1. Напишіть цикл for для перевірки функції MonteCarlo\_double та усереднення результатів виконання функції. Візьміть декілька різних n. Якщо відомо, що точна відповідь 3, побудуйте графік залежності похибки розрахунку при різних n.

 Параметри залишаються з лекції, тобто [0, 2]×[3, 4.5] та [0, 3]×[2, 5]

Функція:

function result = MonteCarlo\_double(f, g, x0, x1, y0, y1, n)

 %

 % Monte Carlo integration of f over a domain g>=0, embedded

 % in a rectangle [x0,x1]x[y0,y1]. n^2 is the number of

 % random points.

 % Draw n^2 random points in the rectangle

 x = x0 + (x1 - x0)\*rand(n,1);

 y = y0 + (y1 - y0)\*rand(n,1);

 % Compute sum of f values inside the integration domain

 f\_mean = 0;

 num\_inside = 0; % number of x,y points inside domain (g>=0)

 for i = 1:length(x)

 for j = 1:length(y)

 if g(x(i), y(j)) >= 0

 num\_inside = num\_inside + 1;

 f\_mean = f\_mean + f(x(i), y(j));

 end

 end

 end

 f\_mean = f\_mean/num\_inside;

 area = num\_inside/(n^2)\*(x1 - x0)\*(y1 - y0);

 result = area\*f\_mean;

end

1. Промоделюйте демонстраційну функцію SIR по крокам.

function demo\_SIR()

% Test case using an SIR model

dt = 0.1; % 6 min

D = 30; % Simulate for D days

N\_t = floor(D\*24/dt); % Corresponding no of hours

T = dt\*N\_t; % End time

U\_0 = [50 1 0];

f\_handle = @f;

[u, t] = ode\_FE(f\_handle, U\_0, dt, T);

S = u(:,1);

I = u(:,2);

R = u(:,3);

plot(t, S, ’b-’, t, I, ’r-’, t, R, ’g-’);

legend(’S’, ’I’, ’R’);

xlabel(’hours’);

% Consistency check:

N = S(1) + I(1) + R(1);

eps = 1E-12; % Tolerance for comparing real numbers

for n = 1:length(S)

err = abs(S(n) + I(n) + R(n) - N);

if (err > eps)

error(’demo\_SIR: error=%g’, err);

end

end

end

function result = f(u,t)

beta = 10/(40\*8\*24);

gamma = 3/(15\*24);

S = u(1); I = u(2); R = u(3);

result = [-beta\*S\*I beta\*S\*I - gamma\*I gamma\*I];

end

function [u, t] = ode\_FE(f, U\_0, dt, T)

N\_t = floor(T/dt);

u = zeros(N\_t+1, length(U\_0));

t = linspace(0, N\_t\*dt, length(u));

u(1,:) = U\_0; % Initial values

t(1) = 0;

for n = 1:N\_t

u(n+1,:) = u(n,:) + dt\*f(u(n,:), t(n));

end

end

3. Розглянемо систему рівнянь Лотчі-Вольтерра «хижак-жертва».



Функція  визначає кількість особин здобичі,  – кількість хижаків.

Система рівнянь Лотчі-Вольтерра описує, як зміниться кількість особин в популяціях здобичі і хижака (як будь-яка модель описує тільки в певному наближенні).

 Розглянемо значення коефіцієнтів в цьому рівнянні. Вважається, що при відсутності хижаків популяція здобичей буде збільшуватися. Швидкість цього збільшення задається коефіцієнтом  (береться з плюсом). При цьому популяція хижаків при відсутності видобутку буде зменшуватися, що характеризується коефіцієнтом  (береться з мінусом). Коли ці популяції стикаються, хижаки з'їдають здобич, і ми вважаємо, що кількість з'їдених особин пропорційно добутку . За рахунок поїдання здобичі її популяція буде скорочуватися, на цьому відбивається коефіцієнт  (береться з мінусом). У той же час при нормальному харчуванні популяція хижаків буде збільшуватися, це відбивається на коефіцієнті  (береться з плюсом).

 У нашому випадку будемо вважати, що в початковий момент часу налічувалося 30 особин здобичі і 20 особин хижаків. Коефіцієнти:

;

;

;

.

 Вважаємо, що час t вимірюється в місяцях.

 Побудуйте графіки зміни чисельності особин в популяції здобичі і в популяції хижака протягом 20 місяців по одних і тих же осях. Дайте відповідь на питання (в коментарях), скільки приблизно буде особин в популяціях за 10 місяців? Передбачається, що відповідь можна визначити візуально за графіком, при цьому не потрібно намагатися обчислити точні значення.

 Також побудуйте фазовий портрет системи Диф.рівнянь – .

Рекомендується використовувати розв'язувач ode23s.