

2 差分方法

2.1 一维差分法

一维差分格式(常系数):

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

离散格式:

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

例 2.1

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

真解: $u(x) = \sin(\pi x)$.

```
1 % fdm1d1.m
2 % finite difference method for 1D problem
3 % -u''+u=pi^2*sin(pi*x)+pi*cos(pi*x) in [0,1]
4 % u(0)=0, u(1)=0 ;
5 % exact solution : u=sin(pi*x)
6 clear all
7 h=0.05;
8 x=0:h:1;
9 N=length(x)-1;
10 A=diag((2/h^2)*ones(N-1,1))...
11     +diag((1/(2*h)-1/h^2)*ones(N-2,1),1)...
12     +diag((-1/(2*h)-1/h^2)*ones(N-2,1),-1);
13 b=pi^2*sin(pi*x(2:N))+pi*cos(pi*x(2:N));
14 u=A\b';
15 u=[0;u;0];
16 ue=sin(pi*x)';
17 plot(x,ue,'b-',x,u,'r+', 'LineWidth',1.5)
18 Error=max(abs(u-ue))
19 xlabel('x','fontsize',16), ylabel('y','fontsize',16,'Rotation',0)
20 legend('Exact','Numerical','location','North')
21 title('Finite Difference Method','fontsize',14)
22 set(gca,'fontsize',14)
```

一维差分格式(变系数):

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

离散格式:

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

例 2.2

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

真解: $u(x) = \sin(\pi x)$.

```

1 % fdm1d2.m
2 % finite difference method for 1D problem
3 % -(xu')'+xu'=pi^2*x*sin(pi*x)-pi*cos(pi*x)+pi*x*cos(pi*x) in [0,1]
4 % u(0)=0, u(1)=0 ;
5 % exact solution : u=sin(pi*x)
6 clear all
7 h=0.05;
8 x=0:h:1;
9 N=length(x)-1;
10 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)...
11     +diag(-x(3:N)./(2*h)-(x(3:N)-0.5*h)./h^2,-1);
12 b=pi^2*x(2:N).*sin(pi*x(2:N))+pi*(x(2:N)-1).*cos(pi*x(2:N));
13 u=A\b';
14 u=[0;u;0];
15 ue=sin(pi*x');
16 plot(x,ue,'b-',x,u,'r+', 'LineWidth',1.5)
17 Error=max(abs(u-ue))
18 xlabel('x','fontsize',16), ylabel('y','fontsize',16,'Rotation',0)
19 legend('Exact','Numerical','location','North')
20 title('Finite Difference Method','fontsize',14)
21 set(gca,'fontsize',14)

```

2.2 二维差分法

考虑二维泊松问题:

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

离散格式:

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

定义向量: $\mathbf{u}_j = (u_{1j}, u_{2j}, \dots, u_{N-1,j})^T$, $0 \leq j \leq M$.

离散格式写成矩阵形式:

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

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

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

进一步写成如下矩阵形式:

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

例 2.3

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

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

```
1 % fdm2d1.m
2 % finite difference method for 2D problem
3 % -d^2u/dx^2-d^2u/dy^2=f(x,y)
4 % f(x,y)=-2*pi^2*exp(pi*(x+y))*(sin(pi*x)*cos(pi*y)+cos(pi*x)*sin(pi*y))
5 % exact solution: ue=exp(pi*x+pi*y)*sin(pi*x)*sin(pi*y)
6 clear all
7 h=0.01;
8 x=[0:h:1]';
9 y=[0:h:1]';
10 N=length(x)-1;
11 M=length(y)-1;
12 [X,Y]=meshgrid(x,y);
13 X=X(2:M,2:N);
14 Y=Y(2:M,2:N);
15 % generate the matrix of RHS
16 f=-2*pi^2*exp(pi*X+pi*Y).*(sin(pi*X).*cos(pi*Y)+cos(pi*X).*sin(pi*Y));
17 % constructing the coefficient matrix
18 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);
19 D=-1/h^2*eye(N-1);
20 A=kron(eye(M-1),C)+kron(diag(ones(M-2,1),1)+diag(ones(M-2,1),-1),D);
21 % solving the linear system
22 f=f';
23 u=zeros(M+1,N+1);
24 u(2:M,2:N)=reshape(A\f(:),N-1,M-1)';
25 u(:,1)=0;
26 u(:,end)=0;
27 ue=zeros(M+1,N+1);
28 ue(2:M,2:N)=exp(pi*X+pi*Y).*sin(pi*X).*sin(pi*Y);
29 % compute maximum error
30 Error=max(max(abs(u-ue)))
31 mesh(x,y,u)
32 xlabel('x','fontsize',16), ylabel('y','fontsize',16), ...
    zlabel('u','fontsize',16,'Rotation',0)
33 title('Finite Difference Method','fontsize',14)
34 set(gca,'fontsize',14)
```

```

1 % fdm2d1_error.m
2 % finite difference method for 2D problem
3 %  $-d^2u/dx^2-d^2u/dy^2=f(x,y)$ 
4 %  $f(x,y)=-2\pi^2\exp(\pi(x+y))\cdot(\sin(\pi x)\cos(\pi y)+\cos(\pi x)\sin(\pi y))$ 
5 % exact solution:  $ue=\exp(\pi x+\pi y)\sin(\pi x)\sin(\pi y)$ 
6 clear all
7 Nvec=2.^[3:10]; Err=[];
8 for n=Nvec
9     h=1/n;
10    x=[0:h:1]'; y=[0:h:1]';
11    N=length(x)-1; M=length(y)-1;
12    [X,Y]=meshgrid(x,y);
13    X=X(2:M,2:N);
14    Y=Y(2:M,2:N);
15    % generate the matrix of RHS
16    f=-2*pi^2*exp(pi*(X+Y)).*(sin(pi*X).*cos(pi*Y)+cos(pi*X).*sin(pi*Y));
17    %constructing the coefficient matrix
18    e=ones(N-1,1);
19    C=1/h^2*spdiags([-e 4*e -e],[-1 0 1],N-1,N-1);
20    D=-1/h^2*eye(N-1);
21    e=ones(M-1,1);
22    A=kron(eye(M-1),C)+kron(spdiags([e e],[-1 1],M-1,M-1),D);
23    % solving the linear system
24    f=f';
25    u=zeros(M+1,N+1);
26    u(2:M,2:N)=reshape(A\f(:),N-1,M-1)';
27    u(:,1)=0;
28    u(:,end)=0;
29    ue=zeros(M+1,N+1); % numerical solution
30    ue(2:M,2:N)=exp(pi*X+pi*Y).*sin(pi*X).*sin(pi*Y);
31    err=max(max(abs(u-ue))); % maximum error
32    Err=[Err,err];
33 end
34 plot(log10(Nvec),log10(Err),'ro-','MarkerFaceColor','w','LineWidth',1.5)
35 grid on,hold on, plot(log10(Nvec), log10(Nvec.^(-2)), '--')
36 xlabel('log_{10}N','fontsize',16),ylabel('log_{10}Error','fontsize',16),
37 title('Convergence of Finite Difference Method','fontsize',14)
38 set(gca,'fontsize',14)
39 for i=1:length(Nvec)-1 % computating convergence order
40     order(i)=-log(Err(i)/Err(i+1))/(log(Nvec(i)/Nvec(i+1)));
41 end
42 Err
43 order

```

例 2.4

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

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 $x_i = ih_1$, $y_j = jh_2$, $i, j = 0, 1, \dots, N$.

差分方程:

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

$$i, j = 1, 2, \dots, N-1.$$

边界条件:

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

离散格式:

$$\mathbf{D}u_{j-1} + \mathbf{C}u_j + \mathbf{D}u_{j+1} = \mathbf{f}_j, \quad 1 \leq j \leq M-1$$

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

$$\mathbf{D} = \begin{pmatrix} -\frac{1}{h_2^2} & & & & \\ & -\frac{1}{h_2^2} & & & \\ & & \ddots & & \\ & & & -\frac{1}{h_2^2} & \\ & & & & -\frac{1}{h_2^2} \end{pmatrix}$$

$$\mathbf{f}_j = \begin{pmatrix} f(x_1, y_j) \\ f(x_2, y_j) \\ \vdots \\ f(x_{N-2}, y_j) \\ f(x_{N-1}, y_j) \end{pmatrix}$$

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。

```

1 % fdm2d2_error.m
2 % finite difference method for 2D problem
3 %  $-\Delta u = \cos(3x) \sin(\pi y)$  in  $(0, \pi) \times (0, 1)$ 
4 %  $u(x, 0) = u(x, 1) = 0$  in  $[0, \pi]$ 
5 %  $u_x(0, y) = u_x(\pi, y) = 0$  in  $[0, 1]$ 
6 % exact solution:  $u_e = (9 + \pi^2)^{-1} \cos(3x) \sin(\pi y)$ 
7 clear all; close all;
8 Nvec=2.^[2:7]; Err=[];
9 for N=Nvec
10     h1=pi/N; h2=1/N;
11     x=[0:h1:pi]'; y=[0:h2:1]';
12     [X,Y]=meshgrid(x,y);
13     X1=X(2:N,2:N); Y1=Y(2:N,2:N);
14     % generate the matrix of RHS
15     f=cos(3*X1).*sin(pi*Y1);
16     % constructing the coefficient matrix
17     e=ones(N-1,1);
18     C=diag([1/h1^2+2/h2^2, (2/h1^2+2/h2^2)*ones(1,N-3), ...
19             1/h1^2+2/h2^2])...
20         -1/h1^2*diag(ones(N-2,1),1)-1/h1^2*diag(ones(N-2,1),-1);
21     D=-1/h2^2*eye(N-1);
22     A=kron(eye(N-1),C)+kron(diag(ones(N-2,1),1)+diag(ones(N-2,1),-1),D);
23     %A=kron(eye(N-1),C)+kron(spdiags([e e],[-1 1],N-1,N-1),D);
24     % solving the linear system
25     f=f';
26     u=zeros(N+1,N+1);
27     u(2:N,2:N)=reshape(A\f(:),N-1,N-1)';
28     % Neumann boundary condition
29     u(:,1)=u(:,2);
30     u(:,end)=u(:,end-1);
31     ue=1/(9+pi^2)*(cos(3*X)).*(sin(pi*Y));
32     err=max(max(abs(u-ue))); % maximum error
33     Err=[Err,err];
34 end
35 plot(log10(Nvec),log10(Err),'ro-','MarkerFaceColor','w','LineWidth',1.5)
36 grid on,hold on, plot(log10(Nvec),log10(Nvec.^(-1)),'--')
37 xlabel('log_{10}N','fontsize',...
38         16),ylabel('log_{10}Error','fontsize',16),
39 title('Convergence of Finite Difference Method','fontsize',14)
40 set(gca,'fontsize',14)
41 for i=1:length(Nvec)-1 % computing convergence order
42     order(i)=log(Err(i)/Err(i+1))/(log(Nvec(i)/Nvec(i+1)));
43 end
44 order

```