

MATLAB 上的有限元网格自动生成

刘 瑶, 谭建国

(国防科技大学, 湖南 长沙 410073)

摘 要: DistMesh 是基于 MATLAB 的有限元网格自动生成程序, 具有程序代码简短明晰、网格质量高、移植性好等突出特点。但是因为 DistMesh 用以筛选和优化节点的距离函数是解析表达式, 用户无法实现对复杂模型的构造, 限制了 DistMesh 的工程应用和推广。由于非均匀有理 B 样条 (NURBS) 是计算机辅助技术 (CAD) 中三维建模的通用和标准技术, 故基于 MATLAB 中的 NURBS 工具箱, 提出一种利用 NURBS 曲线和曲面的方向来判断节点与曲线、曲面位置关系的方法, 以完成对节点的筛选和移动。基于力的平衡原理的迭代循环, 通过对比理想的与实际的网格长度, 以不断调整网格边长, 完成对网格的优化。将通用的 NURBS 建模与 DistMesh 相结合, 突破了 DistMesh 原来建模复杂的限制, 既能生成高质量的网格, 又提高了 DistMesh 的适用性。

关键词: 计算机辅助设计; 有限元网格生成; MATLAB; NURBS

中图分类号: TP311.1

文献标识码: A

文章编号: 1673-629X(2019)01-0017-04

doi: 10.3969/j.issn.1673-629X.2019.01.004

Automatic Generation of Finite Element Mesh on MATLAB

LIU Yao, TAN Jian-guo

(National University of Defense Technology, Changsha 410073, China)

Abstract: DistMesh is an automatic generation program of finite element mesh based on MATLAB, with prominent features such as short and clear code, high-quality mesh and great portability. However, because the DistMesh's distance function for filtering and optimizing nodes is an analytic expression, users are unable to construct complex models, which limits the application and promotion of DistMesh in engineering. Non-uniform rational B-spline (NURBS) is a general and standard technique for 3D modeling in computer-aided design (CAD), so based on the NURBS toolbox in MATLAB, we propose a method which determines the position relationship between nodes, curve and surface according to the direction of NURBS curves and surfaces, so as to complete the selection and movement of nodes. Based on the iterative cycle of force balance principle, by comparing the ideal mesh length with the actual mesh length, the mesh edge length is adjusted constantly to complete the mesh optimization. The combination of general NURBS modeling and DistMesh breaks through the limitations of DistMesh's original complex modeling, which can not only generate high-quality grids, but also improve the applicability of DistMesh.

Key words: CAD; finite element mesh; MATLAB; NURBS

0 引言

一般地, 可以将有限元分析^[1]划分为: 建立模型 (前处理)、计算求解、结果处理和评定 (后处理)。其中, 各阶段所用的时间占比分别为: 40% ~ 60%、5% ~ 10%、30% ~ 50%。在数值分析过程中, 网格划分^[2]是极其关键的一步, 可以说直接决定了最终结果的准确性、有效性。MATLAB^[3]具有简洁的语言、丰富的基础库函数、强大的数据处理和可视化能力等突出特点,

利用 MATLAB 进行有限元前处理是值得研究的问题。

DistMesh^[4]是基于 MATLAB 的有限元网格自动生成程序, 由加州大学伯克利分校的 Per-Olof Persson 和麻省理工学院数学系的 Gilbert Strang 开发, 相较于 MATLAB 自带的 PDE 工具箱仅能求解比较简单的二维模型的缺陷, 它可以生成三角形或四面体的网格, 对一般的二维和三维的建模都能进行良好的处理, 具有

收稿日期: 2018-03-18

修回日期: 2018-07-24

网络出版时间: 2018-11-15

基金项目: 国家自然科学基金 (11272351, 91441121); 湖南省研究生创新项目 (CX2016B001)

作者简介: 刘 瑶 (1995-), 男, 硕士研究生, CCF 会员 (90466G), 研究方向为航空宇航科学与技术; 谭建国, 博士, 研究员, 博导, 通讯作者, 研究方向为高超声速推进技术。

网络出版地址: <http://cnki.net/kcms/detail/61.1450.TP.20181114.1558.060.html>

程序简短明晰、网格质量高、可移植性好等突出优点,在 MATLAB 有限元前处理方面应用广泛。

但是由于 DistMesh 用来筛选和优化节点的距离函数是解析表达式,对简单的模型来说可以得到良好的应用,用户却无法实现对复杂模型的构造,这限制了 DistMesh 的工程应用和推广。目前比较通用的三维建模方式^[5]有网格建模、多边形建模、面片建模、非均匀有理 B 样条(NURBS)^[6]建模等,而因为良好的数学特性和强大的建模功能,NURBS 能够比传统的建模方式更好地控制物体表面的曲线度,从而创建出更逼真、生动的造型,成为 CAD 中几何定义的标准技术。通用的三维 CAD 软件也都采用 NURBS 建模方法。

文中基于 MATLAB 的 NURBS 工具箱^[7],对 DistMesh 进行改进,提出一种利用 NUBRS 曲线和曲面的方向来判断节点与曲线、曲面位置关系的方法,以完成对节点的筛选;再基于力的平衡原理对网格边的长度进行调整,以完成对节点位置的迭代优化。

1 网络生成器 DistMesh

1.1 DistMesh 设计原理

DistMesh 基本原理是使用距离函数 $d(x, y)$ 来定义几何边界,通过判断节点位置来筛选节点,再基于力的平衡来完成对网格的迭代优化。使用 $d(x, y)$ 计算每个节点到边界的距离,规定在区域内部时的距离为负,外部为正,从而将在边界外的点去除。后续优化是基于三角形网格的边与桁架结构或者是线性弹簧之间的物理类比^[8]。平面中的任何一组点都可以由 Delaunay 算法^[9]进行三角化。在物理模型中,三角形网格的边(顶点之间的连接)对应于桁条,顶点对应于桁架的节点。

每一条边的力 F 取决于它的当前长度 l 及其理想长度 l_0 的差值:

$$|F| = \begin{cases} k(l_0 - l), & l < l_0 \\ 0, & l > l_0 \end{cases} \quad (1)$$

其中, k 为模拟线性弹簧的系数,可设置为 1。排斥力 $F > 0$,使网格边界拉长;吸引力 $F < 0$,使网格边界缩短。对于所有的内部节点,存在平衡: $\sum_i F_i = 0$ 。

网格的疏密分布与几何形状的复杂程度有关^[10]。在该算法中,所需的网格大小由网格密度函数 $h(x, y)$ 提供。 $h(x, y)$ 由用户指定,可以使用自适应方法^[11]来实现局部特征变化,其大致是区域的边界之间的距离。自适应求解器可估计每个三角形中的误差,从而给出约束条件来精细化网格以获得更好的分布。

1.2 二维网络生成器算法描述

function [p, v] = distmesh2d($f_d, f_h, h_0, \text{bbox}$,

pfix, varargin)

程序的输入参数如下:

(1)几何形状由距离函数 f_d 给出。 $d = f_d(p)$ 返回值为每个节点位置 P 到其最近边界的带符号距离。

(2)所需的边缘长度 $h(x, y)$ 由函数 f_h 给出,为所有输入点返回 h 。

(3) h_0 是网格的初始大小。对于均匀网格 $h(x, y) = C$ 。

(4)边界(最大外围矩形范围)。

(5)固定节点(必须出现在网格中的点)。

输出有:

(1)节点位置 P : 该 $N \times 2$ 数组包含 N 个节点的 x, y 坐标。

(2)三角指标 t : 指定三角形的节点编号。

基本步骤为:

(1)在涵盖待划分几何形状的边界框中划分初始分布,初始分布可由用户任意规定。

(2)利用距离函数除去不在边界内的所有节点。

(3)进入主循环,节点位置不断迭代优化:

①去除质心在边界外的三角网格;

②计算理想的三角形边长,基于与实际边长的差值调整边长、移动节点;

③将移动到边界外的点拉回到边界上,避免节点流失。

(4)终止循环,终止条件由当前迭代中的节点最大移动距离决定。

1.3 存在的问题

距离函数是解析表达的,对于复杂图形很难构造,这大大限制了 DistMesh 的适用性。概括来说,算法中需要改进的地方有:步骤 2,利用距离函数去除边界外的节点;进入循环后的第 1 步,用 Delaunay 方法划分为网格后,去除质心在边界外的三角网格;进入循环后的第 3 步,将移出的节点移回边界。

2 基于 NURBS 的网格生成方法

2.1 改进方法

NUBRS 曲线和曲面都是具有方向的,这是解决问题的关键所在。对于二维平面,如图 1 所示,输入曲线参数后,若令外环曲线为逆时针、内环曲线为顺时针,则边界内部的点必在边界曲线的左侧。从任意点 P 指向其边界最近点 A 引向量 \vec{PA} ,再求最近点 A 处的切线 \vec{dA} , $\vec{PA} \times \vec{dA}$ 的 z 分量 z_1 为正时,点 P 在曲线的左侧,即边界内部;反之,在曲线外部。

$$P \text{ 在曲线} \begin{cases} \text{左侧, } z_1 > 0 \\ \text{右侧, } z_1 < 0 \end{cases} \quad (2)$$

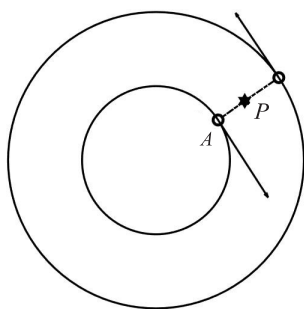


图1 点的位置判断

同理,对于三维实体,如图2所示的球面,其法线方向是 u 方向和 v 方向的向量积。曲面法线朝外,从曲面外一点 $pout$ 到其最近点 $pntm$ 引向量,由于最近点 $pntm$ 处法线和所引向量的夹角大于 90° ,二者数量积必为负;反之,对于曲面内的点数量积应为正。同理,当存在内表面时,若内表面法线方向朝内,则内表面内的点(即边界外的点)向最近点所引向量与最近点处法线的数量积为负,边界内的点为正。故可以定义外表面法线方向朝外,内表面法线方向朝内。

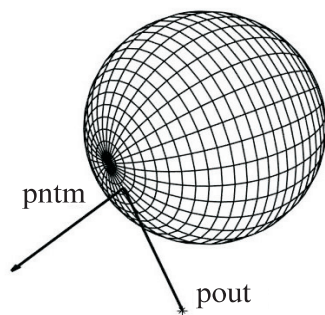


图2 通过点与曲面位置关系判断其内外

2.2 算法描述

二维平面指定调用函数: $\text{function}[p, t] = \text{nrbdist-mesh2d}(\text{crvout}, \text{crvin}, f_h, h_0, \text{bbox}, \text{pfix})$, 函数的输入参数由 f_d 变成了 crvout 和 crvin ; 用户只需要使用 NUBRS 工具箱进行内外环曲线的建模。

三维实体指定调用函数: $\text{function}[p, t] = \text{nrbdist-mesh3d}(\text{srfout}, \text{srfin}, f_h, h, \text{box}, \text{fix})$, 同理, 用户只需要使用 NUBRS 工具箱进行内外曲面的建模。

基本步骤如下:

(1) 先对涵盖用户待划分区域的外围矩形(长方体)创建初始分布: 二维平面为三角形网格, 三维实体为四面体网格。

(2) 判断外环曲线(曲面)的方向: 基于上述原理, 先在边界外确定一点, 通过该点在边界外这一既定结论反推曲线(曲面)的方向。

对于平面, 在外环曲线外有外围矩形, 外围矩形上的点 P 一定是在边界外的, 计算 P 到外环曲线的最近点 A , 再对点 A 处进行求导得到曲线方向 \vec{dA} ; 计算向

量 \vec{PA} 和 \vec{dA} 的向量积。因为点 P 在曲线外, 若向量积 z 分量为负, 曲线为逆时针, 符合要求; 若为正, 说明曲线为顺时针, 此时需要把曲线变换方向。

对于三维实体, 从外围长方体表面上找一点 $pout$, 这一点必在边界外; 找到外表面上最近点 $pntmout$, 求出 $pntmout$ 所在处法线方向 $wout$; 再从 $pout$ 向 $pntmout$ 引向量, 计算这个向量与法线方向的数量积 $zout$; 若 $zout$ 小于 0, 说明外表面法线方向朝外, 反之, 改变外表面方向。

(3) 判断用户是否输入内环(内表面): 如果存在, 应当先定义其方向; 否则直接执行第 5 步。

(4) 定义内环曲线(内表面)的方向: 内环曲线为顺时针方向, 内表面法向朝内。

对于平面, 与外环原理相同, 将外围矩形上的点替换为刚刚得到的外环上的 A 点, 此点必在内环外; 当这个点指向最近点的向量与最近点处导数的向量积 z 分量为正, 说明内环是顺时针; 否则, 变换方向。对于三维实体, 也从外表面上找点, 符合要求的内表面, 得到的数量积应该是正的, 否则将内表面变换方向。

(5) 除去边界外的点。与原方法计算距离的正负不同, 这里仍然利用向量积(数量积)。

对于平面, 因为满足条件的点都在曲线左侧或在曲线上, 故而向量积的 z 分量小于 0 的都是在边界外的, 需要去除; 对于三维实体, 因为外表面法线朝外, 内表面法线朝里, 满足条件的点都在曲面法线方向的相反侧, 则从节点向边界上最近点引向量, 与最近点处的法线方向是相同的(小于 90°), 故而数量积小于 0 的都是需要去除的。

(6) 进入循环, 进行节点的迭代优化。

① 利用网格质心与内外边界的位置关系, 去除质心在边界外的网格;

② 计算理想的网格边长, 基于与实际边长的差值调整边长、移动节点;

③ 将移动到边界外的点拉回到边界上, 把边界外点相对应的边界上最近点赋给边界外点即可。

(7) 终止迭代循环, 与原方法相同。

3 验证与分析

3.1 网格质量评价指标

对于三角形网格, 网格质量^[12]常根据三角形最大内切圆与其最小外接圆的半径比率来判断:

$$q = 2 \frac{r_{in}}{r_{out}} = \frac{(b+c-a)(c+a-b)(a+b-c)}{abc} \quad (3)$$

其中, a, b, c 是三角形的边长。对于正三角形来说, $q = 1$; 对于退化三角形, $q = 0$ 。如果所有的三角形

网格都有 $q > 0.5$, 可认为是高质量的。

对于四面体网格^[13], 一般使用边长比 t 来评定, 定义为单元最长边与最短高之间的比值。对于正四面体, 其边长比 $t = \frac{\sqrt{6}}{3}$ 。

3.2 二维平面改进结果

(1) 单位圆和带孔正五边形的优化过程如图 3 和图 4 所示, 从左至右分别是迭代 1 次、10 次、100 次的效果。从图中可以看出, 边界处的三角形网格变化尤为明显, 边界也趋于清晰理想化, 最终整体网格质量很高, q 都在 0.99 以上。

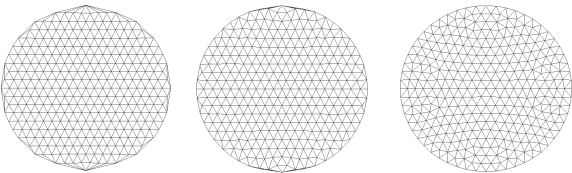


图 3 单位圆网格优化过程

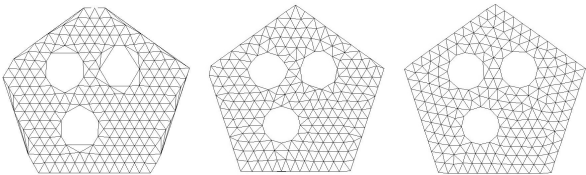


图 4 带孔正五边形网格优化过程

3.3 三维实体改进结果

图 5 为球体的四面体网格优化过程的剖面图, 由形状千奇百怪最后趋于均匀的正四面体。

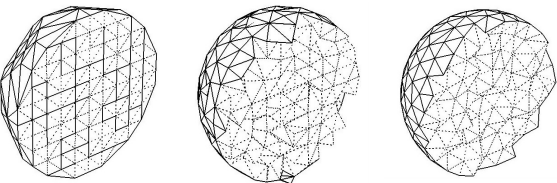


图 5 球体四面体网格优化过程

3.4 结果分析

以上所有二维的例子 q 的平均值超过 0.98, 显著好于拉普拉斯平滑的 Delaunay 细化网格^[14]。和原来的 DistMesh 进行对比, 如表 1 所示, 其网格质量仍然保持在高水平。

表 1 改进算法与 DistMesh 网格质量对比

| 图 形 | DistMesh | 改进算法 |
|--------------|----------|------|
| 单位圆的 q 均值 | 0.99 | 0.99 |
| 带孔五边形 q 均值 | 0.98 | 0.98 |
| 球体 t 均值 | 0.78 | 0.80 |

4 结束语

文中介绍了基于 MATLAB 的网格自动生成器 DistMesh 的主要原理, 由于其核心的距离函数是解析

表达的, 对于复杂图形很难构造, 故基于 MATLAB 中的 NUBRS 工具箱, 将 NURBS 建模嵌入到算法中, 解决了 DistMesh 不能实现复杂模型构造的缺陷。在二维平面的改进中, 利用点到边界曲线上最近点处的向量, 与最近点处切线方向向量的向量积, 来定义曲线方向、判断点与曲线的位置关系, 以筛选节点; 在三维实体的改进中, 利用点到边界曲面上最近点处的向量, 与最近点处法线方向向量的数量积, 来定义曲面的法线方向、判断点与曲线的位置关系, 以筛选节点; 基于力的平衡原理的迭代循环, 通过对比理想的与实际的网格长度来不断调整网格边长, 完成对网格的优化。验证结果表明, 改进算法在保证网格生成质量的前提下, 提高了 DistMesh 的适用性。

参考文献:

[1] 程中娜. 基于 ABAQUS 的智能有限元分析前处理技术研究[D]. 济南: 山东大学, 2017.

[2] 蒲军平, 姚宇龙. 平面有限元网格实用划分方法研究[J]. 浙江工业大学学报, 2012, 40(1): 88-91.

[3] 冯桂莲. 偏微分方程的 MATLAB 数值解法及可视化[J]. 计算机技术与发展, 2013, 23(12): 120-123.

[4] PERSSON P O, STRANG G. A simple mesh generator in Matlab[J]. SIAM Review, 2004, 46(2): 329-345.

[5] 施法中. 计算机辅助几何设计与非均匀有理 B 样条[M]. 第 2 版. 北京: 高等教育出版社, 2013.

[6] PIEGL L A, TILLER W. The nurbs book[M]//Monographs in visual communication. [s. l.]: [s. n.], 2007: 133-158.

[7] ETIENE T, NONATO L G, SCHEIDEGGER C, et al. Topology verification for isosurface extraction[J]. IEEE Transactions on Visualization & Computer Graphics, 2012, 18(6): 952-965.

[8] 周奇才, 吴青龙, 熊肖磊, 等. 桁架结构拓扑及截面尺寸优化设计方法[J]. 西安交通大学学报, 2016, 50(9): 1-9.

[9] SHEWCHUK J R. Delaunay refinement algorithms for triangular mesh generation[J]. Computational Geometry Theory & Applications, 2014, 47(7): 741-778.

[10] 李海峰, 吴冀川, 刘建波, 等. 有限元网格剖分与网格质量判定指标[J]. 中国机械工程, 2012, 23(3): 368-377.

[11] 范宜仁, 吴 飞, 邢东辉, 等. 自适应方法: CN 104297278A [P]. 2015.

[12] 秦茂玲. 三角网格模型的分割、评价及其应用研究[D]. 济南: 山东师范大学, 2013.

[13] 王 磊, 聂玉峰, 李义强. Delaunay 四面体网格并行生成算法研究进展[J]. 计算机辅助设计与图形学学报, 2011, 23(6): 923-932.

[14] 王 卉, 黄玉清. 基于拉普拉斯算子的三角网格模型的平滑与压缩算法[J]. 计算机系统应用, 2015, 24(12): 191-195.