Experimental stability analysis of networked control systems with constant time-delays

Experimental Stability Analysis of Networked Control Systems with Constant Time-delays

T. Kniknie (s000810)

DCT 2006.085

DCT short traineeship report

Supervisors:
Dr. Ir. Nathan van de Wouw
Ir. Marieke Cloosterman

Technische Universiteit Eindhoven
Department of Mechanical Engineering
Dynamics and Control Technology Group

Abstract

To control a remote (dynamical) system, data networks can be used. This has some advantages, but also introduces a time-delay between the controller and the plant. Systems, that use a network to close the control loop, are called Networked Control Systems (NCSs). This research treats the experimental validation of the influence of constant time-delays on the stability of an NCS.

Under certain assumptions concerning the properties of a network, a model is formulated, representing an NCS with constant time-delays. A second-order dynamical system (an inertia controlled by a PD controller) is used to analyze the stability properties for different controller gains and different constant time-delays. The stability analysis for this system shows that the maximum allowable velocity feedback gain for a stable system varies with the amount of delay in a remarkable manner. It increases until a delay $\delta_t = 0.25h$, with h the sample time, and then decreases for increasing delay. This peak in the stability region can be explained with the aid of a Bode and Nyquist diagrams. The Bode and Nyquist diagrams show that a time-delay lowers the phase of the system response, but also the magnitude. At higher time-delays the decrease of the magnitude can not compensate for the phase lag anymore.

To validate the stability analysis, experiments are performed. First, a suitable setup is chosen, based on the extent to which it represents a real NCS, the ease of use, reliability and accuracy. After comparing, the PATO experimental setup is chosen to conduct the experiments. The system parameters of the setup are estimated, based on Frequency Response Function (FRF) measurements. From the FRF measurements it can be concluded that besides a time-delay, due to discretization, more time-delay is present in the system, which will influence the stability properties. The measurements are conducted, using a step function in the position as reference profile for the motor. The stability of the system is evaluated experimentally by determining whether the error is constant.

The final measurements do not show resemblance to the analysis of the model. The reason for this mismatch most probably is the presence of static friction in the setup. Consequently a step function is not a suitable reference profile, since the friction effects dominate. Another reason is that a velocity estimator and a lowpass filter are used in the experiments, but these are not included in the model. Therefore, the model may not be a good representation of the real setup.

It is recommended to decrease the step in the time-delay, to obtain more measurement points. The method to determine the stability of the system has to be altered, because a step in the

reference position results in static friction problems. A constant reference velocity may lead to better results. The model has to be reconsidered, because the velocity estimation and lowpass filter are not included in the model. It is interesting to also investigate the control performance properties, like bandwidth and settling time.

Samenvatting

Om een (dynamisch) systeem op afstand te besturen, kunnen data netwerken gebruikt worden. Dit heeft enkele voordelen, maar introduceert ook een tijdvertraging tussen de regelaar en de machine. Systemen die een netwerk gebruiken om de regellus te sluiten, worden Networked Control Systems (NCS) genoemd. Dit onderzoek behandelt de experimentele validatie van de invloed van een constante tijdvertraging op de stabiliteit van een NCS.

Onder bepaalde aannames, betreffende de eigenschappen van een netwerk, is een model geformuleerd, dat een NCS met constante tijdvertraging beschrijft. Een tweede orde dynamisch systeem (een traagheid, geregeld door een PD regelaar) wordt gebruikt om de stabiliteitsvoorwaarden te bepalen, voor verschillende constante tijdvertragingen en versterkingswaarden van de regelaar. De stabiliteitsanalyse wijst uit dat de maximaal toelaatbare versterking van de snelheidsterugkoppeling, die het systeem nog stabiel maakt, op een merkwaardige manier verloopt met de variërende tijdvertraging. Deze versterking stijgt tot een vertraging $\delta_t = 0.25h$, met h de sample tijd, en daalt daarna weer met toenemende tijdsvertraging. Deze piek kan beter verklaard worden door te kijken naar de Bode en Nyquist diagrammen. Hierin is te zien dat een vertraging in het regelsignaal niet alleen de fase-achterstand van de responsie vergroot, maar ook de amplitude verkleind. Bij hogere vertragingen kan de amplitude afname de fase-achterstand niet meer compenseren.

Om de analyse te valideren, zijn experimenten uitgevoerd. Hiervoor wordt eerst een keuze gemaakt voor een geschikte experimentele opstelling. Criteria voor deze keuze zijn de wijze waarop de opstelling overeenkomt met een echt NCS, het gebruiksgemak, de betrouwbaarheid en de nauwkeurigheid. Na een vergelijking is de PATO experimentele opstelling gekozen om de experimenten mee uit te voeren. De systeemparameters zijn geschat met behulp van Frequentie Respons Functie (FRF) metingen. Uit de FRF metingen kan geconcludeerd worden dat naast een tijdvertraging als gevolg van discretisatie, er nog meer vertraging aanwezig is in het systeem. Dit zal de stabiliteitseigenschappen beïnvloeden. De metingen worden uitgevoerd door een stapfunctie in de positie als referentie aan te bieden aan de motor. De tijdvertraging worden gevarieerd tussen $\delta_t = 0$ en $\delta_t = h$ in vijf stappen. De stabiliteit van het systeem wordt experimenteel beoordeeld door te kijken of de positie-fout constant is en de regel-input nul is.

De uiteindelijke metingen komen niet overeen met de analyses. De reden voor deze verschillen is hoogst waarschijnlijk de statische wrijving in de opstelling. Een stapfunctie is dus geen goede keuze als referentiesignaal, omdat wrijvingseffecten overheersen. Een andere reden is dat in de experimenten een snelheidsschatter en laag doorlaatfilter zijn gebruikt, maar deze zijn niet opgenomen in het model. Het model is daarom geen goede weergave van de opstelling.

Het is aan te bevelen om meer verschillende tijdvertraging te testen, om zo meer meetpunten te verkrijgen. De methode om de stabiliteit van het systeem te bepalen zal veranderd moeten worden, omdat een stap in de positie resulteert in problemen met statische wrijving. Een constante snelheid als referentie kan betere resultaten opleveren. Een uitbreiding van het model is nodig, om de snelheidsschatter en het laag doorlaatfilter erin op te nemen. Het kan ook interessant zijn om de regeltechnische prestaties van het systeem te onderzoeken, zoals de bandbreedte en de tijdconstante.

Contents

Chapter 1

Introduction

To control a remote (dynamical) system, data networks can be used. The advantage of such a configuration is that multiple systems can be controlled from one computer, which increases the flexibility and ease of maintenance. Systems, that use a network to close the control loop, are called Networked Control Systems (NCSs). Figure 1.1 gives an example of a Networked Control System.

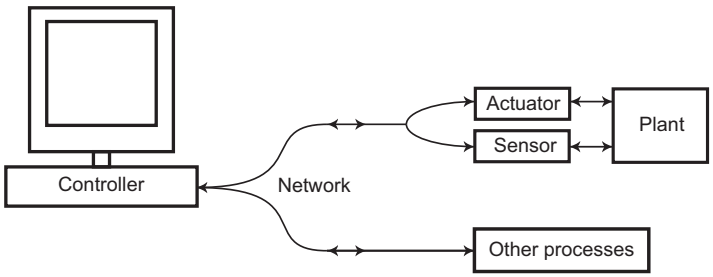


Figure 1.1: A Networked Control System.

The use of a network has some disadvantages, that need to be considered [7]:

1. The transfer of data over a network introduces a time-delay, which degrades the performance and stability of the system;
2. Data is sent over the network in packets, which can be lost during transmission;
3. The packets have a limited size, thus information can be divided over multiple packets, which can be delayed or get lost.

Because of the use of packets, network transfers can be considered as discrete-time events, which make an NCS a sampled data system.

1.1 Problem formulation and research approach

This research treats the experimental validation of the influence of constant time-delays on the stability of an NCS, under the assumption that packet loss and multiple-packet transmission

do not occur. Previous research on Networked Control Systems with constant time-delay was already done, but experimental validation is not yet available [5], [6], [7]. The emphasis of this work is on the validation of the theoretical stability region by experiments. A setup will be chosen, selected on the following criteria:

1. The extent to which it represents a real NCS;
2. The ease of use;
3. Reliability;
4. Accuracy.

Furthermore, the measurement data will be compared to the model results.

In the experiments, the stability of the system is determined by looking at the ability to follow a reference trajectory. Analytically, the stability can is investigated by looking at the eigenvalues of the system or is determined based on frequency domain information. The feedback gains of a controller and the amount of delay will be varied to obtain a complete overview of the stability region.

1.2 Outline

The outline of the report is as follows. In Chapter 2, the behavior of an NCS is explained and a discrete-time model is derived. This model is used to investigate the stability bounds of the NCS. In Chapter 3, an experimental setup is chosen, based on a comparison of two different setups. In Chapter 4, the chosen setup is used to validate the stability region experimentally. Hereto, the parameters of the setup are estimated and the resulting model and measurements are compared. Finally, conclusions and recommendations are presented in Chapter 5.

Chapter 2

Modeling and stability analysis of a Networked Control System

In order to analyze the stability of an NCS, a model will be formulated. The model will be analyzed both in time- and frequency domain and the results will be discussed. Before we present the NCS model, a generic schematic representation of an NCS is given in Figure 2.1.

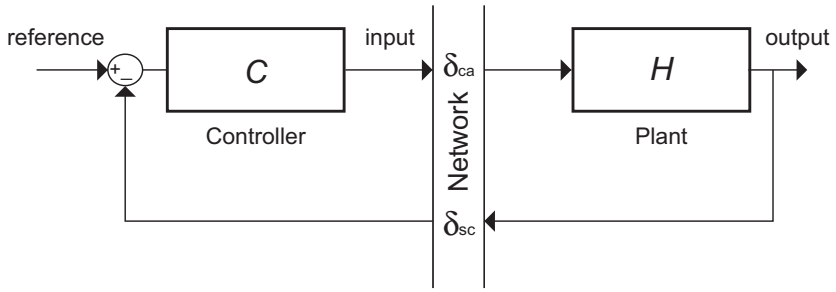


Figure 2.1: Schematic overview of a Networked Control System.

A continuous-time plant H is controlled by a discrete-time controller C over a network. The network induces a controller to actuator delay δ_{ca} and a sensor to controller delay δ_{sc}. To model the NCS, a few assumptions are made [5], [6]:

1. The total delay δ_t is smaller than the sample-time $h : \delta_{ca} + \delta_{cs} = \delta_t < h$.
   Since the actuator and controller are event-driven and the sensor is time-driven, δ_{ca} and δ_{cs} can be added and placed either between the controller and the plant, or in the feedback loop;
2. The delay is constant;
3. There is no computational delay.
   It is assumed that the computation-time cost is negligible compared to the time to transport data. Note that under the assumptions stated in assumption 1, the computational delay can be added to the network delays;
4. Data loss does not occur.

Based on these assumptions a model of the NCS with network induced delays can be derived. In order to combine the continuous-time plant and the discrete-time controller in one model, the plant will be discretized using a zero-order-hold conversion [2].

2.1 An NCS model

Consider a continuous-time linear time-invariant system:

$$\begin{aligned} \dot{\underline{x}}(t) &= \mathbf{A}\underline{x}(t) + \mathbf{B}u(t) \\ \underline{y}(t) &= \mathbf{C}\underline{x}(t), \end{aligned} \tag{2.1}$$

with $x \in \mathbb{R}^n$ the state, $u \in \mathbb{R}^m$ the input, $y \in \mathbb{R}^p$ the output and A, B, C the corresponding system matrix, input matrix and output matrix, respectively. After discretization and introduction of a delay δ_t, under the assumptions stated above, the discrete time model is derived as follows [2], [5]. A full-state feedback control law is formulated by:

$$u^*(t) = -K\underline{x}(kh) \quad \forall \ t \in [kh + \delta_t, kh + h + \delta_t], \tag{2.2}$$

with sample time h, delay time δ_t and feedback gain K. This results in the following system:

$$\begin{aligned} \underline{x}(kh + h) &= \mathbf{\Phi}\underline{x}(kh) + \mathbf{\Gamma_0}u(kh) + \mathbf{\Gamma_1}u(kh - h) \\ \underline{y}(kh) &= \mathbf{C}\underline{x}(kh), \end{aligned} \tag{2.3}$$

with $u(kh) = -K\underline{x}(kh)$ and

$$\mathbf{\Phi} = e^{\mathbf{A}h}, \qquad \mathbf{\Gamma_0} = \int_0^{h-\delta_t} e^{\mathbf{A}\eta} d\eta \mathbf{B} \quad \text{and} \qquad \mathbf{\Gamma_1} = \int_{h-\delta_t}^{h} e^{\mathbf{A}\eta} d\eta \mathbf{B}, \tag{2.4}$$

with the system matrices from (2.1), representing a dynamical system. Rewriting (2.3) and (2.4) in state-space notation yields:

$$\underline{z}(k+1) = \mathbf{\Psi}\underline{z}(k), \tag{2.5}$$

where

$$\underline{z} = \begin{bmatrix} \underline{x}(kh) \\ \underline{x}(kh - h) \end{bmatrix} \quad \text{and} \quad \mathbf{\Psi} = \begin{bmatrix} \mathbf{\Phi} - \mathbf{\Lambda_0} & -\mathbf{\Lambda_1} \\ \mathbf{I} & \mathbf{0} \end{bmatrix}, \tag{2.6}$$

with $\mathbf{\Lambda_0} = -\mathbf{\Gamma_0}K\mathbf{C}$ and $\mathbf{\Lambda_1} = -\mathbf{\Gamma_1}K\mathbf{C}$.

2.2 Analysis of a 2^{nd} order dynamical system

In order to analyze an NCS, a model will be made. A second order dynamical system (a single inertia) is chosen, because it represents a common dynamical motion system and the behavior of a second-order networked control system is known.

The second-order dynamical networked control system will be modeled, using the equations of section 2.1. The model will be used for eigenvalue analysis and analysis in the frequency domain; both aiming at the assessment of the stability of the NCS.

2.2.1 The equations of motion

Consider a wheel without damping and friction, depicted in Figure 2.2.

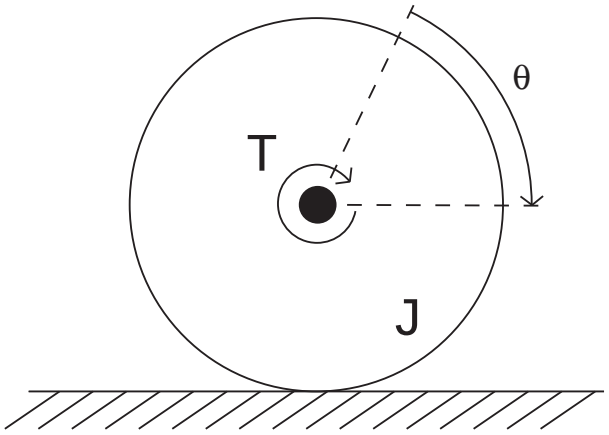


Figure 2.2: Schematic representation of a wheel.

The system can be described by the following equations of motion (under no-slip condition):

$$J\ddot{\theta} = T, \tag{2.7}$$

with T the input torque of the motor, θ the angular displacement of the wheel and J the inertia of the motor.

The system can be written in state-space notation, with $\underline{x} = \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix}$ as state variables:

$$\begin{array}{l} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}u^*(t) \\ y(t) = \mathbf{C}\underline{x}(t) \end{array}, \tag{2.8}$$

with $u^*(t)$ as in (2.2) and

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} 0 \\ \frac{1}{J} \end{bmatrix}, \qquad \mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

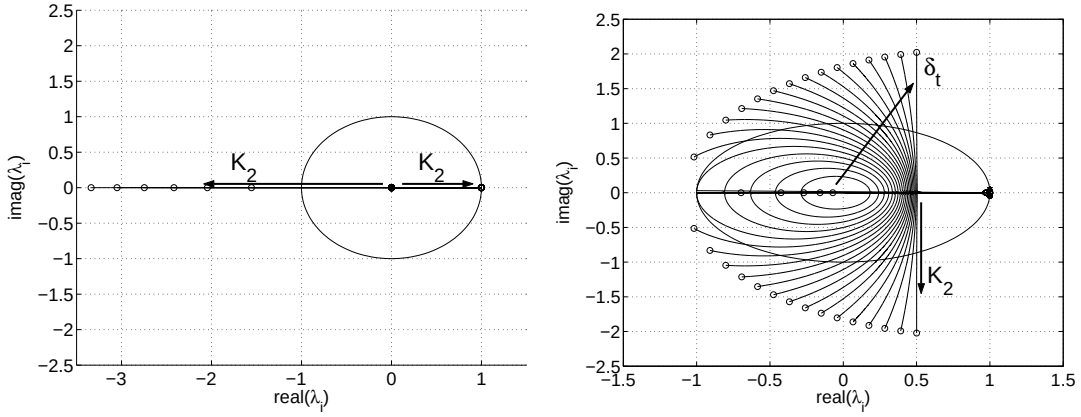
The system matrix $\mathbf{\Psi}$ of the discrete-time model, given in (2.5) now becomes:

$$\mathbf{\Psi} = \begin{bmatrix} 1 - \frac{1}{2}\frac{(h-\delta_t)^2}{J}K_1 & h - \frac{1}{2}\frac{(h-\delta_t)^2}{J}K_2 & -\frac{\frac{1}{2}h^2 - \frac{1}{2}(h-\delta_t)^2}{J}K_1 & -\frac{\frac{1}{2}h^2 - \frac{1}{2}(h-\delta_t)^2}{J}K_2 \\ -\frac{h-\delta_t}{J}K_1 & 1 - \frac{h-\delta_t}{J}K_2 & -\frac{\delta_t}{J}K_1 & -\frac{\delta_t}{J}K_2 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \quad (2.9)$$

2.2.2 Stability analysis

For the system matrix (2.9), an eigenvalue analysis can be used to evaluate the stability properties of the NCS. Therefore, the delay δ_t and the controller gains K_1 and K_2 are varied and for each resulting parameterset the eigenvalues are computed. For a discrete time system to be stable, the eigenvalues should not lie outside the unit circle [2]. When $K_1 = 0$, one eigenvalue λ_i equals 1, regardless of the value of K_2. Since a stable system requires all eigenvalues to be smaller than 1 in modulus ($|\lambda_i| < 1$), this implies that for the system to be stable, $K_1 > 0$.

For an inertia $J = 1.30{\cdot}10^{-5}\ kgm^2$, controller gains $K_1 = 0.0282\ Nm/rad$ and $K_2 \in [0, 0.0564]$ Nms/rad, delay $\delta_t \in [0, 1]\ ms$ and a sample time $h = 1\ ms$, the results of the eigenvalue analysis are shown in Figure 2.3.



(a) (b)

Figure 2.3: Root loci of the 2^{nd}-order NCS (2.5), (2.9).

The start- and endpoint of the eigenvalues as function of K_2 are marked respectively by an asterisk ($*$) and a circle ($\circ$). The arrows indicate in which direction the eigenvalues change as function of an increasing parameter (δ_t of K_2). In Figure 2.3, it can be seen that the eigenvalues all start within the unit circle and approach the unit circle with increasing K_2. Figure 2.3a shows two eigenvalues. One eigenvalue always stays in the point (0,0), while another one approaches and passes the unit circle with increasing gain K_2. Figure 2.3b shows another set of two eigenvalues. Here, the eigenvalues start on the left-hand side of the circle and with increasing gain they approach the unit circle. The influence of the time-delay δ_t and gain K_2 are shown in both figures, marked with the arrows. When the delay δ_t is increased, the eigenvalues in Figure 2.3b seem to tend towards the unit circle faster, but the gain can be set

higher, before the unit circle is passed. It seems like there is more 'room' for the eigenvalues to move inside the unit circle when the delay is increased. At some point, however, this effect is reversed and the delay has a negative effect on the path of the eigenvalues. This can be seen in the plot of the stability region in Figure 2.4.

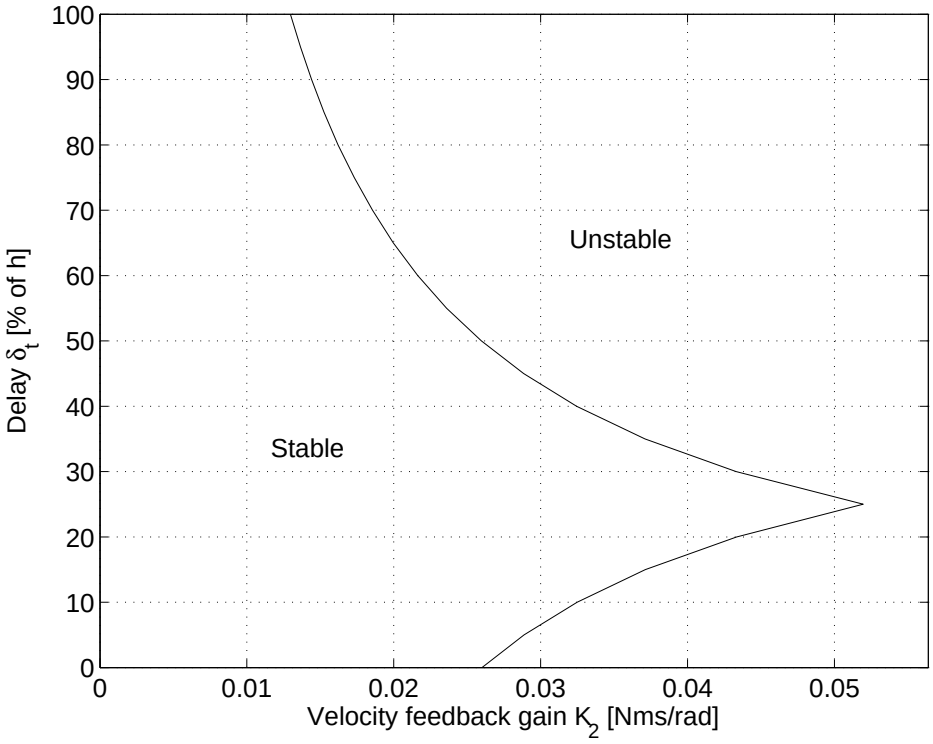


Figure 2.4: Stability region of the 2nd order system ($K_1 = 0.0282\ Nm/rad$)

Here, the maximum allowable gain K_2, that still stabilizes system (2.5), (2.9), is plotted against the time-delay, quantified in percentage of the sample time (% of h). Remarkable is that the maximum allowable gain K_2 first increases until $\delta_t = 0.25h$ and then decreases again.

To gain a complete overview of the stability region for variable K_1, K_2 and δ_t, a surface plot is made in Figure 2.5. For parameter values below the plotted surface, the system is stable. The peak at $\delta_t = 0.25h$ keeps appearing, also for larger values of K_1. Increase of the position feedback gain K_1 affects the stability only to a small extent. However, note that the performance of the system at higher gains will differ, but this is not part of this research.

To understand the peak in the stability region (see Figure 2.4) better, we study a Bode and Nyquist diagram of the system. In order to draw these graphs, a transfer function in the Z-domain must be derived. Therefore, (2.3) is rewritten:

$$\begin{aligned} \underline{x}_{k+1} &= \mathbf{\Phi}\underline{x}_k + \mathbf{\Gamma_0}\underline{u}_k + \mathbf{\Gamma_1}\underline{u}_{k-1} \\ \underline{y}_k &= \mathbf{C}\underline{x}_k, \end{aligned} \tag{2.10}$$

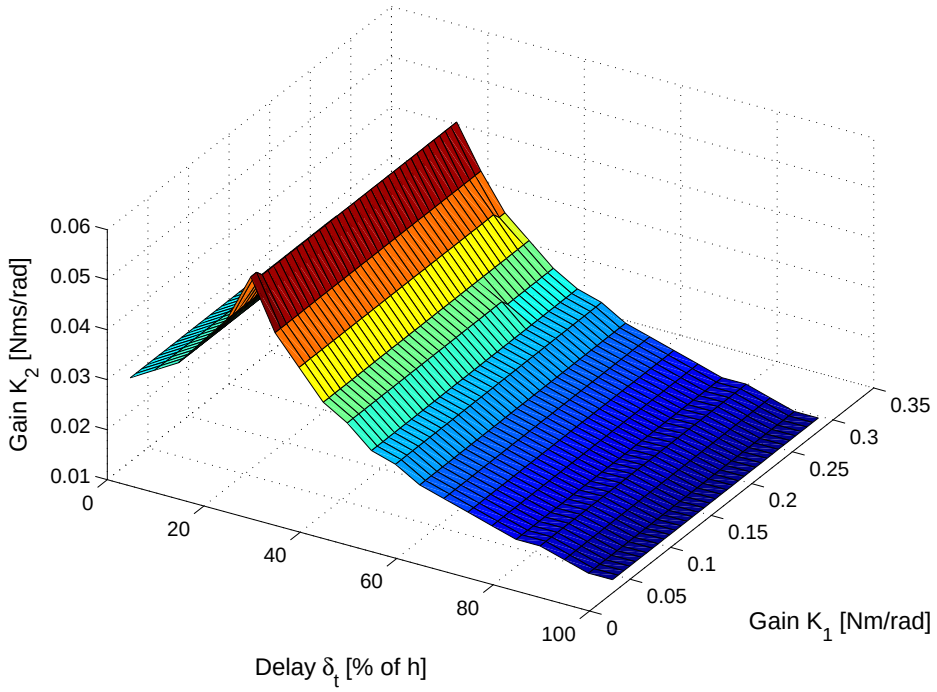


Figure 2.5: Surface plot of the stability region.

with $\mathbf{\Psi}$, $\mathbf{\Gamma_0}$ and $\mathbf{\Gamma_1}$ from (2.4). The discrete-time Z-transform of (2.10) [3] gives:

$$\begin{aligned} \underline{X}_k Z &= \mathbf{\Phi}\underline{X}_k + \mathbf{\Gamma_0}\underline{U}_k + \mathbf{\Gamma_1}\underline{U}_k Z^{-1} \\ \underline{Y}_k &= \mathbf{C}\underline{X}_k, \end{aligned} \tag{2.11}$$

where $\underline{X}_k, \underline{U}_k, \underline{Y}_k$ are the Z-transforms of $\underline{x}_k, \underline{u}_k, \underline{y}_k$, respectively.

In order to make comprehensive Bode and Nyquist diagrams, the formulation of a Single Input Single Output (SISO) system is desirable. The input u_k is the torque on the wheel and as output x_k, the angular position of the wheel, is chosen. However, for full-state feedback both position x_k and velocity v_k have to be known. Due to the fact that v_k is not measured, an estimator has to be formulated. The backward difference method gives the following velocity estimate:

$$v_k = \frac{x_k - x_{k-1}}{h}. \tag{2.12}$$

In the Z-domain (2.12) gives:

$$V_k = \frac{X_k - X_k Z^{-1}}{h} = \frac{Z-1}{Zh} X_k. \tag{2.13}$$

Now, the transfer function H_p between the control input U_k and the position output X_k can

be computed:

$$H_p = \frac{X_k}{U_k} = \frac{1}{J}\frac{\frac{1}{2}(h-\delta_t)^2 + \frac{1}{2}(h^2 - (h-\delta_t)^2)Z^{-1}}{Z - 1 - (Z-1)Z^{-1}}. \tag{2.14}$$

The transfer function of the controller H_c, from U_k to X_k, including estimator for the velocity, becomes:

$$H_c = K_1 + K_2\frac{Z-1}{Zh}. \tag{2.15}$$

The Bode plot of the open-loop transfer function H_pH_c, given (2.14) and (2.15), with $K_1 = 0.0282\ Nm/rad$ and $K_2 = 0.0564\ Nms/rad$, is depicted in Figure 2.6; the corresponding Nyquist plot is given in Figure 2.7.

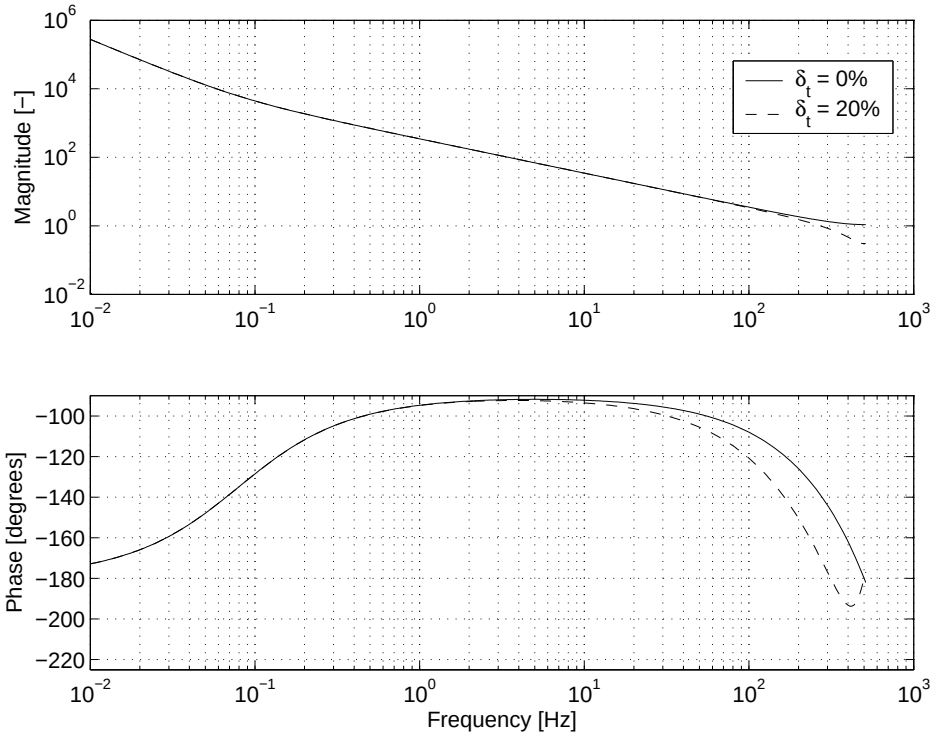


Figure 2.6: Bode diagram of the open-loop transfer functions H_pH_c for $K_1 = 0.0282$, $K_2 = 0.0564$ ($\delta_t = 0, \delta_t = 0.2h$).

The Bode diagram shows that without delay (solid line) the system is unstable. With a time-delay of $\delta_t = 0.2h$ (dashed line), the phase margin and magnitude decrease. Consequently, there is still some phase margin at a magnitude of 1 (equivalent to 0 dB), thus the system is stable.

In Figure 2.7, the same phenomenon is shown. With $\delta_t = 0$, the graph passes the point -1 at its right-hand side, implying an unstable situation. When $\delta_t = 0.2h$, the point -1 is passed at

the left-hand side, so the system is stable. When the $\delta_t = 0.4h$, the phase lag is too big to be compensated for by the decrease of magnitude, thus the system is unstable.

The Nyquist and Bode diagrams show that time-delay lowers the magnitude of the response, but also the phase in such a way, that the system is stable for higher gains for a specific time-delay. This can also be seen in Figure 2.4.

Because the transformation to the Z-domain requires the use of an estimator, the analysis of the system slightly differs from the analysis in time-domain. The stability peak can still be distinguished at $\delta_t = 0.25h$, but K_2 can be increased more, than when both position and velocity are measured.

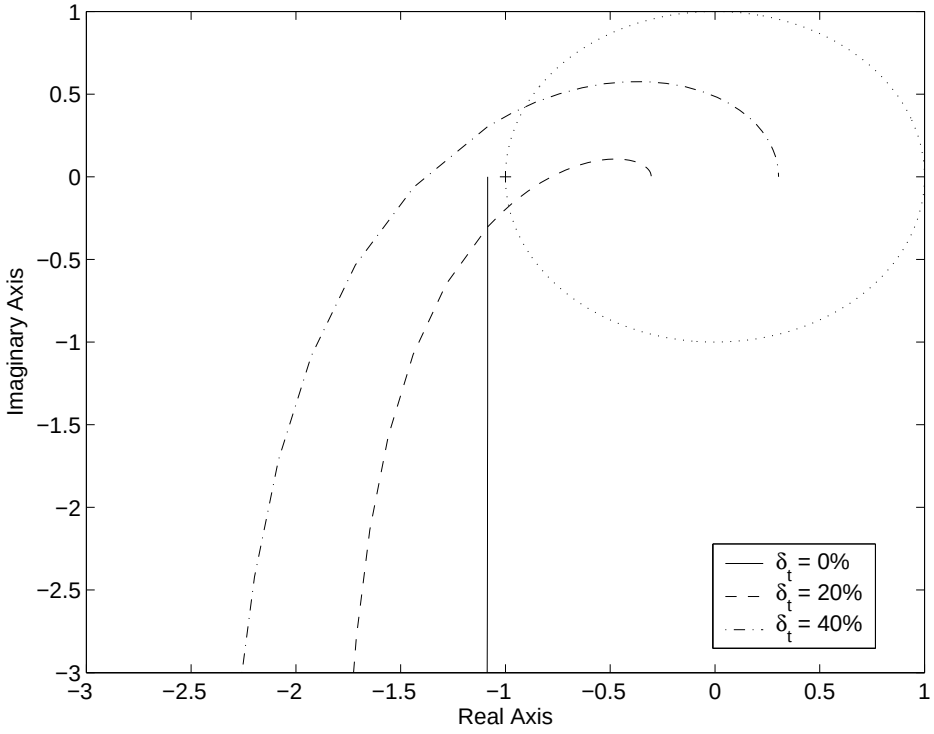


Figure 2.7: Nyquist diagram of the open-loop transfer functions H_pH_c for $K_1 = 0.0282$ Nm/rad, $K_2 = 0.0564$ Nms/rad ($\delta_t = 0, \delta_t = 0.2h, \delta_t = 0.4h$).

Chapter 3

Choice of an experimental setup

To validate the stability analysis of section 2.2, experiments will be performed. In this chapter, the choice of a suitable setup will be described. Two setups, Trilobot and a PATO setup, will be compared, based on the following criteria:

1. The extent to which it represents a real NCS,
2. The ease of use,
3. Reliability,
4. Accuracy.

These points of interest are crucial for obtaining good measurements and to ensure that the chosen setup truly represents an NCS. Advantages and disadvantages will be discussed and finally one setup will be chosen to conduct the experiments and investigate the stability region.

3.1 The Trilobot mobile robot

Trilobot is a two-wheeled robot, which can steer by letting one of the two wheels turn faster or slower than the other wheel. The robot can also avoid obstacles by using whiskers, detect the presence of moving objects with sonar and has many more special features. It can be programmed by copying programs into its RAM (Random Access Memory), using a wireless infrared link or a serial communication cable. This feature is very useful for controlling the robot. Trilobot has already been used for research on an NCS [5]. Figure 3.1 shows a picture of Trilobot.

Representation of a real NCS

Trilobot can be considered as an embedded system, because it has its own processor and operating system. In practice, a Networked Control System is controlled over a network and operates with its own hard- and software.

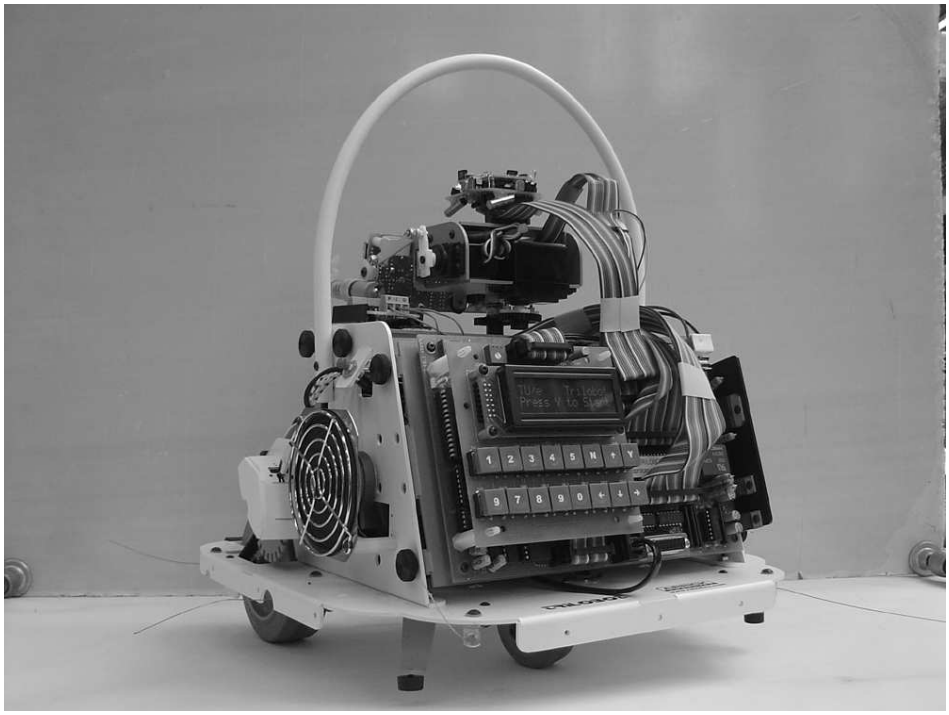


Figure 3.1: The Trilobot mobile robot.

Ease of use

In previous research, a thorough study on Trilobot and its possibilities resulted in the design of a communication link between MATLAB/Simulink, Microsoft Windows and the Trilobot operating system [1], [4]. Simulink is used to implement the controller and obtain measurement data. An advantage of the Trilobot setup is that it is already adapted for stability measurements on systems with delay [5]. It is adjusted to use only one wheel, which simplifies the analysis and measurements. Using two wheels would make the system more complex, which is undesirable in this case, because the current work only requires a simple plant and controller.

Accuracy

To implement a full state-feedback controller, both the position and velocity have to be known at the beginning of every sample interval. The position, obviously, can be read out from the encoder signal, measuring the angular position of the wheel. The velocity has to be estimated. In the program files, the estimator is stated as follows [5]:

$$\underline{\hat{x}}(k+1) = \mathbf{\Phi}\underline{\hat{x}}(k) + \mathbf{\Gamma}u(k) + L[y(k) - \mathbf{C}\underline{\hat{x}}(k)]. \tag{3.1}$$

The velocity is estimated using knowledge of the system matrices $\mathbf{\Phi} = e^{Ah}$ and $\mathbf{\Gamma} = \int_0^h e^{\mathbf{A}\eta} d\eta \mathbf{B}$. L is the column of estimator gains, which can be tuned to obtain adequate results. Both $\mathbf{\Phi}$ and $\mathbf{\Gamma}$ are dependent of the sample time, but in fact $\mathbf{\Gamma}$ is also influenced by the delay δ_t. In (2.4) the correct formulation for $\mathbf{\Gamma}$ is presented. This means that the

estimator has to be altered when the delay is altered, to accurately estimate the velocity. This has to be done in C-files, so every time the delay changes, the file has to be compiled again. This is very time-consuming and thus slows down the measurement process.

In (2.3), the discrete-time equations of motion are presented. The estimator in (3.1), implemented in the program files, uses these equations with $\delta_t = 0$. A proper estimator in which the time-delay is also accounted for, would be stated as follows:

$$\underline{\hat{x}}(k+1) = \mathbf{\Phi}\underline{\hat{x}}(k) + \mathbf{\Gamma_0}u(k) + \mathbf{\Gamma_1}u(k-1) + L[y(k) - \mathbf{C}\underline{\hat{x}}(k)], \tag{3.2}$$

with

$$\mathbf{\Gamma_0} = \int_0^{h-\delta_t} e^{\mathbf{A}\eta} d\eta \mathbf{B} \quad \text{and} \quad \mathbf{\Gamma_1} = \int_{h-\delta_t}^{h} e^{\mathbf{A}\eta} d\eta \mathbf{B}.$$

This estimator depends on δ_t. Furthermore, both the input $u(k)$ and $u(k-1)$ have to be known. Therefore samples have to be saved in the memory, which requires changes of the Trilobot program files. Furthermore, the estimator gains in L may have to be altered when the time-delay changes. Learning how to program this in C and C++ is not part of this internship.

Reliability

During test measurements, it was noticed that the measurements were not reproducible. The estimator of the velocity (which in principle already is wrong), gave different results for equal experiments. Possibly some data, sent to the robot, is corrupted or influences subsequent experiments. Another reason can be that the encoder has different starting points when initiating an experiment, which results in differences between seemingly similar experiments.

Summary

Considering the problems, explained in previous paragraphs, Trilobot is not a suitable setup for our experiments. The lack of reproducibility is a major drawback. Though it seems easy to use, the difficulty to change the program files is a disadvantage. With help from the designer of the programs a lot of problems were fixed, but it remains difficult to adjust the program files to our own wishes. An advantage is that Trilobot shows a good resemblance with a real NCS.

3.2 The PATO experimental setup

The PATO setup is often used for educational purposes by the Dynamics and Control Technology (DCT) group. The system consists of two servomotors with encoders and a mass mounted on both driveshafts. The servomotors can be connected by a belt. For our work, the belt is removed, thus only one motor is used. Figure 3.2 shows a picture of the PATO setup.

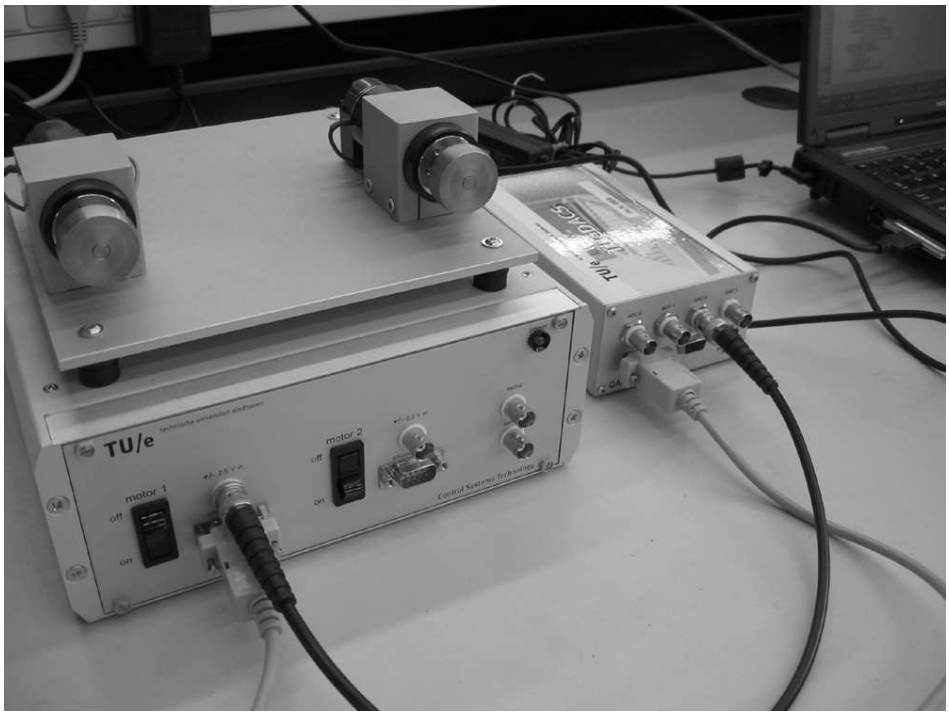


Figure 3.2: The PATO setup.

Representation of a real NCS

Because the PATO setup is directly controlled by the Linux system, it is not considered to be an embedded system. Introducing an artificial time-delay is the only way to make it look like an NCS.

Ease of use

The system is controlled by a controller running on a Linux system. The communication is made possible by a TUeDACS system with a serial connection via a PCMCIA slot. Again MATLAB Simulink is used to implement the controller. A Simulink model, containing some control options and an option to measure transfer functions, is available. The system is controlled by a PD-controller with a lowpass filter. The lowpass filter alternates high-frequency measurement noise. The system has an adjustable sample rate and the break point of the lowpass filter is placed at 200 Hz. The PATO setup is easy to use, because of all these standard features and the use of software, with which the students of the DCT group are familiar.

Accuracy

The PATO setup does not use a velocity estimator that is dependent on the system matrices or the time-delay. It uses the backward difference method, which is a sufficient estimator of the velocity. The analytical results, though, will be different, because in the eigenvalue analysis the exact velocity is used.

Reliability

The PATO setup is being used often for educational purposes and no problems were reported concerning the reliability of the system.

Summary

An advantage of the use of the PATO setup is that it is less difficult to change the configuration and start experiments, compared to Trilobot. This makes it very easy to use. It does not represent a real NCS very well, but the accuracy is sufficient and it is known to be reliable.

3.3 Discussion

Based on the descriptions of the setups and the comparison of the criteria, presented in section 3.1 and 3.2 a final choice will be made. The Trilobot setup is interesting because of its resemblance with a real NCS. The existence of previous research and available program files are an advantage [4] [5], but a closer look to the files showed the shortcoming of the setup. Reliability is poor and the difficulty to adjust the program files is a big drawback.

On the other hand, the PATO setup does not represent a real NCS, but its reliability, ease of use and accuracy seem good. Therefore, the PATO setup is chosen to conduct the experiments. However, to prepare the setup for measurements on the stability of an NCS with constant time-delay, some problems have to be solved. These will be discussed in the next chapter, along with the analytical results and the final experiments.

Chapter 4

Analysis of the PATO setup

In this chapter, further analysis on the PATO setup will be performed. For simulation and controller design purposes, the system parameters will be estimated. The stability analysis, explained in Chapter 2, is adapted for this particular system and the measurements are compared with these simulations.

4.1 Estimation of the system parameters

The system parameters are estimated, based on a frequency response function (FRF) measurement. To measure an FRF, random noise is added to the system (see Figure 4.1). The Sensitivity function (S) and the Complementary Sensitivity function (CS) can be measured. Since S is the transfer function from the noise w to input u and CS is the transfer function from w to the error e, the plant FRF H can be computed by $H = \frac{CS}{S}$. A schematic representation of the system, used for measuring the FRF, is shown in Figure 4.1.

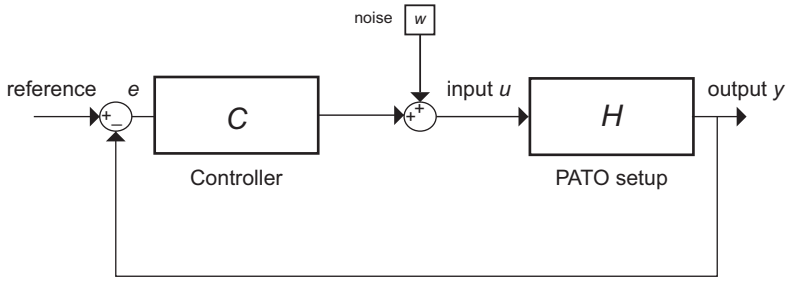


Figure 4.1: Schematic representation of the FRF measurement setup.

Before computing the inertia, a unit conversion is necessary. In the experimental setup, the controller applies a voltage V to the system, which is converted to a current I and finally to a torque T, which is applied to the driveshaft. In the simulations it is assumed, that the controller can directly apply a torque to the wheel. So the input u has different units in simulation and experiment, namely Nm in simulations and V in experiments. To compare the simulations with the experiments and to compute the inertia J in Nm, the conversion factor c from V to Nm must be known. To measure the voltage to current conversion factor c_v, a constant voltage is applied to the system and the current at the motor is measured. This gives the voltage to current graph, shown in Figure 4.2.

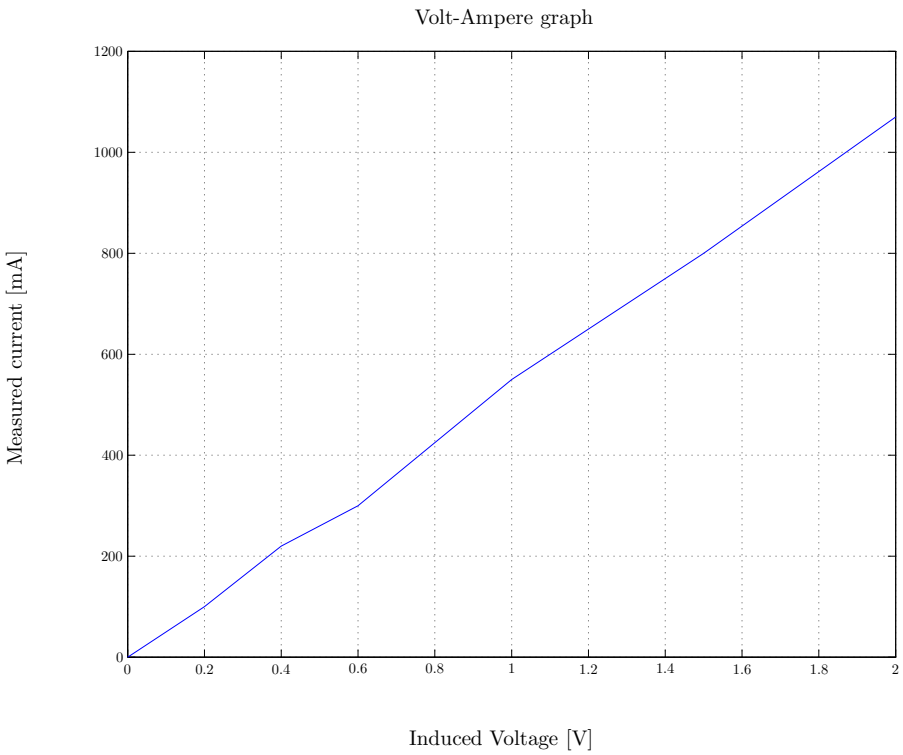


Figure 4.2: Volt-Ampere graph of the motor.

Determining the slope of the graph, using the least squares method, gives $c_v = 537\ mA/V$. The torque constant c_T is obtained from the data sheet of the servomotor (Appendix B) and yields $c_T = 52.5 \cdot 10^{-3}\ Nm/A$. This gives a total conversion factor $c = c_v c_T = 28.19 \cdot 10^{-3}\ Nm/V$.

Since the encoder signal is read out in [rad], and the controller action satisfies $u_k = K_1 x_k + K_2 \dot{x}_k$, the units of K_1 and K_2 are respectively [V/rad] and [Vs/rad]. In the simulations, these gains have to be converted as well, to be comparable to the gains in the experiments.

Now, the Sensitivity function and Complementary Sensitivity function are measured. These measurements are given in Appendix A. When noise is chosen with a variance $w = 0.5$, the sample rate is 4 kHz and the controller gains $K_1 = 0.0282\ Nm/rad$ and $K_2 = 0.0014\ Nms/rad$, the results as shown in Figure 4.3 are obtained.

Clearly, in the area where the coherence is acceptable, a -2 slope can be distinguished. In this part of the frequency domain, the system behaves like a double integrator (an inertia). A double integrator is formulated as follows:

$$H_p(j\omega) = \frac{1}{-J\omega^2} \tag{4.1}$$

where ω is the frequency in rad/s and J the inertia, which can be estimated. Because the computed FRF $H_p(j\omega)$ contains information about the amplitude $|H(j\omega)|$ at certain frequency

ω, the inertia can be determined as follows:

$$|H(\omega)| = \frac{1}{J\omega^2} \tag{4.2}$$

$$J = \frac{1}{|H(\omega)|\omega^2}. \tag{4.3}$$

This results in an estimation of $J = 1.30 \cdot 10^{-5}\ kgm^2$. When the inertia is computed using the standard inertia formulation for cylindrical bodies, $J = \frac{1}{2}mR^2$ and the inertia of the servo-motor (Appendix B) is added, the total inertia is $J = 1.26 \cdot 10^{-5}\ kgm^2$, which is comparable to the estimated value. Note that at low frequencies, where the coherence is not acceptable, the system may behave differently and some damping or friction may influence the dynamic behavior. This, however, can not be concluded from the measurements.

Zero-order hold discretization of a continuous-time system already induces some phase lag and consequently a delay. Multiplying (4.1) with $e^{\frac{-j\omega h}{2}}$ accounts for this delay (with sample-time h) [2]. When the phase of the plant, computed from the FRF measurements, is smaller or equal to the modeled phase, one can conclude that no extra delay is present in the system. In Figure 4.3 the FRF, obtained from the measurements, is presented. Figure 4.4 shows the phase plot of Figure 4.3 and the phase of the FRF, computed with the model of (4.1) (including the discretization delay).

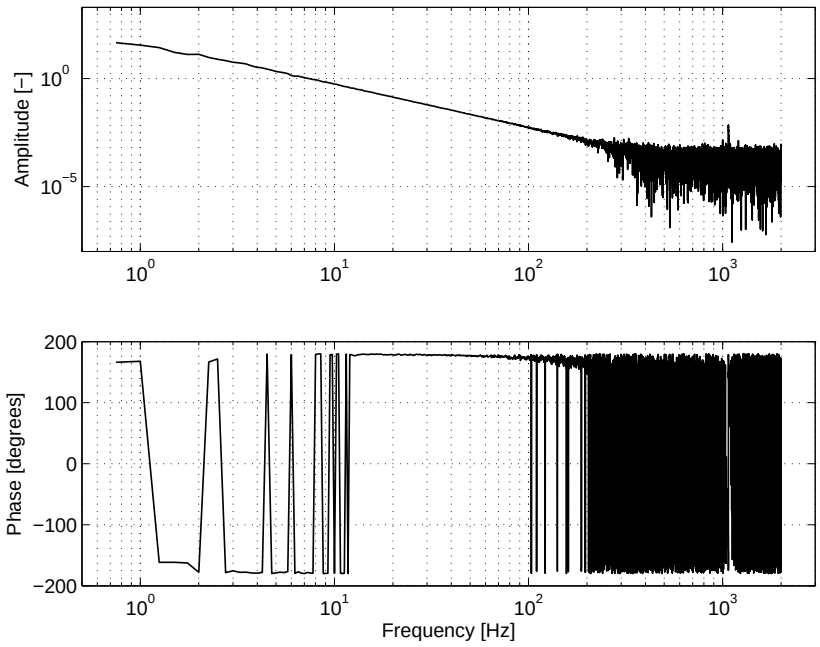


Figure 4.3: FRF between u_k and x_k of the PATO setup, obtained from measurements.

In Figure 4.4 can be seen that the measured time-delay is larger than the delay, due to discretization. This implies that the system suffers from more time-delay than assumed in the model. This will influence the stability properties.

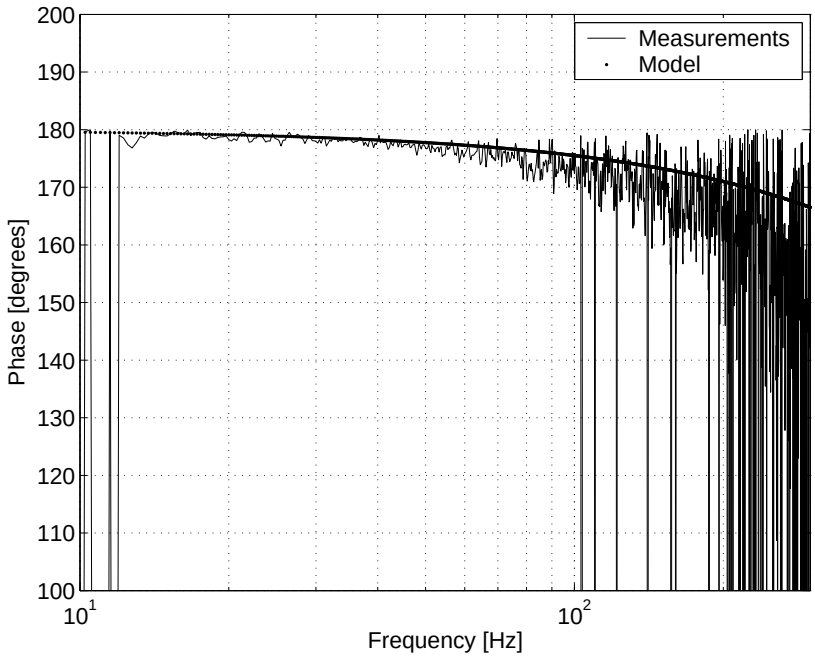


Figure 4.4: Phase plots of the modeled and the measured FRFs of the plant AT 4 kHz.

4.2 Implementation of constant time-delays

As stated in Chapter 2, the time-delays δ_t are assumed to be constant and smaller than the sample time h. This implies that a standard Simulink unit delay block cannot be used. To solve this problem, two sample frequencies are used. Simulink runs at 5 kHz, while the plant is sampled effectively at 1 kHz. Dividing the 1 kHz signal of the plant into pieces with a sample rate of 5 kHz, makes it possible to use a Simulink block, that delays a sample for one sample interval. A schematic representation of this method is shown in Figure 4.5.

A sampling interval of 1 ms (sampled at 1 kHz) is divided in pieces of 0.2 ms. Since Simulink runs at 5 kHz, a unit delay induces a delay of 0.2 ms. When the plant samples at 1 kHz, this means the samples are delayed for 20% of their sample rate. For higher amounts of delay, additional unit delay blocks can be added.

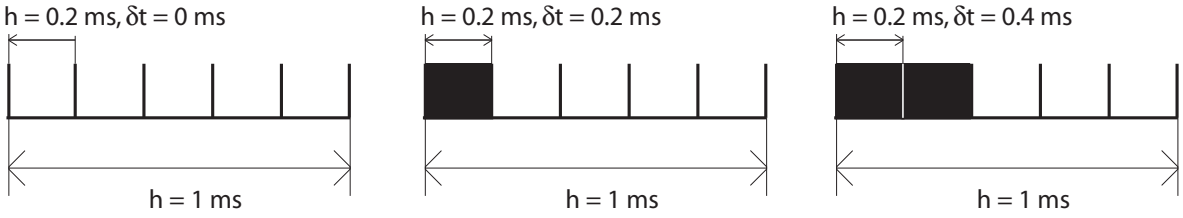


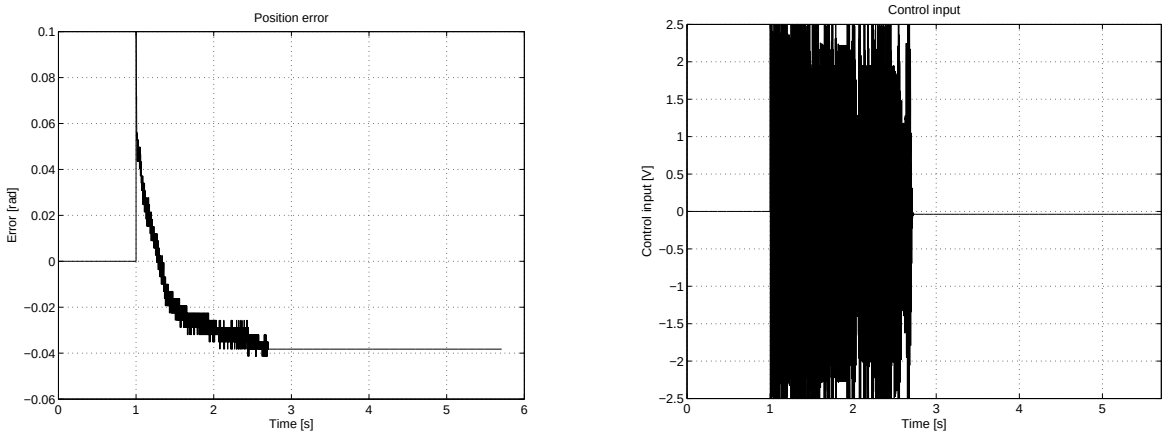
Figure 4.5: Implementation of constant time-delay on the plant.

Note that the Trilobot setup uses the same principle, embedded in the available program files.

4.3 Experimental Results

To validate the analytical results, experiments are conducted. A step function of 0.1 rad in the position is induced as a reference profile for the motor. For five different values of time-delay δ_t, the maximum allowable gain, that still induces stable behavior in the setup, is found and is then compared to the model results. The eigenvalue-based stability analysis for this system is already presented in section 2.2.2, since the inertia J and the conversion factor c from section 4.1 are used there. To explain how the stability border is determined, two situations are shown in Figures 4.6 and 4.7. The position error e_k and the control input u_k are plotted in time. When the error remains constant, the system is considered stable. When the error keeps increasing or keeps oscillating and the controller input u_k keeps oscillating, the system is considered unstable.

Figure 4.6 is considered to be a stable situation, while Figure 4.7 is considered to be unstable. However, taking a closer look at the control input in Figure 4.6, we see that the control input is not zero, while the system has stopped moving. Apparently, friction is present in the system, so the stablity is difficult to assess.



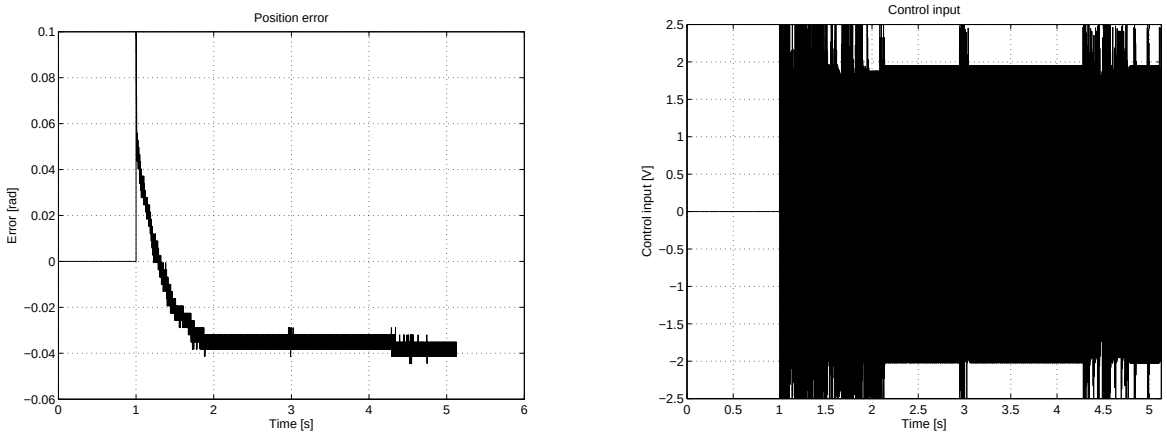
(a) Position error

(b) Control input

Figure 4.6: $\delta_t = 0$, $K_1 = 0.0282\ Nm/rad$, $K_2 = 0.006907\ Nms/rad$, $h = 1$ ms.

In Figure 4.7, we can see that the error is not constant, and that the controller is heavily reacting on this error. The presence of noise in the system can be the reason for this behavior. All measurement data can be found in Appendix C. The resulting measured stability region, compared to the model, is shown in Figure 4.8. Clearly, it can be seen that the measurement results and the model results are not comparable. The measured maximum allowable gain is much lower than the one found for the model. The possible reasons for this mismatch between the model and the experiments are listed here:

1. The accuracy of the system parameters is insufficient;
2. The real setup is not modeled correctly;
3. Static friction in the system.



(a) Position error

(b) Control input

Figure 4.7: $\delta_t = 0$, $K_1 = 0.0282\ Nm/rad$, $K_2 = 0.006935\ Nms/rad$, $h = 1$ ms.

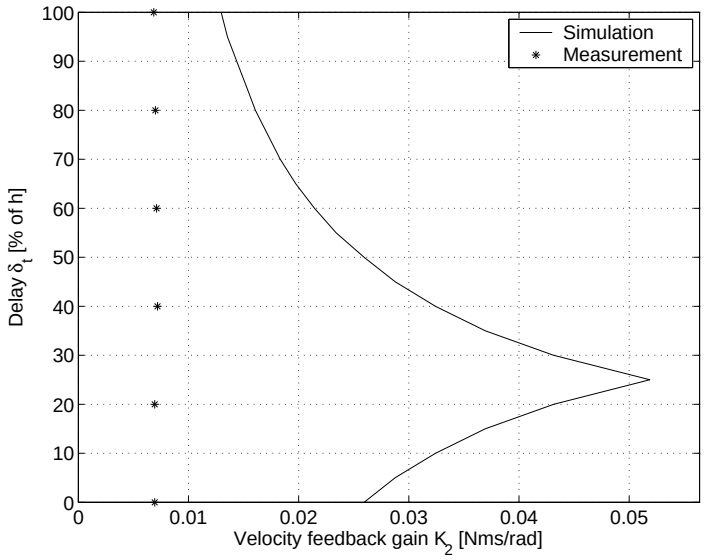


Figure 4.8: Comparision of measured and model-based stability region ($K_1 = 0.0282\ Nm/rad$)

Accuracy

The inertia is estimated with measurement data. Still the system may suffer from friction and damping in the low-frequency area, where the measurements are not accurate enough. This can influence the stability properties of the system such that the simulations do not match the measurements.

The velocity is also estimated. During the experiments the velocity is computed real-time, based on the current and previous sample of the position measurement. This method may be not accurate enough, so the velocity, used in the feedback loop is not the real velocity. During experiments, adjusting the lowpass filter may lead to a more accurate estimation of

the velocity.

Model versus setup

As stated above, a velocity estimator and lowpass filter are used during the experiments. Since a lowpass filter introduces extra phase lag in the system, it will influence the stability properties. The use of a velocity estimator may also influence the stability properties. In the model, we can add this estimator, in order to give comparable results. Adding the lowpass filter in the model may also lead to a better comparison with the experiments.

Determining stability

To determine whether the system is stable under certain conditions, a step function in the position is used as reference signal. When the response of the system results in a constant small error, the system can considered to be stable. The stability region shown in Figure 4.8, was determined in this way. However, a closer look at the results in Appendix C shows that the controller input is not exactly zero. This can be explained by the fact that the system suffers from friction and the controller input force is in equilibrium with the friction force. This makes it difficult to determine whether the system is stable or not.

Apparently, a step response is not a suitable choice to determine stability in this case. The static friction may be larger than the control forces due to the position error, which makes it impossible to achieve accurate measurements on the stability border. The use of a constant velocity as a reference trajectory may give better results.

Chapter 5

Conclusions and recommendations

In this chapter, conclusions will be stated and some recommendations for further research will be presented.

5.1 Conclusions

A model of an NCS with constant time-delays is formulated. A single motor with an encoder, which is modeled as a second-order system, is used to analyze the stability properties of the NCS with constant time-delays. The stability region of an NCS is determined analytically, based on a discrete-time model.

Eigenvalue analysis and analysis in the frequency domain showed a remarkable shape of the stability region. First the stability range (in terms of the control gain) increases for increasing delay and after a certain delay it decreases for increasing delay. This can be explained by the fact that delay influences both the gain and phase of the discrete-time system.

To validate these results, an experimental setup is chosen. Two setups, Trilobot and the PATO experimental setup, were compared on basis of the following criteria:

1. The extent to which it represents a real NCS;
2. The ease of use;
3. Reliability;
4. Accuracy.

The Trilobot setup is a good representation of a real NCS, but is not suitable as experimental setup, due to its complicated communication structure, its poor reliability and inaccuracy. The PATO setup is easier to use, because of the better knowledge of the software. It is known to be reliable and accurate, but does not represent a real NCS. The PATO setup is chosen to conduct the experiments.

The system parameters of the PATO setup are determined based on the frequency response measurements. Unfortunately, the stability region obtained experimentally and analytically

are not comparable. Probably the accuracy or the method to determine the stability experimentally cause a mismatch between the experiments and the analysis. Moreover, the estimator of the velocity and the lowpass filter, present in the experiments, are not modeled, so the model may not be a good representation of the real setup. Using a step function as reference signal proved to be not a suitable choice, because the system suffers from static friction.

5.2 Recommendations

To obtain good experimental results, the PATO setup should be investigated further. The method of implementing the delay must be refined in order to have a smaller step in the time-delay. Then more measuring points are available.

The measurements to determine whether the system is stable or not are not reliable. As stated above, a step function as position reference trajectory is not a suitable choice. A constant reference velocity is a better choice.

Furthermore, the model of the system must be reconsidered. Both the velocity estimation and lowpass filter need to be taken into account when the analytic stability region is determined on a model level.

Investigating the performance properties, like bandwith or settling time, at different amounts of time-delay and controller gains is an interesting addition to this research.

Bibliography

[1] Roger Arrick: *Trilobot, Mobile Robot for Research and Education: User Guide* Arrick robotics, Hurst, Texas, USA (September 1998).

[2] Karl J. Åström and Bjorn Wittenmark: *Computer Controlled Systems: Theory and Design*, Prentice Hall (1990).

[3] Gene F. Franklin, J. David Powell and Abbas Emami-Naeini: *Feedback Control of Dynamic Systems, Third Edition*, Addisson-Wesley Publishing Company (1995).

[4] H.L.Hagenaars: *A trilobot interface in matlab/simulink*, Technical Report DCT 2004.64, University of Technology Eindhoven (July 2004).

[5] H.L. Hagenaars: *Stability Analysis of Sampled-Data Systems with Network Delays*, Technical Report DCT 2005.136, University of Technology Eindhoven (2005).

[6] Johan Nilsson: *Real-Time Control Systems with Delays*, Phd Thesis, Department of Automatic Control, Lund Institute of Technology, Sweden (1998).

[7] Wei Zhang, Michael S. Branicky, and Stephen M. Phillips: *Stability of Networked Control Systems*, IEEE Control Systems Magazine, (February 2001).

Appendix A

Frequency Response Functions

Both the Sensitivity and Complementary Sensitivity functions of the PATO-setup are presented here. During the measurements, the noise has a variance $w = 0.5$, the sample rate is 4 kHz and the controller gains are $K_1 = 0.0282\ Nm/rad$ and $K_2 = 0.0014\ Nms/rad$.

Figure A.1 shows the Sensitivity function, which is reliable above 7 Hz, as can be concluded from the coherence. Figure A.2 shows the Complementary Sensitivity function, which is reliable below 100 Hz.

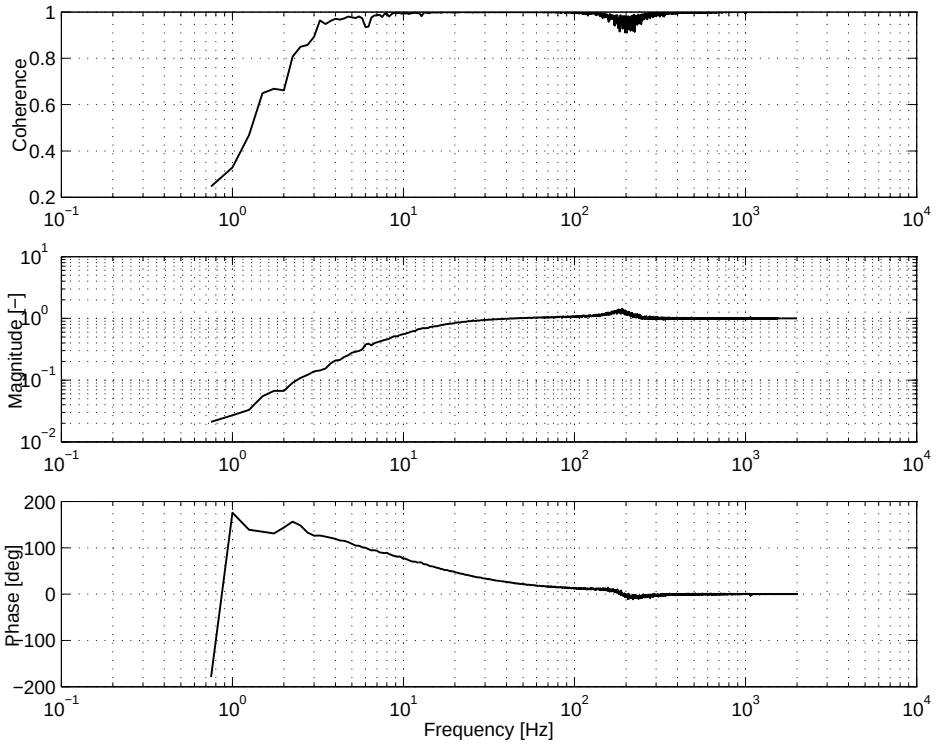


Figure A.1: The measured sensitivity function.

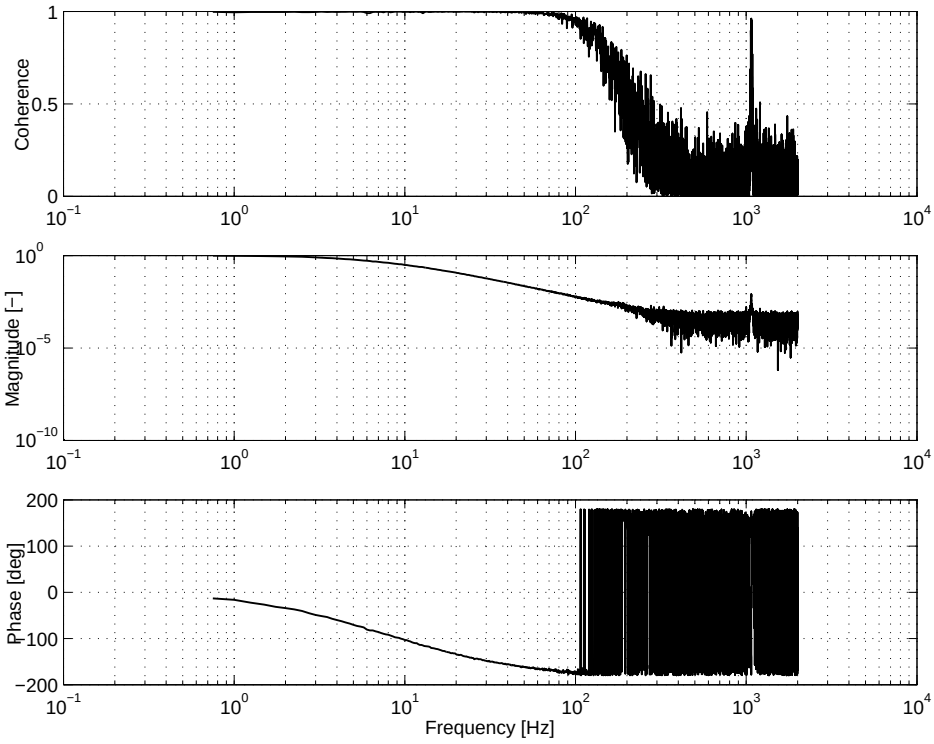


Figure A.2: The measured complementary sensitivity function.

Appendix B

Datasheet of Maxon motor 118778

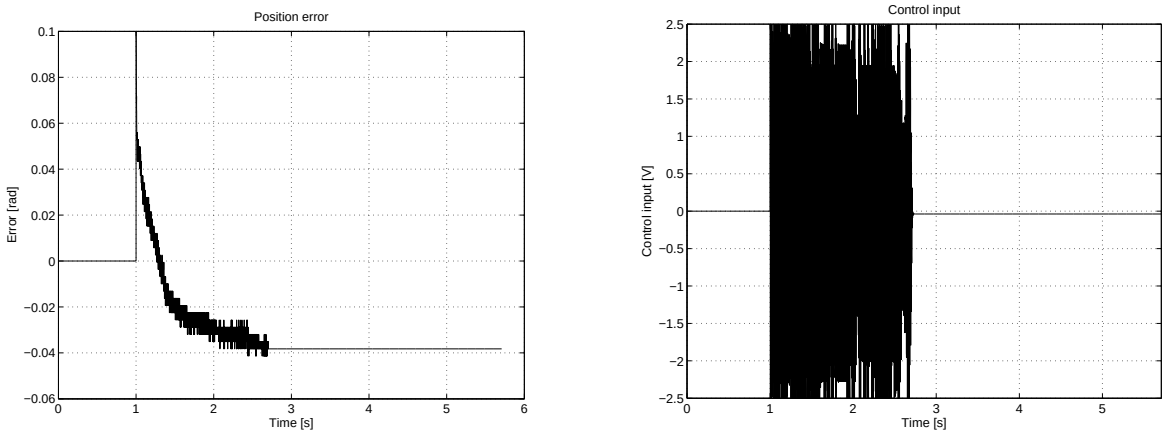
Stock program
Standard program
Special program (on request!)

			Order Number		
			118776	118777	118778
Motor Data					
1	Assigned power rating	W	90	90	90
2	Nominal voltage	Volt	15.0	30.0	42.0
3	No load speed	rpm	7070	7220	7530
4	Stall torque	mNm	872	949	1070
5	Speed/torque gradient	rpm/mNm	8.45	7.77	7.17
6	No load current	mA	245	124	93
7	Starting current	A	44.9	24.4	20.3
8	Terminal resistance	Ohm	0.334	1.23	2.07
9	Max. permissible speed	rpm	8200	8200	8200
10	Max. continuous current	A	4.00	2.74	2.15
11	Max. continuous torque	mNm	77.7	107	113
12	Max. power output at nominal voltage	W	152	175	206
13	Max. efficiency	%	81	84	86
14	Torque constant	mNm/A	19.4	38.9	52.5
15	Speed constant	rpm/V	491	246	182
16	Mechanical time constant	ms	6	5	5
17	Rotor inertia	gcm²	65.5	65.5	69.6
18	Terminal inductance	mH	0.09	0.34	0.62
19	Thermal resistance housing-ambient	K/W	6.2	6.2	6.2
20	Thermal resistance rotor-housing	K/W	2.0	2.0	2.0
21	Thermal time constant winding	s	27	27	29

Appendix C

Measurement results

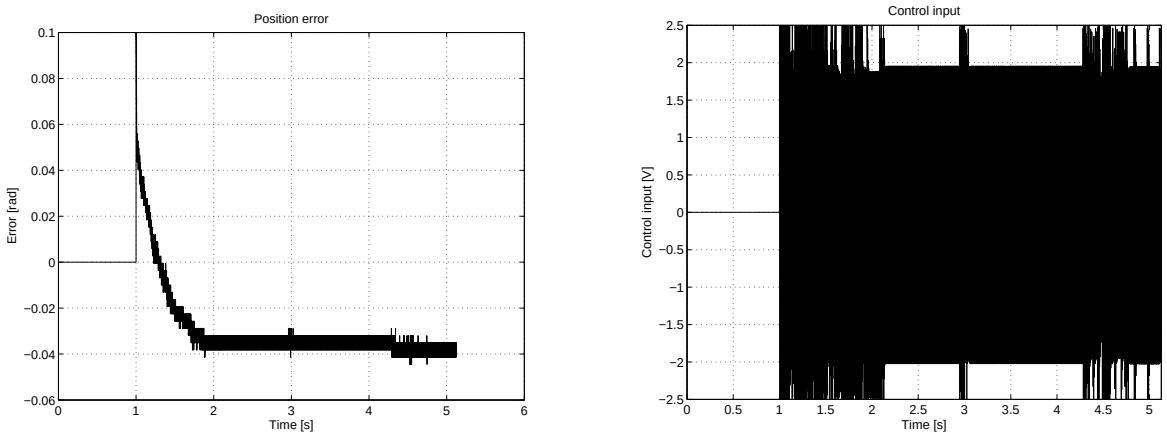
The following figures are the results of measurements on the PATO setup. For time-delays $\delta_t \in [0, 1]$ ms, the maximum allowable gain K_2 for which the system is stable is determined. For every amount of time-delay, measurements of the stable and the unstable situation are presented. The position error e_k and control input u_k are plotted in time.



(a) Position error

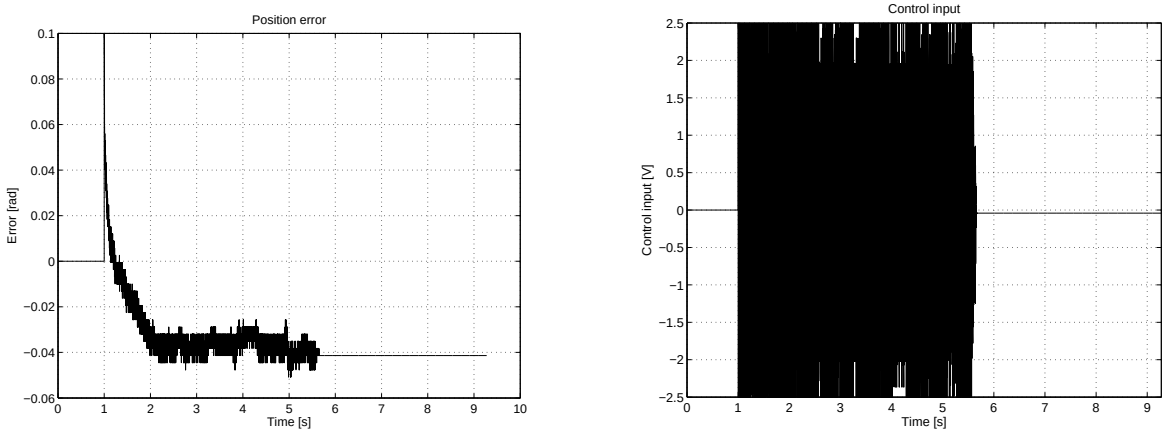
(b) Control input

Figure C.1: $\delta_t = 0$, $K_1 = 0.0282$ Nm/rad, $K_2 = 0.006907$ Nms/rad, $h = 1$ ms.



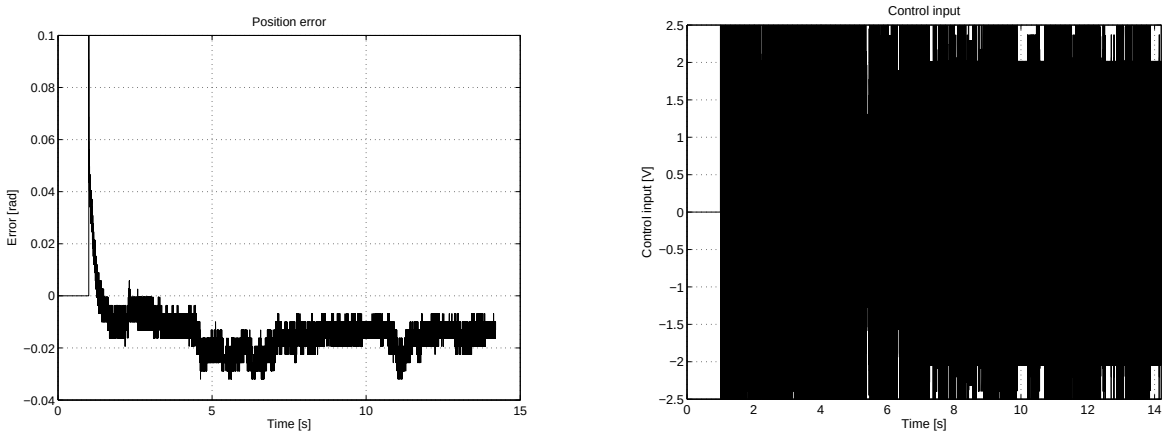
(a) Position error (b) Control input

Figure C.2: $\delta_t = 0$, $K_1 = 0.0282\ Nm/rad$, $K_2 = 0.006935\ Nms/rad$, $h = 1$ ms.



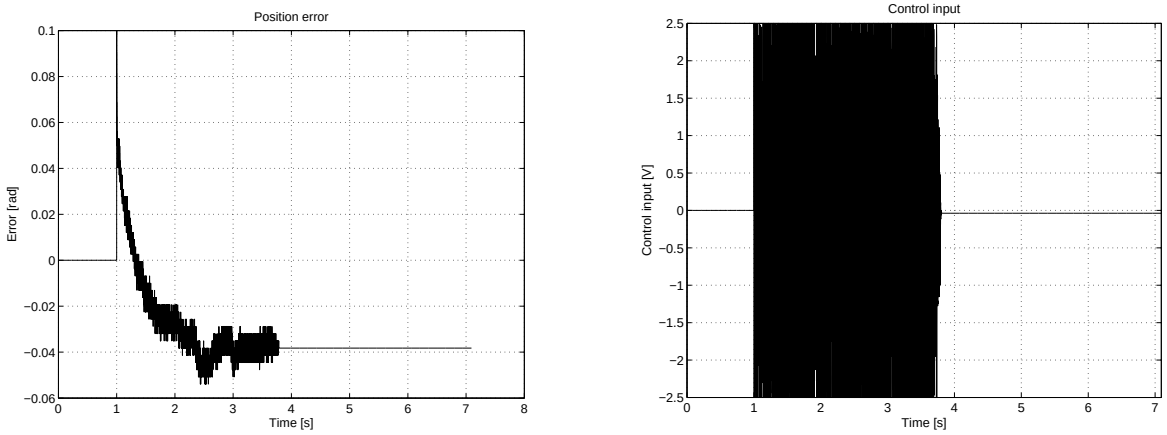
(a) Position error (b) Control input

Figure C.3: $\delta_t = 0.2h$, $K_1 = 0.0282\ Nm/rad$, $K_2 = 0.006935\ Nms/rad$, $h = 1$ ms.



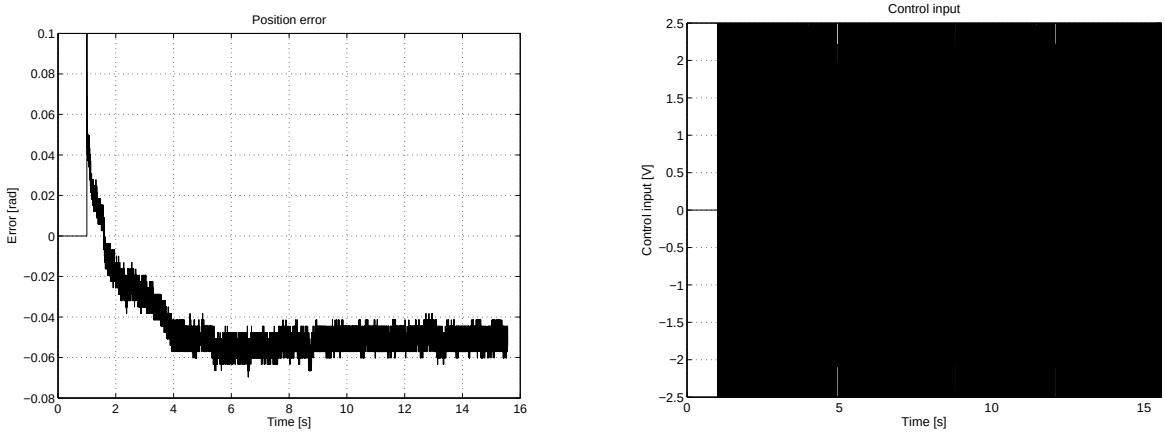
(a) Position error (b) Control input

Figure C.4: $\delta_t = 0.2h$, $K_1 = 0.0282\ Nm/rad$, $K_2 = 0.006964\ Nms/rad$, $h = 1$ ms.



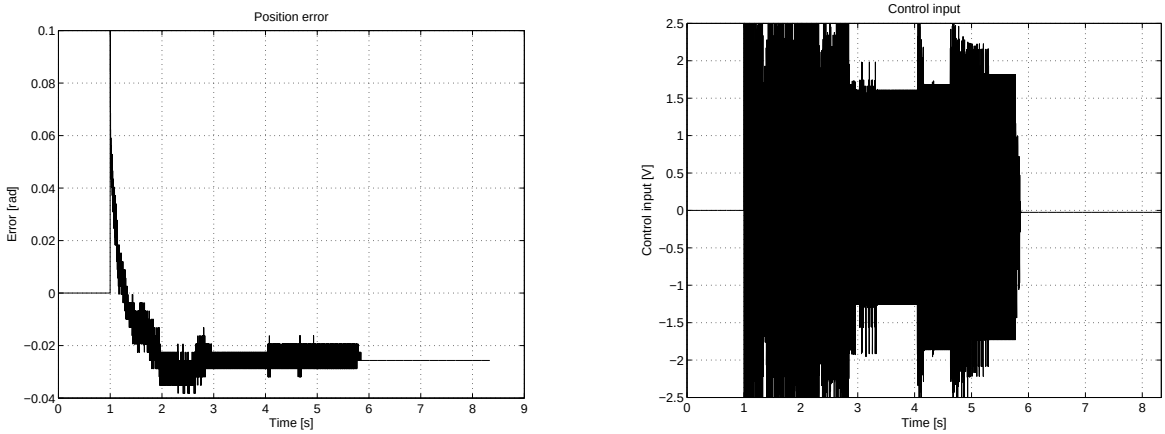
(a) Position error (b) Control input

Figure C.5: $\delta_t = 0.4$, $K_1 = 0.0282\ Nm/rad$, $K_2 = 0.007189\ Nms/rad$, $h = 1$ ms.



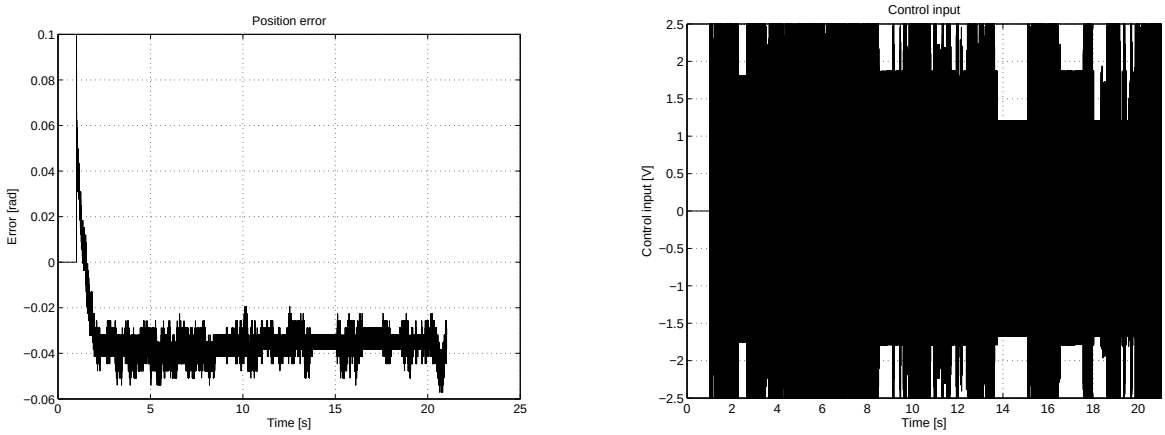
(a) Position error (b) Control input

Figure C.6: $\delta_t = 0.4$, $K_1 = 0.0282\ Nm/rad$, $K_2 = 0.007217\ Nms/rad$, $h = 1$ ms.



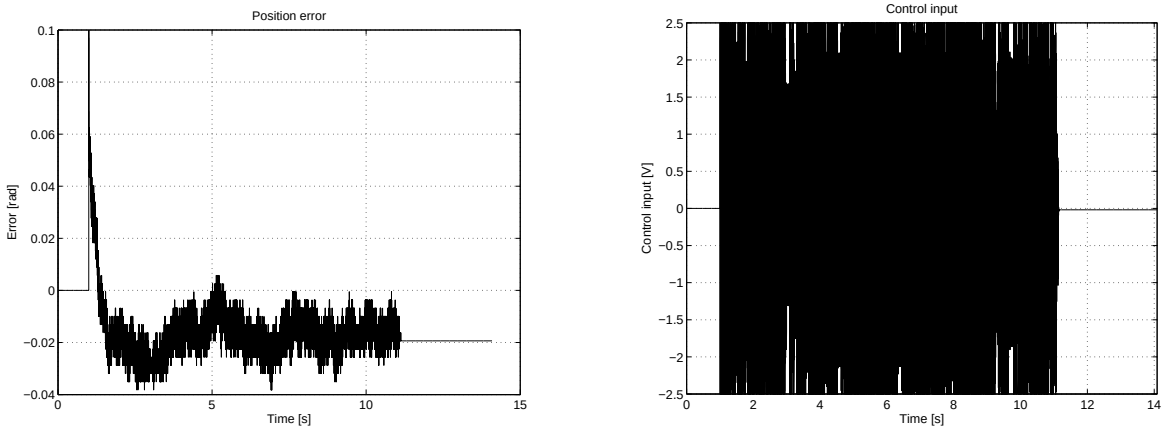
(a) Position error (b) Control input

Figure C.7: $\delta_t = 0.6h$, $K_1 = 0.0282\ Nm/rad$, $K_2 = 0.007105\ Nms/rad$, $h = 1$ ms.



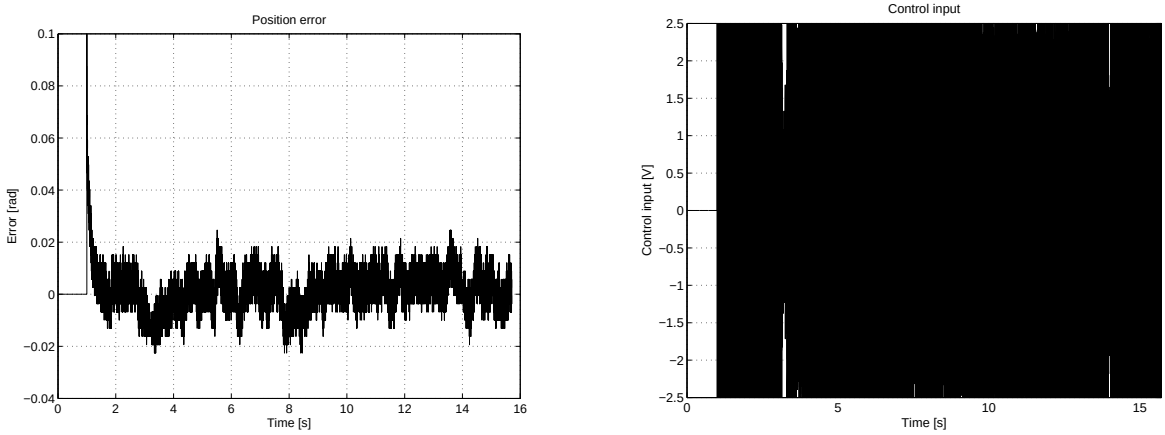
(a) Position error (b) Control input

Figure C.8: $\delta_t = 0.6h$, $K_1 = 0.0282\ Nm/rad$, $K_2 = 0.007133\ Nms/rad$, $h = 1$ ms.



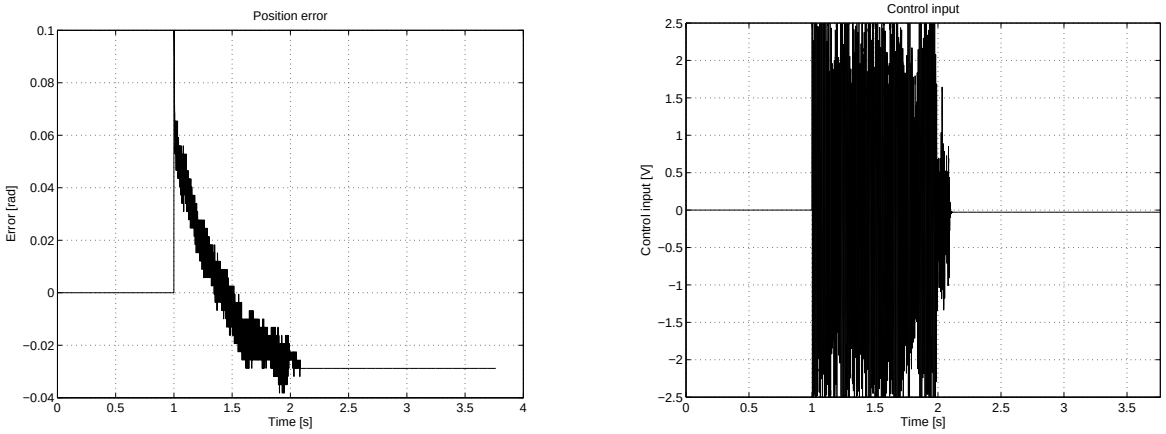
(a) Position error (b) Control input

Figure C.9: $\delta_t = 0.8$, $K_1 = 0.0282\ Nm/rad$, $K_2 = 0.006992\ Nms/rad$, $h = 1$ ms.



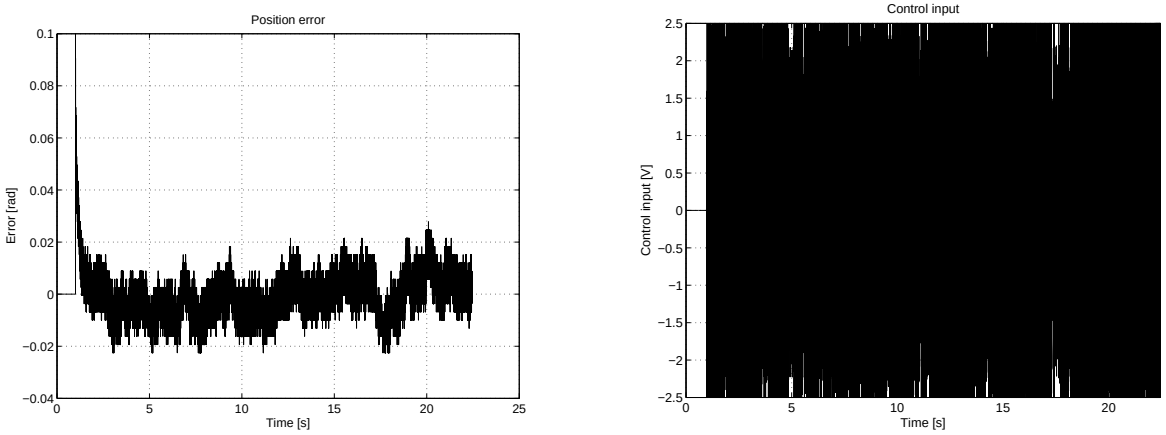
(a) Position error (b) Control input

Figure C.10: $\delta_t = 0.8$, $K_1 = 0.0282\ Nm/rad$, $K_2 = 0.007020\ Nms/rad$, $h = 1$ ms.



(a) Position error (b) Control input

Figure C.11: $\delta_t = h$, $K_1 = 0.0282\ Nm/rad$, $K_2 = 0.006851\ Nms/rad$, $h = 1$ ms.



(a) Position error (b) Control input

Figure C.12: $\delta_t = h$, $K_1 = 0.0282\ Nm/rad$, $K_2 = 0.006879\ Nms/rad$, $h = 1$ ms.