|  |  |
| --- | --- |
| Федеральное агентство железнодорожного транспорта  Омский государственный университет путей сообщения  Кафедра «Автоматика и системы управления»  Численные методы решения СИСТЕМ нелинейных уравнений  Лабораторная работа №2  по теме: «Теоретические основы аппаратно-программных средств» | |
|  | Студент гр. 21 м  \_\_\_\_\_\_\_\_\_\_\_\_ К.Н. Юрукина  «\_\_»\_\_\_\_\_\_\_\_\_ 2022 г.  Руководитель – доцент кафедры АиСУ  \_\_\_\_\_\_\_\_\_\_\_\_ А.С. Окишев  «\_\_»\_\_\_\_\_\_\_\_\_ 2022 г. |
| Омск 2022 | |

Лабораторная работа №2

Численные методы решения СИСТЕМ нелинейных уравнений

Цель работы:

Найти приближенные решения системы нелинейных уравнений с помощью метода Ньютона-Рафсона. Изобразить графики в плоскости и пространстве.

Задание 1

Решите систему нелинейных уравнений (вариант 27)

|  |  |
| --- | --- |
|  | 3(2.1) |

методом Ньютона-Рафсона (многомерный вариант метода Ньютона).

Выводим две эти функции на одном графике (рисунок 1), используя программу Matlab.

|  |
| --- |
|  |
| Рисунок 1 – График функций |

Синим цветом обозначена первая функция, красным вторая.

x1=-5:0.1:5;

f1=-((x1+1).^3) + 2;

f2=(1/7)\*x1.^3 + 4;

figure(1);

plot(x1,f1,'b',x1,f2,'r');

axis([-5 5 -10 10]);

xlabel('x1');

ylabel('x2');

grid on;

legend('x2=f1(x1)','x2=f2(x1)');

title ('Графики зависимостей x2=f(x1)');

Листинг 1 – Код графика 1

Изобразим эти функции в пространстве (рисунок 2). В этом случае каждая из функций  геометрически представляет собой поверхность в трехмерном пространстве, а уравнение  описывает кривую, которая получается при сечении этой поверхности плоскостью *z*=0. Точка пересечения двух таких кривых, расположенных в плоскости *z*=0, и будет решением системы уравнений.

Для построения точки начального приближения необходимо выразить из каждого уравнения аргумент *x*2 как функцию первого аргумента *x*1:



|  |
| --- |
|  |
| Рисунок 2 – График функций в пространстве |

На рисунке 2 функции для удобства изображены разными цветами.

Xmin=-5; % минимальная координата по x1 и x2

Xstep=0.25; % шаг сетки графика

Xmax=5; % максимальная координата по x1 и x2

figure(2);

[X1, X2]=meshgrid(Xmin:Xstep:Xmax);

F1=-((X1+1).^3) + 2 - X2;

F2=(1/7)\*X1.^3 + 4 - X2;

Z=zeros(size(X1));

mesh(X1,X2,F1,'EdgeColor','b');

hold on;

mesh(X1,X2,F2,'EdgeColor','y');

mesh(X1,X2,Z,'EdgeColor','g');

axis([-5 5 -5 5 -10 10]);

xlabel('X1'),

ylabel('X2'),

zlabel('F1(X1,X2), F2(X1,X2)');

title('3D-графики поверхностей z=F1(x1,x2), z=F2(x1,x2), z=0');

Листинг 2 – Код 3д-графика

После этого нужно построить с помощью функции plot() на плоскости (т.е. z=0) графики функции  и  в таком масштабе, который позволяет увидеть точку их пересечения (рисунок 3). Ближайшую точку с целочисленными координатами можно взять как точку начального приближения.

Примерный вид кривых  и , точка их пересечения и решение системы с точностью  приведены для самопроверки в правой части задания.

|  |
| --- |
|  |
| Рисунок 3 – График линий уровней |

figure(3);

H=(0:0.5:10); % диапазон и шаг линий уровня

contour(X1,X2,F1,H);

hold on;

contour(X1,X2,F2,H);

xlabel('X1'), ylabel('X2');

title('Линии равных уровней для поверхностей F1 и F2');

grid on;

Листинг 3 – Код графика уровней

Теперь необходимо получить решение системы уравнений с заданной точностью .

Итерационный процесс строится аналогично одномерному методу Ньютона, который рассматривался в лабораторной работе №1, но расчетные формулы теперь нужно записать в векторной форме.

Задается начальное приближение – вектор *x*, требуемая точность  (скалярная величина) и максимальное число итераций  для предотвращения зацикливания метода в случаях, когда итерационный процесс расходится (удаляется от корня). Переменной-счетчику итераций присваивается значение *k* = 0.

На каждой итерации уточненное значение корня вычисляется по формуле:

,

где  – вектор-столбец нового приближения к корню;

– вектор-столбец предыдущего приближения к корню;

 – значение вектор-функции, описывающей систему уравнений, на *k*-ой итерации;

 – матрица первых производных вектор-функции (матрица Якоби).

В итоге найдена точка x (-1,9686; 2,9096) за 8 итераций (рисунок 4).

|  |
| --- |
|  |
| Рисунок 4 – Результат программы |

eps=0.001; % заданная точность

kmax=50; % максимальное число итераций

X=[-1; -1]; % начальное приближение

[F, J]=fun(X);

[Xnew,D]=Newton(X,F,J);

k=1; % счетчик итераций

while (max(abs(D)>eps))&&(k<kmax)

X=Xnew;

[F, J]=fun(X);

[Xnew,D]=Newton(X,F,J);

k=k+1;

disp(Xnew);

end

disp('Root:');

disp(Xnew);

disp('Number of iterations:');

disp(k);

function [F, J] = fun(X)

% Определяем размерность переменных F и J

n=size(X,1);

F=zeros(n,1);

J=zeros(n,n);

% Вектор-функция F = [F1;F2] левых частей уравнений системы

F(1)=-((X(1)+1).^3) + 2 - X(2);

F(2)=(1/7)\*X(1).^3 + 4 - X(2);

% Матрица первых производных (Якобиан)

J(1,1)=-3\*X(1).^2-6\*X(1)-3; % dF1/dX1

J(1,2)=-1; % dF1/dX2

J(2,1)=(3/7)\*X(1).^2; % dF2/dX1

J(2,2)=-1; % dF2/dX2

end

Листинг 4 – Метод Ньютона-Рафсона

Вывод

Таким образом, в ходе выполнения лабораторной работы был найден приближенное решение системы нелинейных уравнений с помощью метода Ньютона-Рафсона и изображены графики в плоскости и пространстве.