Automatic Control Laboratory, ETH Zürich
Prof. J. Lygeros

Manual prepared by: T. Grämer, S. Richter, U. Mäder, A. Zgraggen
Revised by: Kevin Mondoloni
Revision from: October 3, 2022

IfA Fachpraktikum - Experiment 2.4 :

Speed Control

Solutions Manual

Speed controlled drives play a central role in the industry as actuators, e.g. in machine tools and most other process and energy engineering facilities. Nowadays, more and more dc motors are being replaced by asynchronous motors with variable-frequency drives. In a first approximation these two control systems behave similarly.

In this experiment the effects of a P-, PI- and PID-controller will be presented and analysed with the help of a lab model.

- In the **preparation phase** (so *before* the lab session) you will analyse the plant with a MATLAB simulation and design a P-, PI- and PID-controller using the Ziegler-Nichols method.
- During the **lab session** the closed loop response is to be analysed systematically using a reference and a step response and the controller designed at home is evaluated. The control-platform is already implemented, you just have to parametrise it. To fulfill the specifications you may have to retune your controller.

For the experiment preparation you will need the following files which you can download from the IfA Fachpraktikum webpage: `http://people.ee.ethz.ch/~ifa-fp/wikimedia/images/6/6e/IfA_2-4_template.zip`

`ifa24ol.m`	**O**pen **l**oop simulation
`ifa24cl.m`	**C**losed **l**oop simulation

During the lab session you will need this additional file which is already saved on the lab computer:

`ifa24_get_data.m`	Download the measurement data from the control platform

Contents

Chapter 1

Problem Setup and Notation

1.1 Experimental Setup

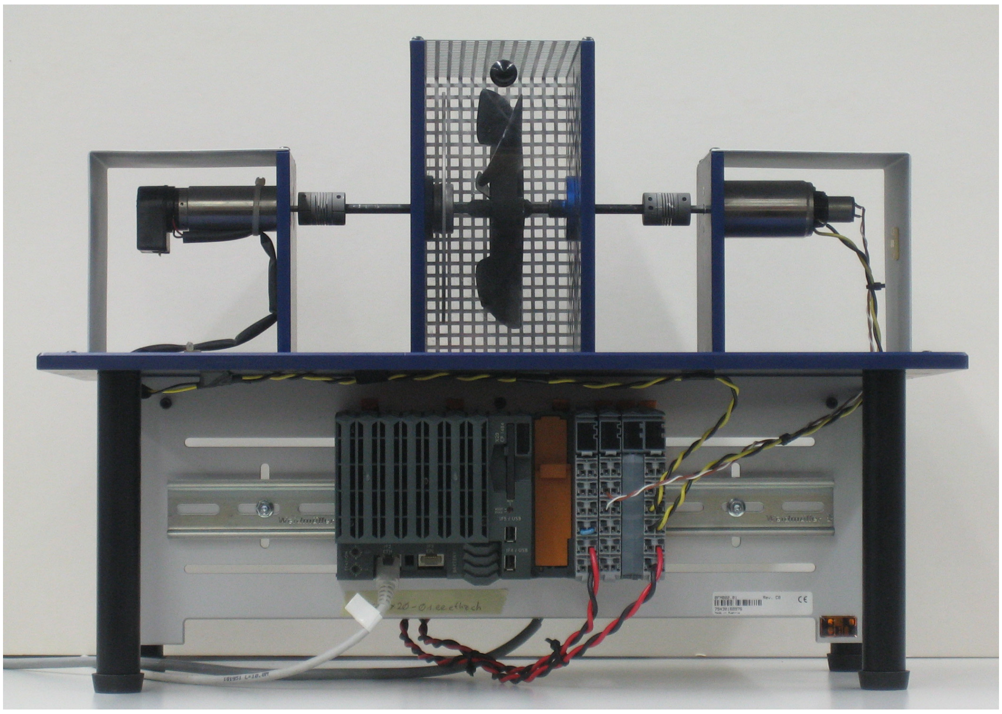

Figure 1.1: The experimental setup for fast speed control

Figure 1.1 shows the setup of the experiment. On the right, there is the DC motor which is supplied by a voltage source. On the left there is the brake motor, which creates a load torque due to the applied current. The shaft in the middle is composed of a rigid coupling and ventilation rotor screwed on. Furthermore a tachometer on the shaft provides us with information for the angular velocity.

Below the test rig is the controller which performs the measurement of the angular velocity, the steering of the motors and the actual control. The controller is configured over an ethernet port and consists of multiple modules (from left to right):

- Steering module (X20 CP 1484) for communication and control
- Analog input module (X20 AI 2622) for measuring the tachometer signal
- Motor bridge module (X20 MM 2436) for the motor power supply

The motor bridge module has two outputs which can be configured as either voltage- or current sources. The load motor is driven by a current source. This causes a load moment proportional to the anchor current. The main rotor is driven by a voltage source of $\pm 24V$. In table 1.1 the most important parameters of the motor bridge module are summarised.

amplification factor	1
output voltage (drive motor)	$\pm 24V$
output current (load motor)	$\pm 1.4A$

Table 1.1: Controller Data

The digital motor controller provides a pulse-width modulated (PWM) signal of amplitude $\pm$ 24 V to both motors. A PWM-signal is a digital, clocked signal whose amplitude is either $\pm 24V$ (high) or 0 V (low). The duty cycle D ($0 \leq D \leq 1$) defines the fraction of one period during which a signal is in the high state. For D = 1, the full voltage is always applied, for D = 0 no voltage is present across the motor. Due to the low pass characteristic of the motor the digital signal is smoothed and can approximately be seen as an analog signal of a voltage $D * \pm 24\ V$.

1.2 Open Loop Plant Model

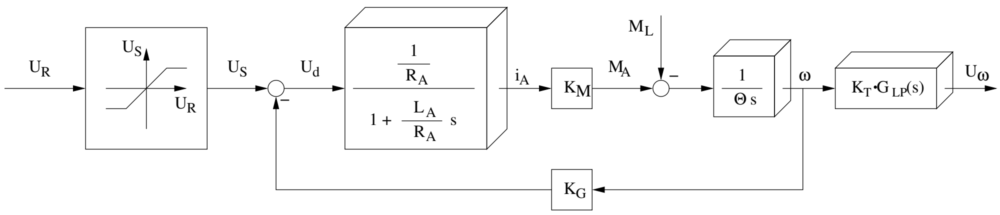

Figure 1.2: Plant Model $P(s)$

Figure 1.2 shows the block diagram of the open loop of the plant. The control input U_R is limited to $\pm 24V$ due to the limited operational range of the motor controller. This results in the following saturation characteristic:

$$U_S = \begin{cases} -24V, & \text{for } U_R < -24V \\ U_R & \text{for } -24 \leq U_R \leq 24V \\ +24V, & \text{for } U_R > 24V \end{cases} \tag{1.1}$$

From DC motor theory we get the following differential equation for the anchor current i_A:

$$L_A \frac{d\,i_A}{dt} = U_d - R_A\, i_A. \tag{1.2}$$

Here L_A is the anchor inductivity and R_A the anchor resistance. U_d is the differential voltage between the motor input voltage U_S and the induced voltage $K_G\,\omega$ (counter EMF). Equation (1.2) leads to the structure depicted in figure 1.2 .

The anchor current generates a driving torque

$$M_A = K_M \cdot i_A, \tag{1.3}$$

which counteracts the external load moment M_L. The resulting moment leads to an acceleration of the shaft:

$$\Theta \frac{d\,\omega}{dt} = M_A - M_L. \tag{1.4}$$

Where Θ is the moment of inertia of the shaft. The angular velocity ω of the shaft follows through integration and is measured by the tachometer generator (generator constant K_T). The resulting voltage signal shows a considerable ripple and is therefore smoothed with the second order low pass filter $G_{LP}(s)$:

$$G_{LP}(s) = \frac{{\omega_F}^2}{s^2 + 2\,D_F\,\omega_F\,s + {\omega_F}^2} \tag{1.5}$$

Here s is the laplace–variable, ω_F the cutoff frequency and $2D_F$ the damping factor of the low pass filter.

The model of the plant results in a fourth order system. The differential equation for the state variable i_A results from equation (1.2), the one for the state variable ω from equation (1.4). Since the low pass filter is of order two, two additional state variables are introduced: The output voltage U_ω of the filter and another auxiliary variable without a significant physical meaning. Thus we get the following state space representation:

$$\begin{aligned}
\begin{bmatrix} \dot{i}_A \\ \dot{\omega} \\ \dot{x}_3 \\ \dot{U}_\omega \end{bmatrix} &= \begin{bmatrix} -\frac{R_A}{L_A} & -\frac{K_G}{L_A} & 0 & 0 \\ \frac{K_M}{\Theta} & 0 & 0 & 0 \\ 0 & K_T\,\omega_F & -2\,D_F\,\omega_F & -\omega_F \\ 0 & 0 & \omega_F & 0 \end{bmatrix} \begin{bmatrix} i_A \\ \omega \\ x_3 \\ U_\omega \end{bmatrix} + \begin{bmatrix} \frac{1}{L_A} & 0 \\ 0 & \frac{-1}{\Theta} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U_S \\ M_L \end{bmatrix} \\
U_\omega &= \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} i_A \\ \omega \\ x_3 \\ U_\omega \end{bmatrix} + \begin{bmatrix} 0 & 0 \end{bmatrix} \begin{bmatrix} U_S \\ M_L \end{bmatrix}
\end{aligned} \tag{1.6}$$

Remarks:

- The actual output (i.e. the signal we wish to track) of the plant is the shaft angular velocity (ω). As the output signal is corrupted with ripples and is therefore smoothed with a filter, the (filtered) U_ω is used as the output variable. In this experiment the dynamics of the filter do not play an important role for the control dynamics. This does not always have to be the case!
- The nonlinear saturation characteristic $U_S = f(U_R)$ (eq. 1.1) is represented as a separate block in the plant model (see fig. 1.2).
- For the purposes of this experiment we assume an operation in the linear part of the saturation characteristic, where the transfer function from U_R to U_ω is denoted by $P(s)$ (the plant). Note that the voltage might get saturated for certain experiments during the lab session and the occurring nonlinearity can make the control accuracy differ from a lot as compared to the simulation in MATLAB.

Chapter 2

Theoretical Exercises - Preparation at Home

The goal of the controller is to stabilise the output signal U_ω around a generally non-zero operating point. For this experiment the operating point is the angular velocity $\omega = 200\ rad/s$. The constant load moment induced by the load/brake motor should be 30 mNM. For this operating point, calculate the steady state values. Use the parameters from table 2.1 and the italic parameters in appendix A.1. Remember that in equilibrium all time derivatives are zero. With these parameters you can get the steady state values for U_S and U_ω from the linear system in equation (1.6).

Remark: The load moment of the ventilator increases quadratically with rotational speed. As a simplification you can assume a constant load moment of 6 mNm for an angular velocity of 200 rad/s

angular velocity ω	$200\ rad/s$
moment of inertia Θ of the shaft with ventilator	$125 \cdot 10^{-6}\ kg\, m^2$
load moment M_L	$(30+6)\ mNm$

Table 2.1: Operating Point Data

Task 1: Equilibrium Point

Calculate the steady state values for U_ω and i_A as well as the control input U_S using equation (1.6) and fill in the table 2.2.

Operating Point	Value
ω	$200\ rad/s$
U_ω	
i_A	
U_S	

Table 2.2: Steady State Values at the Operating Point

Solution to Task 1

By setting all time derivatives to zero one gets the steady state. Thus we get the following equations:

$$\begin{aligned}
0 &= -\frac{R_A}{L_A}\, i_A - \frac{K_G}{L_A}\,\omega + \frac{1}{L_A}\,U_S && (2.1)\\
0 &= \frac{K_M}{\Theta} i_A - \frac{1}{\Theta}\,M_L && (2.2)\\
0 &= K_T\,\omega_F\,\omega - 2\,D_F\,\omega_F\,x_3 - \omega_F\,U_\omega && (2.3)\\
0 &= \omega_F\,x_3 && (2.4)
\end{aligned}$$

From equation (2.4) it immediately follows:

$$x_3 = 0 \tag{2.5}$$

Therefore using equation (2.3) and the data from the appendix we get:

$$U_\omega = K_T \cdot \omega = 45.82\frac{\text{mV}}{\text{rad/sec}} \cdot 200\text{rad/sec} = 9.16\text{V} \tag{2.6}$$

From equation (2.2) we get the anchor current:

$$i_A = \frac{M_L}{K_M} = \frac{36\text{mNm}}{51.2\text{mNm/A}} = 0.703\text{A} \tag{2.7}$$

Therefore U_S results in:

$$U_S = R_A\,i_A + K_G\,\omega = 8.5\text{V/A} \cdot 0.703\text{A} + 52.5\text{mV/(rad/sec)} \cdot 200\text{rad/sec} = 16.47\text{V} \tag{2.8}$$

Remark: In steady state the moment of inertia is not relevant (!), only the load moment is.

Filling the just calculated values into table 2.3:

Operating Point	Ventilator
ω	200 rad/s ≈ 1909 rpm
U_ω	9.16 V
i_A	0.703 A
U_S	16.47 V

Table 2.3: Steady State Values

In this experiment we use P-, PI- and PID controllers which are an industry standard. Figure 2.1 shows the closed loop PID controller. Here, $P(s)$ is the plant shown in Figure 1.2

Besides a normal feedback we also use a feedforward term to increase the control accuracy, namely we add an additional voltage U_{FF} at the input of the plant. This corresponds to the steady-state voltage U_S calculated in Task 1. The PID controller is then used to control (correct for) deviations from the steady state values, this includes disturbances and modelling errors.

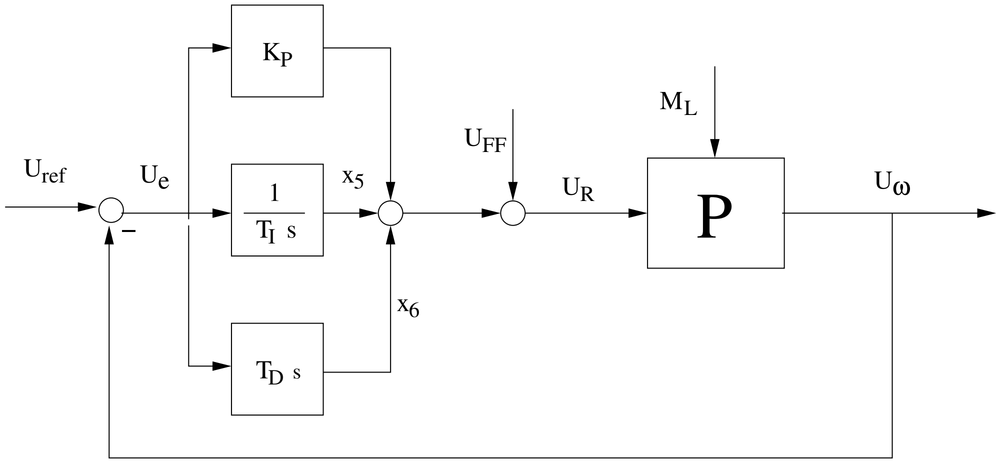

Figure 2.1: Closed-Loop PID Controller

The *closed loop* state space equations are:

$$\begin{bmatrix} \dot{i}_A \\ \dot{\omega} \\ \dot{x}_3 \\ \dot{U}_\omega \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} -\frac{R_A}{L_A} & -\frac{K_G}{L_A} & 0 & -\frac{1}{L_A}(K_P+k) & \frac{1}{T_I L_A} & -\frac{k}{L_A} \\ \frac{K_M}{\Theta} & 0 & 0 & 0 & 0 & 0 \\ 0 & K_T\,\omega_F & -2\,D_F\,\omega_F & -\omega_F & 0 & 0 \\ 0 & 0 & \omega_F & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -\frac{1}{T} & 0 & -\frac{1}{T} \end{bmatrix} \cdot \begin{bmatrix} i_A \\ \omega \\ x_3 \\ U_\omega \\ x_5 \\ x_6 \end{bmatrix}$$

$$+ \begin{bmatrix} \frac{1}{L_A} & 0 & \frac{1}{L_A}(K_P+k) \\ 0 & -\frac{1}{\Theta} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & \frac{1}{T} \end{bmatrix} \cdot \begin{bmatrix} U_{FF} \\ M_L \\ U_{ref} \end{bmatrix}$$

$$\begin{bmatrix} U_\omega \\ U_R \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -K_P-k & \frac{1}{T_I} & -k \end{bmatrix} \cdot \begin{bmatrix} i_A \\ \omega \\ x_3 \\ U_\omega \\ x_5 \\ x_6 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & K_P+k \end{bmatrix} \cdot \begin{bmatrix} U_{FF} \\ M_L \\ U_{ref} \end{bmatrix}$$

K_P is the gain of the P-controller. Additionally, the output voltage of the integrator has been introduced as the new state variable x_5. T_I is the integral time constant of the I-part of the PI-controller. Since a pure D-part is not realisable (the transfer function of a pure D has a numerator of higher order than the denominator i.e. a non causal transfer function), a so called "Dirty-D" is implemented, a realisable differentiator with a limited gain. We get this implementation by adding a time delay T $\left(\overset{\mathcal{L}}{\to} e^{-Ts}\right)$ approximated to first order, to a normal D-controller. The output of this element is the state variable x_6 and its transfer function is $G(s) = k \cdot \frac{Ts}{Ts+1}$. The derivative time constant T_D is given by the delay T and the gain k: $T = \frac{T_D}{k}$ where $k \approx 10 \ldots 20$.

Remark:

- For the plant simulation in the MATLAB-files `ifa24ol.m` and `ifa24cl.m` the nonlinear characteristic of the voltage amplifier, see definition (1.1), is replaced by a proportional unity gain.

- In Figure 2.1, for convenience only, a pure D-element is displayed instead of a Dirty-D.

2.1 The Ziegler–Nichols Methods

In 1943, on the basis of a lot of experiments, Ziegler and Nichols developed adjustment standards in a tabular form for PID-controllers, which are still widely used even today. These empirical methods are designed to tune PID-controllers for unknown plants. If the plant is known, often higher quality methods are used. In this experiment you will get to know these empirical methods even though the plant model is known. To get a feeling for the mode of action of a P-, PI- and PID-controller you will apply the Ziegler-Nichols method to a simulation.

The unknown plant is approximated with a dominantly first order model of *average relative time delay* with a transfer function of:

$$G_{u_e}(s) = e^{-sT_{t_e}} \frac{k_{s_e}}{1 + sT_{1_e}}. \tag{2.9}$$

Index e indicates a substitute model (german:"'**E**rsatz"') for the actual transfer function of the plant. For these methods Ziegler and Nichols set the following range of validity:

$$0.1 \lesssim T_{t_e}/T_{1_e} \lesssim 1.0, \tag{2.10}$$

where, from experience, results in the range $0.167 \lesssim T_{t_e}/T_{1_e} \lesssim 0.33$ are good. Outside of that range manual corrections are required. Note that the range with a 'small' relative delay $0 \leq T_{t_e}/T_{1_e} \lesssim 0.10$ are **not** part of this method!

2.1.1 The Closed Loop Tuning Method

For the closed loop tuning method the control loop is closed with a P-controller (i.e. I- and D-controllers are deactivated by setting $T_I = \infty$ (or $1/T_I = 0$) and $T_D = 0$). Then K_P is incrementally increased until the control loop becomes marginally unstable (i.e. reaches oscillation). The corresponding K_P is noted down as **ultimate gain** K_u. Additionally one measures the **oscillation period** P_u. From these 2 measured values, gains for the P-, PI- and PID-controller can be set according to table 2.4.

Method	**Controller**	K_P	$1/T_I$	T_D
Oscillation	P	$0.5K_u$	–	–
	PI	$0.45K_u$	$1.2/P_u$	–
	PID	$0.6K_u$	$2/P_u$	$(1/8)P_u$
Step Response	P	$T_{1_e}/(T_{t_e}k_{S_e})$	–	–
	PI	$0.9T_{1_e}/(T_{t_e}k_{S_e})$	$0.3/T_{t_e}$	–
	PID	$1.2T_{1_e}/(T_{t_e}k_{S_e})$	$0.5/T_{t_e}$	$0.5T_{t_e}$

Table 2.4: Derivation of parameters according to Ziegler-Nichols

Task 2: Closed Loop Tuning Method

Determine control parameters using the closed loop tuning method and fill in the Table 2.5. To setup the controller use the MATLAB function `ifa24cl(KP,1/TI,TD)` and analyse the closed loop stability margin with a step response of 5 V. Both parameters $1/TI$ and TD are to be set to 0.

Remark: Note that for the closed loop tuning method in MATLAB the voltage U_R occasionally exceeds the saturation limit. This does not pose a problem though as we do not have such a saturation limit for the simulation.

Method	Controller	K_P	$1/T_I$	T_D
Oscillation	P		–	–
	PI			–
	PID			
Step Response	P		–	–
	PI			–
	PID			

Table 2.5: Control parameters according to Ziegler-Nichols (model)

Method	Controller	K_P	$1/T_I$	T_D
Oscillation ($K_u = 15.25, P_u = 0.29$)	P	7.6	–	–
	PI	6.9	4.1	–
	PID	9.2	6.9	0.036
Step Response ($T_{t_e} = 0.0667, T_{1_e} = 0.495, k_{S_e} = 0.872$)	P	8.6	–	–
	PI	7.7	4.5	–
	PID	10.3	7.5	0.033

Table 2.6: Control parameters according to Ziegler-Nichols (model)

Solution to Task 2

See Table 2.6.

2.1.2 The Open Loop Tuning Method

Not all control circuits are allowed to be brought to oscillation. That is why Ziegler-Nichols developed a second method. For this a unit step input is applied to the plant and the step response is analysed as in Figure 2.2. By drawing a tangent through the inflection point (point where the sign of the curvature changes) of the step response one gets the auxiliary latency T_{t_e}, the plant gain k_{S_e} and the auxiliary rise time T_{1_e}. If a non-unit step input is applied the measured factor k_{S_e} has to be corrected by dividing it by the height of the step.

Task 3: Open Loop Tuning Method

Determine the control parameters using the open loop tuning method and fill in the Table 2.5. Use the MATLAB-function `ifa24ol` to simulate the step response of the open loop for a step input of 5 V. From the step response and its first derivative you should be able to extract the parameters T_{t_e}, T_{1_e} and k_{S_e}:

As in Figure 2.2, the plant gain corresponds to the final value of the step response divided by the step input height (here: 5 V), the auxiliary latency is the intersection of the tangent through the inflection point and the time axis and the auxiliary rise time is the time between the previously mentioned intersection and the intersection of this tangent with the final steady state value. By looking at the derivative graph of the input response one gets and slope of the tangent and can then calculate where the tangent intersects with other lines. Then calculate the values for K_P, $\frac{1}{T_I}$ and T_D according to Table 2.4.

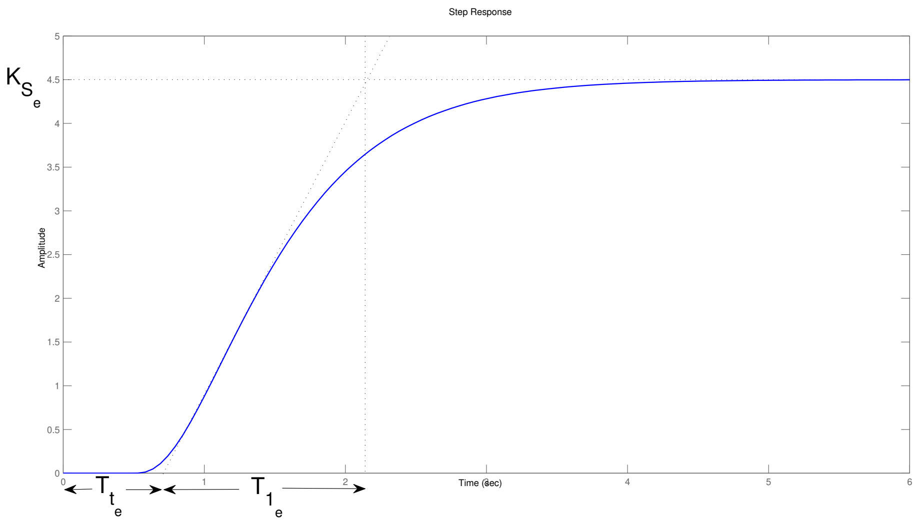

Figure 2.2: Open loop tuning method according to Ziegler-Nichols. The tangent is to be put through the inflection point of the step response. In this example $T_{t_e} = 0.75$, $T_{1_e} = 1.4$ and $k_{S_e} = 4.5$, where a unit step was used as an input.

Solution to Task 3

Plant gain $k_{S_e} = 0.872$, Inflection point of step response at $t = 0.165$s. Slope at inflection point: $k_{S_e} = 8.813$. From this one gets a auxiliary latency of $T_{t_e} = 0.0667$s and a auxiliary rise time of $T_{1_e} = 0.495$s.

The control parameters are in table 2.6.

Task 4: Comparison of Both Methods

Compare the control parameters you got for both methods. How do you justify the differences? Are the conditions (validity ranges) set by Ziegler-Nichols fulfilled?

Solution to Task 4

The conditions set by Ziegler-Nichols are fulfilled, $T_{t_e}/T_1 \approx 0.133$. The values of both methods are close but since we are dealing with empirical methods they are not identical.

The step response of the controller should fulfill the following specifications (see appendix B) :

- rise time (german:Anregelzeit) ≤ 0.3s for 90% of the steady state value.
- settling time (german:Ausregelzeit) ≤ 1.5s for a tolerance band of $\pm 5\%$ of the steady state value.
- overshoot (max. peak value, german:Überschwingen) $\leq 20\%$ of the steady state value.

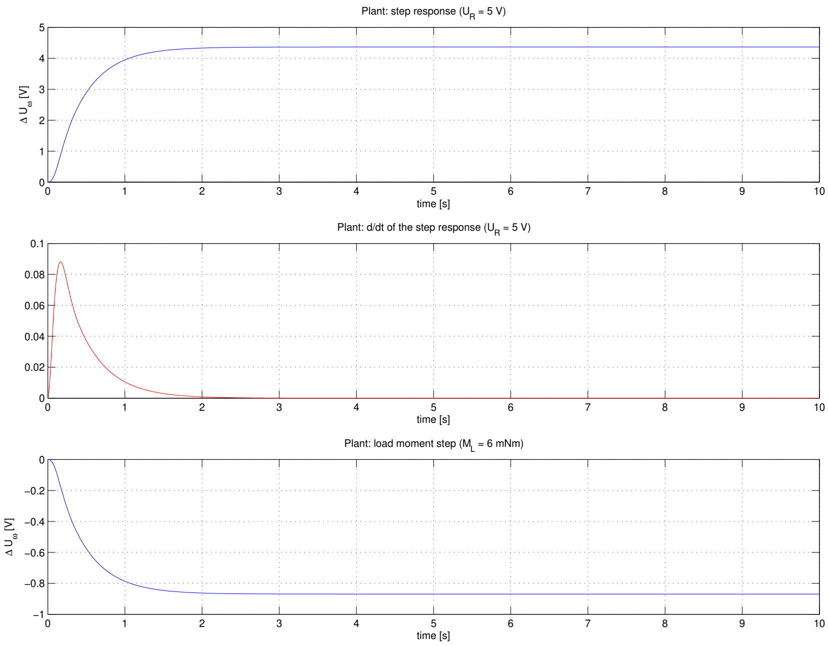

Figure 2.3: Open loop step responses for a step of 5 V

If these specifications are not met with the control parameters of Ziegler-Nichols you have to manually tune the parameters a bit. Table 2.7 indicates how an increase in control parameter affects the other values.

parameter	**rise time**	**overshoot**	**settling time**	**steady state error**
$K_P \uparrow$	$\downarrow$	$\uparrow$	small influence	$\downarrow$
$1/T_I \uparrow$	$\downarrow$	$\uparrow$	$\uparrow$	$\rightarrow 0$
$T_D \uparrow$	small influence	$\downarrow$	$\downarrow$	no influence

Table 2.7: Effect of increasing the individual control parameters

Task 5: Compliance with Specifications

- Use the MATLAB script `ifa24cl(KP,1/TI,TD)` in order to simulate the closed loop step response of a 5V step, a U_{FF} step of 5V, and a load moment of 10mNm. Fill in the Table 2.8 by using either the parameters from the oscillation or the open loop tuning method.
- Which of the above specifications are fulfilled by the P-, PI-, and PID-controllers?
- Tune the PID-controller as to keep the overshoot within the given boundary. The effect of each individual control parameter is given in Table 2.7

Controller	P-controller	PI-controller	PID-controller	PID-controller retrimmed
K_P $\frac{1}{T_I}$ T_D				
Reference Response (U_{ref})				
rise time [s] settling time [s] overshoot $e_{max}(t)$ [%] steady state error e_∞ [%]				
Disturbance Response of U_{FF}				
settling time [s] overshoot $e_{max}(t)$ [%]				
Disturbance Response of M_L				
settling time [s] overshoot $e_{max}(t)$ [%]				

Table 2.8: Results of controller tuning using simulation

Remark: Keep an eye on the saturation limit: The working range for the voltage amplifier is ± 24 V, afterwards it saturates.

Solution to Task 5

- The specification regarding the rise and settling time are fulfilled by all controllers. However the overshoot is more than twice as high as it is allowed to be.
- See Table 2.9. One gets slightly different values for the open loop tuning method.

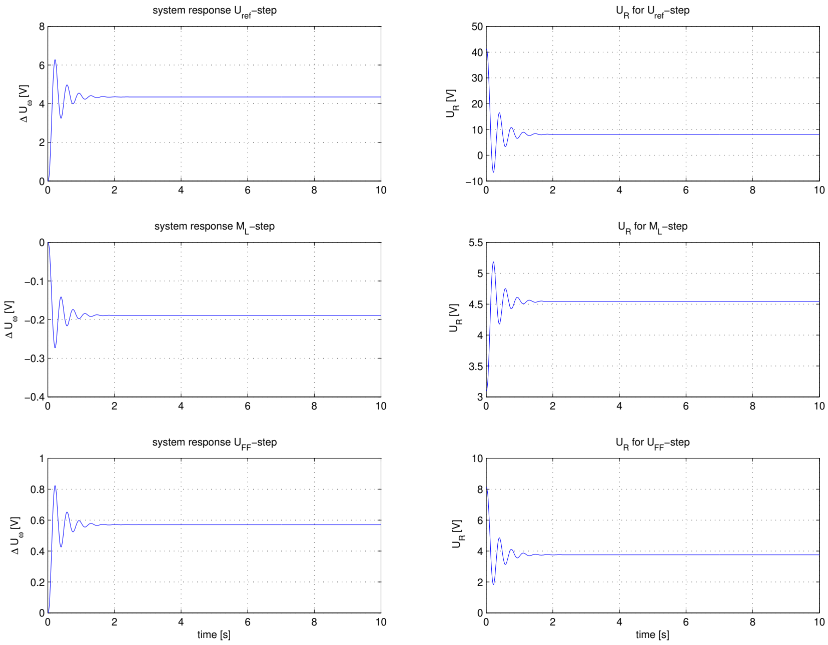

Figure 2.4: Step response of closed loop and controlled variables, $K_P = 7.63$, $1/T_I = 0$ 1/sec, $T_D = 0$ sec

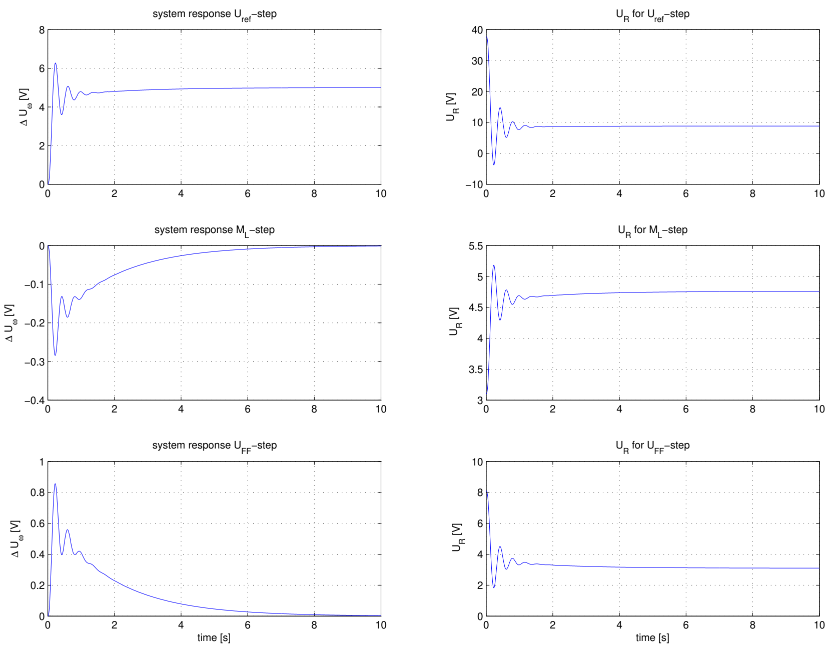

Figure 2.5: Step response of closed loop and controlled variables, $K_P = 6.87$, $1/T_I = 4.14$ 1/sec, $T_D = 0$ sec

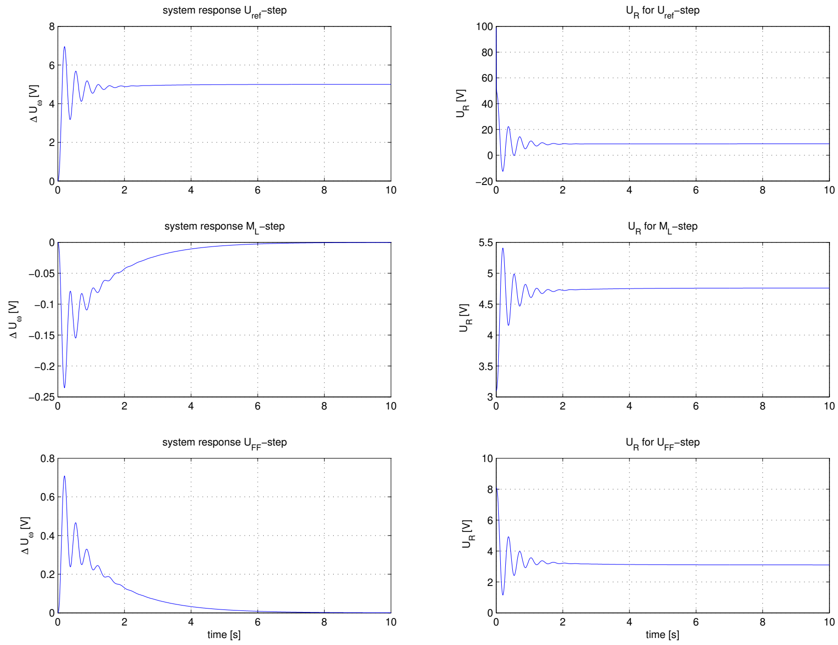

Figure 2.6: Step response of closed loop and controlled variables, $K_P = 9.15$, $1/T_I = 6.90$ 1/sec, $T_D = 0.036$ sec

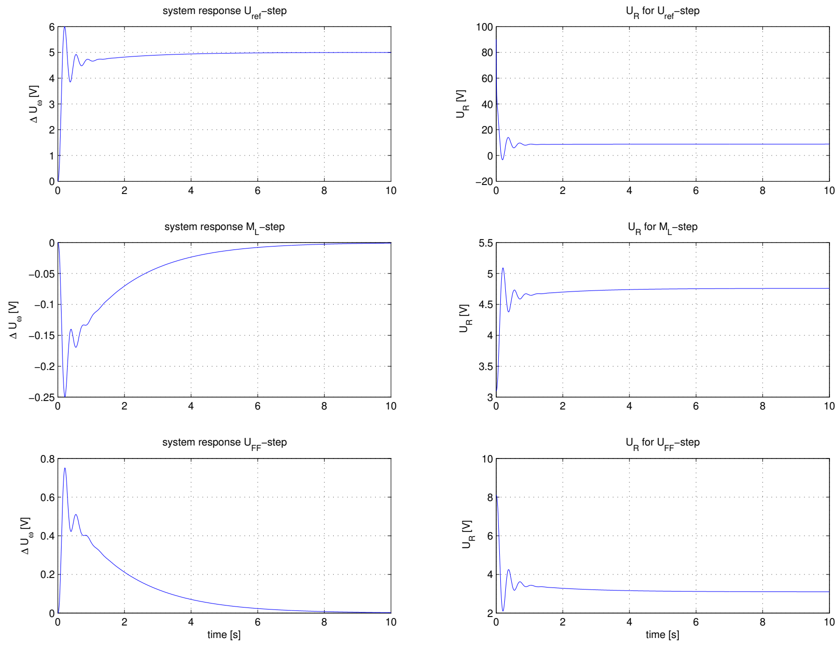

Figure 2.7: Step response of closed loop and controlled variables, $K_P = 7.4$, $1/T_I = 4.5$ 1/sec, $T_D = 0.1$ sec

Controller	P-controller	PI-controller	PID-controller	PID-controller retrimmed
K_P	7.63	6.87	9.15	7.4
$\frac{1}{T_I}$	–	4.14	6.90	4.5
T_D	–	–	0.036	0.1
Reference Response (U_{Soll})				
rise time [s]	0.14	0.16	0.13	0.14
settling time [s]	0.82	**1.63**	1.13	1.46
overshoot $e_{max}(t)$ [%]	**44.4 %**	**25.5 %**	**39.1 %**	20.0 %
steady state error e_∞ [%]	13 %	0 %	0 %	0 %
Disturbance Response of U_{FF}				
settling time [s]	0.82	2.43	1.77	2.27
overshoot $e_{max}(t)$ [%]	44.4 %	40.3 %	63 %	32.3 %
Disturbance Response of M_L				
settling time [s]	0.82	0.49	0.75	0.45
overshoot $e_{max}(t)$ [%]	44.3 %	8.9 %	13.6 %	7.0 %

Table 2.9: Results of controller tuning using simulation and the control parameters from the closed loop tuning method

Chapter 3

Lab Session Tasks

During the lab afternoon you will examine if the model corresponds to the actual plant and test the controller that you've designed at home on the real physical system.

Start the script `IfA_2.4` on the desktop to copy the required files into the directory C:\Scratch\Speed_Control.

The experiment apparatus is managed through a GUI (see Fig. 3.1) where you can enter the control parameters. Since the graphical displaying of the different voltage curves over time is coarse you can also record the values and process them further with MATLAB. Open the VNC-viewer and use the following login to start the GUI:

```
Server:     autx20-01
Passwort:   control
```

Click on the button `Start` to begin the experiment. With the big button on the top left you can (de)activate the controller. The P-, I- and D-branches can be switched on/off separately. If all branches are switched off we have the open loop system. The numbers inside the black boxes can be changed, the ones in grey are only there for display. With the `Record` button you can start and stop the recording of the voltage signals. With a temporal resolution of 1 ms you can record maximum for 30 seconds, with a coarser temporal resolution a longer time span can be recorded. During the recording certain functions are locked. After the completion of a recording the data has to be written to memory and therefore you cannot start a new recording during that time. Note that the control parameters in the GUI are K_P, K_I and K_D , where $K_I := \frac{1}{T_I}$ and $K_D := T_D$.

3.1 Model Validation

Before you close the control circuit, you have to make sure the your model is close enough to the real plant. For this you will validate your model i.e. your simulation results on the physical plant.

Performance in the low frequency regime, i.e. the plant gain and the dominant time constant can be most easily determined from the step response. The better the measured step response corresponds to the simulation, the more reliable the previously calculated control parameters are and the lower the risk for the plant when the control loop is closed is.

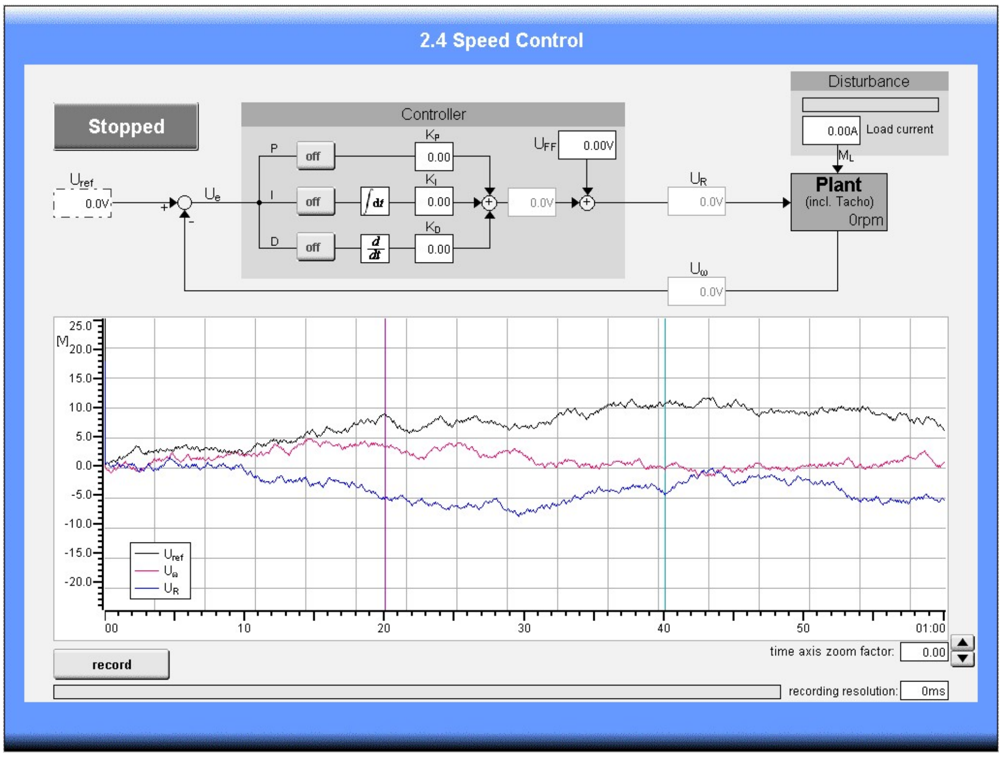

Figure 3.1: GUI for the Controller

Task 6: Validation of the Equilibrium Point and the Model

Check if the equilibrium point calculated in Task 1 is correct. Adapt the feed forward voltage U_{FF} until the shaft rotates with 200 rad/s (don't forget to convert rad/s to rpm). The voltages U_R and U_ω can be read of directly. The anchor current i_A does not need to be calculated. Since you are analysing the open loop behaviour the controller obviously has to be shut off. You, however, have to set the load moment M_L. This is done through the load motor whose load moment is proportional to the current of the current source. With the data from appendix A.1 you can calculate the required current for a load moment of 30 mNm:

$$M_L = K_M \cdot i_L \Longrightarrow i_L = 0.91A. \tag{3.1}$$

state	value
ω	
U_ω	
U_R	

Table 3.1: Measured steady state values at the operating point

Next, plot the open loop step response using the MATLAB-function `ifa24ol`. As a comparison to the real system set the load moment to 0 and start the recording. Change the feed forward voltage from 0 V to 5 V to simulate an open loop step input. Stop the recording and retrieve the data using the MATLAB Function `ifa24_get_data`. Adapt the MATLAB-script if necessary in order to set the step input to time 0. Compare both step responses. Does the model correspond to the real system?

Solution to Task 6

- For a load current of 0.91 A the calculated value differs from the real one. See table 3.2

Operating Point	Rotor
ω	$\sim 200\ rad/s \approx 1909\ rpm$
U_ω	$9.2\ V$
U_R ($= U_{FF}$ since OL)	$14.0\ V$

Table 3.2: Measured steady state values at the operating point

- The measured step response, on the other hand, deviates significantly from the simulated response. This deviation is due to the neglected friction in the model and the breaking moment of the ventilation rotor which has not been taken into account in the MATLAB simulation, see Fig. 3.2.

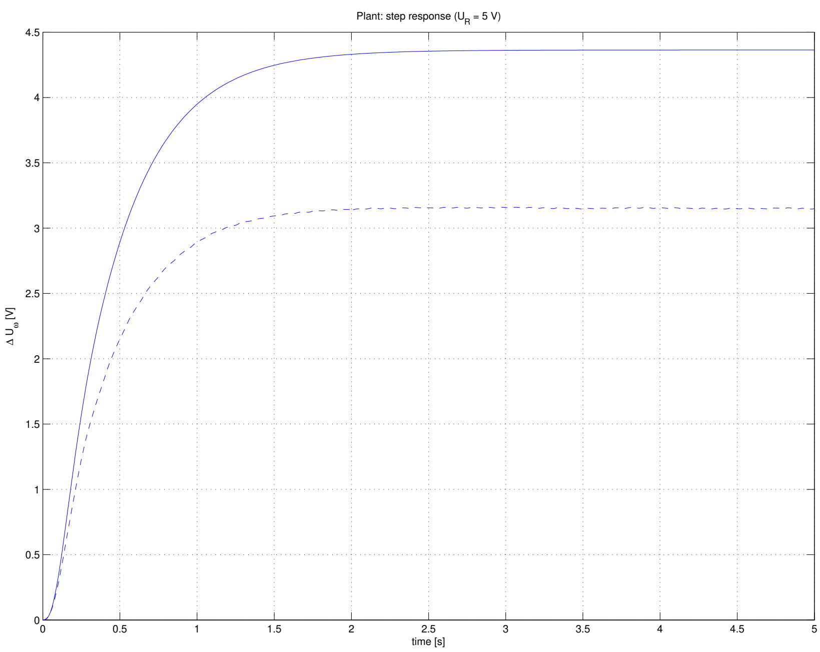

Figure 3.2: Simulated (solid line) and measured (dashed line) open loop step responses for a step of 5 V

3.2 Closed Loop System

Task 7: Closed Loop Tuning Method on the Real System

Since the simulated input response does not match well enough with the measured one you will perform the closed loop tuning method on the real system. In order to get good results with regards to PID-control parameters we will perform the closed loop tuning method *at* the operating point. The load moment will be set to 0 just like for the open loop tuning method, the feed forward voltage U_{FF} is thus reduced compared to the one calculated in task 6. Determine this voltage just like in Task 6.

In contrast to the simulation Task 2 the step input should have an amplitude of 3 V in order to avoid nonlinear effects due to saturation. If you still see them reduce the amplitude further to 2 V. Instead of a step you can also use a rectangle signal with a sufficiently large period. For this set U_ω as offset and an amplitude of 1.5 V (1 V). This will also result in a step of 3 V (2 V). To measure the critical period P_u you have to record the data and evaluate it with MATLAB. Determine the control parameters according to Ziegler-Nichols, fill in the Table 3.3 and compare the parameters to those from your simulation of the closed loop tuning method (Table 2.5).

Method	**Controller**	K_P	$1/T_I$	T_D
Oscillation	P		–	–
	PI			–
	PID			

Table 3.3: Control parameters according to the Ziegler-Nichols method (real system)

Solution to Task 7

For a load current of $0A$ we now get $U_{FF} = 13.5\ V$. For the resulting Ziegler-Nichols parameters see Table 3.4

Task 8: Testing the Controller

Method	**Controller**	K_P	$1/T_I$	T_D
Oscillation (simulated) ($K_u = 15.25, P_u = 0.29$)	P	7.6	–	–
	PI	6.9	4.1	–
	PID	9.2	6.9	0.036
Oscillation (real system) ($K_u = 15.7, P_u = 0.32$)	P	7.85	–	–
	PI	7.07	3.75	–
	PID	9.42	6.25	0.04

Table 3.4: Control parameters according to Ziegler-Nichols (real system)

Controller	PID-controller retrimmed
K_P $\frac{1}{T_I}$ T_D	
Reference Response (U_{ref})	
rise time [s] settling time [s] overshoot $e_{max}(t)$ [%] steady state error e_∞ [%]	
Disturbance Response of U_{FF}	
settling time [s] overshoot $e_{max}(t)$ [%]	
Disturbance Response of M_L	
settling time [s] overshoot $e_{max}(t)$ [%]	

Table 3.5: Performance of the retrimmed controller

Now you can close the loop. First set the feed forward voltage U_{FF} back to the voltage U_R determined without a load moment in the previous exercise. This will allow us to work at the operating point. Use U_ω from the previous exercise as a reference value (*set point*) or offset (for a sine or rectangle signal).

Test the closed loop system with the parameters that you got from the closed loop tuning method of the real system. Start with a P-controller and also test the PI- and PID-controllers. Investigate the behaviour for different reference signals (sine, rectangle). Is the control loop stable? Tune the controller as in Task 5 and fill in the Table 3.5 for a reference step of 2 V around the operating point, a U_{FF} step of 2 V and a load moment of 10 mNm (corresponds to $i_{load} = 0.3\ A$).

Remark: Mind the nonlinear effects due to saturation of the input voltage when steps bigger than the set ones are carried out. Which part (P, I, or D) is especially sensitive towards saturation.

Solution to Task 8

The closed loop system is stable. Since the parameters of the closed loop tuning method in the simulation don't deviate much from the ones in the real system one can use the values from Task 5 for retuning. Especially the I-part is sensitive towards an input saturation. In this case, by reducing the factor $1/T_I$ the control accuracy can be improved. One could also think of implementing an anti-windup in the PID-controller.

See Table 3.6

Controller	PID-controller retrimmed
K_P	7.4
$\frac{1}{T_I}$	4.5 1/sec
T_D	0.1 sec
Reference Response (U_{Soll})	
rise time [s]	0.21
settling time [s]	2.73
overshoot $e_{max}(t)$ [%]	4.1 %
steady state error e_∞ [%]	0 %
Disturbance Response of U_{FF} for a 5 V step	
settling time [s]	0.4
overshoot $e_{max}(t)$ [%]	3 %
Disturbance Response of M_L	
settling time [s]	0 / not measurable
overshoot $e_{max}(t)$ [%]	0 % / not measurable

Table 3.6: Control parameters according to Ziegler-Nichols (closed loop tuning method on a real system)

Chapter 4

Lessons Learned

This chapter summarises the most important learning objectives of this experiment through questions. Think about the questions and discuss your answers with the supervisor.

Lesson Learned 1: Modelling

For the modelling and the open loop simulation certain influences have been neglected (sometimes implicitly), which lead to differences between the simulation and the measurement. Which influences could these be and could you model them?

Answer to Lesson Learned 1

The bearing friction (implicitly), the braking torque of the fan (implicitly) and the input variable limitation (explicitly) have been neglected. The neglect of friction and the braking torque of the fan lead to the different steady state step response values, the motor itself corresponds well with the model. The friction corresponds to a ω-proportional breaking moment (linear) and the breaking moment of the fan is proportional to ω^2. So the system is actually nonlinear. A further nonlinearity arises due to limited actuation.

Lesson Learned 2: The Ziegler-Nichols Methods

The Ziegler-Nichols methods are empirical procedures to tune a PID-controller for an unknown plant. What assumptions about the unknown plant have to be made and what criteria have to be fulfilled in order to see a good result from Ziegler-Nichols?

Answer to Lesson Learned 2

See Section 2.1.

Lesson Learned 3: PID-controller

- Overshoot is increased by both a higher P-, and I-part of the controller. Does one of the parts have a higher influence on the overshoot than the other (based on your observations this afternoon)?
- Which part eliminates a steady state error?
- The analog tachometer signal of the shaft has a ripple and therefore also passes through a low pass filter. How would you have to adapt the control parameters if there wasn't a filter (which control parameters should increase/decrease or not change)?

Answer to Lesson Learned 3

- The P-, and I-part have an equal influence on the overshoot.
- An integral part eliminates steady state errors. (But can build up an error resulting in a high overshoot!)
- A noisy measurement signal has a lot of small, fast changing signals. These fast changes affect the D-part a lot. Therefore the D-part has to be decreased.

Lesson Learned 4: Task Completion

We hope that you had fun with the experiment and could learn something new. Please fill out the online feedback form on the registration website under `MyExperiments`. Each student has to fill out his/her own feedback. This helps us to further improve our experiments. Thank you very much for your inputs. Now discuss the experiment with your supervisor.

Answer to Lesson Learned 4

No solution needed.

Appendix A

Technical Data Sheet

A.1 Data for the Controlled System

Voltage source for the drive motor

Output Range		± 24	V
max. Continuous Output Current		± 3.0	A

Drive Motor

Rated Voltage		± 24	V
Idle Speed (4400 rpm)		461	$\frac{rad}{s}$
Initial Torque		149	mNm
max. Continuous Output		37	W
Average no-load Current		50	mA
max. Continuous Current		1.2	A
max. Speed (9000 rpm)		942	$\frac{rad}{s}$
Back-EMF	≙ Generator Const. K_G 5.5 mV/rpm	52.5	$\frac{mV}{rad/s}$
Anchor Inductivity		1.3	mH
Anchor Resistance R_A		8.5	Ω
Torque Constant	≙ Motor Const. K_M	51.2	$\frac{mNm}{A}$
Rotor Moment of Inertia		$5.2\ 10^{-6}$	kgm^2

Shaft with Fans

Moment of inertia		ca. $125\ 10^{-6}$	kgm^2
Braking Torque	at nominal rotational speed of 200 rad/s	ca. 10	mNm

Current Source for the Load Motor

Output Range		± 3	A
Limits imposed by Software		± 1.4	A

Load Motor

Rated Voltage		±24	V
Idle Speed (6750 rpm)		707	$\frac{rad}{s}$
Initial Torque		128	mNm
Continuous Output (max.)		27	W
Average no-load Current		110	mA
max. Continuous Current		1.4	A
max. Speed (9000 Upm)		942	$\frac{rad}{s}$
Back-EMF	≙ Generator Const. K_G 3.4 mV/Upm	32.5	$\frac{mV}{rad/s}$
Anchor Inductivity		0.75	mH
Anchor Resistance		6.2	Ω
Torque Constant	(Motor Const. K_M)	33	$\frac{mNm}{A}$
Rotor Moment of Inertia		$2\ 10^{-6}$	kgm^2

Tachometer with Amplifier

Amplification Factor K_T	24 V with 5000 rpm	45.82	$\frac{mV}{rad/s}$
Voltage Ripple		300	$mVpp$

Low Pass Filter

Input Voltage		± 24	V
Output Range		± 24	V
Nominal Cutoff Frequency	ω_F	20	$\frac{rad}{s}$
Damping Factor	$2D_F$ (max. flat)	1.414	

Note: K_G and K_M have the same SI-units.

Appendix B

Performance Parameters in the Time Domain

To describe the reference response of a closed loop control system one uses parameters depicted in Figure B.1. The plot represents a closed loop step response for a unit step input.

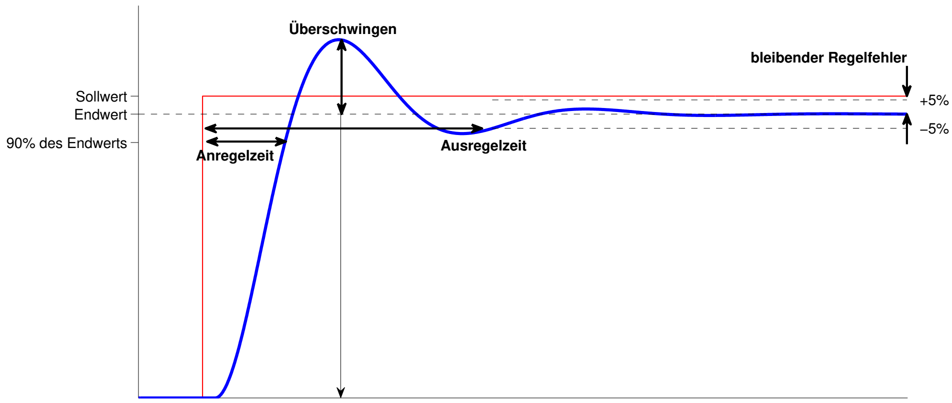

Figure B.1: Performance Parameters in the Time Domain

- **Rise Time** [s] **(german:Anregelzeit):** The time until the steady state value is achieved for the first time. This steady state value does not necessarily have to be the input step height (steady state error). Additionally the steady state value is often reached only asymptotically. Then we can define the rise time as the time until 90% of the steady state value is reached.
- **Settling Time** [s] **(german:Ausregelzeit):** The time starting from which the deviations from the steady value stay inside the tolerance band.
- **Tolerance Band** describes the deviation from the steady state value. Usually the tolerance band has values between $\pm 1\%$ and $\pm 5\%$
- **max. control error (overshoot, german:Überschwingen):** Usually values between 5% and 20% of the steady state value.