



**《数值代数》实验报告**

**学 院**：

**班 级**：

**学 号**：

**姓 名**：

**指导教师**：

教 务 处

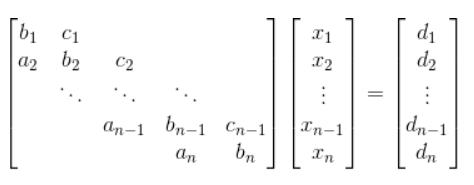
2021 年 12 月

**实验一、追赶法**

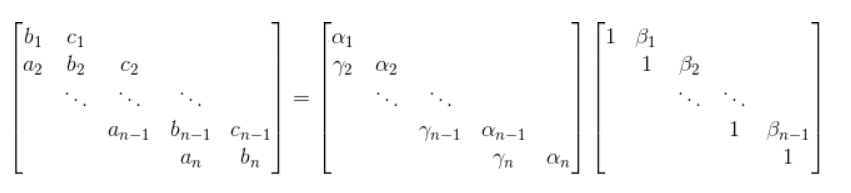
一、实验目的

追赶法算法主要针对如下的三对角矩阵进行LU分解后进行求解。

三对角矩阵的形式如下：



需要将三对角阵分解成如下形式:



求解Ax=d等价于求解Ly=d和Ux=y

其中计算的公式如下：



因而经过推导得到的解三对角线性方程组的追赶法公式如下：

1. 计算的递推公式：



1. 解Ly=d:



1. 解Ux=y:



使用追赶法求解线性方程组。通过代码实现追赶法，理解追赶法的迭代过程，对于该法的迭代过程有更深刻的理解。

二、实验内容

题目：

用追赶法求解线性方程组：



代码：

n=4;

a=[0 -1 -1 -1];

b=[5 5 5 5];

c=[-1 -1 -1 0];

x=[1 2 3 4];%存放常数项

s=zeros(n,1);

t=s;

temp=0;

for k=1:n

s(k)=b(k)-a(k)\*temp; %temp相对于t[k-1],t[-1]看做0

t(k)=c(k)/s(k);

temp=t(k);

end

temp=0;

for k=1:n

x(k)=(x(k)-a(k)\*temp)/s(k); %x[k]中存放的是y[k]

temp=x(k); %temp相对于y[k-1],y[-1]看做0

end

for k=n-1:-1:1

x(k)=x(k)-t(k)\*x(k+1);

end

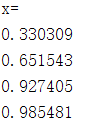
fprintf('\nx=\n');

for k=1:n

fprintf('%f\n',x(k));

end

结果：



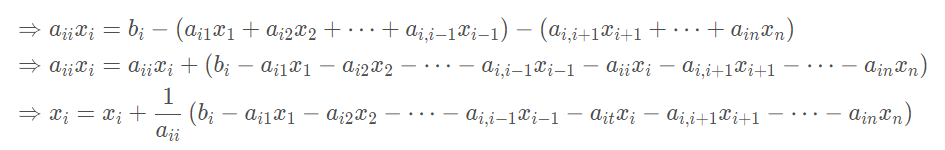
**实验二、SOR迭代法**

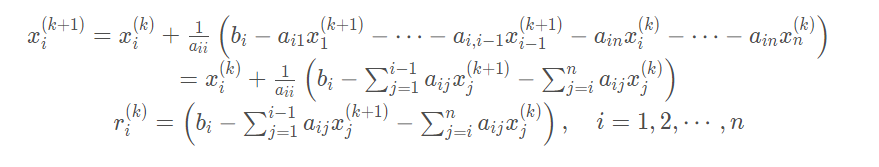
一、实验目的

将方程组Ax=b写成

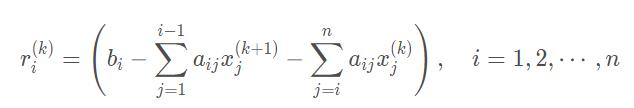


则：

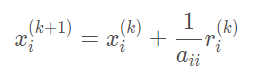
于是Gauss-Seidel迭代法可写成（）



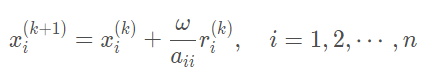
记：

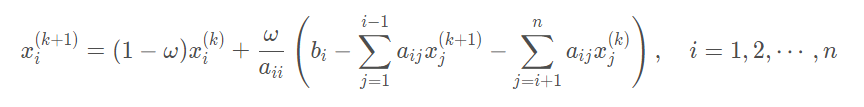


则：

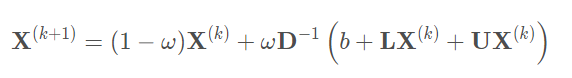


可以看到，Gauss-Seidel迭代法的第k+1步的基础上每个分量上加上。为了获得更快的收敛效果，在修正项的前面乘以一个参数w，于是得到逐次超松弛算法：

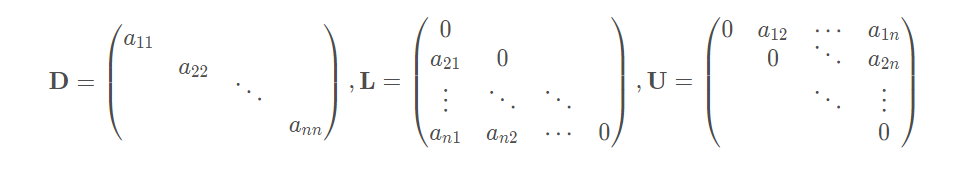


既 

其矩阵形式为：



其中，A=D-L-U



使用SOR迭代法求解线性方程组。通过代码实现SOR迭代法，理解SOR迭代法的迭代过程，对于该法的迭代过程有更深刻的理解。

二、实验内容（题目、代码、结果）

题目：

用熟悉的计算机语言编写SOR迭代法通用程序，并求解下列方程组。



取，

代码：

A=[-4 1 1 1;1 -4 1 1;1 1 -4 1;1 1 1 -4];

b=[1 1 1 1]';

x0=[0 0 0 0]';

times=9;

w=1.3;

[M, x]=SOR(A,b,x0,times,w);

function [M,x]=SOR(A,b,x0,times,w)

%[M,x]=SOR(A,b,x0,times,w)

%

%M为jacobi迭代矩阵

%x为方程组的解

%A为系数矩阵

%b为矩阵右侧

%times为迭代次数

%x0为初始x0

%w为参数因子

D=diag(diag(A));

L=D-tril(A);

U=D-triu(A);

M=(D-w.\*L)\((1-w).\*D+w.\*U);

if vrho(M)>=1 %求B的谱

error('jacobi迭代法不收敛！');

end

i=1;

while i<=times

x=M\*x0+(D-w.\*L)\(w.\*b); %来自书上

x0=x;

disp(['第',num2str(i),'次']);

disp(x);

i=i+1;

end

end

结果：

第9次

-1.0000

-1.0000

-1.0000

-1.0000

**实验三、幂法**

一、实验目的

幂法的数学迭代公式为：



使用幂法求解线性方程组。通过代码实现幂法，理解幂法的迭代过程，对于该法的迭代过程有更深刻的理解。

以此类推即可。

二、实验内容（题目、代码、结果）

题目：

用熟悉的计算机语言编写幂法的通用程序，并求下列矩阵的主特征值和相应的特征向量。



代码：

clear;clc;

all long;

A = [1,1,0.5;1,1,0.25;0.5,0.25,2];

e = 10;

u(:,1) = [1 1 1]';

v(:,1) = u(:,1);

i = 1;

while e>1e-10

v(:,i+1) = A\*u(:,i);

u(:,i+1) = v(:,i+1)/max(v(:,i+1));

e = abs(max(v(:,i+1))-max(v(:,i)));

i = i+1;

if i>100

break;

end

end

u(:,i)

max(v(:,i))

结果：

ans =

0.7482

0.6497

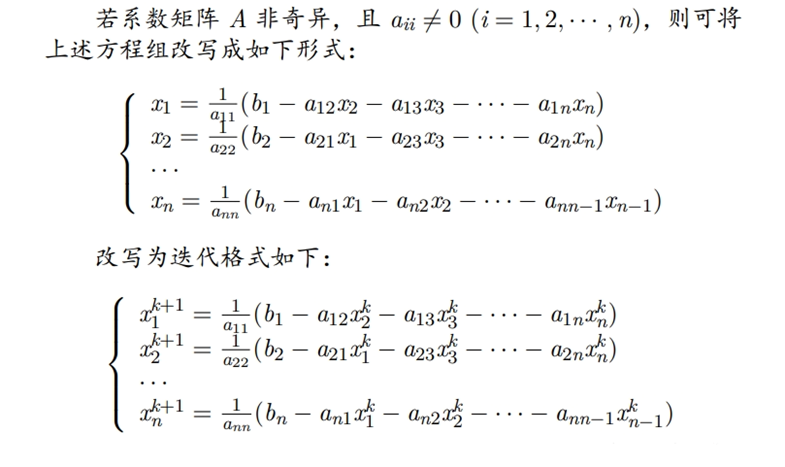
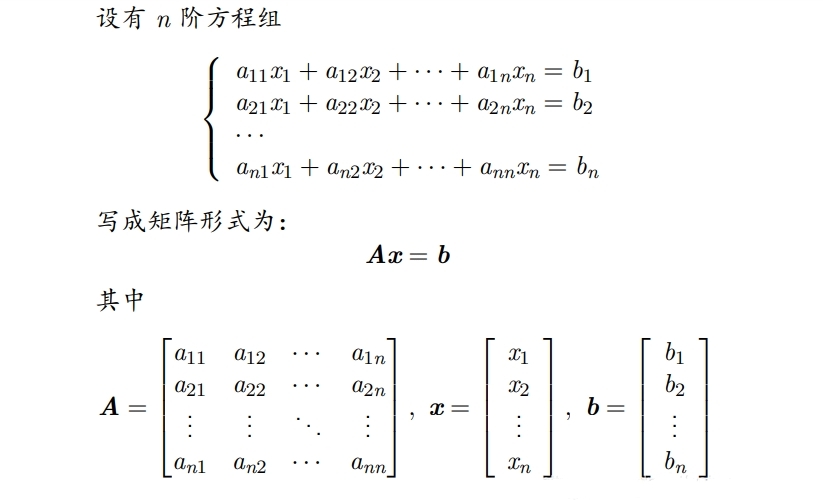
1.0000

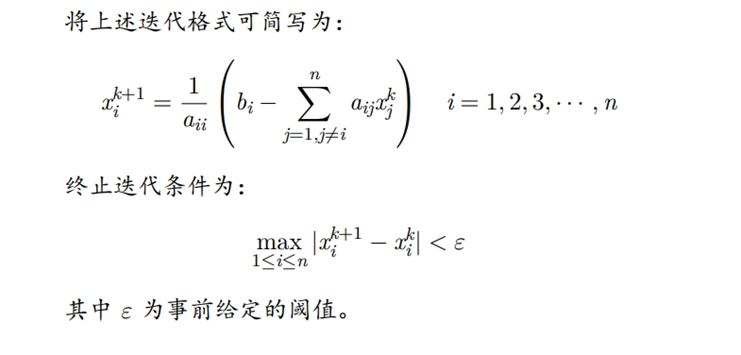
ans =

2.5365

**实验四、Jacobi方法**

一、实验目的





使用Jacobi迭代法求解线性方程组。通过代码实现Jacobi迭代法，理解 Jacobi迭代法的迭代过程，对于该法的收敛性部分理论有更深刻的理解。

二、实验内容

题目：

用熟悉的计算机语言编写Jacobi方法通用程序，并求下列实对称矩阵的全部特征值。



代码：

function [D,V]=jacobi(A)

%检验输入是否合法

b=size(A);

if b(1)~=b(2) %行列不等

error('MATLAB:Jaco:Invalid Matrix,The Matrix input should be a Symmetry Phalanx. See Jaco.');

end

n=max(b);

for i=1:n %非对称

for j=1:n

if abs(A(i,j)-A(j,i))>eps %不能用不等号，因为机器有误差

error('MATLAB:Jaco:Invalid Phalanx,The Phalanx input should be a Symmetric one. See Jaco.');

end

end

end

%实际计算

%初始化D，V为单位矩阵，避免多次分配空间

%并且相乘不会造成影响

D=eye(n);

V=eye(n);

%采用扫描绝对值最大A(p,q)的算法

p=0; %储存最大元素所在行

q=0; %储存最大元素所在列

maxpq=0; %储存绝对值最大元素

for i=1:n-1

for j=i+1:n

if abs(A(i,j))>abs(maxpq)

maxpq=A(i,j);

p=i;

q=j;

end

end

end

while abs(maxpq)>eps

maxpq=0;%务必清零，否则会死循环

phi=atan2(2\*A(p,q),A(p,p)-A(q,q))/2;

U=eye(n);

U(p,p)=cos(phi);

U(q,q)=cos(phi);

U(p,q)=-sin(phi);

U(q,p)=sin(phi);

D=U'\*A\*U;

V=V\*U;

A=D;

%改写maxpq

for i=1:n-1

for j=i+1:n

if abs(A(i,j))>abs(maxpq)

maxpq=A(i,j);

p=i;

q=j;

end

end

end

end

end

A=[4,2,2;2,5,1;2,1,6];

[D,V]=jacobi(A)

结果：

D =

8.3876 0.0000 0

-0.0000 4.4865 -0.0000

0.0000 0.0000 2.1259

V =

0.5387 0.1555 -0.8280

0.5149 0.7172 0.4697

0.6669 -0.6793 0.3062