1. Дана система линейных уравнений

Перепишем систему в виде

где

, ,

Проверим выполняется ли условие диагонального преобладания для матрицы А. Для этого напишем простую функцию в Matlab

IsDiagDom = @(A)all( 2\*abs(diag(A))>sum(abs(A),2));

A = [1.7 -2.2 3.0;

2.1 1.9 -2.3;

4.2 3.9 -3.1];

% Проверяем условие диагонального преобладания

IsDiagDom( A )

Результат выполнения IsDiagDom равен 0, т.е. «ЛОЖЬ»

Преобразуем матрицу A к виду

Здесь

B(1,:) = A(1,:)+A(2,:)

B(2,:) = A(1,:)+3\*A(2,:)-2\*A(3,:)

B(3,:) = 3\*A(1,:)-A(3,:)

Или то же самое

, где

B = [ 3.80 -0.30 0.70;

-0.40 -4.30 2.30;

0.90 -10.50 12.10];

IsDiagDom(B)

Результат 1, т.е «ИСТИНА». Матрица В обладает свойством диагонального преобразования.

Тогда задача преобразуется в виду

где

2. Вычислим нормы

Код в MATLAB

Bnorm1 = max(sum(abs(B)))

BnormInf = max(sum(abs(B')))

Результат

Bnorm1 = 15.1

BnormInf = 23.5

Так как первая норма меньше выбираем ее.

Разрешим систему относительно диагональных элементов, тогда система преобразуется к эквивалентному виду

где

Код в MATLAB

% Система

% Bx=d

% преобразуется к эквивалентному виду

% x=alpha\*x+beta

for i=1:3

alpha(i,1:3) = B(i,1:3)/B(i,i);

beta(i) = d(i)/B(i,i);

end

alpha = eye(3)-alpha;

3. Исползуя априорную оценку вычислим число итераций

Если , то

betanorm = sum(abs(beta));

alphanorm = max(sum(abs(alpha)));

e = 0.1;

k\_apriori = (log(e)-log(betanorm)+log(1-alphanorm))/log(alphanorm)-1;

4. Выберем начальное приближене

Запустим итерационный процесс с остановкой по апостериорному критерию

Код в Matlab

% Выбираем начальное приближение

x(1:3,1) = beta;

for k=2:k\_apriori

x(:,k) = alpha\*x(:,k-1)+beta;

if sum(abs(x(:,k)-x(:,k-1)))<=e\*(1-alphanorm)/alphanorm

fprintf('Решение системы [%f; %f; %f]\n',x(:,end))

fprintf('Фактическое число итераций %d\n', k-1)

break

end

end

Фактически получаем 12 итераций и решение

8. Вычислим вектор невязки

9. Листинг скрипта Matlab

clc

clear all

A = [1.7 -2.2 3.0;

2.1 1.9 -2.3;

4.2 3.9 -3.1];

b = [1.8; 2.8; 5.1];

% Проверяем условие диагонального преобладания

IsDiagDom = @(A)all( 2\*abs(diag(A))>sum(abs(A),2));

IsDiagDom(A)

% Преобразуем матрицу А

C = [ 1 1 0;

1 3 -2;

3 0 -1];

B = C\*A;

d = C\*b;

IsDiagDom(B)

Bnorm1 = max(sum(abs(B)))

BnormInf = max(sum(abs(B')))

% Система

% Bx=d

% преобразуется к эквивалентному виду

% x=alpha\*x+beta

for i=1:3

alpha(i,1:3) = B(i,1:3)/B(i,i);

beta(i,1) = d(i)/B(i,i);

end

alpha = eye(3)-alpha;

betanorm = sum(abs(beta));

alphanorm = max(sum(abs(alpha)));

e = 0.1;

k\_apriori = (log(e)-log(betanorm)+log(1-alphanorm))/log(alphanorm)-1;

% Выбираем начальное приближение

x(1:3,1) = beta;

for k=2:k\_apriori

x(:,k) = alpha\*x(:,k-1)+beta;

if sum(abs(x(:,k)-x(:,k-1)))<=e\*(1-alphanorm)/alphanorm

fprintf('Решение системы [%f; %f; %f]\n',x(:,end))

fprintf('Фактическое число итераций %d\n', k-1)

break

end

end

% Вектор невязки

f=A\*x(:,end)-b

sum(abs(f))