RPO project report

Artyom Voronin

May 24, 2021

Contents

Chapter 1

Review odborného članku

Název článku: Aircraft Control System Using LQG and LQR Controller with Optimal Estimation-Kalman Filter Design
Autoři článku: Labane Chrif, Zemalache Meguenni Kadda
Odkaz na článek: www.sciencedirect.com
Seznam Příloh: aircraft.pdf

1.1 Review

Tato práce se zabývá implementaci řízení letadla pomocí metod **LQG** (Linear–Quadratic–Gaussian control) a **LQR** (Linear–Quadratic regulator) pomocí nástrojů Matlab/Simulink. Kombinace metody řízení LQR a **Kálmánová filtra** umožnuje lepší odhad parametrů, získaných ze sensorů a následovné řízení letadla.

V oblastech letectvy nároky na systémy řízení jsou násobně větší vzhledem k účinnosti a spolehlivostí systémů. S rostoucí požadavky na řízení a autonomnost letadel jedním řešením je zvětšení čísla senzoru a akčních členu. Což má za následek zvětšení ceny výrobku. Nicméně moderny přístupy k řízení systému máji byt schopny pracovat s mnoha vstupní a výstupní parametry. Jedním z řešení je LQG, který je vhodný pro použiti v praktických úlohách, kde systém ovlivněn rušením a šumem měřeni.

Pro řízeni letadla jsou 3 dostupné rotace, které umožnují změnit směr letu letadla. To jsou **Pitch**, **Roll** a **Yaw**. Řízení dále se dělí na podélný směr a boční směr.

Dynamika v podélným směru (longituadinal dynamic) ve stavové representace vypadá nesledující:

$$\begin{bmatrix} \dot{w} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \mathbf{A} \begin{bmatrix} w \\ q \\ \theta \end{bmatrix} + \mathbf{B}[\delta_e] \tag{1.1}$$

$$y = \mathbf{C} \begin{bmatrix} w \\ q \\ \theta \end{bmatrix} + \mathbf{D} \tag{1.2}$$

Matice **A**, **B**, **C**, **D** lze dohledat v članku.

Dynamika v bočním směru (lateral dynamic) ve stavové representace má tvar:

$$\begin{bmatrix}\dot{\beta}\\ \dot{p}\\ \dot{r}\\ \dot{\phi}\end{bmatrix} = \mathbf{A}\begin{bmatrix}\beta\\ p\\ r\\ \phi\end{bmatrix} + \mathbf{B}\begin{bmatrix}\delta_a\\ \delta_r\end{bmatrix} \tag{1.3}$$

$$y = \mathbf{C}\begin{bmatrix}\beta\\ p\\ r\\ \phi\end{bmatrix} + \mathbf{D} \tag{1.4}$$

Matice **A**, **B**, **C**, **D** lze dohledat v članku.

LQG regulátor se skládá z filtru Kalmana a LQR regulátora zařazeních za sebou, jak znázorňuje následující diagram 1.1.

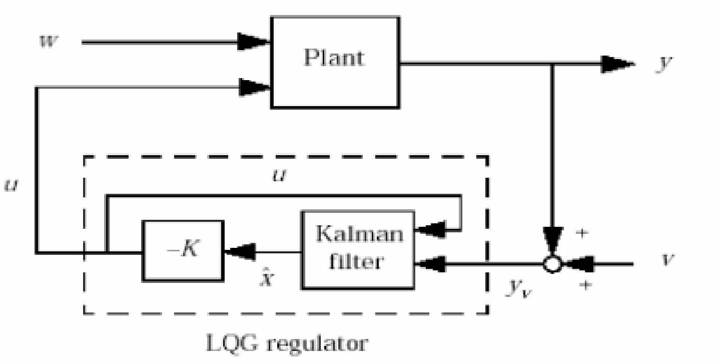

Figure 1.1: Plant model in Simulink.

Pro přenos regulátora platí:

$$\frac{d}{dt}\hat{x} = [A - LC - (B - LD)K]\hat{x} + Ly_v \tag{1.5}$$

$$u = -Kx \tag{1.6}$$

Soustava obsahující rušeni je doplněna o w a v, které představuji bily sum:

$$\dot{x} = Ax + Bu + Gw \tag{1.7}$$

$$y_v = Cx + Du + Hw + v \tag{1.8}$$

Při návrhu LQR regulátoru uživatel na základě zkušeností odhaduje matice Q, N, a R které představuje rychlost, se kterou požadovaný parametr se přibližuje k žádané poloze a také úsilí, které regulátor musí vykonat pro dosazeni této polohy. LQR regulátor pracuje se znalostí celého stavového vektoru. Ale ve většině případu nejsme schopny měřit cely stavový vektor x. Proto je nutné použití Kálmánův filtr, který, pokud systém pozorovatelný a řiditelný, je schopný tento stavový vektor odhadnout.

Při návrhu Kálmánova filtru počítáme že systém obsahuje určitou míru rušení (v případě letadla to může způsobit vítr, nebo změny v hustotě vzduchu, které vyvolá vibrace) a také měření ze sensorů obsahuje šum. Hlavním cílem je navrhnou Kálmánův filtr tak aby odhad parametrů byl co nejpřesnější.

Výsledky simulace znázorněny v článku.

1.2 Krirické zhodnocení

Z uvedených výsledku lze konstatovat že řízení LQG je schopné řídit pitch angle, roll angle a sideslip angle letadla. Přítomnost Kálmánova filtru zaručuje optimální odhad a rekonstrukce parametru stavového vektoru pří výskytů bílého sumu.

Článek vyžaduje určitou míru znalosti v problematice dynamiky letadla. Většina koeficientu v kapitolách 2. a 3. vyžaduje studium dalších zdrojů. Při čtení mě přispěla presentace adl.stanford.edu/ tykající základní informace o dynamice letadla. Nicméně následující kapitoly dostatečně přehledné a informativní. Implementace diskrétního Kalmana vyžaduje od čtenáře znalost použiti přechodů v Simulinku mezi systémem se spojitým časem a diskrétním. Toto lze zrealizovat například blokem Zero-Order-Hold.

Chapter 2

State-space Control (LQG) of Stable Aircraft

2.1 Model Overview

Figure 2.1 represent a general aircraft longitudinal motion. The dynamic model of the aircraft system 2.1 is taken from the article [1].

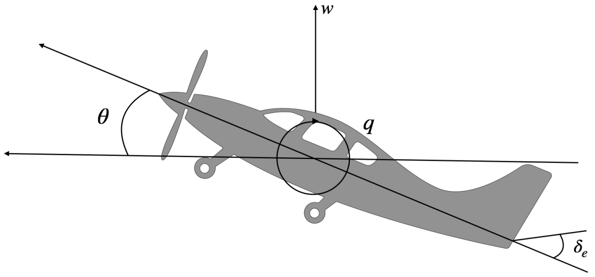

Figure 2.1: Aircraft longitudinal motion

β	rad	Sideslip angle
p	rad/s	Roll rate
r	rad/s	Yaw rate
Φ	rad	Roll angle
w	m/s	Vertical velocity
q	rad/s	Pitch rate
θ	rad	Pitch angle
δ_e	rad	Elevator deflection

Table 2.1: System parameters

For longitudinal motion:

$$\beta = p = r = \Phi = 0 \tag{2.1}$$

The transfer function is represented in state-space form by matrices:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \tag{2.2}$$

$$A = \begin{bmatrix} -0.3149 & 235.8928 & 0 \\ -0.0034 & -0.4282 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} -5.5079 \\ 0.0021 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$$

$$D = 0$$

$$x^T = \begin{bmatrix} w & q & \theta \end{bmatrix}$$

$u = \delta_e$

The implemented Model in Simulink is presented in the following diagram 2.2.

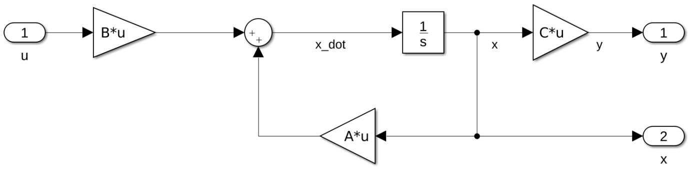

Figure 2.2: Plant model in Simulink.

In the Matlab system can be represented by the state space function:

```
>>sys = ss(A,B,C,D)
```

Figure 2.3 represents the system behavior with non zero initial conditions, where the pitch rate $q_0 = 1$[rad/s].

2.2 Open-loop Response

The system is stable only if all the real parts of the eigenvalues are negative.
The dynamics of the model is determined by the following parameters:

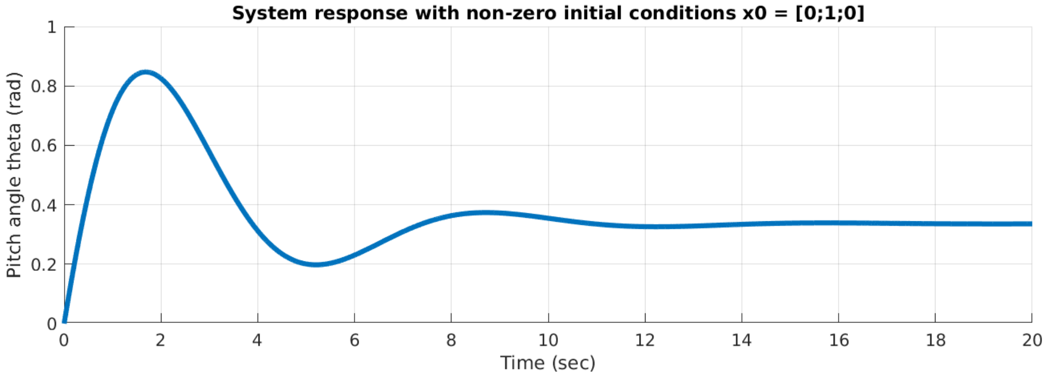

Figure 2.3: System response with non zero initial conditions

Pols:

$$\begin{array}{r} 0.0000 + 0.0000i \\ -0.3715 + 0.8938i \\ -0.3715 - 0.8938i \end{array}$$

The real parts of eigenvalues are negative. The system is **stable**.
Controllability matrix rank: 3
If the rank of Controllability matrix $Co = [B\ AB\ A^2B\ \dots\ A^{n-1}B]$ is equal to n (number of states), the system is **controllable**.
Observability matrix rank: 3
If the rank of Observability matrix $O = [C\ CA\ CA^2\ \dots\ CA^{n-1}]$ is equal to n, the system is **observable**.

The whole system is described by 2.2 equations. Open-loop impulse and step responses are presented in the figure 2.4. The result corresponds to our expectations.

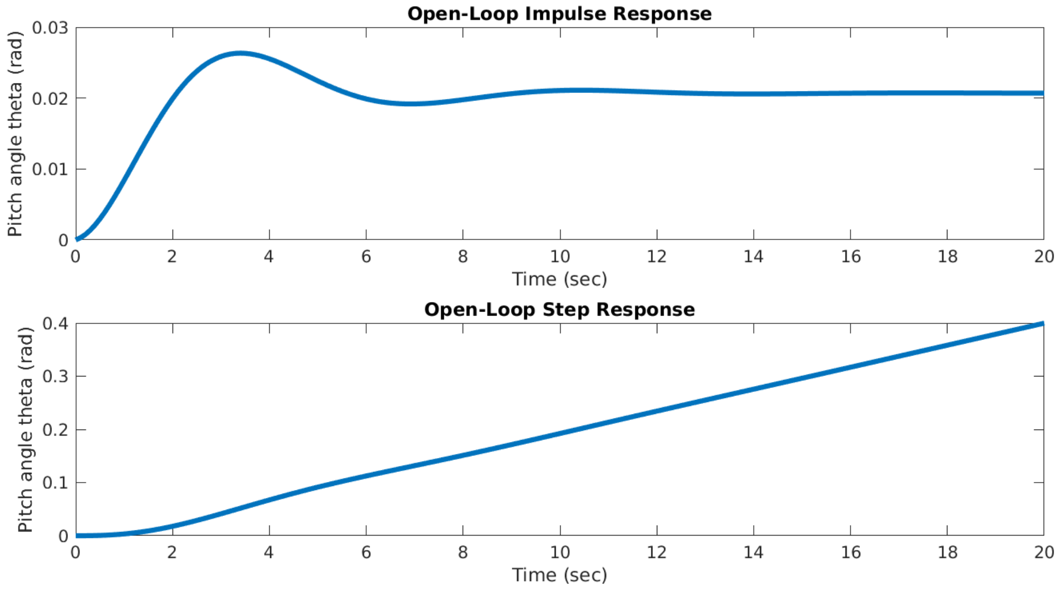

Figure 2.4: Open-loop system response

2.3 LQR Design - zero control

In modeling zero control response, the input of the system will be $u = -Kx$, $(r = 0)$. Substituting into equation 2.2, the result of Close-loop system will correspond to equation 2.3.

$$\dot{x} = (A - BK)x \tag{2.3}$$

For calculation K gain were used LQR method. This method based on quadratic cost function J. Where Q matrix represent how "fast" controller will approximate the following state. R matrix represent how much we depend on energy that we add to system to control.

$$J = \int (x^T Qx + u^T Ru)dt \tag{2.4}$$

In Matlab we define Q and R matrix and use a **lqr** command.

```
Q = diag([0, 0, 500]);
R = .1;
K = lqr(A,B,Q,R);
```

Simulink model of close-loop system is presented in the following diagram 2.6.

Figure 2.6 represent the system behavior controlled by LQR controller with non-zero initial conditions.

2.4 LQR Design - control to the point - rscale

N gain is used to scalar our input for a full-state feedback system to eliminate the steady-state error.

$$u = -Kx + rN \tag{2.5}$$

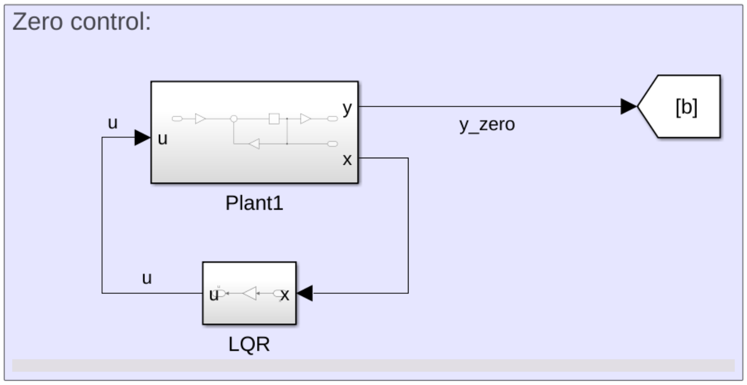

Figure 2.5: Zero control model in Simulink.

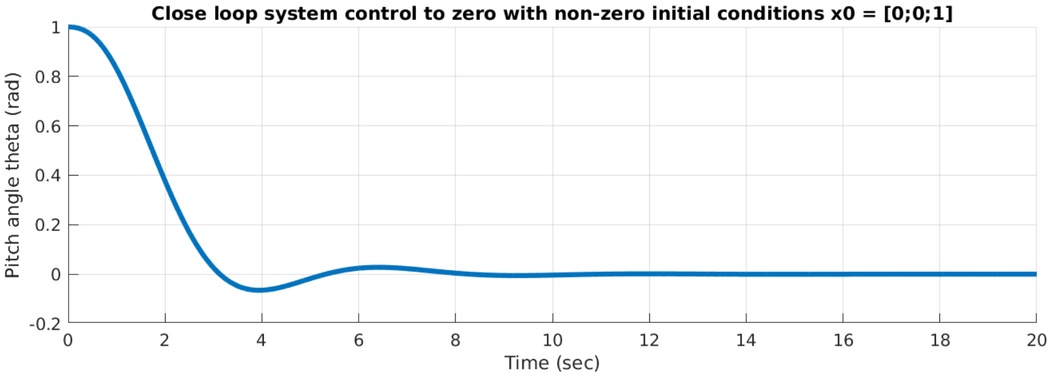

Figure 2.6: Zero control with non-zero initial conditions

where r is final point. Substituting equation 2.5 into equation 2.2, the result of Close-loop system controlled to r will correspond to equation 2.6.

$$\dot{x} = (A - BK)x + BNr \tag{2.6}$$

For calculation **N** gain in Matlab were used *rscale* function from [2].

```
N = rscale(A,B,C,D,K);
```

Simulink model of Close-loop system controlled to point is presented in the following diagram 2.7.

In the Matlab system can be represented by the state space function:

```
>>sys = ss(A–B*K,B*N,C,D)
```

The Close-loop impulse and step response are presented in the following figure 2.8.

2.5 Study of System Behavior

In case Longitudinal dynamic it's possible to measure θ (pitch angle). At this moment, there is no specific conditions for deflector angle actuator. The $R = 0.1$ matrix we will keep equal to 0.1. In the following figure 2.9, there are impulse responses for different Q matrices. Where we change x parameter from 1 to 500 with 50 as step.

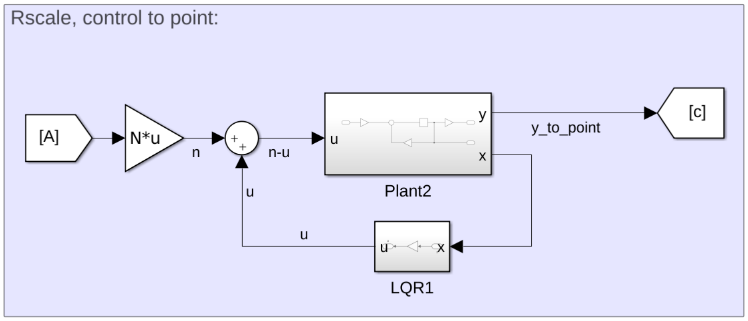

Figure 2.7: Rscale control to point model in Simulink.

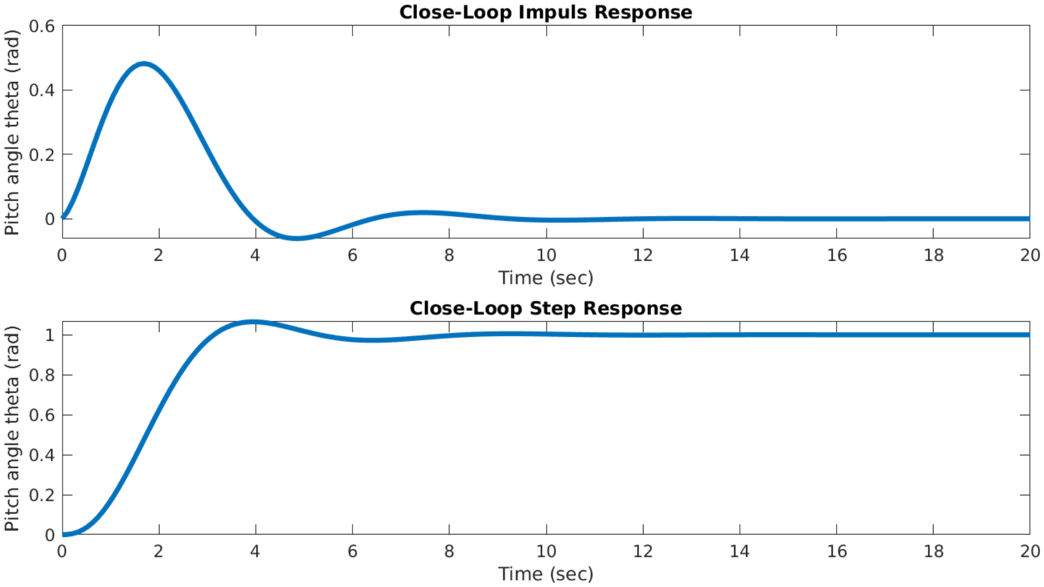

Figure 2.8: Close-loop impulse and step system responses

$$Q = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & x \end{bmatrix}$$

The following Q and R matrices will suffice for our purposes. These values will be used in the following simulations.

$$Q = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 500 \end{bmatrix} \quad R = 0.1$$

2.6 Acting Value Saturation

It's very hard to find some examples of used actuators in aircraft field. In this section different saturation values was studied. However, in this section we will try to see how different saturation values affect the behavior of the system. Assuming that the elevator deflector δ_e possible positions are situated in range from 0.5 to 1.5 [rad]. Figure 2.10 present different system responses for

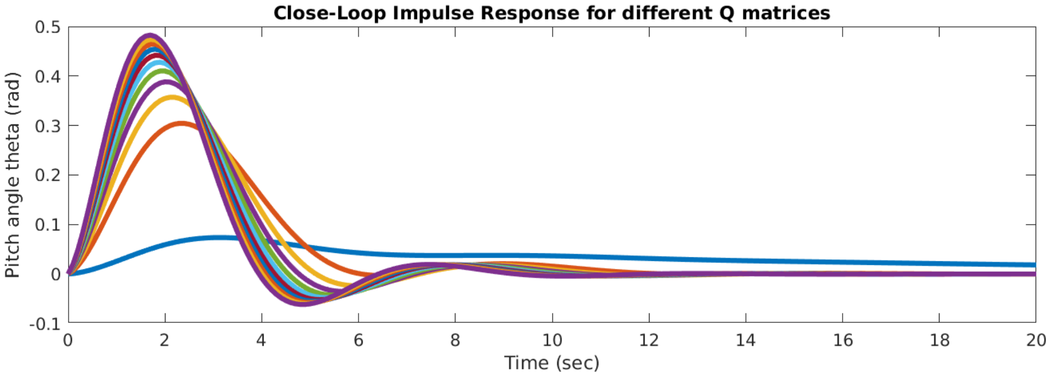

Figure 2.9: Impulse response for different Q matrices.

different saturation values in the range from 0.5 to 1.5 [rad].

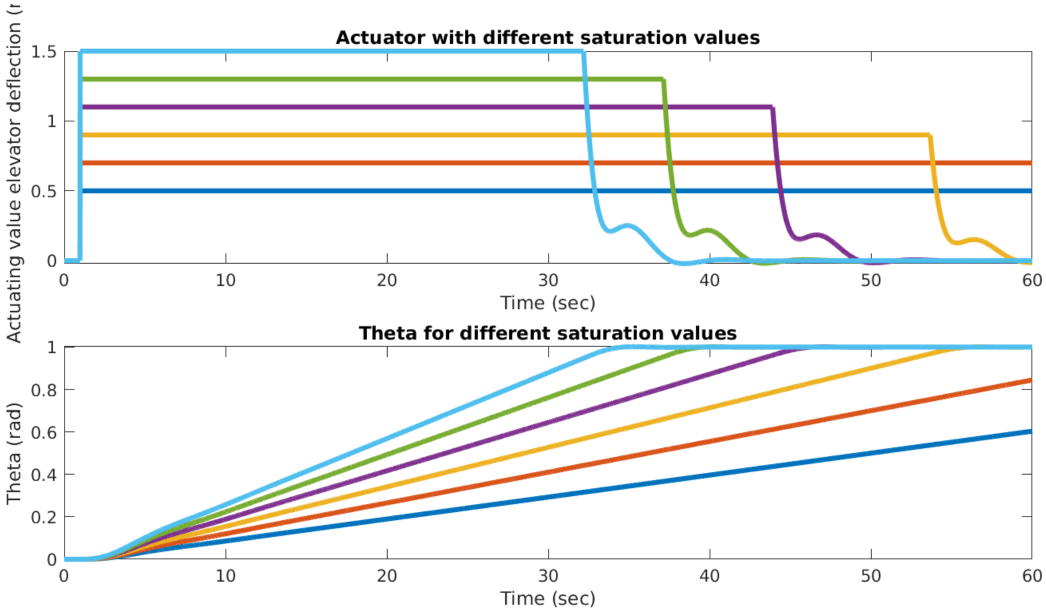

Figure 2.10: System response controlled LQR with different saturation values

At this step we will satisfying with fastest saturation value 1.5rad $\approx 90^o$. The system response with saturation 1.5 rad shown in figure 2.11.

As we can see the aircraft dynamic in longitudinal motion with saturated elevator deflection shows a sufficiently slow behavior, which, however, corresponds to reality.

2.7 Kalman Filter Implementation

The LQR controller can be used if we have information about the whole state x. A Kalman filter can be used to reconstruct the state from the y measurement. The following system of equations

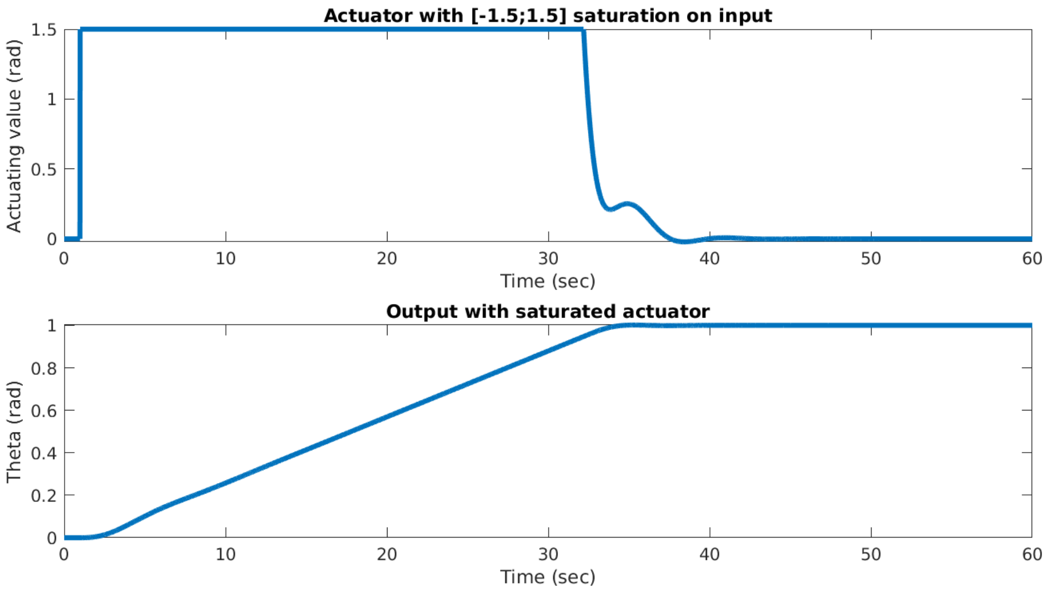

Figure 2.11: System response with input saturation = 1.5 rad

2.7 representing Kalman filter.

$$\begin{aligned} \frac{d}{dt}\hat{x} &= A\hat{x} + Bu + Kf(y - \hat{y}) \\ \hat{y} &= C\hat{x} \end{aligned} \tag{2.7}$$

Simulink model of Kalman filter is presented in the following diagram 2.12.

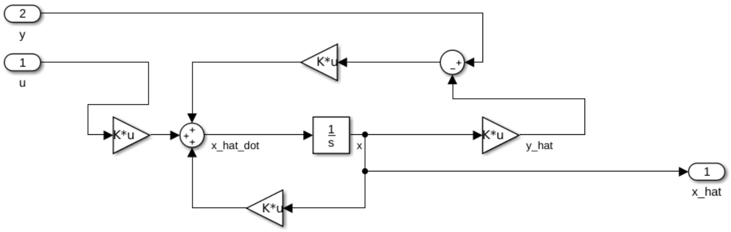

Figure 2.12: Kalman filter implementation in Simulink

Kalman filter use knowledge about disturbance of state and measurement noise magnitudes. Vd matrix representing covariance of state disturbance and Vn matrix is covariance of measurement noise. There are more options how to calculate Kf gain. Define Vd and Vn matrices we can use the same **lqr** function as follow:

```
Sw = .1;
Sv = 1;
[kalmf, Kf, P] = kalman(sys_ss, Sw, Sv);
```

Simulink model of system is presented in the following diagram 2.13.

Implementing Kalman filter in Matlab:

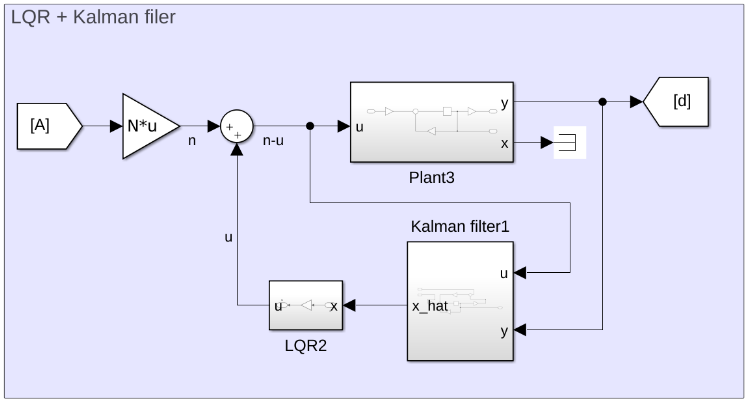

Figure 2.13: Rscale control to point model in Simulink.

```
Akf = A-Kf*C;
Bkf = [B Kf];
Ckf = eye(3);
Dkf = 0*[B Kf];
sys_kf = ss(Akf, Bkf, Ckf, Dkf);
```

2.8 Study of system behavior with an observer (simulation)

The real system control contain measurement noise and disturbance. In model 2.14 were used w_d and w_n disturbance and noise inputs as Gaussian white noise.

The whole system with LQG implementation, disturbances and noise can be describe as following system of equations 2.9.

$$\epsilon = x - \hat{x} \tag{2.8}$$

$$\begin{bmatrix} \dot{x} \\ \dot{\epsilon} \end{bmatrix} = \begin{bmatrix} A - BK & BK \\ 0 & A - KfC \end{bmatrix} \cdot \begin{bmatrix} x \\ \epsilon \end{bmatrix} + \begin{bmatrix} I & 0 \\ I & -Kf \end{bmatrix} \cdot \begin{bmatrix} w_d \\ w_n \end{bmatrix} \tag{2.9}$$

The whole model is controlled by placing eigenvalues in $A - BK$ and $A - KfC$ by K and Kf matrices.

Simulink model is presented in the following diagram 2.14.

The following graph 2.15 shows the correct operation of the LQG controller. Kalman filter correctly estimates the state with permissible noise level.

With increasing noise amplitude, the model is able to remain functional. This can be verified by changing the disturbance and noise magnitude in the Simulink model.

2.9 Evaluation of the whole task and conclusion

The aircraft model used in this task is stable, observable, and controllable. System behavior was studied, and an LQR regulator was added. Using Kalman filter as state observer, LQG control was implemented. Using saturation on the system's input with noisy measurement, LQG control is still correctly working. LQG can be used to control the pitch angle of an aircraft and sufficiently resistant

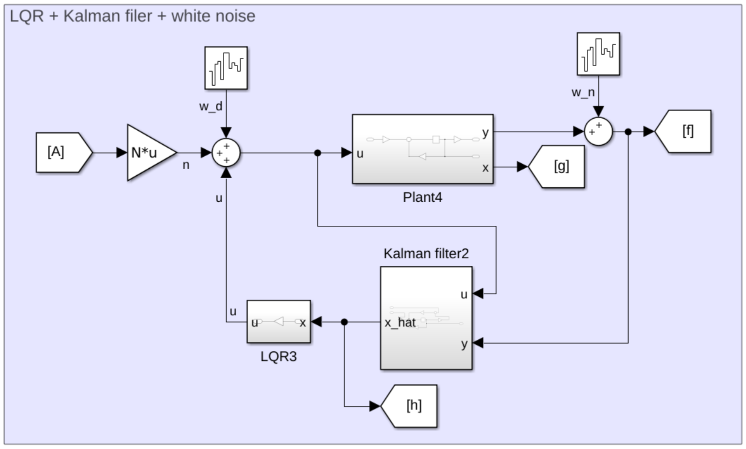

Figure 2.14: LQG simulink model

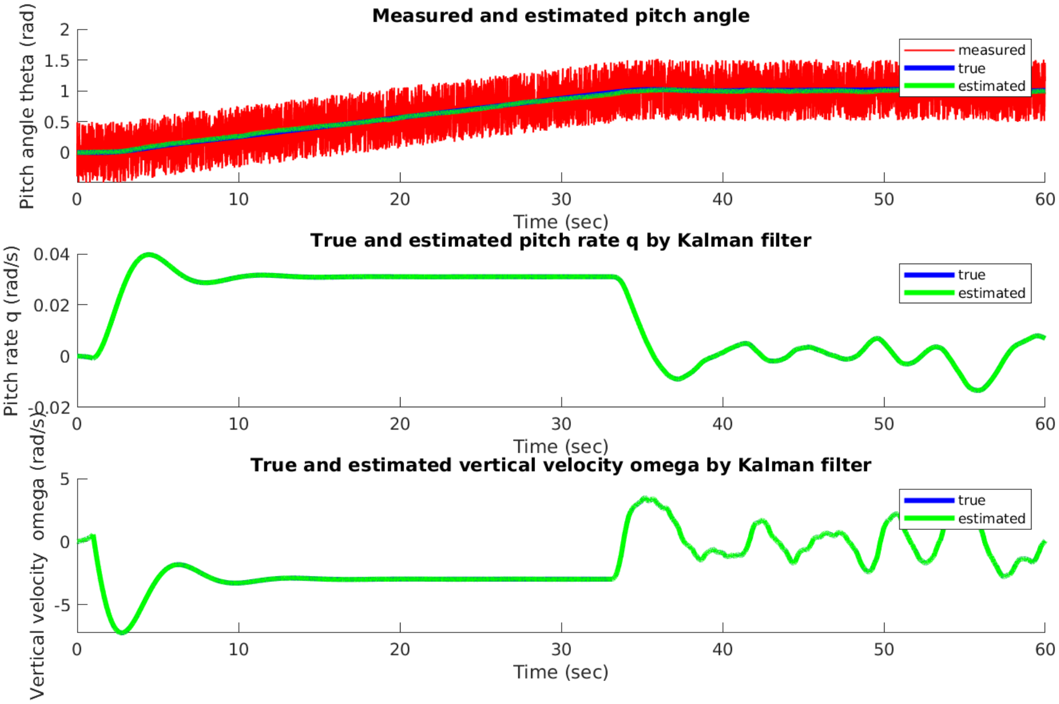

Figure 2.15: LQG regulator output

to the measurement noise. Implementation in the Matlab/Simulink environment sufficiently clear and easily salable.

All Simulink models and Matlab scripts are available in appendix.

Chapter 3

System Identification

System name: Helicopter azimuth

List of files:

data_.mat	measured data
main.m	load parameters, open models, plot input data
model.slx	model
ident_session.sid	model identification session
estimation_session.mat	estimation session

3.1 System analysis and generation I/O data

The following Simulink model was used to measure the data 3.1

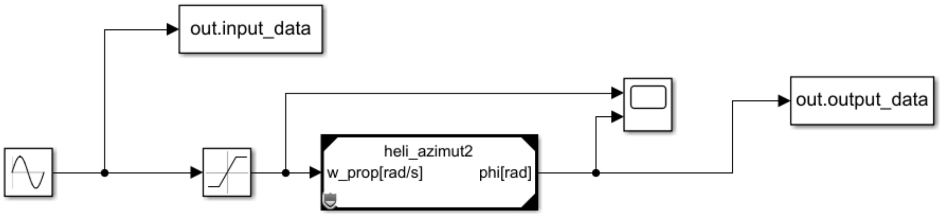

Figure 3.1: Simulink model for data measurement

3 types of input data were used: step, impulse, random numbers responses. 3.2.

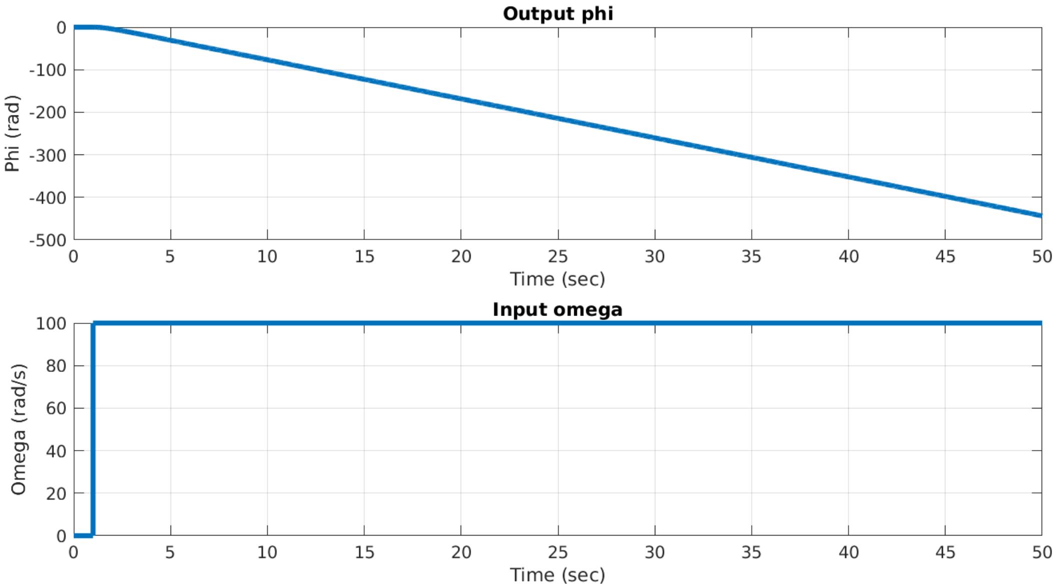

Figure 3.2: Simulink model for data measurement

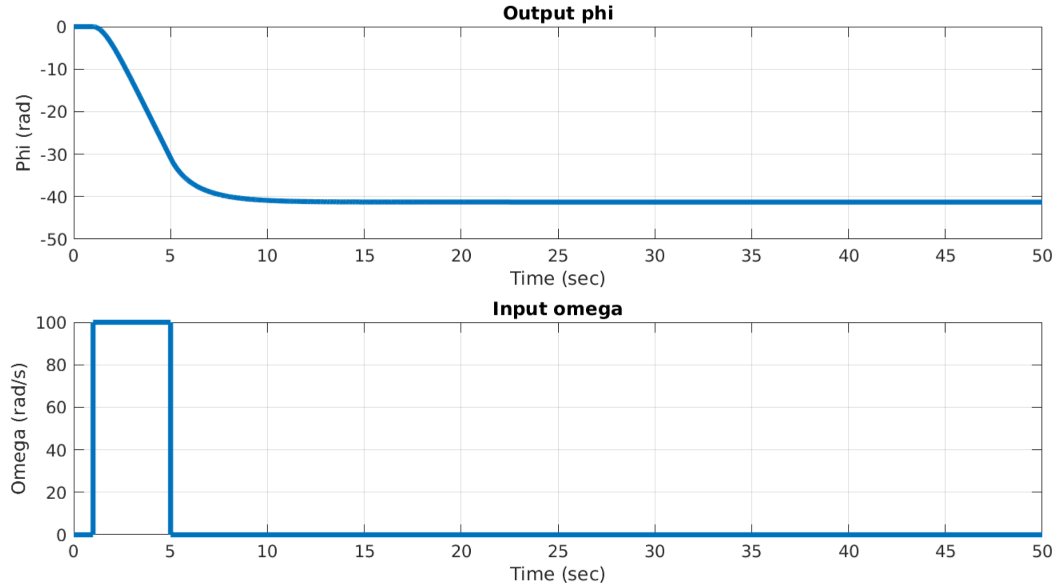

Figure 3.3: Simulink model for data measurement

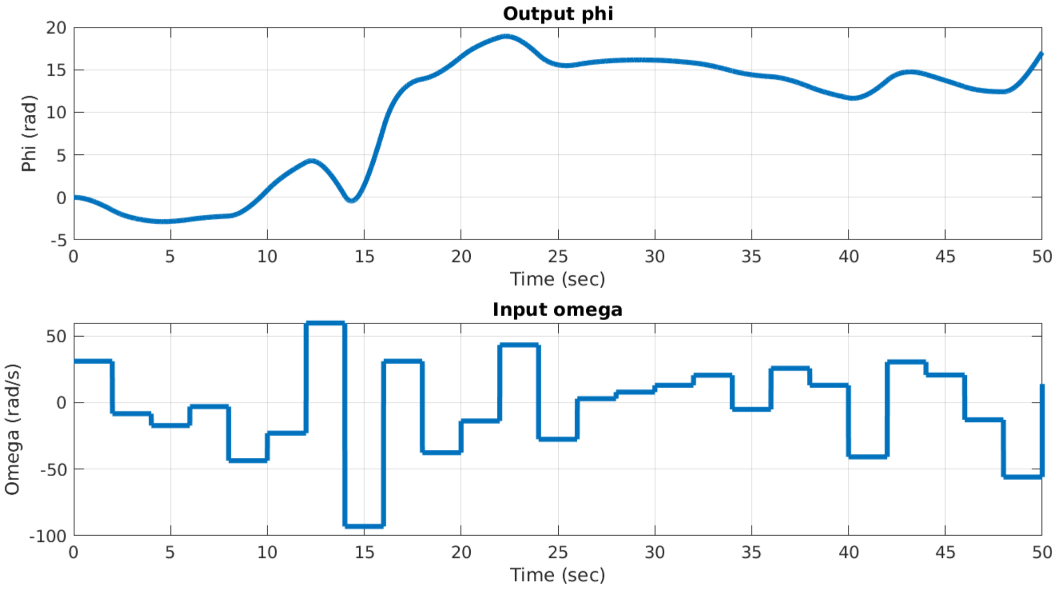

Figure 3.4: Simulink model for data measurement

3.2 Model description

a	Side length of flat plate perpendicular to flow $[m]$
θ	Turn angle $[rad]$
m	Mass $[kg]$
ω	Angular velocity of propeller $[rad/s]$
L	Length $[m]$
I	Moment of inertia $[kg/m^2]$
b	Friction coefficient [-]
$C_x \approx 1.15$	Drag coefficient [-]
$\rho \approx 1.2$	Mass density of the fluid $[kg/m^3]$
A	Propeller area $[m^2]$
$S = a^2$	Area of flat plate $[m^2]$

Math model of helicopter azimuth described by equation 3.2.

$$I\ddot{\theta} = M_{prop} - (M_f + M_d) \tag{3.1}$$

$M_d = \frac{1}{2}C_x\rho S\frac{L}{2}\dot{\theta}^2$ - drag equation. $M_f = b\dot{\theta} + F_c \cdot \text{sign}(\dot{\theta})$ - friction momentum. $M_p = \frac{\rho AL}{4}\omega^2$ - momentum from propeller thrust.

$$I\ddot{\theta} = \frac{\rho AL}{4}\omega^2 - b\dot{\theta} - F_c \cdot \text{sign}(\dot{\theta}) - \frac{1}{2}C_x\rho S\frac{L}{2}\dot{\theta}^2 \tag{3.2}$$

Parameters to estimation: $p_0 = \frac{AL}{I}$, $p_1 = \frac{b}{I}$, $p_2 = \frac{SL}{I}$, $p_3 = \frac{F_c}{I}$
Simulink model represented in following diagram **??**.

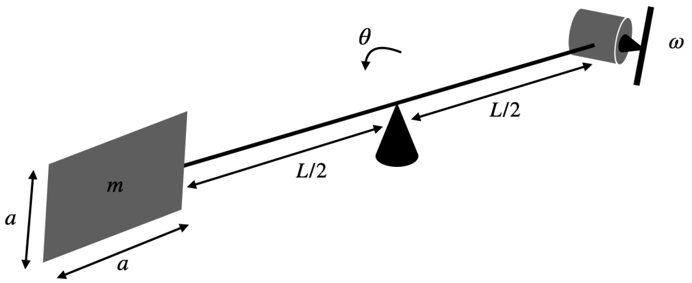

Figure 3.5: Model helicopter azimuth

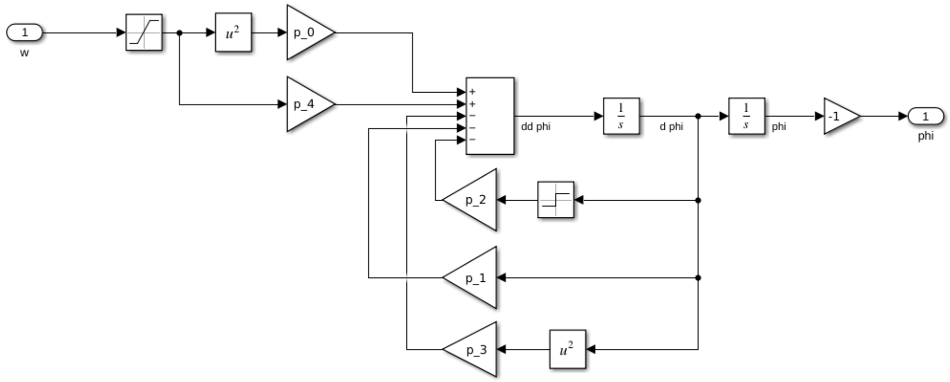

Figure 3.6: Simulink model of helicopter azimuth

3.3 Parameter estimation

The results are saved and can be checked in **estimation_session.mat**. The best courses that were achieved are shown in the graphs 3.7, 3.8 and 3.9.

3.4 Feed-forward

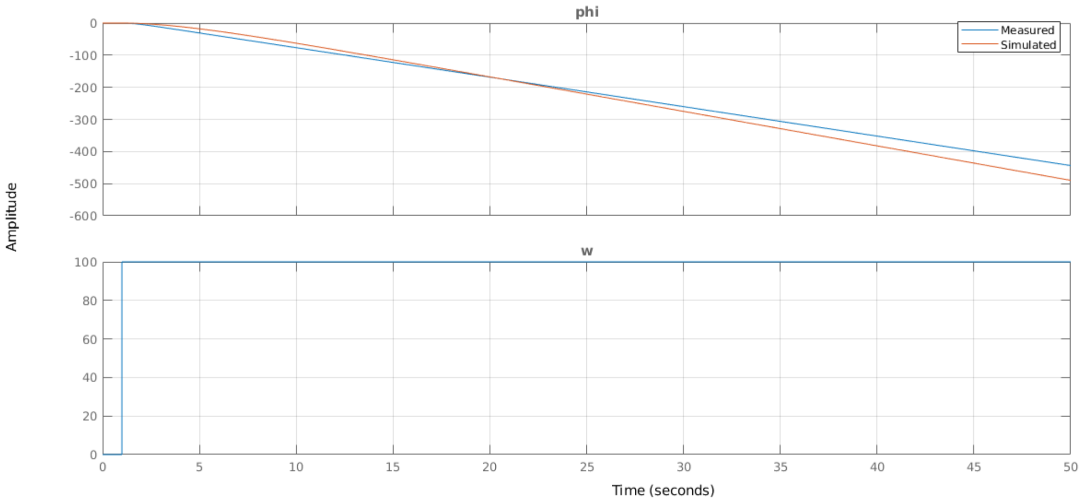

Figure 3.7: Measured and estimated response on step input

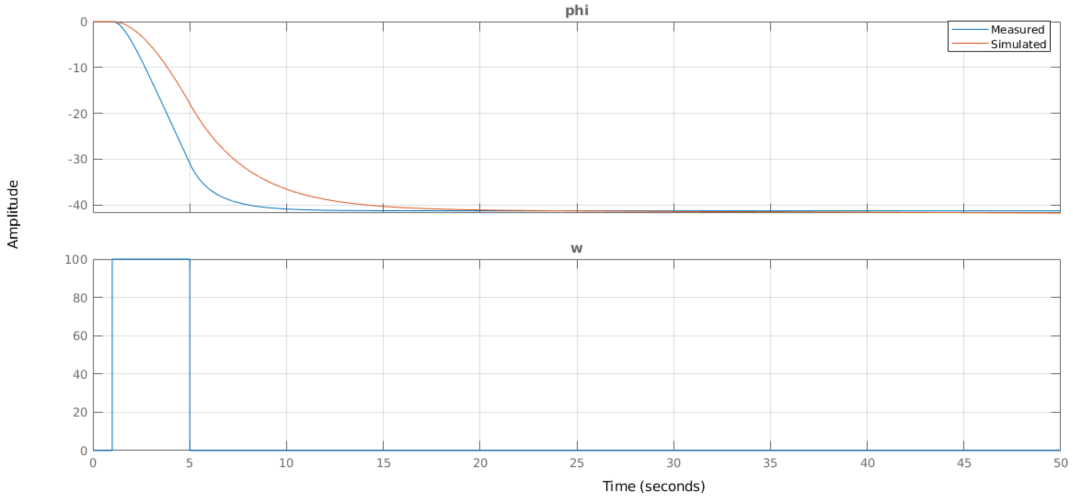

Figure 3.8: Measured and estimated response on impulse input

3.5 System Identification (Black-box)

Black-box model was developed using System Identification toolbox. As working signals, measurements from section was used 3.1.

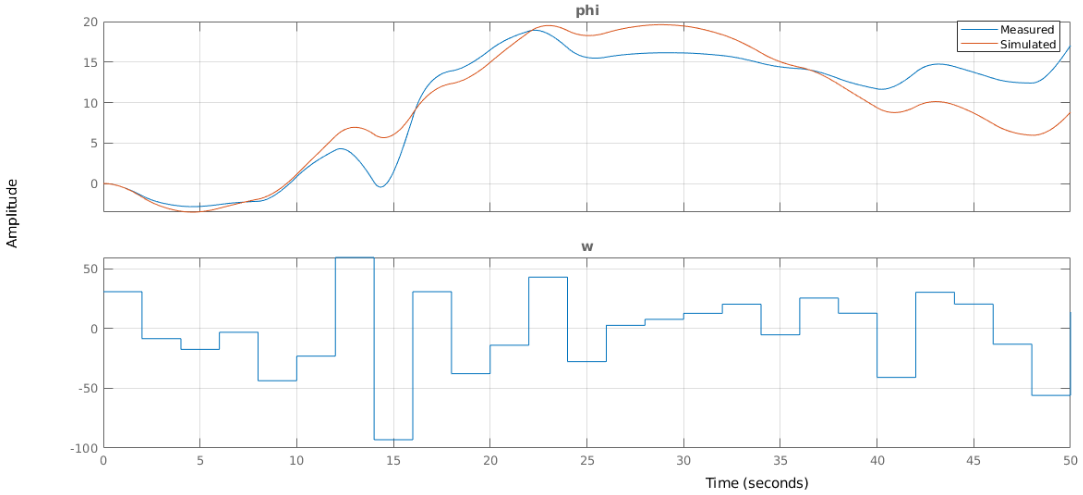

Figure 3.9: Measured and estimated response on random input

3.6 Evaluation of the whole task and conclusion

The gray box model seems correct, but the estimated parameters do not match the reference signals well. Estimation was performed responsibly, but the expected results were not achieved. A possible cause may be some effects that were not taken into account in the model. The Blackbox model similarly corresponds to the reference signals quite inaccurately. A possible reason is the large nonlinearities and effects that were not taken into account in the models.

Chapter 4

HIL

System name: Pendulum with DC Motor **List of files:**

main.m	Parameters initialization
model.slx	Main model

Datasheets:

adxl335.pdf	Accelerometer
lem.pdf	LEM sensor
moc23series.pdf	DC motor
potenc.pdf	Potentiometer

4.1 Selection and description of the controlled system

Modeled system is pendulum driven by dc motor, figure 4.1 schematically shows the system.

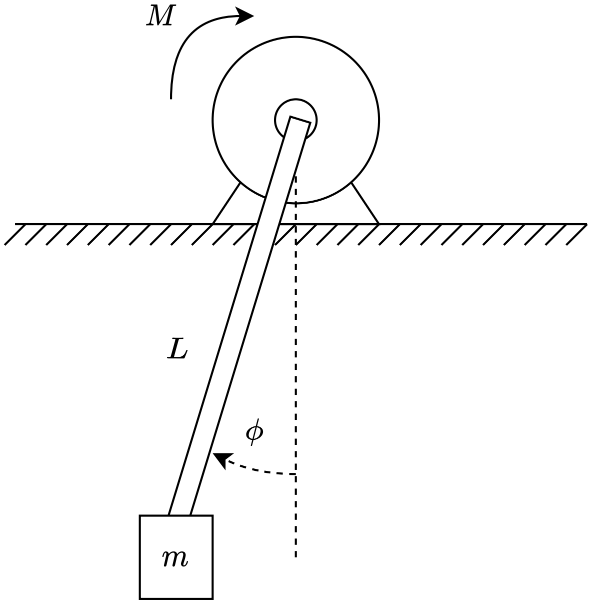

Figure 4.1: Pendulum driven by dc motor

System parameters 4.1.

U	36	[V] Terminal Voltage
R	4	[Ω] Resistance
L	5.5e-3	[H] Inductance
J	1.55e-5	[kg.m3] Inertia
b	9.55e-5	[Nm/(rad/s)] Damping coefficient
K	.03;	[V/(rad/s)] Constant of Proportionality
M	1024;	Encoder
I_max	10	[A] Max current
phi_max	2	[rad] Phi max angle
m	200;	[g] Mass
l	0.1;	[m] Length

Table 4.1: System Parameters

4.2 Model of a controlled system in Simulink

Controlled system is DC motor with digital controlled H-bridge with extinction mechanical pendulum. 4.2

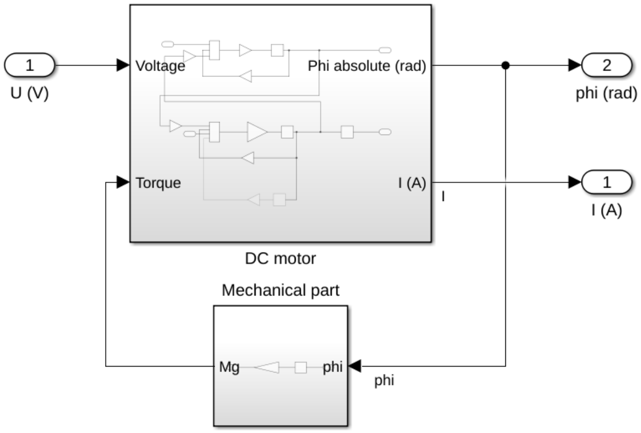

Figure 4.2: Controlled system Simulink model

DC motor with sensors and digital controlled H-bridge is on figure 4.3 4.2

4.3 Models of sensors and actuators

As a sensors were used encoder with 3 channels A,B and Z measure position and velocity of dc motor 4.4. LEM sensor to measure current in dc motor 4.5.

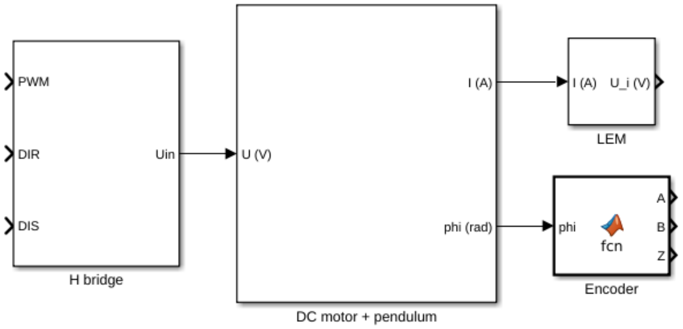

Figure 4.3: Controlled system Simulink model with sensors

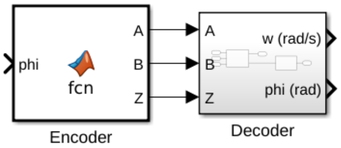

Figure 4.4: Encoder sensor

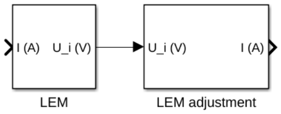

Figure 4.5: LEM sensor

4.4 Control unit and signal adaptation

Control Unit process encoder position and generate PWM using PID regulator; LEM sensor control provide motor current protection4.6.

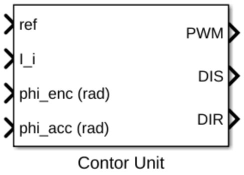

Figure 4.6: Control sub-block

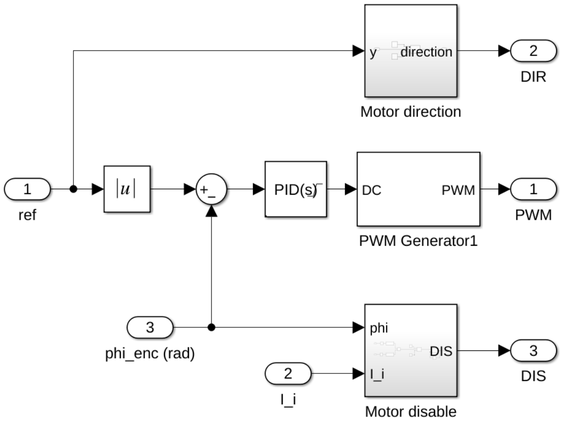

Figure 4.7: Control unit

4.5 Tests

4.5.1 Correct Setup

Correct setup with normal operation conditions. For potentiometer regulation Simulink knob was used 4.8.

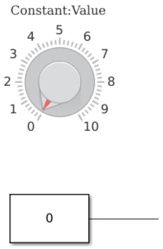

Figure 4.8: Potentiometer regulation knob

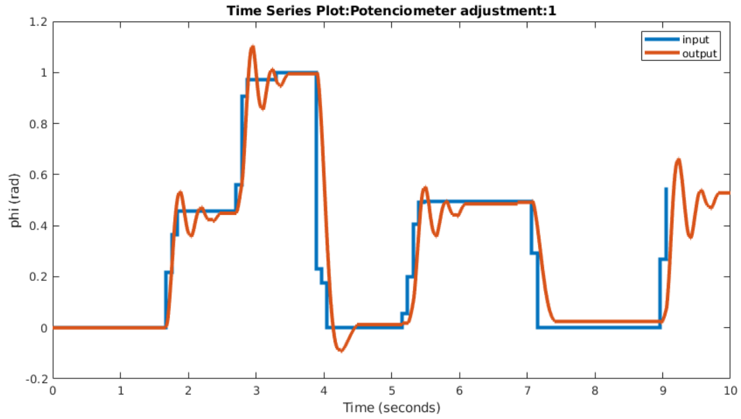

Figure 4.9: Correct setup test

4.5.2 Changing Pendulum Parameters

Pendulum length parameter changing from $l = 0.1$ [m] to $l = 0.2$ [m]. Result system behavior (fig. 4.10.

4.5.3 Encoder Resolution Fault

Encoder fault is given by decreasing encoder resolution from 1024 cycle to 512 cycles 4.11.

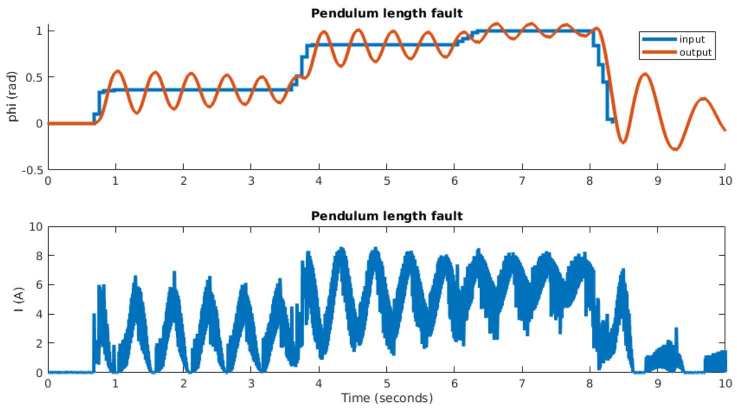

Figure 4.10: Changing parameters test

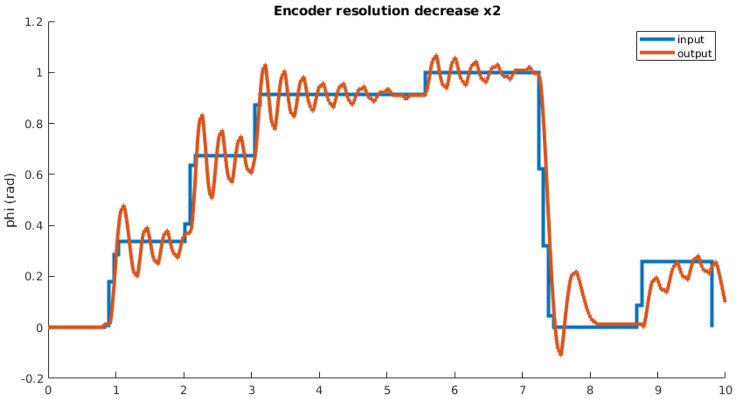

Figure 4.11: Encoder resolution fault

4.5.4 Encoder B channel fault

After encoder channel B was disconnected the system has no longer right position information 4.12

4.5.5 Current Protection lower value

If we change current protection threshold to lower value from 10 to 7 A. The control unit will immediately stop dc motor 4.13.

4.5.6 Noisy Input from Potentiometer

In this situation noisy signal from input potentiometer was modeled **??**.

4.6 Evaluation of the whole task and conclusion

The DC motor with a pendulum was modeled as a controlled system. PID controller was used as a regulator. The plant was extended by sensors and digital control H-bridge. Using the control

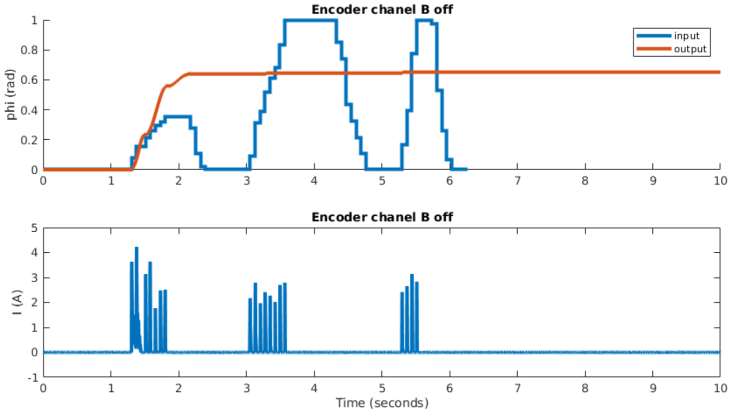

Figure 4.12: Encoder B channel fault

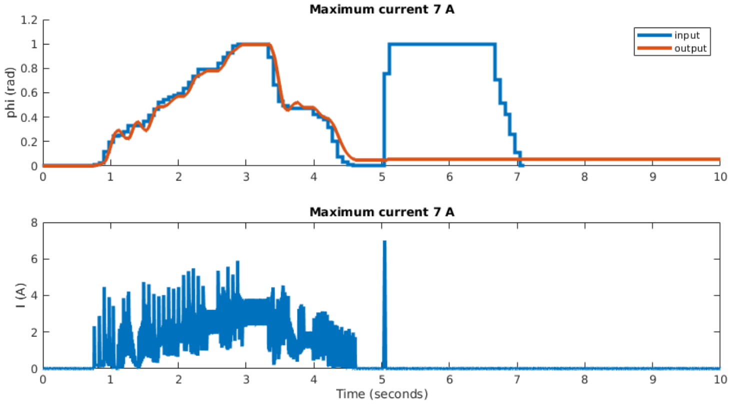

Figure 4.13: Current Protection

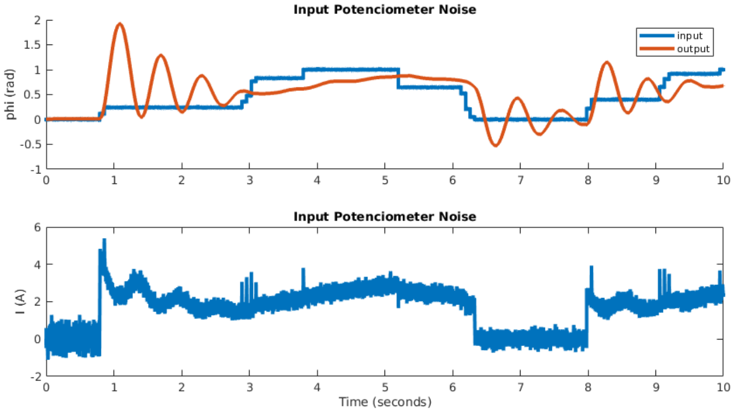

Figure 4.14: Noisy Potentiometer

unit, position regulation by a potentiometer and current protection was implemented and tested as a HIL object. All used parameters were extracted from included datasheets.

Bibliography

[1] Labane Chrif and Zemalache Meguenni Kadda. Aircraft control system using lqg and lqr controller with optimal estimation-kalman filter design. *Procedia Engineering*, 80:245 – 257, 2014. ISSN 1877-7058. doi: https://doi.org/10.1016/j.proeng.2014.09.084. URL `http://www.sciencedirect.com/science/article/pii/S1877705814011771`. 3rd International Symposium on Aircraft Airworthiness (ISAA 2013).

[2] Rscale function. URL `http://ctms.engin.umich.edu/CTMS/index.php?aux=Extras_rscale`.

[3] One wheel. URL `https://onewheel.com/`.