CE 150

HELICOPTER MODEL

Educational Manual

Table of Contents

List of Figures

Acknowledgement

This Educational Manual was written by Dr. Petr Horáček from the Department of Control Engineering, Faculty of Electrical Engineering, Czech Technical University of Prague. The Helicopter Model is based on the idea of Dr. Horáček and the first prototype built by him and his students. HUMUSOFT Ltd. wants to express thanks to all the people at the Department of Control for a lot of helpful advice during the development of the model.

Preface

The **CE 150 Helicopter Model** described in this manual is one of a range offered by HUMUSOFT for teaching systems dynamics and control engineering principles. The **Helicopter Model** and the associated manuals are teaching aids for control engineering students at all academic levels and the experiments cover wide range of problems found in industry.

The model belongs to the range of teaching systems directly controllable by an IBM PC or compatible computer in real time. There is no need for using any standard laboratory instrumentation, such as oscilloscope, plotter, signal generator, voltmeter, etc. All functions of laboratory instrumentation mentioned are done by software running on the personal computer. The student has no direct physical access to the signals coming from and to the model. Thus there is no danger of damaging the model or data acquisition card electronics by manipulation with the cables. Instead of direct physical access, software access to all signals measured and manipulated is available to the user.

There are five types of software environment available to the user:

Demo program - directly executable program written in C language. Friendly user interface facilitates first experiments with PID control.

Matlab demo program - more complex demonstration program with graphical user interface running under Matlab. This program allows to export recorded data directly into the Matlab environment which facilitates tasks like simulation, data analysis or model tuning.
Matlab is a high performance software package for scientific and numeric computation, signal processing and graphics in an environment where problems and solutions are expressed just as they are written mathematically - without traditional programming. The power of Matlab environment is further extended by Simulink - a block oriented environment for simulation of dynamic systems and numerous toolboxes. Some of them are highly recommended for the experimentation with the CE150 model: Control System Toolbox, System Identification Toolbox, Optimisation Toolbox, Real-Time Windows Target and Virtual Reality Toolbox. An absolute minimum to conduct experiments from this manual is the Control Systems Toolbox and Simulink.

The Real Time Toolbox for Matlab - supplied with the CE150 Helicopter Model enables the user to communicate directly with the system from the Matlab environment.

The Real-Time Windows Target - this environment offers best control performance. Real-Time Windows Target is not included, but offered separately by The MathWorks. Simulink controller exaples are included.

User developed environment - this option might be useful for solving special tasks not supported by Matlab and Real Time Toolbox, e.g. from the area of nonlinear control.

1. Getting Started

1.1. Introduction

The **CE150 Helicopter Model** is one of a unique range of products designed for the theoretical study and practical investigation of basic and advanced control engineering principles. This includes system dynamics modelling, identification, analysis and various controllers design by classical and modern methods.

A system configuration for the CE150 follows from **Figure 1.1**, where the system is connected to an IBM PC compatible computer. The model consists of a body carrying two DC motors. These motors drive the propellers. The body has two degrees of freedom. The axes of the body rotation are perpendicular as well as the axes of the motors. Both body position angles, i.e. azimuth angle in horizontal and elevation angle in vertical plane are influenced by the rotating propellers simultaneously. The DC motors for driving propellers are controlled proportionally to the output signal of the computer, the center of gravity is driven by a servomechanism. The helicopter model is a multivariable dynamical system with up to three manipulated inputs and two measured outputs. All inputs and outputs are coupled. The system is essentially nonlinear and at least of the sixth order, depending on the modelling precision. The mathematical model can be linearized around the set point.

The aim of the manual is to present typical problems and to suggest experiments, giving some guidelines for the problem solution. The need for consulting control theory textbooks is minimal, however the beginners should use them for learning the principles not described in this manual. The text is divided into blocks. Each block consists of the formulation of the problem, followed by the description of the solution principles. Number of experiments related to the problem are listed and typical results are given. It is important to note that the numerical results shown in the manual are relative and might be different from model to model due to differences in drives, propellers, etc.

The manual is divided into chapters. Chapter 2 is devoted to mathematical modelling. General model structure valid under some simplifications is derived in terms of nonlinear state space description and the corresponding block diagram. Identification of system parameters is done in Chapter 3. Number of experiments for direct and indirect measurement of system parameters is described. Typical numerical values are derived and used for initial tuning of the system model. The model is linearized around the different setpoints. Transfer function matrix as an Input/Output model and linear state space model are derived. The following chapters are devoted to controllers design. Chapter 4 gives guidelines for selection of the sampling period for digital control systems. Chapter 5 shows the design of the conventional PID control and State Feedback control by pole placement. Figures from measurements can be found in Chapter 6.

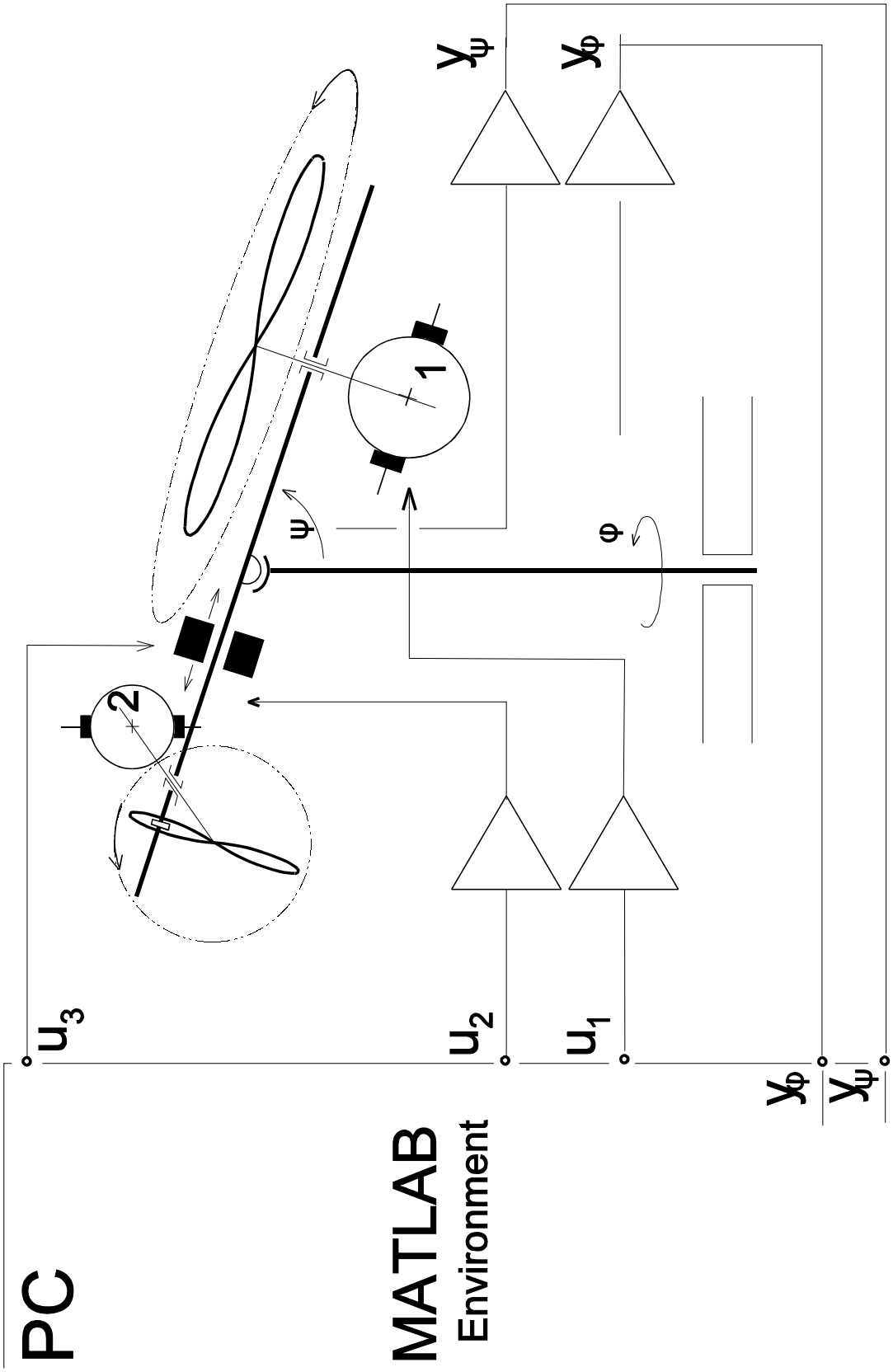


Figure 1.1 Schematic diagram of the Helicopter Model

1.2. MATLAB Demonstration Program

The MATLAB demonstration program is a fully menu-driven, user-friendly package to start with. The MATLAB code is quite complicated to read; this is the tradeoff of the user-friendly interface. This demonstration is included to let you feel how powerful a controller, written completely in MATLAB, can be. If you are not familiar with MATLAB (and maybe even if you are), please run this demo first and then proceed with reading the rest of this manual. The other MATLAB programs described later are almost opposite of this one; they are not very user friendly but their code is easily readable, and thus much better for learning to control the model. You can read the code of this demo when you feel you understand the following easier programs.

After installing the REAL TIME TOOLBOX according to its manual please copy the all the *.M files from the diskette to some directory on your hard disk; they are the demo files. Then run MATLAB and type HE2DEMO. Several screens will appear, for the first time it is best to accept the defaults by pressing Enter. The experiment should run and after it is finished all the responses are plotted. The plot should be similar to that on Figure 1.2, which was recorded with some parameters changed from defaults. This verifies that the MATLAB interface to the model is working correctly and you can proceed either with more experimenting with this demo, or skip to Chapter 2 to learn about the dynamics of the model and return here later.

Demonstration Program Description

The menu screens are mostly self-explaining. To choose one of menu options (trajectory etc...) type number corresponding to your choice and press Enter. To accept default values (in angle brackets), just press Enter. To modify parameter value, type row number of the parameter and press Enter. When the prompt appears, enter data followed by Enter. Note that you can use arithmetic operators, MATLAB variables and a predefined variable named default holding old parameter value. Vectors may but need not be enclosed in angle brackets.

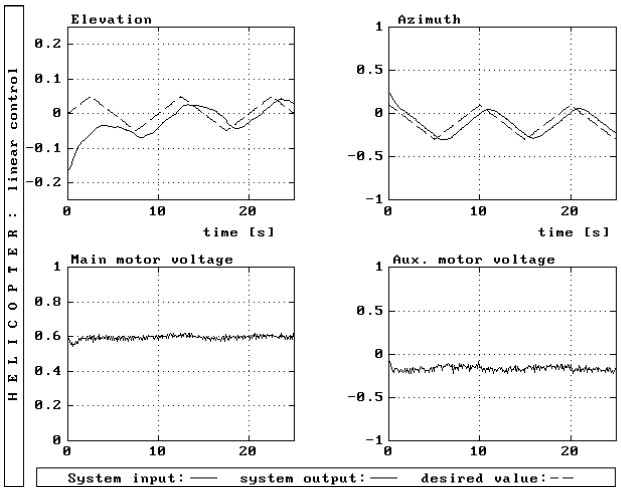


Figure 1.2 Demonstration Program Screen

After the experiment is finished, all parameters and experimental data stay in the workspace. They are used as defaults for the next experiment. If you want to start with defaults, simply clear all variables.

There are two versions of the demo program on your diskette. The basic one, HE2DEMO, controls the helicopter in both degrees of freedom. The HE1DEMO controls only the elevation, azimuth must be fixed. It may be simpler to experiment with the constants of this controller, at least for the beginning.

Demonstration Program Algorithms

The following describes some details of the demonstration program. Don't bother to read this unless you actually want; rather skip to Chapter 2 and return here when you will be more familiar with the model.

Coefficients of discrete polynomial controller can be entered either as coefficient of continuous PID controller or directly as polynomials in *z* operator.

PID controller is defined either by its coefficients K_p, K_i, K_d and K_{dd}

$$u(z) = (K_p - K_i T_s \frac{z}{z-1})\ e(z) - (\frac{K_d}{T_s} - \frac{K_{dd}}{T_s^2}\frac{z-1}{z})\ \frac{z-1}{z}\ F(z)\ y(z)$$

or by total gain G and time constants T_i, T_d and T_{dd},

$$u(z) = G\ (\ (1 - \frac{T_s}{T_i}\frac{z}{z-1})\ e(z) - (\frac{T_d}{T_s} - \frac{T_{dd}^2}{T_s^2}\frac{z-1}{z})\ \frac{z-1}{z}\ F(z)\ y(z)\ )$$

where

T_s is sampling period
y(z) is system output
e(z) is output error *w(z)* - *y(z)*
u(z) is system input
F(z) is first order discrete filter with time constant T_f.

Polynomial controller may be entered either in the general form

$$u(z) = -\frac{b(z)}{a(z)}y(z) - \frac{s(z)}{r(z)}w(z)$$

or in the r-s-t-form with common denominator

$$u(z) = \frac{-r(z)y(z) - t(z)w(z)}{s(z)}$$

or as open-loop compensator.

$$u(z) = \frac{s(z)}{r(z)}w(z)$$

The input filter is implemented as discrete polynomial filter in *z* operator with sampling period T_{sf}. The sampling period may be higher than the controller sampling period T_s.

$$y_f(z) = \frac{b(z)}{b(z)}y(z)$$

where

$y_f(z)$ is filtered system output
y(z) is physical system output.

2. Modelling

An attempt to model the system dynamics in detail leads to extremely complicated, not readable and not useful model. The engineer should decide what is the model used for and under what conditions it will work. In our case the model will be used for investigating the system dynamics with respect to control tasks. The system will operate in some working conditions and not all of the dynamical properties will be invoked. This leads to the assumptions which will simplify the derivation of the model.

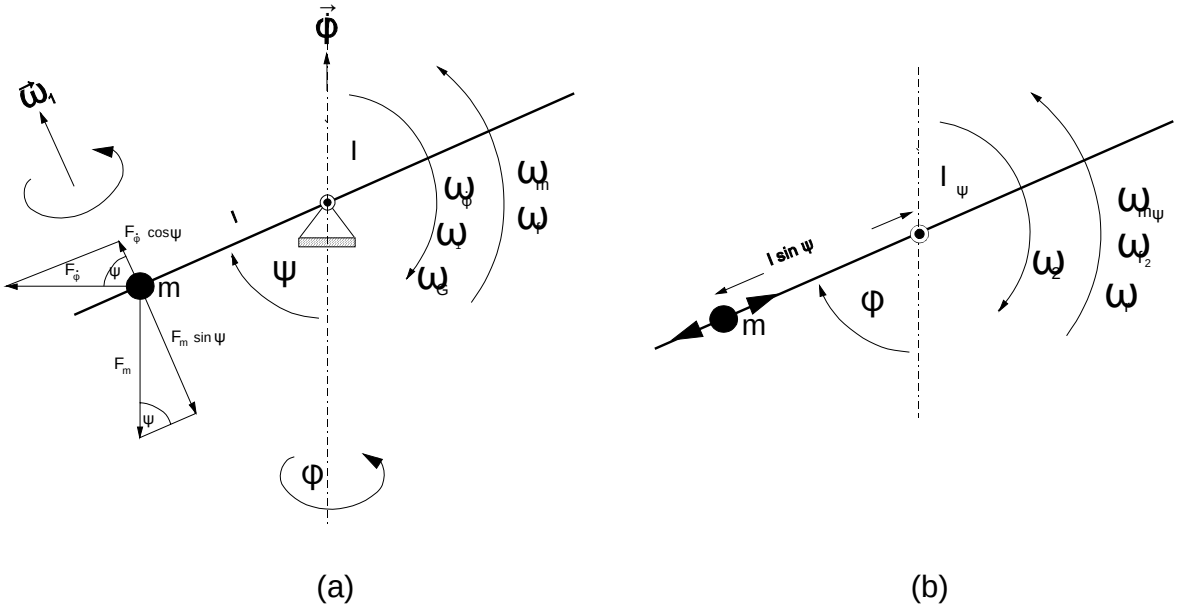
We propose two ways how to get the model. The first one is a systematic modelling method based on variational approach, i.e. Lagrange's equations. The second approach is a direct derivation of the model by computing the force balances. The second approach is described in the following text. Both methods lead to the model in the form of nonlinear differential equations.

2.1. Elevation dynamics modelling

Resolving the forces in the vertical plane acting on the helicopter body, based on the **Figure 2.1**, torque balance is computed

$$I\ddot{\psi} = \tau_1 - \tau_{\dot{\varphi}} - \tau_{f_1} - \tau_m - \tau_G \tag{1}$$

where $\zeta_{\dot{\varphi}}$ centrifugal torque [N.m]



(a) (b)

Figure 2.1 Torques acting on the body in the vertical (a) and horizontal (b) planes

ζ_G	gyroscopic torque [N.m]	
ζ_m	gravitational torque [N.m]	
ζ_{f1}	friction torque (Coulomb and viscous)	[N.m]
ζ_1	elevation driving torque (main propeller influence)	[N.m]
I	moment of inertia of the helicopter body around horizontal axis	[kg.m^2]

The following relations hold:

$$\tau_m = F_m l \sin\psi = mgl \sin\psi = \tau_g \sin\psi \tag{2}$$

$$\tau_{\dot\varphi} = ml\,\dot\varphi^2 \sin\psi \cos\psi = \frac{1}{2} ml\dot\varphi^2 \sin 2\psi \tag{3}$$

$$\tau_1 = k_{\tau_1}\,\tau_1^2 \tag{4}$$

$$\tau_{f_1} = C_\psi\, sign\dot\psi - B_\psi \dot\psi \tag{5}$$

$$\tau_G = k_G \dot\varphi \tau_1 \cos\psi \qquad for \quad \dot\varphi \ll \tau_1 \tag{6}$$

Some influences are neglected, e.g. stabilizing motor reaction torque and varying air resistance depending on the turnings of the main propeller. While the influence of the side motor on the elevation angle is almost negligible, varying damping of body oscillation in elevation is noticeable. The influence of the speed of the main propeller on friction torque in elevation is hardly to be modelled analytically and must be evaluated by an experiment and, if significant, nonlinear coupling must be introduced.

The gyroscopic effect is to be considered, however the equation for gyroscopic torque computation is simplified due to the assumption according to (6).

2.2. Azimuth dynamics modelling

The following equation shows the balance of torque in the horizontal plane, taking into account main forces acting on the helicopter body in the direction of φ angle:

$$I_\psi \ddot\varphi = \tau_2 - \tau_{f_2} - \tau_r \tag{7}$$

where

ζ_2	...	stabilizing motor driving torque	[N m]
ζ_{f2}	...	friction torque (Coulomb and viscous)	[N m]
ζ_r	...	main motor reaction torque	[N m]

$$I_\psi = I \sin\psi \tag{8}$$

$$\tau_2 = k_2 l_2 \sin\psi\, \tau_2^2 \tag{9}$$

$$\tau_{f_2} = C_\varphi\, sign\dot\varphi - B_\varphi \dot\varphi \tag{10}$$

Similar to the body dynamics in elevation, no connection between the speed of the side propeller and friction torque around vertical rotational axis has been introduced into the derivation of an analytical model of the helicopter dynamics. For the reaction torque ζ_r we refer to the block diagram of the overall system dynamics in **Figure 2.2**. Torque ζ_r is significant and arises from the torque generated by the main motor acting on rotating body.

2.3. DC motor and propeller dynamics modelling

The textbook model of a DC motor dynamics is to be adapted for the following reasons. The armature inductance is very low, Coulomb friction and resistive torque generated by rotating propeller in the air are significant. The resistive torque generated by rotating propeller depends on ζ in low and ζ^2 in high rpm. Starting with the control input voltage *u(t)* , the cause-and-effect equations are

$$\begin{aligned} i &= \frac{1}{R}(u - K_b\tau) \\ \tau &= K_i i \\ \tau_c &= C\,sign(\tau) \\ \tau_p &= B_p\tau - D_p\tau^2 \\ I\dot{\tau} &= \tau - \tau_c - B\tau - \tau_p \end{aligned} \tag{11}$$

where

			R	armature resistance	[ζ]
u	control input voltage	[V]	K_b	back-emf constant	[V.s]
i	armature current	[A]	K_i	torque constant	[N.m/A]
u_b	back emf	[V]	I	rotor and propeller inertia	[kg.m^2]
ζ	rotor angular velocity	[rad/s]	B	viscous-friction coefficient	[N.m.s]
ζ	motor torque	[N m]	C	Coulomb-friction coefficient	[N.m]
ζ_c	Coulomb friction load torque		B_p	air resistance coefficient (laminar flow)	
ζ_p	air resistance load torque		D_p	air resistance coefficient (turbulent flow)	

A propeller, which is attached directly to the motor rotor, generates torque while revolving. The quadratic, ventilator characteristic holds for the complete set of angular velocities considered

$$\begin{aligned} \tau_1 &= K_{\tau_1}\tau_1^2\,sign(\tau_1) \\ \tau_2 &= K_{\tau_2}\tau_2^2\,sign(\tau_2) \end{aligned} \tag{12}$$

where

ζ_1	angular velocity of the main propeller	[rad/s]
ζ_1	elevation torque generated by the main propeller	[N m]
ζ_2	angular velocity of the stabilizing propeller	[rad/s]
ζ_2	azimuth torque generated by the side propeller	[N m]

2.4. Sensor and power amplifier modelling

As the user communicates with the system via MATLAB REAL TIME TOOLBOX interface, all input/output signals are scaled into the interval <-1,+1> , where "1" is called Machine Unit and such a signal has no physical dimension. This will be referred in the following text as MU.

Incremental encoders are used for direct angle measurement. These parts have no dynamics and are considered to be linear in the whole extent of measured angles.

$$\begin{aligned} y_\psi &= k_\psi \psi - y_{\psi_0} \\ y_\varphi &= k_\varphi \varphi \end{aligned} \tag{13}$$

where

ζ	elevation	[rad]
y_ζ	elevation	[MU]
k_ζ	elevation constant	[MU/rad]
$y_{\zeta o}$	elevation sensor offset for ζ =0	
φ	azimuth	[rad]
y_φ	azimuth	[MU]
k_φ	azimuth constant	[MU/rad]
	azimuth offset is not significant	

Power amplifiers driven by the PC plug-in card are used for driving DC motors. They are linear and the function is simply described by

$$u_a = K\, u \tag{14}$$

where

u	computer output value	[MU]
u_a	armature voltage	[V]
K	amplifier gain	[V/MU]

2.5. Complete system dynamics

Block diagram of nonlinear dynamics of a complete system is to be assembled from the above derivations and the result is shown in **Figure 2.2**.

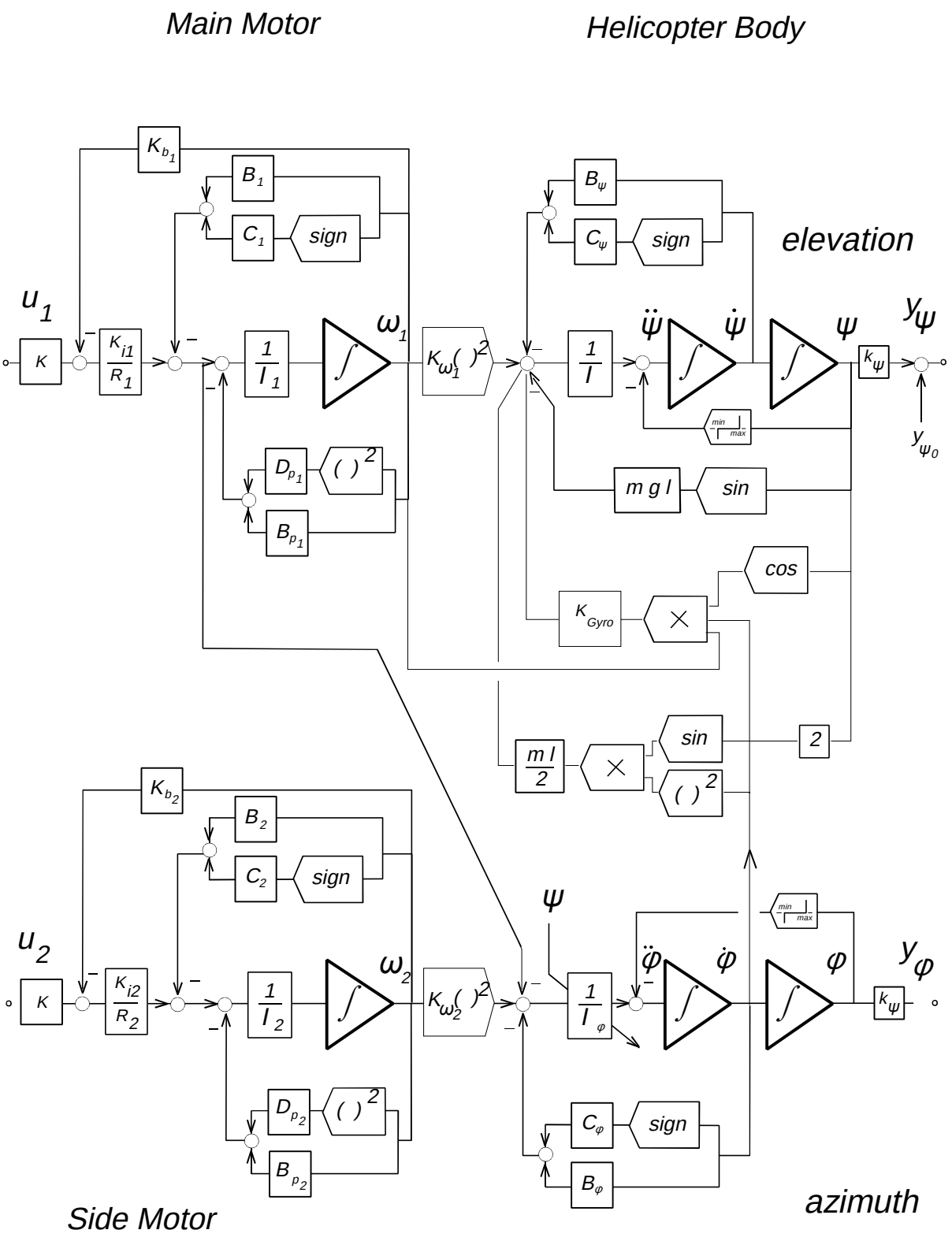


Figure 2.2 Block diagram of a complete system dynamics - theoretical model

3. Identification

There are two identification approaches for model parameters estimation:

(i) identification by parts by direct measurement of physically accessible parameters and identification of model subsystems ;
(ii) processing the input and output signals, considering the system as a black box.

The first method is time consuming but gives good understanding of the system behavior taking into account physical nature of the system functioning. The disadvantage is that not all of the model parameters are accessible by direct measurement.

The second approach is general, elegant, but is used when the system model is expected to be linear. This could be done around the chosen set point.

The following model subsystems are identified:

- Power amplifier and main DC motor
- Helicopter body in vertical plane
- Elevation angle sensor
- Power amplifier and stabilizing motor
- Helicopter body in horizontal plane
- Azimuth angle sensor

There are couplings between the subsystems as shown in the block diagram of the overall system dynamics in **Figure 2.2**.

3.1. Calibration of sensors

$k_\zeta , y_{\zeta 0} , k_\varphi$

Calibration of the angle sensors is straightforward. Offset in elevation shifts the zero output of the sensor from the zero vertical position so to have zero signal in the middle of the output scale. Typical values are

$k_\zeta = 1/\zeta$ MU/rad, $y_{\zeta 0} = -1/2\zeta$ MU , $k_\varphi = 1/\zeta$ MU/rad

3.2. Measurement of the gravitation torque of the body mass

$\zeta_g = m.g.l$

The task is to identify the torque ζ_g generated by the mass positioned in the center of the gravity of the helicopter body. The following torque balance equation is used

$$\tau_g \sin(\psi_1) = \tau\, mg l_0 \sin(\psi_2) \tag{15}$$

where

$$\tau_g = mgl \tag{16}$$

according to **Figure 3.2**.

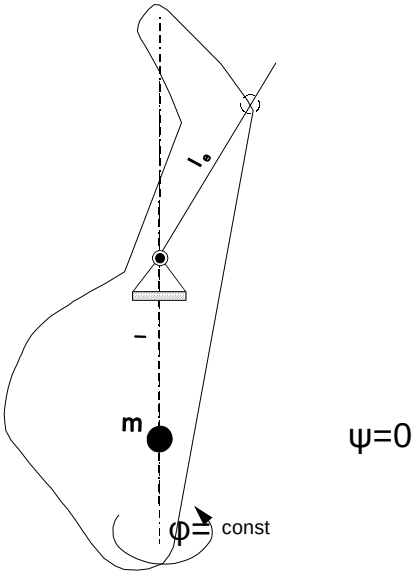


Figure 3.1 Helicopter position for zero elevation angle

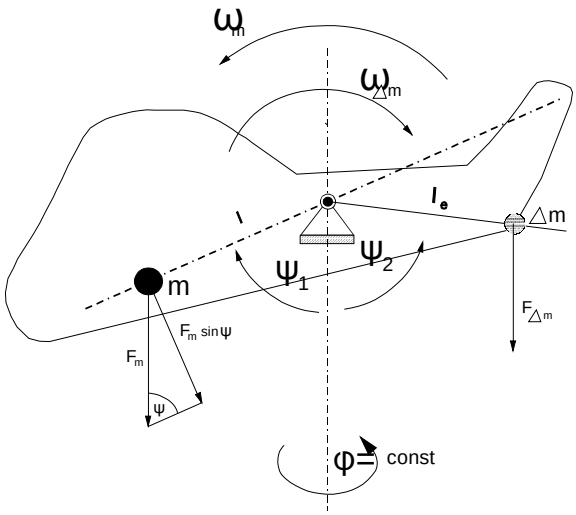


Figure 3.2 Weight balancing for elevation torque calibration

This coefficient is simply determined by adding an extra weight at the appropriate position and measuring an inclination angle as shown in **Figure 3.1** and **Figure 3.2**.

Note: The procedure of identification of ζ_g is simple, but must be done very carefully as this value will be used for derivation of all remaining parameters. To do number of experiments with different masses and their position is highly recommended in order to get reliable results.

Typical result:

$\zeta_g = 3.83\ 10^{-2}$ Nm

3.3. Identification of the motor - propeller static characteristic

Before we identify dynamics of the motor and helicopter body, the static characteristic of the actuator, i.e. motor-propeller, has to be known. According to the block diagram in **Figure 2.2**, one has to determine the relationships

$$\begin{aligned} \tau_1 &= f_1(u_1) \\ \tau_2 &= f_2(u_2) \end{aligned} \tag{17}$$

Experiment 3.3.1 : Identification of the physical parameters specified in the theoretical model

The theoretical model of the motor and propeller, with Coulomb friction neglected, is:

$$\begin{aligned} I_1\dot{\tau}_1 - D_{p_1}\tau_1^2 - \overline{B}_1\tau_1 &= K_1 u_1 \\ \tau_1 &= K_{\tau_1}\tau_1^2 \end{aligned} \tag{18}$$

where all linear negative feedbacks have been replaced by one, represented by the term $B_1\zeta_1$. Thus the analytic form of the static input/output characteristic is

$$\tau = \tau(u;a,b,c) = \frac{c}{2a^2}\,(b^2 - b\sqrt{b^2-4au} - 2au) \tag{20}$$

where

$$a = \frac{D_{p_1}}{K_1} \quad , \quad b = \frac{\overline{B}_1}{K_1} \quad , \quad c = K_{\tau_1} \tag{21}$$

The following balance of the moments holds in the vertical plane, when the azimuth is fixed and the steady state is achieved,

$$\begin{aligned} (\tau_g - \tau\, m\, g\, l_0)\sin\psi &= \tau_1 \\ y_\psi &= k_\psi\psi - y_{\psi_0} \end{aligned} \tag{22}$$

The input/output static characteristic of the system from u_1 to y_ζ is then described analytically by

$$= y_\psi(u_1;a,b,c) = y_{\psi_0} - k_\psi \arcsin\left(\frac{c}{2a^2(\tau_g - \tau\, mgl_0)}(b^2 - b\sqrt{b^2-4au_1} - 2au\right. \tag{23}$$

with unknown parameters *a,b,c*. To identify these parameters, number of measurements of the elevation angle y_ζ has to be done for different values of u_1. An extra mass ζm fixed to the helicopter body in an appropriate place with in the distance l_0 from the horizontal axis might be used to balance the helicopter body around the vertical position. After the collection of data, values of y_ζ are recomputed in order to get the vector of transformed values *y* according to (25). Then we solve the problem of nonlinear regression or nonlinear curve fitting using the fit-function in the form of

$$y_{model} = y_{model}(a,b,c) = \frac{c}{a^2}(b^2 - b\sqrt{b^2-4au_1} - 2au_1) \tag{24}$$

We fit these values to the values of *y* get from the measurements via the transformation

$$y = 2(\tau_g - \tau\, mgl_0)\sin\left(\frac{y_\psi - y_{\psi_0}}{k_\psi}\right) \tag{25}$$

The nonlinear curve-fitting problem is frequently formulated as the non-linear least squares problem

$$\min_{a,b,c} \sum_{k=1}^{N} \left(y(u_{1k}) - y_{model}(a,b,c;u_{1k}) \right)^2 \tag{26}$$

Note 1: If MATLAB and the OPTIMIZATION TOOLBOX is available, then the parameters a,b,c are obtained by calling the function leastsq. This function implements the Levenberg-Marquardt method with a mixed quadratic and cubic line search procedure. A Gauss-Newton method could be selected as an option.

Note 2: The proposed solution for the determination of the motor and propeller parameters is badly numerically conditioned. The goal function has many local minima. The solution is good only if there is a good estimate of the initial values of the unknown parameters.

Experiment 3.3.2 : Identification of the parameters of an empirical model for the main motor and propeller static characteristic

The alternative approach for identification of the motor-propeller static characteristic is to suggest a simpler block structure which will cope with the input-output data, but no correspondence to the physical structure will be provided. The following input-output function with unknown parameters *a, b* is proposed in order to preserve the dominant ventilator characteristic

$$\tau_1 = \tau_1(u_1;a,b) = a u_1^2 - b u_1 \tag{27}$$

To identify these parameters, number of steady state measurements of the elevation angle y_ζ has to be done for different values of u_1. An extra mass ζm fixed to the helicopter body in an appropriate distance l_0 from the horizontal axis might be used to balance the helicopter body around the vertical position. After the collection of data, values of gravitation torque are computed according to (28)

$$\tau_{g1} = (\tau_g - \tau\, m g l_0) \sin(\frac{y_\psi(u_1) - y_{\psi_0}}{k_\psi}) \tag{28}$$

The nonlinear curve-fitting problem is formulated as the non-linear least squares problem

$$\min_{a,b} \sum_{k=1}^{N} (\tau_{g1}(u_{1k}) - \tau_1(a,b;u_{1k}))^2 \tag{29}$$

The typical values of the parameters and measured characteristics are

a_1 = 0.105 N.m/MU^2 b_1 = 0.00936 N.m/MU
Figure 3.6, Figure 3.7

Experiment 3.3.3 : Identification of the parameters of an empirical model of a static characteristic for the side motor and propeller

The system is unstable from the input u_2 to the output y_φ as shown in the block diagram in **Figure 2.2**. To use the same procedures as for elevation dynamics identification one can stabilize the system mechanically, by inclining the vertical axis of the helicopter body rotation. The $\zeta/2$ inclination is recommended. We fix the elevation angle and we do the same steps as for the identification of the nonlinear model for the main motor and the elevation dynamics. First an external gravitation torque ζ_{gext} artificially added to the system dynamics due to the inclination of the azimuth axis is measured, see Problem 3.2. Then the parameters of a cubic function are

identified as in the Experiment 3.3.2. The stabilizing feedback via ζ_{gext} will be removed from the model after all parameters are measured.

Typical result:

a_2 = 0.033 N.m/MU^2 b_2 = 0.0294 N.m/MU

3.4. Identification of the helicopter body dynamics in elevation

B_ζ, I

Experiment 3.4.1 : Fitting of a linear second order response around the setpoint

The setpoint is set by the steady state torque generated by the rotating propeller. Nonlinear model in **Figure 3.4** is replaced by the linear one according to **Figure 3.4** for the identification purposes and correction of the changed setpoint is introduced. The linearization is valid for small changes in the elevation angle.

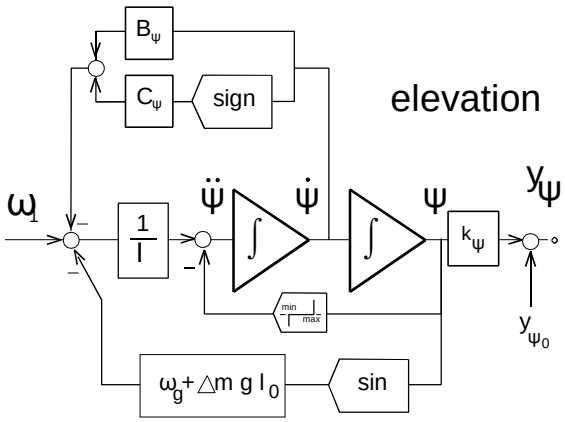


Figure 3.3 Nonlinear dynamics of the helicopter body in elevation

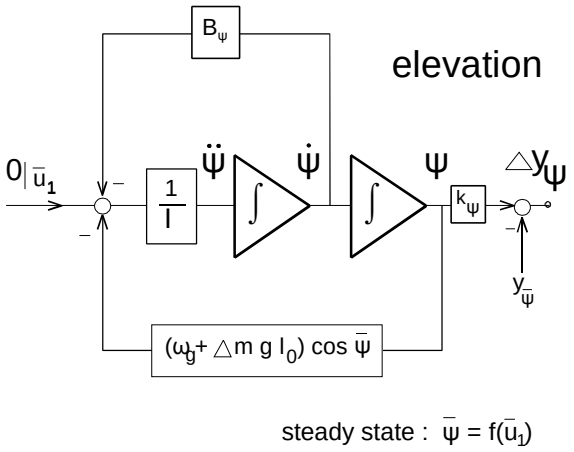


Figure 3.4 Linearized dynamics around the setpoint in elevation

The parameters B_ζ, I are identified by approximating the recorded response to nonzero initial conditions in elevation. Linear second order prototype system is used as an approximation and its parameters are tuned according to the following equations.

Prototype second order transfer function:

$$\frac{Y(s)}{U(s)} = \frac{\tau_n^2}{s^2 - 2\tau_n s - \tau_n^2} \tag{30}$$

Unit step response of the prototype second order system:

Conditional (damped) frequency:

$$\tau = \tau_n\sqrt{1-\tau^2} \tag{31}$$

Maximum overshoot:

$$t_{max} = \frac{\tau}{\tau_n\sqrt{1-\tau^2}} \tag{32}$$

$$y_{max} = 1 - e^{-\tau/\sqrt{1-\tau^2}} \tag{33}$$

Delay time:

$$t_d \int \frac{1 - 0.7\tau}{\tau_n} \tag{34}$$

Rise time:

$$t_r \int \frac{0.8 - 2.5\tau}{\tau_n} \tag{35}$$

Settling time:

$$t_s \int \frac{3.2}{\tau_n} \tag{36}$$

Parameter correspondence:

$$I = \frac{\tau_g \cos\overline{\psi}}{\tau_n^2} \tag{37}$$

$$B_\psi = 2\tau_n I \tag{38}$$

Damped oscillations are to be recorded. The coordinates of the peaks are used in the equations for computation of the maximum overshoot and the natural frequency and damping factor are computed. These values are then used in the equations (23), (24) and inertia and damping factor are derived. To improve the results, several peaks could be read and the mean value from the parameters could be used instead of a single computation.

Note: Instead of peak to peak fitting procedure, point to point fit by the least squares could be applied for comparison.

Typical results and recordings:

B_ζ = 1.84 10^{-3} kg.m^2/s
I = 4.37 10^{-3} kg.m^2
Figure 3.8

Experiment 3.4.2 : Fitting of a nonlinear second order response to the measured data

The nonlinear model of the form

$$\begin{aligned} I\ddot{\psi} - B_\psi\dot{\psi} - \tau_g \sin\psi &= a u_1^2 - b u_1 \\ y_\psi &= k_\psi \psi - y_{\psi 0} \end{aligned} \tag{39}$$

parametrized in I and B_ζ generates the time response to nonzero initial conditions of the elevation angle, for given constant input u_1 and known parameters a,b,ζ_g, k_ζ and $y_{\zeta 0}$. Formulate the optimization problem and use the algorithm described in the problem 3.2.

Experiment 3.4.3 : Identification of the nonlinear dumping function $B_\zeta(\zeta_1)$

The value of the parameter B_ζ increases with the velocity of the main propeller. This influence is noticeable in the recordings. Identify the value of B_ζ in different setpoints defined by the nominal velocities of the main motor and plot the corresponding function $B_\zeta(u_1)$. Discuss the obtained result and formulate the assumptions under which the parameter B_ζ can be considered constant.

Typical results

Figure 3.8 , Figure 3.9

3.5. Identification of the helicopter body dynamics in azimuth

B_φ, I_φ

Experiments 3.4.1, 3.4.2 and 3.4.3 are to be repeated for mechanically stabilized azimuth rotation as was prepared in the Experiment 3.3.3.

Typical results:

B_φ = 8.69 10^{-3} kg.m^2/s
I_φ = 4.14 10^{-3} kg.m^2
Figure 3.11, Figure 3.12

3.6. Motor and propeller dynamics identification

T_1, T_2

The static characteristics of the subsystem has been identified in the Problem 3.3. The identification of the physical parameters taken from the general block diagram derived from the theoretical modelling is not possible. None of the internal signals as the voltage, current or angular velocity can be measured. In such a case we identify the subsystem dynamics as a black box with a predetermined input-output model structure. The model is supposed to be composed of a linear dynamic and nonlinear static part. The static nonlinear part has been identified in the Problem 3.4. The dynamics could be neglected in some cases or modelled by the first or second order transfer function

$$\frac{U_{d1}(s)}{U_1(s)} = \begin{cases} 1, \\ \dfrac{1}{T_1 s - 1}, \\ \dfrac{1}{(T_1 s - 1)^2} \end{cases} \tag{40}$$

The output of the subsystem, i.e. the elevation driving torque is then computed as

$$\tau_1 = a\, u_{d1}^2 - b\, u_{d1} \tag{41}$$

Experiment 3.6.1 : Least squares step response curve fitting for the main motor dynamics identification

This experiment takes the advantage of the already identified elevation dynamics of the helicopter body and nonlinear static characteristic of the motor and propeller subsystem. The only parameter to be identified is the time constant T_1 of the motor. Thus response to the step in the signal u_1 is going to be recorded and compared with the nonlinear model response parametrized in T_1. The procedure of fitting described in the Problem 3.3 can automatically adjust the initial estimate of the parameter.

Typical result:

T_1 = 0.3 s for the 2nd order delay
Figure 3.10

Experiment 3.6.2 : Least squares step response curve fitting for the side motor dynamics identification

The same procedure as the one described in the Experiment 3.5.2 is applied for the mechanically stabilized azimuth rotation.

Typical result:

T_2 = 0.25 s for the 2nd order delay Figure 3.13

3.7. Main Motor Reaction Cross Coupling Identification

The identification of the cross coupling dynamics influenced by the driving torque of the main motor rotor is possible only as the black box identification as no internal signals are measurable. Fortunately no significant nonlinearity is located between the input u_1 and the output y_φ and the dynamics of the helicopter body in azimuth has been already identified in the Experiment 3.5. The remaining black box part is assumed to be linear, first order system with one stable pole and zero as follows from the general block diagram in **Figure 2.2**

The linear model to be identified is

$$\frac{\tau_r(s)}{U_1(s)} = -K_r \frac{T_{0r}\, s - 1}{T_{pr}\, s - 1} \tag{42}$$

and remaining, previously identified mechanically stabilized dynamics in azimuth has the transfer function

$$\frac{Y_\varphi(s)}{\tau_r(s)}_{|\bar{\varphi}} = \frac{k_\varphi}{I_\varphi s^2 - B_\varphi s - \tau_{gext}\cos\bar{\varphi}} \tag{43}$$

Experiment 3.7.1 : Estimation of the parameters of the first order system for the model of the cross coupling from the main motor to the azimuth

K_r, T_{0r}, T_{pr}

Azimuth angle y_φ response to the step made from the setpoint input u_1 is recorded and fit to the model response parametrized in three parameters K_r, T_{0r}, T_{pr}. The fitting procedure described in the Experiment 3.3.1 can be used.

Typical results:

K_r = 0.00162 N.m/MU T_{0r} = 2.7 s T_{pr} = 0.75 s Figure 3.14

3.8. Gyroscopic cross coupling identification

We assume that the dynamics of the main motor is faster than the dynamics of the helicopter body in the azimuth. Under this assumption one of the gyroscopic coupling input which is originally driven by the main motor speed, according to **Figure 2.2**, could be connected directly to the input u_1. The only parameter to be identified is the gain of the nonlinear coupling K_{Gyro}.

Experiment 3.8.1 : Identification of gyroscopic gain

K_{Gyro}

The helicopter is put to its standard position, i.e. azimuth axis is vertical. The main motor is started and driven in a constant speed. Steps in the gyroscopic torque are generated by manually changed azimuth angle. If several cycles "stop - move.clockwise - stop - move anticlockwise" are generated, the parameter K_{Gyro} can be roughly estimated from the elevation angle change. The following torque balance equation is used for the derivation of the K_{Gyro} estimate

$$a_1 u_1^2 - b_1 u_1 - \tau_g \sin\psi - K_{Gyro}\dot{\varphi}\, u_1 \cos\psi = 0 \qquad (44)$$

Typical result

K_{Gyro} = 0.015 N.m/s Figure 3.15

3.9. Revision of the block diagram based on measured data

The task is to redraw the block diagram from **Figure 2.2** which describes the complete dynamics and internal structure of the model and sketch revised block diagram corresponding to actual data measured on the real model.

The results from the identification experiments are used and the block diagram is modified according to Figure 3.5.

3.10. Linearization of an updated nonlinear model

A	B	C	D	state space model
$\frac{Y_\zeta(s)}{U_1(s)}$	$\frac{Y_\zeta(s)}{U_2(s)}$	$\frac{Y_\varphi(s)}{U_1(s)}$	$\frac{Y_\varphi(s)}{U_2(s)}$	input/output model

Experiment 3.10.1 : Setpoint selection and linearization

Select different setpoint conditions, characterized by the steady state values of y_ζ, y_φ and corresponding inputs u_1, u_2. Derive the state space linear model, compute the matrices *A, B, C, D* and draw the block diagram of the linearized dynamics. Compute the transfer function matrix. Both models are to be parametrized for the setpoint conditions.

Experiment 3.10.2 : Model simplification

Specify the conditions under which the couplings are not necessarily considered.

3.11. Linear input-output transfer function identification

The task is to identify the linear model describing the dynamics between inputs and outputs based on input output measurements. Parametric Auto Regressive model with External inputs (ARX) that corresponds to

$$\frac{Y(s)}{U(s)_{\int_{\bar{\psi}}}} = \frac{b_1 - b_2 q^{-1} - \ldots - b_m q^{-m}}{1 - a_1 q^{-1} - \ldots - a_n q^{-n}} \qquad (45)$$

where q^{-1} is the delay operator, is to be identified. Least Squares estimation method is used in straightforward manner to solve overdetermined linear equations resulting from collected data and the above model.

Experiment 3.11.1 : Least squares identification of the parameters of the linear model

Stabilize the system mechanically as in the Experiment 3.3.3. Let the system settle in the steady state characterized by constant elevation and azimuth angles. Use different signals to move the system around the steady state and record the input output data. Suggest the order of the polynomial in numerator and denumerator for the particular transfer function to be identified. Run the Least Squares algorithm and discuss the results.

Note: The success of the method depends on the input signal selection as different signals give different results. Random signal, oscillating around the set point is used, input and output history is recorded and processed by the MATLAB arx function. Two poles found describe the mechanical part and the remaining belong to the DC motor and propeller dynamics. Two mechanical poles should be close to those identified by previous experiment with physical pendulum. Similarity in value of two complex poles gives the estimate of correctness of the experiment with least squares method.

Experiment 3.11.2 : Validation of the identified linear model

Use arbitrarily tuned PID controller which gives the steady closed loop behavior and compare the responses of simulated and real system to the signals driving the desired angles around the setpoint for which the linearized model has been computed.

4. Sampling Frequency Selection

T_s

Selection of sampling period is necessary for implementing digital controllers. Too long a sampling period will make impossible to reconstruct the continuous-time signal. Too short a sampling period will increase the load on the computer. Two approaches are considered, selection based on frequency content of sampled signals and common empirical rule using the knowledge of a step response.

4.1. Spectral analysis approach for sampling frequency selection

Frequency content of an output signal of a system is described well by the Bode plot of the corresponding transfer function, of course under assumption that the input signal which drives the system is white noise. This will be never true, but theoretically it is the worst case we should investigate. To preserve frequency content of sampled signals we follow the conclusions of Shannon theorem. Nyquist frequency is chosen for the frequency where the magnitude of the Bode plot drops to 3% of its global maximum. Sampling frequency must be twice as large as the Nyquist frequency. This procedure gives relatively short sampling period.

Experiment 4.1.1 : Select the sampling frequency for elevation output and compare the Bode plot for continuous and sampled signal.

Experiment 4.1.2 : Select the sampling frequency for azimuth output.

4.2. Sampling frequency selection based on step response analysis

Sampling period is usually characterized by a variable that is dimension-free and has a good physical interpretation. For oscillatory systems it is natural to normalize the period of oscillation; for nonoscillatory systems, the rise time is a natural normalizing factor. N_r is introduced as the number of sampling periods per rise time,

$$N_r = \frac{T_r}{T_s} \tag{46}$$

where T_r is the rise time. It is reasonable to choose the sampling period so that

$$N_r \int 4 - 10 \tag{47}$$

Experiment 4.2.1 : Record the step responses for all input-output combinations and find the sampling frequency according to rules given in Problem 4.2. Compare the results with those found in the Experiment 4.1.1 and 4.1.2.

5. Controller Design

5.1. PID controller design for SISO configuration

The task is to design two PID Single Input Single Output controllers without any attempt of eliminating the influence of couplings in the system dynamics.

The modified continuous time PID controller law

$$U(s) = K\left[W(s) - Y(s) - \frac{1}{sT_i}(W(s) - Y(s)) - \frac{sT_d}{1 - s\frac{T_d}{N}}Y(s)\right] \tag{48}$$

is used. The modification of the textbook version is that the derivative term process thus the output of a system and not the error signal.

Discrete version of a PID controller

Continuous version of a PID controller is discretized by the following method

Proportional term

$$P(t) = K(w(t) - y(t)) \tag{49}$$

Integral term - rectangular approximation

$$I(t) = \frac{K}{T_i}\int_{\cdot}^{t} e(\tau)\,d\tau \qquad \int \qquad I(kT_s - T_s) = I(kT_s) - \frac{KT_s}{T_i}e(kT_s \tag{50}$$

Derivative term - backward difference approximation

$$\frac{T_d}{N}\frac{dD}{dt} - D = -KT_d\frac{dy}{dt}$$

$$\int \tag{51}$$

$$D(kT_s) = \frac{T_d}{T_d - NT_s}D(kT_s - T_s) - \frac{KT_{dN}}{T_d - NT_s}(y(kT_s) - y(kT_s - T_s)$$

Resulting form

$$u(kT_s) = P(kT_s) - I(kT_s) - D(kT_s) \tag{52}$$

Experiment 5.1.1 : Implementation of Digital PID Controller and PID Parameter Tuning for Elevation control.

T_{ie} , T_{de} , K_e

Consider the main motor as an actuator and elevation angle as measured output regardless the influence of the motion in azimuth. Implement the discrete time version of the PID controller under REAL TIME TOOLBOX for MATLAB. Use root locus technique for tuning of the parameters K, T_i, T_d. Plot the root locus in *s* and *z* planes. Test the influence of changes in the sampling rate on the location of closed loop poles. Move the dominant poles of the closed loop transfer function by manipulating with the PID parameters in order the poles have the same natural frequency and the lowest natural damping ratio. This tuning results in small overshoot and short rise time of a step response. Plot the influence of T_d , T_i to the change in closed loop root contours and parametrize the curves in the gain K. Mark the dominant poles, plot the damping coefficient ζ, overshoot and response time with respect to gain variation.

Experiment 5.1.2 : Implementation of Digital PID Controller and PID Parameter Tuning for Azimuth control.

T_{ia} , T_{da} , K_a

Consider the side motor as an actuator and azimuth angle as measured output regardless the influence of the cross couplings between the elevation and azimuth dynamics. Implement and tune a PID controller for a SISO case using the same procedure as for the elevation control loop.

Experiment 5.1.3 : Evaluation of simultaneous running of PID azimuth and elevation control

Compare the simulation and reality for the case of two independently designed PID controllers for elevation and azimuth control and evaluate the influence of the cross couplings.

Experiment 5.1.4 : PID control with coupling elimination

K_c

The results achieved up to now does not solve the basic problem in Helicopter control which is decoupling effect of a controller.

Assuming that the dynamics of DC motors is faster then the dynamics of a helicopter body, the input u_1 influences the torque driving the azimuth without any delay. This static coupling can be easily eliminated by the coupling between the outputs of separately designed single loop PID controllers. The block diagram in **Figure 5.1** shows the idea of coupling elimination by additional feedback.

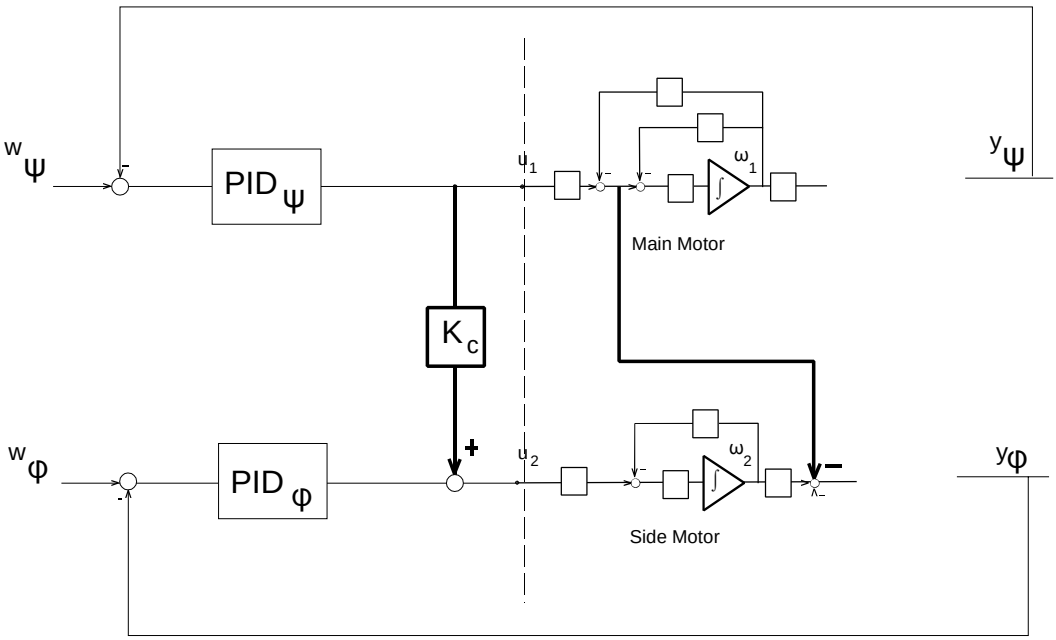


Figure 5.1 Approximate elimination of a coupling from main motor to azimuth

To tune the corrector, the cross gain K_c must be chosen. Find the gain in order to reduce the coupling from the main rotor reaction. Use the static characteristic and gain of the motors in corresponding direction to tune the gain K_c. Introduce the nonlinear gain if necessary. Compare the behavior of the system with coupling elimination to the results of the Experiment 5.1.3.

5.2. Pole placement design of state feedback controller

The main idea of this design method is that the location of the closed loop poles of a linear system determine the closed loop system dynamics. In this problem the poles of a closed loop system are placed according to the position specified by the designer. This may be done only by the state feedback. The block structure of the closed loop system enabling us to place the poles arbitrarily is shown in **Figure 5.2** , where *A*, *B*, and *C* are the matrices of the linearized model identified in the Problem 3.9 and *K* is a matrix of feedback gains to be designed. The discrete time model is considered, but the same procedure is applied when continuous time model of the process is used.

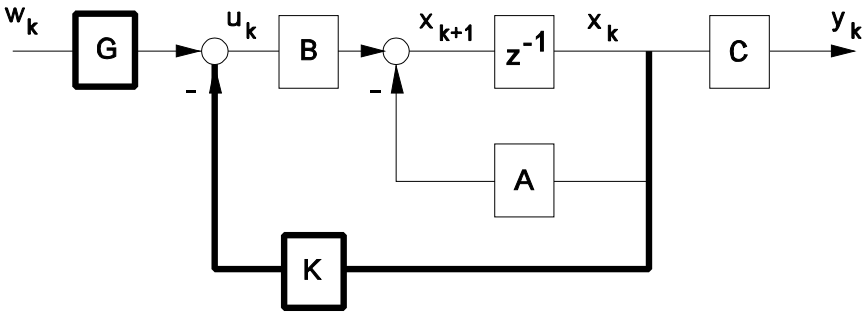


Figure 5.2 Block structure of state feedback control

For static decoupling a gain matrix *G* is added. The matrix should be computed so that the transfer matrix from *w* to *y* is identity matrix, thus

$$G = [C(I - (A - BK))^{-1} B]^{-1} \tag{53}$$

Experiment 5.2.1 : Design the state feedback controller for stabilization of an elevation angle

This is an experiment for Single Input Single Output (SISO) Controller design. Fix the azimuth rotation and consider fourth order dynamics between the input u_1 and elevation according to the block diagram in Figure 3.5. Linearize the model around the given setpoint. Choose the position of four closed loop poles and compute the vector of feedback gains. Simulate the system behavior in the closed loop and record the angle and control action signals. Repeat the design and simulation for number of different pole locations and comment the change in the dynamics. Check the magnitude of the control action as it should stay within the interval <-1,+1> , allowed by the PC to system interface.

Note: When working under MATLAB and CONTROL SYSTEM TOOLBOX, one may utilize the function place to compute the feedback gains.

Experiment 5.2.2 : Design the state feedback controller for stabilization of an azimuth angle

SISO Controller design. Fix the elevation angle and consider fourth order system between input u_2 and azimuth according to the Figure 3.5. Repeat the tasks formulated in the Experiment 5.2.1.

Experiment 5.2.3 : Design the state feedback controller for azimuth and elevation tracking

This is a Multi Input Multi Output (MIMO) control system design. Design a matrix *G* for decoupling. This is to minimize the influence of one angle to another. Repeat the tasks from the Experiment 5.2.3 and modify the structure of the feedback in order to incorporate the nonzero input which has to be followed.

5.3. Pole placement design of state feedback controller with observer

As just two of the state variables are directly measurable, we are not able to close the state feedback in reality. In such a case we reconstruct the missing signals from the dynamic system called an state estimator or state observer. This works as a software sensor and estimates elevation and azimuth speed as well as the states of the DC motors, i.e. speed and acceleration. The observer is implemented according to the following block diagram.

The gain *L* of the observer, which influences the position of observer poles, is to be designed. The input data are *A* and *C* matrices of the system model and desired pole locations of an observer. The task for the designer is prescribing the observer's speed by designing the pole locations.

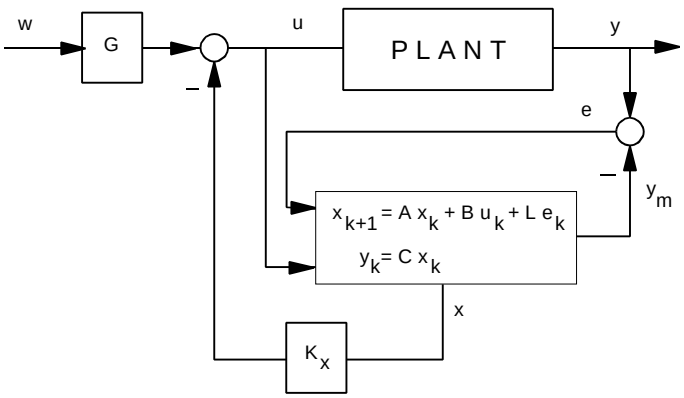


Figure 5.3 State feedback control with an observer

It is not easy to prescribe the pole locations for the regulator and the observer as well. An unexperienced designer should check whether the magnitude of the output of the controller is not in saturation. When in saturation, the desired speed of the closed loop system must be slowed down. The observer should be at least twice as fast as the closed loop system.

Experiment 5.3.1 : Full order observer design

Design a full rank observer able to provide a controller with the signals corresponding to all system states. Utilize the location of poles designed for the control loop in the Experiment 5.2.1 and specify the location of poles for the observer according to the guidelines given in the Problem 5.3. The same function place as for the controller design can be used for computing of matrix *L*. The observer is to be designed for different system configuration as in the Experiments 5.2.1 - 5.2.3. Implement the observer under REAL TIME TOOLBOX for MATLAB and

compare the output of the controller corresponding to the angle with directly measured signal. Explain the differences.

Experiment 5.3.2 : Reduced order observer design

As the measurements of the angles are supposed to be more accurate than any estimate of the state based on that measurement, it is logical to use the measurements instead of their estimate for the control feedback. The task is to design a reduced order observer for all input-output configurations considered.

5.4. Pole placement design of state feedback controller with observer and integral action

The problems 5.2 and 5.3 ignore nonlinearities like Coulomb friction in the system model used for the state feedback controller design. In this case the high gain in low frequencies in the controller is essential. An integral action must be part of the controller behavior. The improved behavior is designed by incorporating integrators according to the block diagram in Figure 3.

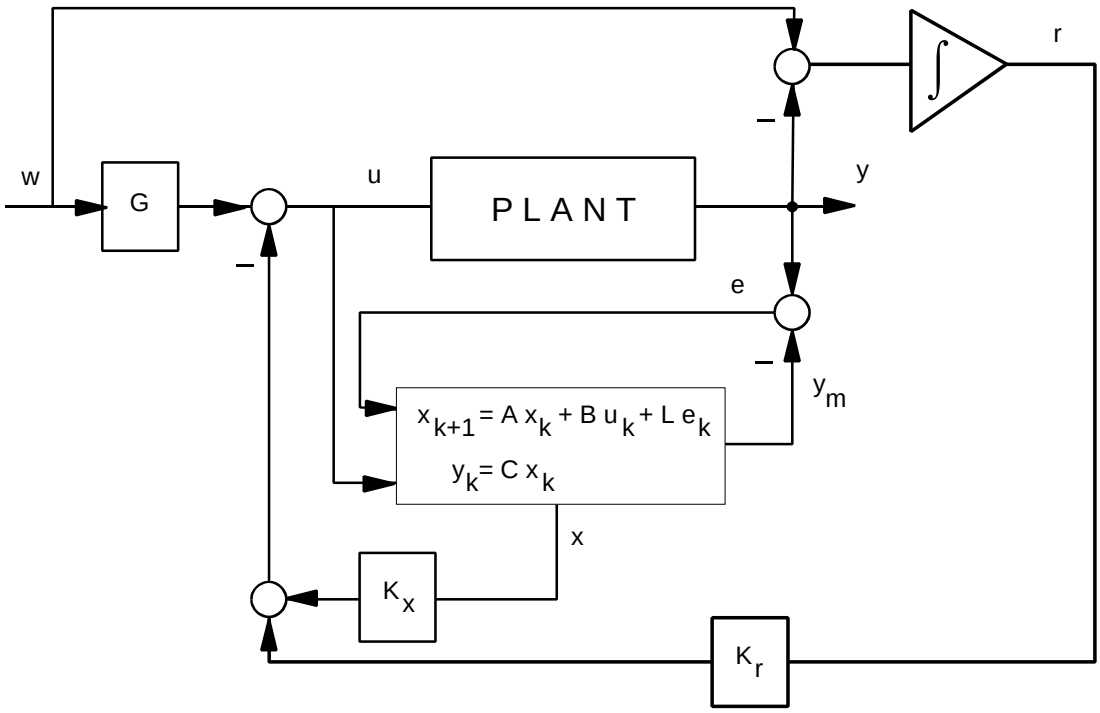


Figure 5.4 State feedback control with an observer and integral action

The order of the control system is increased by two as two integrators are added according to the block diagram. Then the same design procedure as in Problem 5.2 and 5.3 is applied for the computation of the feedback gain matrix.

Experiment 5.4.1 : State feedback design with integral action

6. Figures

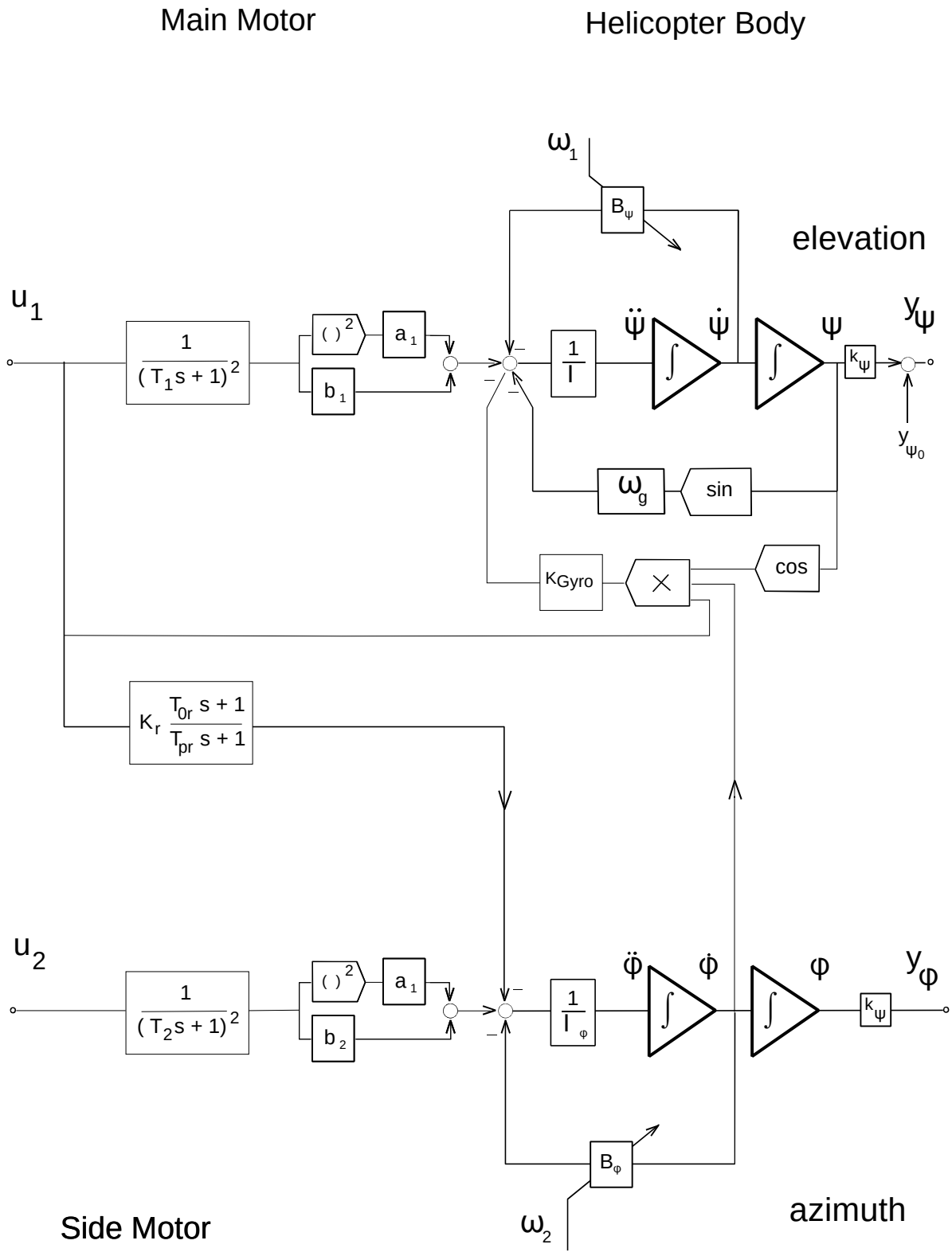


Figure 3.5 Block diagram of the nonlinear dynamics - empirical model

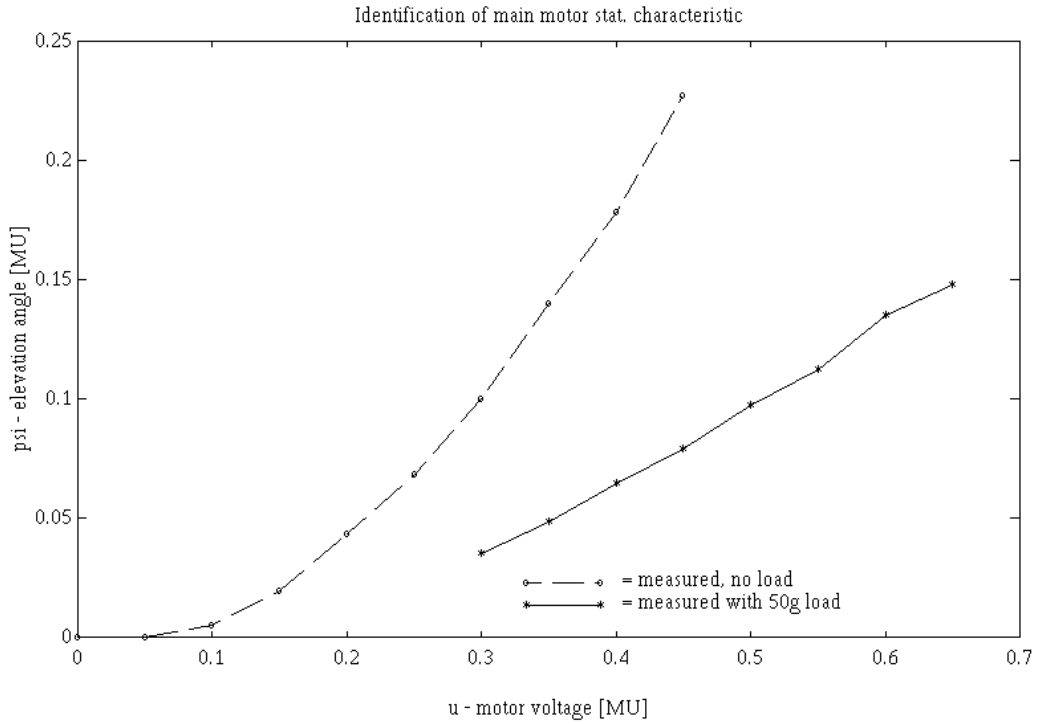


Figure 3.6 Main motor and propeller static characteristic (additional mass position l_0=0.135m)

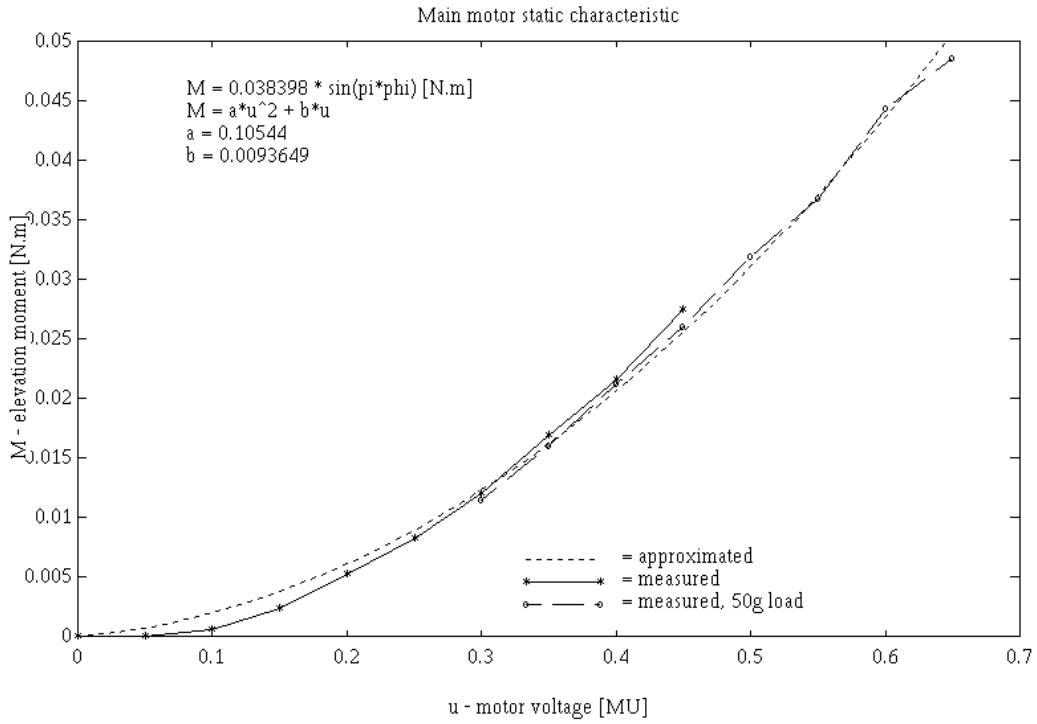


Figure 3.7 Main motor and propeller static characteristic approximation

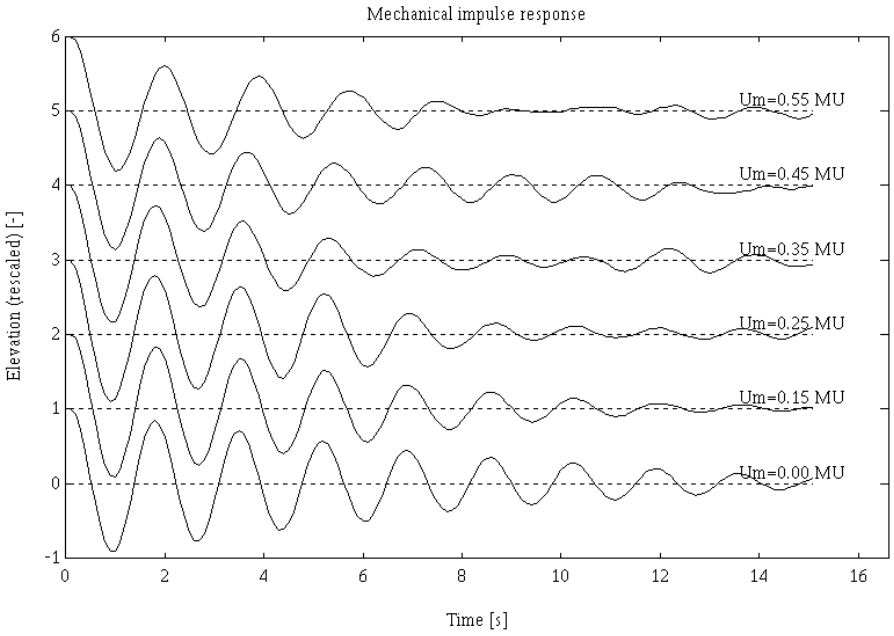


Figure 3.8 Time response of the helicopter body in the elevation angle to the nonzero initial condition, constant input u_1 set to different levels

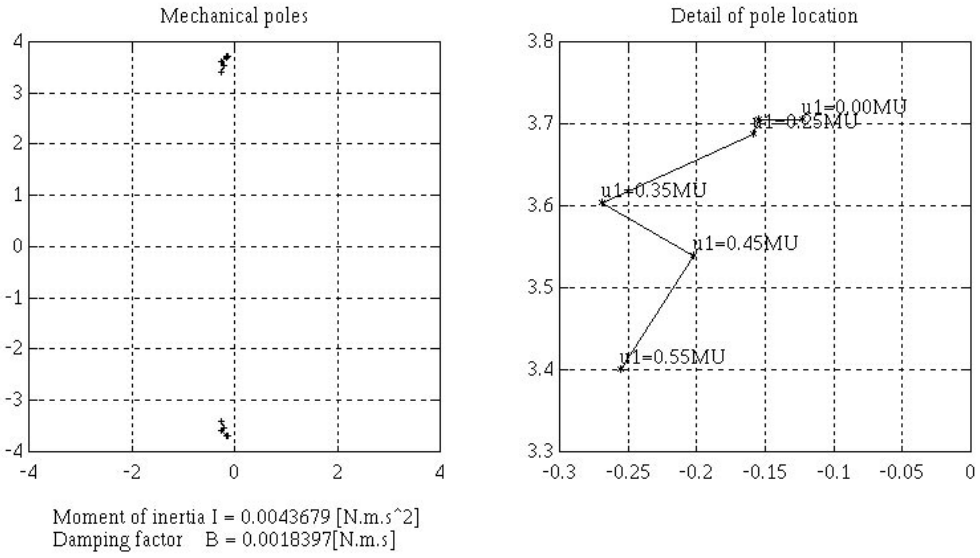


Figure 3.9 Root contours of the helicopter body elevation transfer function poles dependent on the varying dumping effect of the main propeller speed

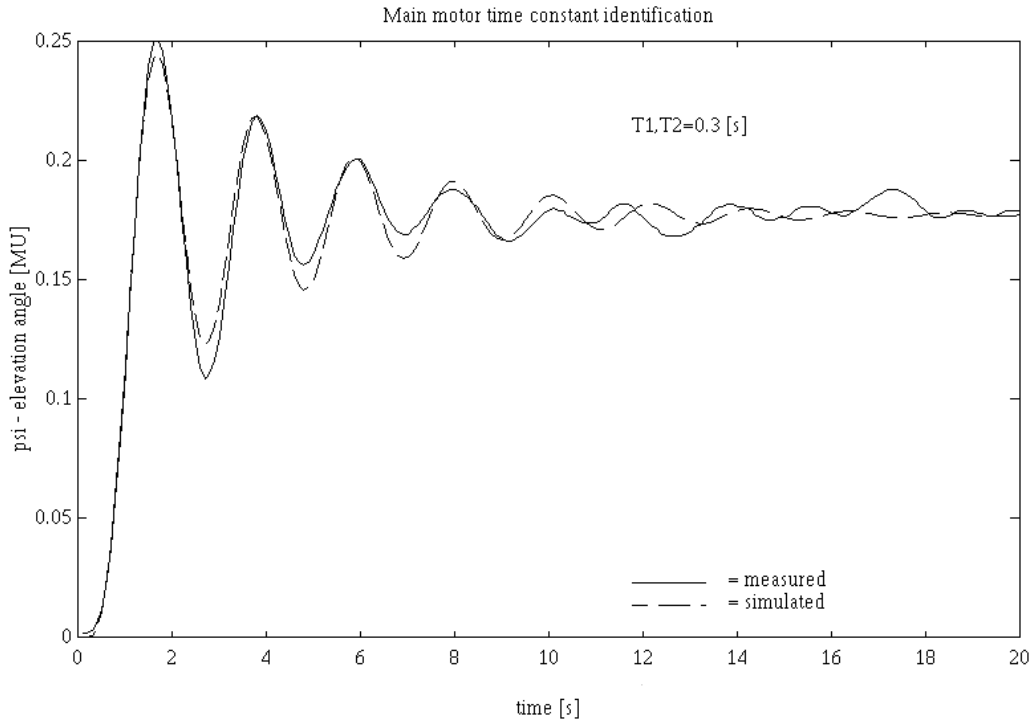


Figure 3.10 Response of the elevation angle to the step in the main input signal - model and reality

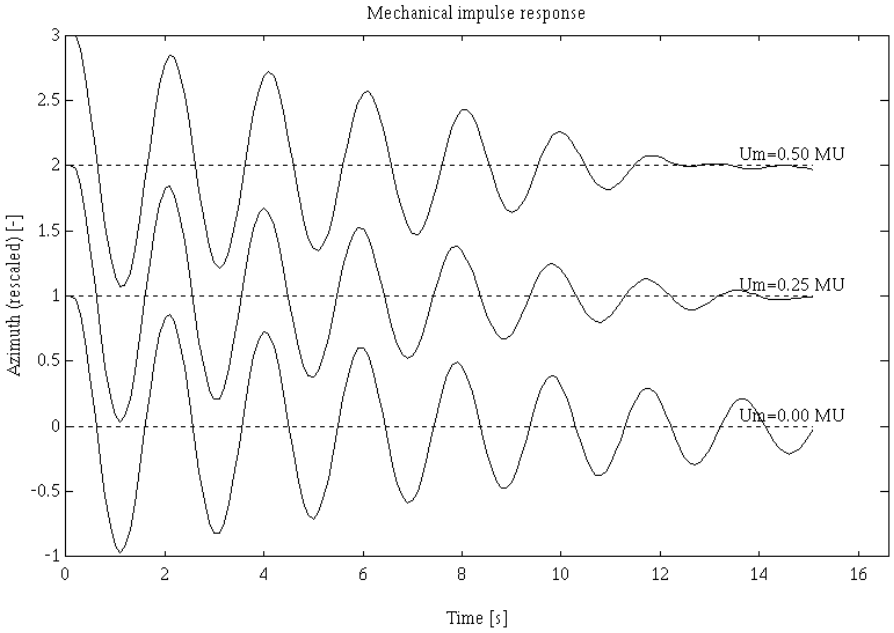


Figure 3.11 Damped oscillation of the helicopter body in the artificially stabilized dynamics, for the derivation of I_φ and B_φ

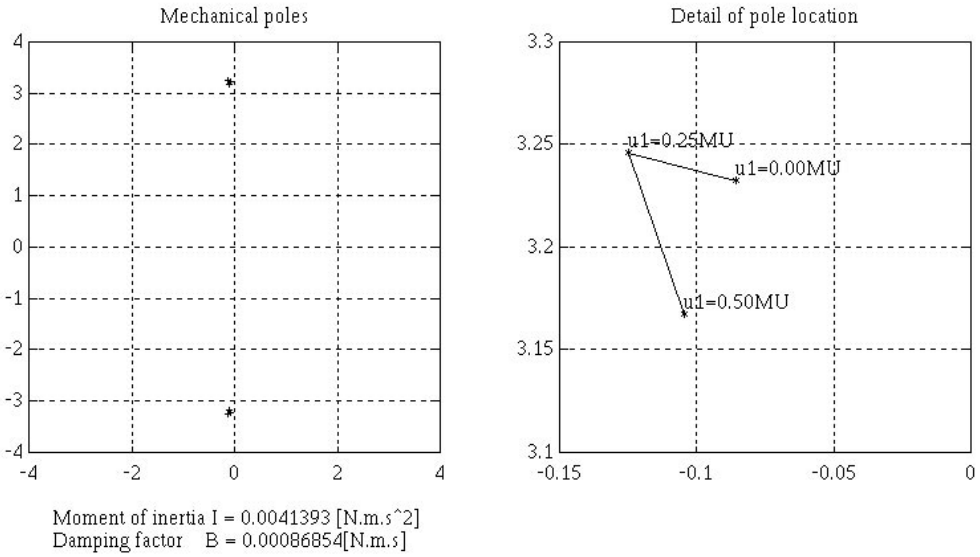


Figure 3.12 Relative dumping in the stabilized azimuth dynamics influenced by the speed of the side propeller

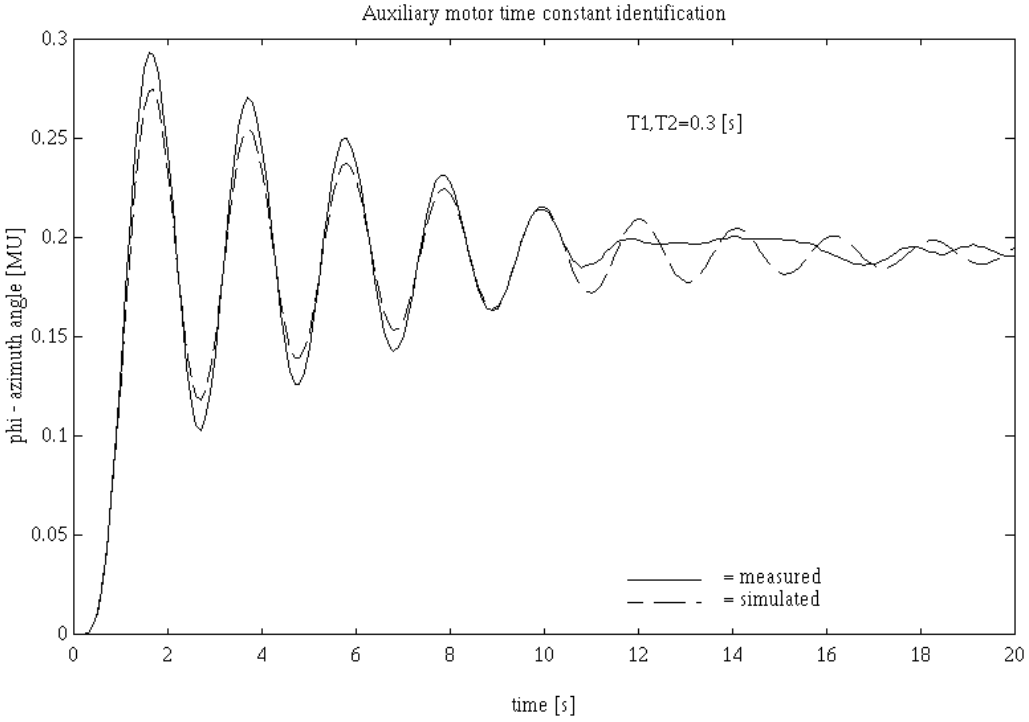


Figure 3.13 Fitting of the model to the side motor and stabilized azimuth response used for the identification of the motor dynamics

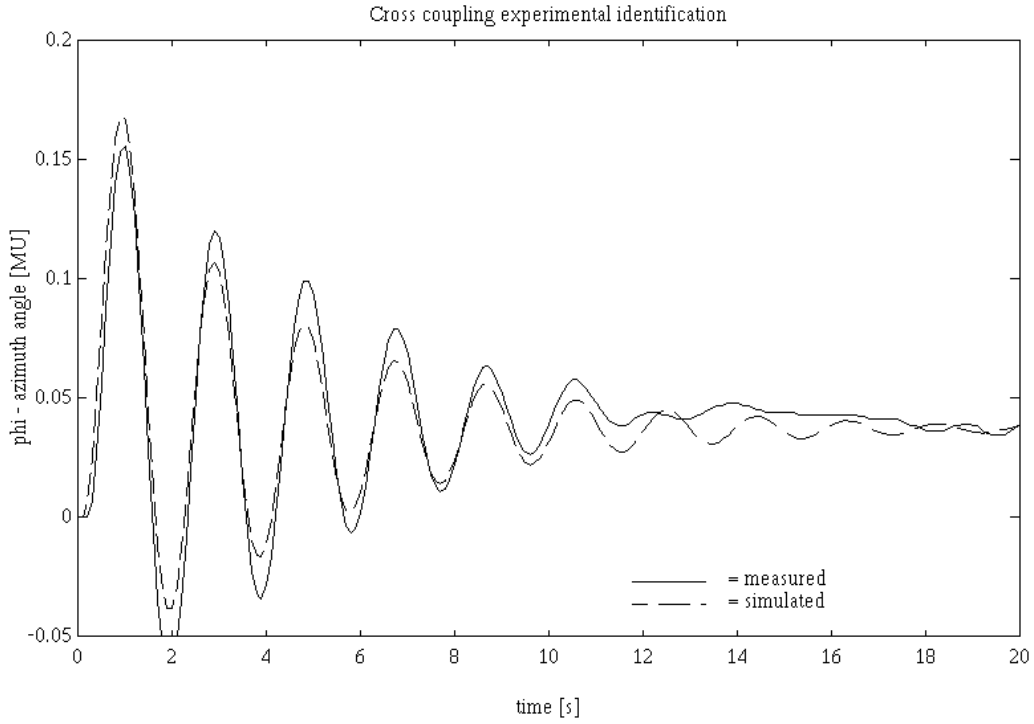


Figure 3.14 Response of the stabilized azimuth to the step in the main motor input - identification of the cross-coupling term