2 差分方法

2.1 一维差分法

考虑两点边值问题(常系数):

$$-\frac{d^2u}{dx^2}+\frac{du}{dx}+u=f(x),\quad x\in[a,b]$$

数值格式:

$$-\frac{u(x_{i+1})-2u(x_i)+u(x_{i-1})}{h^2}+\frac{u(x_{i+1})-u(x_{i-1})}{h}+u(x_i)=f(x_i), i=1,\cdots,N-1.$$

例 2.1

$$\begin{cases} -\dfrac{d^2u}{dx^2}+\dfrac{du}{dx}=\pi^2\sin(\pi x)+\pi\cos(\pi x), \quad x\in[0,1] \\ u(0)=0, u(1)=0. \end{cases}$$

真解: $u(x)=\sin(\pi x)$.

```
% fdm1d1.m
% finite difference method for 1D problem
% -u''+u'=pi^2*sin(pi*x)+pi*cos(pi*x)  in [0,1]
% u(0)=0, u(1)=0 ;
% exact solution : u=sin(pi*x)
clear all
h=0.05;
x=0:h:1;
N=length(x)-1;
A=diag((2/h^2)*ones(N-1,1))...
    +diag((1/(2*h)-1/h^2)*ones(N-2,1),1)...
    +diag((-1/(2*h)-1/h^2)*ones(N-2,1),-1);
b=pi^2*sin(pi*x(2:N))+pi*cos(pi*x(2:N));
u=A\b';
u=[0;u;0];
ue=sin(pi*x)';
plot(x,ue,'b-',x,u,'r+','LineWidth',1.5)
Error=max(abs(u-ue))
xlabel('x','fontsize', 16), ylabel('y','fontsize',16,'Rotation',0)
legend('Exact ','Numerical','location','North')
title('Finite Difference Method','fontsize',14)
set(gca,'fontsize',14)
```

考虑两点边值问题(变系数):

$$-\frac{d}{dx}(p\frac{du}{dx})+r\frac{du}{dx}+qu=f(x),\quad x\in(a,b)$$

数值格式:

$$-\frac{2}{h_i+h_{i+1}}\left[p_{i+\frac{1}{2}}\frac{u(x_{i+1})-u(x_i)}{h_{i+1}}+p_{i-\frac{1}{2}}\frac{u(x_i)-u(x_{i-1})}{h_i}\right]+$$
$$\frac{r_i}{h_i+h_{i+1}}(u(x_{i+1})-u(x_{i-1}))+q_iu(x_i)=f(x_i),i=1,\cdots,N-1.$$

例 2.2

$$\begin{cases}-\dfrac{d}{dx}\left(x\dfrac{du}{dx}\right)+x\dfrac{du}{dx}=\pi^2x\sin(\pi x)+\pi(x-1)\cos(\pi x),x\in(0,1)\\ u(0)=0,u(1)=0.\end{cases}$$

真解: $u(x)=\sin(\pi x)$.

```matlab
% fdm1d2.m
% finite difference method for 1D problem
% -(xu')'+x*u'=pi^2*x*sin(pi*x)-pi*cos(pi*x)+pi*x*cos(pi*x) in [0,1]
% u(0)=0, u(1)=0 ;
% exact solution : u=sin(pi*x)
clear all
h=0.05;
x=0:h:1;
N=length(x)-1;
A=diag(2*x(2:N)./h^2)+diag(x(2:N-1)./(2*h)-(x(2:N-1)+0.5*h)./h^2,1)...
    +diag(-x(3:N)./(2*h)-(x(3:N)-0.5*h)./h^2,-1);
b=pi^2*x(2:N).*sin(pi*x(2:N))+pi*(x(2:N)-1).*cos(pi*x(2:N));
u=A\b';
u=[0;u;0];
ue=sin(pi*x');
plot(x,ue,'b-',x,u,'r+','LineWidth',1.5)
Error=max(abs(u-ue))
xlabel('x','fontsize', 16), ylabel('y','fontsize',16,'Rotation',0)
legend('Exact ','Numerical','location','North')
title('Finite Difference Method','fontsize',14)
set(gca,'fontsize',14)
```

2.2 二维差分法

考虑二维泊松问题：

$$\begin{cases} -\Delta u = f(x,y), \quad (x,y) \in \Omega \\ u|_{\partial\Omega} = \phi(x,y), \quad (x,y) \in \partial\Omega. \end{cases}$$

离散格式：

$$-\frac{1}{h_2^2}u_{i,j-1} - \frac{1}{h_1^2}u_{i-1,j} + 2\left(\frac{1}{h_1^2} + \frac{1}{h_2^2}\right)u_{i,j} - \frac{1}{h_1^2}u_{i+1,j} - \frac{1}{h_2^2}u_{i,j+1} = f\left(x_i, y_j\right),$$
$$1 \leqslant i \leqslant N-1, 1 \leqslant j \leqslant M-1.$$

定义向量: $\boldsymbol{u}_j = (u_{1j}, u_{2j}, \ldots, u_{N-1,j})^{\mathrm{T}}, \quad 0 \leqslant j \leqslant M.$

离散格式写成矩阵形式：

$$\boldsymbol{D}\boldsymbol{u}_{j-1} + \boldsymbol{C}\boldsymbol{u}_j + \boldsymbol{D}\boldsymbol{u}_{j+1} = \boldsymbol{f}_j, \quad 1 \leqslant j \leqslant M-1.$$

$$\boldsymbol{C} = \begin{pmatrix} 2\left(\frac{1}{h_1^2}+\frac{1}{h_2^2}\right) & -\frac{1}{h_1^2} & & & \\ -\frac{1}{h_1^2} & 2\left(\frac{1}{h_1^2}+\frac{1}{h_2^2}\right) & -\frac{1}{h_1^2} & & \\ & \ddots & \ddots & \ddots & \\ & & -\frac{1}{h_1^2} & 2\left(\frac{1}{h_1^2}+\frac{1}{h_2^2}\right) & -\frac{1}{h_1^2} \\ & & & -\frac{1}{h_1^2} & 2\left(\frac{1}{h_1^2}+\frac{1}{h_2^2}\right) \end{pmatrix}$$

$$\boldsymbol{D} = \begin{pmatrix} -\frac{1}{h_2^2} & & & & \\ & -\frac{1}{h_2^2} & & & \\ & & \vdots & & \\ & & & -\frac{1}{h_2^2} & \\ & & & & -\frac{1}{h_2^2} \end{pmatrix} \qquad \boldsymbol{f}_j = \begin{pmatrix} f(x_1,y_j) + \frac{1}{h_1^2}\phi(x_0,y_j) \\ f(x_2,y_j) \\ \vdots \\ f(x_{N-2},y_j) \\ f(x_{N-1},y_j) + \frac{1}{h_1^2}\phi(x_N,y_j) \end{pmatrix}$$

进一步写成如下矩阵形式：

$$\begin{pmatrix} \boldsymbol{C} & \boldsymbol{D} & & & \\ \boldsymbol{D} & \boldsymbol{C} & \boldsymbol{D} & & \\ & \ddots & \ddots & \ddots & \\ & & \boldsymbol{D} & \boldsymbol{C} & \boldsymbol{D} \\ & & & \boldsymbol{D} & \boldsymbol{C} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_1 \\ \boldsymbol{u}_1 \\ \vdots \\ \boldsymbol{u}_{M-2} \\ \boldsymbol{u}_{M-1} \end{pmatrix} = \begin{pmatrix} \boldsymbol{f}_1 - D\boldsymbol{u}_0 \\ \boldsymbol{f}_1 \\ \vdots \\ \boldsymbol{f}_{M-2} \\ \boldsymbol{f}_{M-1} - D\boldsymbol{u}_N \end{pmatrix}$$

例 2.3

$$\begin{cases} -\Delta u = f(x,y), \quad (x,y) \in \Omega = (0,1) \times (0,1) \\ u = 0, (x,y) \in \partial\Omega. \end{cases}$$

其中:$f(x,y) = -2\pi^2 e^{\pi(x+y)}(\sin\pi x\cos\pi y + \cos\pi x\sin\pi y)$.

真解:$u(x,y) = e^{\pi(x+y)}\sin\pi x\sin\pi y, \quad (x,y) \in \Omega = (0,1) \times (0,1)$.

```
% fdm2d1.m
% finite difference method for 2D problem
% -d^2u/dx^2-d^2u/dy^2=f(x,y)
% f(x,y)=-2*pi^2*exp(pi*(x+y))*(sin(pi*x)*cos(pi*y)+cos(pi*x)*sin(pi*y))
% exact solution: ue=exp(pi*x+pi*y)*sin(pi*x)*sin(pi*y)
clear all
h=0.01;
x=[0:h:1]';
y=[0:h:1]';
N=length(x)-1;
M=length(y)-1;
[X,Y]=meshgrid(x,y);
X=X(2:M,2:N);
Y=Y(2:M,2:N);
% generate the matrix of RHS
f=-2*pi^2*exp(pi*X+pi*Y).*(sin(pi*X).*cos(pi*Y)+cos(pi*X).*sin(pi*Y));
% constructing the coefficient matrix
C=4/h^2*eye(N-1)-1/h^2*diag(ones(N-2,1),1)-1/h^2*diag(ones(N-2,1),-1);
D=-1/h^2*eye(N-1);
A=kron(eye(M-1),C)+kron(diag(ones(M-2,1),1)+diag(ones(M-2,1),-1),D);
% solving the linear system
f=f';
u=zeros(M+1,N+1);
u(2:M,2:N)=reshape(A\f(:),N-1,M-1)';
u(:,1)=0;
u(:,end)=0;
ue=zeros(M+1,N+1);
ue(2:M,2:N)=exp(pi*X+pi*Y).*sin(pi*X).*sin(pi*Y);
% compute maximum error
Error=max(max(abs(u-ue)))
mesh(x,y,u)
xlabel('x','fontsize', 16), ylabel('y','fontsize',16), ...
    zlabel('u','fontsize',16,'Rotation',0)
title('Finite Difference Method','fontsize',14)
set(gca,'fontsize',14)
```

```matlab
% fdm2d1_error.m
% finite difference method for 2D problem
% -d^2u/dx^2-d^2u/dy^2=f(x,y)
% f(x,y)=-2*pi^2*exp(pi*(x+y))*(sin(pi*x)*cos(pi*y)+cos(pi*x)*sin(pi*y))
% exact solution: ue=exp(pi*x+pi*y)*sin(pi*x)*sin(pi*y)
clear all
Nvec=2.^[3:10];  Err=[];
for n=Nvec
   h=1/n;
   x=[0:h:1]';     y=[0:h:1]';
   N=length(x)-1;  M=length(y)-1;
   [X,Y]=meshgrid(x,y);
   X=X(2:M,2:N);
   Y=Y(2:M,2:N);
   % generate the matrix of RHS
   f=-2*pi^2*exp(pi*(X+Y)).*(sin(pi*X).*cos(pi*Y)+cos(pi*X).*sin(pi*Y));
   % constructing the coefficient matrix
   e=ones(N-1,1);
   C=1/h^2*spdiags([-e 4*e -e],[-1 0 1],N-1,N-1);
   D=-1/h^2*eye(N-1);
   e=ones(M-1,1);
   A=kron(eye(M-1),C)+kron(spdiags([e e],[-1 1],M-1,M-1),D);
   % solving the linear system
   f=f';
   u=zeros(M+1,N+1);
   u(2:M,2:N)=reshape(A\f(:),N-1,M-1)';
   u(:,1)=0;
   u(:,end)=0;
   ue=zeros(M+1,N+1);        % numerical solution
   ue(2:M,2:N)=exp(pi*X+pi*Y).*sin(pi*X).*sin(pi*Y);
   err=max(max(abs(u-ue)));  % maximum error
   Err=[Err,err];
end
plot(log10(Nvec),log10(Err),'ro-','MarkerFaceColor','w','LineWidth',1.5)
grid on,hold on, plot(log10(Nvec), log10(Nvec.^(-2)), '--')
xlabel('log_{10}N','fontsize',16),ylabel('log_{10}Error','fontsize',16),
title('Convergence of Finite Difference Method','fontsize',14)
set(gca,'fontsize',14)
for i=1:length(Nvec)-1     % computating convergence order
   order(i)=-log(Err(i)/Err(i+1))/(log(Nvec(i)/Nvec(i+1)));
end
Err
order
```

例 2.4

$$\begin{cases} -\Delta u=\cos 3x\sin \pi y, \quad (x,y)\in G=(0,\pi)\times(0,1) \\ u(x,0)=u(x,1)=0, \quad 0\leqslant x\leqslant \pi, \\ u_x(0,y)=u_x(\pi,y)=0, \quad 0\leqslant y\leqslant 1 \end{cases}$$

真解: $u=(9+\pi^2)^{-1}\cos 3x\sin \pi y.$

以步长$h_1=\frac{\pi}{N}$, $h_2=\frac{1}{N}$ 作矩形剖分，网格节点为$x_i=ih_1$, $y_j=jh_2$, $i,j=0,1,\cdots,N.$

差分方程:

$$-\left(\frac{u_{i+1,j}-2u_{ij}+u_{i-1,j}}{h_1^2}+\frac{u_{i,j+1}-2u_{ij}+u_{i,j-1}}{h_2^2}\right)=\cos 3x_i\sin \pi y_j$$
$$i,j=1,2,\cdots,N-1.$$

边界条件:

$$\begin{aligned} &u_{i0}=u_{iN}=0, i=0,\cdots,N \\ &u_{0j}=u_{1j}, j=1,\cdots,N-1 \\ &u_{Nj}=u_{N-1,j}, j=1,\cdots,N-1 \end{aligned}$$

离散格式:

$$\boldsymbol{D}\boldsymbol{u}_{j-1}+\boldsymbol{C}\boldsymbol{u}_j+\boldsymbol{D}\boldsymbol{u}_{j+1}=\boldsymbol{f}_j, \quad 1\leqslant j\leqslant M-1$$

$$\boldsymbol{C}=\begin{pmatrix} \left(\frac{1}{h_1^2}+\frac{2}{h_2^2}\right) & -\frac{1}{h_1^2} & & & \\ -\frac{1}{h_1^2} & 2\left(\frac{1}{h_1^2}+\frac{1}{h_2^2}\right) & -\frac{1}{h_1^2} & & \\ & \ddots & \ddots & \ddots & \\ & & -\frac{1}{h_1^2} & 2\left(\frac{1}{h_1^2}+\frac{1}{h_2^2}\right) & -\frac{1}{h_1^2} \\ & & & -\frac{1}{h_1^2} & \left(\frac{1}{h_1^2}+\frac{2}{h_2^2}\right) \end{pmatrix}$$

$$\boldsymbol{D}=\begin{pmatrix} -\frac{1}{h_2^2} & & & & \\ & -\frac{1}{h_2^2} & & & \\ & & \vdots & & \\ & & & -\frac{1}{h_2^2} & \\ & & & & -\frac{1}{h_2^2} \end{pmatrix} \qquad \boldsymbol{f}_j=\begin{pmatrix} f(x_1,y_j) \\ f(x_2,y_j) \\ \vdots \\ f(x_{N-2},y_j) \\ f(x_{N-1},y_j) \end{pmatrix}$$

矩阵形式：

$$
\begin{pmatrix}
\boldsymbol{C} & \boldsymbol{D} & & & \\
\boldsymbol{D} & \boldsymbol{C} & \boldsymbol{D} & & \\
& \ddots & \ddots & \ddots & \\
& & \boldsymbol{D} & \boldsymbol{C} & \boldsymbol{D} \\
& & & \boldsymbol{D} & \boldsymbol{C}
\end{pmatrix}
\begin{pmatrix}
\boldsymbol{u}_1 \\ \boldsymbol{u}_1 \\ \vdots \\ \boldsymbol{u}_{M-2} \\ \boldsymbol{u}_{M-1}
\end{pmatrix}
=
\begin{pmatrix}
\boldsymbol{f}_1 \\ \boldsymbol{f}_1 \\ \vdots \\ \boldsymbol{f}_{M-2} \\ \boldsymbol{f}_{M-1}
\end{pmatrix}
$$

最后再利用边界条件处理边界处的值。

```
% fdm2d2_error.m
% finite difference method for 2D problem
% -\Delta u=cos(3*x)*sin(pi*y)  in (0,pi)x(0,1)
% u(x,0)=u(x,1)=0  in [0,pi]
% u_x(0,y)=u_x(pi,y)=0 in [0,1]
% exact solution: ue=(9+pi^2)^(-1)*cos(3*x)*sin(pi*y)
clear all; close all;
Nvec=2.^[2:7];  Err=[];
for N=Nvec
    h1=pi/N;  h2=1/N;
    x=[0:h1:pi]';   y=[0:h2:1]';
    [X,Y]=meshgrid(x,y);
    X1=X(2:N,2:N);  Y1=Y(2:N,2:N);
    % generate the matrix of RHS
    f=cos(3*X1).*sin(pi*Y1);
    % constructing the coefficient matrix
    e=ones(N-1,1);
    C=diag([1/h1^2+2/h2^2, (2/h1^2+2/h2^2)*ones(1,N-3), ...
        1/h1^2+2/h2^2])...
        -1/h1^2*diag(ones(N-2,1),1)-1/h1^2*diag(ones(N-2,1),-1);
    D=-1/h2^2*eye(N-1);
    A=kron(eye(N-1),C)+kron(diag(ones(N-2,1),1)+diag(ones(N-2,1),-1),D);
    %A=kron(eye(N-1),C)+kron(spdiags([e e],[-1 1],N-1,N-1),D);
    % solving the linear system
    f=f';
    u=zeros(N+1,N+1);
    u(2:N,2:N)=reshape(A\f(:),N-1,N-1)';
    % Neumann boundary condition
    u(:,1)=u(:,2);
    u(:,end)=u(:,end-1);
    ue=1/(9+pi^2)*(cos(3*X)).*(sin(pi*Y));
    err=max(max(abs(u-ue)));   % maximum error
    Err=[Err,err];
end
plot(log10(Nvec),log10(Err),'ro-','MarkerFaceColor','w','LineWidth',1.5)
grid on,hold on, plot(log10(Nvec),log10(Nvec.^(-1)),'--')
xlabel('log_{10}N','fontsize', ...
    16),ylabel('log_{10}Error','fontsize',16),
title('Convergence of Finite Difference Method','fontsize',14)
set(gca,'fontsize',14)
for i=1:length(Nvec)-1     % computating convergence order
    order(i)=log(Err(i)/Err(i+1))/(log(Nvec(i)/Nvec(i+1)));
end
order
```