**Министерство образования и науки Российской Федерации** ФГАОУ ВО «Уральский федеральный университет имени первого Президента России Б.Н.Ельцина»

**МОДАЛЬНЫЙ СИНТЕЗ УПРАВЛЕНИЯ В ЛИНЕЙНЫХ НЕПРЕРЫВНЫХ СИСТЕМАХ**

Методические указания и задания к лабораторной работе по дисциплине "Основы теории управления" для студентов направления подготовки 09.03.01 - Информатика и вычислительная техника

Студенты:

Бальбола Айад шади Ри-320944

Митреи Джошва Саид Камал РИ-320913

Салама нур самер махмуд саад

Преподаватель:

**Цветков Александр Владимирович**

Екатеринбур 2016

Joooe amsa7 de w 700t el graph

1. %% Modal Control Synthesis for Linear Continuous Systems

% Clear workspace and close figures

clear all; clc; close all;

% System Parameters (replace T1, T2, T3 with specific values from your variant)

T1 = 4; % Example value

T2 = 1; % Example value

T3 = 48; % Example value

% Define Laplace variable

p = tf('p');

% Define open-loop transfer function

R = T1\*p + 1;

Q = (T2\*p + 1) \* (T3\*p - 1) \* p;

W\_open = R / Q;

disp('Open-loop Transfer Function:');

W\_open

% Convert to State-Space representation

sys\_open = ss(W\_open);

[A, B, C, D] = ssdata(sys\_open);

% Display state-space matrices

disp('State-Space Matrices:');

disp('A:'), disp(A)

disp('B:'), disp(B)

disp('C:'), disp(C)

disp('D:'), disp(D)

% Compute poles and zeros of the open-loop system

[z\_open, p\_open, k\_open] = zpkdata(zpk(W\_open), 'v');

disp('Poles of Open-loop System:'), disp(p\_open)

disp('Zeros of Open-loop System:'), disp(z\_open)

% Desired poles for closed-loop system (replace based on your design)

desired\_poles = [-2 + 2j, -2 - 2j, -0.25];

disp('Desired Closed-Loop Poles:'), disp(desired\_poles)

% Calculate feedback matrix

[L, prec, msg] = place(A, B, desired\_poles);

disp('Feedback Matrix L:'), disp(L)

% Construct closed-loop system

A\_closed = A - B \* L;

B\_closed = B;

C\_closed = C;

D\_closed = D;

sys\_closed = ss(A\_closed, B\_closed, C\_closed, D\_closed);

% Compute gain for unit static response

W\_closed = tf(sys\_closed);

[num, den] = tfdata(W\_closed, 'v');

K\_v = den(end) / num(end);

disp('Gain for Command Signal (K\_v):'), disp(K\_v)

% Adjust B matrix with gain

B\_closed = K\_v \* B;

sys\_closed = ss(A\_closed, B\_closed, C\_closed, D\_closed);

% Verify closed-loop poles

[z\_closed, p\_closed, k\_closed] = zpkdata(zpk(sys\_closed), 'v');

disp('Poles of Closed-loop System:'), disp(p\_closed)

disp('Zeros of Closed-loop System:'), disp(z\_closed)

% Plot pole-zero map

figure;

plot(real(p\_open), imag(p\_open), 'ro', 'MarkerSize', 8, 'DisplayName', 'Open-loop Poles');

hold on;

plot(real(z\_open), imag(z\_open), 'rx', 'MarkerSize', 8, 'DisplayName', 'Open-loop Zeros');

plot(real(p\_closed), imag(p\_closed), 'bo', 'MarkerSize', 10, 'DisplayName', 'Closed-loop Poles');

plot(real(z\_closed), imag(z\_closed), 'bx', 'MarkerSize', 10, 'DisplayName', 'Closed-loop Zeros');

grid on;

title('Pole-Zero Map');

xlabel('Real Axis'); ylabel('Imaginary Axis');

legend;

% Step Response Comparison

figure;

step(W\_open, 'r--', sys\_closed, 'b-');

title('Step Response Comparison');

legend('Open-loop', 'Closed-loop');

xlabel('Time (s)'); ylabel('Amplitude');

grid on;