|  |  |
| --- | --- |
| Федеральное агентство железнодорожного транспорта  Омский государственный университет путей сообщения  Кафедра «Автоматика и системы управления»  Численные методы решения СЛАУ  Лабораторная работа №3  по теме: «Теоретические основы аппаратно-программных средств» | |
|  | Студент гр. 21 м  \_\_\_\_\_\_\_\_\_\_\_\_ К.Н. Юрукина  «\_\_»\_\_\_\_\_\_\_\_\_ 2022 г.  Руководитель – доцент кафедры АиСУ  \_\_\_\_\_\_\_\_\_\_\_\_ А.Н. Смалев  «\_\_»\_\_\_\_\_\_\_\_\_ 2022 г. |
| Омск 2022 | |

Лабораторная работа №3

Численные методы решения СЛАУ

Цель работы:

Решить систему нелинейных алгебраических уравнений методом простых итераций, с помощью операции обращения матрицы, методом Гаусса с выбором ведущего элемента по столбцу.

Задание 1

Решить систему (3.1) с помощью операции обращения матрицы (вариант 27):

|  |  |
| --- | --- |
|  | 3(3.1) |

Находим корни векторным произведением транспонированной матрицы коэффициентов и матрицы правых частей уравнения (рисунок 1).

|  |
| --- |
|  |
| Рисунок 1 – Корни уравнения |

Решением уравнения являются корни -0,8003, 6,9819, 8,0127, которые совпадают с заранее известными корнями.

Погрешность составляет 3,6380е-12 – самый максимальный модуль.

clc;

A = [700 -8 7; 3.5 1400 6.5; -3 3 2100];

b = [-560; 9824; 16850];

x = A\b;

disp("Корни:");

disp(x);

eps = max(abs(b - A\*x));

disp("Погрешность:");

disp(eps);

Листинг 1 – Операция обращения матрицы

Задание 2.

Решить СЛАУ (3.1) методом Гаусса с выбором ведущего элемента по столбцу.

Алгоритм поиска решения методом Гаусса состоит из двух этапов:

1) прямой ход – преобразование расширенной матрицы СЛАУ к верхней треугольной матрице, т. е. зануление всех элементов ниже главной диагонали и получение единиц на главной диагонали;

2) обратный ход – преобразование верхней треугольной матрицы к диагональной, в которой все элементы, кроме главной диагонали, равны нулю. В этом случае правый столбец расширенной матрицы и будет решением СЛАУ.

Кроме прямого и обратного хода метода Гаусса, необходимо реализовать проверку системы на наличие решений () и перестановку строк матрицы с выбором ведущего элемента.

На каждом шаге прямого хода метода Гаусса выбирается ведущая строка, которая делится на ведущий элемент и далее вычитается из всех нижележащих строк системы для зануления элементов в столбце под ведущим элементом. Перед выполнением операции деления необходимо убедиться в том, что ведущий элемент не равен нулю, иначе требуется переставить строки матрицы (рисунки 2-4).

|  |
| --- |
|  |
| Рисунок 2 – Метод Гаусса (Прямой ход 1) |
|  |
| Рисунок 3 – Метод Гаусса (Прямой ход 2) |
|  |
| Рисунок 4 – Метод Гаусса (Обратный ход) |

Решением уравнения являются -0,8003, 6,9819, 8,0127, которые совпадают с корнями из первого задания.

Погрешность составляет 3,6380е-12, которая тоже совпадает с погрешностью из первого задания.

A0 = [700 -8 7; 3.5 1400 6.5; -3 3 2100];

b = [-560;9824;16850];

eps = 0.001;

if det(A0) == 0

disp('Решений нет!');

else

A = [A0 b];

disp('Расширенная матрица:');

disp(A);

n = 3;

i = 1;

disp('Прямой ход:');

for k = 1:n

if abs(A(k,k)) < eps

[A] = fun\_1(A, k);

fprintf('k=%i i=%i j=%i\n',k,i,j);

disp(A);

end

for j = (n+1):(-1):k

A(k,j) = A(k,j)/A(k,k);

fprintf('k=%i i=%i j=%i\n',k,i,j);

disp(A);

end

for i = (k+1):n

for j = (n+1):-1:k

A(i,j) = A(i,j) - A(i,k)\*A(k,j);

end

fprintf('k=%i i=%i j=%i\n',k,i,j);

disp(A);

end

end

disp('Обратный ход:');

for i = (n):-1:1

X(i) = A(i,n+1);

for j = (i+1):n

X(i) = X(i) - A(i,j)\*X(j);

end

disp(X);

end

D = max(max(abs(b - A0\*X')));

disp('MAX L-норма:');

disp(D);

end

function [A] = fun\_1(A, k)

t = A (k);

A(k) = A(k+1);

A(k+1) = t;

End

Листинг 2 – Первая версия кода метода Гаусса

Вторая часть подразумевает выбор ведущего элемента, как самого большого снизу находящегося. Поэтому код примет немного иной вид:

A0 = [700 -8 7; 3.5 1400 6.5; -3 3 2100];

b = [-560;9824;16850];

eps = 0.001;

if det(A0) == 0

disp('Решений нет!');

else

A = [A0 b];

disp('Расширенная матрица:');

disp(A);

n = 3;

disp('Прямой ход:');

for k = 1:n

[m, t] = max(abs(A(k:n,k)));

h = k + t - 1;

if h ~= k

[A] = fun\_1(A, k, i);

fprintf('k=%i i=%i j=%i\n',k,i,j);

disp(A);

end

for j = (n+1):(-1):k

A(k,j) = A(k,j)/A(k,k);

fprintf('k=%i i=%i j=%i\n',k,i,j);

disp(A);

end

for i = (k+1):n

for j = (n+1):-1:k

A(i,j) = A(i,j) - A(i,k)\*A(k,j);

end

fprintf('k=%i i=%i j=%i\n',k,i,j);

disp(A);

end

end

disp('Обратный ход:');

for i = (n):-1:1

X(i) = A(i,n+1);

for j = (i+1):n

X(i) = X(i) - A(i,j)\*X(j);

end

disp(X);

end

D = max(max(abs(b - A0\*X')));

disp('MAX L-норма:');

disp(D);

end

function [A] = fun\_1(A, k, h)

t = A (k);

A(k) = A(h);

A(h) = t;

end

Листинг 3 – Вторая версия кода метода Гаусса

Результат от этих изменений никак не поменялся.

Задание 3.

Решить СЛАУ (3.1) методом простых итераций с точностью .

Для этого задания понадобятся формулы приближения:

|  |  |  |
| --- | --- | --- |
|  | , | (3.2) |
| где | *x* – предыдущее (менее точное) приближение к решению;  *x1* – новое (более точное) приближение к решению;  – преобразованный вектор правых частей СЛАУ, элементы которого определяются по формуле:  ;  – преобразованная матрица коэффициентов СЛАУ, элементы которой определяются по формуле:  . | |

Используя эти формулы находятся корни уравнения (рисунок 5).

|  |
| --- |
|  |
| Рисунок 5 – Корни уравнения, найденные методом простых итераций |

Всего за 2 итерации корни нашлись, они такие же, как и в прошлых заданиях, однако погрешность отличается. Она составляет 2,9122е-04, что намного меньше прошлых результатов.

A0 = [700 -8 7; 3.5 1400 6.5; -3 3 2100];

b0 = [-560;9824;16850];

A = [A0 b0];

a = A0;

b = [0 0 0];

n = 3;

for i = 1:n

for j = 1:n

if i == j

a(i,j) = 0;

else

a(i,j) = -A0(i,j)/A0(i,i);

end

end

b(i) = b0(i)/A0(i,i);

end

disp(a);

disp(b);

eps = 0.001;

k=1;

kmax=100;

x0 = b;

disp(x0);

x1 = a\*x0'+b';

disp(x1);

D = max(abs(x1'-x0));

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

x0 = x1;

x1 = a\*x0+b';

k=k+1;

D = max(abs(x1-x0));

end

x = x1;

disp('Корни:');

disp(x);

disp('Количество итераций:');

disp(k);

disp('Погрешность:');

disp(D);

Листинг 4 – Код метода простых итераций

Вывод

Таким образом, была решена система нелинейных алгебраических уравнений методом простых итераций, с помощью операции обращения матрицы, методом Гаусса с выбором ведущего элемента по столбцу. Результаты корней получились одинаковыми, однако погрешность сильно отличалась в одном из случаев. Все методы достигли поставленной задачи.