应用数学和力学，第14卷第2期（1993年2月）
Applied Mathematics and Mechanics

应用数学和力学编委会编
重庆出版社出版

差分方程解的数值余量消元法*

黄 平 董正筑

（中国矿业大学数力系，1990年2月15日）

摘 要

本文提出一种新的消元方法，该法利用数值的直接迭代产生余量方程，从而构成已消去很多未知量的线性方程组．本文的方法具有求解简便、精确和快速的优点．

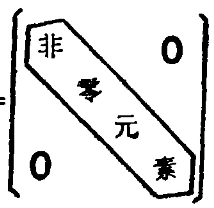
关键词 有限差分法 有限元法 余量方程 直接迭代

一、引 言

差分方法[1]为偏微分方程主要解法之一．采用差分格式，可将某些偏微分方程离散化，并结合边界条件建立“差分有限元”模型，最后转化为求解大型代数方程组

$$\boldsymbol{K}\boldsymbol{x}+\boldsymbol{F}=0 \tag{1.1}$$

上式中，$\boldsymbol{x}$为待求节点未知向量，总刚度矩阵$\boldsymbol{K}$为对角阵，即有

$K=$ 

表示非零元素集中在主对角线两侧的45°斜线内．如果采用线性有限元法求解工程力学（或求解某些偏微分方程），我们最后需要求解含带状稀疏矩阵的线性方程组

$$\tilde{\boldsymbol{K}}\tilde{\boldsymbol{x}}+\tilde{\boldsymbol{F}}=0 \tag{1.2}$$

上式中，$\tilde{\boldsymbol{x}}$为未知节点位移列向量，$\tilde{\boldsymbol{K}}$为总刚度矩阵．由于$\tilde{\boldsymbol{K}}$是由各单元刚度矩阵组装而成，故它有带状、稀疏的性质．

在采用满阵法或在内存进行的等带宽法求解方程(1.1)或(1.2)时，即使在容量很大的计算机上，矩阵阶数很大时，其解题能力也将受到极大的限制[3]．于是产生了各种三角分解解法[2]，等带宽、变带宽等方法[4][5][7]和各种迭代算法[6][7][8]．迭代法一般迭代次数较多，需要机时较多，有时导致迭代的失败[6]，对求解的时间也只能作大概的估计，因为收敛所要求的迭代次数与矩阵$\boldsymbol{K}$的条件数和所用的加速收敛因子是否有效有关[5][7][8]．为了减少迭代次数，减小计算误差，加速迭代过程，本文提出一种大大降低方程阶数，可以得到精确解的新

* 陈至达推荐．

的迭代算法.

二、算法推导

为了说明方便起见，对于方程(1.2)，我们可以通过简单的消元过程，将其化为

$$\boldsymbol{K}\boldsymbol{x}+\boldsymbol{F}=0 \tag{2.1}$$

的形式，其中

$$\boldsymbol{K}=\begin{bmatrix} \text{非} & & & 0 \\ & \text{零} & & \\ & & \text{元} & \\ 0 & & & \text{素} \end{bmatrix}, \qquad \boldsymbol{F}=\begin{Bmatrix} F_1 \\ F_2 \\ \vdots \\ F_n \end{Bmatrix}$$

即$\boldsymbol{K}$中第1行从第1列开始有$m+1$个非零元素，且从第1行第$m+1$列开始向右下方的45°斜线上各元素均不为零(斜线以下可以为零)，如上表所示．零元素在左下角和右上角出现．对于方程(1.1)，则不需作任何简化工作，上述条件是显然满足的.

我们选初始迭代值为

$$x_1=x_1^0,\ x_2=x_2^0,\ \cdots,\ x_m=x_m^0$$

则从x_{m+1}到x_n均可由下列各式迭代而得到

$$\left.\begin{aligned} x_{m+1}^0&=L_1(x_1^0,\ x_2^0,\ \cdots,\ x_m^0)-f_1 \\ x_{m+2}^0&=L_2(x_1^0,\ x_2^0,\ \cdots,\ x_m^0)-f_2 \\ &\ \vdots \qquad\quad \vdots \qquad\quad \vdots \\ x_n^0&=L_{n-m}(x_1^0,\ x_2^0,\ \cdots,\ x_m^0)-f_{n-m} \end{aligned}\right\} \tag{2.2}$$

以上共计$(n-m)$个迭代计算式，式中$f_i(i=1,2,\cdots,n-m)$为(2.1)式中荷载列阵$\boldsymbol{F}$的分量F_i除以x_{m+i}的系数得到的相应值．由于初始迭代值不可能恰好为真值，因此产生了m个余量式，它们在计算机中均以数值显示，即有

$$\left.\begin{aligned} R_1&=F_{n-m+1}+L_{n-m+1}(x_1^0,\ x_2^0,\ \cdots,\ x_m^0) \\ R_2&=F_{n-m+2}+L_{n-m+2}(x_1^0,\ x_2^0,\ \cdots,\ x_m^0) \\ &\ \vdots \qquad\quad \vdots \qquad\quad \vdots \\ R_m&=F_n+L_n(x_1^0,\ x_2^0,\ \cdots,\ x_m^0) \end{aligned}\right\} \tag{2.3}$$

(2.2)、(2.3)式中的$L_i(\cdots)$均为由原方程得到的变量x_1^0，x_2^0，$\cdots$，x_m^0的线性函数，在计算机中同样以数值显示．现将方程(2.3)改写为

$$\left.\begin{aligned} R_1&=F_{n-m+1}+K_{11}x_1^0+K_{12}x_2^0+\cdots+K_{1m}x_m^0 \\ R_2&=F_{n-m+2}+K_{21}x_1^0+K_{22}x_2^0+\cdots+K_{2m}x_m^0 \\ &\ \vdots \qquad\quad \vdots \qquad\quad \vdots \\ R_m&=F_{n-m+2}+K_{m1}x_1^0+K_{m2}x_2^0+\cdots+K_{mm}x_m^0 \end{aligned}\right\} \tag{2.4}$$

当x_1^0，x_2^0，$\cdots$，x_m^0为精确解时，即当$x_1^0=x_1$，$x_2^0=x_2$，$\cdots$，$x_m^0=x_m$时，有

$$\left.\begin{aligned}0&=F_{n-m}+K_{11}x_1+K_{12}x_2+\cdots+K_{1m}x_m\\0&=F_{n-m+1}+K_{21}x_1+K_{22}x_2+\cdots+K_{2m}x_m\\&\vdots\qquad\quad\vdots\qquad\quad\vdots\\0&=F_n+K_{m1}x_1+K_{m2}x_2+\cdots+K_{mm}x_m\end{aligned}\right\}\tag{2.5}$$

将(2.5)式减去(2.4)式，得

$$\left.\begin{aligned}-\Delta R_1&=K_{11}\Delta x_1+K_{12}\Delta x_2+\cdots+K_{1m}\Delta x_m\\-\Delta R_2&=K_{21}\Delta x_1+K_{22}\Delta x_2+\cdots+K_{2m}\Delta x_m\\&\vdots\qquad\quad\vdots\qquad\quad\vdots\\-\Delta R_m&=K_{m1}\Delta x_1+K_{m2}\Delta x_2+\cdots+K_{mm}\Delta x_m\end{aligned}\right\}\tag{2.6}$$

上式中$R_i=\Delta R_i$，且

$$\Delta x_i=x_i-x_i^0\qquad(i=1,2,\cdots,m)\tag{2.7}$$

将(2.6)式改写为

$$\left.\begin{aligned}-\Delta R_1&=\frac{\partial R_1}{\partial x_1}\Delta x_1+\frac{\partial R_1}{\partial x_2}\Delta x_2+\cdots+\frac{\partial R_1}{\partial x_m}\Delta x_m\\-\Delta R_2&=\frac{\partial R_2}{\partial x_1}\Delta x_1+\frac{\partial R_2}{\partial x_2}\Delta x_2+\cdots+\frac{\partial R_2}{\partial x_m}\Delta x_m\\&\vdots\qquad\quad\vdots\qquad\quad\vdots\\-\Delta R_m&=\frac{\partial R_m}{\partial x_1}\Delta x_1+\frac{\partial R_m}{\partial x_2}\Delta x_2+\cdots+\frac{\partial R_m}{x_m}\Delta x_m\end{aligned}\right\}\tag{2.8}$$

写成矩阵形式为

$$-\Delta\boldsymbol{R}=\left[\frac{\partial R_i}{\partial x_j}\right]\Delta\boldsymbol{x}\qquad(i,j=1,2,\cdots,m)\tag{2.9}$$

这里$\Delta\boldsymbol{R}=(\Delta R_1\ \Delta R_2\ \cdots\ \Delta R_m)^T$，$\Delta\boldsymbol{x}=(\Delta x_1\ \Delta x_2\ \cdots\ \Delta x_m)^T$。比较(2.6)与(2.8)两式，有

$$K_{ij}=\partial R_i/\partial x_j\tag{2.10}$$

考察K_{ij}之数学意义：当x_j单独改变，而其余$x_i(i\neq j)$不变时余量的改变量与Δx_j之比值即为K_{ij}之值。

如果我们计算出K_{ij}之值，则将其代入(2.9)式，可算出Δx_i，再代入(2.7)式问题便迎刃而解。现在我们来求K_{ij}。由(2.4)式算出$R_i=\Delta R_i$后，我们将$x_1^0,x_2^0,\cdots,x_m^0$中各量依次按顺序

$$(x_1^0,x_2^0,\cdots,x_m^0)\to(x_1^*,x_2^0,\cdots,x_m^0)\to(x_1^0,x_2^*,\cdots,x_m^0)\to\cdots\to(x_1^0,x_2^0,\cdots,x_m^*)$$

即每次仅改变一个量，并将这些量分m次代入(2.2)和(2.3)式，可算出$m\times m$个余量，它们分别为列向量

$$\begin{Bmatrix}r_1(x_1^*)\\r_2(x_1^*)\\\vdots\\r_m(x_1^*)\end{Bmatrix},\quad\begin{Bmatrix}r_1(x_2^*)\\r_2(x_2^*)\\\vdots\\r_m(x_2^*)\end{Bmatrix},\quad\cdots,\quad\begin{Bmatrix}r_1(x_m^*)\\r_2(x_m^*)\\\vdots\\r_m(x_m^*)\end{Bmatrix}$$

的分量。上面各列向量分量表示该余量是仅由改变括号内变量（即$x_i^0\to x_i^*$，而x_j^0不变，$i\neq j$，$i,j=1,2,\cdots,m$）分别代入(2.2)、(2.3)式算出的。根据(2.10)式的数学意义，显然有

$$K_{ij}=\frac{r_i(x_j^*)-R_i}{x_j^*-x_j^0}\qquad(i,j=1,2,\cdots,m)\tag{2.11}$$

三、计算程序及算例

根据上文的推导，求解方程(2.1)的程序为：

（1） 任选一初始迭代值x_1^0，x_2^0，…，x_m^0，并将其代入(2.2)、(2.3)式，算出余量R_i；

（2） 计算K_{ij}值：

将由第(1)步选定的x_i^0($i=1,2,\cdots,m$)值每次改变其中的一个值，然后分m次代入(2.2)、(2.3)两式，分别算出相应的$m\times m$个$r_i(x_j^*)$ ($i,j=1,2,\cdots,m$)值，再将由第(1)步算出的R_i和$r_i(x_j^*)$值代入(2.11)式算出K_{ij}；

（3） 计算Δx_i和x_i($i=1,2,\cdots,m$)：

将第(1)步算出的$R_i=\Delta R_i$和第(2)步算得的K_{ij}值代入(2.9)式算出Δx_i后再代入(2.7)式，便得到x_i($i=1,2,\cdots,m$)；

（4） 计算x_i($i=m+1$，$m+2$，…，n)各值：

将x_i($i=1,2,\cdots,m$)值代入(2.2)式，依次算出x_i($i=m+1$，$m+2$，…，n)各值。至此，方程(2.1)解毕。

算例 解方程

$$\begin{bmatrix} 1 & -1 & 2 & & & \\ -1 & 2 & 1 & 1 & & \mathbf{0} \\ 2 & 1 & 1 & 2 & 2 & \\ & 1 & 2 & 1 & 1 & 2 \\ \mathbf{0} & & 2 & 1 & 2 & 1 \\ & & & 2 & 1 & 1 \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{Bmatrix} = \begin{Bmatrix} 2 \\ 3 \\ 8 \\ 7 \\ 6 \\ 4 \end{Bmatrix} \tag{3.1}$$

根据(2.2)式，列出表1。这里$m=2$。

表 1

计 算 式	$x_1^0=2$，$x_2^0=3$	$x_1^*=1$，$x_2^0=3$	$x_1^0=2$，$x_2^*=2$
$x_3=0.5(2-x_1+x_2)$	1.5	2	1
$x_4=3+x_1-2x_2-x_3$	−2.5	−4	0
$x_5=0.5(8-2x_1-x_2-x_3-2x_4)$	2.25	4.5	0.5
$x_6=0.5(7-x_2-2x_3-x_4-x_5)$	0.625	−0.25	1.25
$R_1=6-2x_3-x_4-2x_5-x_6$	0.375	$r_1(x_1^*)=-2.75$	$r_1(x_2^*)=1.75$
$R_2=4-2x_4-x_5-x_6$	6.125	$r_2(x_1^*)=7.75$	$r_2(x_2^*)=2.25$

根据(2.11)式，计算K_{ij}：

$$K_{11}=\frac{r_1(x_1^*)-R_1}{x_1^*-x_1^0}=\frac{-2.75-0.375}{1-2}=3.125$$

$$K_{12}=\frac{r_1(x_2^*)-R_1}{x_2^*-x_2^0}=\frac{1.75-0.375}{2-3}=-1.375$$

$$K_{21}=\frac{r_2(x_1^*)-R_2}{x_1^*-x_1^0}=\frac{7.75-6.125}{1-2}=-1.625$$

$$K_{22}=\frac{r_2(x_2^*)-R_2}{x_2^*-x_2^0}=\frac{2.25-6.125}{2-3}=3.875$$

将以上各值代入方程(2.9)，我们得到一个二元一次方程组：

$$\begin{bmatrix} 3.125 & -1.375 \\ -1.625 & 3.875 \end{bmatrix}\begin{Bmatrix} \Delta x_1 \\ \Delta x_2 \end{Bmatrix}=\begin{Bmatrix} -0.375 \\ -6.125 \end{Bmatrix} \tag{3.2}$$

方程(3.2)的解为$\Delta x_1=-1$，$\Delta x_2=-2$，将其代入方程(2.7)，得到$x_1=1$，$x_2=1$，再回代入表中计算式，得到精确解

$$\boldsymbol{x}=(1\ 1\ 1\ 1\ 1\ 1)^T$$

四、计算机程序

根据上文方法，编制的计算机程序，命名为“NRE”，其流程图如图1：

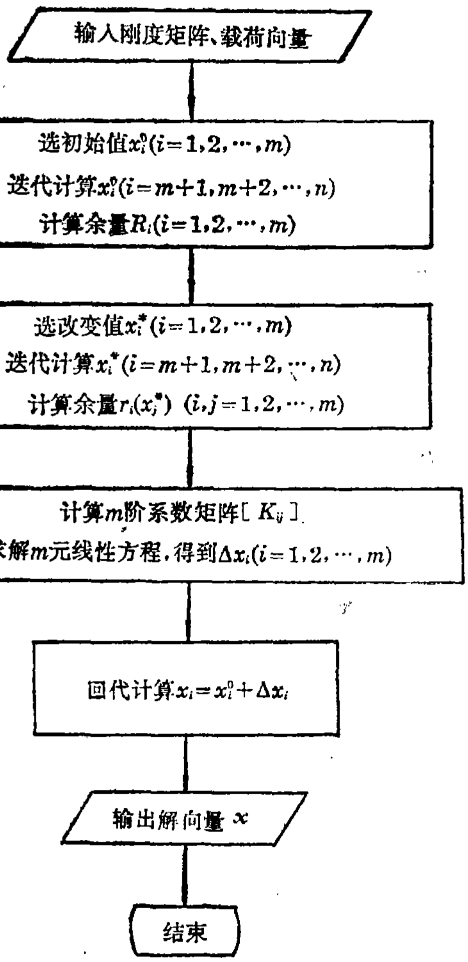


图 1

算例 求解由弹性理论中负荷杆体问题形成的线性代数方程组

$$\boldsymbol{A}\boldsymbol{x}+\boldsymbol{B}=0$$

其中

$$
A=\begin{bmatrix}
5 & -4 & 1 & & & & & \\
-4 & 6 & -4 & 1 & & & \mathbf{0} & \\
1 & -4 & 6 & -4 & 1 & & & \\
 & \cdots & \cdots & & & & & \\
 & & & 1 & -4 & 6 & -4 & 1 \\
 & \mathbf{0} & & & 1 & -4 & 6 & -4 \\
 & & & & & 1 & -4 & 5
\end{bmatrix}
$$

为$n\times n$阶矩阵，本例中取$n=40$计算.

$$B=(-1\ 0\ \cdots\ \cdots\ 0)^T$$

计算结果与文献[6]比较见表2.

表 2

	x_1	x_2	…	x_{39}	x_{40}
文献[6]	13.17072	25.36583	…	13.63413	6.829262
本程序	13.17073	25.36585	…	13.63415	6.829268

文献[6]采用共轭斜量法计算，迭代次数为101次，本程序的实际迭代次数包括回代为$2\times2+2=6$次.

五、结 论

我们将本文的方法称为数值余量消元法，简称NRE (Numerical Residual Elimination)方法. 本文算例说明，若采用NRE法求解方程(2.1)无论划分多少单元，最后只需求解m阶线性方程组，而m与偏微分方程的性质及阶数有关. 与其它算法相比，NRE法具有如下优点：

(1) NRE法兼有迭代算法与直接解法的优点. NRE法采用迭代计算，但却得到精确解，它不存在不收敛问题，计算精度仅由计算机本身精度所引起.

(2) 对于对角带状矩阵，迭代次数明显地大大少于其它迭代算法，而且迭代的次数确定，为$m\times m+2$次. 因此可以极大地提高运算速度，这对于解算大型差分方程和利用微机解题尤其具有优越性.

本文得到中国矿业大学陈至达教授的指导与帮助，在此一并致谢.

参 考 文 献

[1] 冯康，《数值计算方法》，国防工业出版社 (1987)，492.

[2] 李大潜等，《有限元法续讲》，科学出版社 (1978)，135—175.

[3] Cheung, Y.K. and M.F. Yeo, *A Practical Introduction to Finite Element Analysis*, 中译本，《有限单元法实用导论》，罗松发等译，广西人民出版社 (1982)，112.

[4] 南京大学数学系计算数学专业编，《偏微分方程数值解法》，科学出版社 (1979)，171—227.

[5] Bathe, K.J. and E.L. Wilson, *Numerical Methods in Finite Element Analysis*, Prentice Hall, Inc. (1976)，中译本，《有限单元分析中的数值方法》，林公豫、罗恩译，科学出版社 (1987)，293—307.

[6] 中国科学院沈阳计算技术研究所等编，《电子计算机常用算法》(1976)，144—150.

[7] Bathe, K.J., Nonlinear finite element analysis and ADINA, *J. Computers and Structures*, 13(5/6) (1981).

[8] Wunderlich, W., E. Stein and K.J. Bathe. (Eds.), *Nonlinear Finite Element Analysis in Structural Mechanics*, Springer-Verlag, Berlin (1981).

Numerical Residual Elimination Method of Finite Differential Equations

Huang Ping   Dong Zheng-zhu

(*Chinese University of Mining and Technology, Xuzhou*)

Abstract

In this paper, a new elimination of finite differential equations has been discussed. It applies the numerical direct iteration to obtain the residual equations, in which the number of unknowns has been reduced greatly. The solution process is simple and efficient, and the solution is exact.

Key words finite differential method, finite element method, residual equations, direct iteration