Calibration, identification and position control of a rotating drum

Calibration, Identification and Position Control of a Rotating Drum

G.J.L. Naus and F.P. Wijnheijmer

DCT Report no: 2004.25
februari 2004

TU/e Traineeship Report
februari 2004

Supervisors:
prof.dr.ir. M. Steinbuch (TU/e)
dr.ir. T.A.G. Heeren (Océ)

University of Technology Eindhoven (TU/e)
Department of Mechanical Engineering
Division Dynamical Systems Design
Control System Technology group

Océ-Technologies B.V.
Research and Development
GRT1

Summary

The print quality of a toner copier depends on the positioning accuracy of the toner relative to the paper. An important part in this positioning process is the timing of the toner particles deposit. This is determined by a balance between magnetic and electric forces, which is influenced by the distance between the *sleeve* of the *DIP-knife* and the *DIP-drum*. Making this distance adjustable, online control to respond to present disturbances will improve the print quality.

A test frame is present in which the distance between the *DIP-knife* and the *DIP-drum* can be adjusted by means of elastic hinges. Two independently controllable voicecoil actuators, which are mounted to the hinges, control this distance. The actuators are controlled by a DSP using a PC as interface and for data storage.

First, the various parts of the test frame are calibrated and if necessary linearized: the position sensors, the voicecoil actuators, the elastic hinges, the control of the actuators by the DSP and the total system. Next the input signals are identified, so a list of controller demands can be made.

With these data, a whitebox as well as a blackbox model is designed which gives a good overview of the characteristics of the system. By using these models, the system is decoupled, which simplifies the design of a controller fulfilling the stated demands. A simulation model is made using the whitebox model, so controllers can be tested and evaluated and the closed-loop system is simulated before the controllers are implemented in the actual system.

Using *loopshaping*-techniques a controller is designed. During the design process of the controllers, an important part of the time is spent on the implementation of the continuous controllers designed with Matlab. Unfortunately the final controllers do not fulfill the stated demands, but recommendations to improve some parts of the system are stated.

Samenvatting

De printkwaliteit van een toner copier is afhankelijk van de nauwkeurigheid waarmee toner op het papier wordt gepositioneerd. Een belangrijk onderdeel in deze positionering is het moment waarop wordt bepaald welke tonerdeeltjes wel en welke niet worden geprint. Dit wordt bepaald door een krachtenspel tussen magnetische en elektrische krachten, welke beïnvloed wordt door de afstand tussen de huls (*sleeve*) van het *DIP-mes* en de *DIP-drum*. Door deze afstand instelbaar te maken, kan online op verstoringen worden gereageerd en de printkwaliteit worden gegarandeerd.

Er is een proefopstelling voorhanden waarbij de afstand van het *DIP-mes* tot de *DIP-drum* middels elastische scharnieren in te stellen is. De aansturing wordt door twee onafhankelijk aan te sturen duikspoel actuators welke met de elastische scharnieren zijn verbonden, verzorgd. Een DSP regelt de aansturing van de actuators waarbij de PC als interface en voor data opslag fungeert.

Allereerst zijn de verschillende onderdelen van de opstelling gecalibreerd en zo nodig en indien mogelijk gelineariseerd. Het gaat hierbij om de positie sensoren, de actuators, de elastische scharnieren, de aansturing door de DSP en het systeem als geheel. Vervolgens zijn ook de stoorsignalen geïdentificeerd aan de hand waarvan een eisenpakket voor de te ontwerpen controller is opgesteld.

Met deze gegevens is zowel een whitebox model als een blackbox model ontworpen waarmee een goed beeld van de systeem karakteristieken is verkregen. Aan de hand van deze modellen is een ontkoppeling voor het systeem berekend welke het ontwerpen van een regelaar die voldoet aan de gestelde eisen vereenvoudigd. Aan de hand van het whitebox model is een simulatiemodel gemaakt waarmee de ontworpen controllers kunnen worden getest en het systeem kan worden gesimuleerd alvorens deze in de proefopstelling te implementeren.

Gebruik makend van *loopshaping*-technieken is een controller ontworpen. Bij het ontwerp van de controller heeft een belangrijk gedeelte van de aandacht gelegen op de goede implementatie van de continue controllers die in Matlab zijn ontworpen in de C-code van de DSP. De uiteindelijke controllers voldoen niet aan de vooraf gestelde eisen, maar aanbevelingen om een aantal zaken te verbeteren en verder uit te zoeken, zijn gedaan.

Contents

Preface

This assignment was executed at Océ-Technologies B.V. in Venlo within the framework of an apprenticeship at the department of mechanical engineering of the University of Technology in Eindhoven. The assignment resulted from the GRT-group (Group Research Teams) of Océ that examines optimization of the print quality of the DIP-technology and therefore was performed in this group as well.

The original assignment consisted of two main parts; the identification of a system and the design of a model and a controller to solve a MIMO-positioning problem: A thin sleeve rotates around a magnetic core opposite to a rotating drum. These two cylinders have to be positioned parallel to each other with the help of two independent controllable actuators placed on either side of the cylinders. The resulting work was divided into three main parts: calibration, system identification and controller design.

In the first chapter the problem is explained and defined and the system is described. The calibration and linearization of this system is discussed in chapter 2, which is followed by the identification and modelling in chapter 3. In chapter 4 subsequently the design of a controller for the problem is discussed after which a conclusion and recommendations for further optimization are stated.

Besides the stated assignment, the goal was to become more familiar with working at Océ. A pleasant cooperation with the other members of the group and the daily working in this group were instructively and nice. Furthermore visits to various projects at the R&D-department and some other Océ departments like the toner plant and assembly departments gave a good impression of the company and its dimension.

Chapter 1

Introduction

1.1 Direct Imaging Process

For color printing, Océ developed a technique called 'Direct Imaging Process' (DIP). Several so-called Image Developing Units (Dutch: beeldvormende unit (BVU)) print monochrome color images on an intermediate-drum. This intermediate-drum transfers the image by high temperature and pressure onto the paper (see Figure 1.1). Each BVU consists of a Direct Imaging Process drum (DIP-drum), a feeder drum and a DIP-knife. The feeder drum develops a mono-layer of toner on the rotating DIP-drum which is removed again by the DIP-knife. If an image is printed, tracks in the DIP-drum can electrically charge or decharge independently, changing the magnetic/electronic force balance to keep the toner on the drum or to let it be removed by the DIP-knife. This way an image is developed on the DIP-drum (see Figure 1.2). Finally the monochrome color images are collected on the intermediate-drum, which transfers them onto the paper.

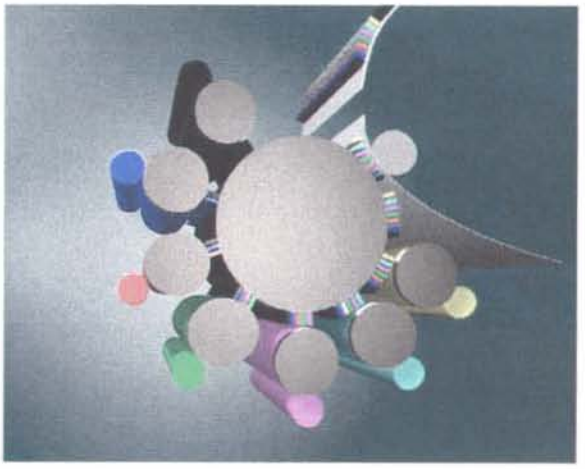

Figure 1.1: The DIP-setup with 7 DIP-units, each developing a monochrome color image onto the intermediate before it is transferred to the paper.

1.2 Print quality

The DIP-knife consists of a strong magnet that is fixated into a rotating sleeve. This sleeve rotates in opposite direction compared to the DIP-drum and removes the superfluous toner. The

distance, d, between the DIP-knife and the DIP-drum influences the balance between the electric and magnetic forces affecting the toner. Disturbances and mechanical errors such as non-roundness of the drums can make the distance, d, fluctuate. Simulations and tests show that disturbances greater than 1μm have a visible effect on the print quality. Therefore, a way to control this distance is proposed by radially moving the DIP-knife. An extra advantage is that disturbances of the print quality caused by other disturbances than the distorted or fluctuating distance d like fluctuations in the quality of the toner, can be compensated as well. To make radial movement of the DIP-knife possible, it is mounted onto the drum with two elastic hinges. These hinges allow independent radial movement of both sides of the knife, which therefore gains two degrees of freedom: a rotation and the before mentioned radial movement. Two actuators which are placed onto the hinges between the knife and the drum, control these movements. If the movement on both sides is controlled well, the print quality over the whole width of the drum can be adjusted online.

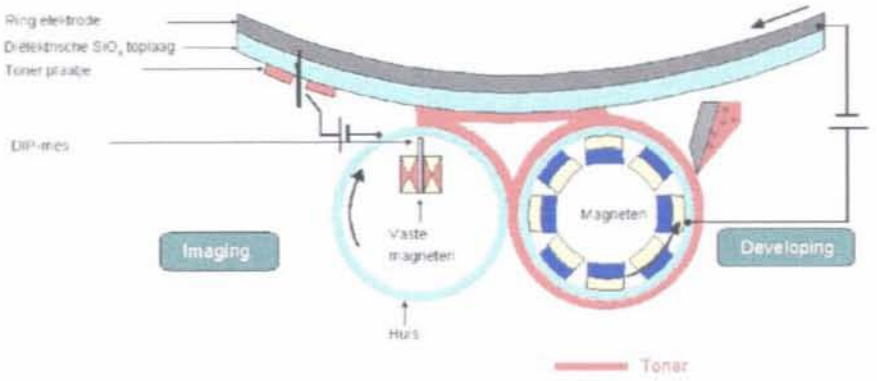

Figure 1.2: The DIP-knife (left cylinder), the feeder drum (right cylinder) and part of the DIP-drum onto which the monochrome image is printed.

1.3 Problem definition

A controller is needed to control both the left and right actuator position. This controller has to meet some demands on accuracy, speed and noise rejection. The accuracy has to be 1μm since larger disturbances have a visible effect on the print quality. The distance should be controllable at 3 Hz, because the rotational frequency of the sleeve can go up to 3 Hz and the controller should be able to correct for non-centricity of the drum or sleeve. The controller has to be stiff enough to reject disturbances like vibrations and surrounding noise. A first estimate for the values of these disturbances is 1 N up to 100 Hz. The disturbance signals will be discussed in section 3.2, the resulting demands to the controller are listed in section 4.1.2.

For better understanding of the system and for the design of a controller, a model of the system is useful. Therefore, the system is first identified after which a model is designed and tuned for further use in the design of the controller. Besides, a simulation model of the system, which can be used to simulate various controllers and configurations before implementing them, will be designed using Simulink. Before modelling the system, an identification and calibration of the sensors and actuators is done. For controller design, it is desirable to have a linear system, because plenty linear techniques are available. Therefore, if possible, small non-linearities in the sensors and actuators are compensated to get more desirable behaviour. This results in the following problem definition, consisting of three parts:

- Calibrate and if necessary, linearize sensors and actuators of the system. These calibrations will be implemented in the system giving the system a linear output.

- Design a model of the system with the help of a good identification of the system and measurements and build a simulation model. The modelling of the system will be done in a whitebox as well as a blackbox manner.
- Design a (MIMO-)controller that controls the distance d by actuating the two independent actuators on the hinges. The exact demands to this controller are listed in section 4.1.2.

Chapter 2

The system setup

A test frame is present of which the distance of the DIP-knife to the DIP-drum is variable by means of 2 elastic hinges on each side of the knife. Two independently controllable voicecoil actuators, which are fixed to the hinges, take care of the control (see Figure 2.1). A DSP regulates the control of the actuators and a PC running Matlab is used as interface and for the storage of data. Before the test frame was built, different alternatives were proposed and judged and during this assignment some alternatives were discussed as well. Parts of the test frame are calibrated and if necessary, linearised to get desirable linear behaviour.

2.1 Test frame

2.1.1 The test frame

The DIP-knife has two degrees of freedom, it can move radially (in x-direction) and because of the independent actuators on both sides rotate (around the y-axis). The sleeve rotates around the DIP-knife (around the z-axis). To position the DIP-knife, actuators are placed at both ends of the DIP-knife. The used voicecoil actuators consist of a housing connected to the frame and a coil connected to the two hinges. With the hinges the coils are connected to the DIP-knife with a transmission of $1:5$. The displacement of the actuators is measured by two foto-diodes. The diodes are partly shielded from the LED's by the moving actuators, so the position of the actuators influences the intensity of the light received by the diodes and their displacement can be measured.

To compensate for steady state gravity forces working on the actuators and the DIP-knife and for the stiffness in the hinges, a negative stiffness is constructed. This stiffness decreases the required force of the actuators and therefore it decreases their size and heat production. The negative stiffness is provided by two magnets placed in a U-formed frame at the outer sides of

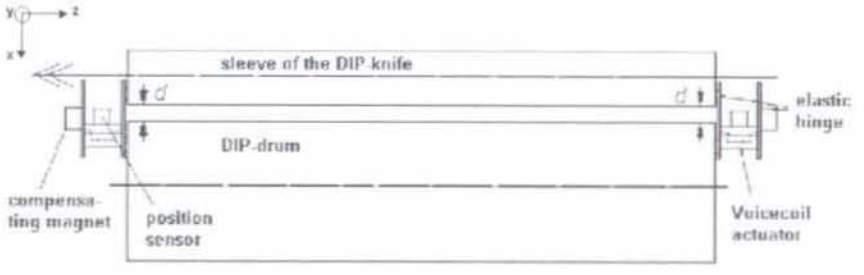

Figure 2.1: Schematic view of the test frame.

both elastic hinges. If a hinge moves, the magnet gets closer to the U-frame and its magnetic attraction leads to a negative stiffness. Both the actuators and the motors are controlled by a DSP, which is coupled to a PC. With a program loaded to the DSP and using Matlab as interface, the system can be controlled.

2.1.2 Voicecoil actuators

The choice to use voicecoil actuators is based on a list of demands: The actuators should be cheap and easy to implement. Furthermore they need to be strong enough to reject the disturbance forces, but just using more electric power is not the solution since that heats the system too much. The actuators need to follow a relatively slow setpoint with a maximum bandwidth of 3 Hz, but they should be stiff enough to reject the disturbances, which are of high frequency. Some options are:

Positioning using a spindle A great advantage of a spindle is its great stiffness between the two drums and resistance against disturbances. A disadvantage is its complexity making it too expensive, besides the fact it consists of many moving parts that can fail.

Position the knife by making the sleeve roll over the DIP-drum This appears to be impossible since too much toner is present around the drums, so the roll area will not stay clean enough. Furthermore the position can not be controlled, it is only better defined relatively to the DIP-drum than it is now. An advantage again is the high stiffness between the drums.

Positioning using voicecoil actuators An advantage of the voicecoil actuators is the lack of moving parts and the low cost price. Two disadvantages are the low provided maximum force and the fact that the stiffness is mainly defined by the controller. But the transition in the hinges between the actuator and the DIP-knife increase 5 times and therefore increases the controller stiffness perceived by the DIP-knife.

Since the disadvantages of the voicecoil actuators can be solved, these are chosen together with elastic hinges, containing a lever with a transmission of $1 : 5$, to control the DIP-knife. Some advantages of the voicecoil actuator are: linear dependency of the resulting forces on the input current, a low moving mass, high motor constants of the actuators can be achieved and no mechanical contact between moving parts, preventing friction and non-linear behaviour. But there are some disadvantages, which have to be taken into account: only small strokes are allowed, otherwise the resulting force will decrease and the stiffness and disturbance rejection fully depends on the controller.

2.1.3 Control by the DSP

The system is controlled using a DSP (TMS320LF2812 Texas Instruments). A PC communicates through a USB- and/or RS232 interface with the DSP, which performs the real-time control of the system and sends data to the PC. The data is buffered in a FIFO-buffer because the speed data is generated may be higher than the USB-bus can handle. The PC is only used for data storage and as User Interface for the system using Matlab-GUI. The DSP has many built-in hardware-options like A/D-conversions, digital I/O, counters, capturing, etc. The control output of the DSP, PWM (Puls Width Modulated) signals, is boosted and used as input for the voicecoil actuators. The sensor readings are directly coupled to the AD-converters of the DSP.

The DSP contains a Flash-memory of 256 kByte in which a library with standard control procedures is uploaded. When the system is started, a DSP-program is uploaded via RS232, which consists of a part that communicates with the PC and contains information about the used procedures and their interconnections, and a controller, which communicates with the system and calculates new setpoint signals. The program is written in C++ and uploaded using a C-compiler.

Every control step the sensor / encoder positions, the setpoint positions and the control signals, which are sent to the actuators are sent to the DSP-FIFO buffer, a data flow of 32 kByte/s. The

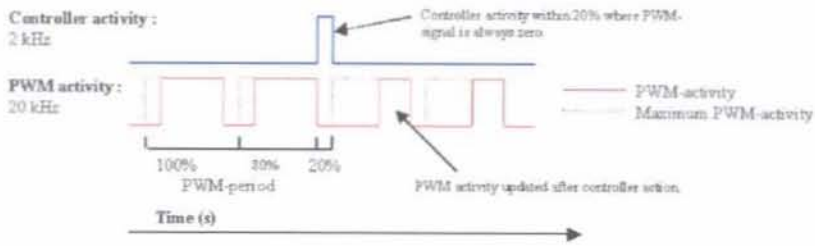

Figure 2.2: Controller activity in time.

actuators and the motors are controlled using 20 kHz PWM-signals: the DSP reads the AD-converters, calculates a new control output, sends this signal to the PWM-outputs and writes a log to the PC using the USB port. A new setpoint for the next time sample is calculated and then it starts waiting for a new interrupt.

The Controller runs at 2 kHz so every sample time contains 10 PWM-periods. The length of the pulses is controlled by the control output. To prevent the switching of the PWM-signals disturbing the measurements and AD-conversions, the PWM-periods length is limited to 80%. The AD-conversions are performed in the other 20% of the time, in which no switching of the PWM-signals occurs. The conversions are performed 8 times and averaged to suppress noise. The pulses of the encoders of the motors are counted and the falling and raising edges of the 8 most recent pulses are captured by hardware in the DSP. Therefore accurate speed estimation by interpolation between encoder pulses can be made.

The PWM-signals are controlled by a 12-bit integer having a range of -2047 to 2048. The range is limited from -1500 to 1500. After the AD-conversions, the controller calculates a control signal and updates the PWM. This results in a time delay between measuring and control output of least at one PWM-period ($1/(20\,\text{kHz}) = 0{,}0005\,\text{s}$) (See also Figure 2.2). The controller calculates its output using integers. This can give problems such as data loss or integer overflow it the control signals get too low or too high. A solution would be using floating point operations. A disadvantage is that this is a slower alternative because it takes more calculation time.

2.2 Calibration

From a control point of view, linear dynamic behaviour of the system is desirable. Furthermore, symmetric behaviour can be useful since the left and right controller can be the same. Therefore the system is calibrated and linearized if necessary and if possible. A resulting overview of input and output signals is made in which becomes clear how the signals evolve in the system (see Figure 2.7).

2.2.1 Position Sensor

The position sensor consists of a LED and a foto-diode. The actuator moves in between them leaving more or less LED-light exposed to the diodes. The voltage through the diodes is used to determine the position of the actuators. The analog signals from the diodes are converted to a 12-bit signal with a range of 0 to 4095. The position sensor is not linear: the real position versus the output signal of the sensor has a S-shape. Furthermore, the output signal drifts in time. Therefore the sensor has to be calibrated and a compensation for the non-linearity has to be implemented.

Calibration

To calibrate the position sensors, the known distance between the physical boundaries of the actuators is used, which is $1,5 \cdot 10^{-3}$ m. The output of the AD-converters varies from 0 to 4096 bit which results in a resolution of about $2,73 \cdot 10^{6}$ bit/m. To ease working with the signals, the resolution after calibration is set to $4,0 \cdot 10^{6}$ bit/m, which results in a range of 6000 bit over the total displacement of $1,5 \cdot 10^{-3}$ m. The actuators are moved from one stop to the other. The minimum sensor reading is used as origin and set to 0, the maximum sensor reading is set to 6000 bit.

Linearization

To identify and compensate the S-shape, a diagram is plotted with the real position versus the output signal. The following assumptions are made: the elastic hinges have linear stiffness and the voicecoils have linear input-output behaviour. Therefore, if the compensating magnets on the outer side of the hinges are removed, the real position is linear with the signal sent to the voicecoil actuators. So moving the actuators with a sawtooth setpoint, and plotting the actuator PWM-signal against the position reading, all non-linearities are caused by the position sensor. Those assumptions are not totally correct. The PWM-signal has a dead-zone around zero, which gives a step in the plots what is not caused by the position sensors. Furthermore some hysteresis is present.

With a least squares method a 5^{th}-order polynomial is fit on the resulting S-shaped output. (The step around zero and the hysteresis are not taken into account since these are not caused by the sensor). A 6^{th}-order fit did not have significant improvement anymore and therefore a 5^{th}-order fit is used. By multiplication of the sensor output with this polynoom, the corrected position reading becomes linear with the real position.

$$\begin{aligned} x_{actual} &= C(x_m) \\ &= C_1 x_m + C_2 x_m^2 + C_3 x_m^3 + C_4 x_m^4 + C_5 x_m^5 \end{aligned} \tag{2.1}$$

with x in pulses and x_m the output of the sensor before linearization.

Look-up table and sensor drift

A look-up table is used to implement the linearization, because this takes less online calcuation time than using equation 2.1 for all the output data. The look-up table with a range of 0 to 4096 is filled with the results of the linearizing polynoom, so for every sensor output a linearized output signal with a range of 0 to 6000 pulses is returned. The resulting output now becomes:

$$\begin{aligned} x_{lin} &= (C(x_m) - C(x_{m_{min}})) * \frac{6000}{(C(x_{m_{max}}) - C(x_{m_{min}}))} \\ &= lookup(x_m) \end{aligned} \tag{2.2}$$

with x_{lin} the resulting linearized output in a range of 0 to 6000 pulses.

The 5^{th}-order fit is performed once because it requires the removal of the compensating magnets. So the coefficients C_i of the polynoom are calculated once and saved. Because the output of the sensors drifts in time, the maximum and minimum output have to be calibrated frequently. Figure 2.4 shows the first two coeficients of the least squares fit. The exponential character of the plot implies that the coefficients converge to a final value, which is likely to be the result of some sort of heating. The first two coefficients, which indicate the boundaries and slope of the fit can be updated frequently by a calibration of the maximum and minimum sensor output. Therefore a calibration function is implemented in the C-code, which can be run easily in between 2 print jobs. The output range is determined and using the saved coefficients the look-up table is refilled. Figure 2.3 shows an example of the measured data together with the accompanied linearized data.

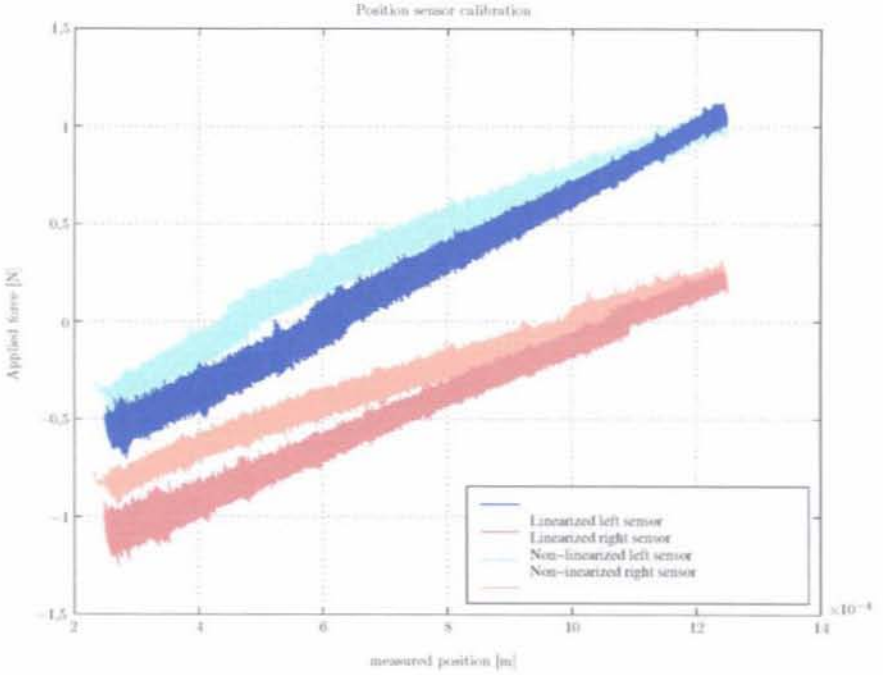

Figure 2.3: lineariseren

Conclusion

Before the compensating magnets are installed, a calibration has to be performed to compensate for the S-shaped input-output behaviour of the sensors from which 5 coefficients follow. After that the calibration has to be repeated frequently to recalibrate the upper and lower boundaries of the output signal because the sensor outputs drifts in time. Every time the calibration is performed the look-up table is refilled.

2.2.2 Hysteresis and dead zone

Because of magnetic charging of the iron, hysteresis is present and the S-shape becomes a loop. This charging is initiated by the movement of the coils, which induces a magnetic field and because the way the coils move varies all the time, the present hysteresis varies too. But hysteresis is some sort of energy dissipation, which is positive for the damping of the system and despite the hysteresis the correctness of the output will not be affected. Therefore some hysteresis is not bad and it is not taken into account furthermore.

The input PWM-signal can be on or off: -1, 0 or 1. If it switches from -1 to 1 or vice versa, it first switches to 0. So around 0 a dead zone is present in the output signal, which is initiated by the PWM input signal. The solution to this problem is to choose an equilibrium or working point for the actuators that lies far enough above or beneath this dead zone.

2.2.3 Elastic hinges

By introducing extra controller stiffness at one side, an attempt is made to equalise the stiffness and right elastic hinges to make the DIP-system more symmetric. The stiffness can be equalised but overall results are not very promising since different actuator masses etc also influence the symmetric dynamical behaviour. Futhermore including varying feedback can result in stability problems so this idea is not implemented in the final controller.

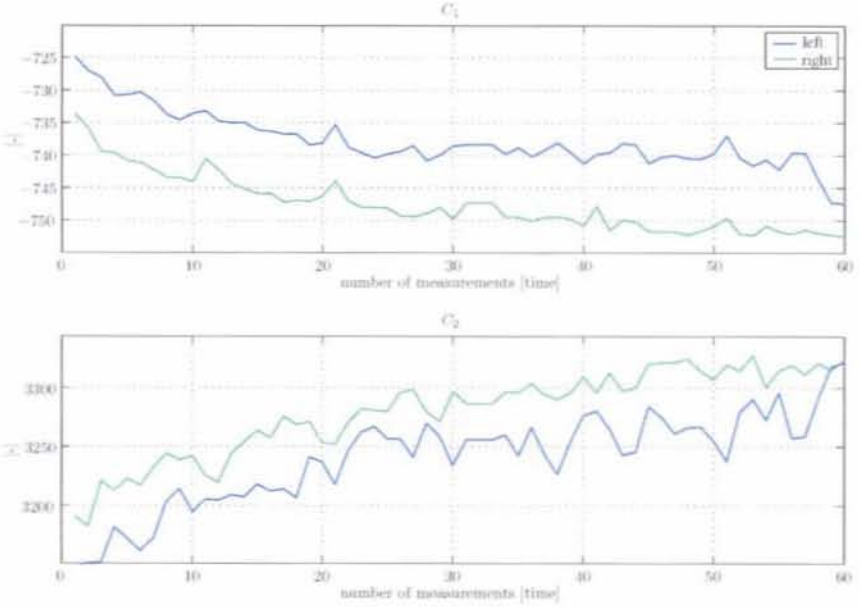

Figure 2.4: The first and second coefficient of the left and right fits resulting from calibrating the position sensor.

2.2.4 Voicecoil actuators

The voice-coil actuators are approximately linear. They are actuated by a 12 bit PWM-signal. The PWM-signal has a pulse-frequency of 20 kHz and a range of −2048 to 2047 bit, depending on the time it is 1 or -1, which is limited to −1500 to 1500 bit. This limits the output to 80% of its maximum so every PWM-cycle 20% of 20 kHz, $1 \cdot 10^{-5}$ s, is free of PWM-signals. In this time AD-conversions for the position measurements are performed.

The actuator forces are calibrated and the controller output is corrected before they are sent to the PWM-generator. Calibration showed that a PWM-signal of about 490 bit results in a force of 1 N. The controller output is multiplied by a correction factor which makes 500 bit results in 1 N (Sensitivity of $2 \cdot 10^{-3}$ N/bit). To determine the left and right correction factors some tests are performed. With a force sensor a known force is applied to the DIP-knife and the signal the controller needs to keep the DIP-knife in position is measured. Figure 2.5 shows the relation between the controller activity and the applied force.

Before the force sensor was available, some tests were performed using the mass of the DIP-knife. By turning the test frame, the gravity forces on the DIP-knife and the actuator mass change. Using formula 2.3 the effect of these changes on the actuator are calculated. So the force on the actuator caused by the tilt of the test frame is known and is compared to the controller activity to keep the DIP-knife into place. This results in 0,0021 and 0,0022 N/PWM-bit for respectively the left and right actuator. From these values a correction factor is calculated leaving 0,002 N/PWM-bit for both sides. For the left and right actuator these factors respectively are: 0,96 and 0,92.

$$F_g = (m_{act} \cdot 9{,}81 \cdot \cos\alpha - \gamma - m_{DIP} \cdot 9{,}81 \cdot \sin\alpha)\,/5 \tag{2.3}$$

With F_g the gravity force acting on the actuators, m_{act} the actuator mass, m_{DIP} the DIP-mass, α the angle of the test frame, γ the angle between the gravity vectors of the DIP-knife and the actuators.

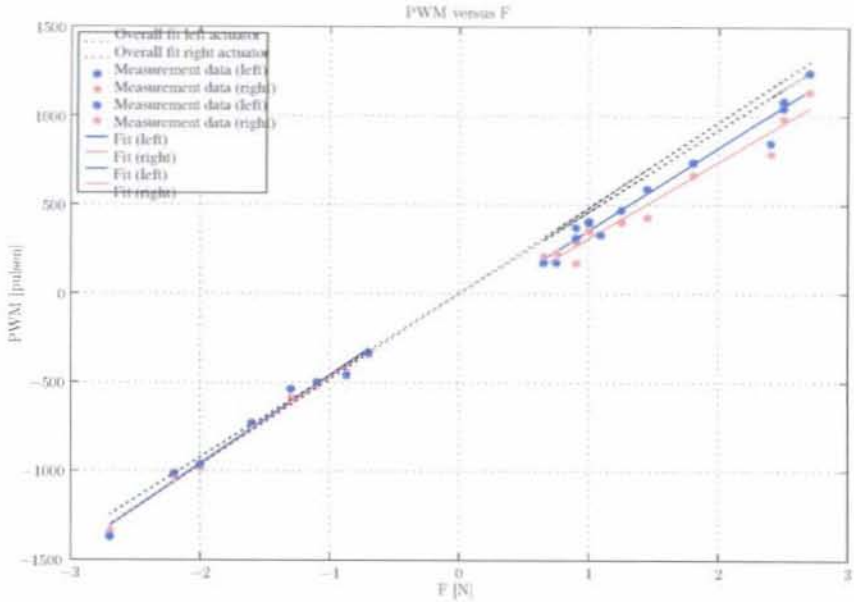

Figure 2.5: Calibration of the actuator motorconstants. Measurement data and first order fit.

2.2.5 Selfinductivity

During the system identification (Chapter 3) it is shown that around 300 Hz the angle of the measured frequency response drops to -270 deg. A possible cause may be the selfinductivity of the coils which causes a counter-EMK influencing the linear behaviour of the voicecoils. The selfinductivity of the coils is measured. This is done by placing the coil in a RC-network [10]. A sine-shaped voltage is applied at the network and the voltage across the coils is measured. The selfinductivity appeared to be too small to influence the system that much.

2.3 Linearity of the system

The linearity of the system is checked as well as it being constant in time. As explained in Chapter 3, the frequency response of the system is measured. This is repeated several times with different implemented controllers. If the system is linear, every time the same response is measured. These measurements are also repeated with the DIP-knife in different positions. Figure 2.6 shows the different frequency responses.

Clearly most responses are equal what implies that the system is linear. The only response that is different from the rest is a response that was measured when the actuators were just turned on. This implies that the system changes when it heats up. This is also the conclusion from the least squares fit on the position sensors (see section 2.2.1) where the coefficients of the fit drift in time.

2.4 System parameters

Before modelling the system, some masses and stifnesses are determined. The actuator stiffness is determined first. Since the actuator force and position signals are calibrated, force and position measurements at different actuator positions provide information about the actuator stiffness. This leads to a stiffness of $2{,}3 \cdot 10^4$ N/m.

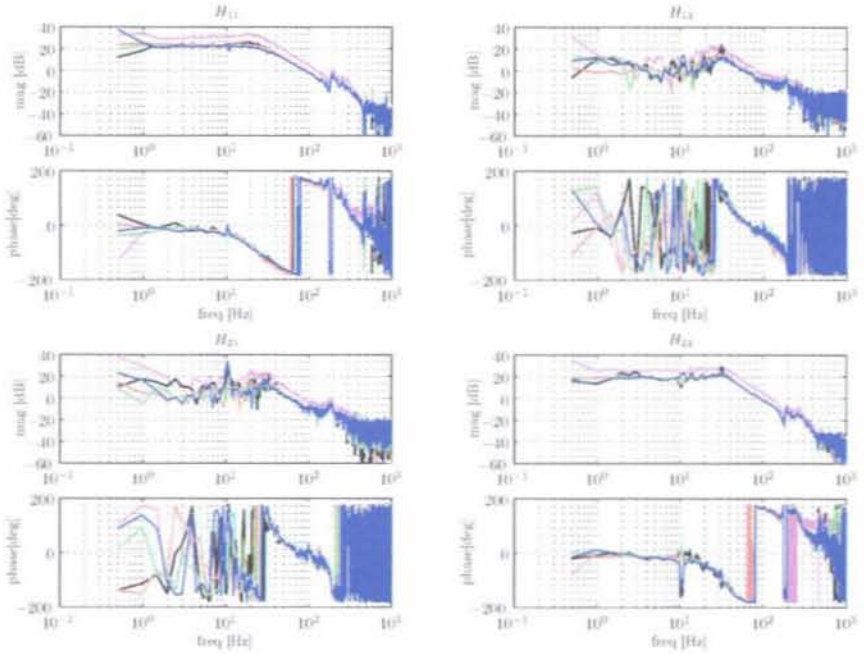

Figure 2.6: Linearity of the system, seven measurements of the same frequency response of the system.

The mass of the DIP-knife and actuators is already known but can be checked with the first resonance frequency. This is the resonance of the DIP-knife mass and actuator masses on the actuator stiffness. This resonance lies at $\omega = \sqrt{k_{act}/(m_{DIP}/i^2 + 2 \cdot m_{act})} = 160$ rad/s. With k_{act} (actuator stiffness) and i (transmission of the elastic hinge) known, the sum of the DIP-knife mass and actuator masses can be checked.

2.5 Conclusion

The designed system, calibrated and linearised. This results in a well defined linear system. To make the system linear and to calibrate the system different steps are taken which are described before. An overview of these steps is given in Figure 2.7. The measured position of the system is linearised and scaled using a lookup-table (see section 2.2.1). This results in a position signal transferred to the controller. To the control output an extra output can be added to correct for different actuator stiffness (see section 2.2.3). This output is transferred to a PWM-signal and send to the actuators (see section 2.2.4).

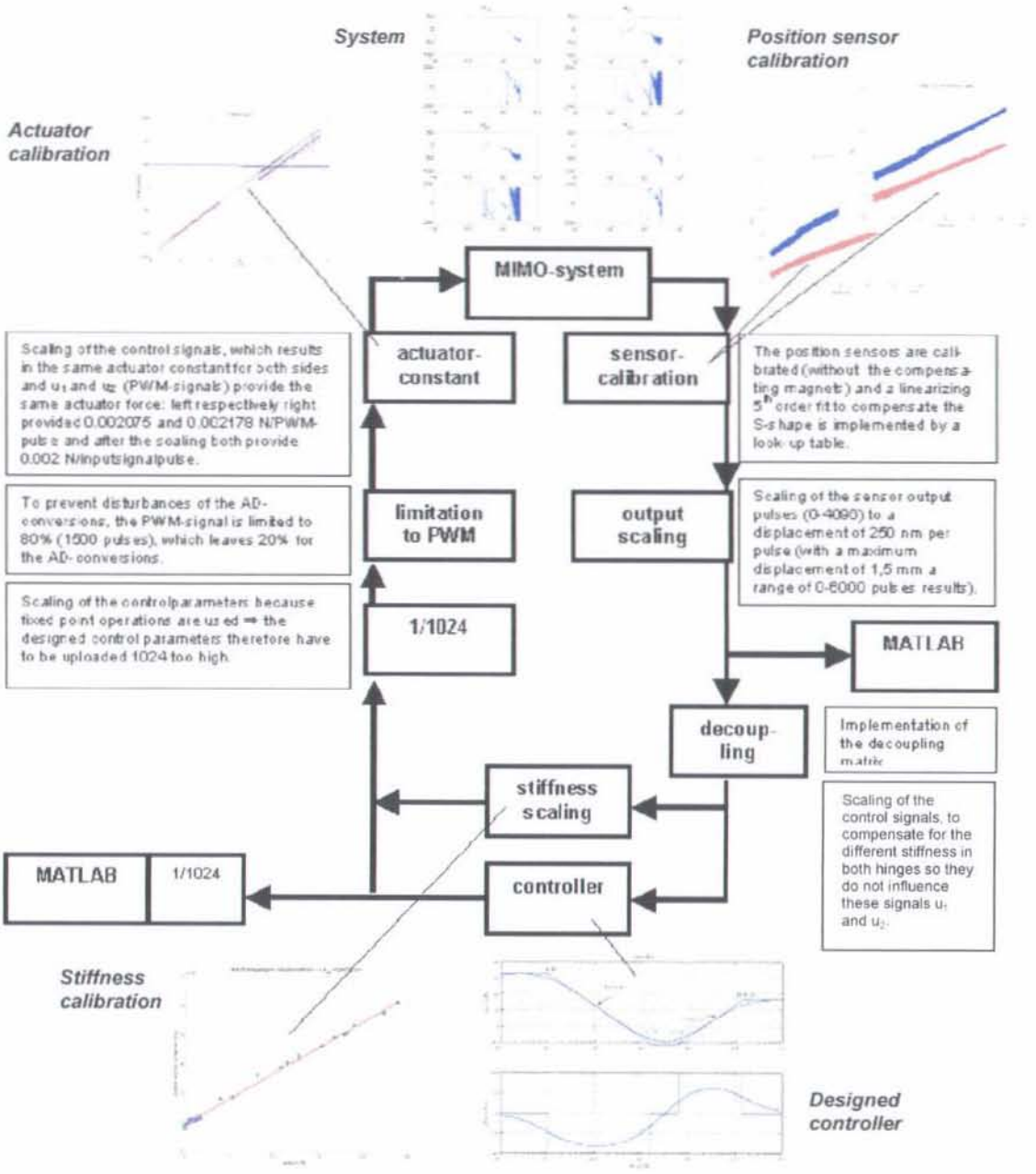

Figure 2.7: An overview of all calibrations and linearizations in the system.

Chapter 3

System Identification

3.1 Introduction to system identification

To gain more knowledge about the behaviour of the system, which can be used in the design of a controller, the system is identified and a model of the system is made. The models can be divided into two main groups: whitebox models and blackbox models. Of the test frame both a blackbox and a whitebox model is made. The whitebox model is used to get a better understanding of the system. The model shows the influence of parameters that also exist in the actual system so it can be used to gain knowledge about how to improve the system. The blackbox model is used in the design of the controller since it is a more accurate representation of the system but its parameters have no physical meaning.

The models are mainly tuned on measurement data in the frequency domain. After tuning, the models and the measured data are compared to judge the models. To retrieve the frequency response data of the system, the output-sensitivity is measured (See Figure 3.1). Since the system has 2 inputs and 2 outputs, the sensitivity is a 2x2 array. Using equation 3.2 the 2x2 frequency response of the system is calculated from the sensitivity.

$$S = \frac{1}{1 + HC} \tag{3.1}$$

$$H = (\mathrm{inv}(S) - I) \tag{3.2}$$

The whitebox model is designed in statespace notation using the variables listed in appendix A and the blackbox model is designed only using the transfer functions. Both models produce frequency response data using the same frequencies as used for the measurement data. The accuracy of the models depends partly on the size of the frequency range they describe. This range is determined by the needed bandwidth of the controller and because this depends on the present input signals, first an identification of these signals is performed.

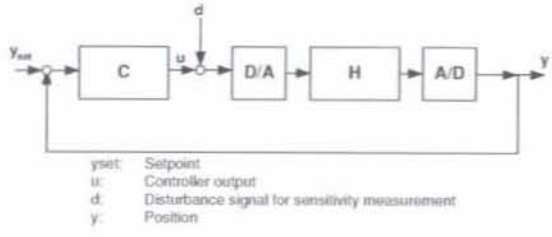

Figure 3.1: General control layout.

3.2 Identification of the disturbance signals

3.2.1 Disturbances

The bandwidth of the controller depends on the present disturbances and input signals. Because the frequency of the setpoint signals is significantly smaller than the maximum frequency of the disturbances the main goal of the controller is to reject disturbances. The main cause of the disturbances is vibrations resulting from the rotation of the sleeve of the DIP-knife, so the sleeve is turned on with a speed of 10000 mm/min, a frequency of about 1,3 Hz. The controller is turned off and the disturbance of the DIP-knife is measured. Using the frequency response of the system the forces initiating the measured disturbances are calculated. In the lower part of Figure 3.2 these disturbance forces are shown and as can be seen disturbance forces of about 2 N should be taken into account with this rotation frequency. If the rotation frequency is doubled, the size of these forces remains about 2 N.

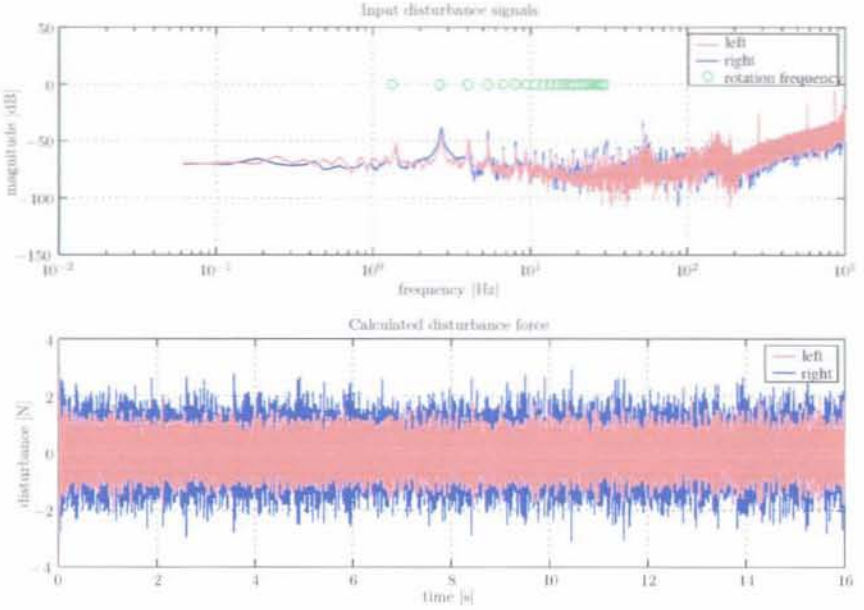

Figure 3.2: Input disturbance signals resulting from the drive of the sleeve in the frequency and time domain. The rotation speed of the sleeve is 10000 mm/min.

The data is transformed to the frequency domain. As can be seen in the upper part of Figure 3.2, peaks appear at the rotation frequency of the sleeve and higher harmonics of this frequency. Larger peaks such as the one just above 50 Hz may result from the gear of the sleeve drive.

To determine the contribution of the measurement noise in these disturbances, the measurements are repeated with fixed actuators. Since in that case the actuators cannot move, the measured position errors result completely from measurement noise. The result is an error of $\pm 1{,}3\mu$m, so the original demand of an accuracy of $\pm 1{,}0\mu$m cannot be achieved.

3.2.2 Cumulative error

The cumulative error is the total error up to a frequency that is initiated by the present disturbances like the drive of the sleeve. These disturbances are identified and therefore per frequency the addition to the total error can be calculated, which off course will resemble the previously calculated error per frequency. The frequency range where the biggest problems can be expected are determined and it is in this range the largest controller gain will be needed.

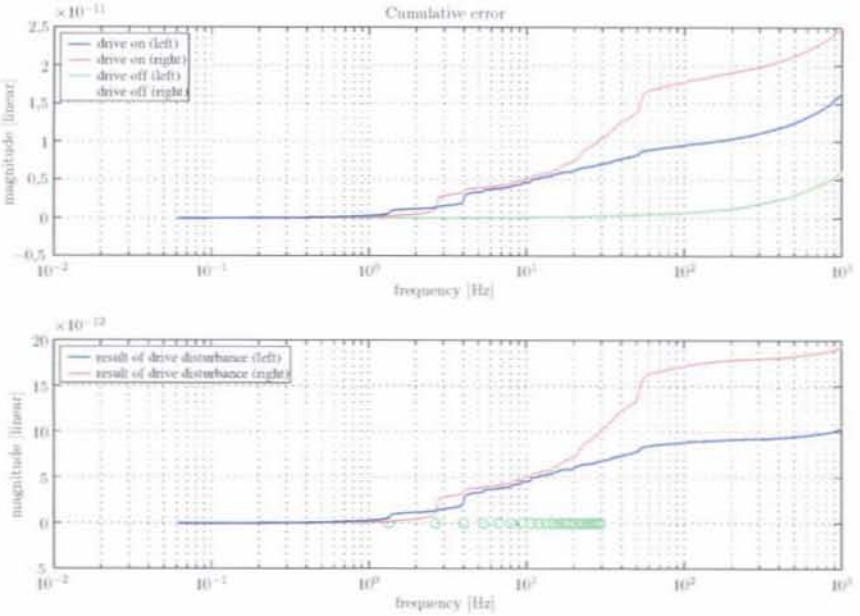

Figure 3.3: Cumulative error resulting from the input disturbance signals like the drive of the sleeve. The rotation speed of the sleeve is 10000 mm/min.

In the plot of this cumulative error, Figure 3.3, one can see that in the frequency range of 20 Hz to 110 Hz the largest addition to the total cumulative error is measured. This error is mainly initiated by the drive of the sleeve. In these tests again a frequency of 1,3 Hz is used. This frequency and some of its higher harmonics are again indicated with the green circles in the lower part of Figure 3.3. In the upper part of the figure also the cumulative error of the continuously present noise like sensor and measurement noise is shown with the green and yellow line. These disturbances are constant and have the same magnitude over the whole frequency range, which results in an exponential growth of the cumulative error on a logarithmic scale. In the lower part of the figure, the addition to the error by the drive is compensated for this constant noise. The same conclusions as in the previous section can be drawn, that the disturbances in the range of 20 Hz to 60 Hz have the most negative influence on the total error. The disturbances should be rejected by the controller, which therefore should have a minimal bandwidth of about 70 Hz to 100 Hz and for some frequencies extra gain is desirable.

3.3 Whitebox model

3.3.1 Model

For the whitebox model a 3D-model with dampers, springs and masses is designed, which represent the total system. In Figure 3.4 the 3D-model of the system, using dampers, springs and masses is shown.

The masses and moment of inertia of the DIP-knife, both actuators and the frame are taken into account. The stiffness and damping between these masses are modelled. The frame is attached to the rest of the machine by springs and dampers in the degrees of freedom of interest. The DIP-knife has two degrees of freedom: a displacement in x-direction, x_2, and a rotation around the y-axis, x_3. Both the actuators have a degree of freedom in the y-direction, x_4 and x_5. The frame is assumed to have two important degrees of freedom with respect to the rest of the machine; the displacements in x- and y-direction, x_1 and x_6. The inputs and outputs of the real system are

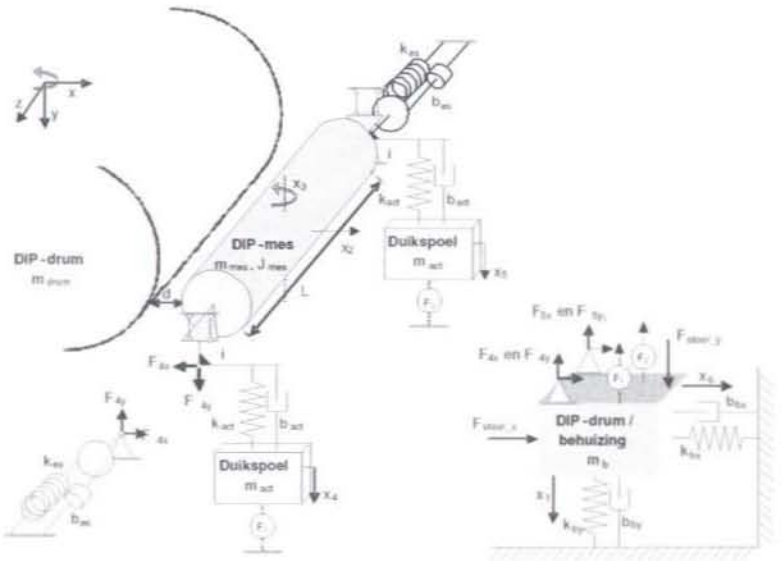

Figure 3.4: Overview of the 3D-whitebox model.

the input forces F_1 and F_2 and the displacement x_4 and x_5 of respectively the left and the right actuator. A complete overview of the used variables can be found in table B.1 in appendix B. In the same appendix the resulting equations, which describe the model are included.

The system is modelled as good as possible till 150 Hz, because the needed bandwidth for the controller is about 60 Hz to 100 Hz (see section 3.2). After the calibrations and identification of the system (see chapter 2) the electronics and sensors are assumed to be linear. So the counter-EMK in the voicecoil actuators for example is not taken into account into the model. Also internal degrees of freedom, which are assumed to be of much higher frequency than the before mentioned 100 Hz like bending of the DIP-knife aren't taken into account.

3.3.2 Parameter optimisation

For the parameter optimisation initially two models were used: a symmetric model with the same parameter values on both sides and a non-symmetric model with different parameter values. Soon already the symmetric model appeared not to have enough degrees of freedom to fit a good model of the system, so the actual parameter optimisation is performed using the non-symmetric model. The main optimisation is performed in the frequency domain. An optimisation in the time domain was tested as well, but the results were not that good. This optimisation probably will succeed if more attention and time is given to, but in this case the optimisation in the frequency domain was good enough already, so the optimisation in the time domain, based on step responses, was not paid enough attention to.

For the optimisation of the parameters in the frequency domain, initially the values of the system parameters are estimated by the values found in literature and measured during the calibration and identification of the system (see appendix A). These initial values are varied and by examining the frequency response of the model the influence of varying the different parameters is traced (see Figure 3.5). Using this information and common knowledge of the influence of masses, springs and dampers on a frequency response, the parameters are tuned to match the model on the measured frequency response data, see Figure 3.6. A reasonably good model is obtained of which the parameter values are listed in table B.1 of appendix B.

Secondly, the model is tuned using a minimisation algorithm (the Matlab built-in tool *fmins*) to minimise the error between the model and the measured data. First all parameters of the non-

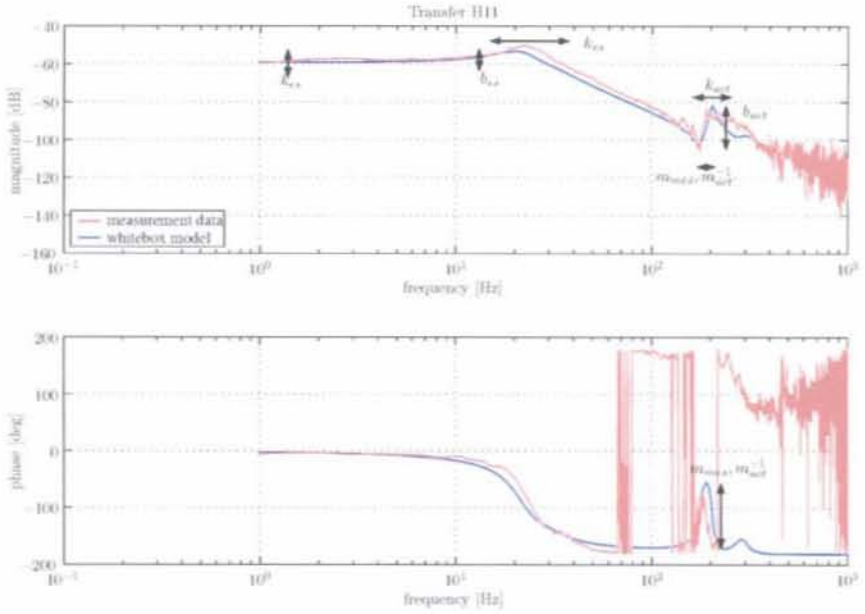

Figure 3.5: Influences on H_{11}, the transfer of F_1 to x_4, of the parameters used in the whitebox model.

symmetric model were estimated by the algorithm. The responses of the resulting model resemble the responses of the measurement data very well as can be seen in Figure 3.6, but the estimated parameters are physically impossible (see appendix B). Some parameters with very deviating values appear to have little effect on the total transfer, like k_{by} and therefore they may be doubted on. But other parameters with deviating values have significant influence on the behaviour of the system and the model, like the length L of the DIP-drum and the damping $b_{es4,5}$. Therefore using these parameters, the model can only be used for loopshaping and not for further design of model-based controllers.

3.4 Blackbox model

A blackbox model can always be fit, even if none of the characteristics of the system are understood since the parameters of the resulting model aren't based on physical interpretation of the system's behaviour. Using a minimisation algorithm, the four parts of the 2x2 frequency response data are fitted separately. As a minimisation algorithm, the frfit-tool from the Diet-toolbox of Matlab [6] is used.

The order of the fit is determined by the number of (anti)resonances that is taken into account. If a model-based controller is designed a good fit / good model over a range up to above the bandwidth is needed, but the better the model, the higher the order of the model and the higher the order of the model, the more complex the resulting controller will be. A start is made for example with the design of a H_∞-controller. The order of the resulting controller will be equal to the order of the used model plus the order of the weighting and shape filters. For this purpose first a controller of reasonably low order was fit, but even then the controller became too complex (see appendix C). Because further on the attention of the controller design stayed with loopshaping the order of the model is not very important since the order of the controller does not depend directly of the order of the model. So a good model of 16^{th} order was fit. The result is a model, which is good up to 300 Hz, see Figure 3.7.

In the final system a decoupling is implemented, so the responses of the system are changed

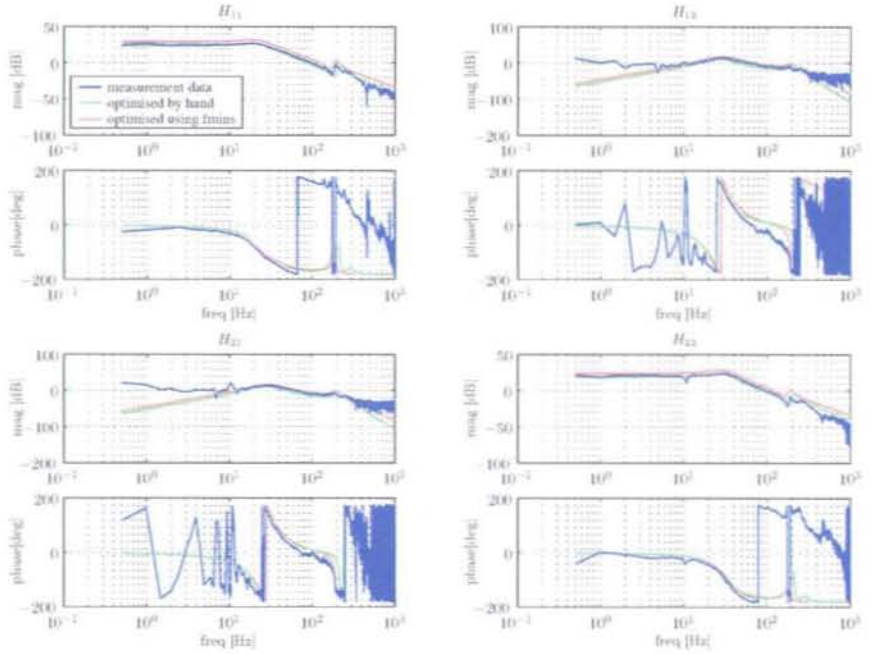

Figure 3.6: Whitebox model vs measurement data. Whitebox model with parameters optimised by hand, using the physical impact of the parameters, and using the minimization algorithm *fmins*.

(see section 3.5). Because this decoupling varied during the design process, no blackbox model of the final decoupled system is designed. The non-decoupled model can be decoupled in the same way the system is, but model-errors may be boosted and the result will not be an optimal model of the decoupled system.

3.5 Decoupling

Because of the 2 inputs and 2 outputs of the system, it consists of 4 transfer-functions (see for example Figure 3.6). The 2 diagonal terms are the direct transfers of the input on one side of the system to the output on that same side. The other 2 terms, the cross terms, can be interpreted as responses that are felt on one side of the system if the other side of the system is actuated. This interaction is relatively large in this case as can be seen in Figure 3.6: the amplitudes of the cross terms are of the same order as those of the diagonal terms. Control actions boost these responses, which can cause instabilities if they become too high. Therefore, a decoupling that decreases this interaction would simplify the control problem. If a good decoupling is found, the control problem focuses on the diagonal elements and instead of a MIMO-control problem, a kind of 2-SISO-control problem remains.

To verify if a system is decoupled well a designed diagonal controller can be implemented, the MIMO-sensitivities calculated and the diagonal sensitivities compared with the sensitivities of a system in which the same controller is implemented but H_{12} and H_{21} are considered to be 0. If the diagonal sensitivities do not differ substantially, the conclusion may be drawn the cross terms do not have significant influence on the decoupled diagonal terms.

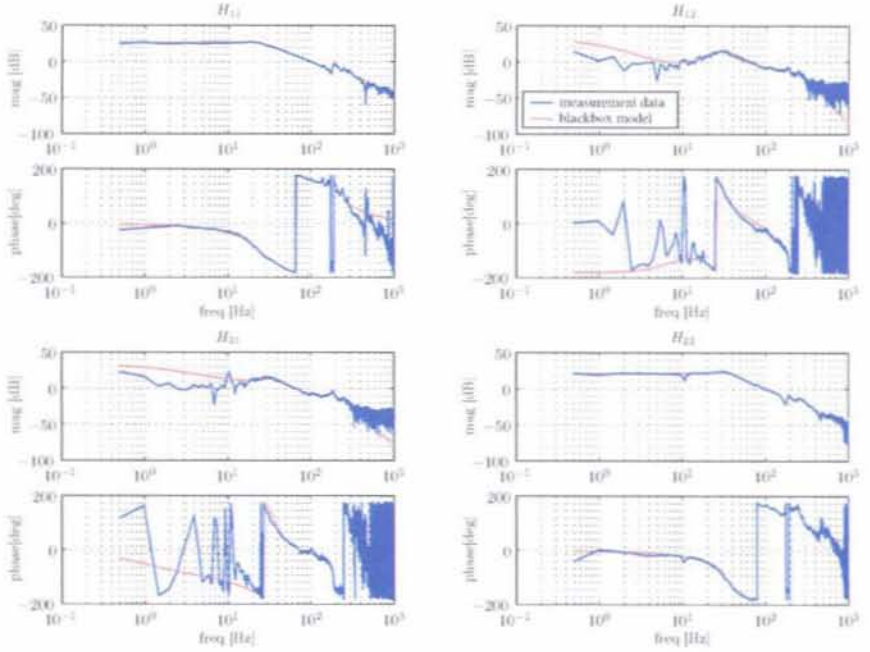

Figure 3.7: Blackbox model vs measurement data.

3.5.1 Force-momentum decoupling

At first sight, a decoupling of the rotation and the radial displacement of the DIP-knife, respectively the states x_3 and x_2 of the whitebox model, seemed a good solution to the decoupling problem. This decoupling can only be obtained if the system is symmetric. If the system is completely symmetric, a pure displacement in x_2-direction is obtained if the input signals of both voice coil actuators are the same and a pure rotation (in x_3-direction) is obtained if the input signals of both voice coil actuators are equally large but of opposite sign. In this way every position can be reached and the measure of interaction is limited to the measure of symmetry (and equality of the input signals). Using the same definitions as in section 3.3 this decoupling is calculated as follows:

The system uses F_1 and F_2 as inputs and x_4 and x_5 as outputs:

$$\begin{bmatrix} x_4 \\ x_5 \end{bmatrix} = H \cdot \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} \tag{3.3}$$

the decoupled system uses F and M as inputs and x_2 and x_3 as outputs:

$$\begin{bmatrix} x_2 \\ x_3 \end{bmatrix} = H_{ont} \cdot \begin{bmatrix} F \\ M \end{bmatrix} \tag{3.4}$$

To decouple the system, F_1 and F_2 are calculated as function of F and M and x_2 and x_3 are calculated as function of x_4 and x_5:

$$\begin{bmatrix} x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1/(2 \cdot i) & 1/(2 \cdot i) \\ 1/(L \cdot i) & -1/(L \cdot i) \end{bmatrix} \cdot \begin{bmatrix} x_4 \\ x_5 \end{bmatrix} \tag{3.5}$$

in which L is the length of the DIP-knife and i the transition size in the hinges.

$$\begin{bmatrix} F_1 \\ F_2 \end{bmatrix} = \begin{bmatrix} 1/2 & 1/L \\ 1/2 & -1/L \end{bmatrix} \cdot \begin{bmatrix} F \\ M \end{bmatrix} \tag{3.6}$$

Substituting 3.5 and 3.6 in 3.3 and 3.4 results in

$$H_{ont} = \begin{bmatrix} 1/(2 \cdot i) & 1/(2 \cdot i) \\ 1/(L \cdot i) & -1/(L \cdot i) \end{bmatrix} \cdot H \cdot \begin{bmatrix} 1/2 & 1/L \\ 1/2 & -1/L \end{bmatrix} \tag{3.7}$$

which leaves the decoupled system H_{ont}.

Theoretically, the system now should be decoupled reasonably well. To check this, the amplitudes of the diagonal terms with those of the cross terms are compared: $(H_{11} \cdot H_{22})/(H_{12} \cdot H_{21})$. The larger this coefficient, the better the decoupling: if both diagonal terms are twice as big (in amplitude) as the cross terms, this coefficient equals 4 so a coefficient larger than 4 is considered good.

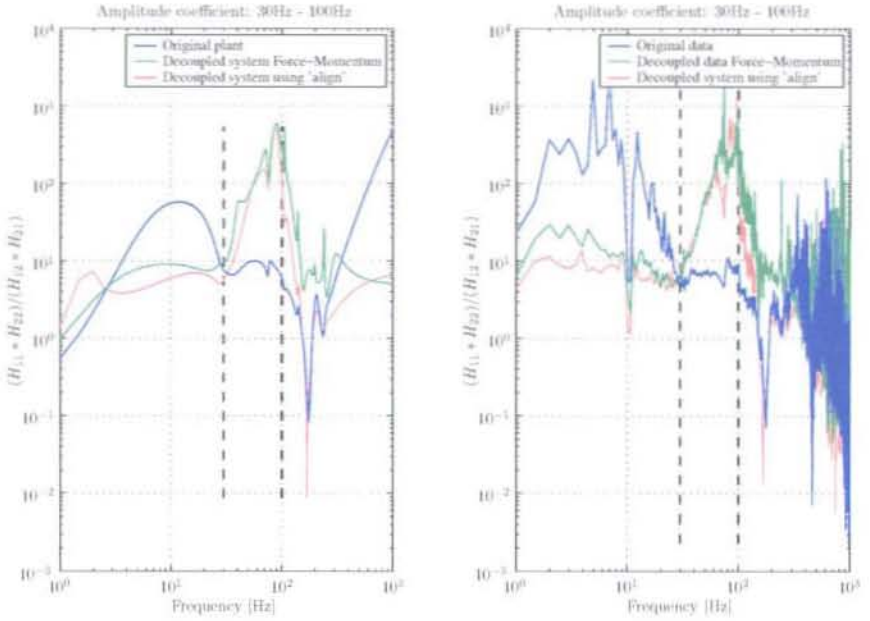

Figure 3.8: Magnitude of the diagonal terms vs the cross terms after decoupling to x_2 and x_3 and decoupling in a frequency range of 20 to 100 Hz using *align*.

In Figure 3.8 this frequency-dependent coefficient is shown compared to the coefficient of the original non-decoupled blackbox model and the measurement data. As can be seen, as soon as the dynamics of the system become more significant than the steady state mechanics, the decoupling has a positive effect and therefore this decoupling seems reasonably good. But if looked at the 4 decoupled responses of the system compared with those of the original non-decoupled system (see Figure 3.9), the amplitudes of the cross terms are still relatively large with respect to the transfer of force as input to the x_2-displacement as output.

The transfer of the momentum as input to the rotation in x_3-direction on the other hand, is large, which explains the large coefficient. An additional problem in this case is that the amplitudes of the force and momentum with respect to each other can't simply be compared one on one, because they have different quantities. A scaling with $\begin{bmatrix} 1 & 0 \\ 0 & \alpha \end{bmatrix}$ is logical: this doesn't affect the amplitude coefficient and the magnitude of H_{21} with respect to H_{11} doesn't change either.

The question arises whether this is the most optimal decoupling, because the assumption the system is symmetric may not be completely legal. To check this, first a rotation matrix of the

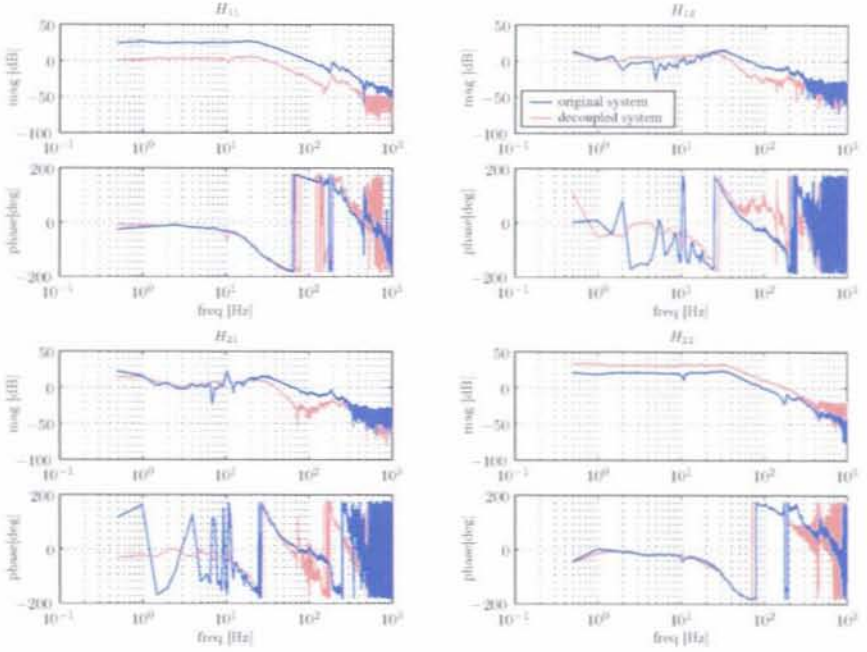

Figure 3.9: The system decoupled using force and momentum as inputs and x_2 and x_3 as outputs.

form

$$R = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \tag{3.8}$$

is added

$$\begin{bmatrix} x_2 \\ x_3 \end{bmatrix} = H_{out} \cdot R \cdot \begin{bmatrix} F \\ M \end{bmatrix} \tag{3.9}$$

which has to rotate the force and momentum inputs in such a way a better decoupling is found and the amplitude responses of the cross terms are minimized. In practice however, simulations with actual response data showed the optimal result for $\theta = 0$. This would mean the decoupling proposed in equation 3.9 is optimal. The standard method to decouple a system, which will be discussed in the next section, is also tested and eventually used because of its ability to specify a frequency-range where the system should be decoupled the best. Besides, the amplitudes of the cross terms with respect to those of the diagonal terms are in both cases smaller instead of one diagonal element with large amplitude.

3.5.2 The 'ideal' amplitude-based decoupling

In theory the ideal decoupling matrix K with which only the diagonal elements of the decoupled system influence the outputs can easily be calculated by stating $H_{out} = K \cdot H = I$, which leads to $K = H^{-1}$. In this way a frequency-dependent 3D-decoupling matrix is calculated. In practice however this can't be implemented and therefore the decoupling matrices over a specific frequency range are averaged and then used for every frequency. The decoupling matrix may only contain real numbers, which means that only the amplitude of the response H_{ii} is taken into account. Therefore a frequency range in which the phases of the 4 responses are the same or a multiple of 180^0 with respect to each other, has to be taken to calculate the accompanying frequency-dependent decoupling matrices. A phase rotation of 180^0 can be taken into account by a multiplication of

the amplitude with -1 for that response. Another option is to make use of the tool *align* [7]. This tool calculates a decoupling matrix on the same principle, but does take into account the differences between the phases of the responses and therefore the calculation isn't restricted to the previous mentioned frequency ranges.

In Figure 3.8 the same coefficient as mentioned in the previous section is shown for a decoupling based on a frequency range of 20 Hz to 100 Hz. Besides the fact the decoupling in the specified frequency range is good, one can see that the result of this decoupling resembles the result of the force-momentum decoupling of the previous section for a great deal. In the range of about 130 Hz to 300 Hz the force-momentum decoupling is even better from which may be concluded the system is symmetric enough to decouple the rotation and radial displacement with the force and momentum as inputs. Both the force-momentum decoupling and the amplitude-based averaged decoupling matrix using the align-tool are implemented, but for further use in the design of the controller, the latter one is used as mentioned before.

Before the effects of the various decoupling matrices were compared using the amplitude coefficients, this was done by looking at the 4 decoupled responses. And because of the ability of the align-tool to optimise the decoupling in a specific frequency range this seemed the best decoupling to use for the design of the controller. If the controller is optimised furthermore in future, a further study to gain better understanding in the use of the force-momentum decoupling may be performed as well.

Because the decoupling using the *align*-tool is based on a specific frequency range, its effect will be best in that range. To determine in which frequency range the decoupling has the most positive influence for the controller to be designed, the cumulative error of section 3.2 is used. Because of the large growth of the cumulative error in the frequency range of 20 Hz to 110 Hz and the peak at 50 Hz, the decoupling is optimised for that part of the total frequency range, see Figure 3.10. The resulting decoupling matrix is $K = \begin{bmatrix} 1{,}0 & 0{,}35 \\ 0{,}35 & 0{,}68 \end{bmatrix}$, which is implemented in the C-code. Other matrices can be calculated and evaluated easily (see appendix D).

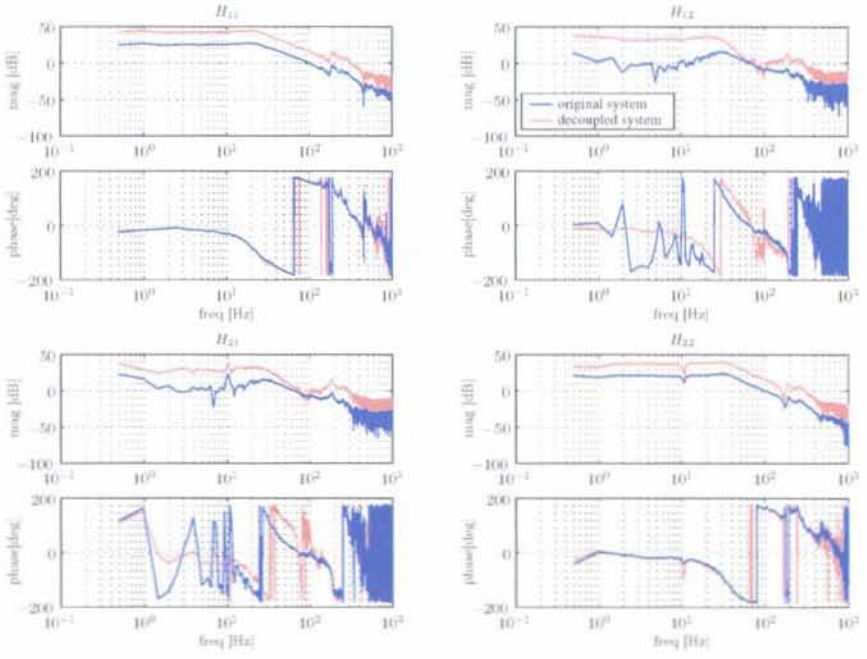

Figure 3.10: The system decoupled using the *align*-tool in a frequency range of 30 Hz to 100 Hz.

Chapter 4

Controller Design

To design a controller for the discussed 2x2 MIMO-system, various strategies are possible [5,9]. If both inputs and outputs have only little interaction, the system can be treated as a pair of independent SISO-systems. If interaction does exists, an attempt can be made to decouple the system to a system with more or less independent states. If the system cannot be decoupled a controller should be designed using MIMO-techniques such as sequential loopclosing or more complex control techniques such as robust control (H_∞), learning control or model predictive control (MPC). Some of these techniques are tested and discussed below.

4.1 Control setup

4.1.1 Design strategies

Since significant interaction exists between left and right actuator (see section 3.5), first an attempt is made to decouple the system, treat as 2 SISO-systems and design a controller using loopshaping techniques (see section 3.5). The controllers are tuned using a model of the system and with measurement data. The controllers are transformed to the discrete domain and loaded into the program in the DSP. A special test algorithm (see appendix D) measures the actual transfer of the uploaded controller and compares it to the designed controller. If the controller is loaded, it is tested in practice by verifying the closed-loop behaviour.

If the decoupling is not sufficient, a MIMO-controller can be tuned by means of sequential loopclosing: one input-output transfer is considered, for example H_{11} and a controller, C_{11}, is designed for it. The controller C_{12} and C_{21} are considered to be zero. Next the controller C_{22} is tuned for the next input-output transfer, in which the controller C_{11} is considered to be part of the system and therefore also influencing this next transfer. So H_{22} becomes H_{22}^*, which is a combination of H_{22} and the closed-loop behaviour of the first control loop:

$$H_{22}^* = H_{22} + \frac{H_{21} \cdot H_{12} \cdot C_{11}}{1 - C_{11} \cdot H_{11}} \tag{4.1}$$

Theoretically, this can be extended to the cross terms of the system.

Parallel to the loopshaping techniques the possibility of robust control (H_∞) has been examined since this technique is especially good with MIMO-control problems and model uncertainties. In this case a controller has to be designed for a test frame, which has to be implemented in many other systems and therefore system variations and uncertainties will be present. The order of the resulting H_∞-controller depends on the order of the model and the used weighting and shape-filters. In this case the model is of reasonably high order so a reduced model of the system has to be used as a result of which characteristic resonances have to be ignored and the controller may not be the optimal, so this strategy is not implemented. Appendix C describes the start of the design of the H_∞-controller.

The main control aim is to suppress disturbances (see section 4.1.2). The rotating DIP-drum and sleeve of the DIP-knife cause most of the noise. If the noise has a repetitive character, it may be possible to suppress some of it with a feedforward term. This feedforward term should be tuned using a learning algorithm. Because the radial counters of the sleeve and DIP-drum do not measure their absolute rotation and therefore the time the input disturbances are initiated is not known. Otherwise the feedforward terms could be tuned once during an initial calibration. Because of this learning algorithm, it will be too slow in this case. Also techniques like MPC are too slow because of the required calculation time.

4.1.2 Control demands

In section 1.3 is already explained the controller should position the DIP-knife with an accuracy of 1μm up to 3 Hz and from section 3.2 can be concluded the original demand of disturbance rejection up to 100 Hz remains, but with extra attention to the frequency band around 50 Hz and the higher harmonics of the rotation frequencies of the sleeve and the DIP-drum. Besides the mean disturbance force is 2 N instead of the original 1 N. To be able to implement enough damping to create phase margin up to 100 Hz, the sampling frequency of the controller, 2 kHz, is chosen 20 times higher than the bandwidth. For disturbance rejection the controller gain should be high enough to keep the error below 1μm with a disturbance force of 2 N. This results in a controller stiffness of 2000 N/mm. The concluding demands to the controller are:

- The maximum frequency of the setpoint signal y_{set} will be 3 Hz.
- The maximum error perturbed by disturbance signals, which are about 2 N may lead to an maximum error of 1μm. This is stated for a frequency range up to 100 Hz because the main disturbances are initiated in this range.
- Within 10 ms, the effects of a step disturbance has to be compensated for 90%.

4.2 Controller design

4.2.1 Controller design using loopshaping techniques

For the design of a controller using loopshaping techniques the method depicted in scheme 4.1 is used. A controller is designed in the frequency domain using components like PID-controllers, notch filters and lowpass filters. For this purpose the tool *MCD* (see appendix D) is used in which the controller can be designed and the effects of various controller designs can easily be evaluated by multiplication of the controllers with the MIMO-transferfunctions. Two SISO-controllers are designed in the frequency domain, assuming the decoupling is well enough. If desired sequential loopclosing can be used in this design process. After a good controller is designed it is transposed to the z-domain and with the same tool the effects of the resulting controller can be verified with the help of calculated open-loop responses, sensitivities and Nyquist plots of the MIMO-system.

The left and right direct transferfunctions resemble a lot and therefore initially the design of the controllers is the same for both. Initially two PD-controllers are designed with enough phase margin around the bandwidth and a gain as high as possible. The resonances at about 200 Hz appeared to be a problem in increasing the gain furthermore.

To suppress these resonances a notch is designed. A pair of two identical notches in the range of 175 Hz up to 190 Hz give the best result and the gain can be increased further. Also an I-action is added in the range of 0,1 Hz up to 15 Hz to increase the gain in the low frequency ranges. This is desirable because of the low rotation frequencies of the sleeve and the DIP-drum and the control demand of a setpoint with a maximum frequency of 3 Hz. The phase decreases by this action. The damping is increased (zero from 60 Hz to 70 Hz) and the gains of the two controllers are increased as much as possible, again taking into account the phase margin. The result is a controller with a bandwidth of 60 Hz respectively 70 Hz on the left and the right side and a steady state openloop

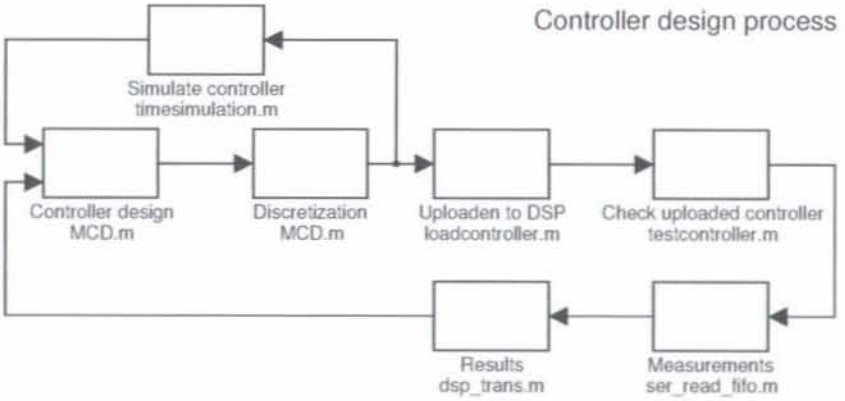

Figure 4.1: Design and implementation of the controller.

gain of about 40 dB on both sides. In Figure 4.2 the original transfer H_{11}, the designed controller and the resulting openloop response are plotted.

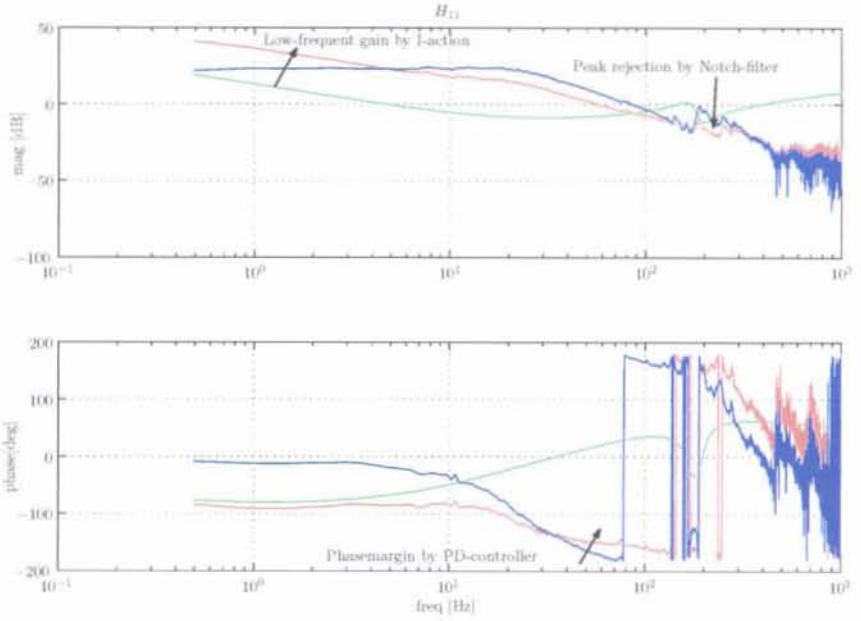

Figure 4.2: The original transfer H_{11} (blue), the implemented controller (green) and the resulting openloop response (red).

Since the identification of the disturbances showed a peak around 50 Hz an attempt is made to design a notch filter in this frequency range to increase the gain locally. When testing the notch in the actual system however, the steady-state controller gain seemed to have decreased dramatically. The reason for this decrease is unknown, but data loss due to the use of integers instead of floating-point operations is a possibility (see section 4.4).

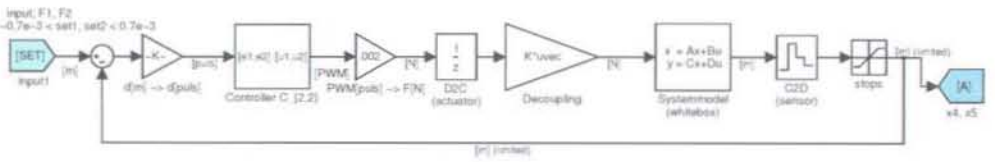

Figure 4.3: Simulation model to simulate time responses on the modelled system.

4.2.2 Simulation model

To check whether a designed controller performs as desired, a simulation model is designed in which the controllers can be checked before uploading it to the DSP. The simulation model is based on the whitebox model of section 3.3, the calibrating and linearizing steps of chapter 2 and the designed controller, which is uploaded directly out of the controller design-tool *MCD*. Implementation of the blackbox model causes instabilities, which are initiated by a wrong phase in the low-frequent range. Part of the simulation model is shown in Figure 4.3. The model is not used a lot because the tools to directly test and upload the designed controllers are very efficient.

4.3 Implementation of the controller

The designed controller is uploaded to the DSP. Before the controller is tested with the system, it is checked by measuring the transfer of the input to the output of the controller using a noise-alike signal as input. The resulting response is compared to the uploaded discrete controller and in this way it is checked if the controller is uploaded well. If the controller is uploaded well, it is tested with the system. Using a rotating sleeve and a constant setpoint signal as input the position of the actuators is measured resulting in an indication of the error.

The sensitivity is determined using a known noise-alike signal as input. The openloop response of the system is calculated, which is compared to the previously designed response. The time response can also be used to compare the resulting response with the designed response using a simulation model of the system. The gain of the system can be increased online using lower or higher voltages for the voicecoil actuators. In this way and with the help of the differences the controller can be optimised.

During implementation of the controllers, some unexpected stability problems appeared. A possible explanation of this problem may be the fact that the control algorithm uses fixed point integers instead of floating point operations to calculate the controller signal. These integers can overflow if the controller parameters are too big, or can suffer from data loss if the parameters are too small. A solution may be to use floating point operations instead of integers. To check this, different parts of the controller are tested separately. The frequency response of the implemented controllers is measured and compared to the designed controller. It appears that a PD-controller and a notch-filter can be implemented very well to a certain point. I-action is also possible to a certain point: too high low-frequent gain causes an integer overflow, which makes the control algorithm unstable. Just like a too less damped high-frequent notch-filter, which causes integer overflow as well. The result of an integer overflow is the control output switching from maximum to minimum if the output arises above its maximum. Figure 4.4 shows three controllers: an analog controller, its discrete version and the measured implemented controller. As can be seen, if the controller becomes too complicated, and therefore the integers too high, the implementation of the controller fails.

Therefore the finally implemented controllers do not include the suggested notches. One of the final controllers is shown in Figure 4.5.

Another difficulty in increasing the controller gain are high frequent non-linearities, which have not been taken into account. When the controller gain and damping are increased further some parts of the system, such as platework etc, start resonating. The first part that started resonating

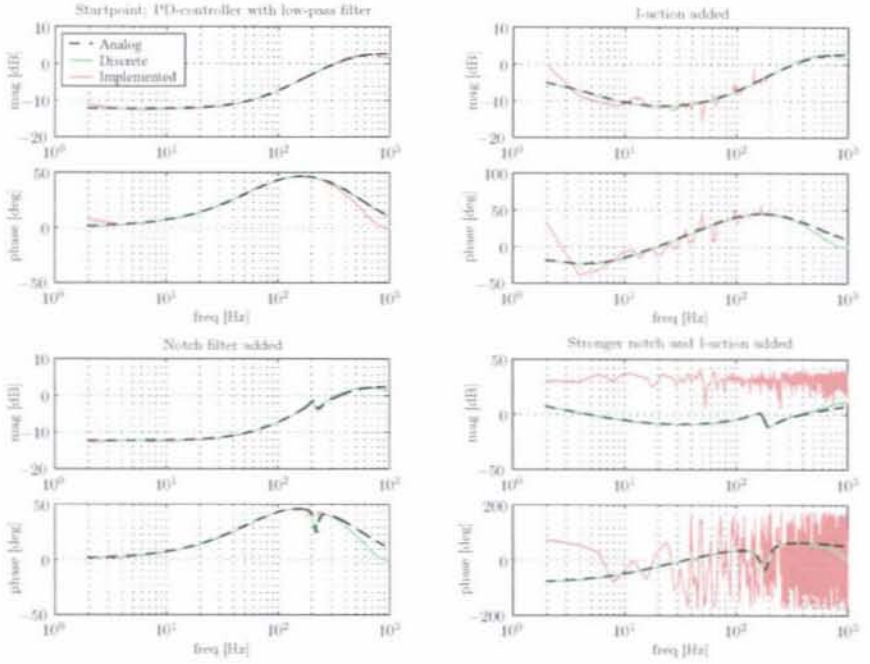

Figure 4.4: Four controllers of increasing difficulty; analog, discrete and the result after implementation.

was one of the connections of the left elastic hinge to the frame, which appeared to have a bit clearance. After removing the connection, the resonance disappeared. Increasing the gain again resulted in another part of the system to start resonating. This resonance does not start nor stop by itself, but it starts if the closed loop system receives an impuls and may stop if the system is damped efficiently for some time. Figure 4.6 shows the start en stop of this resonance. The frequency plot of the frequency range the resonance is present shows a clear peak around 250 Hz, so this non-linearity will be of that frequency.

4.4 Control algorithm

The control algorithm is written in C-code. The program consists of three parts: the first part initialises the processor and the DSP and includes the calibration function (see section 2.2). The second part handles the interaction with the PC, which is used as interface and the third part is the actual control function, which is called every 0.005s. This function starts reading the AD-converters 8 times and uses the mean instead of a single reading to reduce measurement noise. It calculates the position and error of both actuators with the look-up tables with a position range of 0 to 6000 pulses. Using the error, the control signal is calculated using integer operations to reduce calculation time. To have enough parameter resolution, the control parameters are multiplied by 1024 and the resulting control signal is divided by 1024 again. This results in the following controller, which follows from Figure 4.7 [4]:

$$\frac{num}{den} = \frac{num_1 + num_2 \cdot z^{-1} + num_3 \cdot z^{-2}}{den_1 + den_2 \cdot z^{-1} + den_3 \cdot z^{-2}} \tag{4.2}$$

$$\begin{aligned} den_1 \cdot u_k &= (num_1 \cdot e_k + num_2 \cdot e_{(k-1)} + num_3 \cdot e_{(k-2)}) \cdots \\ &\quad -(den_2 \cdot u_{(k-1)} + den_3 \cdot u_{(k-2)}) \end{aligned} \tag{4.3}$$

with u_k and y_k the control output respectively the control input in sample k.

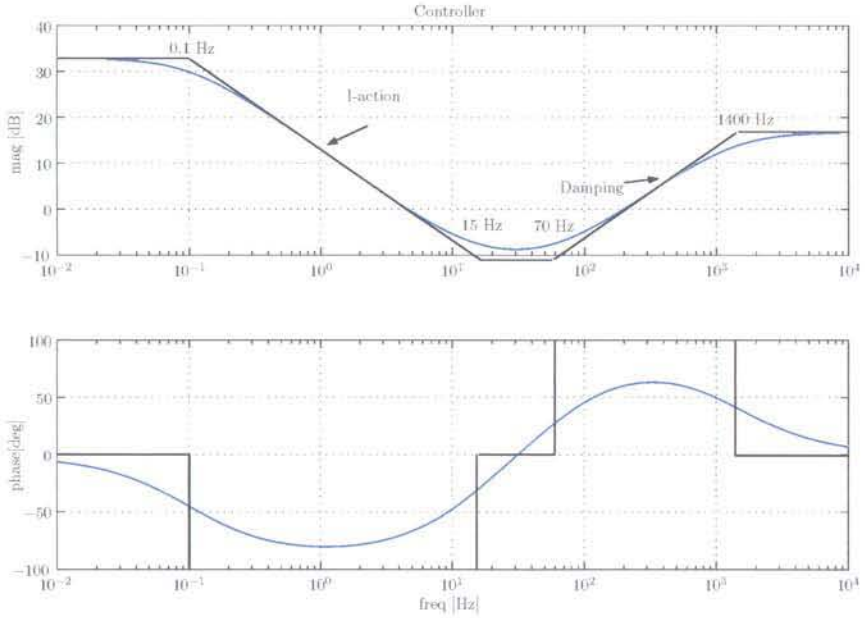

Figure 4.5: Designed controller

The calculation of the second part, the filterterm, is only based on information of earlier samples. Therefore in practice, this part is already calculated the previous time the control function is called to reduce calculation time between measurement and actuation. This results in the following implementation of the controller:

$$u_k = \left(\sum_{i=1}^{k}(e_{k-i+1} \cdot num_i)\right)/1024 - F \tag{4.4}$$

with

$$F = \left(\sum_{i=2}^{k}(u_{k-i+1} \cdot den_i)\right)/1024 \tag{4.5}$$

calculated in the previous call.

To the resulting signal, a noise-alike input signal can be added to be able to calculate the sensitivity. This disturbed input signal is saved. Next the function calculates a few corrections: first both signals are multiplied by the decoupling matrix, K, and secondly they are corrected for the different voicecoil actuator constants with α_i as discussed in section 2.2.4 and for the different spring stiffnesses with α_i and β_i as discussed in section 2.2.3.

$$u_{L_c} = \left(\frac{K_{1,1}}{1024}u_L + \frac{K_{1,2}}{1024}u_R\right)\frac{\alpha_1}{1024} + \frac{\beta_1}{1024} \tag{4.6}$$

$$u_{R_c} = \left(\frac{K_{2,1}}{1024}u_L + \frac{K_{2,2}}{1024}u_R\right)\frac{\alpha_2}{1024} + \frac{\beta_2}{1024} \tag{4.7}$$

with u_{L_c} and u_{R_c} the resulting control inputs and $K_{i,i}$ the components of the decoupling matrix K.

After these calculations, the corrected signals are sent to two PWM-generators and the function starts a control loop for the speed control of the drums. This is a PD-controller using the

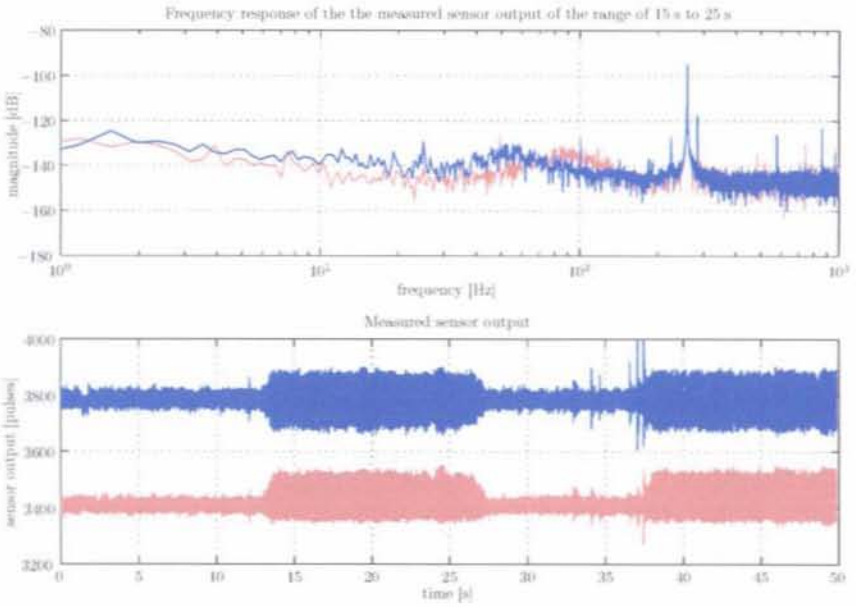

Figure 4.6: Time response of the system using a constant input signal and the sleeve rotating with 10000 mm/min; the frequency response is of the part between 15 s and 25 s of the time response in which a non-linearity of the system starts vibrating (the input signal stays constant).

speed of the DIP-drum and is not further discussed here. After all control outputs are send to the PWM-generators, the filter term, F, for the next sample time is calculated. Finally the measured positions, the setpoints and the control outputs are saved in a USB-buffer that sends the information to the PC.

4.5 Results

The design of the controllers has not resulted in a controller that meets the stated demands. A reasonably good controller, which resembles the controller of Figure 4.5 for a great deal is found. But because it does not meet the demands, the attention has focused on the demand to the bandwidth of the openloop responses and on a good implementation of the controllers.

The original controller resulted in steady state errors of 11μm and 16μm and dynamic errors of $\pm$4,5μm and $\pm$15,5μm for respectively the left and right actuator positions. The resulting controllers as described in this chapter resulted in steady state errors of 7,5μm and 1,0μm and dynamic errors of $\pm$4,5μm and $\pm$6,0μm for respectively the left and right actuator positions. There are several points that are the cause of only the slight improvement of the controller.

- Firstly, the present measurement noise, mainly caused by the position sensors, cause an error of $\pm$1,3μm already, so a maximum error after control of $\pm$1μm is not achievable.

- The mean disturbance forces amount to 2 N, which is reasonably large. In particular on the right side, where the drive of the DIP-drum and sleeve are positioned, the error is big as can be seen in Figure 3.3.

- At a certain point if quite good notches are implemented, the parameters of the total controller become too large for the fixed point implementation. Therefore the bandwidth of

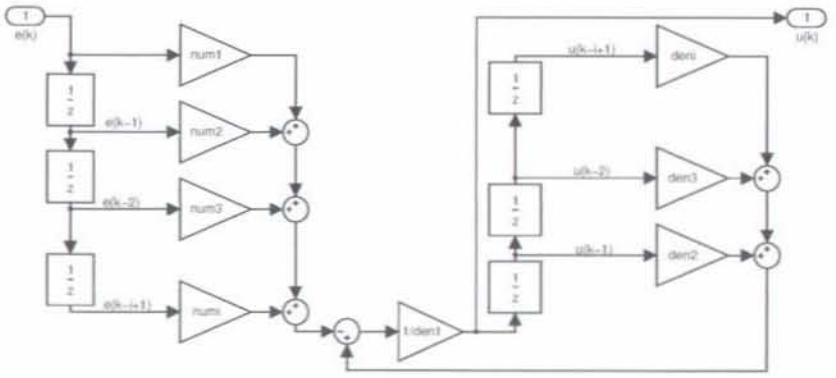

Figure 4.7: Overview of a discrete controller in the z-domain as implemented in the C-code.

the resulting controllers is just too low: on the verge of 60 Hz, while for a good disturbance rejection a bandwidth of at least 70 to 100 Hz is required.

Another way to increase the stiffness of the controller percepted by the DIP-knife is to increase the transmission in the elastic hinges. A disadvantage is the increasing stroke of the actuators. Besides, more damping in the hinges result in significant improvement in disturbance rejection. This was tested by placing some tape over the hinges. The disadvantage of it is a very restricted stroke, therefore some other way of damping may be thought of.

Because the design of the controller stalled at a given point, an important part of the attention was focused on a good implementation of the controllers. This resulted in a easy to handle and good implementation of continuous designed controllers: they are transferred to the discrete domain, can be uploaded to the C-code of the DSP directly or evaluated using the simulation model and can be tested if uploaded to check if the implementation went correct (see section D).

Chapter 5

Conclusions and recommendations

5.1 Conclusions

The aim of this study was to make a model of a test frame and design a controller for it. In addition to this, calibrations and linearizations were performed, which were necessary to identify the system.

Calibration

Before the system is identified, the inputs and outputs of the system are calibrated and if necessary and possible, linearized. The sensor values are linearized and scaled using a look-up table. The control outputs are scaled and then a correction can be added to make the left and right actuator stiffness equal. This latter option is not used since it adds some unknown controller gains to the system, which can cause instabilities. The inputs and outputs are scaled to make a transition from bits to S.I.-units easy. The system's linearity is checked by measuring the frequency response of the system using different controllers and input signals. The result is a linear system with inputs and outputs in S.I.-units. After this calibration some masses and stiffness are determined to get an idea of the system parameters before starting the system identification.

To gain information about the disturbances the controller should reject, these disturbances are identified. The disturbances are higher than expected (about 2 N up to 100 Hz), which result from friction in the drives of the drums. Besides the measurement noise initiates an error of about 1,3μm. Some additional damping in the hinges results in significant improvement in disturbance rejection and by increasing the transmission in the hinges, the stiffness of the controller percepted by the DIP-knife increases.

System identification

A blackbox and a whitebox model of the linearized and calibrated system are made. First the MIMO-frequency response is measured using an open-loop sensitivity measurement. The model parameters are tuned in the frequency domain using these measured responses. The whitebox state-space model uses the physical masses, springs and dampers to model the system. This results in a better fit, but does not give much insight in the influence of different parts of the system on the output. The whitebox model is accurate up to 150 Hz, the blackbox model is also accurate up to higher frequencies. To increase the accurateness of the whitebox model some tests are performed with a time domain optimization algorithm. This can be extended in future for better results. The blackbox model only has a bad low-frequent phase for the non-diagonal terms. Using the whitebox model a simulation model is made. With this model time-simulations of the system can be performed with different controllers to check the controllers before implementing them.

To enable controlling using two SISO-controllers, the system is decoupled. Different ways to decouple the system are tested and finally a method that uses the inverse transfer function of the system in a certain frequency range is chosen. In this way the best decoupling is achieved between 20 Hz and 100 Hz, the bandwidth of the system.

Controller

The final controller does not meet its demands. One of the demands was to decrease the dynamical position error to $\pm 1\mu$m. With the final controller this error is still $\pm 4{,}5\mu$m and $\pm 6\mu$m for respectively the left and right controller. Some improvements have to be made to the system before the controller can be optimized further. The position sensors still have an accuracy of $\pm 1{,}3\mu$m, which needs improvement before the position error can be decreased. Furthermore, some non-linear high-frequent resonances in the system need to be suppressed before the controller gain can be increased.

A toolbox is made for the design of a MIMO-controller for this system, transfer this controller to a discrete controller, load the controller to the actual system and test the uploaded controller. This simplifies designing and testing new controllers. An improvement, which have to be made in the C-code, is changing the fixed-point operations of the online controller to floating point operations. The parameters of the current controller overflow if reasonably complicated controllers are implemented.

5.2 Recommendations

Calibration

To improve the accuracy of the system, the disturbances should be decreased. The measurement noise needs to be decreased below $\pm 1\mu$m before the position error can be decreased to this point. The drive of the drums can also be improved, decreasing the disturbances the controller needs to reject. A last recommendation is to attach the DIP-drum to the test frame since that would make the test frame more realistic and some new system characteristics will be added and to see if additional damping to the elastic hinges can be added.

System identification

A parameter optimisation of the whitebox parameters using time domain signals and information may result in a better approach of the exact parameter values. The simulation model is not used a lot because the tools to upload and test the designed controllers directly are very easy in use. But for further study the model can be of good help in the controller design process.

The decoupling of the force and momentum to the radial displacement and rotation of the DIP-knife is not used in the controller design. It is however implemented in the C-code and some measurements with this decoupling have been performed. Probably a good controller can be designed using this decoupling as well, which should be tested.

Controller

For more extensive controller design, the use of floating point operations instead of the current used fixed point integers, is required. Otherwise the parameters of the controllers will become too large to be implemented.

Chapter 6

References

[1] A.M. Tuytelaars, *Voice Coil Actuator for the control of position*, graduation report Océ-Technologies B.V., Research & Development, January 2002, Venlo.

[2] V.W.C.M. van Lierop, *Mechanical design of an active controllable DIP-unit*, graduation report, Océ-Technologies B.V., Research & Development, April 2002, Venlo (design of elastic hinges, specifications voicecoil actuators).

[3] R.W. Mickers, *Position control for a DIP-unit*, graduation report, Océ-Technologies B.V., Research & Development, June 2002, Venlo (design of an optical sensor and parameter values of the test frame).

[4] G.F. Franklin, J.D. Powell and A. Emami, *Feedback control of dynamic systems*, Addison-Wesley Publishing Company, Inc., third edition, 1994.

[5] S. Skogestad, I. Postlethwaite, *Multivariable feedback control - analysis and design*, 1996

[6] DIET-toolbox, *frfit.m*, a tool to fit blackbox models

[7] *align.m*, a tool to decouple MIMO-systems, see also [11]

[8] A. Damen, S. Weiland, *Robust control*, H_∞ controller design, University of Technology Eindhoven, July 17, 2002

[9] J.J. Kok, *Werktuigkundige Regeltechniek II*, lecture notes of the lecture Control systems of the department of mechanical engineering, University of Technology Eindhoven, 1985/86

[10] A. Vandenput, *Elektromechanica*, lecture notes of the lecture Electromechanics of the department of electrical engineering, University of Technology Eindhoven, January 1999

[11] F.P. Wijnheijmer, G.J.L. Naus, *CD-rom 'Control of a DIP-knife'*, CD-rom including matlabtools for the calibration, identification and control of a test frame of the DIP-knife, see appendix D, February 2004

[12] T.A.G. Heeren, (F.P. Wijnheijmer, G.J.L. Naus,) *Users manual*, users manual for the interface controlling the test frame 'positioning of the DIP-knife', version 2, February 2004

Appendix A

Variables

Table A.1 gives an overview of the variables and parameters of the system. Partly the values are taken from literature [1, 2 and 3], but most of them are checked and measured again during the calibration and identification of the system.

Var.	Unit	Variable name	Value
Frame			
m_{knife}	$[kg]$	mass of the DIP-knife	1,2
m_{drum}	$[kg]$	mass of the DIP-drum	10
Voicecoil actuators			
m_{act}	$[kg]$	mass of the voicecoil actuators, with $i^2 \cdot m_{act} = 0{,}36$, so m_{act} is of the same size as m_{DIP}, which is ok	0,01459
N_{act}	[-]	number of windings in the coils	110 ± 5
d_{act}	$[m]$	diameter of wire used for the windings	$0{,}17 \cdot 10^{-3}$
c_{act}	$[N/A]$	voicecoil actuator constant	3,14
	$[N/\%PWM]$		0,03
R_{act}	$[\Omega]$	resistance of the coil	12,0
Elastic hinges			
k_{act}	$[N/m]$	stiffness of the transmission, the connecting part and the lever	$2{,}9 \cdot 10^3$
		stiffness of the extra hinge in the sandwich construction of the complete hinge	$0{,}5 \cdot 10^3$
		total stiffness at the input of the hinge	$3{,}4 \cdot 10^3$
		total stiffness with the compensating magnet	$0{,}4 \cdot 10^3$
k_y	$[N/m]$	stiffness perpendicular to a single hinge (in case of the sandwich constructed hinge this stiffness increases 100 times)	$1{,}3 \cdot 10^5$
k_t	$[N/m]$	stiffness against pulling	$56 \cdot 10^6$
k_{es}	$[N/m]$	stiffness of the hinge	$16 \cdot 10^3$
i	[-]	transmission proportion	5
b_{es}	$[Ns/m]$	damping in the hinge	1
Optical sensor and LED			
λ_{LED}	$[nm]$	wavelength of the (red) LED-light	635
V_u	$[V]$	range of the output voltage, which is variable by R_{LED}	$1{,}0 - 4{,}0$
V_i	$[V]$	input voltage	5,0
R_{LED}	$[M\Omega]$	variable resistor of the LED	$O(1{,}0)$
C_{os}	$[V/mm]$	mean sensor constant	$6{,}7 \cdot 10^{-3}$
a_{os}	$[V/m]$	mean amplification	1300

Control output: u_i $\%PWM \cdot 15$. $\%PWM$ limited between $-100\%(-24\,\mathrm{V})$ and $100\%(24\,\mathrm{V})$ and therefore u_i is limited between -1500 en 1500. Mickers [3]: $\%PWM$ 0,0372 N, but measurements showed: $\%PWM$ 0,0279 N (left) and 0,0299 N (right).

Table A.1: Variables and parameters of the test frame: measured and taken from literature.

Appendix B

Whitebox model

In figure B.1 the 3D-whitebox model of the system designed in section 3.3 is shown. An overview of the used parameters and the result of their optimisation is given in table B.1 (see also section 3.3.2). The inputs and outputs of the system are the forces F_1 and F_2 respectively the displacements x_4 and x_5 of the left and right voicecoil actuator.

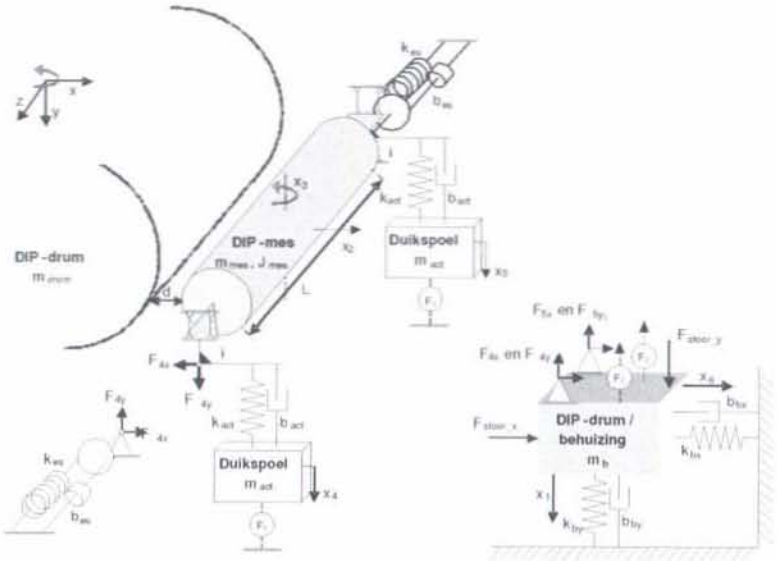

Figure B.1: Overview of the 3D-whitebox model as designed in section 3.3.

The resulting equations are:
In x-direction:

$$\begin{aligned} m_{knife} \cdot \ddot{x}_2 &= i \cdot (k_{act4} \cdot x_4 + k_{act5} \cdot x_5) + i \cdot (b_{act4} \cdot \dot{x}_4 + b_{act5} \cdot \dot{x}_5) \cdots \\ &\quad -(i^2 \cdot (k_{act4} + k_{act5}) + k_{es4} + k_{es5}) \cdot x_2 - (i^2 \cdot (b_{act4} + b_{act5}) \cdots \\ &\quad +b_{es4} + b_{es5}) \cdot \dot{x}_2 + (k_{es4} + k_{es5}) \cdot x_6 + (b_{es4} + b_{es5}) \cdot \dot{x}_6 \end{aligned} \tag{B.1}$$

$$\begin{aligned} m_{act4} \cdot \ddot{x}_4 &= k_{act4} \cdot i \cdot (x_2 + L/2 \cdot x_3) - k_{act4} \cdot x_4 + b_{act4} \cdot i \cdot (\dot{x}_2 + L/2 \cdot \dot{x}_3) \cdots \\ &\quad -b_{act4} \cdot \dot{x}_4 + F_1 \end{aligned} \tag{B.2}$$

$$\begin{aligned} m_{act5} \cdot \ddot{x}_5 &= k_{act5} \cdot i \cdot (x_2 - L/2 \cdot x_3) - k_{act5} \cdot x_5 + b_{act5} \cdot i \cdot (\dot{x}_2 - L/2 \cdot \dot{x}_3) \cdots \\ &\quad -b_{act5} \cdot \dot{x}_5 + F_2 \end{aligned} \tag{B.3}$$

$$\begin{aligned} m_b \cdot \ddot{x}_6 &= -(k_{bx} + k_{es4} + k_{es5}) \cdot x_6 + -(b_{bx} + b_{es4} + b_{es5}) \cdot \dot{x}_6 + (k_{es4} + k_{es5} \cdots \\ &\quad + i^2 \cdot (k_{act4} + k_{act5})) \cdot x_2 + (b_{es4} + b_{es5} + i^2 \cdot (b_{act4} + b_{act5})) \cdot \dot{x}_2 \cdots \\ &\quad + F_{stoorx} - i \cdot k_{act4} \cdot x_4 - i \cdot k_{act5} \cdot x_5 - i \cdot b_{act4} \cdot x_4 - i \cdot b_{act5} \cdot x_5 \end{aligned} \tag{B.4}$$

In y-direction:

$$\begin{aligned} m_b \cdot \ddot{x}_1 &= -k_{by} \cdot x_1 + -b_{by} \cdot \dot{x}_1 + F_{stoory} - F_1 - F_2 - i \cdot (k_{act4} + k_{act5}) \cdot x_2 \cdots \\ &\quad - i \cdot (b_{act4} + b_{act5}) \cdot \dot{x}_2 + k_{act4} \cdot x_4 + k_{act5} \cdot x_5 + b_{act4} \cdot x_4 + b_{act5} \cdot x_5 \end{aligned} \tag{B.5}$$

In ϕ-direction with $x_3 = 2/(i \cdot L) \cdot (x_4 - x_5)$:

$$\begin{aligned} J_{knife} \cdot \ddot{x}_3 &= -L^2/4 \cdot ((k_{es4} + k_{es5}) + i^2 \cdot (k_{act4} + k_{act5})) \cdot x_3 - L^2/4 \cdot ((b_{es4} + b_{es5}) \cdots \\ &\quad + i^2 \cdot (b_{act4} + b_{act5})) \cdot \dot{x}_3 + L/2 \cdot i \cdot (k_{act4} \cdot x_4 - k_{act5} \cdot x_5) \cdots \\ &\quad + L/2 \cdot i \cdot (b_{act4} \cdot \dot{x}_4 - b_{act5} \cdot \dot{x}_5) \end{aligned} \tag{B.6}$$

Var.	Unit	Variable name	Value 1	Value 2
General variables				
i	$[-]$	scaling in the joint	5	8,72
L	$[m]$	axial length of the DIP-knife	0,42	1,48
d	$[m]$	offset between the DIP-knife and the DIP-drum, measured at the actuator sides of the hinges		
$F_{1,2}$	$[N]$	controller forces		
$F_{4,5(x,y)}$	$[N]$	joint forces in x- and y-direction		
Masses				
m_b	$[kg]$	mass of the DIP-drum and the housing	7,0	−14,4
m_{knife}	$[kg]$	mass of the DIP-knife	1,2	1,65
m_{act4}	$[kg]$	mass of the left voicecoil actuator	0,03	0,014
m_{act5}	$[kg]$	mass of the right voicecoil actuator	0,03	0,014
J_{knife}	$[kgm^2]$	inertia of the DIP-knife	0,02	0,05
Stiffnesses				
k_{bx}	$[N/m]$	stiffness in the suspension of the housing in x-direction	$3 \cdot 10^7$	$7{,}4 \cdot 10^7$
k_{by}	$[N/m]$	stiffness in the suspension of the housing in y-direction	$3 \cdot 10^7$	$-9{,}5 \cdot 10^6$
k_{es4}	$[N/m]$	torsional stiffness of the left elastic joint (the DIP-drum or housing are presumed to be torsional stiff and not of importance for this situation)	$5 \cdot 10^4$	$-1{,}8 \cdot 10^4$
k_{es5}	$[N/m]$	torsional stiffness of the right elastic joint	$1 \cdot 10^4$	$1{,}2 \cdot 10^5$
k_{act4}	$[N/m]$	stiffness of the left actuator and the sagging of the joint (as seen from the side of the actuator)	$2{,}36 \cdot 10^4$	$9{,}0 \cdot 10^3$
k_{act5}	$[N/m]$	stiffness of the right actuator	$2{,}3 \cdot 10^4$	$8{,}6 \cdot 10^3$
Dampers				
b_{bx}	$[Ns/m]$	damping in the suspension of the housing in x-direction	100	−177
b_{by}	$[Ns/m]$	damping in the suspension of the housing in y-direction	100	570
b_{es4}	$[Ns/m]$	torsional damping of the left elastic joint	$1 \cdot 10^2$	−24,4
b_{es5}	$[Ns/m]$	torsional damping of the right elastic joint	$1{,}2 \cdot 10^2$	404,2
b_{act4}	$[Ns/m]$	damping of the left actuator and the joint (as seen from the side of the actuator)	0,02	−0,002
b_{act5}	$[Ns/m]$	damping of the right actuator and the joint	0,03	−0,047
States				
x_1	$[m]$	displacement of the DIP-drum and the housing in x-direction		
x_2	$[m]$	displacement of the DIP-knife		
x_3	$[rad]$	rotation of the DIP-knife		
x_4	$[m]$	displacement of the left voice coil actuator		
x_5	$[m]$	displacement of the right voice coil actuator		
x_6	$[m]$	displacement of the DIP-drum and the housing in y-direction		

Table B.1: Parameters used in the whitebox model with their values resulting from optimisation in the frequency domain by hand, using the physical impact of the parameters, (Value 1) and the built-in Matlab minimization algorithm *fmins* (Value 2).

Appendix C

H_∞-control

The possibilities of robust control, H_∞ [8], have been examined since this technique is especially good with MIMO-control problems and model uncertainties. In this case a controller has to be designed for a test frame, which has to be implemented in many other systems and therefore system variations and uncertainties will be present. Other control techniques such as MPC are too slow for this specific problem and therefore in this appendix the start of the design of a H_∞-controller is described.

C.1 H_∞ - Augmented plant

To start with robust controller design, the problem has to be defined as an augmented plant with exogenous and control inputs and measured outputs and outputs to be controlled. The latter in- and outputs are connected to the controller, which is the only block besides the augmented plant (see figure C.1).

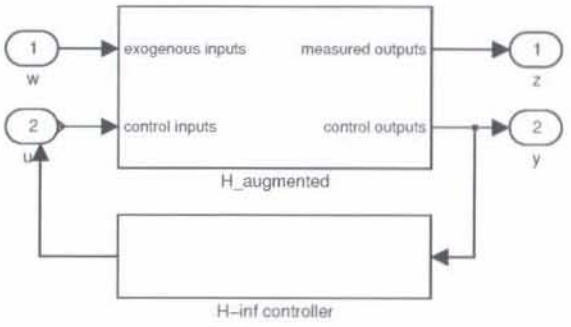

Figure C.1: Setup of the H_∞-control problem.

In this case the exogenous inputs are the shaped setpoints $\tilde{s}_1$ and $\tilde{s}_2$ and the shaped disturbance forces $\tilde{d}_1$ and $\tilde{d}_2$. The control inputs are the outputs of the controller $K(s)$ u_1 and u_2. The outputs to be controlled are the weighted errors $\tilde{s}_1 - y_1$ and $\tilde{s}_2 - y_2$ and the weighted control efforts $\tilde{u}_1$ and $\tilde{u}_2$. As measured outputs the errors are used. To shape the inputs and weight the outputs, shape and weighting filters are used.

C.2 Weighting and shape filters

Using the augmented plant and assuming the same conditions hold for the left and right side of the system concerning the demands to the inputs and outputs, 4 filters have to be designed: We,

Wu, Vs and Vd, respectively an output weighting filter for the error and the control effort and an input shape-filter for the setpoint and the disturbance signals. Per filter, the design of it will be explained briefly. For the origin of the controller demands, see section 4.1.2, for simplicity the bandwith is taken 100 Hz.

We Weighting filter for the error, the difference between the setpoint signal and the real output of the system $\Rightarrow$ this has to be a lowpass filter because errors up to 100 Hz have to be compensated. A maximum error of $2 \cdot 10^{-6}$ m has to be compensated between 0 Hz and the bandwith of 100 Hz. At the actuator a gain of $20 \cdot \log 1/(5 \cdot 2 \cdot 10^{-6})$ m $= 80$ dB is needed. To make the filter proper and limit the high-frequent gain, a zero is placed at $100 \cdot$ bandwith.

Wu weighting filter for the control effort, the input of the plant / output of the controller $\Rightarrow$ to avoid saturation, a highpass filter is needed. The maximum input of the plant is 0,028 N/PWM% $\cdot$ 80 PWM% $= 2{,}2$ N. Up to 100 Hz this has to be the maximum the controller may ask of the actuators to avoid saturation. So a gain of $20 \cdot \log 1/2{,}2 = -7$ dB is needed. Besides a pole at $100 \cdot$ bandwith is placed to make the filter proper.

Vs setpoint shape filter $\Rightarrow$ a lowpass filter is required because the setpoints will have a frequency up to 3 Hz. A maximum amplitude of $1{,}5 \cdot 10^{-3}$ m gives a lowpass gain of $20 \cdot \log 1{,}5 \cdot 10^{-3}$ m $= -56$ dB. Above this 3 Hz there will be no setpoint signal so a zero at $100 \cdot 3$ Hz $= 300$ Hz completes this filter.

Vd disturbances shape filter $\Rightarrow$ a lowpass filter is required because up to 100 Hz disturbances are present. Besides, 90% of the effects of a step in the input has to be compensated within 10 ms. So a lowpass gain or amplitude of $20 \cdot \log 90\% \cdot 1{,}5 \cdot 10^{-3}$ m $= -57$ dB is wanted. A zero at $100 \cdot$ bandwith completes the filter.

A highpass shape-filter to describe the measurement noise can be added as well, but this was not implemented yet. If the measurement noise is stated to be at maximum 5% of the input signal, a gain of $20 \cdot \log 5\% \cdot 1{,}5 \cdot 10^{-3}$ m $= -82$ dB above 100 Hz would be a possible shape-filter.

Using the Matlab built-in tool *zp2sys(zeros,poles,gain)* the first design of the filters is made. The results are shown in figure C.2.

C.3 Results

Using the designed filters and a good model of the system, blackbox or whitebox, and with the help of the *lmi*- and *Mutools*-toolboxes a robust controller can be designed. The order of the resulting H_∞-controller depends on the order of the model and the used weighting and shape-filters. In this case the model is of reasonably high order so a reduced model of the system has to be used as a result of which characteristic resonances have to be ignored and the controller may not be the optimal.

In the design process of the controller, the parameter g is optimized. This parameter indicates if the demands to the controller, initiated by the design of the filters, about performance and robustness can be united in 1 controller. The optimization algorithm searches for a stable controller and next determines every stable controller in the neighborhood of this first controller. Via an optimization algorithm the solution converges to the optimal controller for the stated problem. If g is just a little smaller than 1, a good controller is found. If this value is not reached, the demands concerning the performance and robustness of the controller have to be reviewed.

To determine the bottleneck of the problem, the closed-loop responses can be compared and the mixed sensitivity calculated. With these results, the right filters can be softened or strengthened, optimized and a new controller can be calculated. By repeating this process several times, an optimal g for the stated problem is found.

In this case a reduced model of the system had to be used because the blackbox and whitebox model were of too high order. A controller with a g of about 20 was found, but the calculated controllers could nog be optimized furthermore because the controller still was of too high order

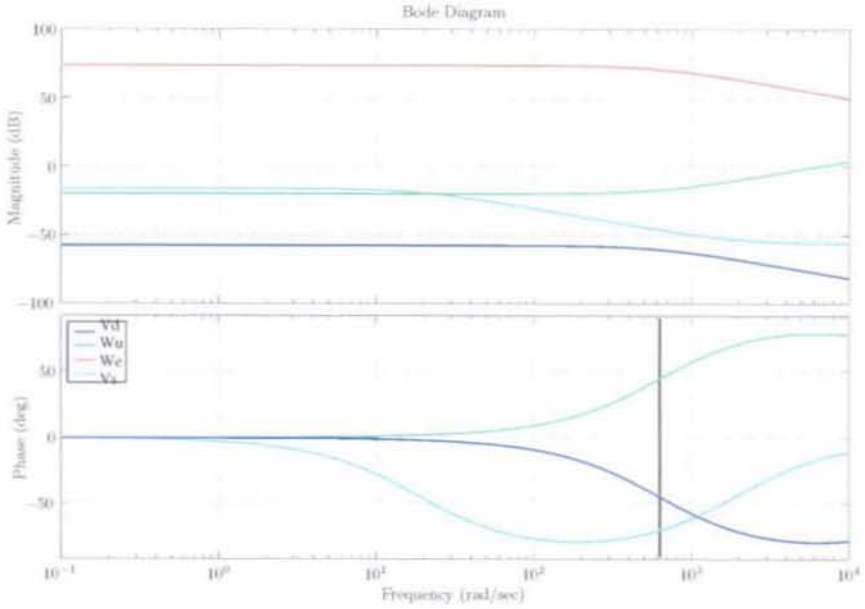

Figure C.2: The weighting and shape-filters for the H_∞-controller with a bandwith of 100 Hz.

and the mixed sensitivity for example could not be calculated. Reducing the order of the model furthermore would have resulted in a bad controller in any case and therefore the design of the H_∞-controller was stopped at this point.

Appendix D

Supporting Matlab-tools

In this appendix only the most general tools [11] are listed that can be of use in further studies of the system and its control problem. Some of them will have Dutch comments.

D.1 Calibration

equalize_springs.m	equalization of the stiffnesses of the hinges, automatically uploading the correction factors (makes use of *ser_read_fifo_es.m*)
leastsquares.m	a least squares fit of desired order is calculated to compensate for the S-shape in the position sensor output (first run *ser_read_fifo.m* using the sawtooth signal as input)

Some other files and figures are included, which have been used to calibrate and calculate the various parts and variables of the system. They are all based on previously performed measurements and therefore can just be used to see the way the calculations are performed. Besides the resulting values are only calculated once for further use in the identification of the system.

D.2 System Identification

disturbance identification

inputid_cumsum.m	identification of the disturbances using the cumulative error (makes use of *metinput10000.mat*, *metinput20000.mat* and *zonderinput.mat*)
inputid.m	identification of the disturbances (makes use of *whitebox_model*, *feeder_ruis.mat* and *zonderinput*)

modelling

measurements_nd.mat	non-decoupled frequency domain measurement data
measurements_d.mat	decoupled frequency domain measurement data
whitebox_model.m	whitebox model in differential and state-space form, including the optimised parameter values (makes use of *measurements_nd.mat* and *X_mins.mat*)
blackbox_model.mat	non-decoupled blackbox model
models.m	comparison between the whitebox and blackbox models and the measurement data (makes use of *measurements_nd.mat*, *whitebox_model.m* and *blackbox_model.mat*)
DEMO_modes.m	DEMO in which the eigenmodes resulting from the whitebox model are shown, see section D.3 (makes use of *DEMO_teken*, *DEMO_trans*, *statespace_model_nsym.m*)

decoupling

align.m	real alignment
decouple.m	to decouple a 2x2 system: over a frequency range using *align.m* or from F and M to x_4 and x_5 (makes use of *measurements_nd.mat*, *blackbox_model.mat*, MCD-toolbox and *align.m*)

D.3 Controller and test frame control

Controller

MCD	tool for MIMO-Controller Design, see section D.4
load_controller.m	to upload or reset the decoupling (force-momentum to x_4 and x_5 or user defined) and to upload or reset a controller (user defined or from the *MCD*-toolbox) (makes use of *ser2_init*)
test_controller.m	to test whether an uploaded controller is uploaded ok, because errors may occur through the use of fixed point integers (makes use of *ser_read_fifo_contr* and *ser2_init*)
HupsaK.m	design of the H_∞-controller (makes use of *HupsaF.m*)
HupsaF.m	design of the weighting and shape filters of the H_∞-controller
mixedsens.m	calculation of the mixed sensitivity of the used H_∞-problem

Test frame control

dsp_trans.m	calculation of the 4 responses of the system after a test with a noise-alike signal has been performed (makes use of the uploaded controller variables *num*, *den*)
ser2_init.m	initialisation of the COM2 gate after the C-code has been uploaded To easy the control of the system using Matlab: *cc.m* and *clearall.m*
ser_read_fifo.m	standard inputs for the system and processing of the output signals: a sawtooth, noise-alike, staircase, etc (see the users manual [12])
DEMO.m	DEMO interface, to perform some simple control actions (makes use of *DEMO_serreadfifo.m*, *DEMO_teken*, *DEMO_trans*, *statespace_model_nsym.m*)
DEMO_modes.m	DEMO in which the eigenmodes resulting from the whitebox model are shown (makes use of *DEMO_teken*, *DEMO_trans*, *statespace_model_nsym.m*)

D.4 Toolbox

- To easy working with more responses at a time, 3D-matrices are useful. Therefore some tools to work quickly with 3D-matrices and cells are designed: *dot3D.m*, *eye3D.m*, *inv3D.m*, *make3D.m*, *star3D.m*, *mean3D.m*, *array2cell3D.m*, *cell2array3D.m*.
- To design multiple SISO-controllers for a MIMO-problem, a *diet*-alike tool is designed for MIMO-problems: *MCD.m* (see section D.3.
- To calculate and simulate simple discrete controllers, some functions that converse continuous controllers in a easy way to discrete controllers are designed: *I_function.m*, *LW_function.m*, *LW2_function.m*, *notch_function.m*, *PD_function.m*.
- Some other tools to easy and quicken working with Matlab are designed: *even.m*, *gfplot.m*, *hplot.m*, *hplot4x4edit.m*, *sys2H.m*.