КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ КОМП’ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ

Звіт

про виконання

Лабораторної роботи №3

Виконала

студентка групи ІПС-33

Петрик Юлія Олександрівна

Київ 2023

Тема: Параметрична ідентифікація параметрів з використанням функцій чутливості.

Зміст

* + - 1. Постановка задачі.
      2. Теоретичні відомості.
      3. Алгоритм виконання.
      4. Розв’язання задачі.
      5. Код програми.

Варіант №8

Постановка задачі

1. Знати означення функцiї чутливостi i вивчити диференцiальне рiвняння, з якого шукається матриця чутливостi. Записати рiвняння чутливостi для математичної моделi.

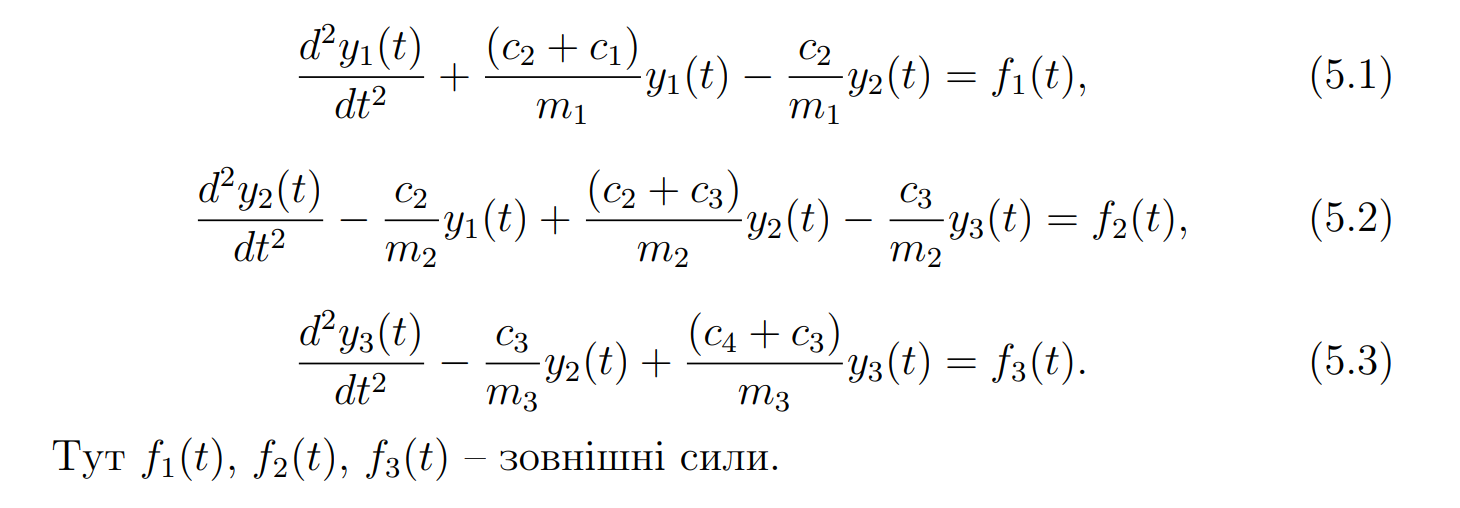
2. Створити програму, яка реалiзує метод параметричної iдентифiкацiї параметрiв з використанням функцiй чутливостi для математичної моделi.

3. Вивести знайденi параметри, значення показника якостi iдентифiкацiї параметрiв.

Теоретичні відомості

Математична модель коливання трьох мас m1, m2, m3, якi поєднанi мiж

собою пружинами з вiдповiдними жорсткостями c1, c2, c3, c4 має вигляд



Означення 5.1. Функцiєю чутливостi системи (5.4) в точцi p =

називається функцiя, яка задається спiввiдношенням .

Матриця чутливостi задовольняє матричне диференцiальне рiвняння (рiвняння чутливостi):

Метод параметричної iдентифiкацiї параметрiв з використанням функцiй чутливостi:

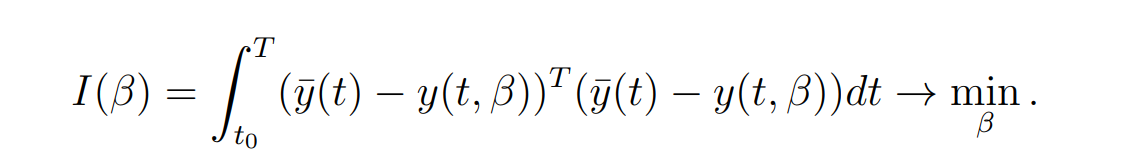
Потрiбно оцiнити частину невiдомих параметрiв моделi (5.1) - (5.3) з

використанням функцiї чутливостi за вiдомими спостереженнями y ̄(t) на

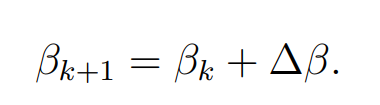
часовому iнтервалi t ∈ [0, T]. Для цього записуємо (5.1) - (5.3) у виглядi

системи диференцiальнiй формi в нормальнiй формi порядку 6.

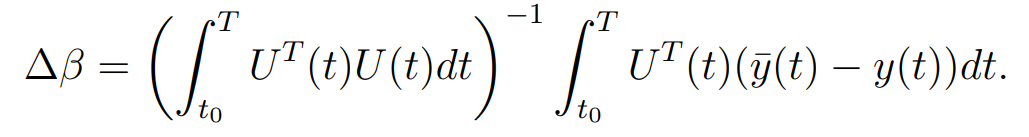
Показник якостi iдентифiкацiї параметрiв β має вигляд



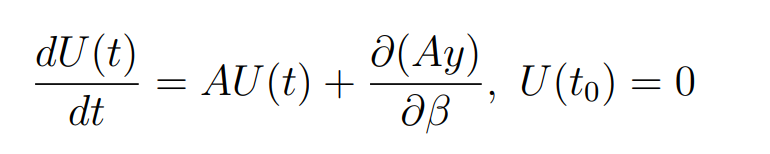
Числовий метод iтерацiйний i має вигляд



Початкове наближення β0 задається,



Матрицi чутливостi U(t) визначається з матричного диференцiального рiвняння



Алгоритм виконання

Iтерацiя 1.

Задаємо параметр точностi ε > 0, початкове наближення β0. В лабораторних роботах початкове наближення задається.

Крок 1. Розв’язуємо систему (5.1) - (5.3) одним з числових методiв

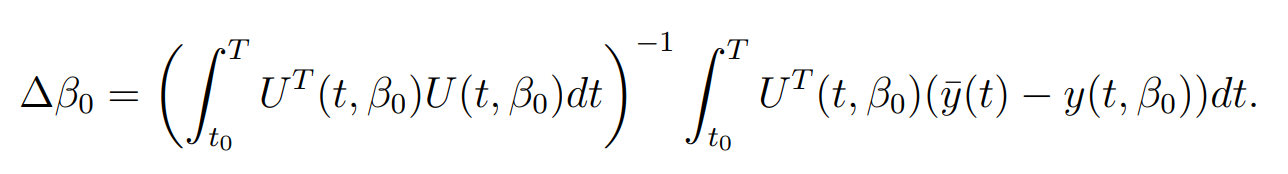
при β = β0, y(t0) = ̄y(t0) (можна застосувати метод Рунге- Кутти з

попередньої лабораторної роботи). Знаходимо y(t, β0).

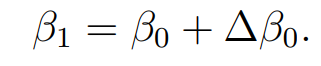
Крок 2. Пiдставляємо y(t, β0) в рiвняння чутливостi 5.7. Розв’язуємо

це рiвняння одним з числових методiв i знаходимо його розв’язок U(t, β0).

Крок 3. Знаходимо



Крок 4. Обчислюємо



Крок 5. В кiнцi першої iтерацiї перевiряємо одну з умов зупинки алгоритму:

1. ∆β1 < ε;

2. I(β1) < ε.

Якщо умова зупинки виконується, то алгоритм зупиняється i β = β1 є наближеним розв’язком задачi. Якщо не виконується то переходимо на iтерацiю 2

Iтерацiя k − 1.

На початку цiєї iтерацiї ми знаємо βk.

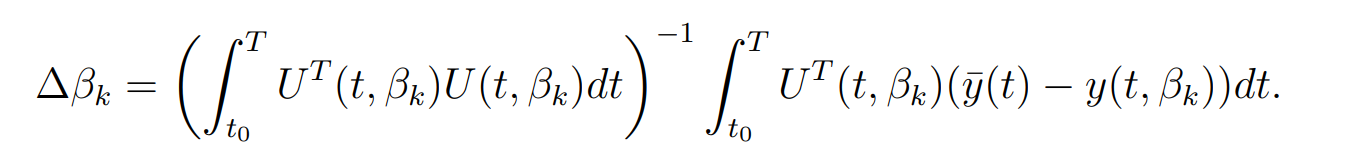
Крок 1. Розв’язуємо систему (5.1) - (5.3) одним з числових методiв

при β = βk, y(t0) = ̄y(t0). Знаходимо y(t, βk).

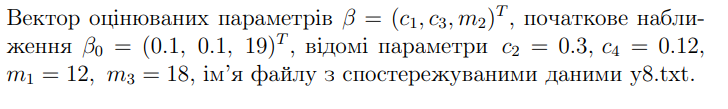
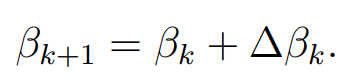
Крок 2. Пiдставляємо y(t, βk) в рiвняння чутливостi 5.7. Розв’язуємо

це рiвняння одним з числових методiв i знаходимо його розв’язок U(t, βk).

Крок 3. Знаходимо



Крок 4. Обчислюємо



Крок 5. У кiнцi iтерацiї перевiряємо одну з умов зупинки алгоритму:

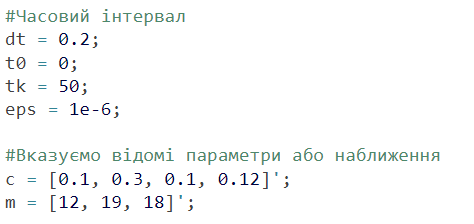
1. ∆βk+1 < ε;

2. I(βk+1) < ε.

Якщо умова зупинки виконується, то алоритм зупиняється i β = βk+1 є наближеним розв’язком задачi. Якщо не виконується то k := k + 1 i переходимо на наступну iтерацiю.

Розв’язання задачі

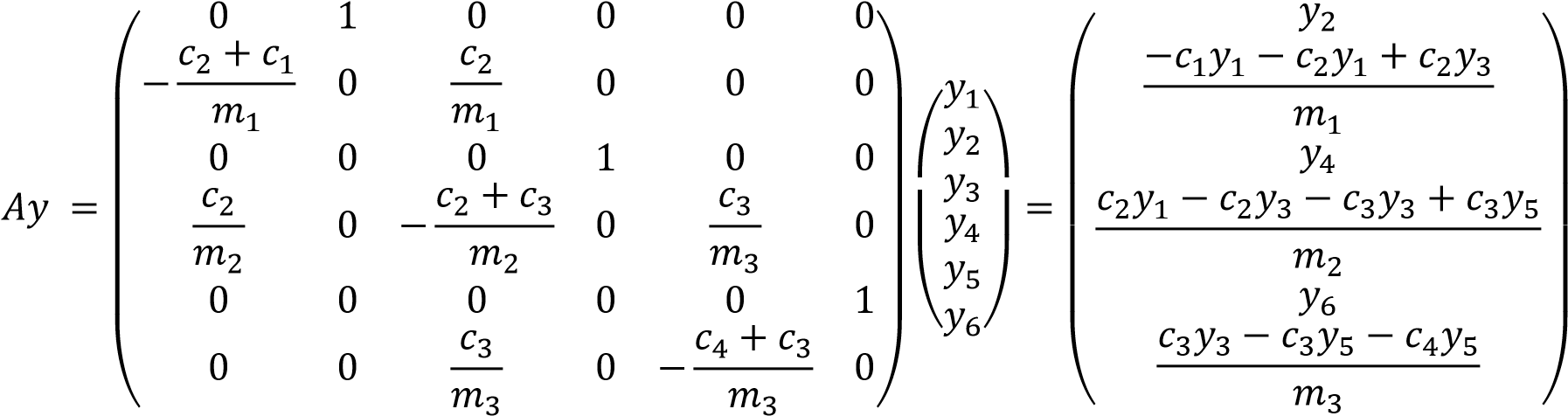
Задаємо вхідні дані:



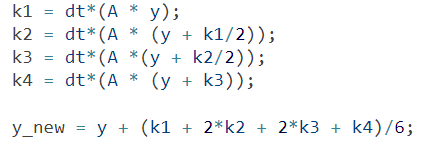
Зчитуємо спостереження з файлу:



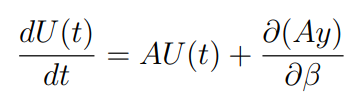
Задаємо матрицю, утворену функцією чутливості:

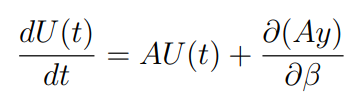


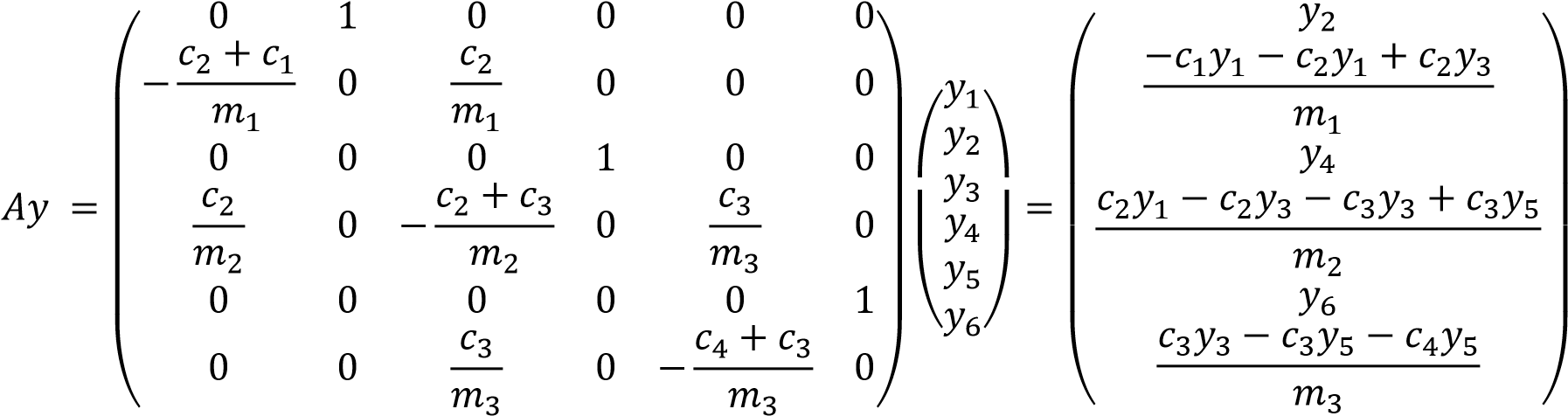
Знаходимо вектор y(t, β0) шляхом розв’язання системи (5.1) - (5.3) методом Рунге-Кутти.

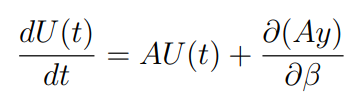


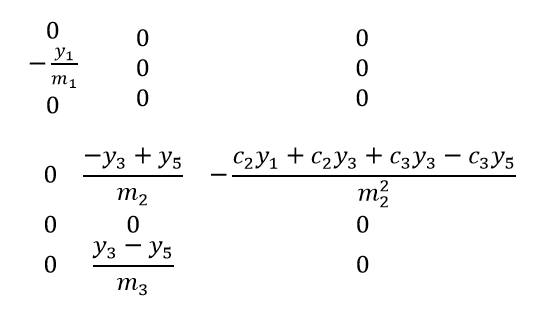
Пiдставляємо y(t, β0) в рiвняння чутливостi 5.7.



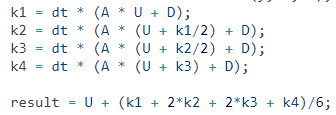
Знаходимо . Добуток матриці A на y має вигляд:



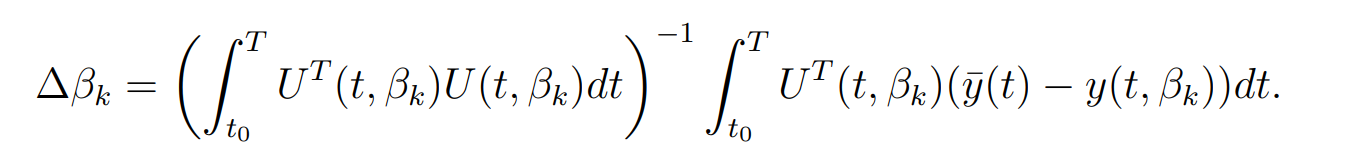
Вектор  за 8 варіантом, тому  матиме вигляд :

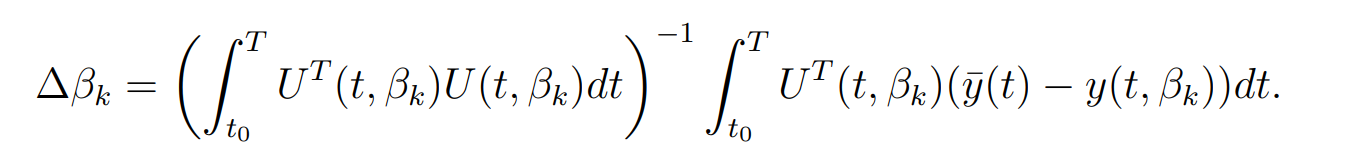


За методом Рунге-Кутти знаходимо U(t, β0).

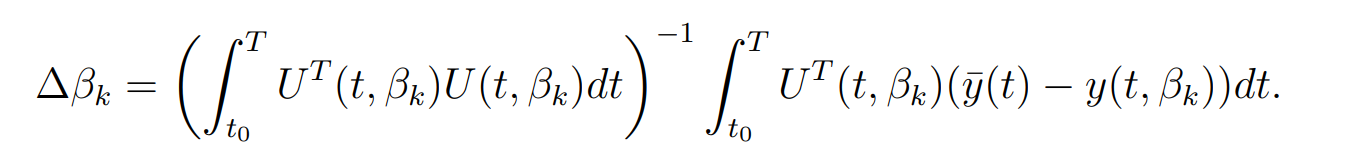


Обраховуємо інтеграли





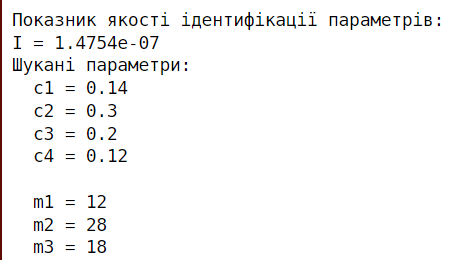
Обраховуємо значення показника якості ідентивікації.

Обраховуємо 

І додаємо значення цього вектора до наближень шуканих параметрів.

Якщо значення показника якості ідентивікації менше за точність зупиняємось, а якщо ні, повторюємо вказаний алгоритм ще раз.

Виводимо результати.



Код програми

#Часовий інтервал

dt = 0.2;

t0 = 0;

tk = 50;

eps = 1e-6;

#Вказуємо відомі параметри або наближення

c = [0.1, 0.3, 0.1, 0.12]';

m = [12, 19, 18]';

# Матриця утворена функцією чутливості

function result = matrixA(m, c)

A = [

0, 1, 0, 0, 0, 0;

-(c(2) + c(1)) / m(1), 0, c(2) / m(1), 0, 0, 0;

0, 0, 0, 1, 0, 0;

c(2) / m(2), 0, -(c(2) + c(3)) / m(2), 0, c(3) / m(2), 0;

0, 0, 0, 0, 0, 1;

0, 0, c(3) / m(3), 0, -(c(4) + c(3)) / m(3), 0

];

result = A;

end

#Метод Рунге-Кутти для функції чутливості

function result = findU (A, U, y, m, c, dt)

D = calculateModelDerivatives(y, m, c);

k1 = dt \* (A \* U + D);

k2 = dt \* (A \* (U + k1/2) + D);

k3 = dt \* (A \* (U + k2/2) + D);

k4 = dt \* (A \* (U + k3) + D);

result = U + (k1 + 2\*k2 + 2\*k3 + k4)/6;

end

#Обчислюємо похідні для моделі

function result = calculateModelDerivatives(y, m, c)

db = zeros(6, 3);

db(2, 1) = -(y(1))/m(1);

db(4, 2) = (y(5)-y(3))/m(2);

db(6, 2) = (y(3)-y(5))/m(3);

db(4, 3) =-(c(2)\*y(1)-c(2)\*y(3)-c(3)\*y(3)+c(3)\*y(5))/m(2)^2;

result = db;

end

#Зчитуємо дані з файлу

data = dlmread('y8.txt');

[size\_m, size\_n] = size(data);

#Задаємо показник якості ідентивікації

I = inf;

i = 1;

while I > eps

y = data(:, 1);

dy = zeros(6, 1);

U = zeros(6, 3);

integral1 = zeros(3, 3);

integral2 = zeros(3, 1);

I = 0.0;

A = matrixA(m, c);

for j = 2:size\_n

#Метод Рунге-Кутти для знаходження y(t, beta0)

k1 = dt\*(A \* y);

k2 = dt\*(A \* (y + k1/2));

k3 = dt\*(A \*(y + k2/2));

k4 = dt\*(A \* (y + k3));

y\_new = y + (k1 + 2\*k2 + 2\*k3 + k4)/6;

U\_new = findU(A, U, y\_new, m, c, dt);

dy\_new = data(:, j) - y\_new;

integral1 = integral1 + dt\*(U'\*U + U\_new'\*U\_new) / 2;

integral2 = integral2 + dt\*(U'\*dy + U\_new'\*dy\_new) / 2;

#Обраховуємо значення показника якості ідентивікації

I = I + dt\*(dy'\*dy + dy\_new'\*dy\_new) / 2;

U = U\_new;

y = y\_new;

dy = dy\_new;

end

dBeta = pinv(integral1) \* integral2;

c(1) = c(1) + dBeta(1);

c(3) = c(3) + dBeta(2);

m(2) = m(2) + dBeta(3);

i = i + 1;

end

#Виводимо значення показника якості ідентивікації на поточному кроці

disp('Показник якостi iдентифiкацiї параметрiв:')

disp(['I = ', num2str(I)])

#Виводимо шукані параметри

disp('Шукані параметри:');

for i = 1:4

fprintf(' c%d = %s\n', i, num2str(c(i)));

end

disp('');

for i = 1:3

fprintf(' m%d = %s\n', i, num2str(m(i)));

end