

力学参考资料(十)

有限元素法及其 在力学中的应用

译文集

115

科学技术文献出版社重庆分社

有限元素法及其在力学中的应用

中国科学技术情报研究所重庆分所 编辑
科学技术文献出版社重庆分社 出版
重庆市市中区胜利路91号

四川省新华书店重庆发行所 发行
重 庆 新 华 印 刷 厂 印刷

开本：787×1092毫米 $\frac{1}{16}$

印张：17

字数：55万

1979年3月第一版

1979年6月第一次印刷

印数：11000

统一书号：13176·40

定价：1.75元



目 录

应用有限元素法的设计方法——程序设计	(1)
有限元素法在计算机上的实现	(24)
反对称矩阵的波前法求解程序	(43)
一种新的超元矩阵 Householder-QR 分解公式	(62)
空间板结构体有限元素法元素自动分割程序《MESH/3D》	(66)
非规则网格上有限元素法的最优 L^{∞} 估计	(75)
非线性有限元方程的收敛性与稳定性	(83)
基于局部混合插值的子结构粗元素	(86)
平面应变问题离散化分析的一种新元素	(95)
固体力学问题的一种新离散模型	(98)
混合型薄壳有限元素	(100)
关于交感有限元素法的数学理论:	
精度、敛速及其有效使用原则	(129)
非线性有限单元结构分析中的摄动法	(136)
有限元素法流体力学应用的最新进展	(143)
加权余量法(述评)	(159)
粘性不可压缩流体动力学有限元解算法	(176)
用有限元素法计算不可压缩粘性流体	
——STREAM200通用程序——	(185)
用加权余量法对跨音速流作有限元分析	(190)
非定常可压边界层流动中的有限元素法分析	(198)
采用康托诺维奇变分的迭代有限元素法	(205)
用有限元素法对涡轮机子午流场的数值计算	(210)

用有限元素法对涡轮机叶轮中叶片间可压缩流场的计算(215)

涡轮机械通流计算的有限元素法(220)

不可压缩润滑问题的有限元解法(234)

用有限元素法分析不可压缩粘性流体中气泡的特性(244)

自由表面流问题的数值解(253)

应用有限元素法的设计方法*

程序设计 (I)

森田喜保

在前几次讲座中就有限元素法的理论进行了详细论述, 并且列举了不少适用的例题。但是, 要想用有限元素法分析实际问题还必须把理论具体化为程序。除了众所周知的大型结构分析通用程序ASKA和NASTRAN已经可以在普通的计算中心很容易地使用以外, 还不断地发表了许多程序实例^[1-4]。

有限元素法的最大优点之一就在于, 比如说如果完成了二维弹性应力分析程序, 则其中相当的部分在编制温度分布分析及轴对称应力分析程序时可以通用。即, 刚度矩阵的计算仅因对应的问题不同而异, 而刚性矩阵的组合以及问题的求解过程往往基本相同。因此如果掌握了一个有限元素法程序的技巧则其实用效果是显而易见的。所以本文在列举一个仅管是就小型电子计算机而言但实用上对大型计算毫无妨碍的程序例子的同时, 将就程序设计的技巧予以论述。

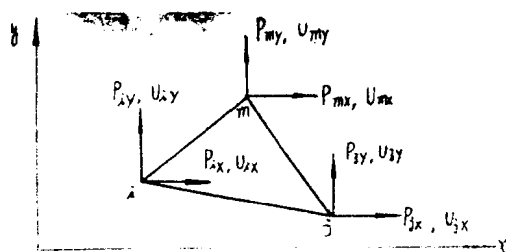


图1 一个三角形元素

解法

1. 一个元素的刚度方程

考虑平面弹性应力问题。由图1所示的三节点(i, j, m)组成的三角形元素的刚度矩阵为:

$$K = A \cdot h \cdot [B^T D B] \quad (2.1)$$

式中A是元素面积, h是板厚, 而B与D分别由下式给

定:

$$B = \frac{1}{2A} \begin{bmatrix} b_i & 0 & b_j & 0 & b_m & 0 \\ 0 & c_i & 0 & c_j & 0 & c_m \\ c_i & b_i & c_j & b_j & c_m & b_m \end{bmatrix}, \quad (2.2)$$

$$D = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \quad (2.3)$$

$$b_i = y_j - y_m, \quad c_i = x_m - x_j, \quad \text{等等}$$

其中E是弹性模量, ν 是泊桑比。如果元素的节点坐标与材料常数一经给定就可以由这些关系式计算元素的刚度矩阵。

2. 刚度矩阵的组合

用有限元素法进行计算的第一步, 需要将结构分析对象分割成三角形代之以许多元素组成的集合体。如果那样, 每个元素的刚度矩阵就可以用(2.1)式求得。因为对所有元素都能得到一个六阶平衡方程式, 故将这些方程式排列起来对整个结构来说就构成了阶数为 $6 \times (\text{元素数})$ 的方程式。然而如果象下面的例题所表示的那样就节点位移进行整理, 则方程式的阶数为 $2 \times (\text{节点数})$ 。现在考虑如图2所示由8个元素构成的物体, 如果元素编号及节点编号如图8所示那样确定, 则元素e的刚度方程为:

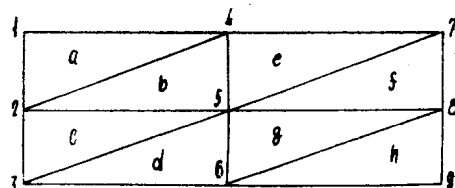


图2 刚度矩阵的组合

* 本文系“有限元素法基础与应用讲座”之一部分

$$\begin{Bmatrix} P_{4x}^e \\ P_{4y}^e \\ P_{5x}^e \\ P_{5y}^e \\ P_{7x}^e \\ P_{7y}^e \end{Bmatrix} = K \begin{Bmatrix} u_{4x}^e \\ u_{4y}^e \\ u_{5x}^e \\ u_{5y}^e \\ u_{7x}^e \\ u_{7y}^e \end{Bmatrix} = \begin{Bmatrix} K_{4x4x}^e & & & & & \\ K_{4y4x}^e & K_{4y4y}^e & & & & \\ K_{5x4x}^e & K_{5x4y}^e & K_{5x5x}^e & & & \\ K_{5y4x}^e & K_{5y4y}^e & K_{5y5x}^e & K_{5y5y}^e & & \\ K_{7x4x}^e & K_{7x4y}^e & K_{7x5x}^e & K_{7x5y}^e & K_{7x7x}^e & \\ K_{7y4x}^e & K_{7y4y}^e & K_{7y5x}^e & K_{7y5y}^e & K_{7y7x}^e & K_{7y7y}^e \end{Bmatrix} \quad \text{对 称} \quad (2.4)$$

对其他元素也可得到同样的关系式。在此 u_{4x}^e 是元素 e 的第4个节点上沿 x 方向的位移,而 P_{4x}^e 是同一位置沿 x 方向的节点力。其他符号意义与此相同(参看图1)。另外如果着眼于节点位移,因为

$$u_{4x}^e = u_{4x}^a = u_{4x}^b, \quad u_{4y}^e = u_{4y}^a = u_{4y}^b, \quad \dots \quad (2.5)$$

必须成立,所以现在若假定结构总体刚度矩阵方程为:

$$\begin{Bmatrix} P_{1x} \\ P_{1y} \\ P_{2x} \\ P_{2y} \\ \vdots \end{Bmatrix} = \begin{Bmatrix} K_{1x1x} & & & \\ K_{1y1x} & K_{1y1y} & & \\ K_{2x1x} & K_{2x1y} & K_{2x2x} & \\ K_{2y1x} & K_{2y1y} & K_{2y2x} & K_{2y2y} \\ \vdots & \vdots & \vdots & \vdots \end{Bmatrix} \begin{Bmatrix} u_{1x} \\ u_{1y} \\ u_{2x} \\ u_{2y} \\ \vdots \end{Bmatrix} \quad (2.6)$$

矩阵的各个分量用单个元素的分量来表示则有:

$$K_{1x1x} = K_{1x1x}^a, \quad K_{2x2x} = K_{2x2x}^a + K_{2x2x}^b + K_{2x2x}^c, \quad \dots \quad (2.7)$$

同样

$$P_{1x} = P_{1x}^a, \quad P_{2x} = P_{2x}^a + P_{2x}^b + P_{2x}^c, \quad \dots \quad (2.8)$$

式中 P_{1x} , P_{2x} 等于作用在各个节点上的外力。

3. 解 法

因为(2.6)式是多元联立一次方程组,所以若把边界条件考虑进去不难求解。但是,如果要求解实际问题,由于所得方程的阶数大体上是节点数的倍数关系,所以直接计算上式从计算机的容量与解的精度来看都是不合适的。

刚度矩阵一般具有如下特性:

(1)零元素多;(2)正定,对称。故通常采用利用这些性质的解法。在此就作为迭代法之一的共轭梯度法予以论述。如果将(b)式简写为:

$$P = K \cdot u \quad (2.9)$$

则共轭斜量法的公式具有如下形式⁽⁷⁾:

$$q_0 = r_0 = P - Ku_0 \quad (\text{初始值}) \quad (2.10)$$

$$\left. \begin{aligned} \alpha_k &= (q_k, q_k) / (Kq_k, q_k) \\ u_{k+1} &= u_k + \alpha_k q_k \\ \gamma_{k+1} &= \gamma_k - \alpha_k Kq_k \\ \beta_k &= (Kq_k, \gamma_{k+1}) / (Kq_k, q_k) \\ q_{k+1} &= \gamma_{k+1} - \beta_k q_k \end{aligned} \right\} (k=0,1,2,\dots) \quad (2.11)$$

式中 $(\cdot, \cdot)$ 表示内积。而 u_0 是位移的第一次近似解。由这些式子可以看出刚度矩阵 K 的各个元素中仅仅非零元素对计算有效。但是按照消去法,因为零元素消元时一般要变成非零元素,故零元素也必须存贮,与此对应,在共轭斜量法中因为只要存贮非零元素就行了,所以更能合理利用机器容量。而且如果不引入计算误差,共轭斜量法即便在病态情况下按照方程式的阶数迭代也能保证达到正确的解答⁽⁶⁾,故作为有限元素法的计算方法而被广泛使用。在(2.9)式中是节点位移全都作为未知数的情况。而在节点位移作为边界条件予以给定的问题中,那些给定的位移必须消去。

如果将该方程具体地写出来,则有如下形式:

$$\begin{Bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \end{Bmatrix} = \begin{Bmatrix} K_{11} & & & & \\ K_{21} & K_{22} & & & \\ K_{31} & K_{32} & K_{33} & & \\ K_{41} & K_{42} & K_{43} & K_{44} & \\ K_{51} & K_{52} & K_{53} & K_{54} & K_{55} \end{Bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{Bmatrix} \quad \text{对 称} \quad (2.12)$$

在上式中,例如 u_4 的值给定作为边界条件,则(2.12)式可如下分解为:

$$\begin{Bmatrix} P_1 - K_{41}u_4 \\ P_2 - K_{42}u_4 \\ P_3 - K_{43}u_4 \\ P_5 - K_{45}u_4 \end{Bmatrix} = \begin{Bmatrix} K_{11} & & & \\ K_{21} & K_{22} & & \\ K_{31} & K_{32} & K_{33} & \\ K_{51} & K_{52} & K_{53} & K_{55} \end{Bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_5 \end{Bmatrix} \quad (2.13)$$

$$P_4 = [K_{41} \ K_{42} \ K_{43} \ K_{44} \ K_{45}] \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{Bmatrix} \quad (2.14)$$

因为在位移未知的节点(1, 2, 3, 5)上,外力全部是已知的,故方程(2.13)就是待求解的方程。节点4上的外力 P_4 在节点位移求出来以后可以用(2.14)求得。

于是,如果根据以上计算求出了节点位移,则用这些结果可以按照下式计算元素内的应力与应变。

$$\{\varepsilon\} = B \cdot \{u\} \quad (2.15)$$

$$\{\sigma\} = D \cdot \{\varepsilon\} = D \cdot B \{u\} \quad (2.16)$$

方 框 图

如果概括地用方框图表示上述计算则如表1所示。
现将每个小框图的内容略述如下：

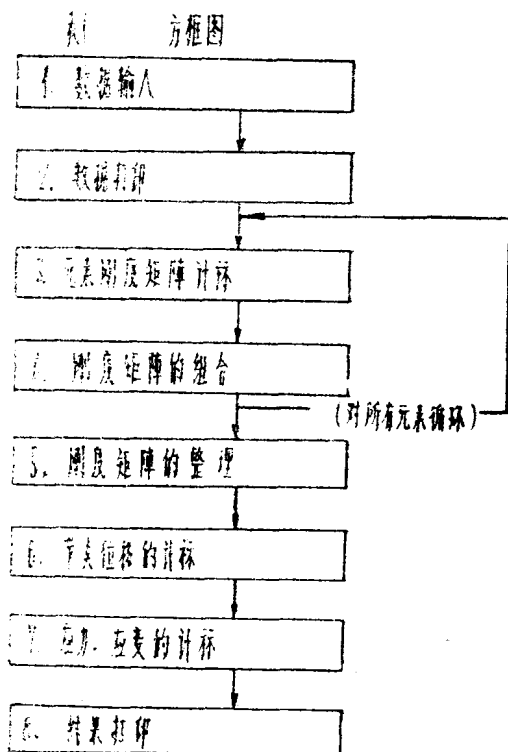
1. 数据输入(参看表2)
2. 数据打印(用与输入同样的格式打印)
3. 元素刚度矩阵的计算(参看表3)
4. 刚度矩阵的组合(参看表4)
5. 刚度矩阵的整理(消去与给定的位移边界条件有关的刚度矩阵的分量)(参看图5)
6. 节点位移的计算(共轭梯度法)(参看表6)
7. 应力、应变的计算与结果打印(参看表7)

程 序 实 例

因为在此列出的程序是按照前一节的框图编制的, 故其主要特征为:

(1) 是复盖结构, 即便是容量很小(64千字节)的计算机若能配上若干外部存贮装置也能进行直到有400个节点800个元素的计算。

(2) 对联立方程的求解采用了共轭斜量法。因而



数据的制作方法与单位分割法相比多少简单些。

(3) 能进行采用三节点三角形元素的平面应力分析。

现将使用本程序时的主要注意事项归纳如下:

(1) 建立元素数据时必须将构成各元素的三个节点按反时针顺序排列。处理简单的形状时采用元素自动划分是方便的^[10]。

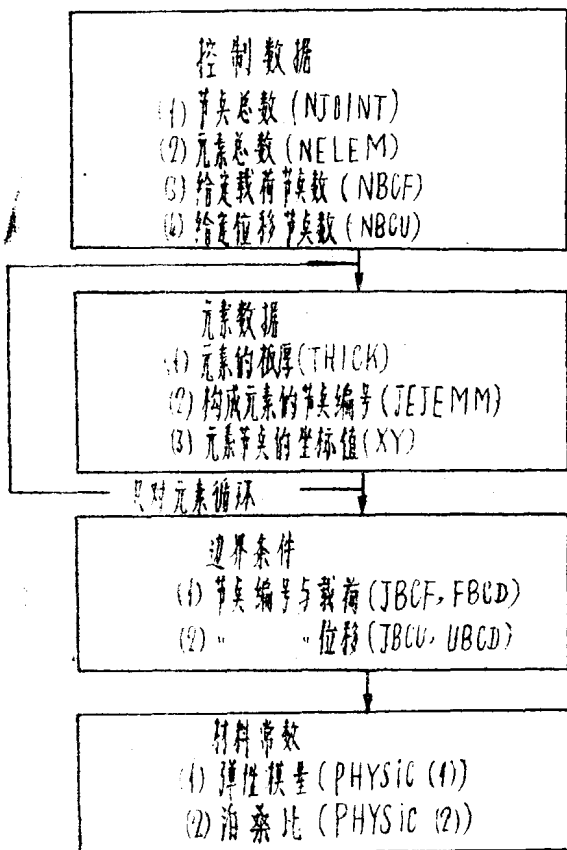
(2) 利用形状的对称性作为位移边界条件进行计算时, 往往变成滚棒支承(例如 y 方向位移是零, x 方向位移未定那样的条件)的情况。

在此因为对于一个节点编号同时输入 x 方向与 y 方向的位移, 输入未知位移后需要予以识别。在本程序中大于99.0的数值作为位移边界条件输入时, 该位移要变为未知位移。

(3) 材料常数以弹性模量与泊桑比作为一组, 可以考虑五组。控制数据的NPHYS是这种数组的总数。

而且在元素数据中指定材料常数数组的编号为LB。

表2. 数据输入



但是LB为1时取零(空白)就行了。(见程序 IODATA 中第I-0130条——译者注)

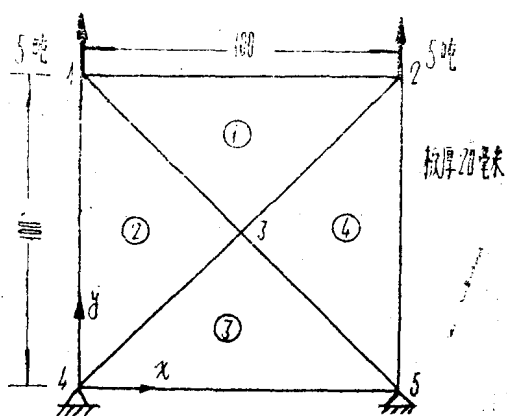


图3 数值计算模型

表3 元素刚度矩阵的计算

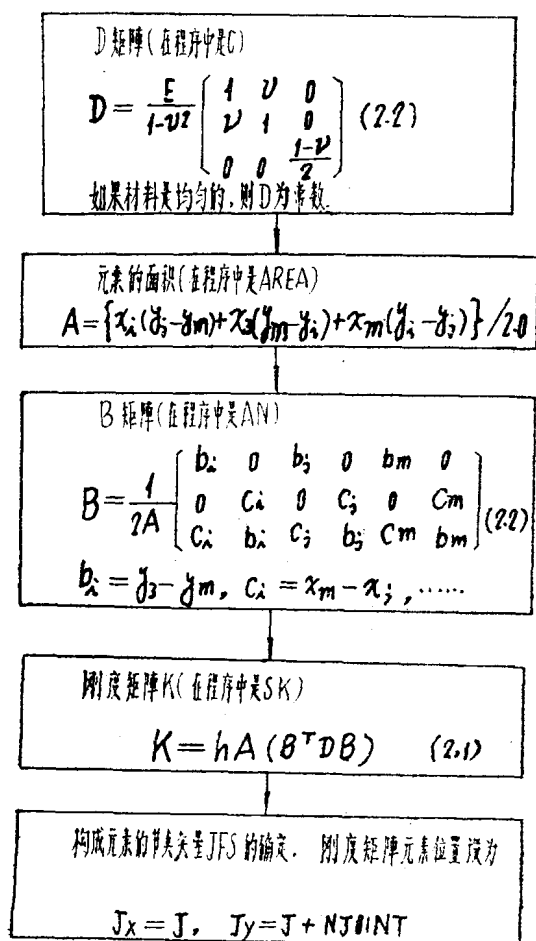
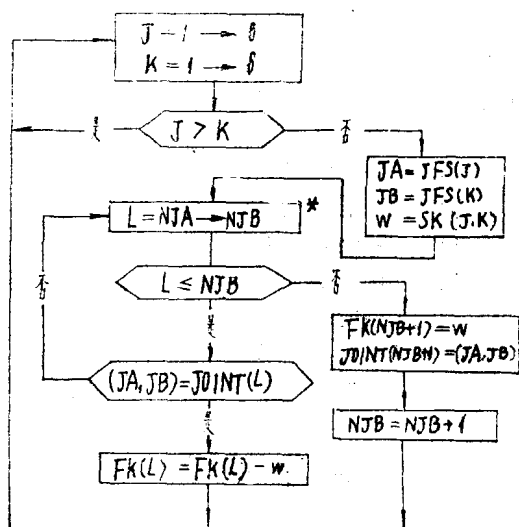
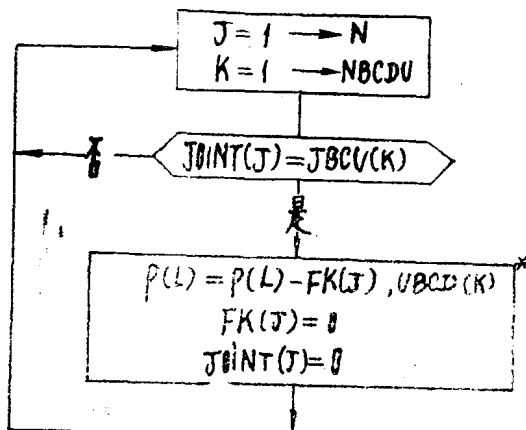


表4 刚度矩阵的组合



* 我们采用了满足不等式 $NJB - JOINT(J) \geq 1$ 的 NJA, 而当 $NJB - JOINT(J) = 0$ 时 $NJA = 1$, 因为 $NJA = 1$ 也行, 但当 NJB 变大时则无意义的计算时间。

表5 刚度矩阵的整理



* JBCU(K) = JOINT(J, 1) 时, $L = JOINT(J, 2)$
JBCU(K) = JOINT(J, 2) 时 $L = JOINT(J, 1)$

(4) 输入数据在程序 IODATA 的 I-0280 及 I-0780 两个地方可以对数据进行检查。在最初的检查中可以检查出元素数据卡的遗漏及不要的数据卡的混入。为了进行这一检查必须在元素数据卡后插入一张从第1列至第5列穿孔为99999的卡片。随后的检查可以发现数据形成的错误。

表6 节点位移的计算

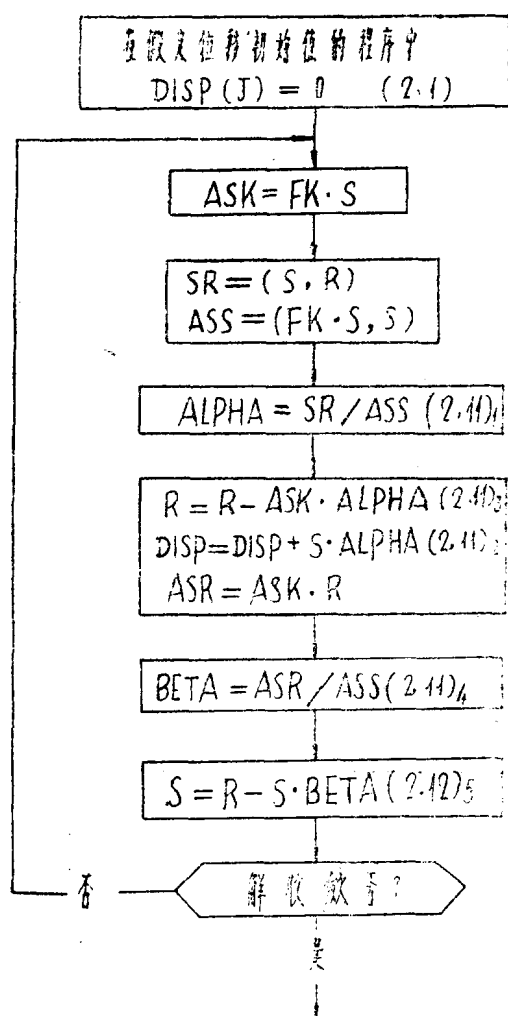
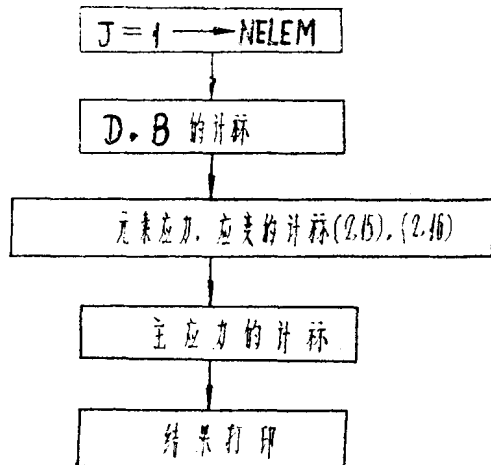


表7 应力、应变的计算及结果打印



另外，在程序设计时考虑了如下几点：

- (1) 使用尽可能易于明了的符号；
- (2) 将程序分割成子程序，便于在其他问题中通用；
- (3) 节省计算时间；
- (4) 有效利用机器内存。

数值计算例题

考虑图3所示的一边固定，其对边作用10吨拉力的厚度为20毫米宽为100×100毫米²的正方形板。现在将此板进行分割，并标以如图所示的节点编号(1, 2, ..., 5)及元素编号(①, ②, ..., ④)。边界条件为：

位移：节点4, 5上的x方向及y方向的位移为零。

载荷：节点1, 2上的y方向的力都是5吨。计算这样的模型时输入数据如表8所示。弹性模量与泊桑比分别为21,000公斤/毫米², 0.3(参看表8)。

一般编制程序时在调整阶段若能知道计算的中间结果往往是方便的。因此对上面的数值计算例题，输出了考虑位移边界条件进行整理前后的结构总体刚度矩阵。

表9是整理前的结果而表10是整理后的结果，每个表的(a)表示刚度矩阵而(b)则表示与之相应的行和列的位置。另外，表11是与表10的结果处于同一时刻的载荷矢量。在表9与表10的(b)中用节点编号本身表示节点的x分量，而用节点总数加节点编号的数值表示节点y的分量。

表12是经过上述计算得到的应力同应变的结果。因为假定节点4同节点5固定，故出现各元素x方向的应力分量，但若将两个节点中某一个的x方向的位移作为未知位移则变成均匀拉伸应力状态，y方向的应力 σ_y 是5.00吨/毫米²，其他应力分量全为零。

另外，表8上节点3的x方向的位移值非常小而不完全为零，这是由于数值计算误差所致。

结 束 语

用有限元素法分析实际问题时需要编制程序。在此仅列出了平面应力问题基本解法的方框图及其具体程序，不可能讲得太详细，但这些对其他问题的程序设计也是有效的，本文的程序只要稍加修改无论对轴对称问题的应力分析还是对热传导等问题的分析都能适用，也是其他方面问题的基础。

表8 数值计算模型输入数据

THE ELAST STRESS IN TWO DIMENSIONAL PROBLEM

TITLE **HEIMEN DANSEI** SHOKYO TO CG NO HIKAKU* NELEM=4 NJOINT=5

NO.			JOINT	X(mm)	Y(mm)	X(mm)	Y(mm)	X(mm)
1	1	3	2	0.0	100.00	50.00	50.00	100.00
2	1	4	3	0.0	100.00	0.0	0.0	50.00
3	4	5	3	0.0	0.0	100.00	0.0	50.00
4	2	3	5	100.0	100.0	50.00	50.00	100.00

NBCDF=2

NBCDU=4

BOUNDARY CONDITION
DISPLACEMENT

Y(mm)

THICK

JOINT

U(mm)

100.00

20.00

1

4

0.0

50.00

20.00

2

9

0.0

50.00

20.00

3

5

0.0

0.0

20.00

4

10

0.0

FORCE

JOINT

F(kg)

1

6

5000.000

2

7

5000.000

YOUNG

POISSN

0.210E 05

0.300E 00

表9 数值计算例题整理前的刚度矩阵

(a)

0.3115E 06	-0.1500E 06	-0.3115E 06	0.1500E 06	-0.7500E 05
0.5769E 04	0.3115E 06	0.1500E 06	-0.3115E 06	-0.5769E 04
0.7500E 05	0.7846E 06	0.0	-0.3115E 06	-0.1500E 06
0.1246E 07	-0.1500E 06	-0.3115E 06	0.3115E 06	0.1500E 06
0.3115E 06	0.7500E 05	-0.5769E 04	0.5769E 04	-0.7500E 05
0.3115E 06	0.1500E 06	-0.3115E 06	-0.1500E 06	0.3115E 06
-0.1500E 06	-0.3115E 06	-0.7500E 05	-0.5769E 04	0.5769E 04
0.7500E 05	0.3115E 06	-0.1500E 06	-0.3115E 06	0.1500E 06
0.3115E 06	0.1500E 06	-0.3115E 06	0.7500E 05	0.5769E 04
-0.5769E 04	-0.7500E 05	0.4615E 06		

(b)

1	1	1	6	1	3	1	8	1	2
1	7	6	6	6	3	6	8	6	2
6	7	3	3	3	8	3	2	3	7
8	8	8	2	8	7	2	2	2	7
7	7	1	4	1	9	6	4	6	9
4	4	4	9	4	3	4	8	9	9
9	3	9	8	4	5	4	10	9	5
9	10	5	5	5	10	5	3	5	8
10	10	10	3	10	8	2	5	2	10
7	5	7	10	3	3				

表10 数值计算例题整理后的刚度矩阵

(a)									
0.3115E 06		- 0.1500E 06		- 0.3115E 06		0.1500E 06		- 0.7500E 05	
0.5769E 04		0.3115E 06		0.1500E 06		- 0.3115E 06		- 0.5769E 04	
0.7500E 05		0.7846E 06		- 0.3115E 06		- 0.1500E 06		0.1246E 07	
- 0.1500E 06		- 0.3115E 06		0.1500E 06		0.1500E 06		0.3115E 06	
0.4615E 06									
(b)									
1	1	1	6	1	3	1	8	1	2
1	7	6	6	6	3	6	8	6	2
6	7	3	3	3	2	3	7	8	8
8	2	8	7	2	2	2	7	7	7
3	3								

表11 整理后的载荷矢量

0.0	0.0	0.0	0.0	0.0
0.5000E 04	0.5000E 04	0.0	0.0	0.0

表12 应力与节点位移

NO.	JOINT	SIGMA-X	SIGMA-Y	SIGMA-Z	TAU-XY	SIGMA-1	SIGMA-2
1	1 3 2	-0.35	5.19	0.0	0.00	5.19	-0.35
2	1 4 3	0.54	5.00	0.0	0.19	5.01	0.54
3	4 5 3	1.44	4.81	0.0	-0.00	4.81	1.44
4	2 3 5	0.54	5.00	0.0	-0.19	5.01	0.54
JOINT	UX(mm)		UY(mm)		X(mm)		Y(mm)
1	0.4550E-02		0.2303E-01		0.0		100.02
2	-0.4550E-02		0.2303E-01		100.00		100.02
3	-0.8611E-09		0.1042E-01		50.00		50.01
4	0.0		0.0		0.0		0.0
5	0.0		0.0		100.00		0.0

采用有限元素法的设计方法

```

INTEGER*2 JBCF JBCU
COMMON NELEM,NJOINT,NBCDF,NBCDU,M,N,NMAXU,NUNIT,TITLE(10),FBCD(20).
1 UBCD(50),PHYSIC(2,5),DISP(800),JBCF(20),JBCU(50)
C MAIN PROGRAM
CALL LOAD('HEIMO1')
CALL IODATA
CALL LOAD('HEIMO2')
CALL SFKMAT
CALL LOAD('HEIMO3')
CALL CGM
CALL LOAD('HEIMO4')
CALL STRPRI
STOP
END
SUBROUTINE IODATA
INTEGER*2 JBCF,JBCU,JELEM
COMMON NELEM,NJOINT,NBCDF,NBCDU,M,N,NMAXU,NUNIT,TITLE(10).
1 FBCD(20),UBCD(50),PHYSIC(2,5),DISP(800),JBCF(20),JBCU(50)
DIMENSION XY(2,400),THICK(800),JELEMM(4,800)
REWIND 1
REWIND 2
REWIND 3
KKK=0
NMAXU=600
NBCDF=0
NBCDU=0
C READ(5,1001) (TITLE(J),J=1,10)
READ(5,1002) NJOINT,NELEM,NBCF,NBCU,NPHYS
DO 11 J=1,NELEM
READ(5,1003) TH,LB,JI,JJ,JK,XI,YI,XJ,YJ,XK,YK
IF(LB.EQ.0) LB=1
JELEMM(1,J)=JI
JELEMM(2,J)=JJ
JELEMM(3,J)=JK
JELEMM(4,J)=LB
XY(1,JI)=XI
XY(2,JI)=YI
XY(1,JJ)=XJ
XY(2,JJ)=YJ
XY(1,JK)=XK
XY(2,JK)=YK
THICK(J)=TH
11 CONTINUE
C READ(5,1002) JA,JB,LB,JI,JJ
IF(JA.EQ.99999) GO TO 12
WRITE(6,2000) KKK,JA,JB,LB
STOP 8001
C 12 JB=0
IF(NBCF.EQ.0) GO TO 15
DO 14 J=1,NBCF
READ(5,1004) JA,WX,WY
IF(WX.EQ.0) GO TO 13
JB=JB+1
JBCF(JB)=JA
FBCD(JB)=WX
13 IF(WY.EQ.0) GO TO 14
JB=JB+1
JBCF(JB)=JA+NJOINT
FBCD(JB)=WY
14 CONTINUE
NBCDF=JA

```

I-013C

I-0280

```

      JB=0
C
15  DO 17 J=1,NBCU
    READ(5,1004) JA,WX,WY
    IF(WX*GE*99*0) GO TO 16
    JB=JB+1
    JBCU(JB)=JA
    UBCD(JB)=WX
16  IF(WY*GE*99*0) GO TO 17
    JB=JB+1
    JBCU(JB)=JA+NJOINT
    UBCD(JB)=WY
17  CONTINUE
    NBCDU=JB
    READ(5,1005) ((PHYSIC(JA,JB),JA=1,2),JB=1,NPHYS)
C
    M=NJOINT*2
    WRITE(6,2001) (TITLE(J),J=1,10),NELEM,NJOINT,NBCDF,NBCDU
    DO 22 J=1,NELEM
      JI=JELEMM(1,J)
      JJ=JELEMM(2,J)
      JK=JELEMM(3,J)
      XI=XY(1,JI)
      YI=XY(2,JI)
      XJ=XY(1,JJ)
      YJ=XY(2,JJ)
      XK=XY(1,JK)
      YK=XY(2,JK)
      TH=THICK(J)
      WRITE(6,2002) J,JI,JJ,JK,XI,YI,XJ,YJ,XK,YK,TH
      IF(XI*(YJ-YK)+XJ*(YK-YI)+XK*(YI-YJ)) 21,21,22
21  WRITE(6,2000) J,JI,JJ,JK
      KKK=1
22  CONTINUE
C
      WRITE(6,2003)
      DO 23 J=1,NBCDU
C
23  WRITE(6,2004) J,JBCU(J),UBCD(J)
C
      IF(NBCDF*EQ*0) GO TO 25
      WRITE(6,2005)
      DO 24 J=1,NBCDF
C
24  WRITE(6,2004) J,JBCF(J),FBCD(J)
C
25  WRITE(6,2006) ((PHYSIC(JA,JB),JA=1,2),JB=1,NPHYS)
      IF(KKK*EQ*1) STOP 8002
      WRITE(1) ((XY(JA,JB),JA=1,2),JB=1,NJOINT),(THICK(JA),JA=1,NELEM,
1  ((JELEMM(JA,JB),JA=1,4),JB=1,NELEM)
      RETURN
C
1001 FORMAT(10A4)
1002 FORMAT(5I5)
1003 FORMAT(F6.1,I1,4X,3I3,3(1X,2F7.2))
1004 FORMAT(I5,2F10.3)
1005 FORMAT(2F10.3)
C
2000 FORMAT(1X,'ERROR*',4I51)
2001 FORMAT(1H1,'THE ELAST STRESS IN TWO DIMENSIONAL PROBLEM'/1X,
1  'TITLE*',10A4,5X,'NELEM=',I3,'NJOINT=',I3,'NBCDF=',I2,2X,
2  'NBCDU=',I2//5X,'NO.',13X,'JOINT',3(6X,'X(mm) Y(mm)'),'THICK')
2002 FORMAT(5X,I3,3X,3I5,3(3X,2F8.2),F10.2)
2003 FORMAT(1H1,'BOUNDARY CONDISION'/1X,'DISPLECEMENT'/10X,
1  'JOINT      U(mm)')
2004 FORMAT(5X,2I5,F10.3)
2005 FORMAT(/1X,'FORCE'/10X,'JOINT      F(kg)')

```

I-0780

```

2006  FORMAT(/10X,'YOUNG          POISSN'/5X,5(2E10.3/5X))
      END
      SUBROUTINE SFKMAT
      INTEGER*2 JBCF,JBCU,JOINT,JELEMM
      COMMON NELEM,NJOINT,NBCDF,NBCDU,M,N,NMAXU NUNIT,TITLE(10),
1  FBCD(20),UBCD(50),PHYSIC(2,5),DISP(800),JBCF(20),JBCU(50)
      DIMENSION XY(2,400),THICK(800),C(3,3),AN(3,6),CAN(3,6),SK(6,6),
1  P(800),FK(600),JFS(6),JOINT(600,2),JELEMM(4,800)
      REWIND 1
      REWIND 3
      DO 5 J=1,M
5  P(J)=0.0
      DO 6 J=1,NBCDF
      JA=JBCF(J)
6  P(JA)=FBCD(J)
      II=0
      NUNIT=0
      READ(1) ((XY(JA,JB),JA=1,2),JB=1,NJOINT),(THICK(JA),JA=1,NELEM),
1  ((JELEMM(JA,JB),JA=1,4),JB=1,NELEM)
C
10  N=0
      NJB=1
      DO 11 J=1,NMAXU
      FK(J)=0.0
      DO 11 K=1,2
11  JOINT(J,K)=0
C
20  IF(II.GE.NELEM) GO TO 40
      II=II+1
      JI=JELEMM(1,II)
      JJ=JELEMM(2,II)
      JK=JELEMM(3,II)
      JB=JELEMM(4,II)
      XI=XY(1,JI)
      YI=XY(2,JI)
      XJ=XY(1,JJ)
      YJ=XY(2,JJ)
      XK=XY(1,JK)
      YK=XY(2,JK)
      TH=THICK(II)
      YOUNG=PHYSIC(1,JB)
      POISS=PHYSIC(2,JB)
      AREA=(XI*(YJ-YK)+XJ*(YK-YI)+XK*(YI-YJ))/2.0
C
      WA=YOUNG*T11/(1.0-POISS*2)
      DO 21 J=1,3
      DO 21 K=1,3
      C(J,K)=0.0
      IF(J.EQ.K) C(J,K)=WA
21  CONTINUE
C
      C(1,2)=WA*POISS
      C(2,1)=WA*POISS
      C(3,3)=WA*(1.0-POISS)/2.0
C
      DO 22 J=1,3
      DO 22 K=1,6
22  AN(J,K)=0.0
      WA=2.0*AREA
      AN(1,1)=(YJ-YK)/WA
      AN(1,3)=(YK-YI)/WA
      AN(1,5)=(YI-YJ)/WA
      AN(2,2)=(XK-XJ)/WA
      AN(2,4)=(XI-XK)/WA
      AN(2,6)=(XJ-XI)/WA

```

```

DO 23 J = 1,6,2
K = J + 1
AN(3,J) = AN(2,K)
23 AN(3,K) = AN(1+J)
C
DO 24 J = 1,3
DO 24 K = 1,6
CAN(J,K) = 0.0
DO 24 L = 1,3
24 CAN(J,K) = CAN(J,K) + C(J,L) * AN(L,K)
C
DO 26 J = 1,6
DO 26 K = 1,6
SK(J,K) = 0.0
DO 25 L = 1,3
25 SK(J,K) = SK(J,K) + AN(L,J) * CAN(L,K)
26 SK(J,K) = SK(J,K) * AREA
C
JFS(1) = JI
JFS(2) = JI + NJOINT
JFS(3) = JJ
JFS(4) = JJ + NJOINT
JFS(5) = JK
JFS(6) = JK + NJOINT
C
DO 37 J = 1,6
JA = JFS(J)
DO 37 K = 1,6
IF(J.GT.K) GO TO 37
JB = JFS(K)
WA = SK(J,K)
IF(NJB.LE.1) GO TO 36
NJA = NJB - 30
IF(NJA.LE.0) NJA = 1
DO 35 L = NJA,NJB
JL = JOINT(L,1)
JM = JOINT(L,2)
IF(JL + JM - JA - JB) 35,32,35
32 IF(IABS(JL - JM) - IABS(JA - JB)) 35,33,35
33 FK(L) = FK(L) + WA
GO TO 37
35 CONTINUE
C
36 N = N + 1
JOINT(N,1) = JA
JOINT(N,2) = JB
FK(N) = WA
37 CONTINUE
NJB = N
IF(NJB.LT.NMAXU - 20) GO TO 20
C
40 NN = 0
WRITE(6,6000) (FK(J),J = 1,N)
WRITE(6,6001) ((JOINT(JA,JB),JB = 1,2),JA = 1,N)
DO 43 J = 1,N
JL = JOINT(J,1)
JM = JOINT(J,2)
WA = FK(J)
IF(ABS(WA).LT.1.0E-9) GO TO 43
DO 42 K = 1,NBCDU
JA = JBCU(K)
IF(JL.NE.JA) GO TO 41
IF(JL.NE.JM) P(JM) = P(JM) - WA * UBCD(K)
GO TO 43
41 IF(JM.NE.JA) GO TO 42

```

```

      IF(JL.NE.JM) P(JL) = P(JL) - WA*UBCD(K)
      GO TO 43
42  CONTINUE
C
      NN = NN + 1
      JOINT(NN,1) = JL
      JOINT(NN,2) = JM
      FK(NN) = WA
43  CONTINUE
      WRITE(6,6000) (FK(J),J=1,NN)
      WRITE(6,6001) ((JOINT(JA,JB),JB=1,2),JA=1,N)
C
      NUNIT = NUNIT + 1
      WRITE(3) NN,(FK(JA),JA=1,NN),((JOINT(JA,JB),JA=1,NN),JB=1,2)
      IF(I.LT.NELEM) GO TO 10
C
      DO 46 J=1,NBCDU
      JA = JBCU(J)
46  P(JA) = 0.0
C
      DO 47 J=1,M
      WA = P(J)
      IF(ABS(WA).LT.1.OE-9) WA = 0.0
47  DISP(J) = WA
      WRITE(6,6002) (DISP(J),J=1,M)
      END FILE 3
      RETURN
6000  FORMAT(5X,10E12.4)
6001  FORMAT(10(4X,2I4))
6002  FORMAT(10E12.4)
      END
      SUBROUTINE CGM
      INTEGER*2 JBCF,JBCU,JOINT
      COMMON NELEM,NJOINT,NBCDF,NBCDU,M,N,NMAXU,NUNIT,TITLE(10),
1  FBCE(20),UBCD(50),PHYSIC(2,5),DISP(800),JBCF(20),JBCU(50)
      DIMENSION R(800),S(800),ASK(800),FK(600),JOINT(600,2)
      DCUBLE PRECISION R,S,ASK,SR,ASS,ASR,DELTA,ALPHA,BETA,WVV,W,DIS
C
      DO 1 J=1,M
      R(J) = DBLE(DISP(J))
      S(J) = R(J)
      ASK(J) = 0.0
1  DISP(J) = 0.0
      NA = 10
      NI = 1
      MN = M - NBCDU
      ERROR = 1.0E-5
C
C  ** ITERATION **
5  REWIND 3
      W = 0.0
      SR = 0.0
      ASS = 0.0
      ASR = 0.0
      DELTA = 0.0
C
      DO 10 K=1,NUNIT
      FEAD(3) N,(FK(JA),JA=1,N),((JOINT(JA,JB),JA=1,N),JB=1,2)
      DO 10 J=1,N
      JJ = JOINT(J,1)
      JK = JOINT(J,2)
      ASK(JJ) = ASK(JJ) + DBLE(FK(J)) * S(JK)
      IF(JJ.EQ.JK) GO TO 10
      ASK(JK) = ASK(JK) + DBLE(FK(J)) * S(JJ)
10  CONTINUE

```

```

C
DO 15 J=1,M
SR=SR+S(J)*R(J)
15 ASS=ASS+ASK(J)*S(J)
ALPHA=SR/ASS

C
DO 20 J=1,M
DIS=DBLE(DISP(J))
W=W+DIS**2
R(J)=R(J)-ASK(J)*ALPHA
WVV=DIS+S(J)*ALPHA
DELTA=DELTA+(DIS-WVV)**2
ASR=ASR+ASK(J)*R(J)
20 DISP(J)=WVV
BETA=ASR/ASS

C
DO 25 J=1,M
S(J)=R(J)-S(J)*BETA
25 ASK(J)=0.0
IF(W.LT.1.0E-10) W=1.0
W=DSQRT(DELTA/W)
IF(W.LT.DBLE(ERROR)) GO TO 35
IF(NI.LT.NA) GO TO 30
NA=NA+10
WRITE(6,1000) NI,W
30 NI=NI+1
IF(NI.LT.MN) GO TO 5
35 CONTINUE
RETURN

C
1000 FORMAT(1X,'ITERATIVE CYCLE NUMBER',I5,5X,'ERROR=',E10.5)
END
SUBROUTINE STRPRI
INTEGER*2 JBCF,JBCU,JELEMM
COMMON NELEM,NJOINT,NBCDF,NBCDU,M,N,NMAXU,NUNIT,TITLE(10),
1 FBCD(20),UBCD(50),PHYSIC(2,5),DISP(800),JBCF(20),JBCU(50)
DIMENSION XY(2,400),THICK(800),C(3,3),AN(3,6),CAN(3,6),WVA(6),
C 1 STRESS(4),S TRAIN(4),JELEMM(4,800)
REWIND 1
READ(1)((XY(JA,JB),JA=1,2),JB=1,NJOINT),(THICK(JA),JA=1,NELEM),
1 ((J,ELEMM(JA,JB),JA=1,4),JB=1,NELEM)
WRITE(6,2001) (TITLE(J),J=1,10)
DO 2 J=1,NBCDU
JA=JBCU(J)
2 DISP(JA)=UBCD(J)

C
DO 20 II=1,NELEM
JI=JELEMM(1,II)
JJ=JELEMM(2,II)
JK=JELEMM(3,II)
JB=JELEMM(4,II)
XI=XY(1,JI)
YI=XY(2,JI)
XJ=XY(1,JJ)
YJ=XY(2,JJ)
XK=XY(1,JK)
YK=XY(2,JK)
TH=THICK(II)
YOUNG=PHYSIC(1,JB)
POISS=PHYSIC(2,JB)
AREA=(XI*(YJ-YK)+XJ*(YK-YI)+XK*(YI-YJ))/2.0

C
AA=1.0-POISS
WA=YOUNG*TH/(1.0-POISS*2)
DO 11 J=1,3

```