

无网格数值求解方法

陶文铨, 吴学红, 戴艳俊

(动力工程多相流国家重点实验室(西安交通大学), 陕西省 西安市 710049)

Numerical Solutions of Meshless Methods

TAO Wen-quan, WU Xue-hong, DAI Yan-jun

(State Key Laboratory of Multiphase Flow in Power Engineering (Xi'an Jiaotong University), Xi'an 710049, Shaanxi Province, China)

ABSTRACT: Meshless method is a recently-developed new numerical approach. Many different meshless methods have been constructed based on different approximation function methods and different discretized schemes of partial differential equations (PDEs). In this paper, its basic principle was introduced and the construction methods for the approximation function and the discretization of the partial differential equations were presented. The characteristics of weight function and the shape function were analyzed in detail taking the moving least-squares (MLS) method as an example. Results show that radial basis function (RBF) and point interpolation methods possess Kronecker Delta function property (δ function), but the robustness is poor in some cases; the MLS approximation function does not possess Kronecker Delta function property, but it has good robustness. Differences among the three discretization schemes of meshless method are as follows: the collocation method requires no numerical integration and very little computational time while its robustness is poor; Galerkin method is not a truly meshless method due to the background meshes required for integration; the Petrov-Galerkin method is a truly meshless method and need numerical integration in each sub-domain, so it needs more computational time.

KEY WORDS: meshless method; approximation function; weight function; moving least-squares (MLS); shape function; discretization scheme

摘要: 无网格方法是近年发展起来的一种新的数值计算方法, 根据近似函数构建方法和微分方程离散方法的不同, 可以构建出许多不同的无网格方法。该文简述了无网格方法的理论基础; 介绍近似函数的构建方法和微分方程的离散方法, 并以移动最小二乘近似方法为例, 分析了权函数和形函数的特征。分析结果显示: 径向基函数和点插值法

均具有 δ 函数属性, 但计算稳定性差; 移动最小二乘近似函数不具有 δ 函数属性, 但计算比较稳定; 无网格方法中的3种离散方法不同之处在于: 配点法不需要积分, 计算量小, 计算稳定性差; Galerkin方法需要借助背景网格进行积分, 它不是真正的无网格方法; Petrov-Galerkin方法, 是一种真正的无网格方法, 它需要对每个子域进行积分, 计算工作量较大。

关键词: 无网格方法; 近似函数; 权函数; 移动最小二乘; 形函数; 离散方法

0 引言

过去几十年中, 在计算力学与计算传热学领域中常用的数值计算方法主要是有限差分法、有限元法、有限体积法、边界元法等。上述几种数值计算方法在工程计算中得到了广泛地应用, 并成功发展了一批商业软件, 如 FLUENT、STAR-CD、ANSYS、CFX、FIDAP 等, 但是研究者一直没有停止对新数值方法的研究, 因为现有的数值计算方法在某些工程问题中存在网格的束缚, 使得计算遇到很大的困难, 这些问题包括: 1) 工业材料冲压成型过程中的局部大变形; 2) 动态裂纹扩展; 3) 高速撞击; 4) 材料的相变; 5) 冲击破坏; 6) 流固耦合; 7) 材料破坏与失效; 8) 大爆炸过程, 等。

在计算上述问题时, 基于网格的方法在计算过程中出现网格畸变, 严重地影响解的精度, 降低计算效率甚至导致计算失效, 这时需要重新划分网格。网格划分对于简单的问题不存在困难, 但是对于比较复杂的区域尤其是三维问题, 网格的重新划分是十分困难的工作。鉴于基于网格的数值方法在处理上述问题时的缺陷, 近十几年来, 国内外许多学者都对无网格方法进行了大量的研究, 如 Belytschko T, Liu W K, Atluri S N, 刘桂荣(Liu G

基金项目: 国家自然科学基金项目(50636050)。

Project Supported by National Natural Science Foundation of China (50636050).

R), 张雄, 刘更等^[1-10]。与基于网格的方法不同, 无网格方法只需要节点的信息, 不需要节点与节点之间的相互联系的信息, 这样很容易在复杂计算区域内布置节点; 无网格方法数值求解的基本思想是: 在每个节点上构建待求物理量近似值的插值函数; 并用加权残量法和该近似函数对微分方程进行离散, 形成与待求物理量相关的各节点近似值的离散方程(一般称为系统刚度方程), 并求解之。然而无网格法也存在一些固有缺陷。例如, 一般计算量大, 效率低; 它在解除网格束缚的同时, 也引入了一些不确定参数, 目前没有理论方法来确定插值域的大小及背景积分域的大小。

近几年来国在无网格方法的研究方面进展迅速, 已经有专著或者专门的综述文章发表^[9-12], 但是都没有针对传热与流动问题系统地介绍无网格方法, 本文系统简述了无网格方法的理论基础; 介绍了近似函数的构建方法和微分方程的离散方法, 并以移动最小二乘近似方法为例, 分析了权函数和形函数的特征。关于无网格发展过程参见文献[13-44]。

1 近似函数的构建

1.1 近似函数的统一表示方法

目前比较常用的近似函数的构建方法有核函数近似方法、再生核近似方法、移动最小二乘近似方法、单位分解(partition of unity, PU)近似方法、径向基函数近似方法、点插值近似方法等。各种近似函数的构建方法虽然构造原理不同, 但都仅仅使用离散节点几何位置的信息, 节点间不需要指定明显的连接关系。其中核函数近似方法、再生核近似方法、移动最小二乘近似方法和径向基函数近似方法在构造近似方案时都使用了具有紧支集特性的节点权函数(核函数、窗口函数), 权函数不等于零的区域被称为权函数紧支集或节点支持域。

在无网格法中, 近似函数可以统一表示为

$$u^h(\mathbf{x}) = \sum_{I=1}^N \Phi_I(\mathbf{x}) u_I \quad (1)$$

式中: $u^h(\mathbf{x})$ 为无网格方法所要求取的物理问题的近似解, 又称为试探函数(trial function); $u_I (I=1, \dots, N)$ 为虚拟的节点值(fictitious node value), N 为支持域中所包含的节点数; $\Phi_I(\mathbf{x})$ 称为形函数。每种无网格方法都要构建如式(1)所示的近似函数, 其关键在于获得形函数, 获得形函数后, 利用离散方法建立起关于 u_I 的代数方程, 求解该代数方程后, 利用式(1)就得到不同节点的被求解物理量的近似值。可以认

为在无网格方法中正是由于引入了如式(1)所示的近似函数表达式, 以另一种形式构建了相邻节点间的联系, 才使得它可以不需要常规数值方法中节点间连接关系的信息。

各种近似函数的构建方法的区别在于构建形函数时使用了不同的数值方法。

1.2 核近似

光滑粒子动力学方法(smoothed particle hydrodynamics, SPH)的核近似是最早的一种无网格近似, 它在计算域 Ω 上的近似函数为

$$u^h(\mathbf{x}) = \int_{\Omega} w(\mathbf{x} - \mathbf{x}_I) u(\mathbf{x}_I) d\Omega \quad (2)$$

式中: $u^h(\mathbf{x})$ 为 $u(\mathbf{x})$ 的近似函数; $w(\mathbf{x} - \mathbf{x}_I)$ 为权函数; \mathbf{x} 和 \mathbf{x}_I 分别为计算区域中两任意点的坐标, 式(2)是 SPH 近似的积分形式, 在数值计算中, 将其离散化为

$$u^h(\mathbf{x}) = \sum_{I=1}^N w(\mathbf{x} - \mathbf{x}_I) u_I \Delta V_I \quad (3)$$

式中 \mathbf{x}_I 为支持域中节点 I 的坐标。支持域的形状可以是 I 点为圆点以 r 为半径的圆或圆球(三维), 也可以是以 I 点为中心的方形域($2d_x, 2d_y$ 边长的矩形域)等。关于支持域的选取方法详细见文献[7]。 N 为以 I 点为中心的支持域中的节点数; ΔV_I 为支持域的面积或体积(三维)。目前常用的权函数 $w(\mathbf{x} - \mathbf{x}_I)$ 有样条型权函数、高斯型权函数。

进一步可以将式(3)化简为

$$u^h(\mathbf{x}) = \sum_{I=1}^N \Phi_I(\mathbf{x}) u_I \quad (4)$$

式(4)中 $\Phi_I(\mathbf{x}) = w(\mathbf{x} - \mathbf{x}_I) \Delta V_I$, 称为形函数。关于 SPH 的详细介绍见文献[7]。

1.3 再生核近似

由于 SPH 方法用于求解有限区域问题时, 数值解在边界处的精度较低, Liu 等^[20]认为这是由于 SPH 的核近似在边界处不满足一致性条件所引起的, 因此其对式(2)的近似函数进行如下修正:

$$u^h(\mathbf{x}) = \int_{\Omega} c(\mathbf{x}, \mathbf{x}_I) (\mathbf{x} - \mathbf{x}_I) u(\mathbf{x}_I) d\Omega \quad (5)$$

式中 $c(\mathbf{x}, \mathbf{x}_I)$ 称为修正函数, 它通过令式(5)满足一致性条件来求得。对于一维问题, 所谓一致性条件即为使式(5)满足以下 2 个等式:

$$\sum_{I=1}^N c(\mathbf{x}, \mathbf{x}_I) (\mathbf{x} - \mathbf{x}_I) \Delta V_I = 1 \quad (6)$$

$$\sum_{I=1}^N c(\mathbf{x}, \mathbf{x}_I) (\mathbf{x} - \mathbf{x}_I) \Delta V_I \mathbf{x}_I = \mathbf{x} \quad (7)$$

离散形式的再生核近似如下:

$$u^h(\mathbf{x}) = \sum_{I=1}^N c(\mathbf{x}, \mathbf{x}_I)(\mathbf{x} - \mathbf{x}_I)u_I \Delta V_I = \sum_{I=1}^N \Phi_I(\mathbf{x})u_I \quad (8)$$

式中 $\Phi_I(\mathbf{x})=c(\mathbf{x}, \mathbf{x}_I)w(\mathbf{x}-\mathbf{x}_I)\Delta V_I$ ，为再生核近似的形函数。

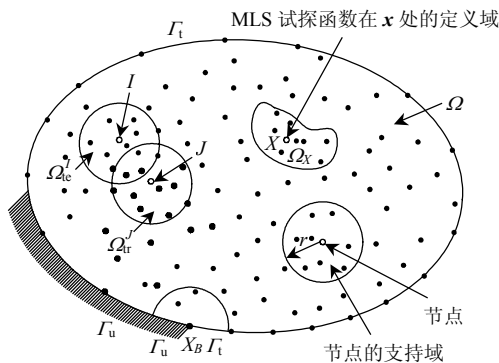
1.4 移动最小二乘近似

移动最小二乘近似方法(moving least-squares, MLS)是目前应用最为广泛的一种近似方法。Lancaster 和 Salkauskas^[45]、Jin^[46]等都对 MLS 的特性进行了详细的研究。关于 MLS 详细的推导见文献[5]。

假定在计算区域 Ω 中，以节点 \mathbf{x} 所在的子域 Ω_x (如图 1 所示)所包含的节点 $\mathbf{x}_I(I=1, \dots, N)$ 来构建位移近似函数 $u^h(\mathbf{x})$ ，其表达式如下：

$$u^h(\mathbf{x}) = \sum_{i=1}^m p_i(\mathbf{x})a_i(\mathbf{x}) = \mathbf{p}^T(\mathbf{x})\mathbf{a}(\mathbf{x}) \quad (9)$$

式中： $u^h(\mathbf{x})$ 为位移矢量的近似函数； $\mathbf{p}^T(\mathbf{x})$ 为 $N \times m$ 的矩阵； $\mathbf{a}(\mathbf{x})$ 为 $m \times 1$ 矢量。式(9)的右端表示矩阵与矢量的乘积。下文的矩阵与矢量的乘积都为如式(9)的表示形式。



Γ_u 为第2类边界条件； Γ_1 为第1类边界条件。

图 1 MLS 的试探函数的定义区域及节点的支持域

Fig. 1 Supports of nodes, the domain of definition of MLS approximation for the trial function at a point

对于二维问题 $\mathbf{x}^T=[x, y]$ 为计算点 \mathbf{x} 空间坐标， $\mathbf{p}^T(\mathbf{x})=[p_1(\mathbf{x}), p_2(\mathbf{x}), \dots, p_m(\mathbf{x})]$ 为单项式基函数， m 为基函数的个数； $\mathbf{a}(\mathbf{x})$ 为系数矢量， $\mathbf{a}(\mathbf{x})=[a_1(\mathbf{x}), a_2(\mathbf{x}), \dots, a_m(\mathbf{x})]$ ， $a_1(\mathbf{x})$ 等为待定系数。通常用多项式作为基函数，对于二维问题，其基函数可取为：

线性基函数

$$\mathbf{p}^T(\mathbf{x})=[1, x, y], \quad m=3 \quad (10)$$

二次基函数

$$\mathbf{p}^T(\mathbf{x})=[1, x, y, x^2, xy, y^2], \quad m=6 \quad (11)$$

对于三维问题，其基函数可取为：

线性基函数

$$\mathbf{p}^T(\mathbf{x})=[1, x, y, z], \quad m=4 \quad (12)$$

二次基函数

$$\mathbf{p}^T(\mathbf{x})=[1, x, y, z, x^2, y^2, z^2, xy, yx, xz], \quad m=10 \quad (13)$$

系数矢量 $\mathbf{a}(\mathbf{x})$ 是通过试探函数在点 $\mathbf{x} = \mathbf{x}_I$ 处误差的加权平方和来决定，其定义式如下：

$$J(\mathbf{a}(\mathbf{x})) = \sum_{I=1}^N w_I(\mathbf{x})[u^h(\mathbf{x}_I) - u_I]^2 = \sum_{I=1}^N w_I(\mathbf{x}) \cdot$$

$$[\mathbf{p}^T(\mathbf{x}_I)\mathbf{a}(\mathbf{x}) - u_I]^2 = (\mathbf{p}\mathbf{a}(\mathbf{x}) - \mathbf{u})\mathbf{w}(\mathbf{p}\mathbf{a}(\mathbf{x}) - \mathbf{u}) \quad (14)$$

式中： \mathbf{x}_I 为 I 点的位置矢量； $w_I(\mathbf{x})$ 为 I 点的权函数，这里要求 $w_I(\mathbf{x}) > 0$ ；矩阵矢量 \mathbf{p} 和 \mathbf{w} 的定义如下：

$$\mathbf{p} = \begin{bmatrix} 1 & x_1 & y_1 & \cdots & p_m(x_1) \\ 1 & x_2 & y_2 & \cdots & p_m(x_2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_N & y_N & \cdots & p_m(x_N) \end{bmatrix} \quad (15)$$

$$\mathbf{w} = \begin{bmatrix} w_1(\mathbf{x}) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & w_N(\mathbf{x}) \end{bmatrix} \quad (16)$$

矢量 \mathbf{u} 的定义为

$$\mathbf{u}^T = [u_1, u_2, \dots, u_N] \quad (17)$$

式中 u_I 为虚拟点值，它与近似函数 $u^h(\mathbf{x})$ 的值的差别如图 2 所示。

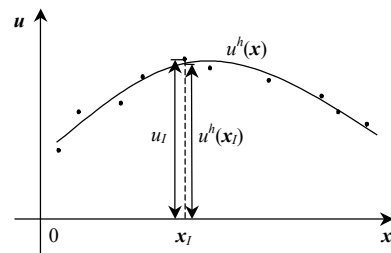


图 2 虚拟点值与近似值的关系图

Fig. 2 Distinction between the approximation values and the fictitious nodal values

令式(14)中 J 取极小值可得

$$\frac{\partial J(\mathbf{a}(\mathbf{x}))}{\partial(\mathbf{a}(\mathbf{x}))} = 2 \sum_{I=1}^N w_I(\mathbf{x}) \left[\sum_{j=1}^m p_j(\mathbf{x}_I) a_j(\mathbf{x}) - u_I \right] \cdot$$

$$p_j(\mathbf{x}_I) = \mathbf{A}(\mathbf{x})\mathbf{a}(\mathbf{x}) - \mathbf{B}(\mathbf{x})\mathbf{u} = 0 \quad (18)$$

其中：

$$\mathbf{A}(\mathbf{x}) = \mathbf{p}^T \mathbf{w} \mathbf{p} = \mathbf{B}(\mathbf{x}) \mathbf{p} \quad (19)$$

$$\mathbf{B}(\mathbf{x}) = \mathbf{p}^T \mathbf{w} \quad (20)$$

由式(18)，计算可得出 $\mathbf{a}(\mathbf{x})$ ，并将其代入式(9)可得

$$u^h(\mathbf{x}) = \Phi^T(\mathbf{x})\mathbf{u} = \sum_{I=1}^N \Phi_I(\mathbf{x})u_I, \quad \mathbf{x} \in \Omega_x \quad (21)$$

其中 $\Phi^T(\mathbf{x}) = \mathbf{p}^T(\mathbf{x})\mathbf{A}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x})$ 为形函数。它的一阶导数及二阶导数的计算如下^[1]。令：

$$\Phi^T(\mathbf{x}) = \mathbf{p}^T(\mathbf{x})\mathbf{A}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x}) = \gamma^T(\mathbf{x})\mathbf{B}(\mathbf{x}) \quad (22)$$

式中

$$\boldsymbol{\gamma}^T(\mathbf{x}) = \mathbf{p}^T(\mathbf{x})\mathbf{A}^{-1}(\mathbf{x}) \quad (23)$$

由式(19)可知, \mathbf{A} 是对称的, $\boldsymbol{\gamma}(\mathbf{x})$ 能够从式(23)计算得来: 将 \mathbf{A} 乘式(23)两端得 $\boldsymbol{\gamma}^T\mathbf{A} = \mathbf{p}^T\mathbf{I}$, 再取转置得 $(\boldsymbol{\gamma}^T\mathbf{A})^T = (\mathbf{p}^T\mathbf{I})^T$, 根据矩阵运算规则, 可有

$$\mathbf{A}\boldsymbol{\gamma} = \mathbf{p} \quad (24)$$

因此关于 $\boldsymbol{\gamma}$ 的各阶导数的计算可以由式(25)~(26)获得:

$$\mathbf{A}\boldsymbol{\gamma}'_i = \mathbf{p}'_i - \mathbf{A}'_i\boldsymbol{\gamma} \quad (25)$$

$$\mathbf{A}\boldsymbol{\gamma}'_{ij} = \mathbf{p}'_{ij} - (\mathbf{A}'_i\boldsymbol{\gamma}'_j + \mathbf{A}'_j\boldsymbol{\gamma}'_i + \mathbf{A}'_{ij}\boldsymbol{\gamma}) \quad (26)$$

式中下标 i, j 表示对坐标 x 或 y 的导数。据式(22)可得形函数的一阶和二阶偏导数的计算如式(27)~(28):

$$[\boldsymbol{\Phi}^T]'_i = [\boldsymbol{\gamma}^T]'_i\mathbf{B} + \boldsymbol{\gamma}^T\mathbf{B}'_i \quad (27)$$

$$[\boldsymbol{\Phi}^T]_{ij}' = [\boldsymbol{\gamma}^T]_{ij}'\mathbf{B} + [\boldsymbol{\gamma}^T]_i\mathbf{B}'_j + [\boldsymbol{\gamma}^T]_j\mathbf{B}'_i + \boldsymbol{\gamma}^T\mathbf{B}'_{ij} \quad (28)$$

1.5 单元分解近似

Duarte 等^[47]基于 MLS 近似提出了一种新的近似函数, 称为单元分解(partition of unity, PU)近似。其近似函数的表达式为

$$u^h(\mathbf{x}) = \sum_{l=1}^N \Phi_l(\mathbf{x})[u_l + \sum_{j=1}^M b_{jl}q_j(\mathbf{x})] \quad (29)$$

式中, 等式右端括号内第 2 项 $\sum_{j=1}^M b_{jl}q_j(\mathbf{x})$ 为附加项;

$q_j(\mathbf{x})$ 为附加基; M 为附加基的项数。 $\sum_{j=1}^M b_{jl}q_j(\mathbf{x})$ 的加入是为了改进运算和加速收敛。 b_{jl} 为附加项的系数, PU 近似方法的详细内容可参见文献[47]。

1.6 径向基函数近似

由于 MLS 近似的形函数不满足 δ_{ij} 函数特性 (即: $i=j, \delta_{ij}=1; i \neq j, \delta_{ij}=0$), 对于第 1 类边界条件 $u = \bar{u}_1, u^h(\mathbf{x}_l) = \sum_{j=1}^N \Phi_j(\mathbf{x}_l)u_j \neq \bar{u}_1$, 因此很难处理第 1 类边界条件。吴宗敏^[36]提出用正定紧支径向基函数代替 MLS 中的单项式基函数。他提出的 2 种正定紧支径向基函数如式(30)~(31):

$$R_l(\mathbf{x}) = \begin{cases} (1-d_l/r_l)^4(4+16d_l/r_l+12d_l^2/r_l^2+3d_l^3/r_l^3), & 0 \leq d_l/r_l \leq 1 \\ 0, & \text{其他} \end{cases} \quad (30)$$

$$R_l(\mathbf{x}) = \begin{cases} (1-\frac{d_l}{r_l})^6(6+36\frac{d_l}{r_l}+82\frac{d_l^2}{r_l^2}+72\frac{d_l^3}{r_l^3}+30\frac{d_l^4}{r_l^4}+5\frac{d_l^5}{r_l^5}), & 0 \leq \frac{d_l}{r_l} \leq 1 \\ 0, & \text{其他} \end{cases} \quad (31)$$

式中: $d_l = \|\mathbf{x} - \mathbf{x}_l\|$ 为点 \mathbf{x} 到节点 \mathbf{x}_l 的距离; r_l 为径向基函数的支持域半径。

$u^h(\mathbf{x})$ 的表达式如下:

$$u^h(\mathbf{x}) = \mathbf{R}^T(\mathbf{x})\mathbf{a}, \quad \mathbf{x} \in \Omega_x \quad (32)$$

式中: $\mathbf{R}^T(\mathbf{x}) = [R_1(\mathbf{x}), R_2(\mathbf{x}), \dots, R_M(\mathbf{x})]$ 为中心在点 \mathbf{x}_l 处的径向基函数, \mathbf{a} 为待定系数。令近似函数 $u^h(\mathbf{x})$ 在节点 \mathbf{x}_l 处的值等于函数 $u(\mathbf{x})$ 在该节点处的值 u_l , 即 $u^h(\mathbf{x}_l) = u_l$, 可得支持域中的所包含的 N 个节点的线性方程组:

$$\mathbf{R}_0\mathbf{a} = \mathbf{u} \quad (33)$$

式中:

$$\mathbf{u}^T = [u_1, u_2, \dots, u_N] \quad (34)$$

$$\mathbf{R}_0 = \begin{bmatrix} R_1(\mathbf{x}_1) & R_2(\mathbf{x}_1) & \cdots & R_N(\mathbf{x}_1) \\ R_1(\mathbf{x}_2) & R_2(\mathbf{x}_2) & \cdots & R_N(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ R_1(\mathbf{x}_N) & R_2(\mathbf{x}_N) & \cdots & R_N(\mathbf{x}_N) \end{bmatrix} \quad (35)$$

由式(33)可得出系数列阵 \mathbf{a} , 代入式(32)中, 可得出近似函数的表达式如下:

$$u^h(\mathbf{x}) = \mathbf{R}^T(\mathbf{x})\mathbf{R}_0^{-1}\mathbf{u} = \sum_{l=1}^N \Phi_l(\mathbf{x})u_l \quad (36)$$

其中形函数为 $\Phi(\mathbf{x}) = \mathbf{R}^T(\mathbf{x})\mathbf{R}_0^{-1}$ 。从上面的推导过程可以看出: $u^h(\mathbf{x}_l) = u(\mathbf{x}_l) = u_l$, 形函数的计算只依靠子域中节点的分布, 满足 $\Phi_l(\mathbf{x}_j) = \delta_{lj}$, 因此很容易施加第 1 类边界条件。图 3 给出了径向基函数的形函数值的分布, 可以看出它只在中心点(0,0)处有值且等于 1, 其它全为零。径向基函数近似方法的详细内容可参见文献[8]。

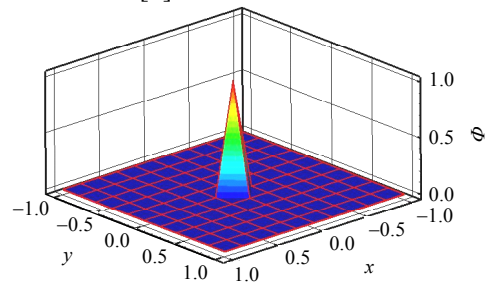


图3 径向基函数的形函数

Fig. 3 Shape function of RBFs

1.7 点插值近似

点插值法近似函数^[39]与 MLS 近似方法基本类似, 只是点插值法的支持域中所包含的节点数 N 等于基函数的个数 m , 而 MLS 中支持域中所包含的节点数 $N \geq m$ 。其近似函数的表达式为

$$u^h(\mathbf{x}) = \sum_{i=1}^m p_i(\mathbf{x})a_i = \mathbf{p}^T(\mathbf{x})\mathbf{a} \quad (37)$$

式(37)中的系数矢量 \mathbf{a} 通过支持域中各函数的值来

确定，计算式为

$$\begin{cases} u_1 = \sum_{i=1}^m a_i p_i(\mathbf{x}_1) = a_1 + a_2 x_1 + a_3 y_1 + \dots + a_m p_m(\mathbf{x}_1) \\ u_2 = \sum_{i=1}^m a_i p_i(\mathbf{x}_2) = a_1 + a_2 x_2 + a_3 y_2 + \dots + a_m p_m(\mathbf{x}_2) \\ \dots \\ u_N = \sum_{i=1}^m a_i p_i(\mathbf{x}_N) = a_1 + a_2 x_N + a_3 y_N + \dots + a_m p_m(\mathbf{x}_N) \end{cases} \quad (38)$$

式(38)能够写成矩阵形式：

$$\mathbf{u} = \mathbf{p}_m \mathbf{a} \quad (39)$$

式中

$$\mathbf{p}_m = \begin{bmatrix} 1 & x_1 & y_1 & \dots & p_m(x_1) \\ 1 & x_2 & y_2 & \dots & p_m(x_2) \\ 1 & x_3 & y_3 & \dots & p_m(x_3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_N & y_N & \dots & p_m(x_N) \end{bmatrix} \quad (40)$$

由式(39)可得出系数矢量 $\mathbf{a}(\mathbf{x})$ ，将其代入到式(37)可得

$$\mathbf{u}^h(\mathbf{x}) = \mathbf{p}^T(\mathbf{x}) \mathbf{p}_m^{-1} \mathbf{u} = \sum_{I=1}^N \Phi_I(\mathbf{x}) u_I = \Phi^T(\mathbf{x}) \mathbf{u} \quad (41)$$

其形函数为 $\Phi^T(\mathbf{x}) = \mathbf{p}^T(\mathbf{x}) \mathbf{p}_m^{-1}$ 。由于该方法形函数是采用通过节点值的方法建立的，故形函数也满足 $\Phi_I(\mathbf{x}_J) = \delta_{IJ}$ 。

2 权函数的选取

在构建近似函数(以 MLS 为例)时，权函数起着重要的作用。权函数的选取要满足以下几个条件：

- 1) 紧支性，即在支持域内不为零，其余全为零。
- 2) 单位特性，即在支持域中心的值为 1。
- 3) 对称性，即其沿支持域中心对称。

常见的权函数有：样条型权函数和高斯型权函数。它们在节点 \mathbf{x}_I 处取最大值，具有紧支特性，即当 $w(\mathbf{x}_I)$ 在以 r 为半径的支持域内 $w_I(\mathbf{x}) > 0$ ，其他情况均为 $w_I(\mathbf{x}) = 0$ 。常用的权函数为：

四次样条函数^[2](W1)

$$w_I(\mathbf{x}) = \begin{cases} 1 - 6D_I^2 + 8D_I^3 - 3D_I^4, & 0 \leq D_I \leq 1 \\ 0, & D_I > 1 \end{cases} \quad (42)$$

三次样条函数^[8](W2)

$$w_I(\mathbf{x}) = \begin{cases} 2/3 - 4D_I^2 + 4D_I^3, & D_I \leq 0.5 \\ 4/3 - 4D_I + 4D_I^2 - 4/3D_I^3, & 0.5 < D_I \leq 1 \\ 0, & D_I > 1 \end{cases} \quad (43)$$

高斯函数^[19](W3)

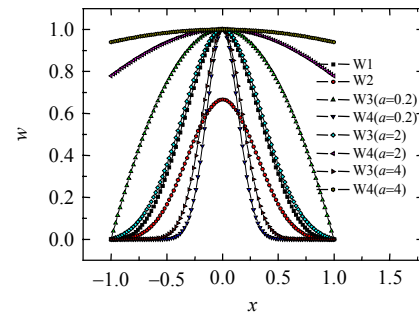
$$w_I(\mathbf{x}) = \begin{cases} \frac{\exp(-D_I^2 \alpha^2) - \exp(-\alpha^2)}{1 - \exp(-\alpha^2)}, & 0 \leq D_I \leq 1 \\ 0, & D_I > 1 \end{cases} \quad (44)$$

高斯函数^[8](W4)

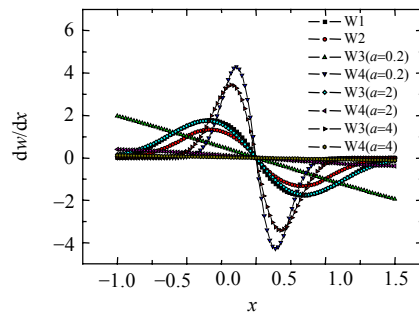
$$w_I(\mathbf{x}) = \begin{cases} \exp(-D_I^2 / \alpha^2), & 0 \leq D_I \leq 1 \\ 0, & D_I > 1 \end{cases} \quad (45)$$

式(42)~(45)中 $D_I = \sqrt{(x-x_I)^2 + (y-y_I)^2} / r_I$ ， r_I 为第 I 个节点 $(\mathbf{x}_I, \mathbf{y}_I)$ 的支持域尺寸， $\alpha = r_I / c_I$ ， c_I 为常数，文献[19]给出了选取 c_I 的方法。

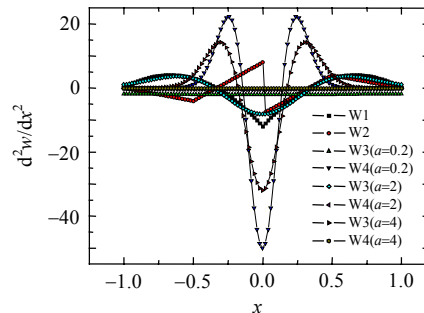
从式(42)~(45)中可以看出 W1 具有三阶连续性，而 W2 具有二阶连续性。后 2 个高斯型函数具有任意阶连续。图 4 给出这些权函数及其一阶导数和二阶导数的一维分布情况。对 2 个高斯型函数的 α 分别给定为常数 0.2, 2.0, 4.0 的 3 种情况进行比较，结果如图 4 所示。从图中可以看出在 W3 中 α



(a) 权函数



(b) 权函数的一阶导数



(c) 权函数的二阶导数

图 4 权函数及其一阶导数、二阶导数的一维分布图

Fig. 4 Weight functions and their first and second order derivatives

值应尽量大,而在 W4 中 α 值应尽量小,这样构造出的权函数好。从二阶导数图可以看出, W2 的导数已经不连续,而且它的权函数的最大值并不为 1。W1 是一种较好的权函数,文献[8,48]都证明 4 次样条型权函数是一种很好的权函数。

3 移动最小二乘近似的形函数

如前所述,形函数(以 MLS 为例)在无网格方法计算中起着重要的作用。在无网格方法数值计算中一般会用到形函数的一阶与二阶偏导数,很少情况下会用到三阶或三阶以上偏导数,因此在本文给出 4 种权函数所构建的形函数及一阶、二阶偏导数情况。图 5~图 7 分别给出 4 种权函数所构建的形函数的值及一阶与二阶偏导数值的一维分布情况,从图中可以看出样条型或高斯型权函数所构建的形函数在中心点(一维是(0),二维是(0,0))处的值并不等于 1,同时基函数 $m=6$ 时的形函数中心点处的值大于基函数为 3 的中心点处的值。从图中还可以看出其形函数与形函数的二阶偏导数沿中心点是对称的,而一阶偏导数沿中心点是反对称的。

由于权函数 W1 与 W2 所构建的形函数的特性比较一致, W3 与 W4 所构建的形函数的特性比较一致,这里只给出基函数 $m=3$ 时 W1 与 W3 的二维

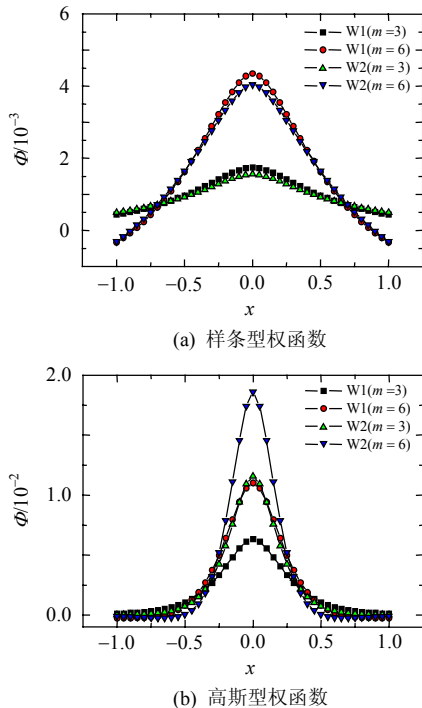


图 5 样条型与高斯型权函数所构建的形函数
Fig. 5 Shape functions using spline and Gaussian type weight function

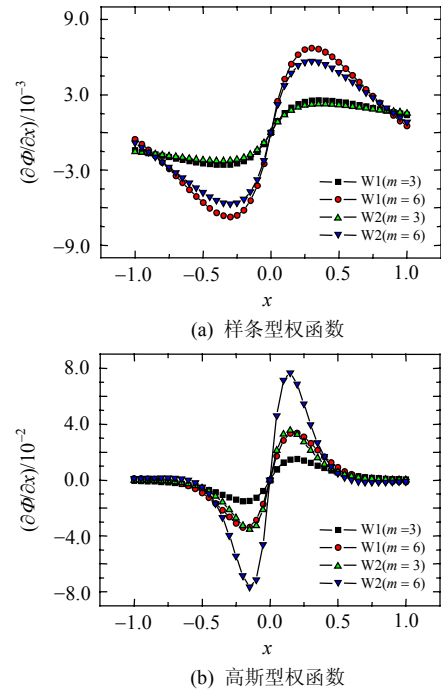


图 6 样条型与高斯型权函数所构建的形函数的一阶导数
Fig. 6 First order derivatives of shape functions using spline and Gaussian type weight function

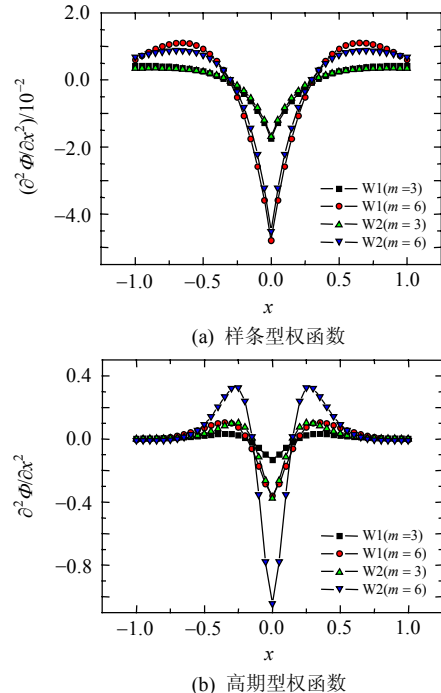


图 7 样条型与高斯型权函数所构建的形函数的二阶导数
Fig. 7 Second order derivatives of shape functions using spline and Gaussian type weight function

形函数的分布图和一阶与二阶偏导数图,如图 8 与图 9 所示。

二维权函数的计算方法为

$$w_i(x, y) = w_{ix}(x) \cdot w_{iy}(y) \quad (46)$$

从式(46)可以看出,形函数对 x 的偏导数与对 y

的偏导数一样，只是换了个方向，因此在二维显示图里只给出对 x 方向的偏导数。

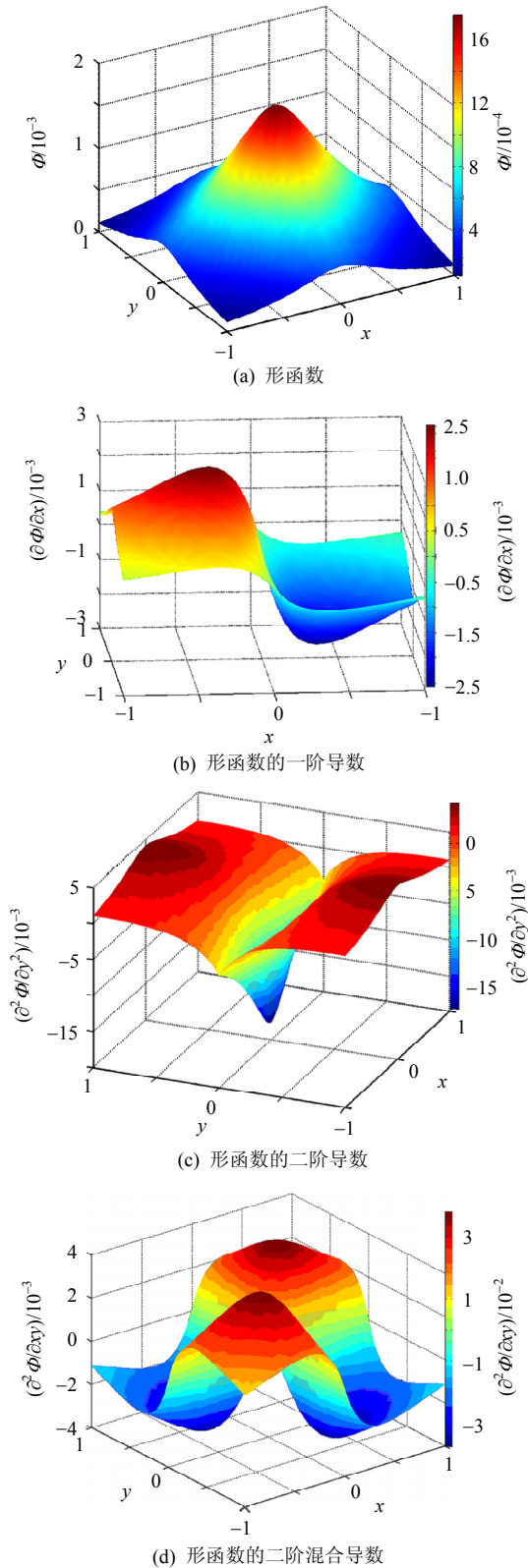


图 8 样条型(W1)权函数所构建的形函数及一阶与二阶偏导数二维分布
 Fig. 8 Shape functions and their first and second order derivatives using spline type weight function (W1)

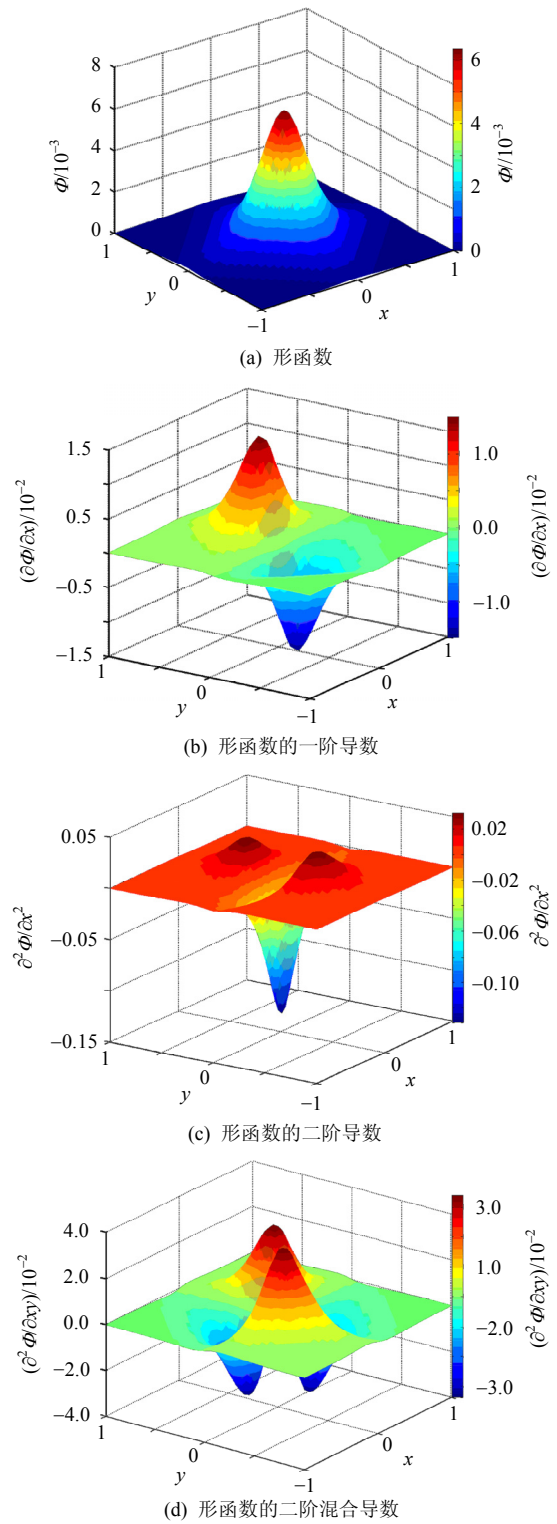


图 9 高斯型(W3)权函数所构建的形函数及一阶与二阶偏导数二维分布
 Fig. 9 Shape functions and their first and second order derivatives using Gaussian type weight function (W3)

4 微分方程的离散方法

4.1 加权残量法

一般的偏微分方程是物理过程的控制方程的

强形式,能获得强形式控制方程的精确解是理想的,但是由于实际工程问题的复杂性,有时要获得问题的强解是十分困难的。在数值计算的发展过程中基于强形式的数值方法已经得到积极的发展,被用于获取物理过程的近似解。有限差分法(finite difference method, FDM)是一种典型的基于强式的方法,它将一个函数在局部域上的导数表达为有限差分的形式(Taylor 级数),用于求解强形式控制方程的近似解^[8]。有限容积法实质上也是获得强形式控制方程近似解的数值方法。

微分方程的弱形式的解相对于强形式的解来说,对近似函数的连续性要求比较弱,基于数学或物理原理积分方程的弱形式被引入到控制方程中。2种常用的获得微分方程弱形式的方法是:1)变分法;2)加权残量法。加权残量法是目前求解偏微分方程弱解的一种有效和广泛应用的方法。

对于一般边值问题的偏微分方程,可写为

$$\begin{cases} L(u) = f, & \text{在 } \Omega \\ B(u) = g, & \text{在 } \Gamma \text{ 上} \end{cases} \quad (47)$$

式(47)中 u 可以用第3节中获得的近似函数代入,但是在通常的情况下这些近似函数(试探函数)并不能满足式(47)的左右两边相等,即:

$$\begin{cases} L(u^h) \neq f, & \text{在 } \Omega \\ B(u^h) \neq g, & \text{在 } \Gamma \text{ 上} \end{cases} \quad (48)$$

因此能够获得式(48)的残量,它们分别定义为 R_j 和 \bar{R}_j :

$$\begin{cases} R_j = L(u^h) - f, & \text{在 } \Omega \\ \bar{R}_j = B(u^h) - g, & \text{在 } \Gamma \text{ 上} \end{cases} \quad (49)$$

如果近似函数是式(47)的精确解,那么式(49)中的 R_j 和 \bar{R}_j 均为零;但在实际问题中精确解很难获得,也即式(49)中的 R_j 和 \bar{R}_j 不为零,因此为了获得微分方程(47)合理的近似解,强迫式(49)的残量在某种平均意义上等于零是通过设置残量的加权积分在计算区域内和边界上趋于零来获得,即:

$$\int_{\Omega} v R_j d\Omega + \int_{\Gamma} \bar{v} \bar{R}_j d\Gamma = 0 \quad (50)$$

式中 v 和 \bar{v} 为检验函数(test function)。如果式(50)对于任意的函数 v 和 \bar{v} 都成立,那么式(47)必定在计算区域内 Ω 和边界 Γ 上任意点都满足。式(50)也称为式(47)的等效积分形式。

4.2 配点法

配点法是在计算中检验函数取 δ 函数,使得微分方程式(47)在所有的计算节点(即配点)处加权余

量为零。设计算区域内节点的检验函数为 v , 边界节点的检验函数为 \bar{v} , 那么它们的定义如下:

$$\begin{cases} v_j = \delta(\mathbf{x} - \mathbf{x}_j) \\ \bar{v}_j = \delta(\mathbf{x} - \mathbf{x}_j) \end{cases} \quad (51)$$

这样式(47)的加权残量为

$$\begin{cases} \int_{\Omega} v(L(u^h) - f) = 0, & \text{在 } \Omega \\ \int_{\Gamma} \bar{v}(B(u^h) - g) = 0, & \text{在 } \Gamma \text{ 上} \end{cases} \quad (52)$$

式(52)就变成了线性方程组:

$$\begin{cases} \sum_{I=1}^M \left(\sum_{J=1}^N \Phi_J(\mathbf{x}) u_J - f \right) = 0, & \text{在 } \Omega \\ \sum_{I=1}^{M_1} \left(\sum_{J=1}^N \Phi_J(\mathbf{x}) u_J - g \right) = 0, & \text{在 } \Gamma \text{ 上} \end{cases} \quad (53)$$

式中 M 为计算区域内部的节点数; M_1 为边界上的节点数; N 为节点支持域中所包含的节点数。这样式(47)的微分方程就变为求式(53)的线性方程组,由式(53)求出的解是问题的强解。

4.3 Galerkin 方法

在一般的 Galerkin 方法中,试探函数和检验函数取自同一个函数空间,它的检验函数取为

$$\begin{cases} v_j = \Phi_j \\ \bar{v}_j = -\Phi_j \end{cases} \quad (54)$$

则式(50)就变为

$$\int_{\Omega} \Phi_I(\mathbf{x}) \left(\sum_{J=1}^N \Phi_J(\mathbf{x}) u_J - f \right) d\Omega + \int_{\Gamma} \Phi_I(\mathbf{x}) \left(\sum_{J=1}^N \Phi_J(\mathbf{x}) u_J - g \right) d\Gamma = 0 \quad (55)$$

从式(55)可以看出,在 Galerkin 方法中插值函数中的形函数被用作检验函数,故得到的求解方程的系数矩阵是对称的。由此方法离散所构造的无网格方法是在整个计算区域 Ω 上进行积分,因此它需要借助于背景网格进行数值积分。

4.4 局部 Petrov-Galerkin 方法

如前所述,一般的 Galerkin 方法中试探函数与检验函数取自同一子空间,但在 Petrov-Galerkin 方法中试探函数与检验函数取自不同的子空间。在局部 Petrov-Galerkin 方法中,要求解的方程的余量在局部子域 Ω_x 内及其边界上 Γ 上的加权积分为零,即:

$$\int_{\Omega} v R_j d\Omega + \int_{\Gamma} \bar{v} \bar{R}_j d\Gamma = 0 \quad (56)$$

式(56)的边界 Γ_I 包括了子域 Ω_x 所包含的边界,它可能包括2部分:一部分为在计算区域内部,这个边界没有给定边界条件;一部分为计算区域的边界,给定了边界条件。

由于式(56)试探函数与检验函数取自不同的子空间, 所以它所得到的求解方程的系数矩阵一般是不对称的。由于它的积分是在规则的子域 Ω_x 上进行, 因此不需要引入背景网格进行积分, 目前常用高斯型积分公式进行数值积分, 可以看出这样构建的无网格方法是一种真正的无网格方法。

5 结论

1) 无网格方法的构建主要包括近似函数的构建方法和微分方程的离散方法 2 个部分, 根据近似函数构建方法的不同和微分方程离散方法的不同可以构建出不同的无网格方法。

2) 在近似函数中, 径向基函数和点插值法的形函数具有 δ 函数特性, 但在计算一些问题时, 稳定性较差; 其他几种近似函数的形函数均不满足 δ 函数特性, 在应用这些近似函数时, 对于第 1 类边界条件需要特殊处理。

3) 从权函数的特性可以看出, 高斯型权函数具有任意阶连续, 而三次样条权函数二阶导数不连续, 4 次样条权函数具有二阶连续, 因此在计算高阶导数时, 需要用高斯型权函数或更高次的样条权函数。

4) 无网格方法的 3 种离散方法中, 配点法不需要数值积分, 但对配点的选取特别严格, 有时稳定性较差, Galerkin 方法需要背景网格进行积分, 它不是真正的无网格方法, 而在 Petrov-Galerkin 方法中其试探函数与检验函数选自同一子空间, 可以直接进行数值积分, 但它的计算工作量较大。

参考文献

- [1] Belytschko T, Krongauz Y, Orgna D, et al. Meshless methods: an overview and recent developments[J]. *Computer Methods in Applied Mechanics and Engineering*, 1996, 139(1-3): 3-47.
- [2] Atluri S N, Shen S P. The meshless local Petrov-Galerkin method [M]. Encino, USA: Tech Science Press, 2002: 93-124.
- [3] Li S F, Liu W K. Meshfree and particle methods and their application [J]. *Applied Mechanics Reviews*, 2002, 55(1): 1-34.
- [4] Li S F, Liu W K. Meshfree particle methods[M]. New York: Springer, 2004: 68-80.
- [5] Atluri S N. The meshless local Petrov-Galerkin(MLPG) method for domain & boundary discretizations[M]. Encino USA: Tech Science Press, 2004: 105-145.
- [6] Liu G R. Mesh free methods: moving beyond the finite element method[M]. Boca Raton: CRC Press, 2003: 7-23.
- [7] Liu G R, Liu M B. Smoothed particle hydrodynamics: a meshfree particle method[M]. Singapore: World Scientific Publishing Co. Pte. Ltd, 2003: 54-81.
- [8] Liu G R, Gu Y T. An introduction to meshfree methods and their programming[M]. Berlin: Springer, 2005: 54-114.
- [9] 张雄, 刘岩. 无网格法[M]. 北京: 清华大学出版社, 2004: 1-13. Zhang Xiong, Liu Yan, Meshless method[M]. Beijing: Tsinghua University Press, 2004: 1-13 (in Chinese).
- [10] 刘更, 刘天祥, 谢琴. 无网格法及其应用[M]. 西安: 西北工业大学出版社, 2005: 48-62. Liu Geng, Liu Tianxiang, Xie Qing. Meshless and its application [M]. Xi'an: Northwestern Polytechnical University Press, 2005: 48-62.
- [11] 张雄, 宋康祖, 陆明万. 无网格法研究进展及其应用[J]. *计算力学学报*, 2003, 20(6): 730-742. Zhang Xiong, Song Kangzu, Lu Mingwan. Research progress and application of meshless method[J]. *Chinese Journal of Computational Mechanics*, 2003, 20(6): 730-742(in Chinese).
- [12] 张雄, 刘岩, 马上. 无网格法的理论及应用[J]. *力学进展*, 2009, 39(1): 1-36. Zhang Xiong, Liu Yan, Ma Shang. Meshfree methods and their applications [J]. *Advances in Mechanics*, 2009, 39(1): 1-36(in Chinese).
- [13] Lucy L B. A numerical approach to the testing of the fission hypothesis[J]. *The Astronomical Journal*, 1977, 82(12): 1013-1024.
- [14] Gingold R A, Monaghan J J. Smoothed particle hydrodynamics: theory and applications to non-spherical stars[J]. *Monthly Notices of the Royal Astronomical Society*, 1977, 181(5): 375-389.
- [15] Monaghan J J. Why particle methods work[J]. *SIAM Journal on Scientific and Statistical Computing*, 1982, 3(4): 422-433.
- [16] Nayroles B, Touzot G, Villon P. The diffuse approximation[J]. *C R Acad Sci Paris Ser II*, 1991, 313(3): 133-138.
- [17] Nayroles B, Touzot G, Villon P. The diffuse element method[J]. *C R Acad Sci Paris Ser II*, 1991, 313(3): 293-296.
- [18] Nayroles B, Touzot G, Villon P. Generalizing the finite element method: diffuse approximation and diffuse element[J]. *Computational Mechanics*, 1992, 10(5): 307-318.
- [19] Belytschko T, Lu Y Y, Gu L. Element-free galerkin method [J]. *International Journal for Numerical Methods in Engineering*, 1994, 37(2): 229-256.
- [20] Liu W K, Jun S, Zhang Y F. Reproducing kernel particle methods [J]. *International Journal for Numerical Methods in Fluids*, 1995, 20(9): 1081-1106.
- [21] Liu W K, Chen Y. Wavelet and multiple scale reproducing kernel method[J]. *International Journal for Numerical Methods in Fluids*, 1995, 21(10): 901-933.
- [22] Duarte C A, Oden J T. An h-p adaptive method using clouds [J]. *Computer Methods in Applied Mechanics and Engineering*, 1996, 139(1-4): 237-262.
- [23] Braun J, Sambridge M. A numerical method for solving partial differential equations on highly irregular evolving grids[J]. *Nature*, 1995, 376(6542): 655-660.
- [24] 王兆清, 冯伟. 自然单元法研究进展[J]. *力学进展*, 2004, 34(4): 437-445. Wang Zaoqing, Feng Wei. Advances in natural element method [J]. *Advances in Mechanics*, 2004, 34(4): 437-445(in Chinese).
- [25] 卢波, 葛修润, 程永辉. 自然单元法研究进展[J]. *岩石力学与工程学报*, 2004, 23(增 1): 4319-4324. Lu Bo, Ge Xiouyun, Cheng Yonghui. Research progress in natural element method[J]. *Chinese Journal of Rock Mechanics and*

- Engineering, 2004, 23(S1): 4319-4324.
- [26] Oñate E, Idelsohn S, Zienkiewicz OC, et al. A finite point method in computational mechanics: Applications to convective transport and fluid flow[J]. International Journal for Numerical Methods in Engineering, 1996, 39(22): 3839-3866.
- [27] Yagawa G, Yamada T. Free mesh method: A new meshless method[J]. Computational Mechanics, 1996, 18(5): 383-386.
- [28] Li S F, Liu W K. Moving least-square reproducing kernel methods(II) fourier analysis[J]. Computer Methods in Applied Mechanics and Engineering, 1996, 139(1-4): 159-193.
- [29] Liu W K, Li S F, Belytschko T. Moving least-square reproducing kernel methods (I) methodology and convergence[J]. Computer Methods in Applied Mechanics and Engineering, 1997, 143(1-2): 113-154.
- [30] Mukherjee Y X, Mukherjee S. The boundary node method for potential problems[J]. International Journal for Numerical Methods in Engineering, 1997, 40(5): 797-815.
- [31] Zhu T, Zhang J D, Atluri S N. A local boundary integral equation (LBIE) method in computational mechanics, and a meshless discretization approach[J]. Computational Mechanics, 1998, 21(3): 223-235.
- [32] Atluri S N, Zhu T. A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics[J]. Computational Mechanics, 1998, 22(2): 117-127.
- [33] Zhu T L. A new meshless regular local boundary integral equation (MRLBIE) approach[J]. International Journal for Numerical Methods in Engineering, 1999, 46(8): 1237-1252.
- [34] Powell M J D. The theory of radial basis function approximation in 1990[M]//Light W. Advances in Numerical Analysis. Oxford: Clarendon Press, 1992: 105-210.
- [35] Wu Z. Hermite-Birkhoff interpolation of scattered data by radial basis function[J]. Approximation Theory and Its Applications, 1992, 8(2): 1-10.
- [36] Wu Z M. Compactly supported positive definite radial functions [J]. Advances in Computational Mathematics, 1995, 4: 283-292.
- [37] Wendland H. Meshless Galerkin methods using radial basis function [J]. Mathematics of Computation, 1999, 68(228): 1521-1531.
- [38] Aluru N R. A point collocation method based on reproducing kernel approximations[J]. International Journal for Numerical Methods in Engineering, 2000, 47(6): 1083-1121.
- [39] Aluru N R, Li G. Finite cloud method: a true meshless technique based on reproducing kernel approximation[J]. International Journal for Numerical Methods in Engineering, 2001, 50(10): 2373-2410.
- [40] Liu G R, Gu Y T. A point interpolation method for two-dimensional solids[J]. International Journal for numerical methods in Engineering, 2001, 50(4): 937-951.
- [41] Liu G R, Gu Y T. A local radial point interpolation method(LRPIM) for free vibration analyses of 2-D solids[J]. Journal of Sound and Vibration, 2001, 246(1): 29-46.
- [42] 张见明, 姚振汉. 一种新型无网格法——杂交边界点法[M]//袁明武, 孙树立. 工程与科学中的计算力学. 北京: 北京大学出版社, 2001: 339-343.
- Zhang Jianming, Yao Zhenghan. A new type meshless method-hybrid boundar ynode method[M]//Yuan Mmingwu, SuShuli. Computational Mechanics of Engineering and Science. Beijing: Peking University Press, 2001: 339-343 (in Chinese).
- [43] Zhang X, Liu X H, Song Z K, et al. Least-square collocation meshless method[J]. International Journal for Numerical Methods in Engineering, 2001, 51(9): 1089-1100.
- [44] 张雄, 胡炜, 潘小飞, 等. 加权最小二乘无网格法[J]. 力学学报, 2003, 35(4): 425-431.
- Zhang Xiong, Hu Wei, Pan Xiaofei, et al. Meshless weighted least-square method[J]. Acta Mechanica Sinica, 2003, 35(4): 425-431(in Chinese).
- [45] Lancaster P, Salkauskas K. Surface generated by moving least squares methods[J]. Mathematics of Computation, 1981, 37(55): 141-158.
- [46] Jin X, Li G, Aluru N R. On the equivalence between least-square and kernal approximations in meshless methods[J]. CMES: Computer Modeling in Engineering & Sciences, 2001, 2(4): 447-462.
- [47] Duarte C A, Oden J T. Hp clouds: a meshless method to solve boundary-value problems[R]. Texas: Texas Institute for Computational and Applied Mathematics, 1995.
- [48] Atluri S N, Kim H, Cho J Y. A critical assessment of the truly meshless local Petrov-Galerkin(MLPG) and local boundary integral equation(LBIE) methods[J]. Computational Mechanics, 1999, 24(5): 348-372.



陶文铨

收稿日期: 2009-09-18.

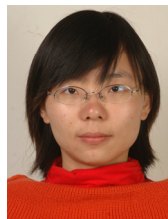
作者简介:

陶文铨(1939—), 男, 中国科学院院士, 西安交通大学教授, 博士生导师。研究领域或方向: 传热与流动的先进数值计算方法及其应用; 强化传热的基本理论与工程应用; 电子元器件的冷却技术; 湍流模型及其工程应用; 小型低温制冷机制和冷系统性能数值模拟; 高效换热器的优化设计与研发等, wqtao@mail.xjtu.edu.cn;



吴学红

吴学红(1979—), 男, 河南人, 博士, 现为河南省郑州轻工业学院机电工程学院讲师, 主要从事强化换热及数值模拟方面的研究, wuxh1212@163.com;



戴艳俊

戴艳俊(1986—), 女, 浙江人, 博士研究生, 主要从事强化换热与高性能算法方面的研究工作, daiyanjun@stu.xjtu.edu.cn.

(责任编辑 王庆霞)