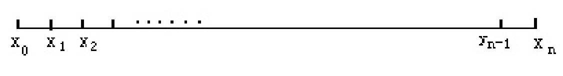
有限元法

制作人：舒适 魏华祎 易年余 岳孝强

1. 有限元方法
    - 线性元计算公式
    - 算法流程与实现、数值实验
    - 误差分析
    - 稳定性分析

线性有限元空间

对求解区间 I 做网格剖分



1. 网格剖分节点

$$a = x_0 < x_1 < \cdots < x_{n-1} < x_n = b$$

称 $x_i (i = 1, 2, \cdots, n-1)$ 为第 i 个内部节点, x_0 和 x_n 为边界节点;

2. 网格剖分单元 分别称 $I_i = [x_{i-1}, x_i]$ 和 $h_i = x_i - x_{i-1}$ 为第 i 个 剖分单元 和 剖分步长.

记 $h = \max\limits_{1 \le i \le n} h_i$.

定义

$$V_E^h = \left\{ u_h \in C(\bar{I}) : \ u_h|_{I_i} \in P_1(I_i), 1 \le i \le n, \ u_h(a) = 0 \right\}$$

$P_1(I_i)$: I_i 上线性代数多项式的全体.

称 V_E^h 为 1 次 Lagrange 型有限元空间 (简称线性元空间).

线性元空间 V_E^h 的维数

$$m := \dim V_E^h = 2n - (n-1) - 1 = n$$

V_E^h 中的函数在 $I_i = [x_{i-1}, x_i]$ 上形如：$a_i + b_i x$.
记 $u_i = u_h(x_i)$, $i = 0, 1, \cdots, n$, 对任给的 $u_h \in V_E^h$, 分别按两种形式给出其表示式.

(a) 分段线性表示（单元形状函数）

单元形状函数 $u_h^k(x)$: 为 $u_h(x)$ 在 $I_k = [x_{k-1}, x_k]$ 上的限制函数.

由 Lagrange 插值公式

$$u_h^k(x) = u_{k-1} l_{k,0}(x) + u_k l_{k,1}(x), \quad x \in I_k$$

其中

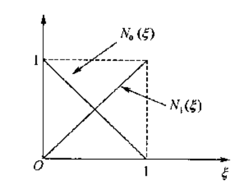
$$l_{k,0}(x) = \frac{x - x_k}{x_{k-1} - x_k}, \quad l_{k,1}(x) = \frac{x - x_{k-1}}{x_k - x_{k-1}}$$

为关于插值点 x_{k-1} 和 x_k 的 1 次 Lagrange 因子.

单元形状函数的另一种表示公式

在参考 (或标准) 单元 [0, 1] 上求得所谓的标准插值基函数:

$$N_0(\xi) = 1 - \xi, \; N_1(\xi) = \xi$$



引入仿射变换:

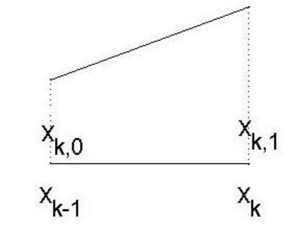
$$\xi = F_k(x) := \frac{x - x_{k-1}}{h_k} : \; I_k \to [0, 1] \tag{1}$$

逆变换

$$x = X(\xi) := x_{k-1} N_0(\xi) + x_k N_1(\xi), \; [0, 1] \to I_k \tag{2}$$

单元形状函数

$$u_h^k(x) = u_{k-1} N_0(\xi) + u_k N_1(\xi), \ \ x \in I_k \tag{3}$$



$u_h(x)$ 可分段线性表示为

$$u_h(x) = \begin{cases} u_h^1(x), x \in I_1 \\ u_h^2(x), x \in I_2 \\ \vdots \\ u_h^n(x), x \in I_n \end{cases}$$

(b) 整体表示

关键: 给出线性元空间 V_E^h 的一组基函数.

在每个非本质边界插值点 x_i 处, 引入函数

$$\phi_i(x) \in V_E^h, \quad i = 1, \cdots, n$$

满足插值条件

$$\phi_i(x_j) = \delta_{i,j}, \quad j = 1, \cdots, n \tag{4}$$

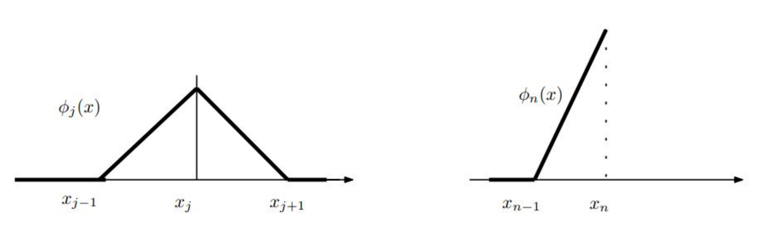
经简单计算, 可得

$$\left\{\begin{array}{l} \phi_i(x) = \begin{cases} 1 + \frac{x - x_i}{h_i}, & x_{i-1} \le x < x_i, \\ 1 - \frac{x - x_i}{h_{i+1}}, & x_i \le x \le x_{i+1}, \\ 0, & \text{在别处}, \end{cases} \\ \qquad\qquad\qquad i = 1, 2, \cdots, n-1 \\ \phi_n(x) = \begin{cases} 1 + \frac{x - x_n}{h_n}, & x_{n-1} \le x \le x_n, \\ 0, & \text{在别处}. \end{cases} \end{array}\right. \tag{5}$$

借助于仿射变换 (1) 及 $[0,1]$ 上的标准插值基函数，则 (5) 也可表示为

$$\begin{cases} \phi_i(x) = \begin{cases} N_1(\xi), & \xi = \frac{x-x_{i-1}}{h_i}, \quad x_{i-1} \le x < x_i, \\ N_0(\xi), & \xi = \frac{x-x_i}{h_{i+1}}, \quad x_i \le x \le x_{i+1}, \\ 0, & \text{在别处}, \end{cases} \\ \qquad\qquad i = 1, 2, \cdots, n-1 \\ \phi_n(x) = \begin{cases} N_1(\xi), & \xi = \frac{x-x_{n-1}}{h_n}, \quad x_{n-1} \le x \le x_n, \\ 0, & \text{在别处}, \end{cases} \end{cases} \tag{6}$$

其几何形状如下图所示



易知 $\{\phi_i(x),\quad i=1,2,\cdots,n\}$ 是一组线性无关的函数系，即构成了线性元空间 V_E^h 中一组基，称之为线性元空间 V_E^h 的 Lagrange 节点基函数.

事实上，若

$$u_n(x)=\sum_{i=1}^{n}c_i\phi_i(x)\equiv 0$$

特别地，取 $x=x_i$ 代入上式，可得 $c_i=0$，这样就证得了 $\{\phi_i\}_{i=1}^{n}$ 是一组线性无关的函数系。 $\square$

利用 (4)，易验证

$$u_h(x)=\sum_{i=1}^{n}u_i\phi_i(x). \tag{7}$$

至此，解决了 $H_E^1(I)$ 的有限维子空间 (线性元空间) 的构造问题. 下面利用 Galerkin 方法求解问题 (A).

基于 Galerkin 方法的线性有限元方程

1) 等价变分问题

基于虚功原理的等价变分问题 (问题 (B)): 求 $u \in H_E^1(I)$, 使

$$a(u,v) = (f,v), \forall v \in H_E^1(I) \tag{8}$$

其中

$$\begin{cases} a(u,v) = \int_a^b \left[p\frac{du}{dx} \cdot \frac{dv}{dx} + quv\right] dx \\ (f,v) = \int_a^b fv dx \end{cases} \tag{9}$$

下面给出基于 $H_E^1(I)$的子空间 V_E^h (线性元空间) 的 Galerkin 数值解计算公式.

2) 近似变分问题

问题（B）的近似变分问题：求 $u_h(x) \in V_E^h$，使得

$$a(u_h, v_h) = (f, v_h), \forall v_h \in V_E^h \tag{10}$$

3) 线性有限元方程

将 $u_h(x) = \sum_{i=1}^{n} u_i \phi_i(x)$ 代人 (10)，并将 v_h 取为基函数 ϕ_j，则有

$$a(\sum_{i=1}^{n} u_i\phi_i, \phi_j) = (f, \phi_j), \quad j = 1, 2, \cdots, n$$

$\Leftrightarrow$

$$\sum_{i=1}^{n} a(\phi_i, \phi_j)u_i = (f, \phi_j), \quad j = 1, 2, \cdots, n \tag{11}$$

$\Leftrightarrow$ (注意: 当 $|j-i| \geq 2$ 时, $\phi_i \cdot \phi_j = 0$)

1. 当 $2 \leq j \leq n-1$ 时，方程 (11) 的左端只有三个非零系数：$a(\phi_j, \phi_j)$ 和 $a(\phi_{j\pm1}, \phi_j)$.
2. 当 $j=1$ 时，方程 (11) 的左端只有两个非零系数：$a(\phi_1, \phi_1)$ 和 $a(\phi_2, \phi_1)$.
3. 当 $j=n$ 时，方程 (11) 的左端只有两个非零系数：$a(\phi_n, \phi_n)$ 和 $a(\phi_{n-1}, \phi_n)$.

基于线性元子空间的 Galerkin 数值解满足的计算公式：

$$KU = b \tag{12}$$

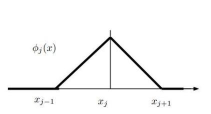
其中 n 阶方阵 $K=(a_{i,j})$ 和 n 维向量 $b=(b_1, \cdots, b_n)^T$ 分别为

$$K = \begin{bmatrix} a(\phi_1,\phi_1) & a(\phi_1,\phi_2) & & \\ a(\phi_2,\phi_1) & a(\phi_2,\phi_2) & a(\phi_2,\phi_3) & \\ \ddots & \ddots & \ddots & \\ & a(\phi_{n-1},\phi_{n-2}) & a(\phi_{n-1},\phi_{n-1}) & a(\phi_{n-1},\phi_n) \\ & & a(\phi_n,\phi_{n-1}) & a(\phi_n,\phi_n) \end{bmatrix}$$

$$b = \begin{bmatrix} (f,\phi_1) \\ \vdots \\ (f,\phi_n) \end{bmatrix}$$

利用一般的 Galerkin 理论可知 (12) 的解存在且唯一.

利用 (6) 和 (9)



$$a(u,v)=\int_a^b\left[p\frac{du}{dx}\cdot\frac{dv}{dx}+quv\right]dx$$

$$
\begin{aligned}
a_{jj} &= \int_{x_{j-1}}^{x_j} [p(\phi_j')^2 + q(\phi_j)^2]dx + \int_{x_j}^{x_{j+1}} [p(\phi_j')^2 + q(\phi_j)^2]dx \\
&= \int_{x_{j-1}}^{x_j} [p(\frac{dN_1(\xi)}{dx})^2 + q(N_1(\xi))^2]dx \ \ (\text{这里 } \xi = \tfrac{x-x_{j-1}}{h_j}) \\
&\quad + \int_{x_j}^{x_{j+1}} [p(\frac{dN_0(\xi)}{dx})^2 + q(N_0(\xi))^2]dx \ \ (\text{这里 } \xi = \tfrac{x-x_j}{h_{j+1}}) \\
&= a(N_1, N_1)_{I_j} + a(N_0, N_0)_{I_{j+1}} \qquad (13)
\end{aligned}
$$

类似可得 (习题):

1. 矩阵非对角元素 $a_{j,j-1} := a(\phi_{j-1}, \phi_j)$ $(j = 2, \cdots, n)$的表示式为

$$a_{j,j-1} = \int_{x_{j-1}}^{x_j} [p(\phi'_{j-1}\phi'_j) + q(\phi_{j-1}\phi_j)]dx = a(N_0, N_1)_{I_j} \quad (14)$$

2. 由对称性知：矩阵非对角元素 $a_{j,j+1} := a(\phi_{j+1}, \phi_j)$ $(j = 1, \cdots, n-1)$的表示式为

$$a_{j,j+1} = a(N_1, N_0)_{I_{j+1}} \quad (15)$$

方程的右端项 $(j=1,\cdots,n-1)$

$$
\begin{aligned}
b_j := \int_a^b f\phi_j dx &= \int_{x_{j-1}}^{x_j} f(x)\phi_j(x)dx + \int_{x_j}^{x_{j+1}} f(x)\phi_j(x)dx \\
&= \int_{x_{j-1}}^{x_j} f(x)N_1(\xi)dx \ \ (\text{这里 } \xi = \tfrac{x-x_{j-1}}{h_j}) \\
&\quad + \int_{x_j}^{x_{j+1}} f(x)N_0(\xi)dx \ \ (\text{这里 } \xi = \tfrac{x-x_j}{h_{j+1}}) \\
&= (f,N_1)_{I_j} + (f,N_0)_{I_{j+1}} \qquad (16)
\end{aligned}
$$

$$
b_n = (f,N_1)_{I_n} = h_n \int_0^1 f(x_{n-1}+h_n\xi)N_1(\xi)d\xi
$$

习题 1 （P.231. 1. （第四版））仅要求

（1）网格剖分：作 2 或 3 段的等距剖分；

（2）利用中矩形公式计算积分 (16)。

习题 2* 导出非齐次两点边值问题

$$\begin{cases} Lu = -\frac{d}{dx}(p\frac{du}{dx}) + qu = f, \quad a < x < b \\ u(a) = \gamma, \quad u'(b) + \alpha u(b) = \beta \end{cases}$$

的线性元方程, 其中

(1) 正的光滑函数 $p(x)$, 非负连续函数 $q(x)$ 可自己定义，如取

$$p(x) = x - a + 1, q(x) = 0$$

(2) $\gamma, \alpha \geq 0, \beta$ 也可自己定义，如取 $\gamma = 2$, $\alpha = 1$, $\beta = 1$.

$$
\begin{aligned}
a(u_h, v_h)_{I_k} &= a(u_{k-1}N_0 + u_k N_1, v_{k-1}N_0 + v_k N_1)_{I_k} \\
&= a(N_0, N_0)_{I_k} u_{k-1}v_{k-1} + a(N_1, N_0)_{I_k} u_k v_{k-1} \\
&\quad + a(N_0, N_1)_{I_k} u_{k-1}v_k + a(N_1, N_1)_{I_k} u_k v_k \\
&= (v_{k-1}, v_k) K^{I_k} \begin{pmatrix} u_{k-1} \\ u_k \end{pmatrix}
\end{aligned}
$$

其中

$$
K^{I_k} := \begin{bmatrix} a_{11}^{I_k} & a_{12}^{I_k} \\ a_{21}^{I_k} & a_{22}^{I_k} \end{bmatrix} = \begin{bmatrix} a(N_0, N_0)_{I_k} & a(N_1, N_0)_{I_k} \\ a(N_0, N_1)_{I_k} & a(N_1, N_1)_{I_k} \end{bmatrix} \tag{17}
$$

称为单元刚度矩阵.

$$\begin{aligned}(f, v_h)_{I_k} &= (f, v_{k-1}N_0 + v_k N_1) \\ &= (f, N_0)_{I_k} v_{k-1} + (f, N_1)_{I_k} v_k \\ &= (v_{k-1}, v_k)\, b^{I_k}\end{aligned}$$

其中

$$b^{I_k} = \begin{pmatrix} b_1^{I_k} \\ b_2^{I_k} \end{pmatrix} = \begin{pmatrix} (f, N_0)_{I_k} \\ (f, N_1)_{I_k} \end{pmatrix} \tag{18}$$

称为单元荷载向量.

由 (13), (14), (15) 和 (17) 可知：总刚度矩阵的元素与单元刚度矩阵的元素有如下关系：

1. 矩阵对角元 $a_{jj} = a_{22}^{I_j} + a_{11}^{I_{j+1}}, \quad \forall j = 1, 2, 3, \cdots, n-1;$
2. 矩阵非对角元 $a_{j,j-1} = a_{21}^{I_j}, \quad \forall j = 2, 3, \cdots, n;$
3. 矩阵非对角元 $a_{j,j+1} = a_{12}^{I_{j+1}}, \quad \forall j = 1, 2, \cdots, n-1;$
4. 矩阵对角元 $a_{n,n} = a_{22}^{I_n}.$

单元刚度矩阵对总刚度矩阵的叠加方式

当 $j=1$ 时,

$$\begin{bmatrix} * & * \\ * & a_{22}^{l_1} \end{bmatrix} \longrightarrow \begin{bmatrix} a_{11} & \cdots & \cdots \\ \vdots & \ddots & \\ \vdots & & \ddots \end{bmatrix}$$

当 $j=2,\cdots,n$ 时,

$$\begin{bmatrix} a_{11}^{l_j} & a_{12}^{l_j} \\ a_{21}^{l_j} & a_{22}^{l_j} \end{bmatrix} \longrightarrow \begin{bmatrix} & \vdots & \vdots & \\ \cdots & a_{j-1,j-1} & a_{j-1,j} & \cdots \\ \cdots & a_{j,j-1} & a_{j,j} & \cdots \\ & \vdots & \vdots & \end{bmatrix}$$

单元载荷向量对总载荷向量的叠加方式

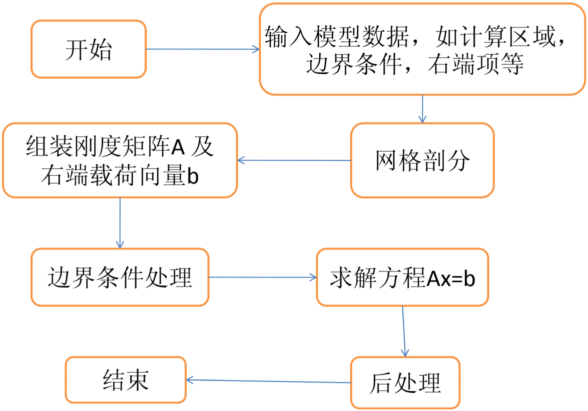
当 $j=1$ 时,

$$\begin{pmatrix} * \\ b_2^{I_1} \end{pmatrix} \rightarrow \begin{bmatrix} b_1 \\ \vdots \end{bmatrix}$$

当 $j=2,\cdots,n$ 时,

$$\begin{pmatrix} b_1^{I_j} \\ b_2^{I_j} \end{pmatrix} \rightarrow \begin{bmatrix} \vdots \\ b_{j-1} \\ b_j \\ \vdots \end{bmatrix}$$

算法流程


开始
输入模型数据，如计算区域，边界条件，右端项等
网格剖分
组装刚度矩阵A 及右端载荷向量b
边界条件处理
求解方程Ax=b
后处理
结束

代码实现 I

```
%% 准备初始数据

% 区间[a,b]
a = 0;
b = 1;

%网格剖分尺寸
h = 0.1;

% 微分方程模型数据。函数 sindata 返回一个结构体 pde
% pde.f : 右端项函数
% pde.exactu : 真解函数
% pde.Du : 真解导数
% pde.g_D: D 氏边界条件函数
pde = sindata();

% 设定积分精度Gauss
option.fQuadOrder = 3;
option.errQuadOder = 3;
```

代码实现 II

```
%% 网格剖分
[node,elem,bdFlag] = intervalmesh(a,b,h);

%% 组装刚度矩阵及右端向量、边界条件处理、求解Ab
uh = Poisson1d(node, elem, pde, bdFlag,option);

%% 计算 L2 和 H1 误差、结果可视化
errL2 = getL2error1d(node,elem,pde.exactu,uh,option.errQuadOder);
errH1 = getH1error1d(node,elem,pde.Du,uh,option.errQuadOder);
showsolution1d(node,elem,uh,'-+k');
```

模型数据 I

```
function pde = sindata( )
%% SINDATA
%
%   u = sin(pi*x)
%   f = pi*pi*sin(pi*x)
%   Du = pi*cos(pi*x)
%

pde = struct('f',@f,'exactu',@exactu,'g_D',@g_D,'Du',@Du);

% right hand side function
function z = f(p)
    x = p;
    z = pi*pi*sin(pi*x);
end
% exact solution
function z = exactu(p)
    x = p;
    z = sin(pi*x);
end
```

模型数据 II

```
% Dirichlet boundary condition
function z = g_D(p)
    x = p;
    z = exactu(p);
end
% Derivative of the exact solution
function z = Du(p)
    x = p;
    z = pi*cos(pi*x);
end

end
```

网格剖分 I

```
function [node,elem,bdFlag] = intervalmesh(a,b,h)
%% INTERVALMESH the uniform mesh on interval [a,b] with size h.
%
% node(1:N,1): node(i) is the coordinate of i-th
% mesh point.
%
% elem(1:NT,1:2): elem(j,1:2) are the indexes of the two end
% vertices of j-th elements.
%

node = a:h:b;
node = node';
N = length(node);
elem = [1:N-1;2:N];
elem = elem';
bdFlag = false(N,1);
bdFlag([1,N]) = true;
```

网格剖分 I

```
function [node,elem,bdFlag] = intervalmesh(a,b,h)
%% INTERVALMESH the uniform mesh on interval [a,b] with size h.
%
% node(1:N,1): node(i) is the coordinate of i-th
% mesh point.
%
% elem(1:NT,1:2): elem(j,1:2) are the indexes of the two end
% vertices of j-th elements.
%

node = a:h:b;
node = node';
N = length(node);
elem = [1:N-1;2:N];
elem = elem';
bdFlag = false(N,1);
bdFlag([1,N]) = true;
```

sparse函数介绍 I

sparse 是 Matlab 中生成稀疏矩阵(只存储非零元素的矩阵)的函数，下面给出三种基本用法:

A = sparse(N,N): 生成一个规模 N*N 的稀疏矩阵，所有元素为0

A = sparse(B): 把 B 转化为相同规模的稀疏矩阵 A

A = sparse(rows,cols,vals,N,N): 通过三元数组组装稀疏矩阵 A

sparse函数介绍 II

```
>> B
B =
     1     0     0
     0     0     2
     3     0     0

>> [rows,cols,vals] = find(B)
rows =      cols =       vals =
     1           1            1
     3           1            3
     2           3            2
```

sparse函数介绍 III

```
>> A = sparse(rows,cols,vals,3,3)

A =

   (1,1)        1
   (3,1)        3
   (2,3)        2
```

accumarray函数介绍 I

accumarray 是 Matlab 中生成满矩阵的函数，基本用法如下：

A = accumarray(subs,vals,[N,1]): 生成一个长度为 N 的列向量，其中 subs 和 vals 是相同长度的向量.

```
>> [subs, vals]
ans =
     1     2.5000
     2     1.3000
     1     2.5000
     4     4.1000
     5     5.1000
```

accumarray函数介绍 II

```
>> b = accumarray(subs,vals,[4,1])

b =

    5.0000
    1.3000
    3
    4.1000
    5.1000
```

组装刚度矩阵、右端项及求解 I

```
function uh = Poisson1d(node, elem, pde, bdFlag,option)
%% POISSON1D solve 1d Poisson equation by P1 linear element.
%
%  uh = Poisson1d(node,elem,pde, bdFlag) produces linear
%  finite element approximation of 1d Poisson equation.

N = size(node,1); NT = size(elem,1); Ndof = N;
%% Compute geometric quantities and gradient of local basis
lens = node(elem(:,2))-node(elem(:,1));
Dphi = [-1./lens,1./lens];

%% Assemble stiffness matrix
A = sparse(Ndof,Ndof);
for i = 1:2
    for j = i:2
        Aij = Dphi(:,i).*Dphi(:,j).*lens;
        if (j==i)
            A = A + sparse(elem(:,i), elem(:,j),Aij,Ndof,Ndof);
        else
```

组装刚度矩阵、右端项及求解 II

```
            A = A + sparse([elem(:,i);elem(:,j)],[elem(:,j);elem
    (:,i)],...
                [Aij; Aij],Ndof,Ndof);
        end
    end
end

%% Assemble the right hand side
[lambda,weight] = quadpts1d(option.fQuadOrder);
nQuad = length(weight);
phi = lambda;
bt = zeros(NT,2);
for i = 1:nQuad
    px = node(elem(:,1))*phi(i,1) + node(elem(:,2))*phi(i,2);
    fp = pde.f(px);
    for k = 1:2
        bt(:,k) = bt(:,k) + weight(i)*fp.*phi(i,k);
    end
end
bt = bt.*repmat(lens,1,2);
b = accumarray(elem(:),bt(:),[Ndof 1]);
```

组装刚度矩阵、右端项及求解 III

```
clear bt px;

%% modify left-hand vector
isFixed = bdFlag;
isFree = ~isFixed;
uh = zeros(Ndof,1);
uh(isFixed) = pde.g_D(node(isFixed));
b = b - A*uh;

%% solve
uh(isFree) = A(isFree,isFree)\b(isFree);
```

计算误差 I

```
function err = getL2error1d(node,elem, exactu, uh, quadOrder)
%% GETL2ERROR1D L2 norm of approximation of linear fintie
%% element
%
% compute L2 error element-wise using quadrature rule with order
% quadOrder

NT = size(elem,1);
err = zeros(NT,1);
[lambda,weight] = quadpts1d(quadOrder);

% basis function at quadrature points
phi = lambda;

nQuad = length(weight);
for i = 1:nQuad
    uhp = uh(elem(:,1))*phi(i,1) + uh(elem(:,2))*phi(i,2);
    px = node(elem(:,1))*phi(i,1) + node(elem(:,2))*phi(i,2);
    err = err + weight(i)*(exactu(px) - uhp).^2;
end
```

计算误差 II

```
lens = node(elem(:,2)) - node(elem(:,1));
err = err.*lens;
err = sqrt(sum(err));

function err = getH1error1d(node,elem,Du,uh,quadOrder)
%% GETH1ERROR1D H1 norm of approximation error of linear finite
%% element
%
% compute H1 error element-wise using quadrature rule
% with order quadOrder

NT = size(elem,1);
err = zeros(NT,1);
[lambda,weight] = quadpts1d(quadOrder);
phi = lambda;
lens = node(elem(:,2))-node(elem(:,1));
Dphi = [-1./lens,1./lens];

nQuad = length(weight);
Duh = uh(elem(:,1)).*Dphi(:,1) + uh(elem(:,2)).*Dphi(:,2);
```

计算误差 III

```
for i = 1:nQuad
    px = node(elem(:,1))*phi(i,1) + node(elem(:,2))*phi(i,2);
    err = err + weight(i)*(Du(px)-Duh).^2;
end

err = err.*lens;
err = sqrt(sum(err));
```

算例 1：我们利用有限元法去求解

$$-u''(x) = 16\pi^2 \sin(4\pi x),$$
$$u(0) = 0, \quad u(1) = 0.$$

问题的真解为

$$u(x) = \sin(4\pi x)$$

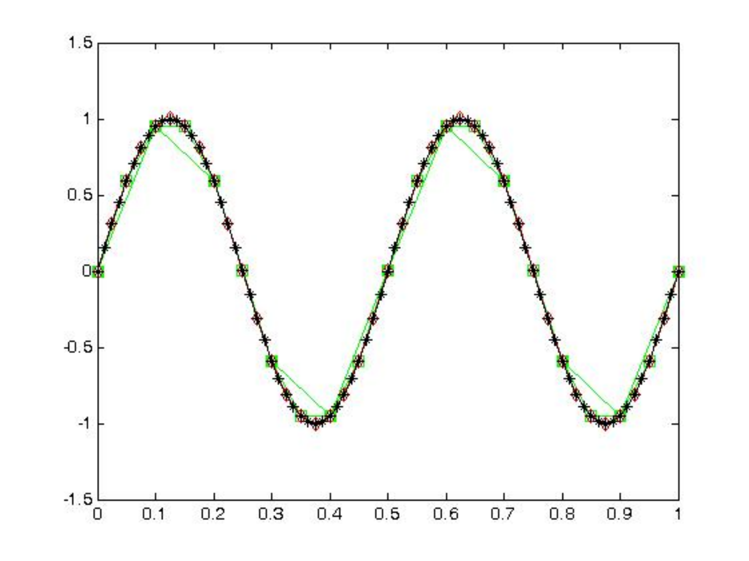
```
function pde = sin4pidata( )
%% SINDATA
%
%   u = sin(4*pi*x)
%   f = 16*pi*pi*sin(4*pi*x)
%   Du  = 4*pi*cos(4*pi*x)
%

pde = struct('f',@f,'exactu',@exactu,'g_D',@g_D,'Du',@Du);

% right hand side function
```

```
function z = f(p)
    x = p;
    z = 16*pi*pi*sin(4*pi*x);
end
% exact solution
function z = exactu(p)
    x = p;
    z = sin(4*pi*x);
end
% Dirichlet boundary condition
function z = g_D(p)
    x = p;
    z = exactu(p);
end
% Derivative of the exact solution
function z = Du(p)
    x = p;
    z = 4*pi*cos(4*pi*x);
end

end
```

```
function [N,errL2,errH1] = femtest1d()
%% the initial data
h = 0.1;
a = 0;
b = 1;
pde = sin4pidata();
option.fQuadOrder = 3;
option.errQuadOder = 3;


maxIt = 5;
errL2 = zeros(maxIt,1);
errH1 = zeros(maxIt,1);
N = zeros(maxIt,1);

for i = 1:maxIt
    [node,elem, bdFlag] = intervalmesh(a,b,h/2^(i-1));
    uh = Poisson1d(node,elem,pde,bdFlag,option);
    N(i) = size(elem,1);

    name = ['solution' int2str(N(i))];
    save(name,'node','elem','uh');
    errL2(i) = getL2error1d(node,elem,pde.exactu,uh,option.
    errQuadOder);
```

```
    errH1(i) = getH1error1d(node,elem,pde.Du,uh,option.
    errQuadOder);
end
```

有限元解



误差

可得如下结果

N	10	20	40	80	160
$\|u-u_h\|_0$	8.8574e-2	2.2976e-2	5.7977e-3	1.4528e-3	3.6341e-4
误差比	-	3.8551	3.9630	3.9907	3.9977
$\|u-u_h\|_1$	3.1532	1.6029	8.0475e-1	4.0279e-1	2.0145e-1
误差比	-	1.9672	1.9918	1.9979	1.9995

Table: 其中N是单元个数。

设h为单元尺寸，由以上数据结果可知:

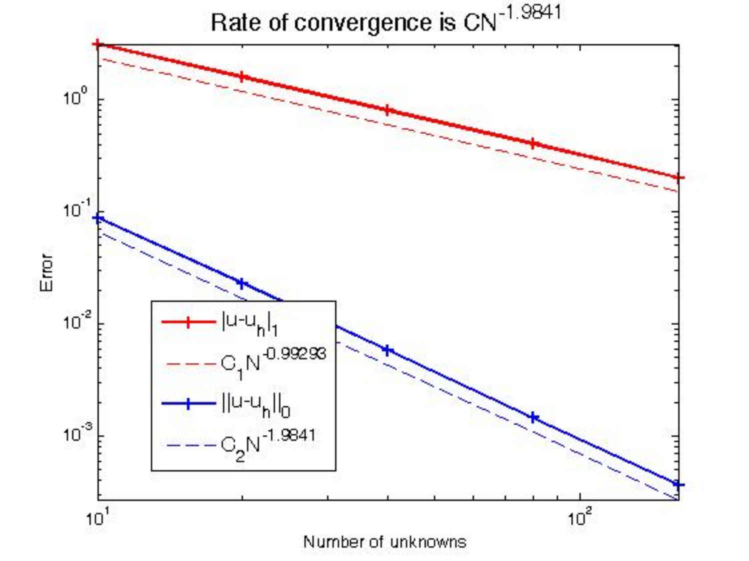
$$\begin{aligned} \|u-u_h\|_0 &\approx O(h^2) = O(N^{-2}) \\ |u-u_h|_1 &\approx O(h^1) = O(N^{-1}) \end{aligned}$$

数据分析

给定单元个数向量 N 和 误差向量 err，我们可以利用 Matlab 中的：

- polyfit: 多项式拟合函数。例子：
  p = polyfit(log(N),log(err),1)
- loglog : X和Y轴都取log的画图函数。
  loglog(N,err,'-*')

对上述计算结果进行数据分析和可视化。


Rate of convergence is CN-1.9841
Error
Number of unknowns
10^0
10^-1
10^-2
10^-3
10^1
10^2
|u-u_h|_1
C_1N^-0.99293
||u-u_h||_0
C_2N^-1.9841

线性元误差估计

给定两点边值问题 (A)

$$\begin{cases} Lu = -\frac{d}{dx}(p\frac{du}{dx}) + qu = f, \quad a < x < b \\ u(a) = 0, u'(b) = 0 \end{cases} \tag{19}$$

设 u_h 是 u 的线性元解函数，下面分别在两种不同的范数下，

$$\|\cdot\|_1 \text{ 和 } \|\cdot\|_0$$

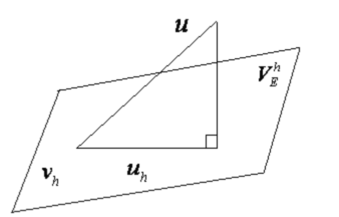
给出有限元函数的误差估计.

线性元误差估计

- H_1 范数 $(\|\cdot\|_1)$ 下的误差估计

由 Ritz-Galerkin 理论，有

$$a(u-u_h, v_h) = 0, \quad \forall v_h \in V_E^h \tag{20}$$



拟最佳逼近性质

$$\|u-u_h\|_1 \le \beta \inf_{\forall v_h \in V_E^h} \|u-v_h\|_1 \tag{21}$$

称 $u_I(x) \in V_E^h$ 为 $u(x) \in H_E^1 \cap C(\bar{I})$ 的分段线性插值函数，如果它满足

$$u_I(x_i) = u(x_i), \quad i = 1, \cdots, n \tag{22}$$

当 $x \in I_k, k = 1, \cdots, n$ 时，有

$$\begin{aligned} u_I(x) &= u_I(x_{k-1})N_0(\xi) + u_I(x_k)N_1(\xi) \\ &= u(x_{k-1})N_0(\xi) + u(x_k)N_1(\xi) \end{aligned}$$

$$x = x_{k-1}N_0(\xi) + x_k N_1(\xi)$$

利用 (21)，有

$$\|u-u_h\|_1 \le \beta \|u-u_I\|_1 \tag{23}$$

设 $u \in C^2(\bar{I})$，下面给出 $\|u-u_I\|_1$ 的估计.

首先给出 $\|u-u_I\|_0^2$ 的估计. 对 $\forall x \in I_i$，有

$$u(x)-u_I(x) = \frac{(x-x_{i-1})(x-x_i)}{2}u''(\xi_i), \quad \xi_i \in (x_{i-1}, x_i)$$

$\Rightarrow$

$$|u(x)-u_I(x)| \le \frac{h_i^2}{2}\max_{x\in I_i}|u''(x)| \le \frac{h^2}{2}\max_{x\in \bar{I}}|u''(x)| := \frac{h^2}{2}\|u''\|_{\infty,\bar{I}}$$

$\Rightarrow$

$$\|u - u_I\|_0^2 \leq \frac{b-a}{4} h^4 \|u''\|_{\infty,\bar{I}}^2 \tag{24}$$

接着给出 $\|u' - u_I'\|_0^2$ 的估计. 易知: $\exists \eta_i \in (x_{i-1}, x_i)$, 使得

$$u_I'(\eta_i) = u'(\eta_i). \tag{25}$$

$\Rightarrow$

$$u'(x) = u'(\eta_i) + (x - \eta_i)u''(\gamma_i) = u_I'(x) + (x - \eta_i)u''(\gamma_i), \quad \gamma_i \in I_i$$

$\Rightarrow$

$$|u'(x) - u_I'(x)| \leq h|u''(\gamma_i)| \leq h\|u''\|_{\infty,\bar{I}}$$

$\Rightarrow$

$$\|u' - u_I'\|_0^2 \leq (b-a)h^2\|u''\|_{\infty,\bar{I}}^2$$

由上式和 (24) 可得

$$\|u - u_I\|_1 \leq Ch\|u''\|_{\infty,\bar{I}} = O(h\|u''\|_{\infty,\bar{I}})$$

结合上式和 (23) 可得

$$\|u - u_h\|_1 \leq O(h\|u''\|_{\infty,\bar{I}})$$

进而得到了线性元的收敛性.

下面在一种更弱的光滑性假设条件下给出 $\|u-u_I\|_1$ 的估计式.

设 $u \in H^2(I) \bigcap H_E^1(I)$, 则分段线性插值函数 $u_I(x)$ 满足

$$\|u-u_I\|_1 \leq Ch\|u''\|_0 = O(h\|u''\|_0). \tag{26}$$

进而有

$$\|u-u_h\|_1 \leq O(h\|u''\|_0).$$

(26) 的证明可由以下几步构成.

S1. 利用带积分余项的 Taylor 展开式知：对 $\forall x \in I_i := [x_{i-1}, x_i]$, 有

$$u(x_l) = u(x) + u'(x)(x_l - x) + R_l(x), \quad l = i-1, i \tag{27}$$

其中

$$R_l(x) = \int_x^{x_l} u''(t)(x_l - t)dt.$$

S2. 证明

$$\|u_I - u\|_0 \leq ch^2\|u''\|_0. \tag{28}$$

事实上，将 (27) 代入

$$u_I(x) = u(x_{i-1})N_0(\xi) + u(x_i)N_1(\xi)$$

可得

$$u_I(x) = u(x) + R_{i-1}(x)N_0(\xi) + R_i(x)N_1(\xi)$$

$\Rightarrow$

$$|u_I(x) - u(x)| \leq |R_{i-1}(x)| + |R_i(x)|$$

注意

$$|R_l(x)| \leq h_i \int_x^{x_l} |u''(t)|dt \leq (h_i)^{3/2}\|u''\|_{0,I_i}$$

$\Rightarrow$

$$|u_I(x) - u(x)| \leq c{h_i}^{3/2} \|u''\|_{0,I_i}$$

$\Rightarrow$

$$\|u_I - u\|_{0,I_i}^2 = \int_{x_{i-1}}^{x_i} (u_I - u)^2 dt \leq c{h_i}^4 \|u''\|_{0,I_i}^2$$

$\Rightarrow$

$$\|u_I - u\|_0^2 \leq ch^4 \sum_{i=1}^{n} \|u''\|_{0,I_i}^2 = ch^4 \|u''\|_0^2$$

即 (28) 成立. ■

S3. 证明

$$\|u'_I - u'\|_0 \leq ch\|u''\|_0. \tag{29}$$

事实上，将 (27) 代入

$$u'_I(x) = u(x_{i-1})N'_0(\xi) + u(x_i)N'_1(\xi)$$

有

$$u'_I(x) \quad = \quad u'(x) + R_{i-1}(x)N'_0(\xi) + R_i(x)N'_1(\xi)$$

注意

$$|N'_l(\xi)| = h_i^{-1}, l = 0, 1$$

$\Rightarrow$

$$|u'_I(x) - u'(x)| \leq ch_i^{1/2}\|u''\|_{0,I_i}$$

$\Rightarrow$

$$\|u_I' - u'\|_{0,I_i}^2 \leq ch^2 \|u''\|_{0,I_i}^2$$

$\Rightarrow$

$$\|u_I' - u'\|_0^2 \quad \leq \quad ch^2 \sum_{i=1}^{n} \|u''\|_{0,I_i}^2 = ch^2 \|u''\|_0^2$$

即 (29) 成立. ■

结合 (28) 和 (29) 即完成了 (26) 的证明. ■

2. L^2 范数 ($\|\cdot\|_0$) 下的误差估计

首先给出微分方程先验估计理论.

引理 1 设问题 (19) 中的 $f, q \in C(\bar{I})$, $p \in C^1(\bar{I})$, $p \geq p_{\min} > 0$, $q \geq 0$, $I = (a, b)$，则解函数 $u(x) \in C^2(\bar{I})$, 且满足

$$\|u''\|_0 \leq C\|f\|_0 \tag{30}$$

证明：关于 $u(x) \in C^2(\bar{I})$ 可见常微理论，下面证明 (30)。

$$a(u, v) = (f, v), \quad \forall v \in H_E^1$$

$\Rightarrow$

$$\begin{aligned} \|u\|_1^2 &\leq \gamma^{-1} a(u, u) = \gamma^{-1}(f, u) \\ &\leq \gamma^{-1}\|u\|_0\|f\|_0 \leq \gamma^{-1}\|u\|_1\|f\|_0 \end{aligned}$$

$\Rightarrow$

$$\|u\|_1 \leq \gamma^{-1}\|f\|_0$$

因为函数 $u(x)$ 满足 (19)，即

$$-p(x)u''(x) = p'(x)u'(x) - q(x)u(x) + f(x)$$

$\Rightarrow$

$$\|u''\|_0 \leq C[\|f\|_0 + \|u\|_1] \leq C_2\|f\|_0$$

利用 Nitsche (尼采)技巧（对偶论证法），给出 $\|u-u_h\|_0$ 的估计. 引入辅助函数 $w(x)$，满足

$$a(v,w)=(u-u_h,v),\forall v\in H_E^1$$

特取 $v=u-u_h$，并利用正交投影性质 (20), 有

$$\|u-u_h\|_0^2=(u-u_h,u-u_h)=a(u-u_h,w)=a(u-u_h,w-w_I)$$

进一步，利用双线性泛函的有界性，(26), (7) 和 (30)，可得

$$\begin{aligned}\|u-u_h\|_0^2 &\leq M\|w-w_I\|_1\cdot\|u-u_h\|_1 \leq O(h^2\|w''\|_0\cdot\|u''\|_0)\\ &\leq O(h^2\|u-u_h\|_0\cdot\|f\|_0)\end{aligned}$$

将上式两边同时除以 $\|u-u_h\|_0$ 可得

$$\|u-u_h\|_0 \leq O(h^2\|f\|_0)$$

□

习题 3 设 $u\in C^2(\bar{I})$，试证明 u_h 一致收敛到 u 且收敛阶为 $O(h)$ (即给出 $\|\cdot\|_\infty$ 下的线性有限元函数的误差估计).

线性元稳定性 I

令 $A := K$, 则线性有限元方程组 (12) 可改写为

$$AU = b.$$

下面考察：当系数矩阵 A 和右端向量 b 发生扰动 (设扰动量分别为 ΔA 和 Δb) 时, 对解向量 U 带来的扰动 (设扰动量为 ΔU).

假设 $\mu = \|A^{-1}\Delta A\| < 1$, 则有

$$\frac{\|\Delta U\|}{\|U\|} \leq \frac{cond(A)}{1-\mu}\left(\frac{\|\Delta A\|}{\|A\|} + \frac{\|\Delta b\|}{\|b\|}\right) \tag{31}$$

其中, $cond(A) = \|A\| \cdot \|A^{-1}\|$ 是 A 的条件数.

线性元稳定性 II

事实上，由

$$\begin{cases} Au = b \\ (A+\Delta A)(U+\Delta U) = b+\Delta b \end{cases}$$

$\Rightarrow$

$$\Delta AU + (A+\Delta A)\Delta U = \Delta b$$

$\Rightarrow$

$$\Delta U = (I + A^{-1}\Delta A)^{-1}A^{-1}(-\Delta A \cdot U + \Delta b)$$

$\Rightarrow$

$$\frac{\|\Delta U\|}{\|U\|} \leq \left\|(I + A^{-1}\Delta A)^{-1}\right\| \cdot \left\|A^{-1}\right\| \left(\|\Delta A\| + \frac{\|\Delta b\|}{\|U\|} \right) \tag{32}$$

线性元稳定性 III

注意 $\frac{1}{\|U\|} \leq \frac{\|A\|}{\|b\|}$, 将该式代人(32), 可得

$$\frac{\|\Delta U\|}{\|U\|} \leq \left\|(I + A^{-1}\Delta A)^{-1}\right\| \cdot \left\|A^{-1}\right\| \|A\| \left(\frac{\|\Delta A\|}{\|A\|} + \frac{\|\Delta b\|}{\|b\|}\right)$$

由此，并利用

结论 1 设 $\|F\| < 1$, 则矩阵 $I + F$ 可逆，且有

$$\left\|(I + F)^{-1}\right\| \leq \frac{1}{1 - \|F\|}.$$

可得 (31). ■

线性元稳定性 IV

关于结论 1 的证明：首先用反证法证明矩阵 $I+F$ 的非奇异性。事实上，若存在向量 $x\neq 0$，使 $(I+F)x=0$，则有

$$\|F\|=\max_{x\neq 0}\frac{\|Fx\|}{\|x\|}\geq 1$$

这与题设 $\|F\|<1$ 相矛盾.

令 $G=(I+F)^{-1}$，并在该式两边同乘 $I+F$，有

$$\|G\|\leq 1+\|G\|\cdot\|F\|$$

从而有

$$\|G\|\leq\frac{1}{1-\|F\|}$$

$\blacksquare$

线性元稳定性 V

由 (31) 知: 解向量的扰动与有限元总刚度矩阵 A 的条件数 $cond(A)$ 的大小有关, 下面给出 A 的条件数估计式.

注意 A 对称正定, 所以可取 $\|.\| = \|.\|_2$, 这时 A 的谱条件数

$$cond_2(A) := \|A\|_2 \cdot \left\|A^{-1}\right\|_2 = \frac{\lambda_{\max}}{\lambda_{\min}} \tag{33}$$

其中 $0 < \lambda_{\min} \leq \lambda_{\max}$ 分别为 A 的最小和最大特征值.

下面给出 (33) 的证明.

事实上, 由 A 的对称正定性可知其存在 n 个正特征值

$$0 < \lambda_1 < \cdots < \lambda_n,$$

线性元稳定性 VI

且 $\lambda_{\max} = \lambda_n$, $\lambda_{\min} = \lambda_1$. 设 $\lambda_i (i = 0, \cdots, n)$ 对应的特征向量为

$$\vec{\beta}_i \in \mathbb{R}^{n+1}, \quad i = 0, \cdots, n,$$

易知, $\{\vec{\beta}_0, \cdots, \vec{\beta}_n\}$ 构成 $\mathbb{R}^n$ 的一组标准正交基, 即

$$(\vec{\beta}_i, \vec{\beta}_j) = \delta_{ij}, \quad i, j = 0, \cdots, n. \tag{34}$$

对 $\forall \vec{\alpha} \in \mathbb{R}^{n+1}$, 有

$$\vec{\alpha} = \sum_{i=0}^{n} c_i \vec{\beta}_i. \tag{35}$$

线性元稳定性 VII

由 (35) 和 (34) 可知

$$\begin{aligned}\|A\vec{\alpha}\|_2^2 &= \|\sum_i c_i A\vec{\beta}_i\|_2^2 \\ &= \|\sum_i c_i\lambda_i\vec{\beta}_i\|_2^2 \\ &= (\sum_i c_i\lambda_i\vec{\beta}_i, \sum_i c_i\lambda_i\vec{\beta}_i) \\ &= \sum_i c_i^2\lambda_i^2, \end{aligned} \tag{36}$$

类似可得

$$\|\vec{\alpha}\|_2^2 = (\vec{\alpha}, \vec{\alpha}) = \sum_i c_i^2. \tag{37}$$

线性元稳定性 VIII

注意

$$\|A\|_2 := \sup_{\vec{\alpha}\in R^N} \frac{\|A\vec{\alpha}\|_2}{\|\vec{\alpha}\|_2}, \tag{38}$$

这里 $\|\vec{\alpha}\|_2 := \sqrt{(\vec{\alpha},\vec{\alpha})_{R^N}}$.
将 (36) 和 (37) 代入 (38)，有

$$\|A\|_2^2 = \sup_{\vec{\alpha}\in\mathbb{R}^n} \frac{\sum\limits_i c_i^2\lambda_i^2}{\sum\limits_i c_i^2} \leq \lambda_{\max}^2(A), \tag{39}$$

又注意：若特取 $\vec{\alpha} = \vec{\beta}_n$，则上式成为等式，因此有

$$\|A\|_2^2 = \lambda_{\max}^2(A). \tag{40}$$

线性元稳定性 IX

同理由

$$\|A^{-1}\|_2^2 = \sup_{\vec{\alpha}\in\mathbb{R}^{n+1}} \frac{\sum\limits_i c_i^2\lambda_i^{-2}}{\sum\limits_i c_i^2} \le \lambda_{\min}^{-2}(A), \tag{41}$$

且特取 $\vec{\alpha} = \vec{\beta}_1$, 上式成为等式, 因此有

$$\|A^{-1}\|_2^2 = \lambda_{\min}^{-2}(A). \tag{42}$$

将 (40) 和 (42) 代入 $cond_2(A)$ 的定义式, 即证得 (33). ■

进一步可证得

$$\lambda_{\max} = \max_{\substack{\vec{\alpha}\in R^n \\ \vec{\alpha}\neq 0}} R(\vec{\alpha}), \quad \lambda_{\min} = \min_{\substack{\vec{\alpha}\in R^n \\ \vec{\alpha}\neq 0}} R(\vec{\alpha}) \tag{43}$$

线性元稳定性 X

其中

$$R(\vec{\alpha}) := \frac{(A\vec{\alpha}, \vec{\alpha})}{(\vec{\alpha}, \vec{\alpha})}$$

为矩阵 A 关于向量 $\vec{\alpha}$ 的雷利 (Rayleigh) 商.

下面给出 (43) 的证明.

事实上，类似于 (36) 的证明, 利用 (35), (34) 和 (37)，有

$$\frac{(A\vec{\alpha}, \vec{\alpha})}{(\vec{\alpha}, \vec{\alpha})} = \frac{(\sum\limits_i c_i \lambda_i \vec{\beta}_i, \sum\limits_i c_i \vec{\beta}_i)}{\sum\limits_i c_i^2} = \frac{\sum\limits_{i=1}^{n} \lambda_i c_i^2}{\sum\limits_i c_i^2}.$$

由此可得

$$\max_{\substack{\vec{\alpha}\in R^n \\ \vec{\alpha}\neq 0}} R(\vec{\alpha}) \leq \lambda_{\max},$$

线性元稳定性 XI

又注意：若特取 $\vec{\alpha} = \vec{\beta}_n$，则上式成为等式，因此 (43) 的第一个等式成立. 类似可证明(43) 的第二个等式亦成立. ■

对 $\forall \vec{\alpha} \in \mathbb{R}^{n+1}$，令线性有限元函数

$$u_h = \sum_{i=1}^{n} \alpha_i \phi_i, \tag{44}$$

则有

$$(A\vec{\alpha}, \vec{\alpha}) = a(u_h, u_h). \tag{45}$$

线性元稳定性 XII

事实上,

$$
\begin{aligned}
A\vec{\alpha} &= (\sum_{j=1}^{n} a_{1,j}\alpha_j, \sum_{j=1}^{n} a_{2,j}\alpha_j, \cdots, \sum_{j=1}^{n} a_{n,j}\alpha_j)^T \\
&= (\sum_{j=1}^{n} a(\phi_1,\phi_j)\alpha_j, \cdots, \sum_{j=1}^{n} a(\phi_n,\phi_j)\alpha_j)^T \\
&= (a(\phi_1, \sum_{j=1}^{n} \alpha_j\phi_j), \cdots, a(\phi_n, \sum_{j=1}^{n} \alpha_j\phi_j))^T \\
&= (a(\phi_1, u_h), \cdots, a(\phi_n, u_h))^T,
\end{aligned}
$$

因此

$$
(A\vec{\alpha}, \vec{\alpha}) = \sum_{i=1}^{n} \alpha_i a(\phi_i, u_h) = a(\sum_{i=1}^{n} \alpha_i\phi_i, u_h) = a(u_h, u_h).
$$

线性元稳定性 XIII

$\blacksquare$

利用 (45) 和 $a(\cdot,\cdot)$ 的强制性条件，显然有

$$\gamma\|u_h\|_1^2 \le (A\vec{\alpha},\vec{\alpha}) \le M\|u_h\|_1^2. \tag{46}$$

下面在等距网格剖分下，证明有限元函数的两个重要估计式.

首先证明：对 $\forall\vec{\alpha} := (u_1,\cdots,u_n)^T \in \mathbb{R}^n$，有 (令 $u_0=0$)

$$\|u_h\|_0^2 \cong h\|\vec{\alpha}\|_{\mathbb{R}^n}^2, \tag{47}$$

其中，线性有限元函数 u_h 由 (44) 定义.

线性元稳定性 XIV

事实上，由 $\|\cdot\|_0$ 范数的定义有

$$
\begin{aligned}
\|u_h\|_0^2 &= \int_a^b u_h^2 dx \\
&= \sum_{i=1}^n \int_{I_i} [u_{i-1}N_0(\xi) + u_i N_1(\xi)]^2 dx \quad (\xi = \frac{x - x_{i-1}}{h}) \\
&= \sum_{i=1}^n h_i \int_0^1 [u_{i-1}(1-\xi) + u_i \xi]^2 d\xi,
\end{aligned}
\tag{48}
$$

线性元稳定性 XV

其中

$$\int_0^1 [u_{i-1}(1-\xi)+u_i\xi]^2 d\xi = \int_0^1 u_{i-1}^2(1-\xi)^2 + 2u_{i-1}u_i\xi(1-\xi) + u_i^2\xi^2 d\xi$$
$$= \frac{1}{3}(u_{i-1}^2 + u_{i-1}u_i + u_i^2), \tag{49}$$

利用不等式 $-\frac{a^2+b^2}{2} \le ab \le \frac{a^2+b^2}{2}$, 有

$$\frac{u_{i-1}^2 + u_i^2}{2} \le u_{i-1}^2 + u_{i-1}u_i + u_i^2 \le \frac{3}{2}(u_{i-1}^2 + u_i^2),$$

将上式和 (49) 代入 (48) 中有

$$\frac{h}{6}\sum_{i=1}^n u_i^2 \le \|u_h\|_0^2 \le h(\sum_{i=0}^{n-1}\frac{u_i^2}{2} + \sum_{i=0}^{n}\frac{u_i^2}{2}),$$

线性元稳定性 XVI

由于 $\|\vec{u}\|^2_{\mathbb{R}^n} = \sum\limits_{i=0}^{n} u_i^2$, 则有

$$\frac{h}{6}\|u_h\|^2_{\mathbb{R}^{n+1}} \leq \|u_h\|^2_0 \leq h\|u_h\|^2_{\mathbb{R}^{n+1}},$$

这就证明了

$$\|u_h\|^2_0 \cong h\|\vec{u}\|^2_{\mathbb{R}^{n+1}}.$$

$\blacksquare$

接着证明有限元的反估计式:

$$\|u_h\|^2_1 \lesssim h^{-2}\|u_h\|^2_0. \tag{50}$$

线性元稳定性 XVII

事实上，由 $\|\cdot\|_1$ 范数的定义可知

$$\|u_h\|_1^2 = \|u_h\|_0^2 + |u_h|_1^2,$$

注意 $h < 1$, 因此 (50) 等价于证明

$$|u_h|_1^2 \lesssim h^{-2}\|u_h\|_0^2,$$

线性元稳定性 XVIII

而

$$
\begin{aligned}
|u_h|_1^2 &= \sum_{i=1}^{n} \int_{I_i} [u_{i-1} N_0'(\xi) + u_i N_1'(\xi)]^2 dx \quad (\xi = \frac{x - x_{i-1}}{h_i}) \\
&= \sum_{i=1}^{n} \int_{I_i} [u_{i-1} \frac{\partial N_0}{\partial \xi} \frac{\partial \xi}{\partial x} + u_i \frac{\partial N_1}{\partial \xi} \frac{\partial \xi}{\partial x}]^2 dx \\
&= \sum_{i=1}^{n} h_i \int_0^1 h_i^{-2} (u_i - u_{i-1})^2 d\xi \\
&= \sum_{i=1}^{n} h_i^{-1} (u_i^2 + u_{i-1}^2 - 2u_{i-1} u_i) \\
&\lesssim h^{-1} \sum_{i=1}^{n} (u_i^2 + u_{i-1}^2) \lesssim h^{-1} \|u_h\|_{\mathbb{R}^n}^2
\end{aligned}
$$

将 (47) 代入上式可知

$$|u_h|_1^2 \lesssim h^{-2}\|u\|_0^2,$$

至此就证明了 (50).

■

有了上述准备就可给出 A 的谱条件数估计.