

文章编号:1000 - 2308(1999)04 - 0048 - 05

有限元平面三角形网格自动划分^{*}

张均锋¹,刘桂斋²,陈刚³

(1. 中国科学院力学研究所; 2. 山东科技大学基础部; 3. 武汉大学软件基地)

摘 要:提出了一种采用单元尺寸函数进行有限元平面三角形网格自动划分的方法,根据给定的几何或物理条件,计算待划分区域上每点的单元尺寸,并以此将整个区域划分为单元尺寸各不相同的若干个子域,在各个子域内部生成均匀网格。

关键词:有限元;网格自动划分;单元尺寸函数;子域;均匀网格

中图分类号: O 157.5 **文献标识码:** A

采用有限元法解决工程中的各种场问题时,首先必须进行“结构离散化”工作,产生有限元计算程序的输入数据,然后才能做进一步的计算,这一步工作更普遍地称为“网格划分”。早在 70 年代初就有人做过统计后指出“在进行有限元分析时,有 50% 或更多的时间和资金是花在输入数据的准备和检查上……”^[1],因此网格划分由于其繁琐和容易出错而成为有限元分析过程中的“瓶颈”。随着计算机技术的快速发展,在大容量、高速度的微机、工作站、小型机上解决大型工程问题已成为可能,各种用于网格划分的算法和程序也越来越多,使得该领域逐渐活跃起来^[2,3,4]。

本文提出了一种新的网格划分方法,它是由边界和内部控制点共同决定待划分区域内不同部分的单元尺寸(从而决定了网格的疏密变化),即先定义单元尺寸函数 $D(x, y)$, 然后根据 D 的等值线将整个区域分成单元尺寸各不相同的若干子域,在各个子域内生成均匀的三角形网格。

1 网格生成方法

网格密度控制涉及到网格单元形状和计算精度问题,在有限元计算中,形状最好的网格是等边三角形网格^[5],但实际上是没有必要生成正三角形网格。因为:(1)进行有限元分析的区域并非每个部分都要求有相同的精度;(2)由于单元尺寸由最小精度来决定,若所有单元都采用最小的单元尺寸,就会增加单元数目从而加大工作量、降低速度。因此实际采用的网格都按如下原则:在总体上接近等边三角形网格,而在局部有网格密度的变化从而形成有梯度的网格。

1.1 单元尺寸函数

单元尺寸函数是表征平面上某点附近单元的平均边长的函数,它可以显式地给出,也可以隐含地表示。如设 $D = D(x, y)$ 是定义在待划分区域上的单元尺寸函数,并设它连续,那么在区域内的每一点 (x, y) , 都有唯一的 D 值与之对应。精度较高的尺寸函数可根据物理的或几何的要求来定义,凡是计算精度要求高的区域,单元尺寸函数的值就较小。我们所采用的单元尺寸函数是通过边界线段的尺寸获得内部控制点信息来构造的,在其它点上通过插值来获得 D 值。

* 收稿日期:1999 - 04 - 29

作者简介:张均锋(1966 -),男,山东青岛人,博士,从事动载荷作用下饱和砂土的液化与密实的研究工作。北京 100080。

边界线段的长度反映了该段边界附近单元的疏密程度,边界线段较短的,附近单元就较密;反之,边界线段长的,附近单元就相对较疏。根据单元尺寸函数的意义,定义任意边界节点的单元尺寸函数值 D 等于共享该节点的两条边界线段长度的均值,如图 1。

为了获得区域内任一点的 D 值,需要在边界节点和内部控制节点间形成网格。由于整个区域内的单元尺寸函数 $D = D(x, y)$ 是由一系列曲面片拟合的,因此这些曲面片在 $X - Y$ 平面上的投影就形成了函数 $D = D(x, y)$ 的投影网络,得到的网格节点集就是由边界节点和内部控制节点组成的集合。对区域内任意一点 (x, y) ,首先考察它落在投影网络的哪个单元内,然后通过该单元对应的曲面方程求得该点的 D 值。

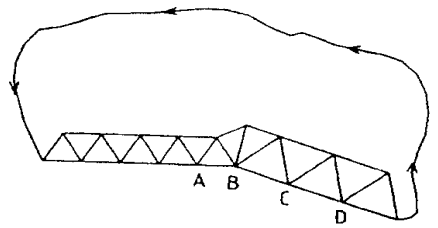


图 1 B 点的单元尺寸函数值为

$$D(B) = D(x_B, y_B) = \frac{1}{2} (|AB| + |BC|)$$

1.2 子域的划分

前面定义单元尺寸函数和形成投影网络都是为划分子域作准备。划分子域主要包括两方面:(1)整个区域分成几级子域;(2)怎样划分子域。前面曾提到,正三角形网格被认为是形状最好的三角形网格,如果一个网格中出现了钝角三角形单元,那么它很可能被一些有限元程序拒绝接受。由于在子域划分完成后,将在各子域内部生成均匀网格,故单元形状的变化将出现在相邻子域的过渡地带,这样就必然要求相邻子域的单元尺寸之间满足一定的条件,相差太大会使过渡地带的单元尺寸发生跳变从而影响网格质量。对等腰直角三角形,其斜边与直角边之比为 $\sqrt{2}$,因此一般认为相邻子域的单元心寸之比 r 的范围为 $1/\sqrt{2} < r < \sqrt{2}$ 时,过渡地带的单元形状可以保持得很好。

设单元尺寸函数 $D = D(x, y)$ 在整个待划分区域内的最大值和最小值分别为 D_{\max} 和 D_{\min} ,相邻子域的单元尺寸比为 $r (r > 1)$,整个区域划分为 N 级子域, N 由下式决定:

$$N = \lceil \ln(D_{\max}/D_{\min})/\ln r \rceil \tag{1}$$

第 i 级子域的单元尺寸为:

$$D_i = D_{\min} \times r^{i-1} \tag{2}$$

在投影网格的每个单元中,依次作 $D(x, y) = D_i$ 的等值线,将所有这些等值线首尾相接(或与边界相接),就将整个区域分割成了许多子域,它们要么由单一的封闭等值线围成(常出现在内部控制点的情况),要么位于几条等值线之间,如图 2。

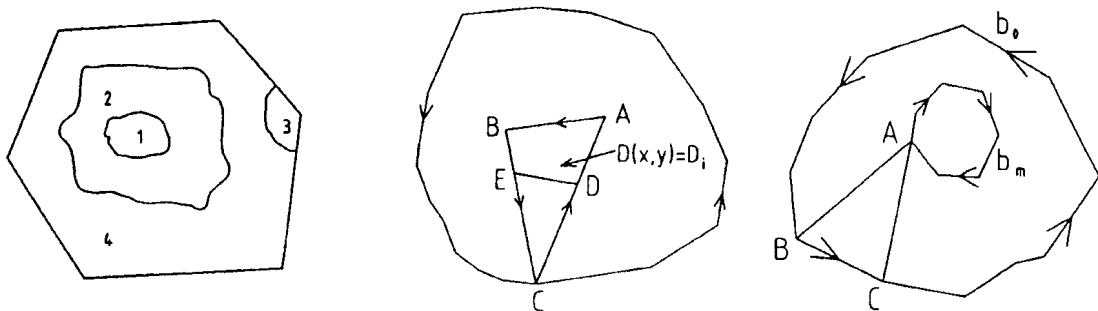


图 2 作出等值线后由四个子域产生 图 3 确定 DE 等值线段的走向 图 4 说明 b_0 必包围 b_m 图

在投影网格单元内作单元尺寸函数的等值线时,需确定等值线的方向,图 3 中等值线的方向是这样确定的:若 $D(c) < D_i$,则等值线的方向是 $D \rightarrow E$,否则等值线方向为 $E \rightarrow D$ 。对任一等值线,将落在投影网格中的所有单元中的等值线段连接起来,就得到第 i 条封闭的环。这些环的走向和相互位置决定了子域的个数(不一定是 N 个)及其定义。

在计算机上实现以上过程实际上是一个循环,由单元尺寸较小的子域开始,一个个地提取各子域的定义,直到把所有边界环用完。在此过程中还需判断一条边界环是否包围另一条边界环。由于各条边界环是互不相交的,故该问题就转换为判断一个点是否落在边界环内部或外部。如图 4 所示,若边界

环 b_m 上一点 A 落在边界环 b_0 内部,那么 b_0 必包围 b_m ,这里可用两种方法进行判断(假设 A 不在边界环 b_0 上):

(1)在边界环 b_0 上取一条边界段 BC ,使 AC 和 CB 不与其它任何边界段相交,计算 BCA 的有向面积 S ,若 $S > 0$,则 A 落在边界环 b_0 内部;反之,若 $S < 0$,则 A 落在边界环 b_0 外部;若 $S = 0$,表示 A 落在边界环 b_0 上。 S 可表示为:

$$S = \frac{1}{2} \begin{vmatrix} x_B & y_B & 1 \\ x_C & y_C & 1 \\ x_A & y_A & 1 \end{vmatrix} \tag{3}$$

(2)过 A 作水平线 $y = y_A$,直线与边界环若相交,将 A 点和所有交点按 x 坐标值递增的顺序排列,序号从 1 开始,则 若 A 的序号是偶数, A 落在边界环的内部; 若 A 的序号是奇数,则 A 落边界环的外部。若过 A 的直线 $y = y_A$ 与边界环无交点,则 A 在边界环外部。

这样就将所有子域的定义全部抽取了出来,每个子域由若干条边界环定义,外边界环只能有一条(沿逆时针方向),内边界环可以有多条(沿顺时针方向),至少有一条内边界环的子域是多连通区域,其它为单连通区域。

1.3 均匀网格的自动生成

所谓的均匀网格是特指等边三角形,在一个子域内部,由于单元尺寸函数值变化不大,可以在该子域内自动生成均匀网格。这里我们采用了“附加网格法”^[6,7],即将事先生成的一张正三角形网格覆盖在待划分区域上来确定节点位置,网格单元有些是与区域边界相交的,因此这些节点位置需进行调整。

本文中所采用的附加三角形网格是令该网格的某个节点与待划分区域的最高点(y 值最大的边界节点)重合,如图 5 所示,均匀网格的生成分两步:(1)生成内部节点;(2)连接这些节点形成网格。

设单元尺寸为 D ,假想有一系列等距(距离 $d = \sqrt{3} D/2$)的水平直线切割区域,那么这些直线与边界环相交,将产生若干落在区域内的直线段,所有内部节点将在这些水平直线段上产生,具体过程如下:

找出最大 y 值的边界节点 (x_0, y_{max}) ,同时得 y_{min} ;

第 i 条水平直线的方程为:

$$y = h_i = y_{max} - i \times d \quad i = 1, 2, \dots, [(y_{max} - y_{min}) / d] \tag{4}$$

确定直线 $y = h_i$ 与区域的交集,将所有交点按 x 递增的顺序排列,则交点 1 2,3 4, ..., (2n - 1) 2n 之间的线段一定落在区域内部;

从点 (x_0, y_{max}) 出发作一条斜率 $k = \text{tg}60^\circ$ 的直线与 $y = h_i$ 相交,设交点为 $S(x_s, y_s)$,则

$$x_s = x_0 + (h_i - y_{max}) / \sqrt{3}, \quad y_s = h_i \tag{5}$$

这样可算出所有落在区域内的线段上距离 S 为 D 的整数倍的节点(见图 5),这些节点是产生内部节点的候选点,在所有的候选点中,那些与已存在的节点(边界节点、指定内部不动点和新产生的节点)距离太近的将被舍弃,余下的为最终产生的内部节点。由于所有内部节点都被限定在某条水平直线上,它们之间不会出现距离太近的情况,因此只需考察一个候选点是否过于靠近边界或指定内部不动点。

设 $C(x_c, y_c)$ 是候选节点, $B(x_b, y_b)$ 是边界节点(或指定节点),若

$$\sqrt{(x_c - x_b)^2 - (y_c - y_b)^2} < \quad (\text{可取 } 0.6D) \tag{6}$$

则认为 C 过于靠近 B 而被舍弃。将每条直线上的候选点处理完后,就得到由边界节点和内部节点组成的全部节点的集合。

连接节点形成网格是从边界开始。对任一有向边界 $n_b n_d$,在其左边找最佳节点 n_c ,构成一个单元 $n_b n_d n_c$,如图 6 所示,线段 A 是 $n_b n_d$

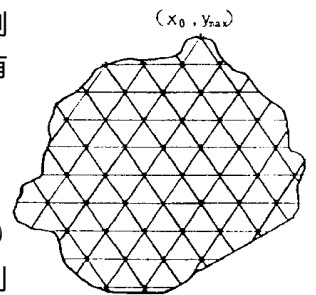


图 5 均匀网格生成示意图

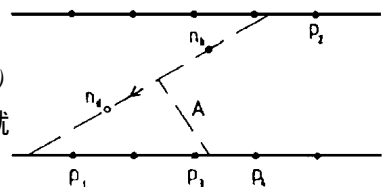


图 6 单元 $n_b n_d n_c$ 的构成图

的垂直平分线,最佳节点 n_c 将在 $p_1、p_2、p_3、p_4$ 中产生。这取决于四个三角形 $n_b n_d n_{p_i} (i = 1, 2, 3, 4)$ 中哪个的 (三角形中最大内角与最小内角之差) 最大,单元 $n_b n_d n_c$ 产生后要在原边界中除去边界段 $n_b n_d$, 而把 $n_c n_d$ 和 $n_b n_c$ 作为新的边界加入边界集, 若 $n_c n_d$ 和 $n_b n_c$ 已在边界集中, 则将其除去, 直到所有边界段用完 (即边界集为空), 网格便自动生成完毕。

2 实例

下面用图例说明采用上述方法自动生成网格的主要步骤。图 7 - a 是原始边界图, 整个区域由 5 条边界环定义, 外边界环沿逆时针走向而内边界环沿顺时针走, 各边界段的长度在网格划分前就已确定。图 7 - b 是利用单元尺寸函数的等值线划分的子域。7 - c 是分别在各子域内生成的均匀网格。

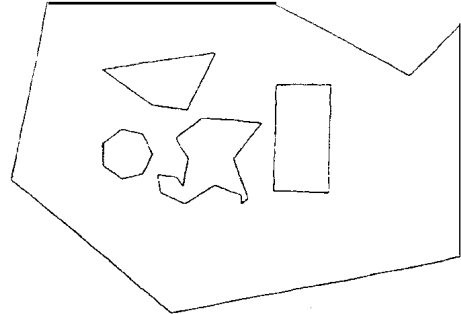


图 7 - a 原始边界由五条边界环构成

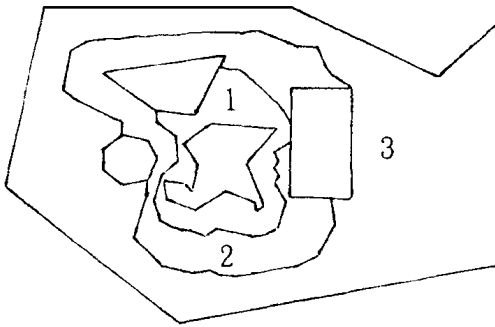


图 7 - b 划分子域之后, 整个区域由三个多连通子域组成, 单元尺寸随子域编号递增

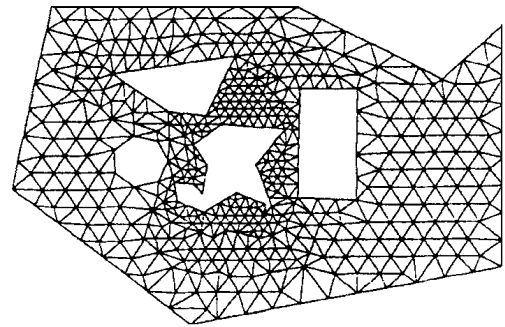


图 7 - c 各子域内部分别生成均匀网格, 形状较差的单元出现在子域交界处

3 结语

利用单元尺寸函数来控制网格密度变化, 而网格的形状取决于单元尺寸函数的定义, 在各子域内部, 由于内部节点的均匀分布, 可以快速地产生产高质量的网格, 因而网格的总体质量取决于边界节点和内部控制节点。在本文的方法中, 边界段的长度变化要合理, 相邻边界段的长度变化不能太大, 否则将影响生成网格的质量, 这应在输入数据的准备中, 对边界段长度作严格的限制, 如何由几何和物理条件决定边界段长度是与网格划分密切相关的问题。

此外, 本文的方法也可进一步推广, 在已生成的二维网格上产生第三维信息 (如水深、温度、海拔高度等)。

参考文献:

- [1] Yate, D. N., et al., The Development of Large Scale Digital Computer Codes for Production Structures Analysis [R]. Lockheed Structures Conference, 1970.
- [2] Zienkiewicz, O. C. and Phillips, D. U., An Automatic Mesh Generation Scheme for Plane and Curves Surfaces by Iso-parametric Co-ordinate[J]. International Journal of Numerical Methods Engineering, Vol. 3, 1970:519 - 528.
- [3] Gerdon, W. J. and Hal, C. A., Construction of Curvilinear Co-ordinate Systems and Applications to Mesh Generation [J]. International Journal of Numerical Methods Engineering, Vol. 7, 1973:461 - 477.
- [4] Cavendish J. C., Automatic Triangulation of Arbitrary Planar Domains for the Finite Element Method[J]. International Journal of Numerical Methods Engineering, Vol. 8, 1974:679 - 696.
- [5] 丁浩江等. 弹性和塑性力学中的有限单元法[M], 北京:机械工业出版社, 1989.

- [6] Shaw ,R. D. and Pitchen ,R. G. , Modification to the Suhara Fukuda Method of Network Generation[J]. International Journal of Numerical Methods Engineering , Vol. 12 , 1978 :93 - 99.
- [7] Lo ,S. H. , A New Mesh Generation Scheme for Arbitrary Planar Domains[J]. International Journal of Numerical Methods Engineering , Vol. 21 ,1985 : 1403 - 1426.

Automatic Generation of Finite Element Planar Triangular Meshes

ZHANG Jun-feng¹ , LIU Gui-zhai² , CHEN Gang³

(1. Inst. of Mech. of CAS; 2. Dept. of Basic Courses , SUST; 3. Software BAse of Wuhan Univ.)

Abstract : A new method of automatically generating finite element planar triangular meshes is presented on the basis of using element-size function. By means of geometrical or physical criteria , the element-size of each point in the region is calculated , the whole region is subdivided into several sub-regions with different element-size , and then uniformed meshes are generated in each sub-region.

Key words : finite element ; automatic mesh generation ; element-size function ; sub-region ; uniformed mesh

(上接 45 页)

参考文献:

- [1] Bondy 和 Muty. 图论及其应用[M]. (中译本)习题 2.5.2
- [2] 魏暹赫. 图论基础[M]. 西安:陕西师范大学出版社,1991
- [3] 陈东灵. 网络最优化及其应用[Z]. 泰安:山东矿业学院,1987

One Algorithm with Constrained Minimum Generating Subgraph

HAN Cong-ying¹ , NING Wei²

(1. Dept. of Appl. Math and Software Eng. , SUST; 2. Dept. of Mathematics , Taian Teachers' College)

Abstract : In this paper , with regard to a kind of practical connection problem we establish a network model and present an algorithm which provides access to the computer implementation ; that is , the network $G = (V, E, W)$ is simplified by contracting the sides in turn , we eventually obtain the minimum generating tree ; then we recover each edge and vertex and we obtain the optimal solution.

Key words : Connection problem ; graph ; generating subgraph ; network algorithm