

曲面有限元及其应用

刘江刚 龚欣

November 5, 2018

基础知识

梯度

首先给出梯度的概念,它是由数量函数 $u(x, y, z)$ 所定义的向量函数

$$\text{grad } u = \left(\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}, \frac{\partial u}{\partial z} \right)$$

而且 $\text{grad } u$ 的方向就是使 $\frac{\partial u}{\partial l}$ 达到最大值的方向,它的大小就是 u 在这个方向上的方向导数.

引进符号向量

$$\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right)$$

当把它作为运算符号来看待时,梯度可写作

$$\text{grad } u = \nabla u$$

关于梯度,有以下一些基本性质:

1. 若 u, v 是数量函数,则

$$\nabla(u + v) = \nabla u + \nabla v$$

2. 若 u, v 是数量函数,则

$$\nabla(uv) = u(\nabla v) + v(\nabla u)$$

3. 若 $u(x, y, z)$ 是数量函数,则 $\nabla^2 u(x) \in R^{3 \times 3}$ 为 $u(x)$ 的 Hessian 矩阵

$$\nabla^2 u(x) = \begin{pmatrix} \frac{\partial^2 u}{\partial x^2} & \frac{\partial^2 u}{\partial x \partial y} & \frac{\partial^2 u}{\partial x \partial z} \\ \frac{\partial^2 u}{\partial y \partial x} & \frac{\partial^2 u}{\partial y^2} & \frac{\partial^2 u}{\partial y \partial z} \\ \frac{\partial^2 u}{\partial z \partial x} & \frac{\partial^2 u}{\partial z \partial y} & \frac{\partial^2 u}{\partial z^2} \end{pmatrix}$$

散度

设

$$\mathbf{A}(x, y, z) = (P(x, y, z), Q(x, y, z), R(x, y, z))$$

为空间区域 V 上的向量函数, 对 V 上每一点 (x, y, z) , 定义数量函数

$$D(x, y, z) = \frac{\partial P}{\partial x} + \frac{\partial Q}{\partial y} + \frac{\partial R}{\partial z}$$

称它为向量函数 \mathbf{A} 在 (x, y, z) 处的散度, 记作

$$D(x, y, z) = \operatorname{div} \mathbf{A}(x, y, z)$$

由前面引进的算符 ∇, \mathbf{A} 的散度的向量形式是

$$\operatorname{div} \mathbf{A} = \nabla \cdot \mathbf{A}$$

关于散度, 有以下一些基本性质:

1. 若 \mathbf{u}, \mathbf{v} 是向量函数, 则

$$\nabla \cdot (\mathbf{u} + \mathbf{v}) = \nabla \cdot \mathbf{u} + \nabla \cdot \mathbf{v}$$

2. 若 φ 是数量函数, \mathbf{F} 是向量函数, 则

$$\nabla \cdot (\varphi \mathbf{F}) = \varphi \nabla \cdot \mathbf{F} + \mathbf{F} \cdot \nabla \varphi$$

3. 若 $\varphi = \varphi(x, y, z)$ 是一数量函数, 则

$$\nabla \cdot \nabla \varphi = \frac{\partial^2 \varphi}{\partial x^2} + \frac{\partial^2 \varphi}{\partial y^2} + \frac{\partial^2 \varphi}{\partial z^2}$$

算符 ∇ 的内积 $\nabla \cdot \nabla$ 常记作 Δ , 于是有

$$\nabla \cdot \nabla \varphi = \Delta \varphi$$

预备知识

连续的曲面 S

设 S 是一个闭曲面, 所以 $\partial S = \emptyset$, 所以 S 把 \mathbb{R}^3 分成三个不同的集合: 曲面内部的点、表面上的点和曲面外部的点, 分别表示为 Ω_- 、 Ω_0 和 Ω_+ . 对于任意的 $x \in \mathbb{R}^3$, 记 $\operatorname{dist}(x, S) = \min_{y \in S} |x - y|$ 为 x 和 S 之间的距离, 其中 $|\cdot|$ 为标准的欧氏距离. 我们可以定义一个带状区域: $U := \{x \in \mathbb{R}^3 | \operatorname{dist}(x, S) < \delta\}$, 其中

$\delta > 0$ 且要足够小, 使得可以定义一个唯一的符号距离函数 $d: U \rightarrow \mathbb{R}$, 满足如下性质:

$$\begin{cases} d \in C^3(U), \\ |d(x)| = \text{dist}(x, S), \quad \forall x \in U, \\ d(x) < 0, \quad \forall x \in \Omega_- \cap U, \\ d(x) = 0, \quad \forall x \in \Omega_0 \cap U, \\ d(x) > 0, \quad \forall x \in \Omega_+ \cap U, \end{cases} \quad (1)$$

对于任意的 $x \in U$, 可视 S 为符号距离函数的零水平集.

记 ∇ 为 \mathbb{R}^3 中通常意义下的梯度算子, $\nabla d(x) \in \mathbb{R}^3$ 为 $d(x)$ 的梯度, $\mathbf{H}(x) := \nabla^2 d(x) \in \mathbb{R}^{3 \times 3}$ 为 $d(x)$ 的 Hessian 矩阵. 对于任意的 $x \in U$, 记 y 为 S 上离 x 最近的点, 也即 $|d(x)| = |x - y|$. 因为 $d(x)$ 是符号距离函数, 且 S 是它的零水平集, 易知 $\nabla d(x)$ 是 S 在 y 处的单位外法向向量, 即 $|\nabla d(x)| = 1$. 对于任意的 $x \in U$, 记 $\mathbf{n}(x) = \nabla d(x)$, 我们可以定义如下唯一的投影 $\mathcal{P}_0: U \rightarrow S$:

$$\mathcal{P}_0(x) := x - d(x)\mathbf{n}(x) \quad (1)$$

对于 $v \in C^1(S)$, 因为 S 是 C^3 的, 我们可以把 v 扩展到 $C^1(U)$, 且仍记为 v . 定义 v 在 S 上的切向梯度为

$$\nabla_S v = \nabla v - (\nabla v \cdot \mathbf{n})\mathbf{n} = (I - \mathbf{n}\mathbf{n}^t)\nabla v = \mathbf{P}\nabla v \in \mathbb{R}^3$$

其中 $\mathbf{P}(x) = (I - \mathbf{n}\mathbf{n}^t)(x)$ 是到点 $x \in S$ 切平面上的投影算子, 因此有 $\mathbf{P}^2 = \mathbf{P}$. 注意到这里我们用 v 的扩展来定义曲面梯度. 然而, 可以证明 $\nabla_S v$ 的定义只依赖 v 的 S 上的值而不是 v 的扩展, 也即 ∇_S 是一个内蕴算子.

类似地, 对于一个向量场 $\mathbf{v} \in (C^1(S))^3$, 我们也可以把它扩展到 $(C^1(U))^3$ 上去, 并定义 \mathbf{v} 在 S 上的切向散度为

$$\nabla_S \cdot \mathbf{v} = \nabla \cdot \mathbf{v} - \mathbf{n}^t \nabla \mathbf{v} \mathbf{n} \in \mathbb{R}$$

曲面 S 上的 Laplace-Beltrami 算子定义如下:

$$\Delta_S v = \nabla_S \cdot (\nabla_S v) = \Delta v - (\nabla v \cdot \mathbf{n})(\nabla \cdot \mathbf{n}) - \mathbf{n}^t \nabla^2 v \mathbf{n} \in \mathbb{R}$$

其中 $v \in C^2(S)$, $\nabla^2 v$ 是 v 的 Hessian 矩阵 (扩展为 $C^2(U)$ 函数)

离散的曲面 S_h 和网格 \mathcal{T}_h

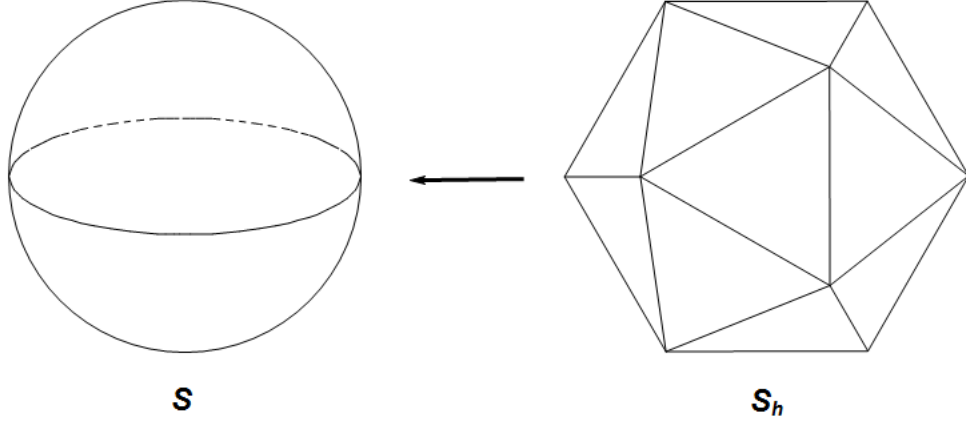


图 1

记 $S_h \subset U$ 是由三角形单元 τ_h 组成的多面体面, 即 S_h 逼近曲面 S , 我们假设 S_h 的这些三角形单元是正则的, 单元尺寸是拟一致的, 且它们的顶点都在曲面 S 上. 其中 $\mathcal{N}_h = \{x_i\}$ 为 S_h 所有顶点的集合, $\mathcal{T}_h = \{\tau_h\}$ 为所有三角形单元的集合, $\mathcal{E}_h = \{E\}$ 为所有边的集合. 对于任意的 $\tau_h \in \mathcal{T}_h$, 记 \mathbf{n}_h 为 S_h 在 τ_h 上的单位外法向向量. 对于 $v_h \in C(S_h)$ 和 $v_h|_{\tau_h}$, 我们有

$$\nabla_{S_h} v_h|_{\tau_h} := \nabla v_h - (\nabla v_h \cdot \mathbf{n}_h) \mathbf{n}_h = (\mathbf{I} - \mathbf{n}_h \mathbf{n}_h^t) \nabla v_h = \mathbf{P}_h \nabla v_h \in \mathbb{R}^3$$

其中 $\mathbf{P}_h = \mathbf{I} - \mathbf{n}_h \mathbf{n}_h^t \in \mathbb{R}^{3 \times 3}$. 显然 $\nabla_{S_h} v_h \in (L^2(S_h))^3$

如果限制投影 $\mathcal{P}_0 : U \rightarrow S$ 到 S_h 上, 就得到一个从 S_h 到 S 的连续可微的双射, 仍记为 \mathcal{P}_0 . 对于任意的 $\tau_h \in \mathcal{T}_h$, 我们可以得到一个曲面三角形 $\tau := \mathcal{P}_0(\tau_h)$, 并记所有的曲面三角形集合为 \mathcal{T}_S .

下面我们建立定义在 S 和 S_h 函数之间的关系. 借助双射投影 \mathcal{P}_0 , 可以由函数 $v : S \rightarrow \mathbb{R}$ 唯一地引入另一个函数 $\bar{v} : S_h \rightarrow \mathbb{R}$. 对于所有的 $x \in S_h$, 有 $\bar{v}(x) = v(\mathcal{P}_0(x))$. 对于任意的 $\tau_h \in \mathcal{T}_h$ 和函数 $v \in C^1(\mathcal{P}_0(\tau_h))$, 我们有

$$\nabla_{S_h} \bar{v}(x) = (\mathbf{P}_h(\mathbf{I} - d\mathbf{H})\mathbf{P})(x) \nabla_S v(\mathcal{P}_0(x)) \quad \forall x \in \tau_h.$$

反过来, 一个函数 $v_h : S_h \rightarrow \mathbb{R}$ 也可以唯一的引入一个函数 $\tilde{v}_h : S \rightarrow \mathbb{R}$, 对于所有的 $x \in S$, 有 $\tilde{v}_h(x) = v_h(\mathcal{P}_0^{-1}(x))$. 对于任意的 $\tau_h \in \mathcal{T}_h$ 和函数 $v_h \in C^1(\tau_h)$ 让 $\tau = \mathcal{P}_0(\tau_h)$, 可得

$$\nabla_S \tilde{v}_h(x) = (\mathbf{I} - d\mathbf{H})^{-1} \left(\mathbf{I} - \frac{\mathbf{n}_h \mathbf{n}_h^t}{\mathbf{n}^t \mathbf{n}_h} \right) \nabla_{S_h} v_h(\mathcal{P}_0^{-1}(x)) \quad \forall x \in \tau.$$

曲面线性有限元

对于三角形 $\tau_h \in \mathcal{T}_h$, 记 $\{\lambda_i\}$ 为 τ_h 的重心坐标. 记 \mathcal{V}_h 为 S_h 上的连续分片线性有限元空间, 也即对于任意的 $v_h \in \mathcal{V}_h$ 和 $\tau_h \in \mathcal{T}_h$, v_h 在 S_h 上连续, 且有 $v_h|_{\tau_h} \in \text{span}\{\lambda_1, \lambda_2, \lambda_3\}$. 我们可定义 S 上的提升空间

$$\tilde{\mathcal{V}}_h = \{\tilde{v}_h \mid \tilde{v}_h := v_h \circ \mathcal{P}_0^{-1}, \text{ 其中 } v_h \in \mathcal{V}_h\},$$

其中 $\mathcal{P}_0 : S_h \rightarrow S$ 为定义在 (1) 中的双射. 对于 $f \in L^2(S)$, 记

$$f_h(x) = \bar{f}(x) - \frac{1}{|S_h|} \int_{S_h} \bar{f} \mathbf{d}\sigma_h, \quad (2)$$

其中 $|S_h|$ 为 S_h 的总面积. 那么有 $\int_{S_h} f_h(x) \mathbf{d}\sigma_h = 0$, 且下列方程存在唯一的有限元解 $u_h \in \mathcal{V}_h$, 满足 $\int_{S_h} u_h \mathbf{d}\sigma_h = 0$

$$\int_{S_h} \nabla_{S_h} u_h \cdot \nabla_{S_h} v_h \mathbf{d}\sigma_h = \int_{S_h} f_h v_h \mathbf{d}\sigma_h \quad \forall v_h \in \mathcal{V}_h$$

曲面高次元

符号说明

S	\mathbb{R}^3 空间中的曲面
$K \subset \mathbb{R}^2$	二维空间中的标准单元
$\mathbf{u} = (u, v)^T$	二维空间中的坐标系
$\tau_h \subset \mathbb{R}^3$	三维空间中的尺寸为 h 的平面三角形
$\mathbf{x} = (x, y, z)^T \in \tau_h$	τ_h 上的一个点
\mathcal{P}_0	S 邻近区域到 S 的投影
$\mathbf{x}_i, i = 1, \dots, n_{dof}$	τ_h 上 p 次 Lagrangian 基函数对应的自由度坐标点
$\tau_p \subset \mathbb{R}^3$	定义在 τ_h 上的 p 次多项式曲面三角形
$\mathbf{x}_p = (x_p, y_p, z_p)^T \in \tau_p$	τ_p 上一个点的三维坐标
$\tau_S \subset \mathbb{R}^3$	把 τ_h 投影到曲面 S 上的曲面三角形
$\mathbf{x}_S = (x_S, y_S, z_S)^T \in \tau_S$	τ_S 上一个点的三维坐标
$\varphi_i(\mathbf{x})$	定义在 τ_h 上第 i 个 Lagrangian 基函数

τ_h τ_p 和 τ_s 之间关系

对于 τ_p 上的任意一点 \mathbf{x}_p , 存在一点 $\mathbf{x} \in \tau_h$, 使得

$$\mathbf{x}_p = \sum_{i=1}^{n_{dof}} \mathbf{x}_i \varphi_i(\mathbf{x})$$

进一步, 存在标准参考单元 K 中存在一点 $\mathbf{u} = (u, v)$, 可得

$$\mathbf{x}(u, v) = \lambda_0 \mathbf{x}_0 + \lambda_1 \mathbf{x}_1 + \lambda_2 \mathbf{x}_2$$

其中 $\mathbf{x}_0, \mathbf{x}_1$ 和 \mathbf{x}_2 为 τ_h 的三个顶点,

$$\lambda_0 = 1 - u - v, \quad \lambda_1 = u, \quad \lambda_2 = v$$

对于 τ_S 上的任意一点 \mathbf{x}_S , 存在 τ_p 上的一点 \mathbf{x}_p , 使得

$$\mathbf{x}_S = \mathcal{P}_0(\mathbf{x}_p)$$

\mathbf{x} 关于 (u, v) 的 Jacobi 矩阵为

$$\frac{\partial \mathbf{x}}{\partial \mathbf{u}} = [\mathbf{x}_1 - \mathbf{x}_0, \mathbf{x}_2 - \mathbf{x}_0]$$

\mathbf{x}_p 关于 \mathbf{x} 的 Jacobi 矩阵为

$$\frac{\partial \mathbf{x}_p}{\partial \mathbf{x}} = \sum_{i=1}^{n_{dof}} \begin{bmatrix} x_i \nabla_{\mathbf{x}} \varphi_i(\mathbf{x})^T \\ y_i \nabla_{\mathbf{x}} \varphi_i(\mathbf{x})^T \\ z_i \nabla_{\mathbf{x}} \varphi_i(\mathbf{x})^T \end{bmatrix}$$

则 \mathbf{x}_p 关于 \mathbf{u} 的 Jacobi 矩阵为

$$\frac{\partial \mathbf{x}_p}{\partial \mathbf{u}} = \left[\frac{\partial \mathbf{x}_p}{\partial u}, \frac{\partial \mathbf{x}_p}{\partial v} \right] = \sum_{i=1}^{n_{dof}} \begin{bmatrix} x_i \nabla_{\mathbf{x}} \varphi_i(\mathbf{x})^T \\ y_i \nabla_{\mathbf{x}} \varphi_i(\mathbf{x})^T \\ z_i \nabla_{\mathbf{x}} \varphi_i(\mathbf{x})^T \end{bmatrix} [\mathbf{x}_1 - \mathbf{x}_0, \mathbf{x}_2 - \mathbf{x}_0] \quad (2)$$

记

$$d\mathbf{x}_p = \frac{\partial \mathbf{x}_p}{\partial \mathbf{u}} d\mathbf{u} = \frac{\partial \mathbf{x}_p}{\partial u} du + \frac{\partial \mathbf{x}_p}{\partial v} dv,$$

其中 $d\mathbf{u} = [du, dv]^T$. 进一步可得曲面三角形 τ_p 上的第一基本形式

$$I = \langle d\mathbf{x}_p, d\mathbf{x}_p \rangle = d\mathbf{u}^T \begin{bmatrix} g_{11} & g_{12} \\ g_{12} & g_{22} \end{bmatrix} d\mathbf{u}$$

其中

$$g_{11} = \langle \frac{\partial \mathbf{x}_p}{\partial u}, \frac{\partial \mathbf{x}_p}{\partial u} \rangle, g_{12} = \langle \frac{\partial \mathbf{x}_p}{\partial u}, \frac{\partial \mathbf{x}_p}{\partial v} \rangle, g_{22} = \langle \frac{\partial \mathbf{x}_p}{\partial v}, \frac{\partial \mathbf{x}_p}{\partial v} \rangle,$$

定义 τ_p 上的基函数如下

$$\varphi_{p,i}(\mathbf{x}_p) = \varphi_i(\mathbf{x})$$

其中

$$\mathbf{x}_p = \sum_{i=1}^{n_{dof}} \mathbf{x}_i \varphi_i(\mathbf{x})$$

则 $\varphi_{p,i}(\mathbf{x}_p)$ 在 τ_p 上的切向导数定义如下:

$$\nabla_{S_p} \varphi_{p,i} = \frac{\partial \mathbf{x}_p}{\partial \mathbf{u}} \begin{bmatrix} g_{11} & g_{12} \\ g_{12} & g_{22} \end{bmatrix}^{-1} \left(\frac{\partial \mathbf{x}}{\partial \mathbf{u}} \right)^T \nabla_{S_h} \varphi_i(\mathbf{x})$$

S 上曲面三角形的面积计算公式

$$\mathcal{P}_0(\mathbf{x}) := \mathbf{x} - d(\mathbf{x}) \mathbf{n}(\mathbf{x})$$

对于 $\mathbf{x}_p \in \tau_p$, 存在 $\mathbf{x}_S \in S$, 有

$$\begin{aligned} \mathbf{x}_S &= \mathcal{P}_0(\mathbf{x}_p) = \mathbf{x}_p - d(\mathbf{x}_p) \mathbf{n}(\mathbf{x}_p) \\ \frac{\partial \mathbf{x}_S}{\partial \mathbf{x}_p} &= I - d(\mathbf{x}_p) H(\mathbf{x}_p) - \mathbf{n}(\mathbf{x}_p) \mathbf{n}(\mathbf{x}_p)^T \\ \frac{\partial \mathbf{x}_S}{\partial \mathbf{u}} &= \frac{\partial \mathbf{x}_S}{\partial \mathbf{x}_p} \frac{\partial \mathbf{x}_p}{\partial \mathbf{u}} \end{aligned}$$

S 上的导数计算

考虑 τ_S 和 τ_p 的关系

则 \mathbf{x}_S 关于 \mathbf{u} 的 Jacobi 矩阵为

$$\begin{aligned} \frac{\partial \mathbf{x}_S}{\partial \mathbf{u}} &= \frac{\partial \mathbf{x}_S}{\partial \mathbf{x}_p} \frac{\partial \mathbf{x}_p}{\partial \mathbf{u}} \\ \frac{\partial \mathbf{x}_S}{\partial \mathbf{x}_p} &= I - d(\mathbf{x}_p) H(\mathbf{x}_p) - \mathbf{n}(\mathbf{x}_p) \mathbf{n}(\mathbf{x}_p)^T \end{aligned}$$

由于

$$\frac{\partial \mathbf{x}_p}{\partial \mathbf{u}} = \left[\frac{\partial \mathbf{x}_p}{\partial u}, \frac{\partial \mathbf{x}_p}{\partial v} \right] = \sum_{i=1}^{n_{dof}} \begin{bmatrix} x_i \nabla_{\mathbf{x}} \varphi_i(\mathbf{x})^T \\ y_i \nabla_{\mathbf{x}} \varphi_i(\mathbf{x})^T \\ z_i \nabla_{\mathbf{x}} \varphi_i(\mathbf{x})^T \end{bmatrix} [\mathbf{x}_1 - \mathbf{x}_0, \mathbf{x}_2 - \mathbf{x}_0]$$

即, 很容易计算

$$\frac{\partial \mathbf{x}_S}{\partial \mathbf{u}}$$

进一步可得到曲面 τ_S 上的第一基本形式记

$$d\mathbf{x}_S = \frac{\partial \mathbf{x}_S}{\partial \mathbf{u}} d\mathbf{u} = \frac{\partial \mathbf{x}_S}{\partial u} du + \frac{\partial \mathbf{x}_S}{\partial v} dv,$$

其中

$$\begin{aligned} d\mathbf{x}_S &= \frac{\partial \mathbf{x}_S}{\partial \mathbf{u}} d\mathbf{u} \\ &= \frac{\partial \mathbf{x}_S}{\partial \mathbf{x}_p} \frac{\partial \mathbf{x}_p}{\partial \mathbf{u}} d\mathbf{u} \\ &= \frac{\partial \mathbf{x}_S}{\partial \mathbf{x}_p} \frac{\partial \mathbf{x}_p}{\partial u} du + \frac{\partial \mathbf{x}_S}{\partial \mathbf{x}_p} \frac{\partial \mathbf{x}_p}{\partial v} dv \end{aligned}$$

故

$$\begin{aligned} \frac{\partial \mathbf{x}_S}{\partial u} &= \frac{\partial \mathbf{x}_S}{\partial \mathbf{x}_p} \frac{\partial \mathbf{x}_p}{\partial u} \\ \frac{\partial \mathbf{x}_S}{\partial v} &= \frac{\partial \mathbf{x}_S}{\partial \mathbf{x}_p} \frac{\partial \mathbf{x}_p}{\partial v} \end{aligned}$$

$$I = \langle d\mathbf{x}_S, d\mathbf{x}_S \rangle = d\mathbf{u}^T \begin{bmatrix} g'_{11} & g'_{12} \\ g'_{12} & g'_{22} \end{bmatrix} d\mathbf{u}$$

其中

$$g'_{11} = \langle \frac{\partial \mathbf{x}_S}{\partial u}, \frac{\partial \mathbf{x}_S}{\partial u} \rangle, g'_{12} = \langle \frac{\partial \mathbf{x}_S}{\partial u}, \frac{\partial \mathbf{x}_S}{\partial v} \rangle, g'_{22} = \langle \frac{\partial \mathbf{x}_S}{\partial v}, \frac{\partial \mathbf{x}_S}{\partial v} \rangle,$$

定义 τ_S 上的基函数如下

$$\varphi_{S,i}(\mathbf{x}_S) = \varphi_i(\mathbf{x})$$

其中

$$\mathbf{x}_S = \sum_{i=1}^{n_{dof}} \mathbf{x}_i \varphi_i(\mathbf{x})$$

则 $\varphi_{S,i}(\mathbf{x}_S)$ 在 τ_S 上的导数定义如下:

$$\nabla_{S_S} \varphi_{S,i} = \frac{\partial \mathbf{x}_S}{\partial \mathbf{u}} \begin{bmatrix} g'_{11} & g'_{12} \\ g'_{12} & g'_{22} \end{bmatrix}^{-1} \left(\frac{\partial \mathbf{x}}{\partial \mathbf{u}} \right)^T \nabla_{S_h} \varphi_i(\mathbf{x})$$

设 $w(\mathbf{x}_S)$ 是定义在 S 上的函数, 利用投影可以定义 S_p 上函数

$$\hat{w}(\mathbf{x}_p) = w(\mathcal{P}_0(x_p))$$

下面讨论如何计算 $\nabla_{S_p} w$.

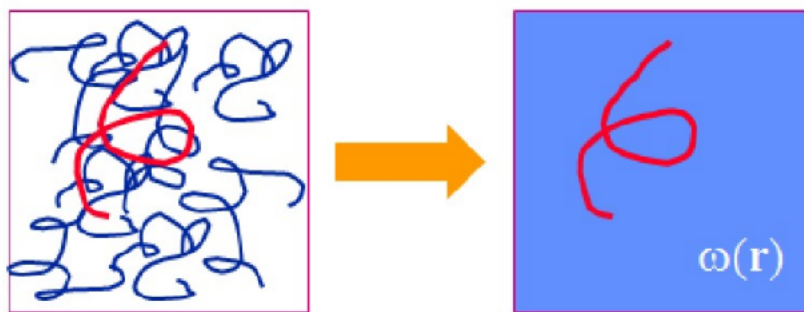
$$\nabla_{S_p} \hat{w}(\mathbf{x}_p) = \frac{\partial \mathbf{x}_p}{\partial \mathbf{u}} \begin{bmatrix} g_{11} & g_{12} \\ g_{12} & g_{22} \end{bmatrix}^{-1} \begin{pmatrix} \hat{w}_u \\ \hat{w}_v \end{pmatrix}$$

$$\begin{pmatrix} \hat{w}_u \\ \hat{w}_v \end{pmatrix} = \left(\frac{\partial \mathbf{x}_S}{\partial \mathbf{u}} \right)^T \nabla_{\mathbf{x}_S} w(x_S)$$

自洽场模型及其数值算法

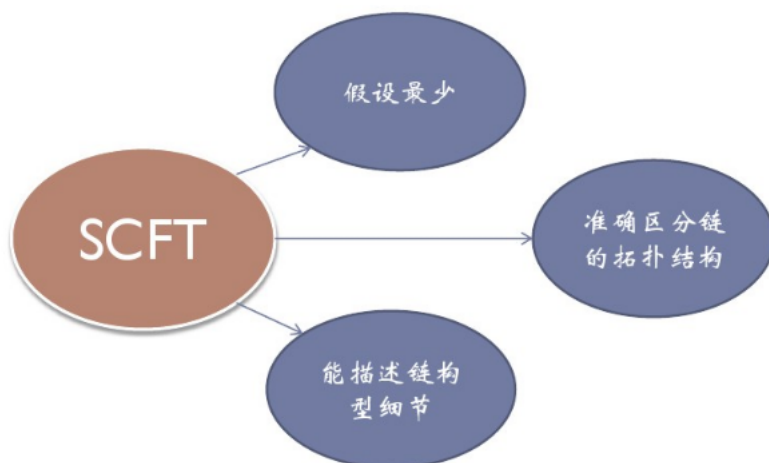
1. 平均场理论

平均场理论是一种处理多体问题的重要方法,它的基本出发点是用一个“平均了的场”来近似代替多体系统中某个特定个体受到的作用,从而把复杂的多体问题近似地转化为单体问题。



多链相互作用体系等效为单链在势场中

自洽场模型是基于平均场近似的粗粒化模型,特别适合研究发生相分离的非均相高分子体系在平衡态的相结构及相图。



2. 共聚物及分类

共聚物: 由两种或更多单体发生聚合反应所形成的聚合物称为共聚物, 与之对应的是只有一种单体聚合而成的均聚物。共聚物的结构中具有至少两种结构单元, 结构单元之间以化学键连接。由于共聚物包含至少两种结构单元, 它可以根据其结构单元的排列顺序分成四种共聚物, 图 2 给出了当结构单元为 A 和 B 时的情况: 依次为均聚物、交替共聚物、无规共聚物、嵌段共聚物和接枝共聚物。



图 2

- 交替共聚物: 共聚物中两结构单元 A 和 B 严格交替相间, 两者在共聚物中的摩尔分数均约为 50%
- 无规共聚物: 共聚物中两结构单元 A 和 B 随机出现, 其中 A 和 B 自身连续的单元数不多, 一般在几个到十几个。从统计上看, 无规共聚物的某结构单元在聚合链上一段的含量等于其在整个聚合物中的含量。

- 嵌段共聚物: 由较长的只有结构单元 A 的链段和较长的只有结构单元 B 的链段构成, 其中每一链段可达到几百到几千结构单元。
- 接枝共聚物: 接枝共聚物在结构上属于支化聚合物, 其不仅有主链, 还有较长的支链, 且主链和支链是由不同种结构单元组成, 如上图, 主链全部是结构单元 A, 而支链全部是结构单元 B。有时候, 接枝聚合物的主链和支链可能都是共聚物, 比如主链是 A 和 B 的无规共聚物, 支链是 A 和 B 的交替共聚物, 整体仍然是接枝共聚物。

3. 相分离

嵌段聚合物在一定温度下会发生相分离现象, 但一般的相分离体系如水和油, 油滴的尺寸在微米尺度, 而嵌段聚合物的嵌段间有化学键的连接, 形成的相结构只有几十到几百纳米, 所以被称为微相分离。嵌段共聚物的相转变温度主要取决于几个参数, 总聚合物 N , 拓扑结构参数 n , 某嵌段所占的体积分数 f 和嵌段之间的 Flory-Huggins 相互作用参数 χ 。随着嵌段所占的体积分数不同, 嵌段聚合物相分离后形成的相结构也不同, 有球状相, 六方柱状相, 层状相和双连续回转状相等结构。

4. 两嵌段柔性链共聚物体系

首先给出两嵌段共聚物体系的连续高分子链的模型

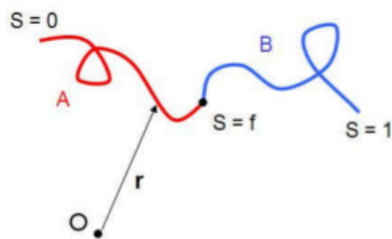


图 3

我们考虑一般曲面上的两嵌段共聚物体系, 假设系统有 n 个共聚物分子, 每

个共聚物分子是由 A 和 B 两类单体构成的两嵌段共聚物, 它们限制在总面积为 S 的曲面上. 单体 A 、 B 的体积分数分别为 f 和 $1 - f$, 在平均场理论下, 该系统的哈密顿量是

$$H[w_+, w_-] = \frac{1}{|S|} \int \mathbf{d}x \left\{ -w_+(x) + \frac{w_-^2(x)}{\chi N} \right\} - \log Q[w_+(x), w_-(x)] \quad (1)$$

方程 (1) 中 $w_-(x)$ 和 $w_+(x)$ 是对应系统的场函数, 其中 x 表示空间曲面上的点, Q 是单链配分函数, S 是曲面的面积, χN 是体系参数, 其中 χ 是相互作用参数, N 是聚合度. 自由能函数关于场的一阶变分推导出下面的自洽场方程

$$\begin{aligned} \frac{\delta H}{\delta w_+(x)} &= \phi_A(x) + \phi_B(x) - 1 = 0 \\ \frac{\delta H}{\delta w_-(x)} &= \frac{2w_-(x)}{\chi N} - [\phi_A(x) - \phi_B(x)] = 0 \end{aligned}$$

其中 $\phi_A(x)$ 和 $\phi_B(x)$ 是单体 A 和单体 B 的密度函数. 然后结合连续高分子链的模型, 得到自洽场系统.

$$\frac{\partial}{\partial s} q(x, s) = \nabla_S^2 q(x, s) - w(x, s) q(x, s), \quad q(x, 0) = 1 \quad (2)$$

$$\frac{\partial}{\partial s} q^\dagger(x, s) = -\nabla_S^2 q^\dagger(x, s) + w(x, s) q^\dagger(x, s), \quad q^\dagger(x, 1) = 1 \quad (3)$$

$$w(x, s) = \begin{cases} w_+(x) - w_-(x) & 0 \leq s \leq f \\ w_+(x) + w_-(x) & f \leq s \leq 1 \end{cases}$$

$$Q = \frac{1}{|S|} \int \mathbf{d}x q(x, s) q^\dagger(x, 1 - s), \quad \forall s \in [0, 1] \quad (4)$$

$$\phi_A(x) = \frac{1}{Q} \int_0^f \mathbf{d}s q(x, s) q^\dagger(x, 1 - s) \quad (5)$$

$$\phi_B(x) = \frac{1}{Q} \int_f^1 \mathbf{d}s q(x, s) q^\dagger(x, 1 - s) \quad (6)$$

在系统中, 正向传播子 $q(x, s)(s: 0 \rightarrow 1)$ 表示长度为 s 的链在空间曲面位置 x 处的概率密度分布函数, 其中变量 s 用于参数化每条共聚物链, $s = 0$ 表示单体 A 的起始位置, $s = f$ 是单体 A 和单体 B 之间的连接点, 反向传播子 $q^\dagger(x, s)(s: 1 \rightarrow 0)$ 表示长度为 s 的链在空间曲面位置 x 处的概率密度分布函数.

5.PDE Solver

这里只给出正向传播子方程 (2) 的离散化, 反向传播子方程 (3) 可以类似的得出. 首先, 给出 (2) 的变分形式对 $q \in H^1(S)$, 满足

$$\left(\frac{\partial}{\partial s} q, v \right)_S = -(\nabla_S q, \nabla v)_S - (wq, v)_S, \quad \text{for all } v \in H^1(S).$$

其中 $(\cdot, \cdot)_S$ 表示在曲面 S 上 L^2 内积.

找到 $V_h \subset H^1(S)$ 得到 (2) 线性曲面有限元的离散形式, 对 $q_h = \sum_{i=1}^N q_i(s) \varphi_i(x)$, 满足

$$\left(\frac{\partial}{\partial s} q_h, v_h \right)_{S_h} = -(\nabla_{S_h} q_h, \nabla_{S_h} v_h)_{S_h} - (w_h q_h, v_h)_{S_h}, \quad \text{for all } v_h \in H^1(S) \quad (4).$$

其中 $(\cdot, \cdot)_{S_h}$ 表示在曲面 S_h 上 L^2 内积. $w_h(x, s) = \sum_{i=1}^N w(x_i, s) \varphi_i(x)$ 是 $w(x, s)$ 的线性插值. 得到方程 (4) 的半离散矩阵形式

$$M \frac{\partial}{\partial s} \mathbf{q}(s) = -(A + F) \mathbf{q}(s) \quad (5)$$

其中

$$\mathbf{q}(s) = (q_1(s), q_2(s), \dots, q_N(s))^T$$

且

$$M_{i,j} = (\varphi_i, \varphi_j), \quad A_{i,j} = (\nabla_S \varphi_i, \nabla_S \varphi_j), \quad F_{i,j} = (w_h \varphi_i, \varphi_j)$$

接下来, 在 s - 方向使用 Crank-Nicolson 格式

$$M \frac{\mathbf{q}^{n+1} - \mathbf{q}^n}{ds} = -\frac{1}{2}(A + F) [\mathbf{q}^{n+1} + \mathbf{q}^n]$$

ds 是时间步长, 整理得全离散的矩阵形式为

$$\left[M + \frac{ds}{2}(A + F) \right] \mathbf{q}^{n+1} = \left[-\frac{ds}{2}(A + F) + M \right] \mathbf{q}^n$$

s - 方向上的积分格式

$$\int_0^{n_s} dt f(s) = \Delta s \left\{ -\frac{5}{8}(f_0 + f_{n_s}) + \frac{1}{6}(f_1 + f_{n_s-1}) - \frac{1}{24}(f_2 + f_{n_s-2}) + \sum_{j=0}^{n_s} f_j \right\}$$

6. 自洽场迭代

这里我们采用梯度下降法达到鞍点, 具体而言, 我们引入虚拟时间 t 并且在每个时间步, 我们给出相应的鞍点搜索表达式

$$\frac{\partial}{\partial t} w_+(x, t) = \frac{\delta H[w_+, w_-]}{\delta w_+(x, t)} \quad (7)$$

$$\frac{\partial}{\partial t} w_-(x, t) = \frac{\delta H[w_+, w_-]}{\delta w_-(x, t)} \quad (8)$$

显然, 当 (2) 和 (3) 达到平衡态时满足 (7) 和 (8)

寻找 SCFT 的鞍点需要迭代方法, 在 SCFT 的标准框架下, 我们通过如下的迭代方案来更新 $w_+(x), w_-(x)$.

step 1 给定初始估计的场 $w_+(x), w_-(x)$.

step 2 计算在一般曲面上的正向传播子 $q(x, s)$ 和反向传播子 $q^\dagger(x, s)$.

step 3 由计算的 $q(x, s)$ 和 $q^\dagger(x, s)$ 通过自洽场系统得到 $Q, \phi_A(x)$ 和 $\phi_B(x)$, 并计算出哈密顿量 H .

step 4 对 (2) 和 (3) 使用迭代方法更新场 $w_+(x, t)$ 和 $w_-(x, t)$.

step 5 重复步骤 2-4, 直到达到收敛标准.

自适应网格加密

1. 网格数据结构

以一个三角形网格为例,如下图

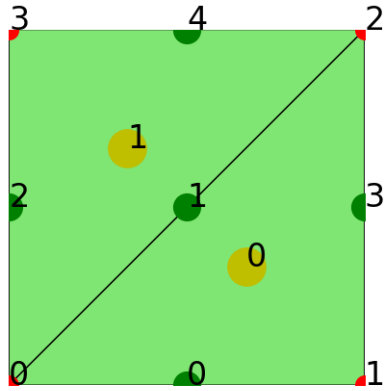


图 4

1. 边数组 'edge'

(a) 二维 $NE \times 2$ 数组

(b) 'edge[i, 0]' 和 'edge[i, 1]' 分别存储第 i 条边的起点和终点的全局编号

(c) 如果第 i 条边是边界边, 则规定从 'edge[i, 0]' 看向 'edge[i, 1]', 网格区域一定在左手边

```
1 array([[0, 1],
2        [2, 0],
3        [3, 0],
4        [1, 2],
5        [2, 3])
```

2. 边与单元的相邻关系数组 'edge2cell'

(a) 二维 $NE \times 4$ 的数组

(b) 'edge2cell[i, 0]' 和 'edge2cell[i, 1]' 分别存储第 i 条边左右两个单元的全局编号

(c) 'edge2cell[i, 2]' 和 'edge2cell[i, 3]' 分别存储第 i 条边在左右两个单元中的局部编号

(d) 如果是边界边, 则

i. 'edge2cell[i, 0] = edge2cell[i, 1]'

ii. 'edge2cell[i, 2] = edge2cell[i, 3]'

```
1 array([[0, 0, 1, 1],
2        [0, 1, 0, 0],
3        [1, 1, 2, 2],
4        [0, 0, 2, 2],
5        [1, 1, 1, 1],])
```

3. 单元与单元的相邻关系数组 'cell2cell'

(a) 二维 $NE \times 3$ 的数组

(b) 'cell2cell[i, 0]' 和 'cell2cell[i, 1]' 和 'cell2cell[i, 2]' 分别存储第 i 个单元的邻居单元的全局编号, 如果没有邻居单元, 邻居单元的全局编号就是这个单元的全局编号.

```
1 array([[1, 0, 0],
2        [0, 1, 1]])
```

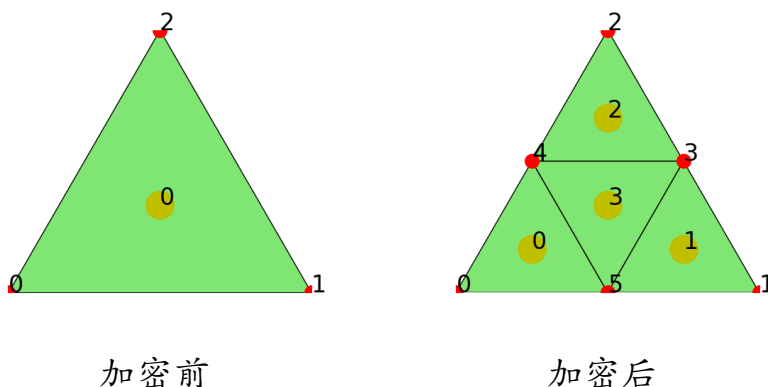
4. 单元与边的关系 'cell2edge'

(a) 二维 $NC \times 3$ 的数组

(b) 'cell2edge[i, 0:3]' 是存储第 'i' 个单元的逆时针对应边的全局编号

```
1 array([[1, 0, 3],
2        [1, 4, 2]])
```

2. 约定



我们的加密方式是取三角形三条边的中点,然后把三条边的中点连起来,这样就把原来的三角形分成了四个小三角形,我们约定四个小三角形的局部编号顺序为 0, 1, 2, 3. 然后约定局部编号为 3 的三角形的顶点的局部编号的顺序和他父单元边的局部编号顺序一致.

3. Tritree 的数据结构

1. 生成 Tritree 类的时候, 是从 'TriangleMesh' 中继承网格
2. 定义初始化时, 需要继承的是网格点 'node', 单元 'cell', 根据 'node' 和 'cell' 初始化网格, 然后得到单元的个数 'NC', 再根据单元的个数生成两个数组 'parent' 和 'child', 这两个数组分别是值为 -1 的 $NC \times 2$ 和 $NC \times 4$ 的数组
3. 定义叶子单元指标 (即 'leaf_cell_index'): 根据数组 'child' 的值为 -1 的所有行维度的索引值 'idx', 返回叶子单元的 'idx'.
4. 定义叶子单元 (即 'is_leaf_cell'): 是根据叶子单元指标 'idx' 来判断是否是叶子单元: 若 'idx' 是 'None', 返回的是 'child[:, 0] == -1', 是布尔型数组, 表示的意思是第 0 列的所有行的值等于 -1 为真; 若 'idx' 不是 'None' 则返回的是 'child[idx, 0] == -1', 是布尔型数组, 表示的意思是第 0 列的所有对应 'idx' 的值等于 -1 为真
5. 定义根单元 (即 'is_root_cell'): 是根据单元指标 'idx' 来判断是否是根单

元: 若 'idx' 是 'None', 返回的是 'parent[:, 0] == -1', 是布尔型数组, 表示的意思是第 0 列的所有行的值等于 -1 为真; 若 'idx' 不是 'None' 则返回的是 'parent[idx, 0] == -1', 是布尔型数组, 表示的意思是第 0 列的所有对应 'idx' 的值等于 -1 为真

4. 加密算法

1. 定义加密 (即 'refine'): 传入的数据是标记 'marker' (标记是自适应加密过程中必不可少的一步, 它构成自适应加密循环的过程, 自适应加密过程为: 求解 \rightarrow 估计 \rightarrow 标记 \rightarrow 加密或放粗). 对于传入的数据 'marker' 若 'marker == None' 为真, 指标的索引 'idx' 就是叶子单元的索引, 否则, 叶子单元指标 'idx' 就是标记之后的索引. 若 'idx' 是 'None' 则返回 'False', 加密结束, 如果 'idx' 的长度大于 0, 则进行如下, 我们定义 'isMarkedCell' 是长度为 'NC' 的布尔型数组, 然后把 'isMarkedCell' 索引中 'idx' 是叶子单元的等于 'True'.

2. 扩展标记单元

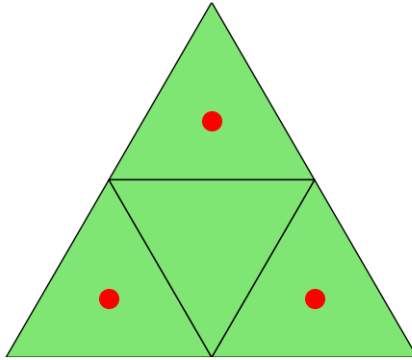


图 5

case 1: 如图 5, 如果这个单元是叶子单元但不是被标记的单元, 但是这个单元的邻居单元至少有两个以上的单元是被标记的单元或者不是叶子单元, 那么这个单元也应该被标记.

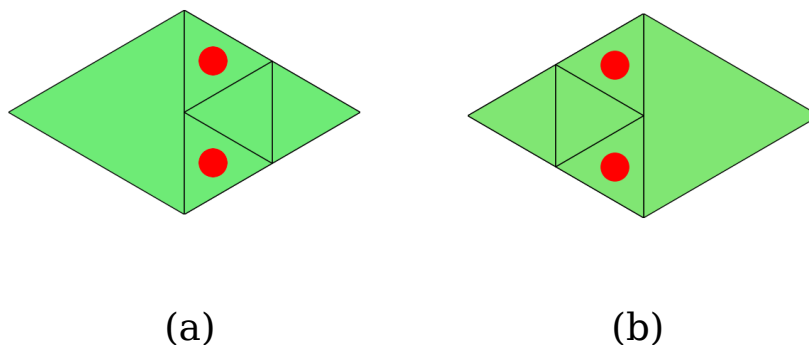


图 6

case 2: 如图 6(a) 它的左边单元是叶子单元但不是被标记的单元, 右边单元不是叶子单元, 但是右边单元中与左边单元相邻的孩子单元是被标记的单元, 那么左边单元也应该被标记.

如图 6(b) 它的右边单元是叶子单元但不是被标记的单元, 左边单元不是叶子单元, 但是左边单元中与右边单元相邻的孩子单元是被标记的单元, 那么右边单元也应该被标记.

扩展单元的时候我们找到是上面两种情况的叶子单元, 把 'isMarked-Cell' 索引中 'idx' 是上面两种情况的叶子单元的等于 'True', 现在我们就找到了所有要被加密的单元.

3. 找到加密的边和 'edge2newNode'

我们定义 'refineFalg' 是长度为 'NE' 的布尔型数组, 通过 'cell2edge' 找到 'isMarkedCell' 的边, 然后把 'refineFalg' 中索引是这些边的等于 'True', 但是这里要除去左边是叶子单元, 右边不是叶子单元或者左边不是叶子单元边, 右边是叶子单元的那些边, 在 'refineFalg' 索引是这些边的等于 'False', 要加密边的总数为 'NNN=refineFalg.sum()'.

'edge2newNode' 是长度为 'NE' 的数组, 加密之后边上的新的点的编号为 'NN + np.arange(NNN)'.

4. 加密成四个小三角形

首先 'NCC' 是标记单元的个数, 再根据标记单元的个数生成 'cell4' 是值为 0 的 '4*NC'×3 数组和 'parent4', 'child4' 分别是值为 -1 的 '4*NC'×2 和 '4*NC'×4 的数组, 然后找到一分为四的每个小三角形

的顶点, 然后把连起来, 最后把所有的 ‘node’, ‘cell’, ‘parent’, ‘child’ 拼到一起.

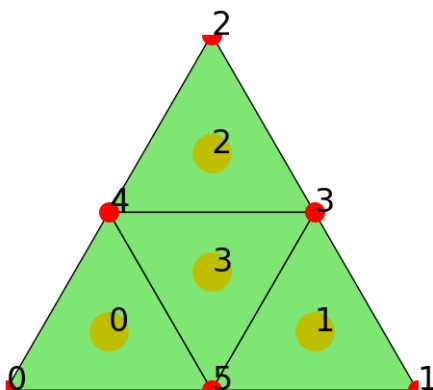


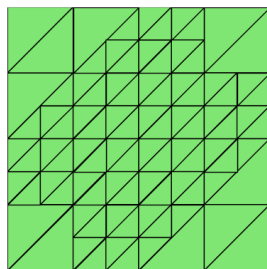
图 7

```

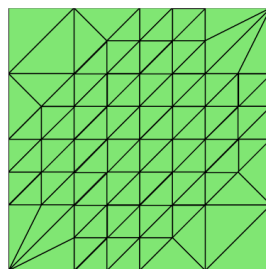
1 cell4[:NCC, 0] = cell[isMarkedCell, 0]
2 cell4[:NCC, 1] = edge2newNode[cell2edge[isMarkedCell, 2]]
3 cell4[:NCC, 2] = edge2newNode[cell2edge[isMarkedCell, 1]]
4 parent4[:NCC, 0] = idx
5 parent4[:NCC, 1] = 0
6 self.child[idx, 0] = NC + np.arange(0, NCC)

```

5. 生成协调化网格



协调化之前



协调化之后

最后我们进行协调化处理, 找到每条加密边的中点和对应单元的顶点把它们连在一起, 这里要注意最后的 ‘cell’ 的处理.

然后调用 ‘TriangleMesh(node, cell)’, 生成协调化网格 ‘Tmesh’.

例子 1: 加密圆穿过的单元

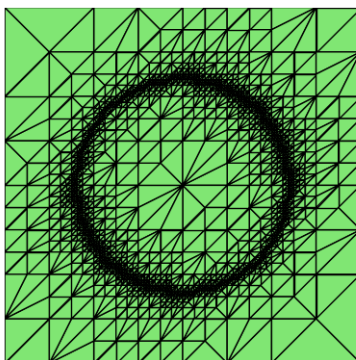


图 8

例子 2: 球面上的加密

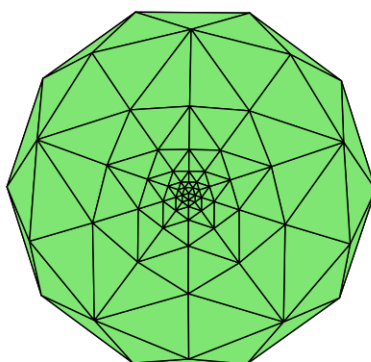


图 9

自适应有限元方法

1. 预备知识

由有限元方法的先验误差估计知,虽然求得了数值解,但不能知道误差的具体大小,特别是当解具有奇性的时候,要得到好的数值解要求网格尺寸 h 足够小,这是不现实的。自适应有限元方法 (AFEM) 是求解科学计算和工程问题的最有效方法之一,特别对存在奇性或者多尺度性质的问题有很强的适应性。它根据计算结果和已知量自动控制计算的过程,逐步调整网格,使得误差分布得比较均匀,从而以尽量少的计算量达到所要求的精度,提高计算效率。自适应方法的研究对象是连续的微分或者积分方程,其基本算法包含两步:当前离散空间上的求解和调整后的离散空间上的校正。自适应算法奏效的基本条件是:后验误差估计子的可靠性和有效性,调整后的离散空间能够捕捉到足够的局部高频信息。后验误差估计子是基于近似解和其他已知条件的一个可计算量,它既是真实误差的一个上界又是真实误差的一个下界。后验误差估计子是调整离散空间的基础。调整离散空间是自适应方法至关重要的部分,目前科学与工程计算中调整离散空间的方法有 h -方法、 p -方法、 hp -方法和 r -方法。 h -方法是一种局部网格加密方法,它保持单元多项式次数不变,通过逐步加密网格来调整离散空间,和 h -方法不同, p -方法保持网格不变,通过提高单元多项式次数来调整离散空间。 hp -方法是 h -和 p -方法的结合,即在加密网格的同时改变单元多项式的次数。 r -方法是在不改变节点总数和多项式次数的前提下通过移动节点来调整离散空间。

自适应有限元算法流程图见图 10,由下面的几步操作构成:

求解 \rightarrow 误差估计 \rightarrow 标记 \rightarrow 加密

- 求解:在当前网格上求得数值解 u_h ; 包括计算刚度矩阵和载荷向量,求解代数方程组.
- 误差估计:根据后验误差估计,由已知数据和数值解 u_h 估计单元误差和全局误差.
- 标记:根据标记策略和计算的单元误差,对网格中需要加密的单元进行标记.

- 加密: 依据加密算法, 对标记的单元进行加密, 得到新网格.

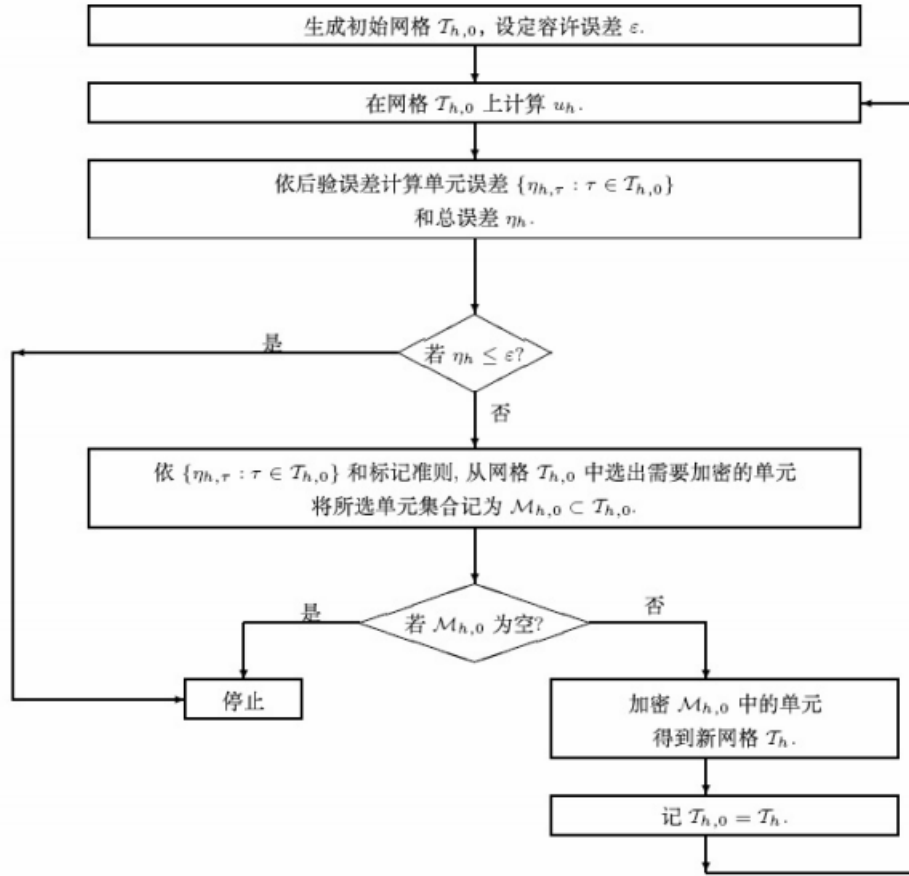


图 10

2. 后验误差估计

后验误差估计是自适应有限元方法中重要的一部分, 它是根据数值解和已知数据构造单元估计子 $\{\eta_\tau : \tau \in T_h\}$ 和全局估计子 $\eta_h^2 = \sum_{\tau \in T_h} \eta_\tau^2$, 来局部和全局的估计误差 $e_h = u - u_h$. 一个好的后验误差估计既是误差的下界也是误差的上界, 即

$$C_1 \eta_h \leq \|e_h\| \leq C_2 \eta_h$$

上界保证了计算结果的准确性, 下界则保证了计算结果的有效性。特别地, $\frac{\eta_h}{\|e_h\|} \rightarrow 1$, 则称后验误差估计是渐进准确的。 η_h 用来控制自适应算法的终止, 单元误差估计子则表现了误差的分布情况。

后验误差估计主要可归为两类：残量型后验误差估计和重构型后验误差估计。残量型后验误差估计是通过计算局部残量而得到误差估计。对残量的估计方式可分为显式法和隐式法，导出的后验误差估计分别称为显式后验误差估计和隐式后验误差估计。重构型误差估计具有与问题（方程）无关、与有限元形式无关等优点，且实现简单、计算量小、具有鲁棒性（是指控制系统在一定（结构，大小）的参数摄动下，维持其它某些性能的特性）。重构型后验误差估计是基于后处理技术的误差估计，后处理是在得到数值解后，通过少量的计算，来改善有限元解的质量。常用的后处理的方法有：加权平均、插值后处理、最小二乘法、局部或全局投影等。用后处理的解代替真解，它可用来估计真实误差。

3. 基于 WAR 的后验误差估计

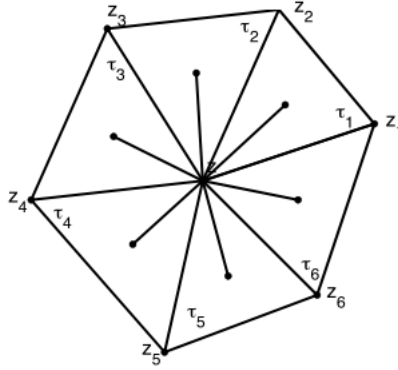


图 11

首先我们给出三角形单元的五种加权平均格式的定义。记 \mathcal{T}_h 为区域 $\Omega \subset \mathbb{R}^2$ 的三角剖分, S_h 为 \mathcal{T}_h 上对应的线性元空间

$$S_h = \{v \in H^1(\Omega) : v \in P_1(\tau), \forall \tau \in \mathcal{T}_h\}$$

其中 $P_k(D)$ 表示 $D \in \mathbb{R}^2$ 上次数不超过 k 次的多项式集合。对任一内节点 z , 记 \mathcal{K}_z 为点 z 的单元片。如图 11, 设 $\tau_1, \tau_2, \dots, \tau_m \in \mathcal{T}_h$ 为单元片 \mathcal{K}_z 中的单元, 记 c_i 为单元 τ_i 的重心, S_i 为单元 τ_i 的面积, $d_i = \|z - c_i\|$ 表示点 z 与 c_i 的距离, α_i 为三角形 τ_i 中以 z 顶点的角的角度。对有限元函数 $v \in S_h$, 定义加权平均格式如下:

简单平均:

$$S_h v(z) := \sum_{i=1}^m \frac{1}{m} \nabla v|_{\tau_i}$$

面积平均

$$G_h v(z) := \sum_{i=1}^m \frac{S_i}{\sum_{j=1}^m S_j} \nabla v|_{\tau_i}$$

面积调和平均

$$H_h v(z) := \sum_{i=1}^m \frac{\frac{1}{S_i}}{\sum_{j=1}^m \frac{1}{S_j}} \nabla v|_{\tau_i}$$

角度平均

$$A_h v(z) := \sum_{i=1}^m \frac{\alpha_i}{\sum_{j=1}^m \alpha_j} \nabla v|_{\tau_i}$$

距离调和平均

$$D_h v(z) := \sum_{i=1}^m \frac{\frac{1}{d_i}}{\sum_{j=1}^m \frac{1}{d_j}} \nabla v|_{\tau_i}$$

上面我们讨论了梯度重构的加权平均方法, 加权平均方法是一种简单的后处理方法, 基于加权平均方法可以获得有效的后验误差估计, 因此在自适应方法中被广泛的应用。基于加权平均方法的后验误差估计具有如下形式:

$$\eta_a = \|R_h u_h - \nabla u_h\|_{0,\Omega}, \quad R_h u_h(z) = \sum_{j=1}^m \nabla u_h(z_j) \omega_j$$

其中, R_h 表示加权平均算子, z_j 为样本点, z 为重构点, ω_j 为权函数。我们考虑基于前面讨论的加权平均方法的后验误差估计。记 S_h, G_h, H_h, A_h, D_h 分别表示简单平均算子、面积平均算子、面积调和平均算子、角度平均算子和距离调和平均算子, 我们记

$$\eta_{S,\tau}^2 = \|S_h u_h - \nabla u_h\|_{0,\tau}^2, \quad \eta_S^2 = \sum_{\tau \in \mathcal{T}_h} \eta_{S,\tau}^2$$

$$\begin{aligned}
\eta_{G,\tau}^2 &= \|G_h u_h - \nabla u_h\|_{0,\tau}^2, & \eta_G^2 &= \sum_{\tau \in \mathcal{T}_h} \eta_{G,\tau}^2 \\
\eta_{H,\tau}^2 &= \|H_h u_h - \nabla u_h\|_{0,\tau}^2, & \eta_H^2 &= \sum_{\tau \in \mathcal{T}_h} \eta_{H,\tau}^2 \\
\eta_{A,\tau}^2 &= \|A_h u_h - \nabla u_h\|_{0,\tau}^2, & \eta_A^2 &= \sum_{\tau \in \mathcal{T}_h} \eta_{A,\tau}^2 \\
\eta_{D,\tau}^2 &= \|D_h u_h - \nabla u_h\|_{0,\tau}^2, & \eta_D^2 &= \sum_{\tau \in \mathcal{T}_h} \eta_{D,\tau}^2
\end{aligned}$$

参考文献