clear

% Załadowanie zmiennych wyliczonych w poprzednich krokach

load("idenyfikacja\_silnika\_zmienne.mat")

load("idenyfikacja\_ciagu\_zmienne.mat")

load("identyfikacja\_wahania\_zmienne.mat", "theta\_opt")

% warunki początkowe [pitch, ?, ?]

% UWAGA ZMIENIŁO SIE ALPHA\_0 z 0.2ileś na 0.186

x0 = [0.186; 0; 0]

% Generowanie sygnalu prostokatnego z rosnaca amplituda, za pomoca bloczka signal builder

t = [0 1 1 2 2 3 3 4 4 5 5 6 6 7 7 8] .\*10;

y = [1 1 -1 -1 2 2 -2 -2 3 3 -3 -3 4 4 -4 -4] .\*0.1;

%input = silnik.signals(1).values;

%pitch = silnik.signals(2).values;

%rpm = silnik.signals(3).values;

%time = silnik.time;

%save("przebieg\_dynamiczny\_pochylenia\_prost\_naras.mat", "time", "pitch", "rpm", "input")

load("przebieg\_dynamiczny\_pochylenia\_prost\_naras.mat")

% Zarejestrowany przebieg odpowiedzi układu na zadane sterowanie

% Sterowanie ograniczono do +/- 0.4

figure

subplot(2, 1, 1);

plot(time, input);

title("Przebieg sygnału sterującego control = f(t)");

ylabel("control [-]")

subplot(2, 1 , 2);

plot(time, pitch);

title("Przebieg położenia kątowego \phi = f(t)");

xlabel("Czas t [s]")

ylabel("\phi [rad]")

% Wyznaczenie optymalnej wartosci kp z równania ... + kp \* (F(x2))

m\_vec = [];

kp = 0:.1:10;

% Wyliczanie wspolczynnikow dla roznych wartosci kp metoda rk4

for KP = kp

[~, x] = rk4\_pochylenie(input, x0, 80, size(input, 1) - 1, theta\_opt, KP, K\_opt, H, F);

m = mean((pitch - x(:, 1)).^2); % sprawdzanie dopasowania metoda najmniejszych kwadratow

m\_vec = [m\_vec, m];

end

figure

plot(kp, m\_vec); grid on;

% Optymalna wartosc kp\_opt

[~, i] = min(m\_vec);

kp\_opt = kp(i)

[time, x] = rk4\_pochylenie(input, x0, 80, size(input, 1) - 1, theta\_opt, kp\_opt, K\_opt, H, F);

figure

plot(time, x(:, 1), time, pitch); grid on

xlim([0 80])

title("Porównanie modelu optymalnego z rzeczywistym przebiegiem \phi = f(t)")

xlabel("Czas t [s]")

ylabel("Amplituda \phi [rad]")

% Generowanie przebiegu piłokształtnego narastającego

t = [0 1 1 2 2 3 3 4 4 5 5 6 6 7 7 8] .\*10;

y = [0 1 0 -1 0 2 0 -2 0 3 0 -3 0 4 0 -4] .\*.1;

%input = silnik.signals(1).values;

%pitch = silnik.signals(2).values;

%rpm = silnik.signals(3).values;

%time = silnik.time;

%save("przebieg\_dynamiczny\_pochylenia\_piłoz\_naras.mat", "time", "pitch", "rpm", "input")

%load("przebieg\_dynamiczny\_pochylenia\_piłoz\_naras.mat")

% Zarejestrowany przebieg odpowiedzi układu na zadane sterowanie

% Sterowanie ograniczono do +/- 0.4

figure

subplot(2, 1, 1);

plot(time, input);

title("Przebieg sygnału sterującego control = f(t)");

ylabel("control [-]")

subplot(2, 1 , 2);

plot(time, pitch);

title("Przebieg położenia kątowego \phi = f(t)");

xlabel("Czas t [s]")

ylabel("\phi [rad]")

% TODO zrobic wyznaczenie optymalnego kp dla powyzszego przebiegu i moze jeszcze dla jednego

% ALE NA PEWNO JESZCZE RAZ ALPHA ZERO OGARNAC

% Zapis optymalnych współczynników do pliku

save("identyfikacja\_pochylenie.mat", "kp\_opt", "theta\_opt", "K\_opt", "H", "F")