

有限元数据前处理的探讨

段 云 岭

(水利系)

提 要: 本文将有限元计算域划分为若干个子域,对于每个子域进行计算网络的自动剖分,探讨了二维计算域的三角形线性元和二次元、四边形线性元和二次元的网格自动生成技术,并利用计算机的图形显示技术动态地显示出计算域的分区情况与网格图形。文中方法概念清晰,原理简单,效率较高。

关键词: 计算域,子域,计算网格,网格的自动生成,线性元,二次元,图形显示

1 前 言

在进行有限元结构应力分析时,要对计算域进行离散化,以一定的网格形式对计算域进行剖分,离散出各单元的结点坐标、单元结点码等信息,供有限元分析用。通常对于较规则区域,可利用自动剖分或半自动剖分程序给出结点有关信息,然后人工绘出网格图以校核。而对于较复杂的区域,常常要靠人工进行离散,然后输入到计算机中,工作量很大,出错的几率也大,不易保证离散网格的正确性。文献[1]—[4]给出的几种自动剖分方法,有的输入信息多,有的变换复杂,有的则要经过多次尝试才能定出一个结点,势必耗费机时过多,效率不高。一个好的自动剖分程序应具备二点,一是用户输入的信息尽可能少,以减少出错率;二是应能方便直观地检验生成的网格的正确性与合理性。本文对二维计算域的自动剖分进行了探讨。将自动剖分技术与计算机图形显示技术结合起来,实现了网格的动态显示校核。对于任一个计算域,只要事先确定出子域的分割即可。利用搭积木的原理,逐域进行剖分,上一子域的剖分结果也为下一子域的剖分提供了初始参数。大大地提高了有限元数据前处理的准确性及处理效率,降低了劳动强度。

2 方法概述

对于一个比较复杂的二维计算域,可以根据材料分区、边界情况以及应力集中等情况分成若干个较规则的任意四边形或曲边四边形子区域。每个子域上应力梯度变化不大,其上的网格可按均匀地变化,因而可对每个子域进行自动剖分。

本文1988年4月28日收到。

网格自动剖分, 关键要解决三个问题, 一是单元边界的自动分割, 二是结点的整体编码, 三是子域间单元、结点连续协调问题。

单元边界的分割采用平面几何的方法。对子域的每条边界进行等长度分割, 二组相互交叉的分割线构成了剖分网格的骨架。结点编码采用逐点累加的方法。为了减少总体刚度矩阵的带宽, 一般按计算域短边方向统一编码, 子域内的结点号可能是不连续的。子域间单元协调、结点编码连续问题由子域公共边相同分割, 结点编码在计算域中统一进行得以保证。

为了减少出错机会, 提高处理速度, 尽量减少输入信息。子域边界的输入信息, 对于任意四边形为四个角点的坐标; 对于曲边四边形子域应输入曲线方程的参数。子域的剖分信息为沿 x 、 y 向的剖分单元个数、第一个单元的单元编号和起始边上的结点整体码以及左、右相邻子域 x 向单元变化情况。

3 四边形单元的剖分

四边形线性元和二次元的局部码如图1所示。

对于每一个单元, 先按局部码计算出各节点的坐标, 然后再由局一整码转换而成整体码下的结点坐标。

对于图2子域, 已知角点 A 、 B 、 C 、 D 的坐标; 单元的层数 N_y , 即沿 y 向的单元个数, 以及每层单元个数 N_x , 即沿 x 向单元个数; 第1个单元的单元编号 J_E ; J_E 单元的起始边上结点整体码 N_1 , N_2 。

3.1 单元结点坐标: 第 i 层单元的四个角点坐标为:

$$\begin{aligned} x_a &= X_A + \frac{x_C - x_A}{N_y} (i - 1), \\ x_b &= X_B + \frac{x_D - x_B}{N_y} (i - 1), \\ x_c &= X_A + \frac{x_C - x_A}{N_y} i, \\ x_d &= X_B + \frac{x_D - x_B}{N_y} i, \\ (i &= 1, 2, \dots, N_y) \end{aligned} \quad (1)$$

类似地可得出 y_a, y_b, \dots, y_d 。

第 i 层单元中第 j 个单元的四个角点坐标为:

$$x_1 = x_a + \frac{x_b - x_a}{N_x} \cdot (j - 1)$$

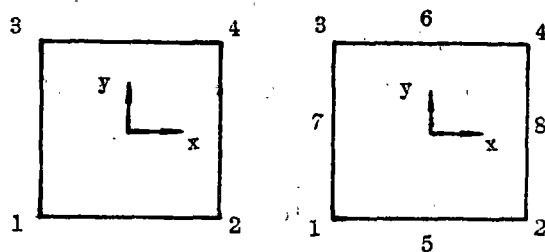


图1 四边形单元局部码

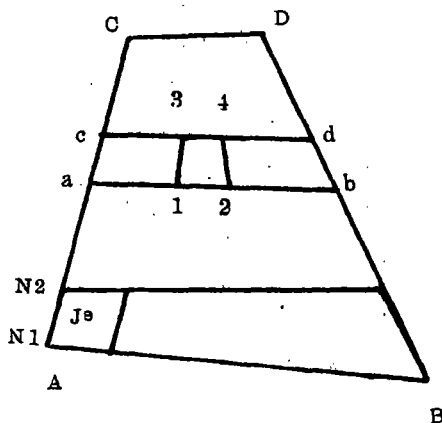


图2 四边形线性元子域

$$\begin{aligned}
 x_2 &= x_a + \frac{x_b - x_a}{N_x} \cdot j, \\
 x_3 &= x_c + \frac{x_d - x_c}{N_x} (j - 1), \\
 x_4 &= x_c + \frac{x_d - x_c}{N_x} \cdot j. \quad (j = 1, 2, \dots, N_x)
 \end{aligned} \quad (2)$$

同理可求得 y_1, y_2, y_3, y_4 的值。

对于二次单元, 不难利用插值法求得中间结点5, 6, 7, 8的坐标。

3.2 单元结点编号: 若令第 i 层单元中第 j 个单元的单元编号为 jk , 则:

$$\begin{aligned}
 jk &= jE + (i - 1) \cdot N_x + j - 1 \\
 (i &= 1, 2, \dots, N_y, \quad j = 1, 2, \dots, N_x)
 \end{aligned} \quad (3)$$

用一个二维数组 $M(i, j)$ 记录单元中诸结点局部码与整体码的关系。其中 i 为单元编号, j 为单元结点的局部码, 其元素 $M(i, j)$ 的值代表第 i 个单元中第 j 个结点对应的整体码。

3.2.1 四结点线性元: 已知第1层单元的起始边结点编号 $N_1^{(1)}, N_2^{(1)}$, 则第 i 层单元的起始边结点编号为:

$$\begin{aligned}
 N_1^{(i)} &= N_2^{(i-1)} = N_2^{(1)} + (i - 2) \cdot (N_2^{(1)} - N_1^{(1)}), \\
 N_2^{(i)} &= N_1^{(i)} + N_2^{(1)} - N_1^{(1)}. \quad (i = 2, 3, \dots, N_y)
 \end{aligned} \quad (4)$$

第 i 层第 j 个单元也就是计算域中第 jk 个单元的局一整码转换数组为:

$$\begin{aligned}
 M(jk, 1) &= N_1^{(i)} + j - 1, \\
 M(jk, 2) &= N_1^{(i)} + j, \\
 M(jk, 3) &= N_2^{(i)} + j - 1, \\
 M(jk, 4) &= N_2^{(i)} + j, \quad (j = 1, 2, \dots, N_x)
 \end{aligned} \quad (5)$$

3.2.2 八结点二次元, 已知第1层单元的起始边结点编号为 $N_1^{(1)}, N_2^{(1)}, N_3^{(1)}$, 则第 i 层单元的起始边结点编号为:

$$\begin{aligned}
 N_1^{(i)} &= N_3^{(i-1)} = N_3^{(1)} + (i - 2) \cdot (N_3^{(1)} - N_1^{(1)}) \\
 N_2^{(i)} &= N_1^{(i)} + N_2^{(1)} - N_1^{(1)} \\
 N_3^{(i)} &= N_2^{(i)} + N_3^{(1)} - N_2^{(1)} \quad (i = 2, 3, \dots, N_y)
 \end{aligned} \quad (6)$$

第 jk 单元的转换数组为:

$$\begin{aligned}
 M(jk, 1) &= N_1^{(i)} + 2j - 2 \\
 M(jk, 2) &= N_1^{(i)} + 2j \\
 M(jk, 3) &= N_3^{(i)} + 2j - 2 \\
 M(jk, 4) &= N_3^{(i)} + 2j \\
 M(jk, 5) &= N_1^{(i)} + 2j - 1 \\
 M(jk, 6) &= N_3^{(i)} + 2j - 1 \\
 M(jk, 7) &= N_2^{(i)} + j - 1 \\
 M(jk, 8) &= N_2^{(i)} + j \quad (j = 1, 2, \dots, N_x)
 \end{aligned} \quad (7)$$

4 三角形单元的剖分

三角形单元可以方便地进行网格疏密过渡。此时每层单元的单元个数呈递增或递减变化。为保证单元具有较好的形态,一般每层单元的变化个数不宜超过二个,即每层单元的始、末端以各变化一个单元为宜。用变量 JE_1 , JE_2 分别表示子域始端和末端沿层数增加方向单元变化情况。其取值为: ± 1 , 0 。

左右相邻子域每层单元的个数变化也将影响子域的单元结点编码。用 JE_L , JE_R 分别记录左、右子域每层单元变化情况。不难看出,它们的取值范围为 0 , ± 1 , ± 2 。

需要注意的是,此时每层单元的个数是在变化的。为方便起见,令 N_x 表示第一层单元的起始线 AB 上的等分个数。如图3所示, $N_x = 6$, 而不是11。第 i 层单元起始线上沿 x 向分割个数为:

$$N_x^{(i)} = N_x + (i - 1)(JE_1 + JE_2) \quad (i = 2, 3, \dots, N_y) \quad (8)$$

4.1 单元结点坐标:

4.1.1 JE_1 , JE_2 均为0: 利用式(1)、(2)可得出第 i 层单元第 j 个四边形的四个角点坐标, 再将这个四边形分成二个三角形即可。

为了保证三角形单元三个内角均不大于 90° , 先计算一下四边形的角 $\angle 312$ 。若 $\angle 312 < 90^\circ$, 则结点2, 3连线形成二个三角元; 反之, 结点1, 4连线。

单元局部码顺序按逆时针向旋, 以满足某些程序中单元面积为正的要求。如图4所示。

根据图4及四个角结点1、2、3、4的坐标, 不难求得两个三角形的结点坐标。

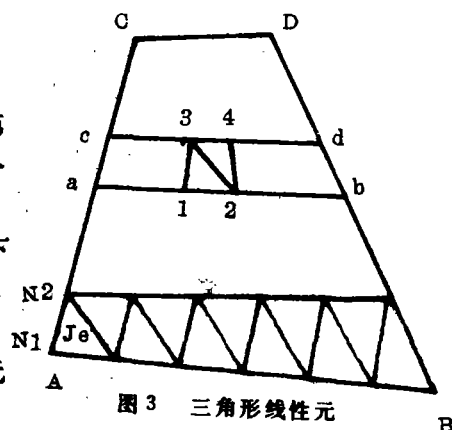
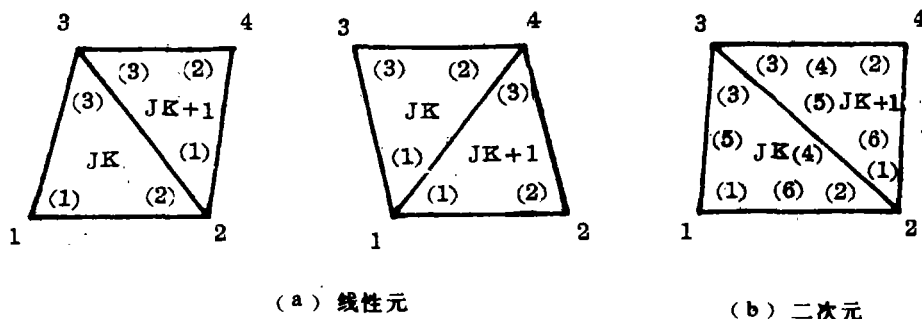


图3 三角形线性元



(a) 线性元

(b) 二次元

图4 三角元局部码顺序

对于二次元, 利用插值的方法可求得中间结点④、⑤、⑥的结点坐标。

4.1.2 JE_1 , JE_2 均不为0 或其一不为0: 先根据其值在第 i 层始端或末端或两端均分

出一个三角元, 然后重复过程1. 即可。

4.2 结点编码: 第 i 层单元第 j 个四边形中第一个三角元的单元号为:

$$jk = JE + \sum_{k=1}^{i-1} (2N_x^k + JE1 + JE2) + |JE1| + 2j - 2, \quad \begin{pmatrix} i = 1, 2, \dots, N_y \\ j = 1, 2, \dots, N_x - |JE1| - |JE2| \end{pmatrix} \quad (9)$$

第二个三角元为 $(jk + 1)$ 。

4.2.1 线性元: 已知第 1 层单元的起始边结点编号 $N_1^{(1)}, N_2^{(1)}$ 。第 i 层单元的起始边结点编号为:

$$\begin{aligned} N_1^{(i)} &= N_2^{(i-1)} \\ N_2^{(i)} &= N_1^{(i)} + N_2^{(1)} - N_1^{(1)} + (i-1)(JE1 + JE2 + JE\ell + JER) \\ &\quad (i = 2, 3, \dots, N_y) \end{aligned} \quad (10)$$

当 $\angle 312 < 90^\circ$ 时, 由图4 (a) 可得:

$$\begin{aligned} M(jk, 1) &= N_1^{(i)} + \text{int} \left(\frac{1 - JE1}{2} \right) + j - 1 \\ M(jk, 2) &= M(jk, 1) + 1 \\ M(jk, 3) &= N_2^{(i)} + \text{int} \left(\frac{1 + JE1}{2} \right) + j - 1 \end{aligned} \quad (11)$$

类似地可得出第 $(jk + 1)$ 单元的转换数组。

4.2.2 二次元: 已知第 1 层单元的起始边结点编号为 $N_1^{(1)}, N_2^{(1)}, N_3^{(1)}$ 。则第 i 层单元的起始边结点编号为:

$$\begin{aligned} N_1^{(i)} &= N_3^{(i-1)} \\ N_2^{(i)} &= N_1^{(i)} + N_2^{(1)} - N_1^{(1)} + (i-1)(JE1 + JE2 + JE\ell + JER) \\ N_3^{(i)} &= N_2^{(i)} + N_3^{(1)} - N_2^{(1)} + (i-1)(JE1 + JE2 + JE\ell + JER) \\ &\quad (i = 2, 3, \dots, N_y) \end{aligned} \quad (12)$$

当 $\angle 312 < 90^\circ$ 时, 如图4 (b) 所示第 jk 个三角元的转换数组为:

$$\begin{aligned} M(jk, 1) &= N_1^{(i)} + 2 \left[\text{int} \left(\frac{1 - JE1}{2} \right) + j - 1 \right] \\ M(jk, 2) &= M(jk, 1) + 2 \\ M(jk, 3) &= N_3^{(i)} + 2 \left[\text{int} \left(\frac{1 + JE1}{2} \right) + j - 1 \right] \\ M(jk, 5) &= N_2^{(i)} + |JE1| + 2j - 2 \\ M(jk, 4) &= M(jk, 5) + 1 \\ M(jk, 6) &= M(jk, 1) + 1 \\ &\quad (j = 1, 2, \dots, N_x - |JE1| - |JE2|) \end{aligned} \quad (13)$$

类似地可以得出第 $(jk + 1)$ 单元的转换数组。

5 三角元与四边形元的混合剖分

四边形单元不易进行每层单元的增减, 即不易实现单元过渡。而三角形单元可以较方便

进行单元过渡。将三角形与四边形元进行混合剖分,不仅可以利用较高精度的四边形元,也可利用易于过渡的三角元。利用三中的方法,不难实现这种混合剖分,但对于第*i*层单元中第*j*个四边形块不再划分为三角元。结点编码只要对式(11)和式(13)稍加改变即可。

6 坐标变换

有限元中总体刚度矩阵的带宽与单元中结点码的差值有关。差值愈大,其带宽愈大。较大的有限元分析程序通常都有结点编号优化处理。一般的有限元程序没有这种处理。为了减少带宽,尽量扩大计算能力,对某些特殊的子域如狭长的子域,宜以短边方向结点编码连续递增。只要将子域的短边方向作为*x*轴向,既可利用上述诸方法进行剖分。但最后要进行局部坐标系下的结点坐标向整体坐标系转化。

设子域的局部坐标系为*x'o'y'*,整体坐标系为*xoy*。局部坐标系原点*o'*在整体坐标系中的坐标为(*x_o*, *y_o*)。局部坐标系沿逆时针旋转 α 度后与整体坐标方向重合,则任一点*P*(*x'*, *y'*)与*P*(*x*, *y*)的转换为:

$$\begin{cases} x = x_o + x' \cdot \cos \alpha + y' \cdot \sin \alpha \\ y = y_o + y' \cdot \cos \alpha - x' \cdot \sin \alpha \end{cases} \quad (14)$$

7 曲线边界的剖分

常用的曲线边界为椭圆类二次曲线。如压力容器的封头等。下面统一按椭圆型曲线处理。

设内边界为曲线Ⅰ,外边界为曲线Ⅱ。将子域的坐标原点放在曲线Ⅰ的园心上,方向与Ⅰ的椭圆轴方向重合,则两边界曲线方程为:

$$\begin{cases} x^I = a^I \cdot \cos \theta \\ y^I = b^I \cdot \sin \theta & (\theta_0^I \leq \theta \leq \theta_1^I) \\ x^{II} = x_o + a^{II} \cdot \cos \theta \\ y^{II} = y_o + b^{II} \cdot \sin \theta & (\theta_0^{II} \leq \theta \leq \theta_1^{II}) \end{cases} \quad (15)$$

式中 θ_0^I 、 θ_1^I 分别表示曲线*i*的起、止离心角;(*x_o*, *y_o*)为曲线Ⅱ的椭圆心坐标。

曲线上某点(*x*, *y*)对应的离心角是与该点*x*相同的、直径为*a*,或者与该点*y*相同的、直径为*b*的园上的点与椭圆心连线和*x*轴所成的夹角。这在输入起、止离心角时要特别注意。

设沿径向单元个数为*N_r*,沿环向的单元层数为*N_θ*,则在曲线Ⅰ、Ⅱ之间可以建立若干条曲线,其方程为:

$$\begin{cases} x(j, \theta) = j \cdot \frac{x_o}{N_r} + \left[a^I + j \frac{a^{II} - a^I}{N_r} \right] \cdot \cos \theta \\ y(j, \theta) = j \cdot \frac{y_o}{N_r} + \left[b^I + j \frac{b^{II} - b^I}{N_r} \right] \cdot \sin \theta \\ \begin{cases} \theta_0^j = \theta_0^I + j \frac{\theta_0^{II} - \theta_0^I}{N_r} \\ \theta_1^j = \theta_1^I + j \frac{\theta_1^{II} - \theta_1^I}{N_r} \end{cases} \quad (j = 0, 1, \dots, N_r) \end{cases} \quad (16)$$

式(16)已经包含了两条边界曲线,当给定一个 (i, θ) 时,便可求出一个点 (x, y) ,便于程序实现。

对于第 i 层单元中第 j 个单元,单元局部编码如图6所示,它是由第 $(j-1)$ 条和第 j 条曲线与第 i 层的经向线围成。其各结点对应的离心角与坐标为:

$$\begin{aligned}
 a_1 &= \theta_0^{j-1} + (i-1) \cdot \Delta Q^{j-1}, \\
 a_2 &= \theta_0^j + (i-1) \cdot \Delta Q^j, \\
 a_3 &= a_1 + \Delta Q^{j-1}, \\
 a_4 &= a_2 + \Delta Q^j, \\
 a_7 &= (a_1 + a_3) / 2, \\
 a_8 &= (a_2 + a_4) / 2, \\
 x_1 &= x(j-1, a_1) & x_2 &= x(j, a_2) \\
 x_3 &= x(j-1, a_3) & x_4 &= x(j, a_4) \\
 x_5 &= (x_1 + x_2) / 2 & x_6 &= (x_3 + x_4) / 2 \\
 x_7 &= x(j-1, a_7) & x_8 &= x(j, a_8)
 \end{aligned} \tag{17}$$

式中: $\Delta Q^{j-1} = (Q_1^{j-1} - Q_0^{j-1}) / N_\theta$
 $\Delta Q^j = (Q_1^j - Q_0^j) / N_\theta$
 $i = 1, 2, \dots, N_r, j = 1, 2, \dots, N_r$

类似地可得到 y_1, y_2, \dots, y_8 。

结点编号: 当采用结点编号沿径向连续递增时, 不难按照前面的四边形二次单元的编号方法进行。而当环向单元的层数少于径向单元的个数时, 按径向结点编号连续递增进行整体编码, 带宽较大, 宜按环向结点编码连续递增的方法来编码。对于曲边域又不能像四边形域那样通过坐标变换来改变整体编码方向。须另行讨论。

在径向编码中, 先按径向逐个单元进行, 沿环向称为层; 在环向编码中, 应先按环向逐个单元进行, 而沿径向称为层。则第 j 层第 i 个单元的坐标计算公式仍可用式(17)。

第 j 层单元的起始边编码为:

$$\begin{aligned}
 N_1^j &= N_3^{j-1} \\
 N_2^j &= N_1^j + N_2^j - N_1^j + (j-1) \cdot (JE1 + JE2 + JE1 + JER) \\
 N_3^j &= N_2^j + N_3^j - N_2^j + (j-1) \cdot (JE1 + JE2 + JE1 + JER) \\
 &\quad (j = 2, 3, \dots, N_r)
 \end{aligned} \tag{18}$$

由此不难利用前面所述方法导出第 j 层第 i 个单元的转换数组 $M(i, j)$ 。

当两对边均为折线边界或其一为折线边界时, 只要对式(17)中相应的结点进行修改即可。

8 后 记

本程序由FORTRAN—77程序语言编制。其中网格图形显示部分由BASIC语言编制。这主要是利用BASIC的绘图功能。业已完成了网格图形的显示、窗口设置, 视见变换以及绘图机输出几部分。这部分内容将另文介绍。

作为有限元的前处理程序, 要保证为有限元计算提供的原始数据的正确性。本文只处理

了网格剖分的数据, 对于荷载、支承情况未加核校, 这部分工作尚在进行。

本文的程序可以为二维计算域的三角形、四边形网格剖分提供数据, 如弹性力学平面问题、轴对称问题以及简单的板壳单元。利用本文的处理方法不难编写出其它类型的单元剖分程序。作为发展方向, 尚可与SAPV程序连接为其提供上述类型的单元的数据。

图5给出了一个实例。

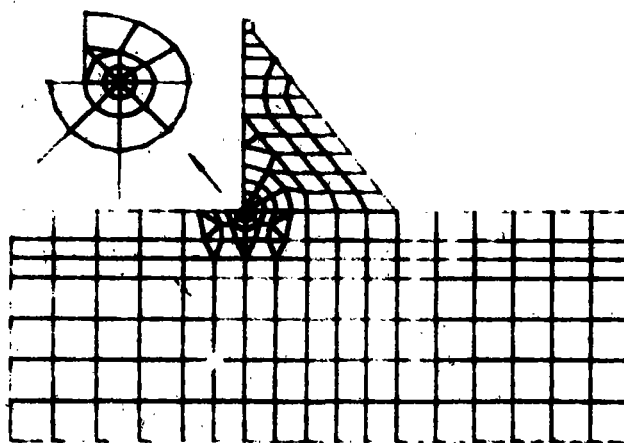


图5 混凝土重力坝坝踵断裂分析用网格

参 考 文 献

- [1] Y. Yamada, R. H. Gallagher, Theory and Practice in Finite Element Structural Analysis, University of Tokyo Press, 1973
- [2] 林泰道, 有限要素法 (自动要素分割手法 (平面构造))

The Discussion of Data Pro-handling for Finite Elements

Duan Yunling

(Dept. of Hydranlic Engineering)

Abstract: In this paper, the calculative region of finite elements is divided into several sub-regions, in which the calculative meshes can be generated automatically. The technic of the automatic generation of meshes, which used for the constant strain element (such as 3-node triangle and 4-node rectangle element) and linear strain element (such as 6-node triangle and 8-node rectangle), is discussed. The sub-regions and net of the calculative region can be displayed dynamically by using the technic of computer graphics. The method used in this paper is clear in conception, simple in principles, and more efficient.

Keywords: the calculative region, sub-region, the calculative meshes, automatic generation of meshes, constant strain element, linear strain element, graphics.