

高等学校教材

热结构分析有限元程序设计

陆 山 编著

西北工业大学出版社

【内容简介】 本书较系统地介绍了结构化程序设计技术、平面结构网格自动生成技术、二维及轴对称结构稳态温度场及热弹性问题分析有限元公式、有限元总体方程组求解波前法及其有限元程序的框图和程序设计技术,并配有完整的源程序,同时对有限元分析中网格划分对数值解精度的影响、有限元分析模型的建立、有限元程序运行中故障的排除等问题进行了阐述。它是一本集有限元法理论、程序设计技术及有限元程序使用方法的实用性教科书。

本书可作为航空宇航推进理论与工程学科研究生“热结构分析有限元程序设计”专业课教材,还可作为机械工程、材料成型、燃气轮机、工程力学、软件设计等专业研究生专业选修课教材或教学参考书,也可供相关专业科技工作者和工程技术人员参考。本书提供的程序系统可作为从事有限元法研究及其程序设计的研究生及工程技术人员的软件开发平台。

图书在版编目 (CIP) 数据

热结构分析有限元程序设计/陆山编著. —西安:西北工业大学出版社,2003.6
ISBN 7-5612-1632-7

I. 热… II. 陆… III. 热应力—结构分析—有限元法—程序设计—研究生—教材
IV. O414.1-39

中国版本图书馆 CIP 数据核字 (2003) 第 028888 号

出版发行:西北工业大学出版社

通信地址:西安市友谊西路127号 邮编:710072 电话:(029)8493844

网 址:www.nwpup.com

印刷者:西安东江印务有限公司

开 本:787 mm×1 092 mm 1/16

印 张:10.75

字 数:259千字

版 次:2003年8月第1版 2003年8月第1次印刷

印 数:1~2 000册

定 价:14.00元

前 言

本书主要是为了航空宇航推进理论与工程学科的研究生专业课“热结构分析有限元程序设计”编写的。自 20 世纪 80 年代末开设热结构分析有限元程序设计课程以来,一直缺少一本既介绍结构温度场又介绍结构热应力数值分析方法及其有限元程序设计内容的合适教材。本书是在作者十余年从事热结构分析有限元程序设计研究生课教学内容及授课讲义的基础上改编而成,其中融入了作者的相关科研成果与结构温度场及应力有限元分析专用程序软件。

本书围绕二维及轴对称结构稳态温度场、热弹性问题有限元法及其程序设计相关内容展开。全书共分 9 章,第 1 章为引言,简要介绍有限元法的概念、发展概况及与其他数值分析方法不同的特点;第 2 章简要介绍结构化程序设计方法;第 3 章介绍二维及轴对称体热弹性有限元分析程序设计的一般要求、程序特点和二维及轴对称体热弹性有限元分析程序主程序;第 4 章介绍结构自动离散技术、网络划分对有限元数值解精度的影响及其程序设计;第 5 章介绍有限元单元刚度方程式、性质、单元刚度系数矩阵生成及其故障诊断子程序设计;第 6 章介绍求解有限元线性方程组的波前法原理、程序设计技术及波前法子程序;第 7 章介绍单元应力、应变计算有限元公式及其程序设计;第 8 章介绍有限元应力、应变场修匀方法、公式及其程序设计,并分析了用有限元程序进行应力计算过程中的主要故障及其可能产生的原因;第 9 章推导了二维及轴对称稳态温度场分析有限元公式,并介绍了二维及轴对称结构稳态温度场分析有限元程序。为配合课程教学及上机实践,把本书各章介绍的子程序综合在一起,便构成完整的二维及轴对称结构稳态温度场分析及热弹性分析有限元法程序系统,源程序采用 FORTRAN 77 计算机语言编写。本书各章还配备了习题以便练习。

本书不仅注重有限元法原理及其相关数值分析方法的基本理论学习,同时强调有限元程序设计的基本技能,以及用有限元程序分析工程问题的实用技术的掌握与实践性环节训练。

本书作为教材,计划需要 40 学时的授课时间,并配备约 10 学时左右的上机实践课时。学习本课程的学生应具备有限元法、FORTRAN 77 计算机语言、线性代数、计算方法课程的基本理论知识。

西安石油大学副校长屈展教授仔细审阅了本书初稿,并提出了许多宝贵的建议和意见,在此表示衷心的感谢。

本书错误及不妥之处,敬请读者批评指正。

编著者

2002 年 11 月

目 录

第 1 章 引言	1
第 2 章 结构化程序设计概论	4
§ 2.1 算法结构及数据结构	4
§ 2.2 结构化程序设计技术与方法	5
习题	9
第 3 章 二维及轴对称体热弹性有限元分析程序设计	10
§ 3.1 二维及轴对称体热弹性有限元分析程序功能及特点	10
§ 3.2 主程序 EA2 的功能	10
§ 3.3 主程序 EA2 程序框图	10
§ 3.4 输入数据	11
§ 3.5 主程序 EA2 及部分辅助子程序源程序	13
习题	18
第 4 章 结构自动离散	19
§ 4.1 单元及节点编号	19
§ 4.2 用形状函数插值自动生成节点坐标	21
§ 4.3 单元及节点编号的显示	25
§ 4.4 面力、位移边界条件及多连域问题的处理	26
§ 4.5 轴对称温度场的自动生成	27
§ 4.6 子程序 AUTPL2 介绍	27
§ 4.7 有限元分析建模要点	29
§ 4.8 算例	29
§ 4.9 自动分网子程序 AUTPL2 源程序	31
习题	49
第 5 章 单元刚阵及其故障诊断	50
§ 5.1 单元刚阵基本公式	50

§ 5.2 单元刚阵的特性及故障诊断·····	56
§ 5.3 单元刚阵子程序 MATREL 简介·····	57
习题·····	71
第 6 章 波前法解总体刚度方程组·····	72
§ 6.1 有限元位移边界条件的处理·····	72
§ 6.2 高斯消去法及其性质·····	73
§ 6.3 波前法解有限元方程组·····	76
习题·····	94
第 7 章 单元应力、应变计算·····	95
§ 7.1 基本公式·····	95
§ 7.2 单元应力、应变计算子程序·····	97
§ 7.3 单元应力、应变计算子程序的源程序·····	98
习题·····	104
第 8 章 应力、应变修匀·····	105
§ 8.1 应力、应变修匀概述·····	105
§ 8.2 基本公式·····	105
§ 8.3 应力、应变修匀子程序·····	106
§ 8.4 应力、应变修匀子程序的源程序·····	108
§ 8.5 程序运行中的主要故障及其可能产生的原因·····	116
习题·····	117
第 9 章 二维及轴对称稳态温度场分析有限元法·····	118
§ 9.1 热传导问题基本方程·····	118
§ 9.2 二维稳态无源温度场有限元方程·····	119
§ 9.3 轴对称稳态无源温度场有限元方程·····	120
§ 9.4 二维及轴对称稳态温度场分析有限元程序 TEMPR2·····	120
§ 9.5 二维及轴对称稳态温度场分析有限元程序 TEMPR2 使用说明·····	161
§ 9.6 算例·····	162
习题·····	165
参考文献·····	166

第 1 章 引 言

从数学角度讲,有限元法是求解偏微分方程的一种数值解法。在结构势能存在的情况下,有限元法可通过最小势能原理建立有限元刚度方程组。从这一观点出发,有限元法更适合于求解椭圆类偏微分方程问题。对势能不存在的物理问题,可从加权余量法出发,通过积分变换得到相应的有限元法列式,如流体力学及瞬态热传导问题中的有限元法。

表 1-1 给出了工程中常见偏微分方程组的数值解法、对应的权函数及试函数类型。权函数及试函数的分片特性决定了采用有限元法比采用其他方法对复杂形状求解域问题分析的适应性更好。

表 1-1 力学问题的偏微分方程数值解法及其特点

	方 法	原 理	权函数	试函数
加 权 余 量 法	有限元法(FEM)	最小势能原理	(局部)形状函数	(局部)形状函数
	边界元法(BEM)	积分恒等式	(全局)基本解	(局部)形状函数
	有限差分法(FDM)	微分的差分格式	δ 函数	(全局)多项式
	配点法(SM)	最小二乘法	δ 函数	(全局)多项式
	迦辽金法(GM)	能量原理	试函数基底	(全局)多项式

注: FEM: Finite Element Method
BEM: Boundary Element Method
FDM: Finite Difference Method
SM: Superposition Method
GM: Galerkin Method

有限元法基本原理及公式可参阅参考文献[1~3],有限元法的基本概念可简述如下:

- (1) 将具有无限自由度的求解域离散为由有限个单元(自由度)表征的问题;
 - (2) 利用满足收敛准则的形状函数描述单元内的位移(或应力),利用虚功原理建立单元刚度(或柔度)方程;
 - (3) 利用节点平衡(或协调)关系建立系统方程组,求解节点未知位移(或力);
 - (4) 求解单元内应变、应力(或位移)。
- 与其他数值分析方法相比较,有限元法具有以下一些特点:
- (1) 节点的定位较灵活(与 FDM 相比);
 - (2) 可同时选用不同类型和不同尺寸的元素组装(与 FDM,GM,SM 相比);

- (3)“改进”元素的可能性宽广,只需增加单元参数(节点数)(与 BEM 相比);
- (4)由于它的边界条件积分形式,易于实现对边界值的逼近(与 FDM,SM 相比);
- (5)采用分片多项式作形状函数,对复杂形状适应性强(与 FDM,BEM,GM 相比);
- (6)产生正定对称、稀疏的线性方程组,降低对内存的要求(与 BEM,SM 相比)。

有限元法主要发展历程可参阅参考文献[1~3],在表 1-2 中仅列出在有限元法发展进程中几个具有里程碑意义的重要事件。

表 1-2 有限元法发展中的几个重要事件

有限元理论与技术	创始人	提出年代
离散化技术	Courant	1943 年
最小势能原理	Argyris	1954 年
直刚法解桁架结构	陈百屏	1954 年
用计算机解平面问题	M. J. Turner, R. W. Clough, H. C. Martin, L. J. Topp	1956 年
有限元法命名	R. W. Clough	1960 年
建立与权余法的关系	J. T. Oden	1969 年
等参元概念	O. C. Zienkiewicz	1969 年

有限元法可应用的领域十分广泛,它可以用于分析固体力学、流体力学、热传导、电磁场等问题。

固体力学中的有限元法单元库非常丰富,从而使得用有限元法求解像飞机结构这样非常复杂的杆、板、梁、壳组合结构成为可能。有限元法按结构形式划分可分为多种单元,如:

- (1)一维杆元、梁元;
- (2)二维三角元、四边元、板元、壳元、轴对称元;
- (3)三维四面体元、六面体元。

按插值函数阶次划分,可分为线性元、二阶元、高阶元。

按总刚方程组性质划分,可分为:

- (1)线性分析;
- (2)材料非线性分析(弹塑性分析、黏弹性分析、蠕变分析等);
- (3)几何非线性分析(大应变问题分析、大变形分析)。

按与时间相关性划分,又可分为:

- (1)静态分析(求位移、应变、应力);
- (2)动态分析(求固有频率、振型、响应)。

目前国际流行的大型通用有限元分析软件有 100 多种,其中大多数是由美国拥有版权并组织开发的。它们的基本线性分析功能差别不大,但在单元库及非线性分析功能方面各有特点。国内航空系统各研究所先后引进了近 10 种大型分析软件,其中引进频率最高,且在国内外最常用的几种大型分析软件为 MSC NASTRAN(65 万条),ANSYS, MARC, ABAQUS, IDEAS 等。

本书主要内容涉及:结构化程序设计基本概念,二维网格划分的方法及程序设计问题,二维及轴对称结构稳态温度场及热弹性问题有限元法所涉及的有限元理论、公式及程序设计问题,二维及轴对称稳态温度场分析基本控制方程及边界条件、有限元法理论及公式推导、程序设计问题。书中配有全套二维及轴对称结构热弹性问题有限元法源程序和二维及轴对称稳态温度场分析有限元法源程序,并配有部分程序框图。

第 2 章 结构化程序设计概论

结构化程序设计包含三个方面的内容:算法结构、数据结构、程序设计方法学^[4]。

§ 2.1 算法结构及数据结构

2.1.1 算法结构

算法结构分为顺序结构、选择结构、循环结构和组合结构,其本质属性为:仅有一个入口;无“死块”;无死循环块。选择结构和循环结构在计算机语言中分别与条件语句和循环语句对应。

2.1.2 数据结构

数据结构分为数组结构和记录结构。

1. 数组结构

根据需要进行适当维数的数组,但应尽可能优先选用低维数组。在不同的子程序中数组维数可根据需要通过形参、等价语句或赋值语句进行转换。

2. 记录结构(文件存储)

文件的存储主要分直接存取文件和顺序存取文件两种形式。

(1)直接存取文件只能采用定长度格式记录文件。

特点:可任意选某个记录进行存取操作,使用方便;但每个记录长度必须事先确定。

(2)顺序存取文件可采用等长度或变长度格式记录文件。

特点:不必事先确定记录长度,但必须按顺序读写文件内容,不便于跳记录操作。

3. 数据在子程序之间的传递方式及其特点

(1)通过文件(记录结构)传递。

特点:便于检查、分段调试、再启动,数据可永久性保存;与外设数据交换运行速度慢,占用外存多,程序移植性较差。

(2)通过形参传递。

特点:符合结构化程序设计要求,可读性强,便于子程序单独调试;运行模块较大,占用内存较多。

(3)通过公共块传递。

特点:传递速度快,运行模块较小;子程序模块化性能差,程序不易阅读、修改、维护,对不

同问题的不同维数要求适应性差。

§ 2.2 结构化程序设计技术与方法

2.2.1 模块化程序设计技术与方法

程序设计中,将一个复杂的算法分解为若干相对独立、功能单一的模块,以便于这些模块可以适当地组合为所需要的全局算法(搭积木)。

1. 模块的定义

模块即子程序或其他程序过程所构成的可被其他程序过程调用而独立执行的模块。

2. 模块的划分原则

相对独立,功能单一,输入、输出数据尽可能少。

3. 模块化程序设计的优点

简化软件研制工作(化整为零,分而治之),易于平行作业,减少重复编程工作,缩短开发周期,节省开发费用,增加系统的可维护性和稳定性;便于组装,条理清晰,可读性强,利于维护。

4. 模块化程序设计的缺点

模块间信息交换量大,模块单独调试及联调较麻烦,运行模块较大,运行时间及内存占用量有所增加。

2.2.2 自上而下的结构化程序设计技术与方法

自上而下的结构化程序设计方法是目前程序设计的主要方法之一,它可以提高程序的阅读性,在需要的情况下再逐层展开。

下节中提到的尽量减少无条件 GOTO 语句,采用 FORTRAN 77 中无语句标号形式的循环语句、DO 语句形式,均是为了保证程序结构总体上的自上而下,提高程序的可阅读性、可维护性。

2.2.3 程序设计风格优化技术与方法

1. 算法的设计风格优化

(1)结构清晰化:层次清楚,结构逻辑关系简单,用无条件转移语句少。

(2)表述简化:与以节省内存、节省语句为目的的设计技巧有矛盾。

【例 2.1】 a 与 b 交换的算法有两种:

方法 1: $a=a+b; b=a-b; a=a-b$ (省内存,可读性差)

方法 2: $x=a; a=b; b=x$ (多占用一内存单元,简明)

(3)规模适度化:并行的模块个数、结构的分支层数、循环结构的嵌套重数以及形参个数一般选 7 ± 2 为宜。

2. 计算机语言的运用风格优化

尽量使用高级语言中意义直观的语句功能。如推荐采用

```
IF(条件) THEN
```

```
    分支 1
```

```
ELSE
    分支 2
END IF
```

而不使用

```
IF(条件 1) GOTO 20
    分支 1
GOTO 30
20 CONTINUE
IF(条件 2) GOTO 30
    分支 2
30 CONTINUE
```

当今结构化程序设计强调程序的可读性,尽可能用顺序执行结构描述过程,同时尽量避免使用 GOTO 语句。

3. 算法正文的行文风格优美化

(1)充分利用注释语句。

(2)一致性及助记忆性。

1)语句标号递增排列,读语句格式语句标号为 5000~5999,写语句格式语句标号为 6000~6999。

2)变量名尽量选用与其物理量符号同名或同音的变量,照顾数学习惯(如 a, b, c 为已知量, x, y, z 为未知量);不使用标准标识符及关键字做变量名(如 IF, COS, SIGN),选字母与数字不易混淆的字符(如数字 0 与 1 和字母 o 与 l 极不易辨认)做变量名。

3)语句顺序:

相对集中——子程序尽量跟在被调用的子程序后面。

分门别类——同一类别说明宜集中排列(常数、赋初值、格式语句、维数定义语句等)。

子程序内部的语序:

```
SUBROUTINE
PARAMETER
IMPLICIT
REAL
DIMENSION
COMMON
EQUIVALENT
FUNCTION
EXTERNAL
DATA
可执行语句
格式语句
END
```

(3)正文编排格式。

1)利用空行、空格改善程序的可读性(块退格,段空行);

2)注意括号的使用;

3) 变量、语句排列有规律, 数组定义按字母顺序排列或按维数大小排列。

4. 输出信息明了化

(1) 显示输入参数, 以便检查。

(2) 输出参数应有文字说明。

(3) 精心设计输入、输出表。

2.2.4 结构化程序设计优化技术与方法

1. 数学模型优化

【例 2.2】 求 $1^3, -2^3, 3^3, -4^3, \dots, (-1)^{n-1}n^3$ 的 N 项和。

方法 1: 直接按公式 $S = \sum_{n=1}^N (-1)^{n-1}n^3$ 求解。

方法 2: 按公式 $S_n = \begin{cases} (2n-1)(n+1)^2/4, & n \text{ 为奇数} \\ n^2(2n+3)/4, & n \text{ 为偶数} \end{cases}$ 求解。

方法 2 不涉及循环结构算法, 应优先选用。

2. 计算方法优化

【例 2.3】 求方程 $X^2 - (10^{15} + 1)X + 10^{15} = 0$ 的两个实根。

方法 1: 由公式 $X_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ 求根, 得 $X_1 = 10^{15}, X_2 = 0$ (不是原方程的根)。误差产生的原因是两相近的数相减运算造成的有效位数丢失。

方法 2: 由韦达定理求根

$$X_1 = \frac{-b - \text{sign}(b)\sqrt{b^2 - 4ac}}{2a}$$

$$X_2 = \frac{c}{aX_1}$$

求得 $X_1 = 10^{15}, X_2 = 1$, 是原方程的根。

显然, 采用方法 2 求根可获得更精确的结果。

3. 算法优化

【例 2.4】 找 n 个实数 $a_i (i = 1, 2, \dots, n)$ 由小到大进行排列。

方法 1: 遴选法。

从 a_i 中找最小者存入 b_1 , 同时用足够大数 M 替换 a_i 中最小者;

从 a_i 中找最小者存入 b_2 , 同时用足够大数 M 替换 a_i 中最小者;

.....

直至找到最小的一个数放入 b_n 。

需要进行比较操作次数为 $n(n-1)$ 次。

方法 2: 择换法。

从 $a_i (i = 1, 2, \dots, n)$ 中找最小者, 并与 a_1 对换位置;

从 $a_i (i = 2, 3, \dots, n)$ 中找最小者, 并与 a_2 对换位置;

.....

从 a_{n-1}, a_n 中找最小者, 并与 a_{n-1} 对换位置。

需要进行比较操作次数为 $n(n-1)/2$ 次。

方法 2 明显操作次数少,且节省内存。

4. 数据结构优化

数据结构的优化需要适应不同算法的要求。如对前述算法优化中的排序问题,采用方法 1 还需再开设一个长度为 n 的一维数组存放排序后的 a_i ,而采用方法 2 则不需要。

5. 存储空间节约技术

(1) 数据共享,如公共块技术;

(2) 常量简化,如取 $a = \ln 3.14159$;

(3) 选择数据类型与精度,如单精度实型数比双精度实型数节约内存;

(4) 变量精简,闲置的变量可作他用;

(5) 子程序技术,如相同功用,多次在不同地方使用的程序段,适合编成子程序,多次调用。

(6) 程序覆盖技术;

(7) 精心组织数据,如对称、稀疏矩阵的对称变带宽存储;

(8) 选择适合的算法,如波前法可节省内存;

(9) 采用动态一维可调数组管理技术;

(10) 尽量减少形参个数。

6. 程序运行加速技术

(1) 变量替换技术,如对多次使用的表达式用变量表示,以备多次使用。

(2) 运算降级技术。

各种运算耗时比值如下:

1) 小型机:加(减)法 / 乘法 / 除法的运算量比值为 1 : 8 : 20;

2) 中型机:加(减)法 / 乘法 / 除法的运算量比值为 1 : 4 : 8;

3) 大型机:加(减)法 / 乘法 / 除法的运算量比值为 1 : 2 : 3;

4) 整型运算:赋值 / 加(减)法 / 乘法 / 乘方的运算量比值为 1 : 1.5 : 8 : 35;

5) 实型运算:赋值 / 加(减)法 / 乘法 / 除法 / 乘方的运算量比值为 2 : 3 : 15 : 30 : 115;

6) 如加法运算量为 1,整型变实型的运算量为 6;

7) 如加法运算量为 1,函数求值如 $\lg x, \sin x$ 等的运算量大于等于 150;

8) 如加法运算量为 1,双下标变量寻址的运算量为 11。

【例 2.5】 $x * 2$ 用 $x * x$ 表示可省时 $3/4$; $1/2/E/PI$ 用 $1/(2 * E * PI)$ 表示可省时 $1/3$ 。

两个二维数组乘运算:

```
DO I=1, M
  DO J=1, N
    A(I, J)=0.
    DO K=1, L
      A(I, J)=A(I, J)+B(I, K) * C(K, J)
    END DO
  END DO
END DO
```

用下面形式运算将更快,则

```

DO I=1, M
  DO J=1, N
    S=0.
    DO K=1, L
      S=S+B(I, K) * C(K, J)
    END DO
    A(I, J)=S
  END DO
END DO

```

因后者将 $2 * K * N * M$ 个双下标寻址任务变为单变量寻址任务。

运算中应精选数据类型,如整型运算比实型运算省时,单精度运算比双精度运算省时。

- (3)循环结构精练化,与循环变量无关的运算尽可能移到外层循环。
- (4)循环嵌套精巧化,若有多层循环,应考虑将涉及循环变量少的运算放到外层。
- (5)计算变量同型化,如 $2. * \ln A$ 比 $2 * \ln A$ 少做一次整型变实型数的转换操作。
- (6)重复求值查表化。

7. 精度保护与防溢技术

尽量减少两相近数作减法运算,如前述例 2.3。对每个除法运算的分母要仔细考虑,避免出现除零运算。

8. 误差修正与误判防治技术

防止用整型数与实型数比较作条件判定的逻辑运算。

9. 人-机接口设计友好化

- (1)数据输入量尽可能少而且方便;
- (2)减少程序的人工干预;
- (3)输入尽量菜单化;
- (4)输出结果表格化、页面化、文档化。

在科学计算类的程序设计过程中,除了程序设计风格优美化、人机界面友好化的一般性要求外,着重要解决计算时间(CPU)、计算空间(内存、硬盘需求量)、计算精度、程序可读性和可维护性以及程序的可扩展性之间的矛盾。对科学计算类的程序,由于具有数据传递量大,内存占用多,计算费时,原始数据准备工作量大等特点,需要根据当时当地的计算硬件条件、计算任务特点及规模,仔细分析,抓住其主要矛盾,选择适当的计算方法、数据传递方式及程序结构。随着时间的推移、硬件条件的改善及计算技术的发展,原来的主要矛盾将会发生转变,在一个时段以后,可对原程序系统进行改编、升级,以适应新环境、新任务的要求。

习 题

2.1 程序设计中,数据在子程序之间的传递方式有哪些? 它们各有何优、缺点?

2.2 简述科学计算程序设计中主要应考虑哪些方面的问题,举例说明它们之间有何矛盾,问题如何解决。

第 3 章 二维及轴对称体热弹性有限元分析程序设计

§ 3.1 二维及轴对称体热弹性有限元分析程序功能及特点

二维及轴对称体热弹性有限元分析程序系统(Thermal Elastic Finite Element Analysis Program for Two-dimensional and Axisymmetric Structures)缩写为 EA2,其主程序名也取名为 EA2。

二维及轴对称体热弹性有限元分析程序 EA2 早期版本的设计是在 20 世纪 80 年代初,计算机硬件条件较落后的情况下进行的,因此对节省计算时间及内存的使用量做了较多的考虑。随着时间的推移,计算机硬件条件的改善,当时计算速度慢、内存不够用的矛盾在很大程度上已得到缓解,因而矛盾的主要方面也发生了变化。该程序系统后来虽几经改版升级,但仍没有从根本上改变原来程序系统的总体框架和数据的基本传递方式。读者可从辩证唯物主义的观点出发,客观地评价该程序系统的各个方面。

二维及轴对称体热弹性有限元分析程序的主要功能及特点可归纳为以下几点:

- (1)采用波前法解线性方程组,节省内存,可在微机上解相对自由度较多的问题;
- (2)波前法对单元节点编号要求低,从而易于实现自动网格划分;
- (3)对单元、节点编号、单元平衡性、正定性、总刚的奇异性进行检查,易于排除故障;
- (4)可自动生成温度、离心力等效节点载荷;
- (5)程序已进行通用化处理,不需修改源程序就可分析一般二维及轴对称体热弹性问题;
- (6)各子程序之间采用文件传递数据,程序可分段启动;
- (7)选用八节点等参元,程序精度较高。

§ 3.2 主程序 EA2 的功能

- (1)控制、模型描述、材料特性、外载荷参数的读入;
- (2)程序主要流程控制。

§ 3.3 主程序 EA2 程序框图

主程序 EA2 程序框图如图 3-1 所示。