

## Poisson 方程 Dirichlet 边值问题的有限元方法

阙嘉豪<sup>1</sup>

(1. 北京师范大学 数学科学学院, 北京 100875)

## 1 模型问题

考虑  $\Omega := (0, 1) \times (0, 1)$  上的 Poisson 方程的 Dirichlet 边值问题:

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

## 2 Galerkin 方法

问题 (1) 相应于虚功原理的弱解的提法为

$$\begin{cases} \text{求 } u \in \mathbb{H}_0^1(\Omega), \text{ 使得} \\ a(u, v) = (f, v), \quad \forall v \in \mathbb{H}_0^1(\Omega), \end{cases} \quad (2)$$

其中

$$a(u, v) = \int_{\Omega} \nabla u \cdot \nabla v d\mathbf{x}, \quad (f, v) = \int_{\Omega} f v d\mathbf{x}.$$

通常将问题 (2) 的允许函数空间称为试探函数空间.

为了数值求解问题 (2), 我们给出一个有限维的子空间  $\mathbb{V}_h(0) \subset \mathbb{H}_0^1(\Omega)$ , 将问题的试探函数空间和检验函数空间分别换成其子空间  $\mathbb{V}_h(0) \subset \mathbb{H}_0^1(\Omega)$ , 就给出了该问题相应的近似解的提法:

$$\begin{cases} \text{求 } u_h \in \mathbb{V}_h(0), \text{ 使得} \\ a(u_h, v_h) = (f, v_h), \quad \forall v_h \in \mathbb{V}_h(0). \end{cases} \quad (3)$$

通常将该提法称为 Galerkin 方法, 下面将其化为线性代数方程组求解.

设  $\{\varphi_i\}_{i=1}^{N_h}$  是  $\mathbb{V}_h(0)$  的一组基, 令

$$u_h = \sum_{j=1}^{N_h} u_j \varphi_j, \quad v_h = \sum_{i=1}^{N_h} v_i \varphi_i,$$

则离散问题 (3) 等价于

$$\begin{cases} \text{求 } \mathbf{u}_h = (u_1, \dots, u_{N_h})^T \in \mathbb{R}^{N_h}, \text{ 使得} \\ \sum_{i,j=1}^{N_h} a(\varphi_j, \varphi_i) u_j v_i = \sum_{i=1}^{N_h} (f, \varphi_i) v_i, \quad \forall \mathbf{v}_h = (v_1, \dots, v_{N_h})^T \in \mathbb{R}^{N_h}. \end{cases}$$

由  $\mathbf{v}_h$  的任意性, 这等价于求解线性代数方程组

$$\sum_{j=1}^{N_h} a(\varphi_j, \varphi_i) u_j = (f, \varphi_i), \quad i = 1, \dots, N_h. \quad (4)$$

通常将线性代数方程组 (4) 写为矩阵形式

$$\mathbf{K} \mathbf{u}_h = \mathbf{f}, \quad (5)$$

称  $\mathbf{K} := (k_{ij}) := (a(\varphi_j, \varphi_i))$  为刚度矩阵,  $\mathbf{u}_h$  为位移向量,  $\mathbf{f} := (f_i) := ((f, \varphi_i))$  为载荷向量. 由  $a(\cdot, \cdot)$  的对称性和 Poincaré-Friedrichs 不等式, 容易证明刚度矩阵  $\mathbf{K}$  是对称正定矩阵. 由此也可以证明线性代数方程组 (4) 的解的存在唯一性.

### 3 有限元方法

下面构造有限维的子空间  $\mathbb{V}_h(0) \subset \mathbb{H}_0^1(\Omega)$  以及选取  $\mathbb{V}_h(0)$  上的一组基, 并给出刚度矩阵和载荷向量的求解方法.

#### 3.1 三角形剖分

对  $\bar{\Omega}$  做如图 1 所示的三角形剖分  $\mathfrak{T}_h(\Omega)$ , 即将其分割为  $M = 2L^2$  个互相之间没有公共内点的三角形单元  $T_i, i = 1, \dots, M$ , 其中参数  $h := \max_{1 \leq i \leq M} \sup_{x, y \in T_i} \|x - y\|$  是剖分中所有单元的最大直径. 将剖分的  $N = (L + 1)^2$  个节点作整体编号  $A_i, i = 1, \dots, N$ , 公共的节点共用一个整体编号, 并将节点分为边界节点和内部节点两类.

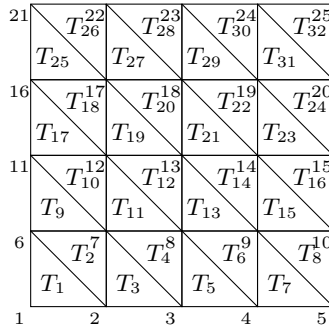


图 1:  $L = 4$  时的三角形剖分

#### 3.2 有限元函数空间

在三角形剖分  $\mathfrak{T}_h(\Omega)$  的基础上定义有限元函数空间

$$\mathbb{V}_h := \left\{ u \in C(\bar{\Omega}) : u|_{T_i} \in \mathbb{P}_1(T_i), \forall T_i \in \mathfrak{T}_h(\Omega) \right\},$$

并根据齐次强制边界条件取有限元试探函数空间和检验函数空间为

$$\mathbb{V}_h(0) := \{u \in \mathbb{V}_h : u(A_i) = 0, \forall A_i \in \partial\Omega\},$$

其中  $\mathbb{P}_k(T_i)$  表示定义在  $T_i$  上的所有次数不超过  $k$  的多项式所构成的线性空间.  $\mathbb{V}_h$  是  $\mathbb{H}^1(\Omega)$  的子空间,  $\mathbb{V}_h(0)$  是  $\mathbb{H}_0^1(\Omega)$  的子空间.

### 3.3 单元基底函数

由于  $\mathbb{V}_h$  和  $\mathbb{V}_h(0)$  都是由分片线性的连续函数构成的, 易证  $\mathbb{V}_h$  和  $\mathbb{V}_h(0)$  中的任意一个函数都由其节点函数值  $\{u(A_i)\}_{i=1}^N$  唯一确定. 设  $\varphi_i \in \mathbb{V}_h$  满足

$$\varphi_i(A_j) = \delta_{ij}, \quad i = 1, \dots, N,$$

则  $\{\varphi_i\}_{i=1}^N$  组成了  $\mathbb{V}_h$  的一组基, 而  $\{\varphi_i\}_{A_i \notin \partial\Omega}$  组成了  $\mathbb{V}_h(0)$  的一组基.

**注 1** 上述有限元函数空间的基底函数的支集很小, 由有限元方法得到的刚度矩阵为稀疏矩阵, 其零元素的分布依赖于节点的排序. 尽管排序并不影响解的性质, 但适当的节点排序有利于提高数值求解线性代数方程的效率.

### 3.4 单元刚度矩阵和单元载荷向量

建立了有限元函数空间  $\mathbb{V}_h(0)$  后, 我们就可以计算刚度矩阵  $\mathbf{K}$  和载荷向量  $\mathbf{f}$ . 首先需要计算单元刚度矩阵和单元载荷向量, 然后进行组装. 为了便于数据管理, 引入以下两个数组:

- $en(\alpha, e)$ : 第  $e$  个单元的第  $\alpha$  个节点对应于第  $en(\alpha, e)$  个整体节点;
- $cd(i, nd)$ : 第  $nd$  个整体节点的空间坐标的第  $i$  个分量.

记

$$a^e(u, v) := \int_{T_e} \nabla u \cdot \nabla v d\mathbf{x},$$

由定义有

$$k_{ij} := a(\varphi_j, \varphi_i) = \sum_{e=1}^M a^e(\varphi_j, \varphi_i) = \sum_{e=1}^M \int_{T_e} \nabla \varphi_j \cdot \nabla \varphi_i d\mathbf{x} = \sum_{e=1}^M k_{ij}^e.$$

由于有大量的  $k_{ij}$  为零, 而且即便  $k_{ij} \neq 0$ , 上式的几个求和式中也只有少数几项非零, 并且只有  $i, j$  两个节点同属一个单元  $T_e$  时  $\int_{T_e} \nabla \varphi_j \cdot \nabla \varphi_i d\mathbf{x} \neq 0$ . 因此通常不采取扫描节点  $i, j$  的方式计算  $k_{ij}$  而是通过扫描单元的方式计算.

对于每一个单元  $T_e$  定义单元刚度矩阵

$$\mathbf{K}^e := (k_{\alpha\beta}^e), \quad k_{\alpha\beta}^e := a^e(\varphi_\alpha^e, \varphi_\beta^e) = \int_{T_e} \nabla \varphi_\alpha^e \cdot \nabla \varphi_\beta^e d\mathbf{x},$$

则有

$$k_{ij} = \sum_{\substack{en(\alpha,e)=i \in T_e \\ en(\beta,e)=j \in T_e}} k_{\alpha\beta}^e.$$

同理也可以通过扫描单元来计算载荷向量  $\mathbf{f} := (f_i)$ :

$$f_i = \sum_{en(\alpha,e)=i \in T_e} \int_{T_e} f \lambda_{\alpha}^e d\mathbf{x} =: \sum_{en(\alpha,e)=i \in T_e} f_{\alpha}^e.$$

我们称  $\mathbf{f} := (f_{\alpha}^e)$  为单元载荷向量.

下面通过变量替换将单元刚度矩阵和单元载荷向量的计算转化到标准三角形上.

设  $\alpha = 1, 2, 3$  为  $T_e$  的三个节点  $A_i, A_j, A_k$  的局部节点序号, 即在单元  $T_e$  上这三个节点分别记作  $A_1^e, A_2^e$  和  $A_3^e$ . 设  $\lambda_{\alpha}^e \in \mathbb{P}_1(T_e)$  满足  $\lambda_{\alpha}^e(A_{\beta}^e) = \delta_{\alpha\beta}$ , 则有

$$\lambda_{\alpha}^e(A) = \frac{|\nabla A A_{\beta}^e A_{\gamma}^e|}{|\nabla A_{\alpha}^e A_{\beta}^e A_{\gamma}^e|}, \quad \forall A \in T_e,$$

其中  $|T|$  为集合  $T$  的面积. 通常称  $\lambda^e(A) := (A_1^e(A), A_2^e(A), A_3^e(A))^T$  为  $A \in T_e$  的面积坐标. 易证  $\lambda_{\alpha}^e = \varphi_{en(\alpha,e)}|_{T_e}$ , 且  $\{\lambda_{\alpha}^e\}_{\alpha=1}^3$  为  $\mathbb{P}_1(T_e)$  的一组基. 事实上, 对任意  $u \in \mathbb{P}_1(T_e)$ , 有

$$u(A) = \sum_{\alpha=1}^3 u(A_{\alpha}^e) \lambda_{\alpha}^e(A), \quad \forall A \in T_e.$$

所有的单元  $T_e$  都与标准三角形

$$T_s := \{\hat{\mathbf{x}} = (\hat{x}_1, \hat{x}_2) \in \mathbb{R}^2 : \hat{x}_1 \geq 0, \hat{x}_2 \geq 0, \hat{x}_1 + \hat{x}_2 \leq 1\}$$

仿射等价, 即对任意的三角形单元  $T_e$ , 存在可逆矩阵  $\mathbf{A}_e \in \mathbb{R}^{2 \times 2}$  和向量  $\mathbf{a}_e$ , 使得仿射变换

$$\mathbf{x} = L_e(\hat{\mathbf{x}}) := \mathbf{A}_e \hat{\mathbf{x}} + \mathbf{a}_e$$

将  $T_s$  映射到  $T_e$ . 同时仿射变换  $L_e(\hat{\mathbf{x}})$  也诱导了  $\mathbb{P}_1(T_s)$  和  $\mathbb{P}_1(T_e)$  之间的一个一一对应关系. 将  $T_s$  的三个顶点坐标记为

$$\mathbf{A}_1^s = (0, 0)^T, \quad \mathbf{A}_2^s = (1, 0)^T, \quad \mathbf{A}_3^s = (0, 1)^T,$$

容易得到  $T_s$  的面积坐标为

$$\lambda_1^s(\hat{x}_1, \hat{x}_2) = 1 - \hat{x}_1 - \hat{x}_2, \quad \lambda_2^s(\hat{x}_1, \hat{x}_2) = \hat{x}_1, \quad \lambda_3^s(\hat{x}_1, \hat{x}_2) = \hat{x}_2.$$

若  $T_e$  的三个顶点的坐标为

$$\mathbf{A}_1^e = (x_1^1, x_2^1)^T, \quad \mathbf{A}_2^e = (x_1^2, x_2^2)^T, \quad \mathbf{A}_3^e = (x_1^3, x_2^3)^T,$$

则有

$$\mathbf{A}_e = (\mathbf{A}_2^e - \mathbf{A}_1^e, \mathbf{A}_3^e - \mathbf{A}_1^e), \quad \mathbf{a}_e = \mathbf{A}_1^e.$$

由  $\lambda_\alpha^s(L_e^{-1}(\mathbf{x}))$  是  $\mathbf{x}$  的线性函数及  $\lambda_\alpha^s(L_e^{-1}(\mathbf{A}_\beta^e)) = \lambda_\alpha^s(\mathbf{A}_\beta^s) = \delta_{\alpha\beta}$  知,  $T_e$  的面积坐标可以用  $T_s$  的面积坐标表示为  $\lambda_\alpha^e(\mathbf{x}) = \lambda_\alpha^s(L_e^{-1}(\mathbf{A}_\beta^e))$ . 由此得

$$\begin{aligned} \hat{\mathbf{x}} &= L_e^{-1}(\mathbf{x}) = \mathbf{A}_e^{-1}\mathbf{x} - \mathbf{A}_e^{-1}\mathbf{A}_1^e, \\ \nabla\lambda^e(\mathbf{x}) &= \nabla\lambda^s(\hat{\mathbf{x}})\nabla L_e^{-1}(\mathbf{x}) = \nabla\lambda^s(\hat{\mathbf{x}})\mathbf{A}_e^{-1}. \end{aligned}$$

因此, 可以将单元刚度矩阵  $\mathbf{K}^e$  和单元载荷向量  $\mathbf{f}$  的计算统一在标准三角形  $T_s$  上进行, 即

$$\begin{aligned} \mathbf{K}^e &= \int_{T_e} \nabla\lambda^e(\mathbf{x})(\nabla\lambda^e(\mathbf{x}))^T d\mathbf{x} \\ &= \int_{T_s} \nabla\lambda^s(\hat{\mathbf{x}})\mathbf{A}_e^{-1}(\nabla\lambda^s(\hat{\mathbf{x}})\mathbf{A}_e^{-1})^T \det \mathbf{A}_e d\hat{\mathbf{x}}. \end{aligned}$$

由此及

$$\nabla\lambda^s(\hat{\mathbf{x}}) = \begin{pmatrix} -1 & -1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \mathbf{A}_e^{-1} = \frac{1}{\det \mathbf{A}_e} \begin{pmatrix} x_2^3 - x_2^1 & x_1^1 - x_1^3 \\ x_2^1 - x_2^2 & x_1^2 - x_1^1 \end{pmatrix}$$

并注意到  $T_s$  的面积为  $\frac{1}{2}$ , 就立即得到单元刚度矩阵为

$$\mathbf{K}^e = \frac{1}{2 \det \mathbf{A}_e} \begin{pmatrix} x_2^2 - x_2^3 & x_1^3 - x_1^2 \\ x_2^3 - x_2^1 & x_1^1 - x_1^3 \\ x_2^1 - x_2^2 & x_1^2 - x_1^1 \end{pmatrix} \begin{pmatrix} x_2^2 - x_2^3 & x_2^3 - x_2^1 & x_2^1 - x_2^2 \\ x_1^3 - x_1^2 & x_1^1 - x_1^3 & x_1^2 - x_1^1 \end{pmatrix}.$$

类似地可以得到单元载荷向量的计算公式

$$\mathbf{f}^e = \int_{T_e} f(\mathbf{x})\lambda^e(\mathbf{x})d\mathbf{x} = \det \mathbf{A}_e \int_{T_s} f(L_e(\hat{\mathbf{x}}))\lambda^s(\hat{\mathbf{x}})d\hat{\mathbf{x}}. \quad (6)$$

一般地, 单元载荷向量的计算要用到数值积分公式. 不过, 当  $f$  为分片多项式时, 可以通过 (6) 式右端的积分给出直接的计算公式. 例如,  $f$  在  $T_e$  上为常数时, 有

$$\mathbf{f}^e = \frac{1}{6}f(T_e)\det \mathbf{A}_e(1, 1, 1)^T = \frac{1}{3}f(T_e)|T_e|(1, 1, 1)^T,$$

其中  $|T_e|$  为  $T_e$  的面积. 而  $f$  为分片线性时, 设  $f|_{T_e} = \sum_{i=1}^3 f_i^e \lambda_i^e(\mathbf{x})$ , 有

$$\mathbf{f}^e = \frac{|T_e|}{12} \begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix} \begin{pmatrix} f_1^e \\ f_2^e \\ f_3^e \end{pmatrix}.$$

至此, 我们完成了单元刚度矩阵和单元载荷向量的计算.

### 3.5 组装刚度矩阵和载荷向量

下面组装刚度矩阵和载荷向量, 算法如下.

**算法 1** 形成刚度矩阵和载荷向量的算法

```

1:  $\mathbf{K} = (k(i, j)) \leftarrow 0, \mathbf{f} = (f(i)) \leftarrow 0.$ 
2: for  $e \leftarrow 1$  to  $M$  do
3:   计算单元刚度矩阵  $\mathbf{K}^e = (k^e(\alpha, \beta)).$ 
4:   计算单元载荷向量  $\mathbf{f}^e = (f^e(\alpha)).$ 
5:    $k(en(\alpha, e), en(\beta, e)) \leftarrow k(en(\alpha, e), en(\beta, e)) + k^e(\alpha, \beta).$ 
6:    $f(en(\alpha, e)) \leftarrow f(en(\alpha, e)) + f^e(\alpha).$ 
7: end for
    
```

由单元刚度矩阵和单元载荷向量得到总的刚度矩阵和载荷向量, 便可根据 (5) 式建立与原问题等价的有限元离散线性代数方程组, 简称有限元方程.

## 4 数值实验

在问题 (1) 中取

$$f(x, y) := -2(x(x-1) + y(y-1)),$$

对应的解析解为

$$u(x, y) = x(x-1)y(y-1).$$

不同单元直径时的  $\mathbb{L}^2, \mathbb{L}^\infty$  误差及收敛阶如表 1 所示. 可以看出, 随着  $h$  的减小, 误差的  $\mathbb{L}^2, \mathbb{L}^\infty$  范数逐渐减小, 收敛阶逐渐接近于 2.

表 1: 不同单元直径时的  $\mathbb{L}^2, \mathbb{L}^\infty$  误差及收敛阶

$\mathbb{L}^2$ 收敛阶	$\mathbb{L}^2$ 误差	$h$	$\mathbb{L}^\infty$ 误差	$\mathbb{L}^\infty$ 收敛阶
	$1.56250 \times 10^{-2}$	$2^{-1}$	$3.12500 \times 10^{-2}$	
1.36505	$6.06597 \times 10^{-3}$	$2^{-2}$	$9.76563 \times 10^{-3}$	1.67807
1.82603	$1.71084 \times 10^{-3}$	$2^{-3}$	$2.56348 \times 10^{-3}$	1.92961
1.94479	$4.44394 \times 10^{-4}$	$2^{-4}$	$6.48499 \times 10^{-4}$	1.98293
1.98011	$1.12641 \times 10^{-4}$	$2^{-5}$	$1.62601 \times 10^{-4}$	1.99576
1.99196	$2.83175 \times 10^{-5}$	$2^{-6}$	$4.06802 \times 10^{-5}$	1.99894
1.99645	$7.09678 \times 10^{-6}$	$2^{-7}$	$1.01719 \times 10^{-5}$	1.99974
1.99835	$1.77623 \times 10^{-6}$	$2^{-8}$	$2.54309 \times 10^{-6}$	1.99993
1.99920	$4.44303 \times 10^{-7}$	$2^{-9}$	$6.35780 \times 10^{-7}$	1.99998
1.99961	$1.11106 \times 10^{-7}$	$2^{-10}$	$1.58945 \times 10^{-7}$	2.00000
1.99985	$2.77792 \times 10^{-8}$	$2^{-11}$	$3.97344 \times 10^{-8}$	2.00007

## 附录

数值实验源码