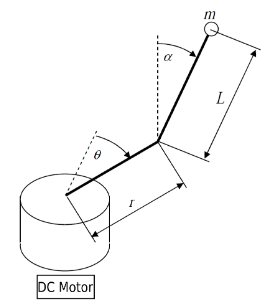
|  |  |
| --- | --- |
| **GDAŃSK UNIVERSITY OF TECHNOLOGY Faculty of Electrical and Control Engineering Department of Control Systems Engineering**  CONTROL AND DECISION SUPPORT SYSTEMS | |
| Laboratory 1 - Multiregional control of inverted pendulum | |
| 25.06.2015 | Patryk Jankowski |

# Introduction

The purpose of the laboratory was to create the linear and nonlinear model of the inverted pendulum and utilise the fuzzy logic to derive a global stabilising controller for this pendulum. Designed system have to guarantee stabilization for largest possible value of the pendulum angle (figure 1).



*Figure 1. The inverted pendulum scheme.*

The dynamics of the plant is given by the following two ODEs

where:

– the angle of the pendulum’s arm,

– the angle of the pendulum,

– the length of the pendulum from the rotation axis to the centre of mass,

– the pendulum mass,

– the arm’s length,

– the equivalent moment of inertia about motor shaft pivot axis,

– the pendulum moment of inertia about its pivot axis,

– the pendulum viscous damping,

– acceleration due to gravity.

The pendulum’s arm is mounted on the rotor of DC motor

where:

– the motor momentum,

– the input voltage,

– the angle of the pendulum’s arm,

– denotes the motor back-emf constant,

– the motor torque constant,

– the PWM amplifier gain,

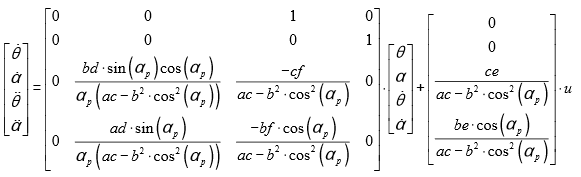
– the motor armature resistance.

# Exercise 1

The first task was to derive a nonlinear global fuzzy model of an inverted pendulum based on Takagi-Sugeno reasoning scheme including dynamics and constraints resulting from actuation system under following assumptions:

* the state vector is to be of the form ,
* the coordinate system is to be oriented in a manner guaranteeing to describe the pendulums unstable equilibrium point,
* utilise triangle shaped membership functions,
* consider operating points , , .

The linearized model in the work point is shown below



where:

The workspace of the pendulum was divided into eight parts for eight work points (figure2). There are following points:

,,,

, , ,

, , .

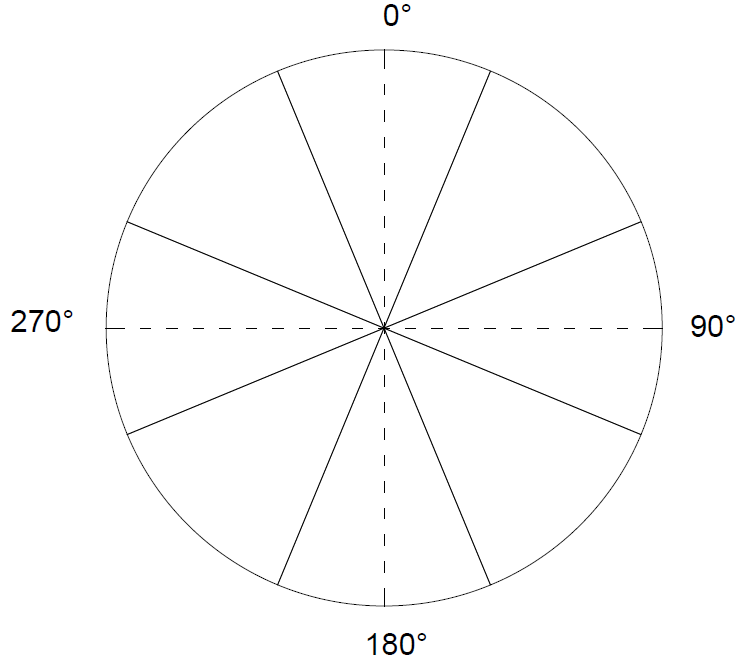


Figure 2. Pendulum workspace divided into eight parts.

The following points were not used: , , , . There is an in the denominator in the matrix and numerator cannot be divided by 0. Moreover, these values are uncontrollable and use of these points is not a good choice. After changing value for example from alfa\_deg3 = 89 to alfa\_deg3 = 90 in the program code which is shown in the listing 1 there is the error that *lqr* command cannot stabilize the plant (listing 2) and is not possible to use values which were given in the task. However, *lqr* method will be used in the exercise 2 and this function is not used in exercise 1.

% Pendulum

r = 0.0826;

m = 0.027;

g = 9.81;

B\_eq = 0;

l\_p = 0.191;

J\_p = 1.20e-4;

% Engine

R\_m = 3.3;

K\_t = 0.028;

K\_m = K\_t;

J\_eq = 1.23e-4;

K\_w = 2.3;

% Model

a = J\_eq + m\*r^2;

b = m\*l\_p\*r;

c = m\*l\_p^2 + J\_p;

d = m\*g\*l\_p;

e = K\_m\*K\_w/R\_m;

f = (K\_m\*K\_t + B\_eq\*R\_m)/R\_m;

% Work point 1 ----------------------------------------

alfa\_deg = 0.0001;

alfa\_rad = pi\*alfa\_deg/180;

h = a\*c-(b\*cos(alfa\_rad))^2;

i = b\*d\*sin(alfa\_rad)\*cos(alfa\_rad)/(alfa\_rad\*h);

j = -c\*f/h;

k = a\*d\*sin(alfa\_rad)/(alfa\_rad\*h);

l = -b\*f\*cos(alfa\_rad)/h;

n = c\*e/h;

o = b\*e\*cos(alfa\_rad)/h;

% Space state model

A1 = [0 0 1 0; 0 0 0 1; 0 i j 0; 0 k l 0];

B1 = [0; 0; n; o];

C = [0 1 0 0];

D = 0;

M1 = [B1 A1\*B1 A1^2\*B1 A1^3\*B1];

% Work point 2 ----------------------------------------

alfa\_deg2 = 45;

alfa\_rad2 = pi\*alfa\_deg2/180;

h2 = a\*c-(b\*cos(alfa\_rad2))^2;

i2 = b\*d\*sin(alfa\_rad2)\*cos(alfa\_rad2)/(alfa\_rad2\*h2);

j2 = -c\*f/h2;

k2 = a\*d\*sin(alfa\_rad2)/(alfa\_rad2\*h2);

l2 = -b\*f\*cos(alfa\_rad2)/h2;

n2 = c\*e/h2;

o2 = b\*e\*cos(alfa\_rad2)/h2;

% Space state model

A2 = [0 0 1 0; 0 0 0 1; 0 i2 j2 0; 0 k2 l2 0];

B2 = [0; 0; n2; o2];

M2 = [B2 A2\*B2 A2^2\*B2 A2^3\*B2];

% Work point 3 ----------------------------------------

alfa\_deg3 = 89;

alfa\_rad3 = pi\*alfa\_deg3/180;

h3 = a\*c-(b\*cos(alfa\_rad3))^2;

i3 = b\*d\*sin(alfa\_rad3)\*cos(alfa\_rad3)/(alfa\_rad3\*h3);

j3 = -c\*f/h3;

k3 = a\*d\*sin(alfa\_rad3)/(alfa\_rad3\*h3);

l3 = -b\*f\*cos(alfa\_rad3)/h3;

n3 = c\*e/h3;

o3 = b\*e\*cos(alfa\_rad3)/h3;

% Space state model

A3 = [0 0 1 0; 0 0 0 1; 0 i3 j3 0; 0 k3 l3 0];

B3 = [0; 0; n3; o3];

M3 = [B3 A3\*B3 A3^2\*B3 A3^3\*B3];

% Work point 4 ----------------------------------------

alfa\_deg4 = 135;

alfa\_rad4 = pi\*alfa\_deg4/180;

h4 = a\*c-(b\*cos(alfa\_rad4))^2;

i4 = b\*d\*sin(alfa\_rad4)\*cos(alfa\_rad4)/(alfa\_rad4\*h4);

j4 = -c\*f/h4;

k4 = a\*d\*sin(alfa\_rad4)/(alfa\_rad4\*h4);

l4 = -b\*f\*cos(alfa\_rad4)/h4;

n4 = c\*e/h4;

o4 = b\*e\*cos(alfa\_rad4)/h4;

% Work point 5 ----------------------------------------

A4 = [0 0 1 0; 0 0 0 1; 0 i4 j4 0; 0 k4 l4 0];

B4 = [0; 0; n4; o4];

M4 = [B4 A4\*B4 A4^2\*B4 A4^3\*B4];

% Work point 6 ----------------------------------------

alfa\_deg5 = 179;

alfa\_rad5 = pi\*alfa\_deg5/180;

h5 = a\*c-(b\*cos(alfa\_rad5))^2;

i5 = b\*d\*sin(alfa\_rad5)\*cos(alfa\_rad5)/(alfa\_rad5\*h5);

j5 = -c\*f/h5;

k5 = a\*d\*sin(alfa\_rad5)/(alfa\_rad5\*h5);

l5 = -b\*f\*cos(alfa\_rad5)/h5;

n5 = c\*e/h5;

o5 = b\*e\*cos(alfa\_rad5)/h5;

% Space state model

A5 = [0 0 1 0; 0 0 0 1; 0 i5 j5 0; 0 k5 l5 0];

B5 = [0; 0; n5; o5];

M5 = [B5 A5\*B5 A5^2\*B5 A5^3\*B5];

% Work point 7 ----------------------------------------

alfa\_deg6 = 225;

alfa\_rad6 = pi\*alfa\_deg6/180;

h6 = a\*c-(b\*cos(alfa\_rad6))^2;

i6 = b\*d\*sin(alfa\_rad6)\*cos(alfa\_rad6)/(alfa\_rad6\*h6);

j6 = -c\*f/h6;

k6 = a\*d\*sin(alfa\_rad6)/(alfa\_rad6\*h6);

l6 = -b\*f\*cos(alfa\_rad6)/h6;

n6 = c\*e/h6;

o6 = b\*e\*cos(alfa\_rad6)/h6;

% Space state model

A6 = [0 0 1 0; 0 0 0 1; 0 i6 j6 0; 0 k6 l6 0];

B6 = [0; 0; n6; o6];

M6 = [B6 A6\*B6 A6^2\*B6 A6^3\*B6];

% Work point 7 ----------------------------------------

alfa\_deg7 = 269;

alfa\_rad7 = pi\*alfa\_deg7/180;

h7 = a\*c-(b\*cos(alfa\_rad7))^2;

i7 = b\*d\*sin(alfa\_rad7)\*cos(alfa\_rad7)/(alfa\_rad7\*h7);

j7 = -c\*f/h7;

k7 = a\*d\*sin(alfa\_rad7)/(alfa\_rad7\*h7);

l7 = -b\*f\*cos(alfa\_rad7)/h7;

n7 = c\*e/h7;

o7 = b\*e\*cos(alfa\_rad7)/h7;

% Space state model

A7 = [0 0 1 0; 0 0 0 1; 0 i7 j7 0; 0 k7 l7 0];

B7 = [0; 0; n7; o7];

M7 = [B7 A7\*B7 A7^2\*B7 A7^3\*B7];

% Work point 8 ----------------------------------------

alfa\_deg8 = 315;

alfa\_rad8 = pi\*alfa\_deg8/180;

h8 = a\*c-(b\*cos(alfa\_rad8))^2;

i8 = b\*d\*sin(alfa\_rad8)\*cos(alfa\_rad8)/(alfa\_rad8\*h8);

j8 = -c\*f/h8;

k8 = a\*d\*sin(alfa\_rad8)/(alfa\_rad8\*h8);

l8 = -b\*f\*cos(alfa\_rad8)/h8;

n8 = c\*e/h8;

o8 = b\*e\*cos(alfa\_rad8)/h8;

% Space state model

A8 = [0 0 1 0; 0 0 0 1; 0 i8 j8 0; 0 k8 l8 0];

B8 = [0; 0; n8; o8];

M8 = [B8 A8\*B8 A8^2\*B8 A8^3\*B8];

% Work point 9 ----------------------------------------

alfa\_deg9 = 359;

alfa\_rad9 = pi\*alfa\_deg9/180;

h9 = a\*c-(b\*cos(alfa\_rad9))^2;

i9 = b\*d\*sin(alfa\_rad9)\*cos(alfa\_rad9)/(alfa\_rad9\*h9);

j9 = -c\*f/h9;

k9 = a\*d\*sin(alfa\_rad9)/(alfa\_rad9\*h9);

l9 = -b\*f\*cos(alfa\_rad9)/h9;

n9 = c\*e/h9;

o9 = b\*e\*cos(alfa\_rad9)/h9;

% Space state model

A9 = [0 0 1 0; 0 0 0 1; 0 i9 j9 0; 0 k9 l9 0];

B9 = [0; 0; n9; o9];

M9 = [B9 A9\*B9 A9^2\*B9 A9^3\*B9];

% LQR -----------------------------------------------

Q = [0.00011 0 0 0; 0 0.0035 0 0; 0 0 0.0000222 0; 0 0 0 0.000335];

R = [0.000133];

N = 0;

F1 = lqr(A1,B1,Q,R,N); F2 = lqr(A2,B2,Q,R,N); F3 = lqr(A3,B3,Q,R,N);

F4 = lqr(A4,B4,Q,R,N); F5 = lqr(A5,B5,Q,R,N); F6 = lqr(A6,B6,Q,R,N);

F7 = lqr(A7,B7,Q,R,N); F8 = lqr(A8,B8,Q,R,N); F9 = lqr(A9,B9,Q,R,N);

initial = pi/2

*Listing 1. The program code.*

Error using lqr (line 44)

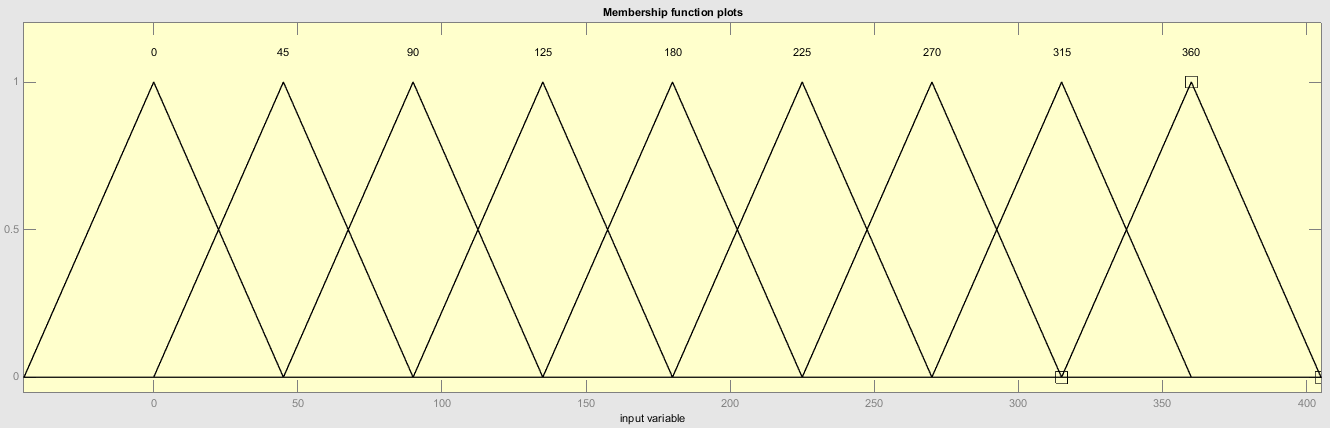
The "lqr" command failed to stabilize the plant or find an optimal feedback gain. To remedy this

problem:

1. Make sure that all unstable poles of A are controllable through B (use MINREAL to check)

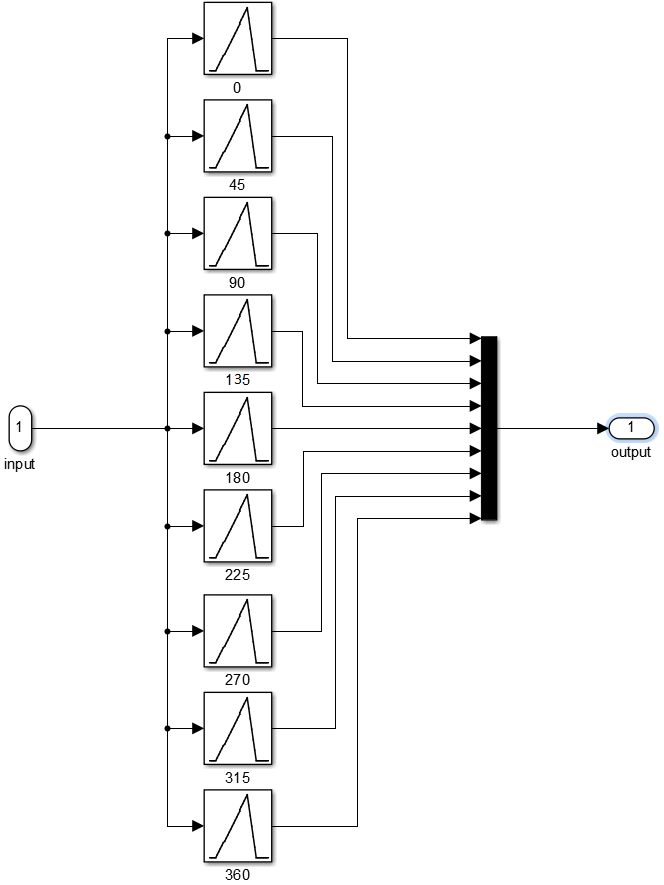
2. Modify the weights Q and R to make [Q N;N' R] positive definite (use EIG to check positivity).

*Listing 2. The lqr error after changing the work point value.*

To create the fuzzy model was used the triangle shaped membership functions which is shown in the figure 3.

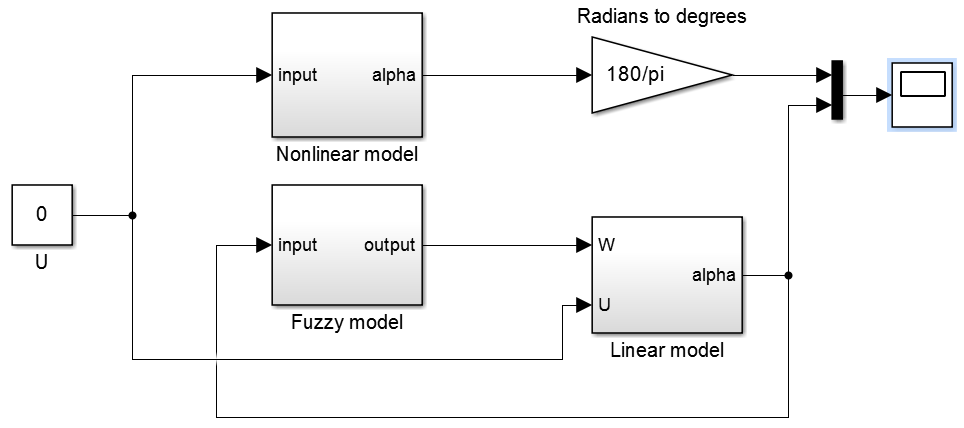
*Figure 3. Triangle shaped membership functions.*

There are nine triangles in the figure 3, where the tops of the triangles are in every work point. The circle was divided into 8 parts but there are nine triangles because it is one of them in *0* and another in *360*. Moreover in the fuzzy membership functions the triangles overlap. The fuzzy model in Simulink is shown in the figure 4.



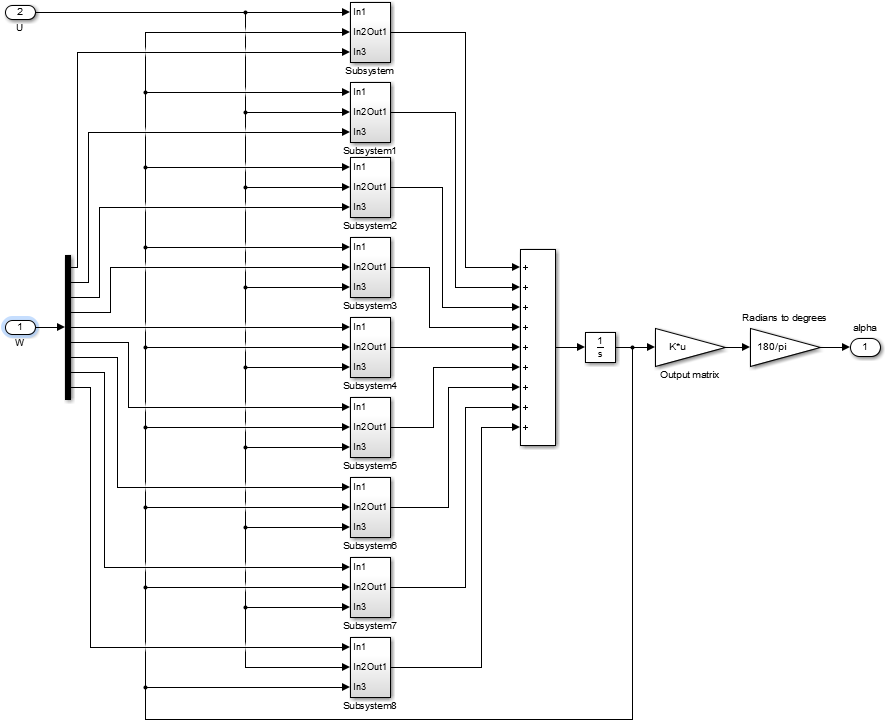
*Figure 4. The fuzzy model in Simulink.*

The model of the system which include linear model, nonlinear model and fuzzy model is shown in the picture 5. There are also *Radians to degrees* which convert block and oscilloscope.

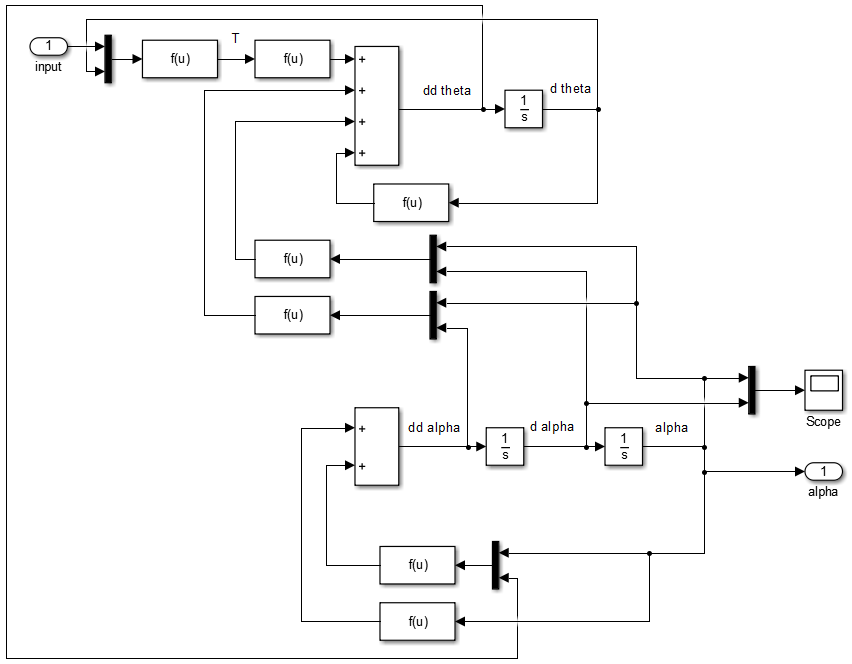
**

*Figure 5. The model of the system in Simulink.*

The linear model and nonlinear model are shown in the figure 6 and in the figure 7.

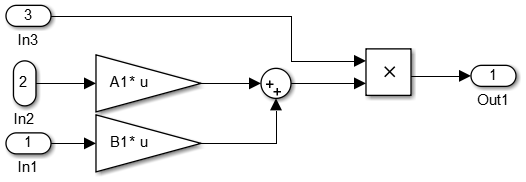


*Figure 6. The linear model in Simulink.*



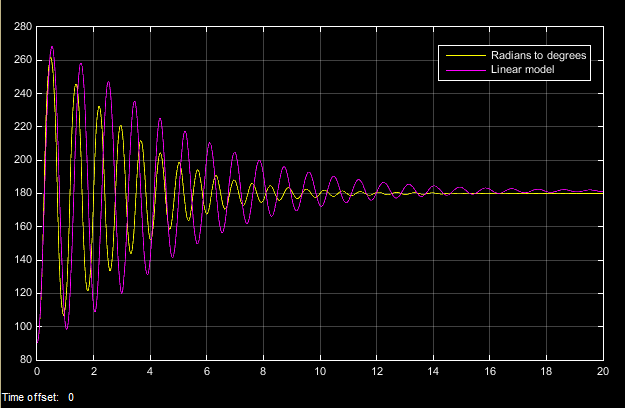
*Figure 7. The nonlinear model in Simulink.*

The example of the subsystem from figure 6 is shown in the figure 8 and the values in the functions and from figure 7 are in the program code from listing 1.

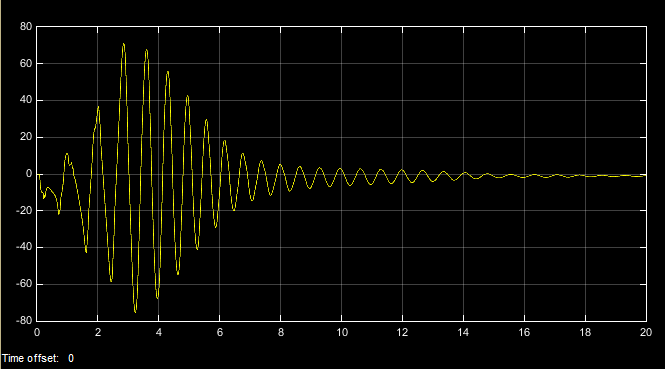


*Figure 8. The subsystem of the linear model, where are A and B matrices.*

At the end there are comparison of the responses of the linear and nonlinear models in the figure 9 for initial point which was equal and in the figure 10 there is shown the difference (error) between response of linear model and response of nonlinear model.

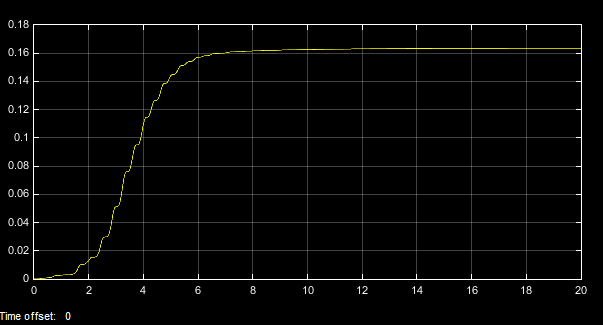
**

*Figure 9. The responses of the linear and nonlinear systems.*

**

*Figure 10. Difference between response of linear model and response of nonlinear model.*

The difference might seem to be quite large between seconds 1 and 7 but there is something like a phase shift between both of the charts. The mean squared error (MSE) is equal about 16% and the chart of this error is shown in the figure 11.

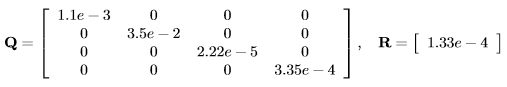


*Figure 11. The chart of the mean squared error (MSE).*

# Exercise 2

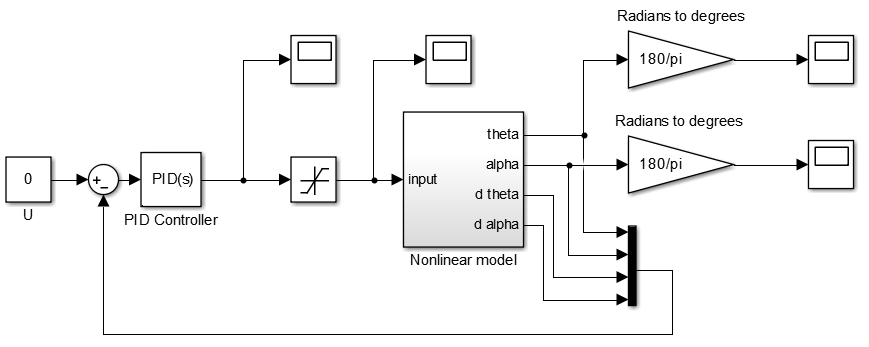
The purpose of this task was to design a multiregional T-S fuzzy controller to stabilise the pendulum in its unstable equilibrium point. The initial deviation of the pendulum form the origin (initial condition in dimension) should be maximized while maintaining good performance of the closed loop system. Moreover, there should be included (as much as possible) in the design procedure the constraints on the arm angle movement so that it (preferably) remains in smallest possible bounds. In the design should be also utilise LQRs as local controllers.

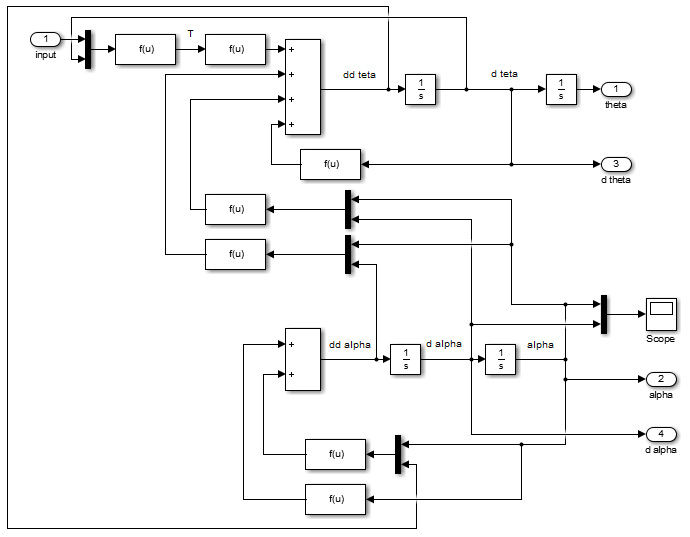
There will be considered the following weight matrices:



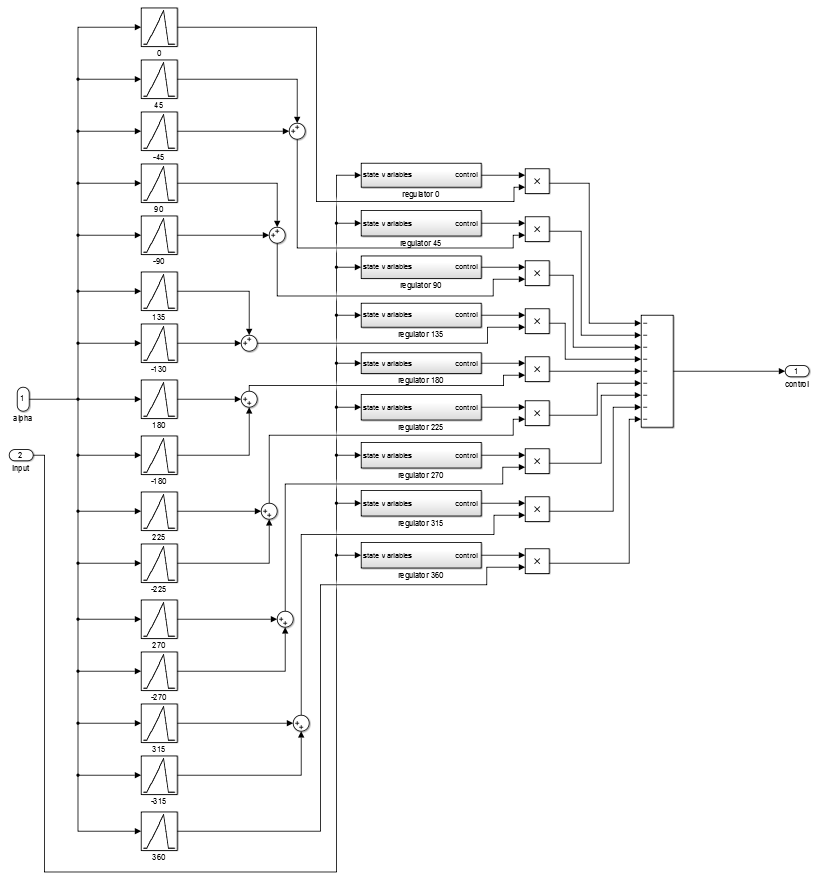
In this exercise will be used the code from the listing 1 included the last part of the code with LQR method.

In the figure 12 is shown the model of the system which included *Fuzzy control system* block, *Linear model* block,the *Saturation* block, *Radians to degrees* blocks and oscilloscopes. In the figure 13 is shown *Non*l*inear model* and in the figure 14 is shown *Fuzzy control system.* The saturation maximal value is equal to *10* and minimal value is equal to *-10*.

*Figure 12. The model of the system.*

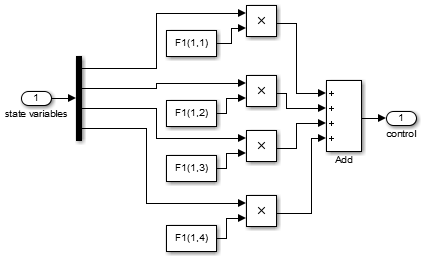
**

*Figure 13. The nonlinear model.*



*Figure 14. Fuzzy control system.*

The linear model (figure 13) is similar to linear model from exercise 1 (figure 7) but there are additional outputs for state variables in this model. The content of the regulator block is shown in the figure 15 and there are state variables.



*Figure 15. The regulator.*

In the figure 14 there are also additional triangles for negative values of pendulum angle. Below in the figures 16, 17, 18, 19 and 20 are shown responses of the system for initial conditions , ,, and.

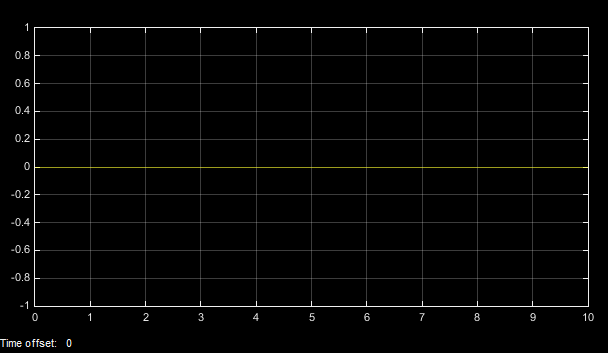
**

Figure 16. The system response for the initial condition equal to .

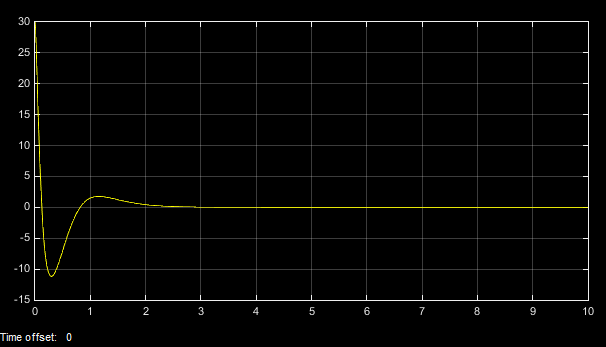
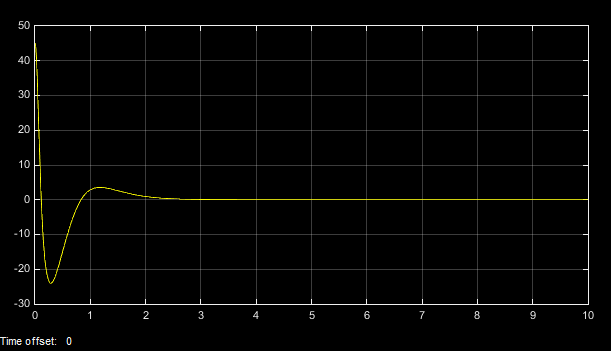
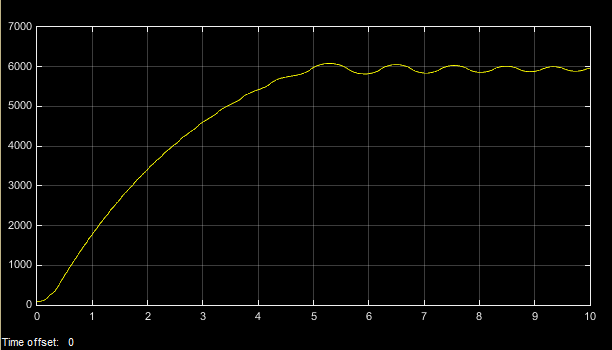


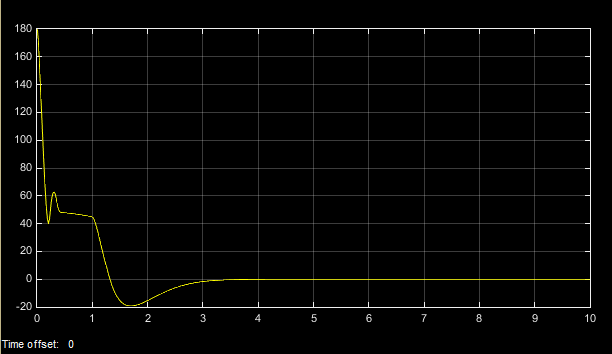
Figure 17. The system response for the initial condition equal to .



*Figure 18. The system response for the initial condition equal to .*



*Figure 19. The system response for the initial condition equal to .*



*Figure 20. The system response for the initial condition equal to .*

For the initial condition equal to the pendulum is in stable position and if there are not any another force which influenced on the pendulum, the pendulum will be stable in this point. This is not possible to check if the designed controller works properly.

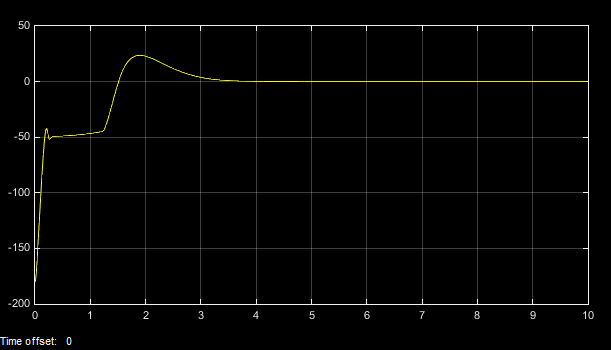
For the initial condition which is equal to and the responses are similar and the pendulum is stabilized in the point 0. The controller work properly.

In the figure 19 there is initial condition which is equal to and this point is uncontrollable.

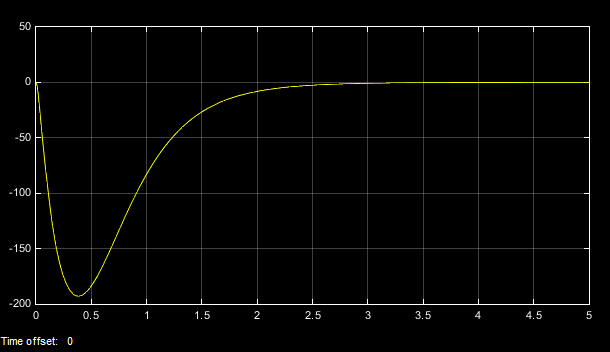
For the initial condition which is equal to the pendulum is stabilized in point 0. The pendulum has been swinging and the controller works properly.

To sum up the controller works properly for values in the range from to (response for value is shown in the figure 21) except the uncontrollable points like .

The maximum value of the angle, for example for the initial condition , is equal to about degrees and this is acceptable value. The chart for the changing of angle and for initial condition is shown in the figure 22.



*Figure 21. The system response for the initial condition equal to .*



*Figure 22. The chart for the changing of angle for the initial condition .*

For comparison there was used also PID controller. For the initial condition which was equal to (and without input signal) the response was of course the same like for fuzzy controller (figure 16).It was possible to stabilize the pendulum by using PID controller and the code which is shown in the listing 3. It means there is the system with A, B, C and D matrices, PID controller and impulse disturbance. The response is shown in the figure 23.

% The system

sys = ss(A1,B1,C,D);

system = tf(sys)

% PID parameters

Kp = 100;

Ki = 1;

Kd = 1;

% The feedback with PID controller

V = pid(Kp,Ki,Kd);

T = feedback(system,V)

% The input signal

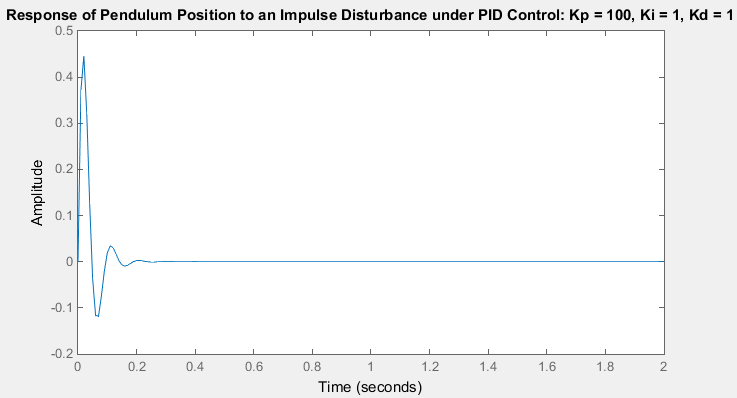
t=0:0.01:2;

impulse(T,t)

%The response

title('Response of Pendulum Position to an Impulse Disturbance under PID Control: Kp = 100, Ki = 1, Kd = 1');

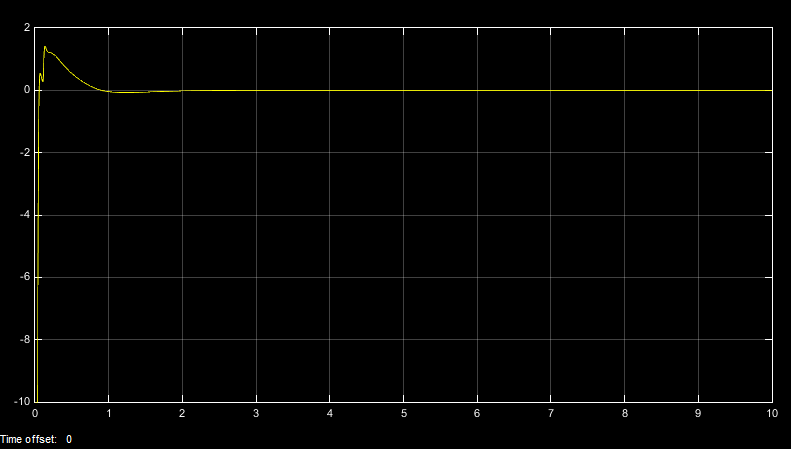
*Listing 3. The code for PID controller.*



*Figure 23. The response of the pendulum to an impulse disturbance.*

To sum up, finally the LQR method with fuzzy system seems better than method with PID controller (for example knowledge of the plant is not required) but the second method is simplest.

At the end there is shown also the input signal before saturation in the figure 24 for example for initial condition *.* The response was shown it the figure 18.



*24. The input signal before saturation for the initial condition .*

# Additional exercise

In classical regulation system obtaining stability is a main problem. In linear systems it is easy to state that the system is stable. We need to check where are poles of closed loop system and if they are in the left half-plane of complex variable s. When it is true, the system is stable. In nonlinear systems the stability criteria are not so easy to define. Most commonly used methods are: Lapunov method, phase plane method, describing function method and Popov method. The Lapunov methods, both direct and indirect are analytical methods. The direct method is a general method but the difficulty with finding so called Lapunov’s function is a disadvantage. The indirect method allow to analyse the stability of nonlinear system due to study its linear approximation in a small environment. The phase plane method is a graphical method and its application is limited to the second order systems. On the other hand, describing function method is not limited by the order of the system, but it is an approximation method. In systems where we can distinguish the linear dynamic element and nonlinear element without dynamics, we can apply the Popov method. Unfortunately, it defines only sufficient conditions of stability.

Studying stability of nonlinear system, including the fuzzy control system, is not easy. Moreover, in most cases the knowledge of the object as a model like in Lapunov method, or as the frequency characteristics like in Popov method is required. However, one of the advantages of the fuzzy controlling is a fact that it is possible to set the regulator without specific knowledge of the object. Unfortunately, this approach makes it impossible to study the stability. Eventually, this problem is either ignored, or it is assumed that a model exists. The model can be approximated (fuzzied) or not. In the second situation, the problem of stability comes down to the problem of stability of a nonlinear system.

It is possible to study control systems with fuzzy controller which will be nonlinearity and linearized plant in the work point. Study of this regulators do not seems to be easy because do not exist one popular method and it is not possible to specify the analytic formula which describe the regulator. To stability testing are also uses Kuderewicz and Junger methods. Analytically are used also proofs of stability for the systems with fuzzy controller for single and multi outputs.

The idea of fuzzy controlling is commonly used by engineers that design automatic systems. However, it does not have theoretical background, therefore it is necessary to develop such a method of analysis and synthesis of fuzzy controller that would be accepted both by practitioners and theroreticians. Moreover, It seems interesting to combine learning of fuzzy controller with studying its stability and quality.

By the way, there are be the differences between the real object and the model of the objects which are connected with discretization error and with other forces acting on an object, for example friction.