**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра КСУ**

отчет

**по идз №3**

**по дисциплине «ПРОЕКТИРОВАНИЕ**

**ОПТИМАЛЬНЫХ СИСТЕМ УПРАВЛЕНИЯ»**

Тема: Модальное управление

**Вариант 12**

|  |  |  |
| --- | --- | --- |
| Студент гр. 4491 | Пономарев Д.А. |  |
| Преподаватель | Ветчинкин А.С. |  |

Санкт-Петербург

2018

**Модальное управление**

**Исходные данные**

Найти значение базовой частоты заданного типа полинома, обеспечивающее минимальное время переходных процессов в замкнутой системе управления и построить графики переходных процессов по состояниям объекта управления и управляющему воздействию.

Максимально допустимое значение управляющего воздействие принимается равным 1 и вектор начальных условий [1 0 0].

Исходные данные заданы в таблице 1.

Таблица 1. Исходные данные к заданию

|  |  |
| --- | --- |
| Вариант | Матрицы объекта управления и полином |
| 12 |  |

**Описание программы**

Файловая архитектура программы представлена на рисунке 1.

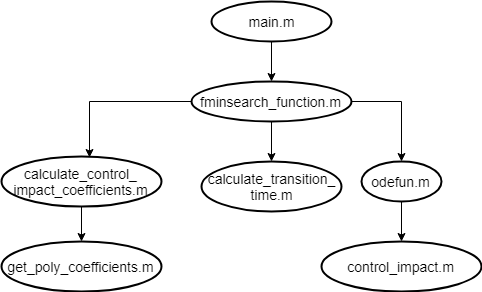


Рисунок 1 – файловая архитектура программы

Основная цель задания – найти значение базовой частоты, обеспечивающее минимальное время переходных процессов.

Поиск данного значения осуществляется с помощью функции fminsearch(), вызов которой осуществляется в файле main.m, код которого представлен на рисунке 1. В main файле также объявляются и определяются значения исходных данных варианта и после всех расчетов строятся графики управляющего воздействия и переходного процесса.

|  |
| --- |
| clear; close all; clc;  % 3. Модальное управление Input variant12 data  global A B POLY\_TYPE SYSTEM\_DEGREE x0  SYSTEM\_DEGREE = 3; POLY\_TYPE = 4;  A = [-1 1 0; 0 -2 2; 0 0 1]; B = [0 0 1]; x0 = [1 0 0];  % Find transition time and base frequency and create state variables graph  w0 = 2;  [w,time] = fminsearch('fminsearch\_function', w0)  figure  [t, x] = ode45('odefun', [0 5], x0);  u = control\_impact(x);  plot(t, u)  grid on; legend('u'); xlabel('t'); ylabel('u(t)');  figure  plot(t, x)  grid on; legend('x1', 'x2', 'x3'); xlabel('t'); ylabel('x(t)'); |

Рисунок 2 – Код основного файла

Аргументом функции fminsearch является ссылка на функцию fminsearch\_function(), код которой представлен на рисунке 3. В данном файле выполняется расчет коэффициентов функции управляющего воздействия по значению базовой частоты с помощью функции calculate\_control\_impact\_coefficients(). После чего происходит вызов функции ode45(), аргументом которой является ссылка на функцию odefun(). В конце происходит расчет максимального времени переходного процесса с помощью функции calculate\_transition\_time().

|  |
| --- |
| function transitionTime = fminsearch\_function(w0)  %@brief find transition time and create graph for w0 value  %@param w0 - base frequency value  %@return max transition time value  global K x0  K = calculate\_control\_impact\_coefficients(w0);  [t, x] = ode45('odefun', [0 10], x0);  transitionTime = calculate\_transition\_time(t, x);    %plot(t, x)  %pause(0.5)  end |

Рисунок 3 – Код файла fminsearch\_function.m

Коэффициенты функции управляющего воздействия К найдем в общем виде:

1. Запишем исходную систему уравнений:

2. ХП замкнутой системы:

3. Сравнивая полученный характеристический полином замкнутой системы со стандартным полиномом 3 порядка:

4. Получаем систему уравнений:

где

Поиск значений коэффициентов для исходных данных:

1. Запишем исходную систему уравнений:

2. ХП замкнутой системы:

3. Сравнивая полученный характеристический полином замкнутой системы со стандартным полиномом 3 порядка:

4. Получаем систему уравнений:

Расчет значений коэффициентов происходит в функции calculate\_control\_impact\_coefficients(), представленной на рисунке 4.

|  |
| --- |
| function k = calculate\_control\_impact\_coefficients(w0)  %brief Calculate control impact coefficients K1, K2, K3 and show roots of  %system Ax=B  %@param w0 - base frequency scalar  %@return control impact coefficient vector with size (3, 1)    global A B POLY\_TYPE SYSTEM\_DEGREE  a = get\_poly\_coefficients(SYSTEM\_DEGREE, POLY\_TYPE);    %A1 = matrix derived from det(A(x)+B(x) - s\*I)  A1 = [0 0 1; ... % s^2  A(1,3) A(2,3) -A(1,1) - A(2,2); ... % s^1  -A(1,3)\*A(2,2)+A(1,2)\*A(2,3) -A(1,1)\*A(2,3)+A(1,3)\*A(2,1) A(1,1)\*A(2,2)]; % s^0    %B1 = matrix derived from det(A(x)+B(x) - s\*I) and poly of POLY\_TYPE type  B1 = [A(1,1) + A(2,2) + A(3,3) + a(3)\*w0; ...  -A(1,1)\*A(2,2) - A(1,1)\*A(3,3) - A(2,2)\*A(3,3) + A(1,3)\*A(3,1) + A(1,2)\*A(2,1) + A(2,3)\*A(3,2) + a(2)\*w0^2; ...  -A(1,3)\*A(2,2)\*A(3,1) - A(3,3)\*A(1,2)\*A(2,1) - A(1,1)\*A(2,3)\*A(3,2) + A(1,2)\*A(2,3)\*A(3,1) + A(1,3)\*A(2,1)\*A(3,2) + A(1,1)\*A(2,2)\*A(3,3) + a(1)\*w0^3];    % Coefficients:  k = A1\B1;    % Original system characteristic polynomial roots with found coefficients:  roots(poly(A - (k\*B)'));  end |

Рисунок 4 – Код файла calculate\_control\_impact\_coefficients.m

Код функции, возвращающей значение коэффициентов полинома в зависимости от его степени и типа представлен на рисунке 5.

|  |
| --- |
| function a = get\_poly\_coefficients(degree, polyType)  %@brief get coefficients of standart polynomial  %@param degree - poly degree  %@param degree - poly type  %return vector with size degree+1  a = zeros(1, degree + 1);  switch polyType  case 1 % Binominal Polynomials (Newton Polynomials)  if degree == 3  a = [1 3 3 1];  end  case 2 % Butterworth Polynomials  if degree == 3  a = [1 2 2 1];  end  case 3 % Polynomials that minimize the functional  if degree == 3  a = [1 2 1 1];  end  case 4 % Polynomials that minimize the functional  if degree == 3  a = [1 2.15 1.75 1];  end  end  end |

Рисунок 5 – Код файла get\_poly\_coefficients.m

Код функции odefun, ссылка на которую передается в качестве аргумента ode45, представлен на рисунке 6.

|  |
| --- |
| function dxdt = odefun(t, x)  %@brief ode45 function  %@param t - time vector with size(pointsAmount, 1)  %@param x - state variables values with size(pointsAmount, stateVariablesAmount)  global A B  dxdt = zeros(3, 1);  dxdt(1) = A(1,1)\*x(1) + A(1,2)\*x(2) + A(1,3)\*x(3) + B(1)\*control\_impact(x');  dxdt(2) = A(2,1)\*x(1) + A(2,2)\*x(2) + A(2,3)\*x(3) + B(2)\*control\_impact(x');  dxdt(3) = A(3,1)\*x(1) + A(3,2)\*x(2) + A(3,3)\*x(3) + B(3)\*control\_impact(x');  end |

Рисунок 6 – Код файла odefun.m

Расчет значения управляющего воздействия по значению переменных состояния происходит в функции control\_impact(), код которой представлен на рисунке 7.

|  |
| --- |
| function u = control\_impact(x)  %@brief Calculate control impact u = K1\*x1 + K2\*x2 + K3\*x3  %@note u <= |Umax|  %@param x - state variables with size (pointsAmount, stateVariablesAmount)  %@return control impact with size(pointsAmount, 1)  global K  Umax = 1;  u = -K(1).\*x(:, 1) - K(2).\*x(:, 2) - K(3).\*x(:, 3);  for i = 1:length(u)  if u(i) > Umax  u(i) = Umax;  elseif u(i) < -Umax  u(i) = -Umax;  end  end  end |

Рисунок 7 – Код файла control\_impact.m

Код функции calculate\_transition\_time(), которая рассчитывает время переходного процесса, представлена на рисунке 8.

|  |
| --- |
| function transitionTime = calculate\_transition\_time(t, x)  %@brief Calculate transition time for each state variabels and return max  %@param t - time vector with size (pointsAmount, 1)  %@param x - state variables matrix with size(pointsAmount, stateVariabelsAmount)  %@return max transition time scalar    [pointsAmount, stateVariabelsAmount] = size(x);    transitionTime = [t(end), t(end), t(end)];  for variableNumber = 1:stateVariabelsAmount  for i = pointsAmount : -1 : 1  if abs(x(i, variableNumber)) > 0.05  transitionTime(variableNumber) = t(i);  break;  end  end  end  transitionTime = max(transitionTime);  end |

Рисунок 8 – Код файла calculate\_transition\_time.m

В результате выполнения программы были построены график переходного процесса с минимальным временем переходного процесса, который представлен на рисунке 9, и график управляющего воздействия, который представлен на рисунке 10. Также были найдены значения базовой частоты, при которой время переходного процесса минимально, и само значение времени переходного процесса.

Значение базовой частоты = 1.9707

Минимальное время переходного процесса = 1.8391c

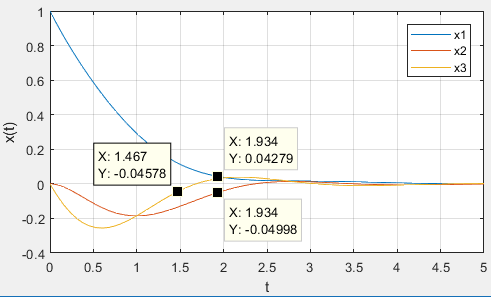


Рисунок 9 – График переходного процесса

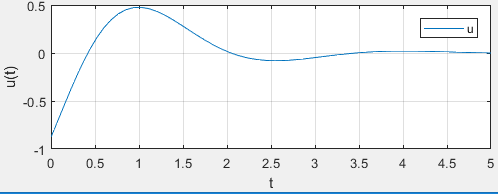


Рисунок 10 – График управляющего воздействия