - 1.703z + 0.3628}{0.08676z^4 - 0.0008676z^3 - 0.1709z^2 + 0.0008546z + 0.08419} \qquad \text{(VII-44)}$$

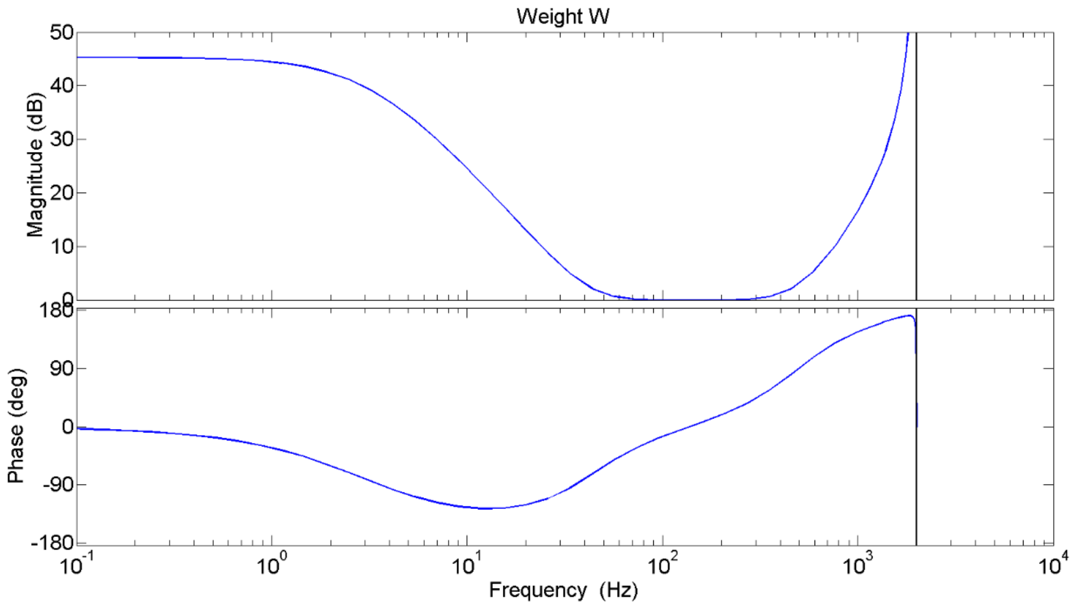

Figure 28: Filter W

SIMULATION

Implementing the updating algorithm for controller C_Q into a SIMULINK model with all the nominal models as given in chapter III, makes it possible to estimate the outputs $\hat{e}_1(t)$ and $\hat{e}_2(t)$ for an unknown input $n(t)$. The results, for white noise input $n(t)$, are shown in Figure 29 and Figure 30 . A sample frequency of $4000\ Hz$ in combination with a 15th order Youla parameter ($n = 15$) is used. The order of the controller is based on the calculation time of the Youla Updating algorithm on the computer in practice. By increasing the order of the controller the calculation time will increase as well. After some trial and error, the highest possible order, at a sample frequency of $4000\ Hz$, equals 15. For $n = 16$ the calculation time of the algorithm is larger than the sample time and the computer will crash. To make both simulation and practice comparable, the same sample frequency and model order will be used in SIMULINK as in practice.

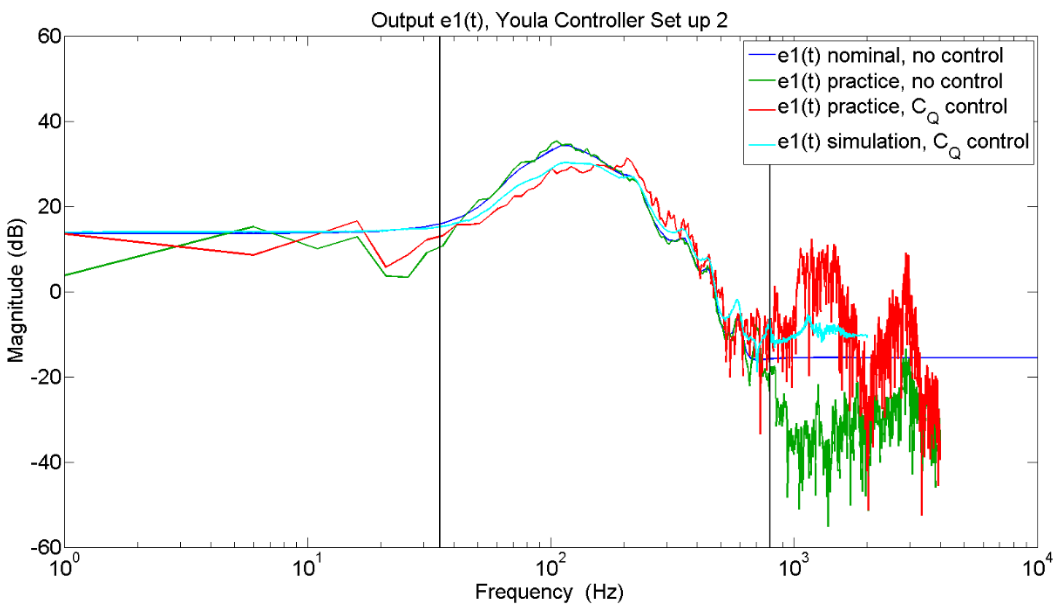

Figure 29: Output $e_1(t)$ Set up 2, Youla Controller

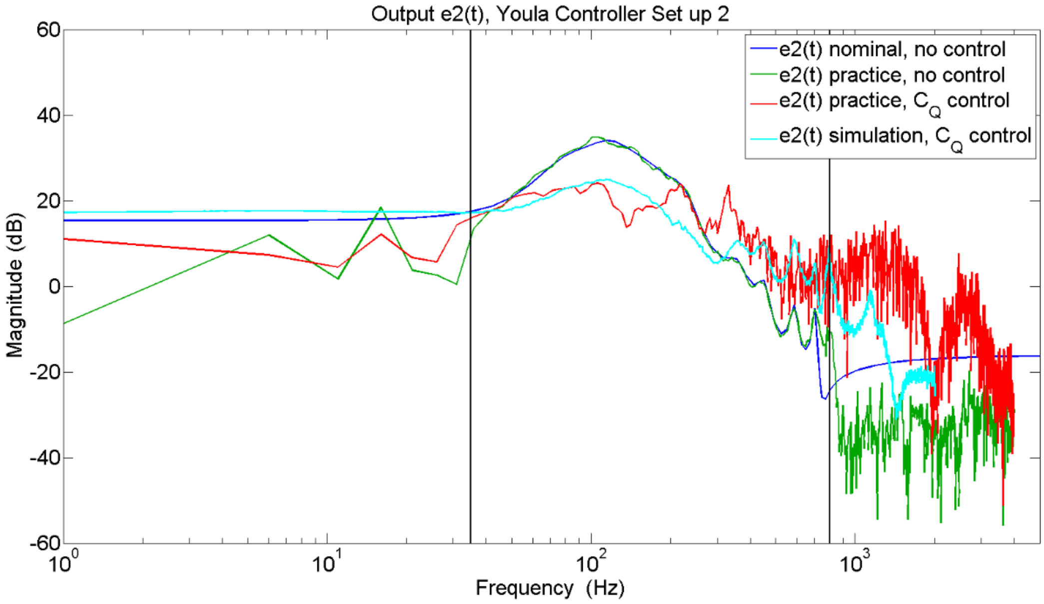

Figure 30: Output $e_2(t)$ Set up 2, Youla Controller

Since the input is white noise the updating algorithm cannot really use its capabilities of updating the controller in time. The result is expected to be better than the nominal loop shaped controller but not that much. Comparing Figure 21 and Figure 30 or Figure 20 and Figure 29 the result is indeed similar but slightly better for the Youla Updating controller. Random noise is never the same for a small moment of time, so updating the controller with the time-filter for stability, will never result into a major controller change compared to the initial stabilizing controller C_i. One possible expected change will be the gain of the controller which will be updated in the Youla Updating Controller to obtain the perfect gain for the situation. Since the gain of the loop shaped controller is chosen at the safe side, according to the stability criteria, the gain of the Youla controller is expected to be larger.

By choosing a forgetting factor in the Youla algorithm, the controller will never totally converge to a final controller but keeps updating all the time when new samples are measured and entering the regressor. After some time the Youla controller C_Q looks like shown in Figure 31.

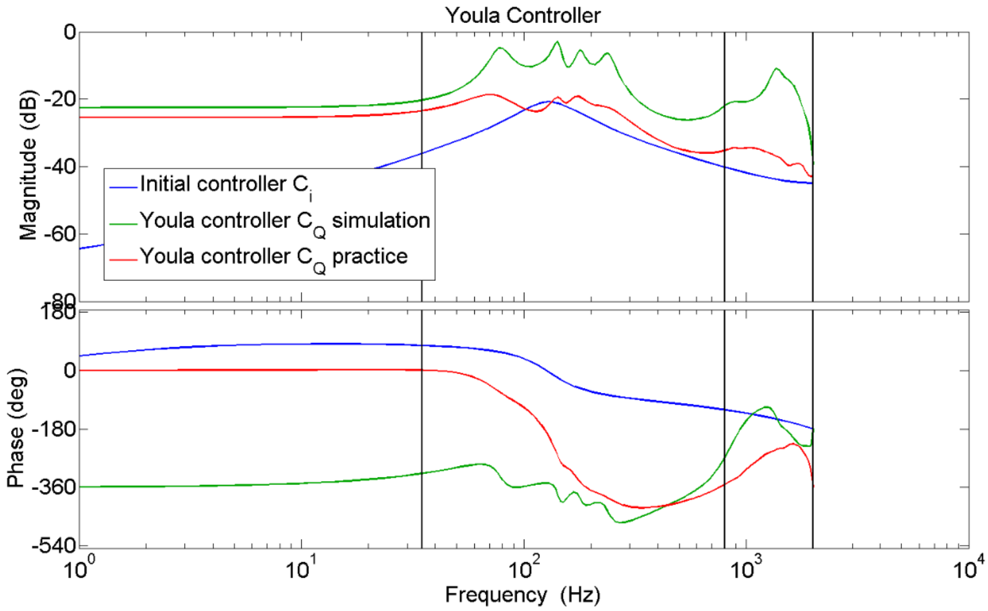

Figure 31: Youla Controller Set up 2

As expected the Youla Controller has a larger gain than the initial controller C_i. A high peak at the end of the frequency domain is introduced by the FIR filter used to describe model Q as a stable model. By looking at the shape of the initial controller, compared to the two Youla Updating controllers, the controller shows similar behavior with a small gain in the lower frequency range, a higher gain in the frequency range $50 \rightarrow 250\ Hz$ and a negative slope for higher frequencies.

UPDATING SIMULATION

To see the updating performance of the algorithm a varying single sine input $n(t)$ is applied to see the response and capabilities of the Youla Updating controller C_Q. Starting with a $80\ Hz$ single sine for 5 seconds, followed by a single sine of $150\ Hz$ for 5 seconds and ending with a single sine of $200\ Hz$. To show the updating performance in time, delta is chosen close to one in (VII-43). $\delta \rightarrow 1$ will slow down the update of the controller and makes the response visible as a small cone, see Figure 32 and Figure 33 for the results and cone behavior.

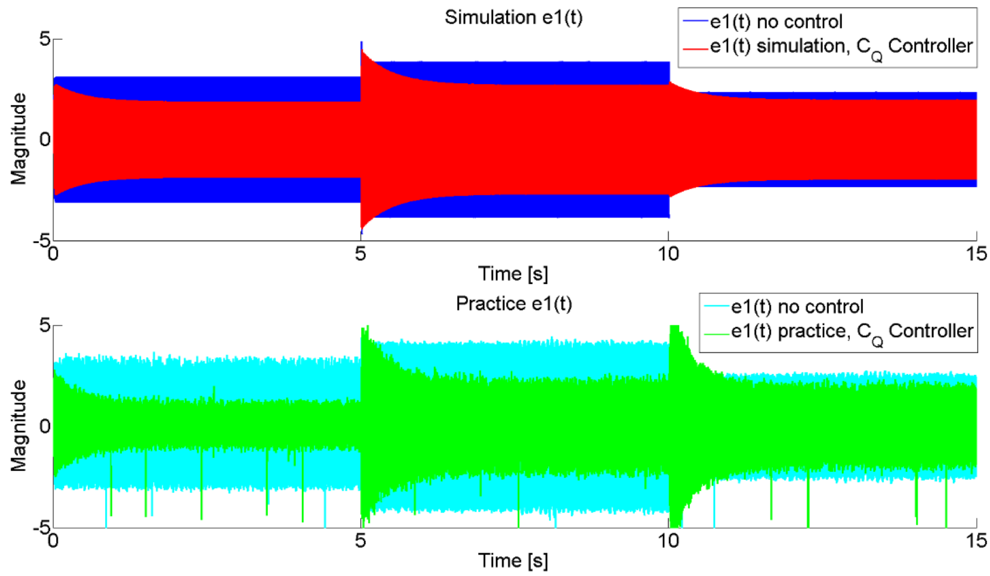

Figure 32: Time output $e_1(t)$ Set up 2, single sine input $n(t)$

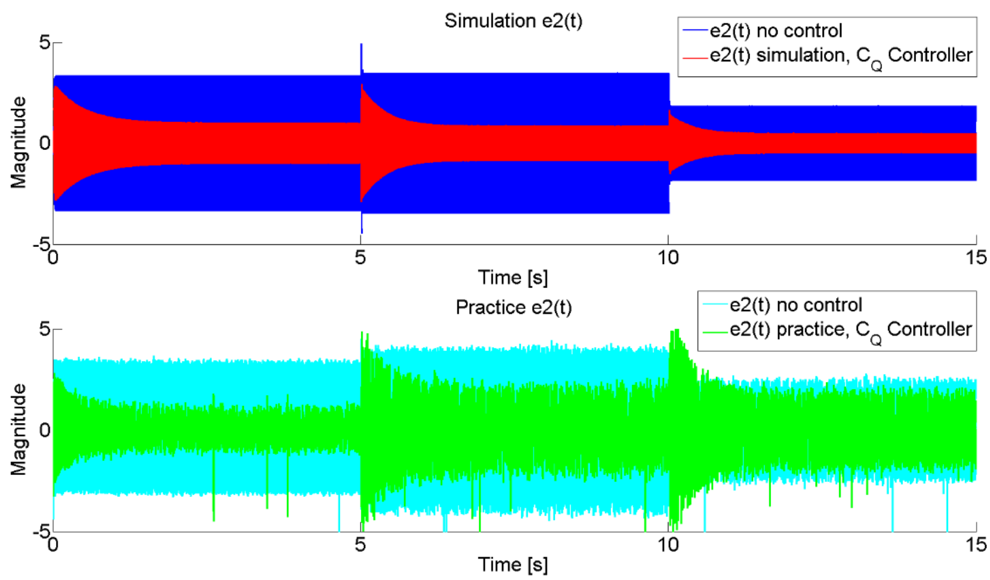

Figure 33: Time output $e_2(t)$ Set up 2, single sine input $n(t)$

The updating capabilities of the controller algorithm are clearly visible in both figures. When the frequency of the input changes (for example at $t = 5\ s$) the error is large because the controller is still based on the previous input frequency. When the new samples are measured and the new frequency is coming into the regressor, the controller updates itself and reduces the new frequency. This behavior repeats itself for every major frequency change in the input $n(t)$.

Compare the red and green lines corresponding to the SIMULINK simulation and results in practice respectively. You can see similar lines for $e_1(t)$ (green line in Figure 32) and $e_2(t)$ (green line in Figure 33) in practice but the red lines in both figures are not that identical. The controller is calculated minimizing the 2-norm of output $e_2(t)$, according to (VII-20), so more reduction is expected for output $e_2(t)$. Since both outputs do have a similar sound path for the frequencies $80, 150$ and $200\ Hz$ (see Figure 7), the difference is small in practice but in the simulation the outputs are totally different. You can see the controller working on $e_2(t)$ much better in the simulation than in practice but $e_1(t)$ in the simulation is worse than in practice. This is mainly introduced by unknown noise in practice not perfectly taken into account in the simulation. White noise is added to the simulation, see Figure 41, to see the behavior of noise on the stability of the Youla Updating controller but is not identical to the noise in practice.

By applying an input signal with only one major frequency a controller with the same dominant frequency is expected. A magnitude increase or peak at the input frequency is expected as a reaction of the Youla Updating controller to reduce that frequency. To see this behavior the FIR models of the controller are plotted for $t = 5, 10$ and 15 just before the frequency change happens, see Figure 34.

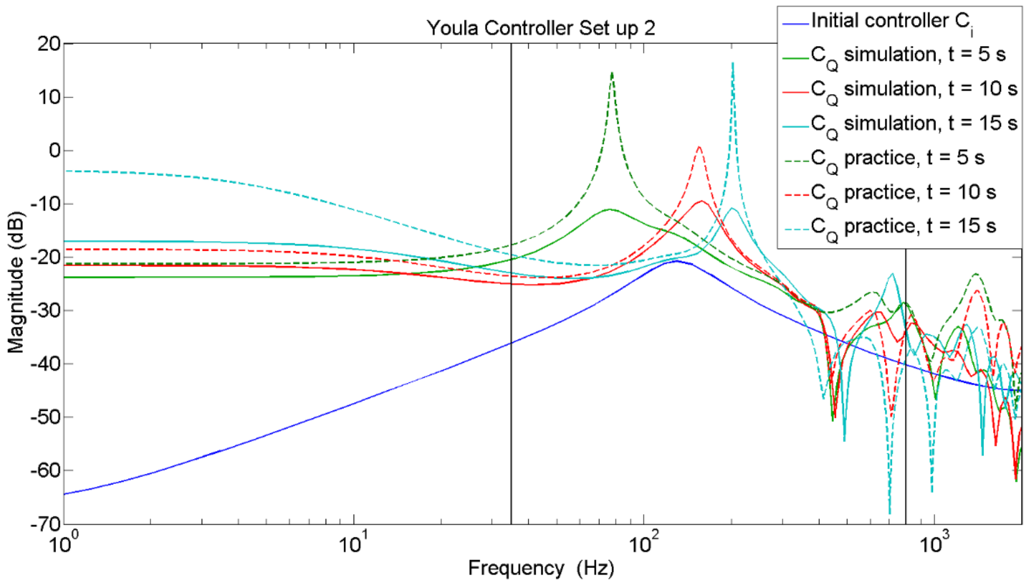

Figure 34: Youla controller Set up 2, single sine input $n(t)$

As you can see, the controller in practice shows the same peaks as the simulated controller but the peaks are larger. Probably introduced by the difference in measurement nose between simulation and practice. For both situations the peaks are exactly at the expected positions corresponding to the input frequencies $80, 150$ and 200 Hz so we can say that the updating controller is working as expected.

VIII YOULA CONTROLLER GENERAL SET UP

In chapter 0 the Youla Controller approach is shown for input $n(t)$ and output $e_2(t)$ according to Set up 2. In practice the most outer microphone, corresponding to output $e_1(t)$, will be the leading microphone measuring the amount of acoustic noise cancellation. To improve the noise cancellation at output $e_1(t)$ the updating Youla Controller can be written according to Set up 1 instead of Set up 2. Unfortunately, the acoustic path G_1 of Set up 1 introduces more delay and that decreases the controllability of the system. The final results are even worse than using Set up 2 and simply measuring the output $e_1(t)$ as in Figure 32. Another possibility is to use the system in Set up 2, but change the Youla Updating controller into a minimizing 2-norm for output $e_1(t)$ instead of $e_2(t)$. This is the so called 'General Set up'.

YOULA PARAMETERIZATION GENERAL SET UP

The closed loop transfer function T, Youla Controller C_Q and equations for the nominal outputs as presented in chapter 0 will remain the same for the General Set up.

REPRESENTATION GENERAL SET UP

A schematic representation of the Youla Controller in the General Set up is given in Figure 35.

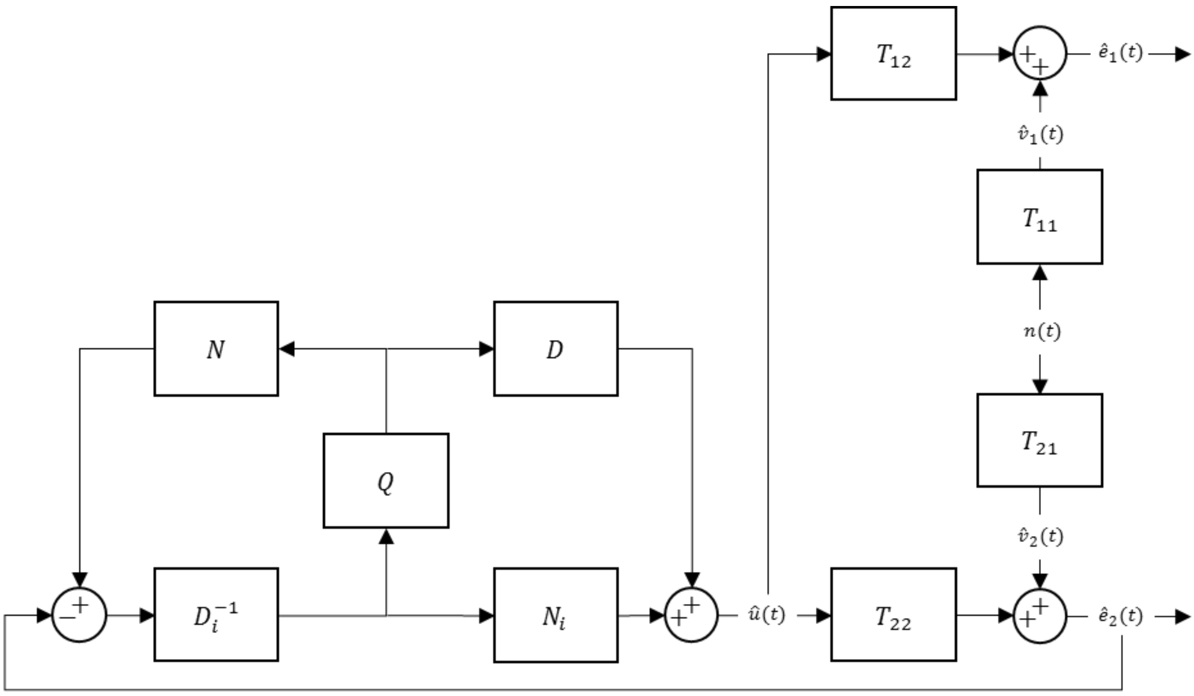

Figure 35: Youla parameterization General Set up

This time, our goal is to minimize the output $\hat{e}_1(t)$ and stabilize the nominal sensitivity according to the stabilizing criteria for output $\hat{e}_2(t)$. To do so the approach of the General Set up minimizes the following weighted two-norm performance measure.

$$\left\| \begin{matrix} \gamma W \hat{u}(t) \\ \hat{e}_1(t) \end{matrix} \right\|_2 \tag{VIII-1}$$

The variance of $\hat{e}_1(t)$ and $\hat{u}(t)$ is driven by the input $n(t)$ and nominal model T_{11} according to (VII-1), in which $n(t)$ is an unknown noise signal. $\hat{u}(t)$ is already given in (VII-8) and $\hat{e}_1(t)$ can be obtained from (VII-1).

$$\hat{e}_1(t) = T_{12}\hat{u}(t) + T_{11}n(t) \tag{VIII-2}$$

Linear in Q for General Set up

The Youla parameterization allows the weighted 2-norm to be written as a function of the stable Youla parameter Q. To write the 2-norm as a function of Q, the Youla Updating controller in (VII-15), $\hat{e}_1(t)$ in (VIII-2) and $\hat{u}(t)$ in (VII-8) are substituted into both terms of the performance measure in (VIII-1).

$$\gamma W\hat{u}(t) = -\gamma W(N_i + DQ)(D_i - NQ)^{-1}\hat{e}_2(t) \tag{VIII-3}$$
$$\hat{e}_1(t) = -T_{12}(N_i + DQ)(D_i - NQ)^{-1}\hat{e}_2(t) + T_{11}n(t) \tag{VIII-4}$$

Substitute the expression for $\hat{e}_2(t)$ in (VII-23) into (VIII-3) and (VIII-4) and rewrite.

$$\gamma W\hat{u}(t) = -\gamma W T_{21} D_i^{-1} \frac{N_i + DQ}{I + ND^{-1}N_i D_i^{-1}} n(t) \tag{VIII-5}$$
$$\hat{e}_1(t) = \left(T_{11} - T_{12}T_{21}D_i^{-1}\frac{N_i + DQ}{I + ND^{-1}N_i D_i^{-1}}\right) n(t) \tag{VIII-6}$$

(VIII-5) and (VIII-6) can be written as a vector summation linear in Q.

$$\begin{bmatrix} \gamma W\hat{u}(t) \\ \hat{e}_1(t) \end{bmatrix} = -\begin{bmatrix} \gamma W N_i \\ T_{12}N_i \end{bmatrix} w(t) - \begin{bmatrix} \gamma W D \\ T_{12}D \end{bmatrix} Q w(t) + \begin{bmatrix} 0 \\ T_{11}T_{21}^{-1} \end{bmatrix} m(t) \tag{VIII-7}$$

With $w(t)$ according to (VII-26) and $m(t)$ given by.

$$m(t) = T_{21}n(t) \tag{VIII-8}$$

$T_{21}n(t)$ is known in (VII-28) and can be substituted in (VIII-8).

$$m(t) = \begin{bmatrix} T_{22} & I \end{bmatrix} \begin{bmatrix} -\hat{u}(t) \\ \hat{e}_2(t) \end{bmatrix} \tag{VIII-9}$$

Based on this analysis, the filtered closed loop signals $w(t)$ and $m(t)$ can be defined with $\Lambda^{-1} \in \mathcal{RH}_\infty$, $P(C_i, T_{22}) \in \mathcal{RH}_\infty$ and $T_{22} \in \mathcal{RH}_\infty$. With $w(t)$ in (VII-29) and $m(t)$ given in (VIII-10).

$$m(t) = \begin{bmatrix} T_{22} & I \end{bmatrix} \begin{bmatrix} -\hat{u}(t) \\ \hat{e}_2(t) \end{bmatrix} = \tilde{D}^{-1}\begin{bmatrix} N & \tilde{D} \end{bmatrix} \begin{bmatrix} -\hat{u}(t) \\ \hat{e}_2(t) \end{bmatrix} \tag{VIII-10}$$

UPDATING

The same Youla Updating controller algorithm and approach as in chapter 0 will be used but with a different error signal. (VII-34) for Set up 2 changes in (VIII-11) for the General Set up.

$$\varepsilon(t,\theta) = y_f(t) - Q(\theta)u_f(t) + n_f(t) \tag{VIII-11}$$

With the filtered output, filtered input and extra term according to.

$$y_f(t) = -\begin{bmatrix} \gamma W N_i \\ T_{12} N_i \end{bmatrix} w(t) \tag{VIII-12}$$

$$u_f(t) = \begin{bmatrix} \gamma W D \\ T_{12} D \end{bmatrix} w(t) \tag{VIII-13}$$

$$n_f(t) = \begin{bmatrix} 0 \\ T_{11} T_{21}^{-1} \end{bmatrix} m(t) \tag{VIII-14}$$

SIMULATION

Again the algorithm is implemented in SIMULINK, this time according to the General Set up. The results, for white noise input $n(t)$, are shown in Figure 36 and Figure 37. A sample frequency of $4000\ Hz$ in combination with a 15th order model ($n = 15$) are used again.

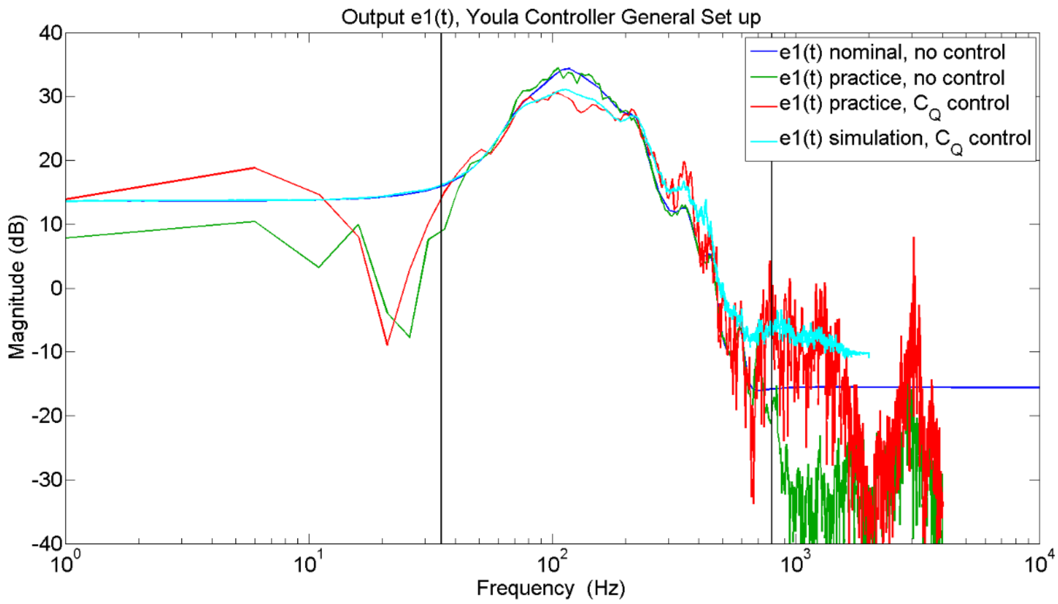

Figure 36: Output $e_1(t)$, Youla Controller General set up

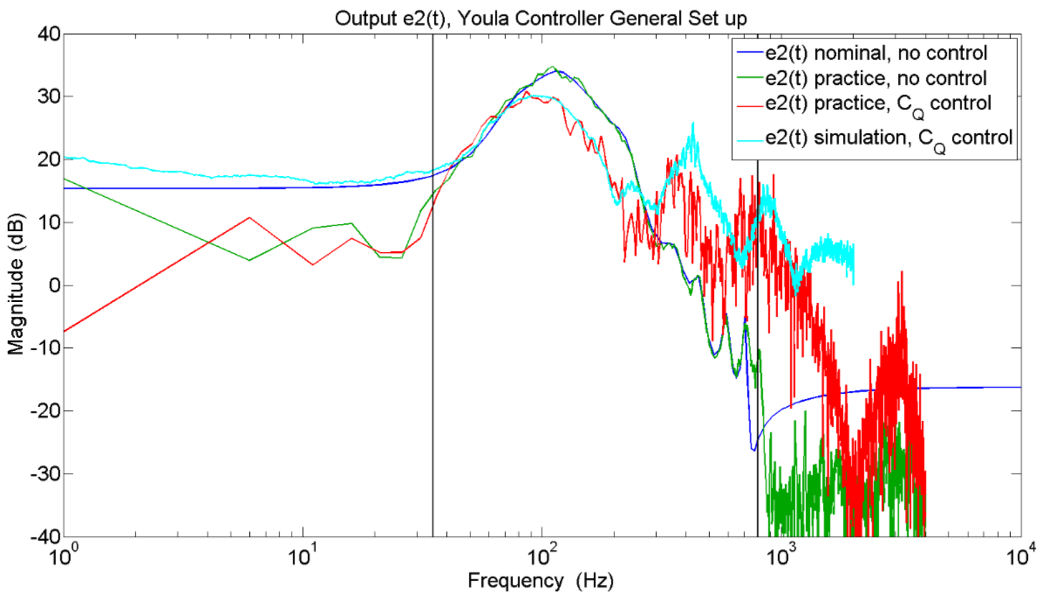

Figure 37: Output $e_2(t)$, Youla Controller General set up

The result for output $e_2(t)$ in Figure 37, compared to the Set up 2 situation in Figure 30, is worse as expected. Calculating the Youla Controller by minimizing the 2-norm of output $e_1(t)$ will result into a better performance for $e_1(t)$ but not for $e_2(t)$. Comparing output $e_1(t)$ in Figure 36 with the previous Set up 2 situation in Figure 29, will show a is slightly better noise reduction for higher frequencies in the General Set up. The differences are small but the General Set up is slightly better for output $e_1(t)$.

UPDATING SIMULATION

Again the updating performance will be checked by applying an input signal with the same changing single sine frequency over time as in the previous chapter. The time domain results are shown in Figure 38 and Figure 39.

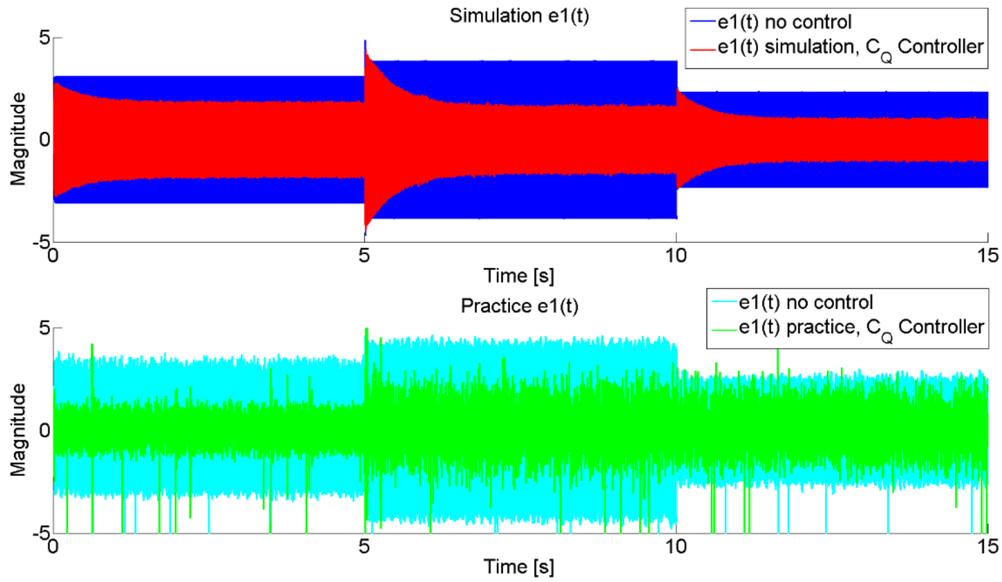

Figure 38: Time output $e_1(t)$ General set up, single sine input $n(t)$

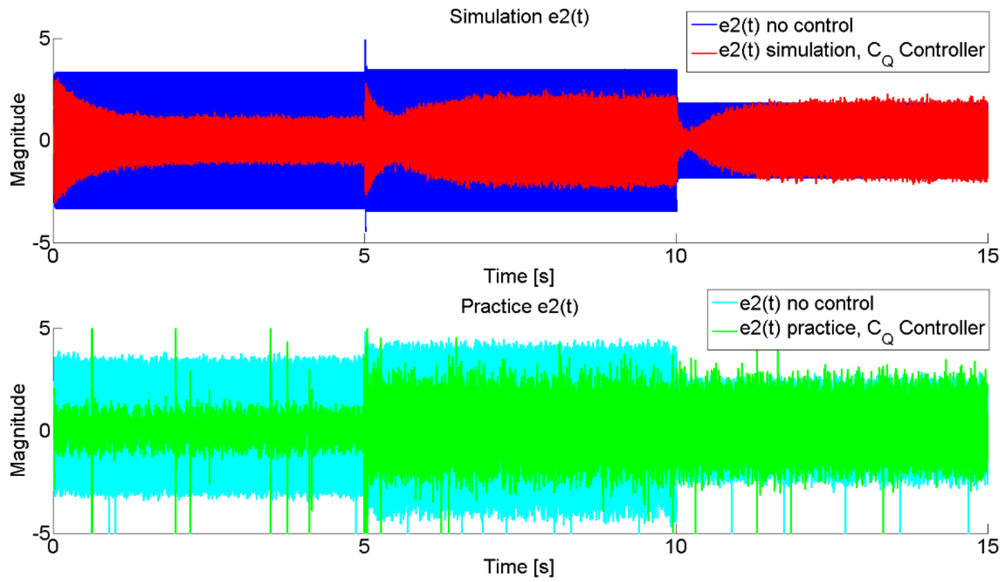

Figure 39: Time output $e_2(t)$ General set up, single sine input $n(t)$

For the random noise input the differences between Set up 2 and the General set up were small but that is not directly the case for the changing single sine input. By looking at the response of output $e_2(t)$ (compare Figure 32 with Figure 38), major changes are visible because the algorithm is not minimizing the 2-norm of $e_2(t)$ anymore but uses $e_2(t)$ only as the control input. That will give the controller the opportunity to tune $e_2(t)$ without any restrictions to optimize output $e_1(t)$ as good as possible. This is visible as a strange behavior for $e_2(t)$ in the simulation part of Figure 39 (red line). The

General Set up minimizes the output $e_1(t)$ and shows a better result in the SIMULINK simulation (compare Figure 33 with Figure 39). In practice the result is slightly better but almost neglectable.

Remark: In the practical calculations for the General Set up the filtering factor δ in (VII-43) wasn't chosen in such a way that the update in time is visible. This was done for all calculations in Set up 2 and the SIMULINK simulation for the General Set up, to make the update visible as small cones in the time domain figures. As you can see the practical green line for the General Set up does not show these cones because the filtering factor is not chosen close to one but the update is still working.

Similar as in the previous chapter a Youla Updating controller figure is generated to compare the results of the SIMULINK simulation with practice. Again the shape of both controllers is the same and fits even better than for the Set up 2 situation in Figure 34. A small unexpected change in the red and light blue lines is the position of the peak. For the Set up 2 situation these peaks were exactly at the frequencies corresponding to the input frequencies. In this situation the green peak is correct but the red and light blue major peaks are positioned at a slightly higher frequency than the incoming frequency. Somehow the change in the 2-norm influences the controller update to come up with slightly higher frequency peaks.

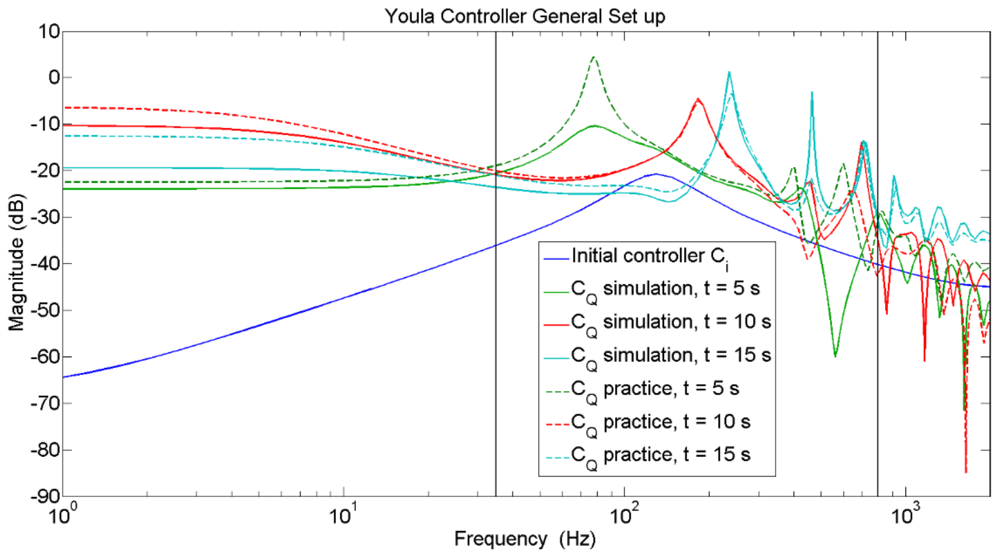

Figure 40: Youla Controller General set up, single sine input $n(t)$

IX SIMULINK MODEL

In the three previous chapters three different controllers are designed and the performances are simulated with SIMULINK. The implementation in SIMULINK of the nominal models representing the practical acoustic paths and the designed controller will be discussed in this chapter.

Since two different set ups (Set up 2 and the General Set up) are used in practice to calculate the Youla Updating Controller, the same is done for the SIMULINK simulation. In both set ups the system representation in SIMULINK is identical but the calculation of the Youla Updating controller differs.

SIMULINK REPRESENTATION

All four nominal models given in equations (VII-2) - (VII-5) will be used to create the nominal set up in SIMULINK according to Figure 22.

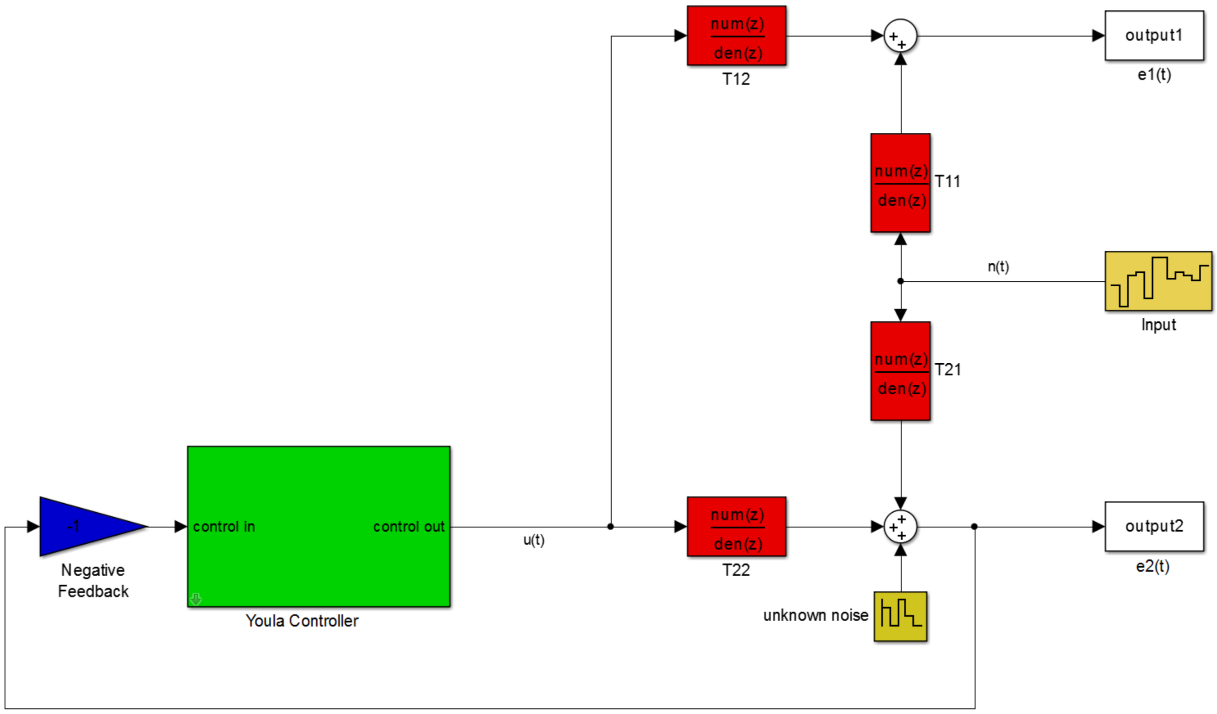

Figure 41: SIMULINK model

In Figure 41 the controller is shown as a green box with in this case one of the Youla Controllers implemented. By replacing the green box all three controllers can be implemented in the SIMULINK simulation. As you can see SIMULINK Figure 41 is similar as the schematic in Figure 22 both representing the practical set up as good as possible.

LOOP SHAPED CONTROLLER

Since the loop shaped controller designed in chapter V is only a second order transfer function, the green box can be replaced by a transfer function according to (V-4).

YOULA UPDATING CONTROLLER

To implement the updating controllers for Set up 2 in chapter 0 and the General Set up in chapter VIII, the green box will be replaced by the Youla Updating controller. Both SIMULINK representations of the controller in Set up 2 and the General Set up are given in appendix C. To update the Youla parameter an Embedded MATLAB function is used calculating the standard recursive least square (RLS) as given in chapter 0. Both Embedded MATLAB functions for Set up 2 and the General Set up are given in appendix D. Remark: Both embedded MATLAB functions are almost identical except for the error function given in (VII-34) and (VIII-11) for respectively Set up 2 and the General Set up.

PRACTICE

Since the nominal set up of Figure 22 is perfectly representing the situation in practice, the Youla Updating Controllers can be used in practice as well. For the loop shaped controller the $zoh * C$ term will be replaced by a zero-order-hold discretized version of C to be used in practice.

X CONCLUSIONS

Four different methods are discussed in the previous four chapters of this report. All these methods are designed to reduce the incoming sound and obtain a stable feedback loop except for the H_2 method in chapter VI. After applying a simple loop shaped low order controller in chapter V, two updating Youla controller approaches are discussed in chapter 0 and VIII. Both updating controllers do improve the performance of the system especially when the response of the input signal contains dominant frequencies. For a random noise input the updating algorithms are not that useful and will not improve the noise reduction that much.

Both updating Youla controllers do behave as expected and do change in time when the frequency spectrum of the incoming sound changes and fills the regressor. By comparing Figure 32 and Figure 38 the algorithm based on the 2-norm minimization of $e_1(t)$ and named as the General Set up, shows a better result for output $e_1(t)$. For the single sine input the reduction is almost up to $50\ \%$ in the frequency range $50 \rightarrow 150\ Hz$ and decreases for higher frequencies. This reduction is clearly hearable with a human ear and visible in Figure 38. Frequencies above $300\ Hz$ are not effectively reduced anymore.

The best method reducing the sound at output $e_1(t)$ is a Youla Updating controller algorithm based on the General Set up with Set up 2 as a close second. In practice it was really hard to make a stable nominal model for the T_{21}^{-1} term in (VIII-14) needed for the General set up updating algorithm. The best stable fit was not that good but still made the algorithm work properly.

In the end it is possible to reduce unknown incoming sound using in a feedback loop system but the reduction is only present in the lower frequency range, say $50 - 300\ Hz$. Fortunately, in the air conditioning air duct high frequencies were already reduced by a passive damper but frequencies in the range $300 - 800\ Hz$ couldn't be reduced at all. The major cause for this problem is the speed of the computer in combination with the calculation time for the updating controller. A sample frequency of $4000\ Hz$ was not fast enough to update the controller for frequencies above $300\ Hz$ but increasing the sample frequency was not possible due to computer limitations.

Finally, it is possible to reduce incoming sound up to a reduction of $50\ \%$ for certain frequencies by using a feedback system with a control speaker and a microphone at a small distance downstream.

XI RECOMMANDATIONS

To improve the performance of the practical system with an updating controller as in the General Set up, the following recommendations are given.

FILTER CUT-OFF FREQUENCY

Stability is obtained by satisfying the stability criteria in chapter I based on the closed loop gain of the system. By adding two filters to the practical set up the closed loop gain changes to equation (IV-13). Both filters do introduce delay influencing the stability criteria in a negative way. In this report both filters do have a chosen cut-off frequency around $1000\ Hz$ based on the right boundary of the effective frequency range $(800\ Hz)$. Further research can be done tuning the cut-off frequency of both filters to obtain high frequency reduction but also looking at the amount of delay introduced by the filters. The more delay in the system, the smaller the effective range of the controller can be and the worse the performance is in practice. Maybe an optimum in the cut-off frequency can be found balancing between high frequency reduction and closed loop gain delay.

SPEAKER AND MICROPHONE

Beside filter delay a major part of the closed loop gain delay is introduced by the acoustic sound path between control speaker and microphone. The distance between the control speaker and microphone 2 is set at $5\ cm$ but is still changeable in practice. In the AC air duct reducing the distance even further did not change the phase diagram of the acoustic path so $5\ cm$ seemed to be the best distance. Unfortunately, in the presented AC air duct, the speaker is much larger than the microphone (see Figure 1). This will result into an unknown sound propagation between control speaker and microphone 2, definitely not a plane wave as is the case when the microphone will be placed further away, like microphone 1. Changing the 'tube' position of the microphone might introduce a different sound path showing non-plane sound waves propagating between the control speaker and microphone 2. In this report the sound waves are assumed to be plane waves but that is almost impossible at such a small distance. Further research can be done by looking at the sound waves and sound propagation between control speaker and microphone. Think of, for example, a smaller control speaker that will probably introduce plane sound waves within a shorter distance and improve the performance of the system.

PRACTICAL IMPLEMENTATION

In this report a maximum sample frequency of $4000\ Hz$ and a 15^{th} order Youla parameter are used. By improving the calculation algorithm or changing the capabilities of the computer, the sample frequency or the updating controller order can be increased. By increasing the sample frequency the system will respond faster to incoming frequencies and will measure higher frequencies as well. On the other hand increasing the updating controller order will result into a better 2-norm minimization. Both changes will improve the performance of the system so speeding up the calculation will give a better result in the end. Further research can be done improving the implemented SIMULINK algorithm or looking at the performance of a faster computer.

MODEL ORDER

For calculating the Youla controller nominal models of the acoustic paths in practice are used. These models are fitted as good as possible and reduced with a Hankel order reduction. By introducing a good low order fit the Youla Updating controller calculation will be as fast as possible. In this report the difference between a better but higher order model compared to a worse but low order model is not investigated. Increasing the order of the nominal models (see Table 3) will increase the Youla controller calculation time which will then decrease the sample frequency or Youla parameter order to make the calculation possible on the computer in practice. Further research can be done by looking at, for example, an optimum between nominal model orders and Youla parameter order or the influence of the nominal model fits at all.

MICROPHONE 1

For the updating algorithm of Set up 2 in chapter 0, only measurable models with stable nominal model fits are used (see equations (VII-29), (VII-35) and (VII-36)). In chapter VIII, the General Set up, a new term appears in equation (VIII-14) given by the inverse of T_{21}. Since T_{21} is measureable and transformed into a nominal model, the inverse can be calculated. By calculating the inverse T_{21}^{-1} the result was unstable and not usable in the algorithm. To overcome this problem a very simple model of T_{21}^{-1} was created with a bad fit but overall stability. Further research can be done improving the stable nominal model T_{21}^{-1} to create a better nominal representation of the practical set up for the calculation of the Youla Updating controller in the General Set up algorithm.

It is also possible to use the income of microphone 1 as an input for the 2-norm performance measure shown in equation (VIII-1), instead of the calculated nominal models representing e_1. In this report microphone 1 is only used to measure the performance but never used as an input in the updating algorithm. Further research can be done using both microphone 1 and microphone 2 in the updating algorithm to improve the performance of the system.

UNKNOWN NOISE

As shown in chapter 0 the simulation results and practical results do differ a lot, especially for higher frequencies. This is mainly introduced by the differences in measurement noise. As you can see in Figure 41 unknown noise is added to the system in the SIMULINK model to make a good representation of system noise in practice. But this is just simple white noise with a certain amplitude. Further research can be done identifying the noise in the practical system to improve the corresponding SIMULINK model. The assumed perfect plane waves for the SIMULINK simulation will also change the results if this is not the case in practice as discussed in the *Speaker and microphone* paragraph of this chapter.

LITERATURE

LECTURE NOTES

[1] R.A. de Callafon, Linear Circuit Experiment (MAE171a), 2014

[2] Y. Hayakawa, Control, chapter 2, 2012

[3] K. Glover & J.Gonçalves, Engineering Tripos Part IIB, 2nd part, 2010

[4] H.T. Toivonen, Robust control methods, chapter 3, 1998

[5] A. Megretski, Model reduction, chapter 10: Hankel optimal model reduction, MIT, 2004

[6] L. Piroddi, Active noise control, chapter 8: Feedback active noise control, 2012

[7] A. Megretski, Multivariable control systems, chapter 5: Standard LTI feedback optimization setup, MIT, 2011

BOOKS AND PAPERS

[8] J.C. Doyle, K. Glover, P.P Khargonekar & B.A. Francis, State-Space Solutions to Standard H_2 and H_∞ Control Problems, IEEE TAC, Vol. 34 (1989), No. 8, p. 831

[9] LM6132,Texas Instruments, SNOS751D, February 2013

[10] SCB-68 User Guide, National Instruments Corporation, March 2009

[11] J. Zeng & R.A. de Callafon, Recursive filter estimation for feedforward noise cancellation with acoustic coupling, Journal of Sound and Vibration 291 (2006), p. 1061

[12] J. Zeng & R.A. de Callafon, Feedforward noise cancellation in an airduct using generalized FIR filter estimaton, Proceedings of the 42nd IEEE conf. on Decision and Control, December 2003

[13] R.A. de Callafon & H. Fang, Adaptive regulation via weighted robust estimation and automatic controller tuning, European Journal of Control 19 (2013) p. 266

[14] R.A. de Callafon & C.E. Kinney, Robust estimation and adaptive controller tuning, Journal of Advanced Mechanical Design, Vol. 4 (2010), No. 1, p. 130

[15] P.M.J. Van den Hof & R.A. de Callafon, Multivariable closed-loop identification: From indirect identification to Dual-Youla parameterization, Proceedings of the 35th conf. on Decision and Control, December 1996

[16] S.C. Heuberger, P.M.J. Van den Hof & O.H. Bosgra, A generalized orthonormal basis for linear dynamical systems, IEEE Transactions on automatic control, Vol. 40 (1995), No. 3

[17] B.J. Zimmer, S.P. Lipshitz, K.A. Morris, J. Van der Kooy & E.E. Obasi, An improved acoustic model for active noise control in a duct, Journal of Dynamic systems, measurements and control, Vol. 125 (2003), p. 382

[18] C.E. Kinney, A. Villalta & R.A. de Callafon, Active noise control of a cooling fan in a short duct, Proceedings of NCAD2008

[19] H.H. Clements, Active noise control with H_2 optimal control in a duct silencer, 2010

[20] L.E. Kinsler, A.R. Frey, A.B. Coppens & J.V. Sanders, Fundamentals of Acoustics, NY, 1982

[21] 10 MHz Butterworth Filter Using the Operational Amplifier THS4001, Texas Instruments, Application Report SLOA032, October 1999

[22] Butterworth filter, en.wikipedia.org/wiki/Butterworth_filter, March 5th, 2014

[23] S. Haykin, Adaptive Filter Theory, Prentice Hall, 2002

[24] M.H. Hayes, Statistical Digital Signal Processing and Modeling, chapter 9.4: RLS, 1996

[25] HP 3563A Operating Manual, Control Systems Analyzer, Hewlett Packard, 1990

APPENDIX

A. SELF-MADE BUTTERWORTH FILTER

A second order low pass Butterworth filter can relatively simple be created with two resistors, two capacitors and an I/O operational amplifier LM6132 [9] according to [1] and [21].

A unity gain low pass filter implemented with a Sallen-Key topology is given in Figure 42.

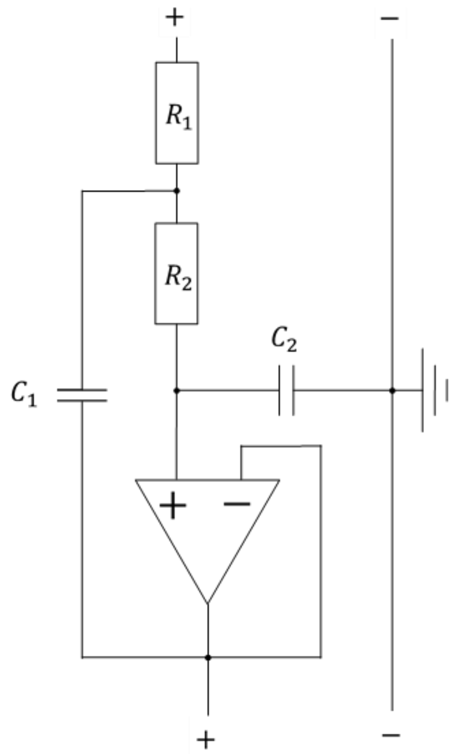

Figure 42: Second order low pass Butterworth filter

The corresponding transfer function for this second order unity-gain low pass filter is given in (A-1) and the variable parameters are given in (A-2) and (A-3).

$$F(s) = \frac{\omega^2}{s^2 + 2\alpha s + \omega^2} \tag{A-1}$$

$$\omega = 2\pi f = \frac{1}{\sqrt{R_1 R_2 C_1 C_2}} \quad \text{(A-2)} \qquad 2\alpha = \frac{1}{C1}\frac{R_1 + R_2}{R_1 R_2} \quad \text{(A-3)} \qquad Q = \frac{\omega}{2\alpha} = \frac{\sqrt{R_1 R_2 C_1 C_2}}{C_2(R_1 + R_2)} \quad \text{(A-4)}$$

(A-1) looks like the second order Butterworth filter given in (II-4). Use (A-4) to rewrite (A-1) in terms of ω and Q instead of ω and α.

$$F(s) = \frac{\omega^2}{s^2 + \frac{\omega}{Q}s + \omega^2} \tag{A-5}$$

Q must be equal to $Q = 1/\sqrt{2}$ to obtain the same second order Butterworth filter as in (II-4).

(A-4) contains 4 unknown parameters, R_1, R_2, C_1 and C_2 and together with (A-2) only two equations are given to design the filter, an under constrained situation. Introduce a ratio m between both resistors and a ratio n for the capacitors to get rid of this issue. For simplicity the remaining resistor will be R and the remaining capacitor C according to (A-7) and (A-9).

$$R_1 = mR \quad \text{(A-6)} \qquad R_2 = R \quad \text{(A-7)}$$
$$C_1 = nC \quad \text{(A-8)} \qquad C_2 = C \quad \text{(A-9)}$$

Substitute the two extra constrains into (A-2) and (A-4).

$$\omega = \frac{1}{RC\sqrt{mn}} \quad \text{(A-10)} \qquad Q = \frac{1}{\sqrt{2}} = \frac{\sqrt{mn}}{m+1} \quad \text{(A-11)}$$

Rewrite (A-11).

$$Q = \frac{\sqrt{mn}}{m+1} = \frac{1}{\sqrt{2}} \rightarrow n = \frac{m^2+2m+1}{2m} \quad \text{(A-12)}$$

A simple model can be obtained by choosing $m = 1$ and calculating $n = 2$ from (A-12). Substitute both ratios into (A-10).

$$\omega = \frac{1}{\sqrt{2}RC} \quad \text{(A-13)} \qquad Q = \frac{1}{\sqrt{2}} \quad \text{(A-14)}$$

In this situation, when $n = 2$ and $m = 1$, two capacitors of size C in parallel will give a sommation of the capacity for $C_1 = 2C$. Both resistors will be identical because the ratio $m = 1$. In practice the second order Butterworth filter can be created with only three capacitors C and two resistors R.

To reduce the high frequency influences on the system, a Butterworth filter with a cut-off frequency of $1000\ Hz$ will be sufficient. To create a $1000\ Hz$ Butterworth filter the product of RC can be calculated with use of (A-13).

$$f = 1000\ Hz = \frac{\omega}{2\pi} = \frac{1}{2\sqrt{2}\pi RC} \rightarrow RC = \frac{1}{2\sqrt{2}\pi f} = 1.125 * 10^{-4} \quad \text{(A-15)}$$

In lab resistors of $R = 10\ k\Omega$ and capacitors of $C = 10\ nF$ were available to create the following Butterworth filter.

$$RC = 1.000 * 10^{-4} \quad \text{(A-16)} \qquad f = \frac{1}{2\sqrt{2}\pi RC} = 1125\ Hz \quad \text{(A-17)}$$

$$R_1 = R_2 = 10\ k\Omega \quad \text{(A-18)} \qquad C_1 = 20\ nF \quad \text{(A-19)} \qquad C_2 = 10\ nF \quad \text{(A-20)}$$

The final cut-off frequency is slightly higher than $1000\ Hz$ but will do the job as well.

B. PROOF FOR (VII-29)

Proof 1

Use (VII-14) to write.

$$\Lambda^{-1}[\widetilde{N} \quad \widetilde{D}] = \begin{bmatrix} \dfrac{\widetilde{N}}{\widetilde{D}D_i + \widetilde{N}N_i} & \dfrac{\widetilde{D}}{\widetilde{D}D_i + \widetilde{N}N_i} \end{bmatrix} \tag{B-21}$$

Multiply nominators and denominators with $D_i^{-1}\widetilde{D}^{-1}$ and use the lcf and rcf of T_{22} and C_i.

$$\Lambda^{-1}[\widetilde{N} \quad \widetilde{D}] = \begin{bmatrix} \dfrac{T_{22}D_i^{-1}}{I + T_{22}C_i} & \dfrac{D_i^{-1}}{I + T_{22}C_i} \end{bmatrix} \tag{B-22}$$

Rewrite (B-22) to obtain.

$$\Lambda^{-1}[\widetilde{N} \quad \widetilde{D}] = D_i^{-1}(I + T_{22}C_i)^{-1}[T_{22} \quad I] \tag{B-23}$$

Proof 2

Use the lcf and rcf factorizations of T_{22} and C_i to rewrite the product of (VII-14) with $D_i^{-1}\widetilde{D}^{-1}$.

$$\Lambda D_i^{-1}\widetilde{D}^{-1} = \left(\widetilde{D}D_i + \widetilde{N}N_i\right)D_i^{-1}\widetilde{D}^{-1} = I + \widetilde{N}N_iD_i^{-1}\widetilde{D}^{-1} = I + T_{22}C_i \tag{B-24}$$

C. SIMULINK

Set up 2 SIMULINK model

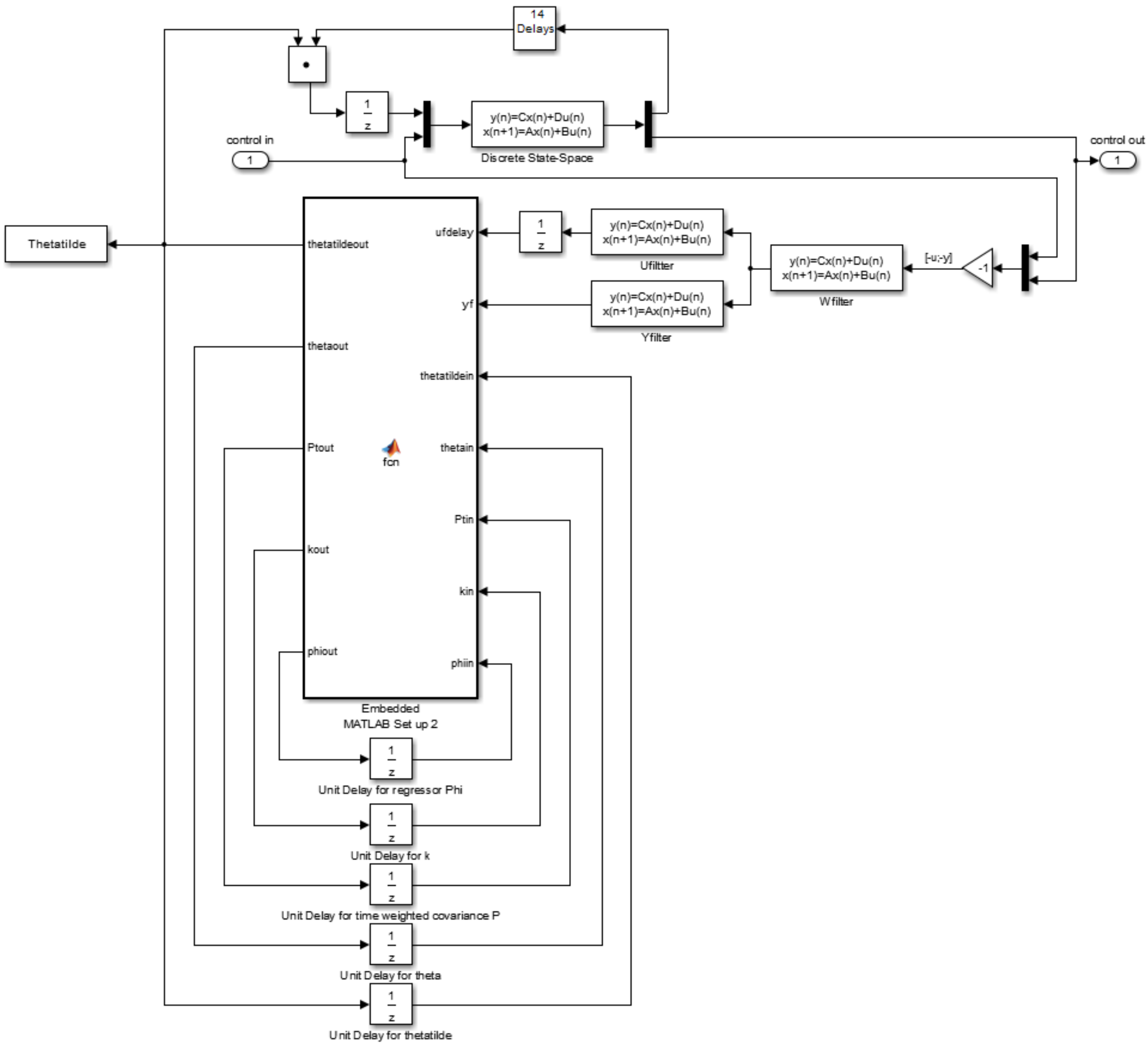

Figure 43: Youla Controller Set up 2

General Set up SIMULINK model

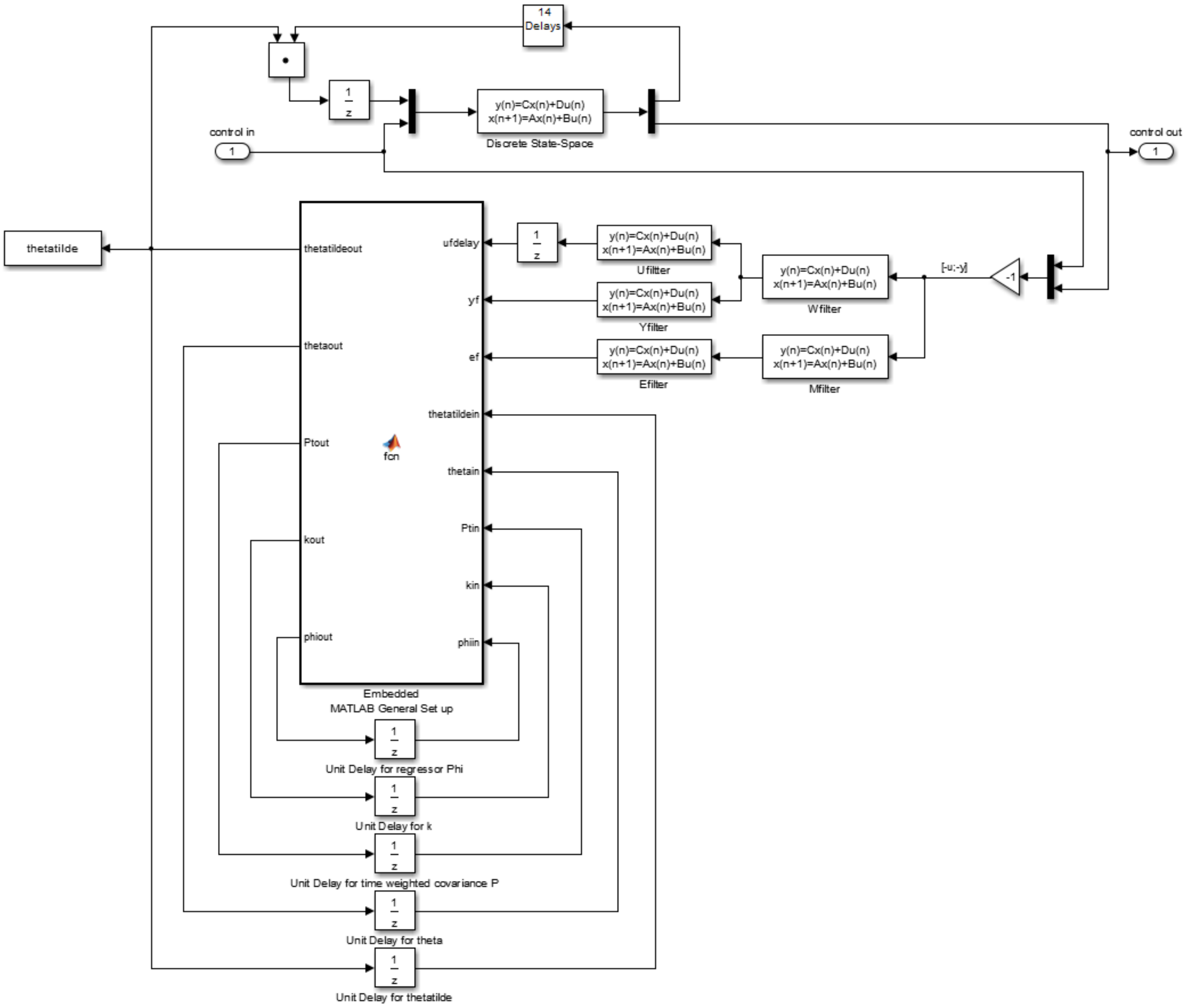

Figure 44: Youla Controller General Set up

D. EMBEDDED MATLAB

Embedded MATLAB function Set up 2

```
function [thetatildeout,thetaout,Ptout,kout,phiout] = fcn(ufdelay,yf,delta,lambda0,kbar,p0,P,thetatildein,thetain,Ptin,kin,phiin)
% update counter (starts at 0)
kout = kin+1;

% Update regressor
phiout = [ufdelay phiin(1:kbar-1,:)']';

%% STANDARD RLS UPDATE ALGORITHM
% wait till we have filled up our regressor
if  kout>kbar+2

    % update prediction error
    pe = yf-phiout'*thetain;

    % update time weighted covariance matrix
    Ptout = Ptin/lambda0 - Ptin/lambda0*phiout*inv(phiout'*Ptin*phiout + lambda0*eye(2))*phiout'*Ptin;

    % update parameter
    thetaout = thetain + Ptout*phiout*pe;

else
    % reset everything back to zero and/or initial values
    Ptout = p0*P;
    thetaout = zeros(kbar,1);
end

% filter updated parameter
thetatildeout = (1-delta)*thetaout+delta*thetatildein;

end
```

Figure 45: Embedded MATLAB Set up 2

Embedded MATLAB function General Set up

```
function [thetatildeout,thetaout,Ptout,kout,phiout] = fcn(ufdelay,yf,ef,delta,lambda0,kbar,p0,P,thetatildein,thetain,Ptin,kin,phiin)
% update counter (starts at 0)
kout = kin+1;

% Update regressor
phiout = [ufdelay phiin(1:kbar-1,:)']';

%% STANDARD RLS UPDATE ALGORITHM
% wait till we have filled up our regressor
if  kout>kbar+2

    % update prediction error
    pe = yf-phiout'*thetain+ef;

    % update time weighted covariance matrix
    Ptout = Ptin/lambda0 - Ptin/lambda0*phiout*inv(phiout'*Ptin*phiout + lambda0*eye(2))*phiout'*Ptin;

    % update parameter
    thetaout = thetain + Ptout*phiout*pe;

else
    % reset everything back to zero and/or initial values
    Ptout = p0*P;
    thetaout = zeros(kbar,1);
end

% filter updated parameter
thetatildeout = (1-delta)*thetaout+delta*thetatildein;

end
```

Figure 46: Embedded MATLAB General Set up