

有限元语言

梁国平 著



科学出版社

www.sciencep.com

12

有限元语言

梁国平 著



科学出版社

北京

内 容 简 介

有限元语言是一种适用于有限元方法求解偏微分方程的模型语言. 采用有限元语言编程就是书写偏微分方程和算法, 然后由生成器产生全部 FORTRAN 语言的有限元程序. 本书的主要内容包括: 微分方程表达式、单物理场算法和多场耦合有限元算法的描述语言; 元件化程序设计方法; 有限元的数据结构; 形函数库、微分算子库、单物理场算法库等.

本书可作为高等院校计算数学、应用数学、计算机等理工类专业的高年级本科生和研究生的参考教材, 也可供相关专业的工程和技术人员参考.

图书在版编目(CIP)数据

有限元语言/梁国平著. —北京: 科学出版社, 2008

ISBN 978-7-03-023080-5

I. 有… II. 梁… III. 有限元法 IV. O241.82

中国版本图书馆 CIP 数据核字(2008) 第 150324 号

责任编辑: 赵彦超 / 责任校对: 陈玉凤

责任印制: 赵德静 / 封面设计: 王 浩

科 学 出 版 社 出 版

北京东黄城根北街 16 号

邮政编码: 100717

<http://www.sciencep.com>

双 青 印 刷 厂 印 刷

科学出版社发行 各地新华书店经销

*

2009 年 1 月第 一 版 开本: B5(720×1000)

2009 年 1 月第一次印刷 印张: 13 3/4

印数: 1—2 500 字数: 260 000

定价: 39.00 元

(如有印装质量问题, 我社负责调换〈双青〉)

前 言

有限元语言是一种采用有限元方法和有限体积法求解偏微分方程的模型语言,采用这种语言编写有限元和有限体积程序,其主要工作就是书写微分方程表达式及其有限元/有限体积算法,然后由该语言的生成器(生成系统)自动产生某种高级语言(如 FORTRAN,C 等)的有限元/有限体积计算程序。

采用有限元语言编程,其代码量比生成器产生的高级语言程序的代码量减少 90%以上,因此能极大地提高人的编程效率。并且由于采用有限元语言编程,就是写微分方程表达式和算法,所以容易阅读、修改和维护。

有限元语言为应用有限元和有限体积法的广大工程师和科技人员免去了繁琐重复的编程劳动,从根本上解决了长期以来难以解决的软件的再用性问题。

有限元语言是一种超平台的语言,它既不依赖操作系统,也不依赖计算机,既适用于单 CPU 的串行机,也适用于多 CPU 的并行机,并且用户无需了解并行程序的编写。有限元语言适用于各学科领域及各种有限元问题,到目前为止,尚未遇到不能采用该语言求解的有限元问题。

在有限元语言及其生成器中,我们采用了三项软件技术:自动生成技术、元件化技术及公式库技术。自动生成技术是生成器的主要技术。元件化技术与现今流行的组件化、中间件等软件技术相似,但是由于采用了自动生成技术,所需的元件程序很少,对于单 CPU 的串行机通常只需五个,因为这些元件程序可以根据微分方程表达式和算法自动改变,所以几个元件程序就能满足各种有限元问题的需要,从而大大提高了软件的可读性和可维护性。有限元语言的一大特点是采用公式库而不是程序库,公式库技术可大大减少用户对各种公式和算法的书写。

本书共有六章。第 1 章是微分方程表达式的描述语言,给出有限元和有限体积法书写微分方程表达式的具体形式;第 2 章给出单物理场描述有限元算法的具体形式;第 3 章给出多场耦合有限元算法的书写方式;第 4 章为元件化程序设计方法及五个常用的元件程序;第 5 章给出有限元数据结构的描述方式;第 6 章是公式库,包括形函数库、微分算子库和算法库等。

借此机会,感谢中国科学院数学与系统科学研究院数学研究所杨乐院士等諸多领导与同事,他们的支持使我在有限元语言研究过程中克服了种种困难。在宣传和推广有限元语言应用的过程中,他们的支持给了我鼓舞和力量。

此项研究工作获得了国家自然科学基金和 863 项目的支持,在此一并感谢。

有限元语言的初期用户,如美国加州大学伯克萊分校王其允教授、德国慕尼黑科技大学霍夫曼教授、香港中文大学邹军教授等曾为此项研究提出宝贵的意见和建议,在此表示衷心的感谢。

我特别要感谢以赵宇波博士 (现任中国科学院深圳先进技术研究院副研究员) 为代表的北京火箭软件公司的年轻人, 他们为本书早日面世做出了积极的努力.

作 者

2006 年 6 月于北京

目 录

前言

第 1 章 微分方程表达式的描述语言	1
1.1 PDE 文件的填写	1
1.1.1 DEFI 信息段	2
1.1.2 FUNC 信息段	5
1.1.3 STIF 信息段	6
1.1.4 MASS 信息段	7
1.1.5 DAMP 信息段	8
1.1.6 LOAD 信息段	8
1.1.7 如何插入 FORTRAN 源程序	9
1.1.8 例题	12
1.2 CDE 文件的填写	13
1.2.1 DEFI 信息段	13
1.2.2 FUNC 信息段	15
1.2.3 STIF 信息段	15
1.2.4 MASS 信息段	16
1.2.5 DAMP 信息段	17
1.2.6 LOAD 信息段	17
1.2.7 如何插入 FORTRAN 源程序	18
1.2.8 例题	18
1.2.9 如何由 CDE 文件生成 PDE 文件	20
1.3 VDE 文件的填写	20
1.3.1 向量和矩阵说明语句	20
1.3.2 array 说明语句	21
1.3.3 张量运算表达式	21
1.3.4 例题	23
1.3.5 如何由 VDE 文件生成 PDE 文件	25
1.4 FDE 文件的填写	25
1.4.1 FDE 文件的书写形式	25
1.4.2 FVECT 与 FMATR 语句	26
1.4.3 @l 命令语句	26
1.4.4 @a 命令语句	26

1.4.5	@w 命令语句	27
1.4.6	@s 命令语句	27
1.4.7	@r 命令语句	27
1.4.8	例题	28
1.4.9	如何由 FDE 文件生成 PDE 文件	29
1.5	FBC 文件的填写	29
1.6	GES 文件的填写	30
1.6.1	GES 文件结构	30
1.6.2	GES 文件的编写规则	31
1.6.3	例题	47
1.6.4	GES 文件和单元子程序	49
1.7	GLT 文件的填写	51
1.7.1	GLT 文件的编写规则	51
1.7.2	由 GLT 文件生成单元子程序	54
1.8	有限体积法文件的填写	54
1.8.1	GVS 文件结构	55
1.8.2	GVS 文件的编写规则	55
1.8.3	FVS 文件的编写规则	57
1.8.4	例题	60
第 2 章	单物理场算法的描述语言	64
2.1	NFE 文件结构	64
2.2	NFE 文件的编写规则	65
2.2.1	DEFI 信息段	65
2.2.2	COEF 信息段	65
2.2.3	EQUATION 信息段	66
2.2.4	SOLUTION 信息段	68
2.2.5	插入 FORTRAN 程序	70
2.2.6	END 信息段	70
第 3 章	多场耦合有限元算法的描述语言	71
3.1	GCN 文件的填写	71
3.1.1	填写方式	71
3.1.2	举例说明	73
3.2	GIO 文件的填写	73
3.2.1	填写方式	73
3.2.2	举例说明	74
3.3	例: 热固耦合问题	74

第 4 章 元件化程序设计方法	79
4.1 有限元程序结构与元件化程序设计方法	79
4.1.1 程序结构	79
4.1.2 元件化程序设计方法	80
4.2 五个元件程序	85
4.2.1 START 元件程序	85
4.2.2 BFT 元件程序	93
4.2.3 E 元件程序	100
4.2.4 SOLV 求解器	113
4.2.5 U 元件程序	136
第 5 章 有限元的数据结构	142
5.1 有限元计算的输入数据组成简述	142
5.1.1 输入数据形式	142
5.1.2 表格文件的读写格式	142
5.2 单场问题的有限元输入数据	143
5.2.1 坐标数据表格	143
5.2.2 节点规格数表格	143
5.2.3 指定节点位移和节点荷载信息表格	144
5.2.4 初始值表格	144
5.2.5 单元信息数据	145
5.3 有限元输入数据的显示和查询	145
5.4 PRE 文件及其自动生成与修改	146
5.4.1 线性稳态例子	147
5.4.2 非线性瞬态例子	150
5.4.3 多场耦合例子	156
5.4.4 PRE 文件的自动生成与修改	162
5.5 计算结果的图形显示: POS 文件	163
第 6 章 形函数库、算子库、NFE 算法库	164
6.1 形函数库	164
6.2 数值积分	164
6.3 算子库	164
6.3.1 基本算子	165
6.3.2 常用程序段	166
6.4 NFE 算法库	166
6.4.1 求解椭圆问题的算法文件	166
6.4.2 求解抛物问题的算法文件	168
6.4.3 求解波动问题的算法文件	174

6.4.4	计算梯度的最小二乘法算法文件 STR.NFE	184
6.5	GCN 库	185
附录 A	插值函数与单元类型	186
A.1	一维 Lagrange 单元	186
A.2	二维单元	188
A.3	三维单元	192
附录 B	等参单元	197
附录 C	数值积分	204
C.1	高斯积分	204
C.2	节点积分	206
附录 D	有限元语言术语总汇	209
附录 E	有限元语言关键字总汇	211

第 1 章 微分方程表达式的描述语言

微分方程的描述是有限元语言的一个重要的基本内容. 我们采用了一个 PDE 类文件表示一个基于弱解形式的微分方程表达式. 此类文件包括 GES, PDE, CDE, VDE, FDE, FBC 等六种文件, 分别属于不同层次的有限元语言文件. 由这些文件, 单元子程序自动生成系统自动生成计算单元刚度矩阵、单元阻尼矩阵和单元荷载向量等的单元子程序.

GES 文件是最基础最底层的文件, 该文件给出有限元方法的全部公式, 如形函数、数值积分等公式. 其他文件都要先生成该文件, 然后, 再由该文件生成单元子程序. PDE 文件可通过公式库取得形函数及数值积分的公式, 避免了这些公式的书写.

CDE, VDE 和 FDE 文件是为了节省用户在书写偏微分方程表达式时所花费的时间而设计的. CDE 文件的主要内容是复变量微分方程表达式, VDE 文件的主要内容是微分方程张量表达式, FDE 文件是可利用算子公式库与算子表达式的微分方程张量表达式文件.

FBC 文件的主要内容是边值条件 (第二类和第三类边值条件) 的微分方程表达式. 由这个文件, 单元子程序自动生成系统自动生成计算边值条件的单元刚度矩阵、阻尼矩阵和荷载向量等子程序. 当问题有第二类或第三类边值条件时填写这种文件. 下面分节介绍这些文件的填写.

1.1 PDE 文件的填写

PDE 文件的主要内容是填写微分方程表达式, 由此文件自动生成计算单元刚度矩阵、单元质量矩阵、单元阻尼矩阵和单元荷载向量等单元子程序. PDE 文件结构如下:

(1) 用户最多需填写 6 段信息, 它们分别以 DEFI, FUNC, STIF, MASS, DAMP, LOAD 为信息段关键字. 用户可根据不同情况, 按上述顺序依次填写各段信息.

本系统规定, 每一段信息 (以下简称信息段) 均以它的关键字开头, 以空行结束. 所有信息段填写完毕后, 还需填写结束字 END, 以表示结束.

(2) 本系统允许用户插入自己编写的 FORTRAN 源程序, 即以 “\$C6”、“\$C0” 和 “\$CV” 字符引导的方式在指定的某些信息段内插入. 另外, 本系统还允许用户在结束字 END 之后附加 FORTRAN 源程序, 即以关键字 FORT 开头, 下面信息便是 FORTRAN 源程序. 关于插入与附加 FORTRAN 源程序的方式, 我们将在以后的章节专门讲述.

(3) 本系统规定字符 “\” (反斜杠) 表示续行号. 续行号 “\” 后面的字符将被本系统跳过, 下一行将为本行的续行. 因此, 反斜杠 “\” 又可作注解行用, 用户可以在 PDE 文件中任意位置加注解行, 此行首字符是 “\”.

下面将逐节讲述各个信息段的填写方式, 每节后面均附有该信息段的填写例子说明, 我们将在 1.1.8 节中以泊松 (Poisson) 方程为例, 具体说明如何填写一个实际问题的 PDE 文件.

1.1.1 DEFI 信息段

PDE 文件的 DEFI 信息段最多需要 10 行信息, 它的一般方式如下:

DISP = 未知函数名, 未知函数名, ..., 未知函数名

COOR = 坐标变量名, 坐标变量名, ..., 坐标变量名

COEF = 非线性系数函数名, 非线性系数函数名, ..., 非线性系数函数名

FUNC = 自定义函数名, 自定义函数名, ..., 自定义函数名

MATE = 材料参数名; ...; 材料参数名; 材料参数缺省值; ...; 材料参数缺省
值

SHAP = 单元几何形状类型符, 单元节点数

GAUS = 单元几何形状类型符或高斯积分点数

MASS = 单元几何形状类型符, 质量密度

DAMP = 单元几何形状类型符, 阻尼系数

LOAD = 表达式; 表达式; ...; 表达式

函数名、变量名和材料参数名可由用户自由定义.

下面分别讲述各行填写的信息及其含义.

DISP 右边空一格 (或等号) 后填写需要求解的微分方程的未知函数名. 可以有任意多个未知函数名, 每个未知函数名对应一个未知函数, 各未知函数名之间用空格、逗号或分号隔开.

COOR 右边空一格 (或等号) 后填写坐标变量名, 可以有任意多个, 每个坐标变量名对应一个坐标变量. 坐标变量名之间用空格、逗号或分号隔开.

COEF 右边空一格 (或等号) 后填写系数变量名, 可以有任意多个, 每个系数变量名之间用空格、逗号或分号隔开. 它们分别对应微分方程中的一个变系数, 如果没有系数变量则此行不填写, 例如对非线性问题或者耦合问题, 需要填写此行.

FUNC 右边空一格 (或等号) 后填写用户自己定义的函数名, 可以定义任意多个函数, 每个函数名对应一个用户自定义函数, 各函数名之间用空格、逗号或分号隔开. 本系统提供函数替换功能, 此行信息用于定义函数名字, 后面必须有 FUNC 信息段与之对应, 在那里具体定义各函数名下的函数表达式, 它之后的各信息段中相同的表达式均可以用相应的自定义函数名来代替. 这样做的好处是可以使得过分繁冗复杂的信息填写变得简洁, 便于阅读和修改. 注意, 该 FUNC 行信息必须与后面的 FUNC 信息段配合使用. 它们的关键字虽然一样, 但是该 FUNC 行信息只是 DEFI 信息段中的一行信息, 它的主要内容都在 FUNC 信息段中填写, 不要混淆了

它们的作用. 如果虚功方程式并不复杂繁冗, 也可以不填写此行和后面的 FUNC 信息段.

MATE 右边空一格 (或等号) 后填写材料参数名, 可以有任意多个, 材料参数名之间用空格或分号隔开, 注意这里不能用逗号. 后面花括号 (也可不用花括号) 中填写材料参数对应的缺省值, 之间用空格或分号隔开, 注意, 这里不能用逗号, 也可以不写材料参数缺省值, 此时, 系统取缺省值为 0. 这些材料参数名可以在其他信息段中使用. 增加 MATE 行后, 还可以在其他信息段中使用变量名 time, dt, imate, ielem 和 nelem, 分别表示当前时刻 (time)、时间步长 (dt)、材料号 (imate)、单元编号 (ielem) 和单元总数 (nelem).

SHAP 右边空一格 (或等号) 后填写单元类型符和单元节点数, 例如, Q, 4 表示双线性四边形单元. 单元类型符规定如下: Q 表示四边形; T 表示三角形; C 表示六面体; W 表示四面体和五面体; L 表示线.

GAUS 右边空一格 (或等号) 后填写单元类型符或一个方向上的高斯积分点的数目. 当采用节点积分的时候, 就填写单元类型符; 当采用高斯积分的时候, 就填写一个方向上的高斯积分点的数目. 例如, 如果是四边形单元, 填写 2 就表示有 2*2 个高斯积分点.

MASS 右边空一格 (或等号) 后填写单元类型符和单元质量密度. 这里, 单元类型符的规定与 SHAP 行的规定相同, 单元质量密度是 FORTRAN 语言表达式, 可以是 FORTRAN 语言规定的标准函数, 也可以是用户自己定义的函数. 本行反映了微分方程对时间的导数项, 所以仅对抛物方程和波动方程才有此行. 本行的目的是给出质量矩阵 MASS 的表达式, 在这里只允许采用集中的方式. 单元质量密度个数可以是未知函数的个数, 若只写一个, 表示所有的单元质量密度都取同一个值, 如不填写单元质量密度, 系统将取单元质量密度为 1.0.

DAMP 右边空一格 (或等号) 后填写单元类型符和单元阻尼系数. 这里单元类型符规定与 SHAP 行的规定相同, 单元阻尼系数是 FORTRAN 语言表达式, 可以是 FORTRAN 语言规定的标准函数, 也可以是用户规定的函数. 本行的目的是给出波动方程的阻尼矩阵, 仅对波动方程才有此行, 在这里只允许采用集中的方式. 单元阻尼系数个数可以是未知函数的个数, 若只写一个, 表示所有的单元阻尼系数都取同一个值, 如不填写单元阻尼系数, 系统将取单元阻尼系数为 1.0.

LOAD 右边空一格 (或等号) 后填写 FORTRAN 语言表达式, 可以是 FORTRAN 语言规定的标准函数, 也可以是用户自己定义的函数, 表达式如果有多项或者是有加减号的一项, 此表达式要用括号括起来. 表达式的个数是未知函数的个数. 本行的目的是给出单元荷载即微分方程右端项的表达式.

注1.1 (1) 在填写 COEF 信息行时, 必须与 NFE 文件 (见 2.2 节) 中的 COEF 信息段保持一致.

(2) PDE 文件后面用到的变量或函数名及坐标变量名必须在 DEF1 信息段中规定.

(3) PDE 文件中 SHAP 行可以使用%1 和%2 的待定参数形式, 分别表示单元类型符和单元节点数, 它们将由 PRE 文件给出. 当执行系统命令 PDE 生成单元子程序时, PDE 文件中 SHAP 段要给出具体的单元类型符和节点数.

(4) DISP 行定义的未知函数如果有多个, 那么每个函数都可以采用不同的插值函数. 具体做法是定义多个 SHAP 行. 比如, 对于三个自由度 U, V, P 的问题, 如果 U, V 采用四边形九节点双二次插值, P 采用四节点双线性插值, 那么 SHAP 行的定义可以如下:

```
SHAP Q 9
```

```
SHAP Q 4 P 1 2 3 4
```

这里, 1, 2, 3, 4 表示我们采用九节点的前四个节点 (即四个顶点) 作为 P 的插值点. 事实上, 九节点单元的节点编号顺序是: 四个顶点、四边中点、中心.

(5) COEF 行定义的已知函数如果有多个, 那么每个函数都可以采用不同的插值函数. 具体做法是定义多个 SHAP 行. 比如, 对于三个已知 UN, VN, PN 的问题, 如果 UN, VN 采用四边形九节点双二次插值, PN 采用四节点双线性插值, 那么 SHAP 行的定义可以如下:

```
SHAP Q 9
```

```
SHAP Q 4 PN 1 2 3 4
```

其他说明与前面相同.

(6) PDE 文件中 GAUS 行可以使用%3 的待定参数形式, 表示数值积分方式, 将由 PRE 文件规定. 当执行系统命令 PDE 生成单元子程序时, PDE 文件中 GAUS 段要给具体的值.

这里及以后每节的例子都只是为了示范前面讲述的信息段填写方式, 我们将在 1.1.8 节结合一个实际问题具体描述 PDE 文件的总体填写风格和特点.

例 1.1

```
DEFI
```

```
DISP u, v
```

```
COOR x, y
```

```
COEF ua, va, fu, fv, gx, gy
```

```
FUNC funa, funb, func
```

```
MATE em; ed; eqx; eqy; 2.5e-7; 1.0e-6; 0.0; 0.0;
```

```
SHAP %1, %2
```

```
GAUS %3
```

```
MASS %1; em
```

```
DAMP %1; ed
```

```
LOAD eqx; eqy
```

```
空一行
```

说明 在例 1.1 中, DISP 行定义了两个未知函数 u 和 v , 由此知道该问题是一个微分方程组, 两个未知函数 u 和 v 需同时求解. COOR 行给出坐标系的坐标

变量名.

从 COEF 行知道, 此问题为一非线性问题, 需要用到上一个迭代步或上一个时间步计算得到的数值结果. 此时相应的 NFE 文件中的 COEF 信息段可能如下填写: COEF w, fu, fv, gx, gy. 对应向量 w , 这里 COEF 行填写为 ua, va. fu 和 fv 是与未知函数 u 和 v 没有直接关系的系数变量, 可照搬过来.

从 FUNC 行知道, 用户定义了三个函数, 它们的表达式将在后面的 FUNC 信息段给出.

MATE 行说明此问题定义了四个材料参数, 分别为 em, ed, eqx 和 eqy, 其缺省值分别为 $2.5e-7$, $1.0e-6$, 0.0 和 0.0.

SHAP 行采用两个待定参数%1, %2, 分别表示单元类型符和单元节点数, 它们将在 PRE 文件中给出. GAUS 行采用待定参数%3, 表示单元积分方法, 亦由 PRE 文件规定.

MASS 行说明取单元质量密度为 em 的质量矩阵, DAMP 行说明取单元阻尼系数为 ed 的阻尼矩阵.

LOAD 行说明对未知函数 u 的载荷是 eqx, 未知函数 v 的载荷是 eqy.

1.1.2 FUNC 信息段

本段信息的目的是给出用户自定义函数的具体表达式 (它们是未知函数及其导数的线性组合), 以便简化 STIF, MASS, DAMP 和 LOAD 信息段的填写. 需要定义的函数名必须在 DEFI 信息段的 FUNC 行给出. 为了叙述方便, 假定 FUNC 行定义了 m 个自定义函数名, 一般填写方式如下:

FUNC

下面一段信息重复 m 次 (包括空一行)

自定义函数名 = \pm [未知函数或其导数]* 表达式

\pm [未知函数或其导数]* 表达式

...

\pm [未知函数或其导数]* 表达式

空一行

这里的“未知函数”是指在 DISP 行规定的未知函数名, 导数是指对原坐标系, 即 COOR 行定义的坐标变量的导函数. 其一阶导数及二阶导数的一般表达方式

未知函数名/坐标变量名

未知函数名/坐标变量名, 坐标变量名

这里的坐标变量名是指 COOR 行给出的变量名.

“表达式”是 FORTRAN 语言表达式, 可以是 FORTRAN 语言规定的标准函数, 也可以是用户自己定义的函数. “表达式”中不允许出现未知函数或其导数, 以下各信息段中的“表达式”都必须遵守此规定.

表示符{./}也可以作为表达式的变量,其中符号{./}一般是{系数变量名/坐标变量名},表示前者对后者的偏导数.例如: x, y 是 COOR 行定义的原坐标变量名, ua 是 COEF 行定义的系数变量名,则{ ua/x }表示 $\frac{\partial ua}{\partial x}$.

本信息段可以占有任意多行,但每行开头的两个字符必须是“±[”.填写方式中的符号“]”后面的乘号“*”也可以是除号“/”.

用户自定义函数与 DISP 行定义的未知函数一样,可以用在下面的信息段中,但是不能用它的导数,这一点与后者是不同的.

例1.2 按照 DEFIN 信息段 FUNC 行规定的符号, FUNC 信息段可以填写如下:

```

FUNC
funa=+[u/x]*{ua/x}
空一行
funb=+[v/y]*{va/y}
空一行
func=+[u/y]+[v/x]+[u/x]*{ua/x}+[v/y]*{va/y}
空一行

```

1.1.3 STIF 信息段

STIF 信息段,连同后面的 MASS, DAMP 和 LOAD 信息段,共同反映了微分方程表达式.本信息段的目的是给出计算虚功方程表达式的刚度矩阵.本段信息是不可缺少的,它的一般方式如下:先写关键字 STIF,然后按照如下方式填写:

```

DIST=±[未知函数或其导数;未知函数或其导数]*表达式
±[未知函数或其导数;未知函数或其导数]*表达式
...
±[未知函数或其导数;未知函数或其导数]*表达式
空一行

```

其中关键字 DIST 表示刚度矩阵为分布矩阵.本系统规定刚度矩阵不允许采用集中矩阵,因此必须使用关键字 DIST,而不能使用 LUMP.从关键字 STIF 开始至空行结束之间可以占有任意多行,但中间不允许有空行出现,并且换行后必须以符号“+ [”或“- [”开头.符号“]”后面的“*”号也可以是“/”号.

这里的未知函数是指在 DEFIN 信息段中 DISP 行规定的未知函数名或 FUNC 行规定的函数名,但是后者不能用其导数方式.这里的导数是指 DISP 行规定的未知函数对原坐标系坐标变量(即 COOR 定义的坐标变量)的导数,它的一般方式与 1.1.2 节 FUNC 信息段定义的导数方式相同.这里的表达式与 1.1.2 节 FUNC 信息段的“表达式”规定相同.

这里的 [未知函数或其导数;未知函数或其导数] 表示内积,分号“;”前面的“未知函数或其导数”应理解为虚功方程中的未知函数或其导数.分号“;”后面的“未知函数或其导数”应理解为虚功方程中的虚位移或其导数.本系统规定未知

函数与它的虚位移取同一个名字, 不另外规定虚位移名. 两个配对的方括号 “[” 和 “]”, 应理解为它们的内积. 一般来说分号前和分号后的表达式交换位置后是不相等的 (即不对称). 例如 $[u/x; u/y] \neq [u/y; u/x]$.

例1.3 拉普拉斯方程

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0$$

的广义解表达式如下

$$\int_{\Omega} \left(\frac{\partial u}{\partial x} \frac{\partial \delta u}{\partial x} + \frac{\partial u}{\partial y} \frac{\partial \delta u}{\partial y} \right) d\Omega = \int_{\Gamma} \left(\frac{\partial u}{\partial x} n_x \delta u + \frac{\partial u}{\partial y} n_y \delta u \right) d\Gamma$$

其刚度矩阵表达式可如下填写:

STIF

DIST=[u/x; u/x]+[u/y; u/y]

空一行

1.1.4 MASS 信息段

在有限元方法中, 一般来说, MASS 信息段反映了微分方程对时间的导数项, 所以仅对抛物方程和波动方程才有此信息段. 本信息段和 DEFI 信息段中的 MASS 行, 用户只要填写一个. 本信息段的目的是给出质量矩阵 MASS 的表达式, 在这里采用分布的方式.

分布质量矩阵的一般表达式为:

MASS

DIST=±[未知函数或其导数; 未知函数或其导数]*表达式

±[未知函数或其导数; 未知函数或其导数]*表达式

...

±[未知函数或其导数; 未知函数或其导数]*表达式

空一行

填写方式与 STIF 信息段完全相同.

本信息段质量矩阵的填写方式均可以占有任意多行, 但换行后必须以 “±[” 这两个字符之一开头.

例1.4 对抛物型方程

$$A \frac{\partial u}{\partial t} + Lu = f$$

MASS 信息段可写为

MASS

DIST=[u; u]*A

空一行

其中, 分号前的 u 可理解为 $\frac{\partial u}{\partial t}$, 分号后的 u 表示虚位移.

1.1.5 DAMP 信息段

本信息段的目的是给出波动方程的阻尼矩阵, 仅对波动方程才有此信息段. 它和 MASS 信息段一样采用分布方式. 填写方式与 STIF 信息段完全相同. 本信息段和 DEFI 信息段中的 DAMP 行, 用户只要填写一个, 如果都不填写, 就意味着取单元阻尼系数为零的阻尼矩阵. 但注意, 本系统规定它必须与 MASS 信息段同时使用. 也就是说, 填写 DAMP 信息段时, 也要填写 MASS 信息段, 或者在不填写 DAMP 信息段而填写 DEFI 信息段 DAMP 行时, 也要填写 DEFI 信息段 MASS 行.

例1.5 对于波动方程

$$A \frac{\partial^2 u}{\partial t^2} + C \frac{\partial u}{\partial t} + Lu = f$$

MASS 和 DAMP 两个信息段有如下的填写方式:

MASS

DIST=[u; u]*A

空一行

DAMP

DIST=[u; u]*C

空一行

或者填写 DEFI 信息段的 MASS 行和 DAMP 行如下:

MASS %1 A

DAMP %1 C

1.1.6 LOAD 信息段

本信息段的目的是给出单元荷载即方程右端项的表达式, 本信息段和 DEFI 信息段中的 LOAD 行, 用户只要填写一个. 它的一般方式为:

LOAD=±[未知函数或其导数]*表达式

±[未知函数或其导数]*表达式

...

±[未知函数或其导数]*表达式

空一行

其中“未知函数或其导数”的定义与 STIF 信息段的规定相同, 这里的未知函数应理解为其相应的虚位移. 符号“]”后的乘号“*”也可以是除号“/”.

此段信息可以占有任意多行, 但换行后必须以“+ [”或“- [”两个符号开头. 其中“表达式”与 STIF 信息段的规定完全相同.

例1.6 按照 DEFI 信息段的符号规定, LOAD 信息段例如可填写如下:

LOAD=-[u]/2*10+[v]/2*10

-[funa]*(fu+fv*2)

-[funb]*(fu-fv)