**Лабораторна робота 8**

**Тема:** задача оптимального управління

**Мета:** отримати практичні навички розв’язання задачі оптимального управління детермінованими об’єктами з звичайними параметрами, які описуються звичайними диференціальними рівняннями методом варіаційного числення

Опишемо символічні змінні. Для розв’язання рівняння Ейлера використовуємо прийняті *в MATLAB* позначення похідних: Dy для *y*' і D2y для *y*''. Аргумент позначимо *t*, а функцію – *u*.

1. clear all % очистили пам’ять

syms t u y1 Dy1 y2 Dy2 L1 DL1 L2 DL2 % описали символічні змінні

F=u^2+L1\*(Dy1-y2)+L2\*(Dy2-u) ; % функція Лагранжа

t0=0; % граничні умови

t1=10;

y1\_t0=12;

y2\_t0=6;

y1\_t1=30;

y2\_t1=8;

disp('Вихідні дані:')

fprintf(['Функція Лагранжа '...

'F(t,u,y)=%s\n'],char(F))

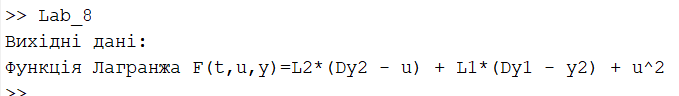


Рисунок 1 - розв’язання рівняння Ейлера

2.Починаємо введення необхідних умов екстремуму (1 – 2). Знайдемо часткові похідні для кожної змінної. Надрукуємо їх.

Fu=2\*u - L2

Fy1=0

dFdu=diff(F,u); % обчислили Fu

dFdy1=diff(F,y1); % обчислили Fy1

dFdDy1=diff(F,Dy1); % обчислили FDy1

dFdy2=diff(F,y2); % обчислили Fy2

dFdDy2=diff(F,Dy2); % обчислили FDy1

dFdL1= diff(F,L1); % обчислили FL1

dFdL2= diff(F,L2); % обчислили FL2

fprintf('FDy1=%s\n',char(dFdDy1))

fprintf('Fy2=%s\n',char(dFdy2))

fprintf('FDy2=%s\n',char(dFdDy2))

fprintf('FL1=%s\n',char(dFdL1))

fprintf('FL2=%s\n',char(dFdL2))

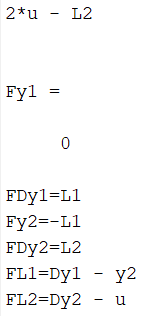


Рисунок 2 - функцію Лагранжа і граничні умови

3. У рівняння Ейлера входить повна похідна *dFy*'/*dx*. Обчислимо її за звичайною формулою диференціювання складної функції:

.

Знайдемо повну похідну

d\_dFDy1\_dt=diff(dFdDy1,t); % Fty1't

d\_dFDy1\_L1=diff(dFdDy1,L1); % Fy1L1

dFy1dt=d\_dFDy1\_dt+d\_dFDy1\_L1\*DL1; %

d\_dFDy2\_dt=diff(dFdDy2,t); % Fty1't

d\_dFDy2\_L2=diff(dFdDy2,L2); % Fy2L2

dFy2dt=d\_dFDy2\_dt+d\_dFDy2\_L2\*DL2; %

fprintf('dFDy1/dt=%s\n',char(dFy1dt))

fprintf('dFDy2/dt=%s\n',char(dFy2dt))



Рисунок 3 - повна похідна для змінних *y*1 та *y*2.

4. Складемо ліву частину диференціального рівняння Ейлера для змінних *y*1 та *y*2 і спростимо її. Перетворимо символічні змінні Euler\_y1 та Euler\_y2 в рядок. Додамо рівняння для змінних u, L1 та L2.

Euler\_y1=simple(dFdy1-dFy1dt); % рівняння Ейлера для y1

deqEuler\_y1=[char(Euler\_y1) '=0']; % в рядок додали =0

fprintf('Рівняння Ейлера для y1:\n%s\n',deqEuler\_y1)

Euler\_y2=simple(dFdy2-dFy2dt); % рівняння Ейлера для y2

deqEuler\_y2=[char(Euler\_y2) '=0']; % в рядок додали =0

fprintf('Рівняння Ейлера для y2:\n%s\n',deqEuler\_y2)

eq\_u = [char(dFdu) '=0'];

eq\_L1 = [char(dFdL1) '=0'];

eq\_L2 = [char(dFdL2) '=0'];

fprintf('Рівняння для u:\n%s\n',eq\_u)

fprintf('Рівняння для L1:\n%s\n',eq\_L1)

fprintf('Рівняння для L2:\n%s\n',eq\_L2)

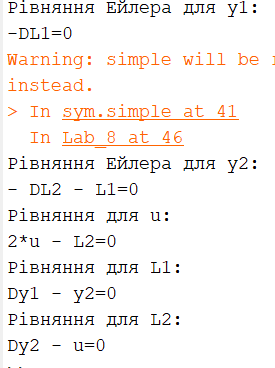


Рисунок 4 - диференціального рівняння Ейлера для змінних *y*1 та *y*2

5.Ми склали рівняння Ейлера, тепер розв’яжемо систему рівнянь. Функція ***dsolve*** дозволяє знаходити як загальне розв’язання диференціального рівняння, так і часткове його розв’язання, що задовольняє задані початкові або граничні умови. Знайдемо розв’язок системи диференціальних рівнянь.

eq\_L22 = char(subs(eq\_L2,u,solve(eq\_u,u)));

Sol=dsolve(deqEuler\_y1, deqEuler\_y2, eq\_L1, eq\_L22, 'y1(0) = 12', 'y2(0) = 6'); % вирішуємо систему рівнянь

% вирішуємо систему рівнянь

6.Сформуємо тепер рівняння для граничних умов, підставивши в знайдений аналітичний розв’язок *Sol* граничні точки *t*1 і *t*2 і дорівнявши їх відповідно до y1\_t1 і y2\_t1.

Left\_y1=subs(Sol.y1,t,t1); % підставили t1

Left\_y2=subs(Sol.y2,t,t1); % підставили t1

Eq\_y1=[char(Left\_y1) '=' char(sym(y1\_t1))]; % =y1\_t1

Eq\_y2=[char(Left\_y2) '=' char(sym(y2\_t1))]; % =y2\_t1

disp('Рівняння для граничних умов:')

fprintf('%s\n',Eq\_y1,Eq\_y2)

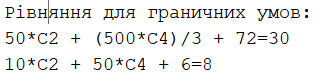


Рисунок 5 - рівняння для граничних умов

7.Розв’язуємо отриману систему кінцевих рівнянь – знаходимо значення довільних постійних *C*3 і *C*4. Присвоюємо знайдені розв’язки символічним константам, отриманим при розв’язанні системи диференціальних рівнянь. Тепер обчислюємо аналітичний розв’язок *Sol\_U*. Таке обчислення зводиться до того, що в нього будуть підставлені знайдені значення констант *C*3 і *C*4. Друкуємо знайдене управління.

Sol\_U= solve(char(subs(eq\_u,L2,Sol.L2)),u);

Con=solve(Eq\_y1,Eq\_y2,'C2,C4'); % розв’язуємо систему

C2=Con.C2; % прирівнюємо отриманні розв’язки

C4=Con.C4; % символічним константам C2 и C3

Sol\_y1=vpa(eval(Sol.y1),10); % підставили C3,C4

Sol\_y2=vpa(eval(Sol.y2),10); % підставили C3,C4

Sol\_U=vpa(eval(Sol\_U),10); % підставили C3,C4

disp('Рівняння управління')

fprintf('u(t)=%s\n',char(Sol\_U))

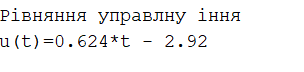


Рисунок 6 - Рівняння управління

8.Обчислимо значення функціонала, що оцінює якість управління.

JJ = u^2;

JJextr=subs(JJ,{u},{Sol\_U});

Jextr=eval(int(JJextr,t,t0,t1))

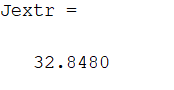


Рисунок 7 - значення функціонала

10. І нарешті, будуємо графік. Задаємо масив аргументів для рисування графіка функції і обчислюємо значення функції. Рисуємо графік, підписуємо заголовок і координатні осі встановленим шрифтом.

tp1=linspace(t0,t1); % задаємо масив абсцис

U2=subs(Sol\_U,t,tp1); % обчислили ординати

figure % фігура

plot(tp1,U2,'-r') % рисуємо графік червоною лінією

set(get(gcf,'CurrentAxes'),...

'FontName','Times New Roman Cyr','FontSize',12)

title('\bfУправління') % заголовок

xlabel('\itt') % мітка осі OX

ylabel('\itU\rm(\itt\rm)') % мітка осі OY

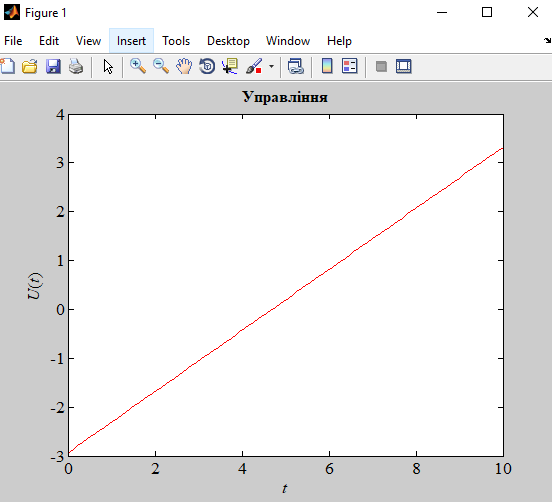


Рисунок 6 – Графік зміни управління

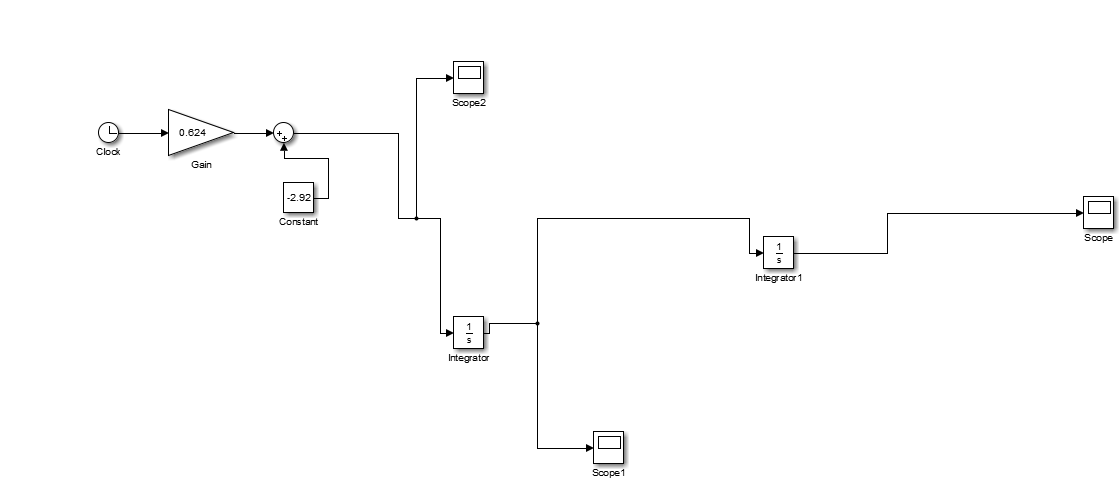


Рисунок 7 - Модель для розв’язання задачі

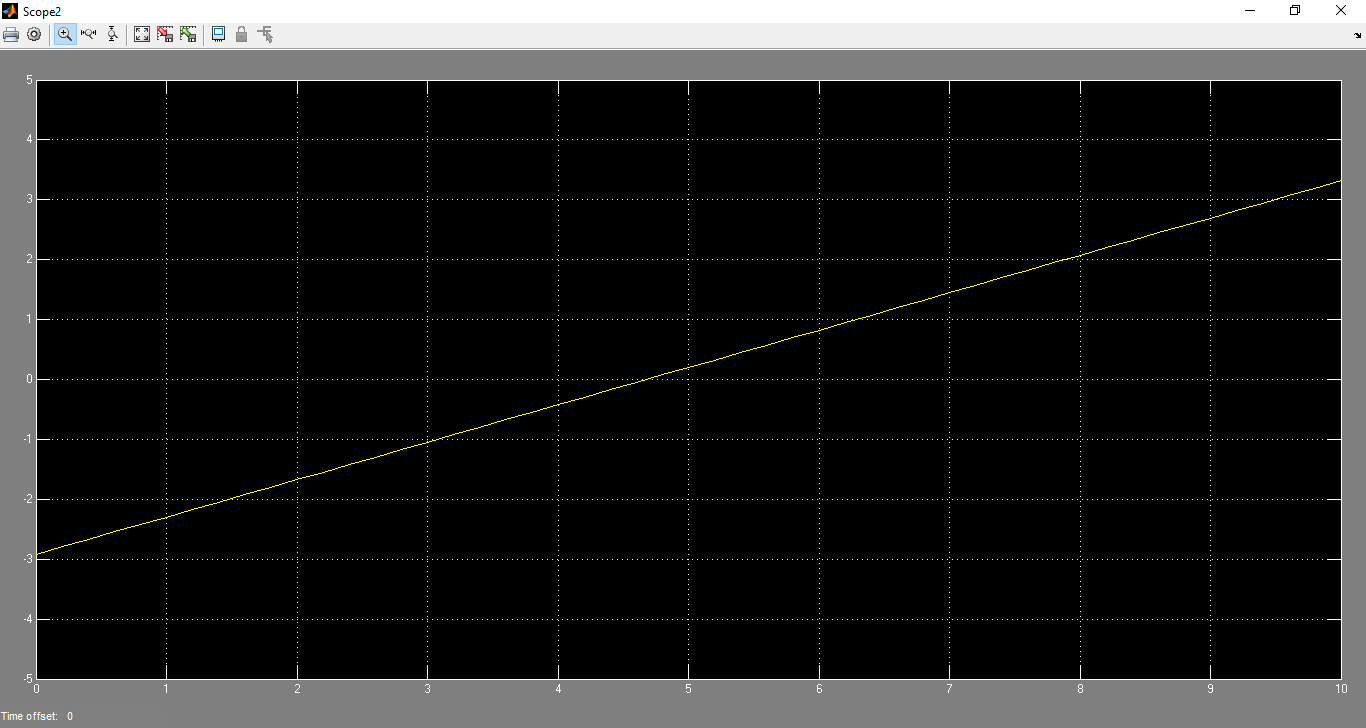


Рисунок 8 - Графік зміни управління (scope 2 )

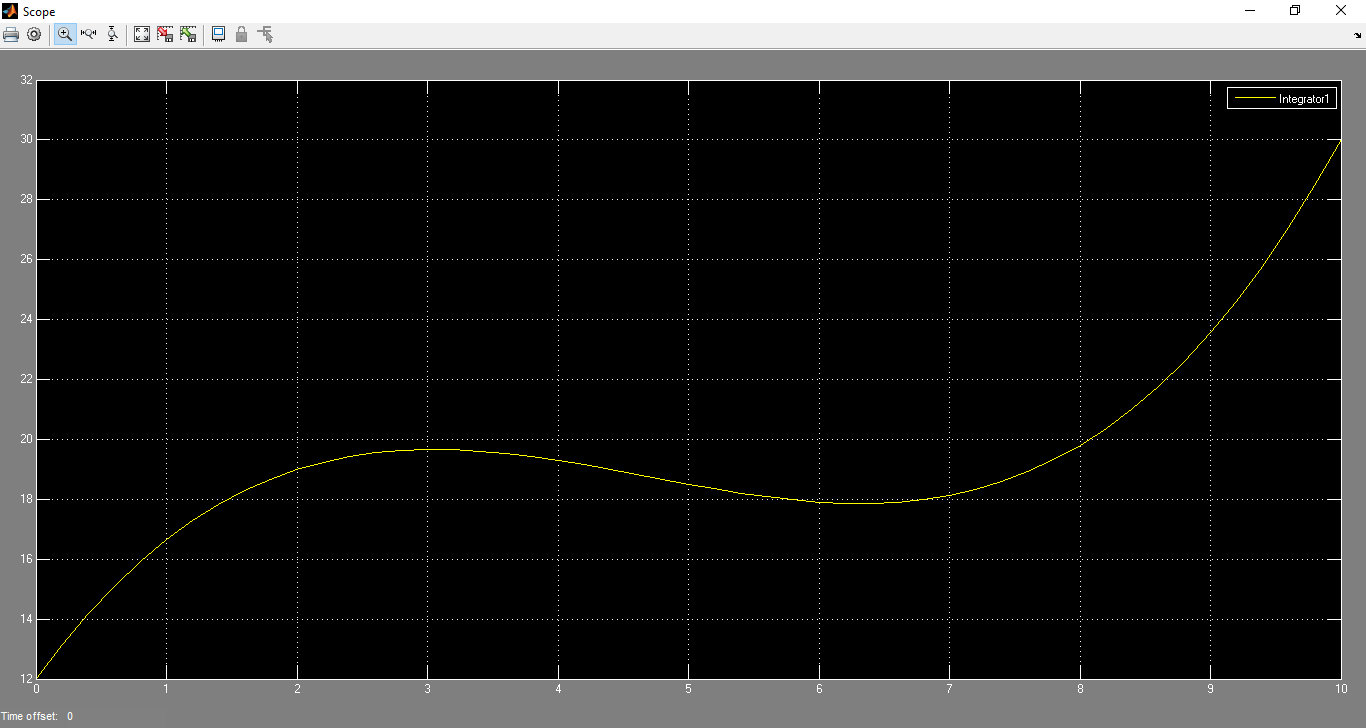


Рисунок 9 - Графік зміни управління (scope )

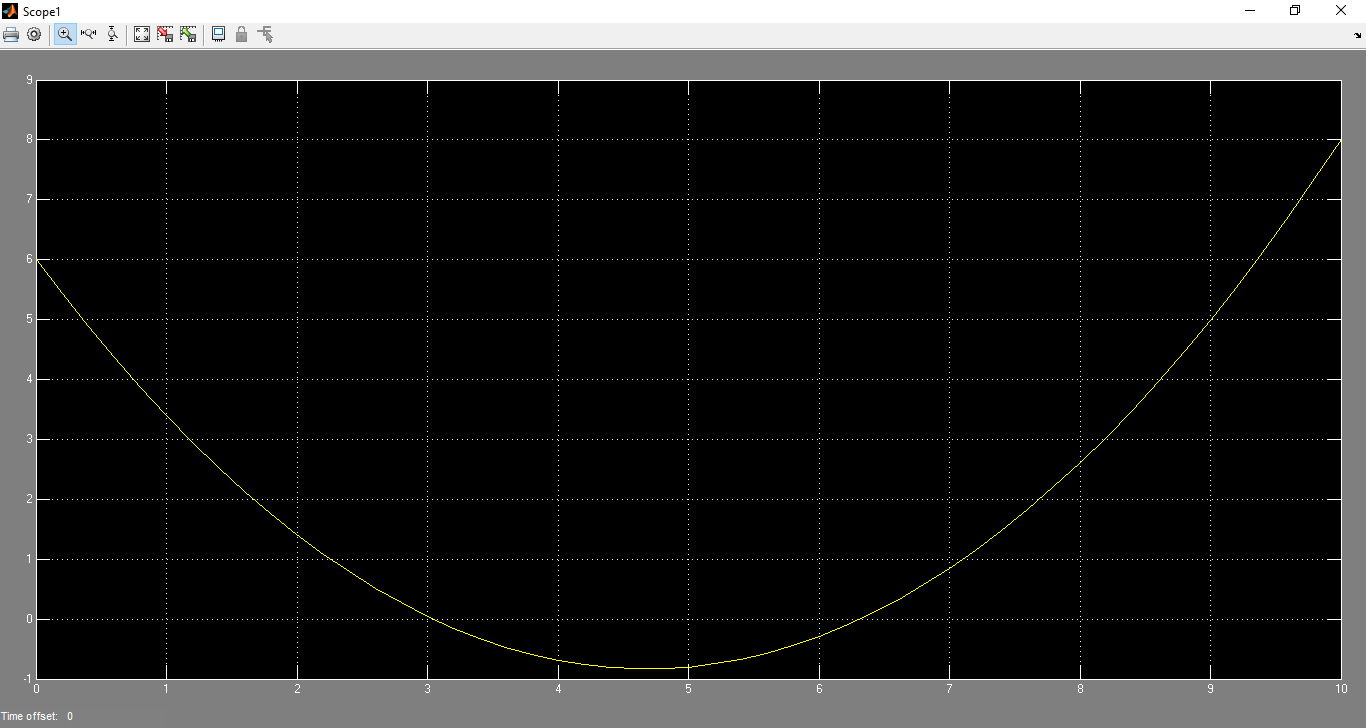


Рисунок 10 - Графік зміни управління (scope 1 )

Висновок

Отримали практичні навички розв’язання задачі оптимального управління детермінованими об’єктами з звичайними параметрами, які описуються звичайними диференціальними рівняннями методом варіаційного числення