

数码工程师系列丛书

MSC.Marc 二次开发指南

陈火红 尹伟奇 薛小香 编著

科学出版社

北 京

内 容 简 介

本书介绍了 MSC.Marc 软件二次开发的基本原理和使用方法,内容全面,包括 MSC.Marc 二次开发的概况及基本过程, Fortran 计算机语言的基本知识, 有关加载及边界条件施加的用户子程序的使用, 各向异性材料和本构关系用户子程序的使用, 粘塑性和粘弹性用户子程序的使用方法, 通过用户子程序修改单元几何形状, 定义用户特殊的输出, 有关滑动轴承分析的用户子程序的使用, 如何对 Marc 的后处理文件进行重新处理, 将用户与温度、应变率相关的材料弹塑性数据文件直接调到程序之中的过程, 将复杂的本构关系及其积分过程加到 Marc 之中的方法及过程, 二次开发在大型结构热分析中的应用以及如何利用 Python 语言进行二次开发。书中提供了很多例题, 涉及各种不同的分析类型, 便于用户学习和练习。

本书可作为广大工程技术人员使用 MSC.Marc 的参考书, 也可作为理工院校相关专业的高年级本科生、研究生及教师学习 MSC.Marc 软件的教材和参考书。

图书在版编目 (CIP) 数据

MSC.Marc 二次开发指南/陈火红等编著. - 北京: 科学出版社, 2004
(数码工程师系列丛书)

ISBN 7-03-014508-9

.M... .陈... .有限元分析-应用软件, MSC.Marc-软件开发
.0241.82-39

中国版本图书馆 CIP 数据核字 (2004) 第 108430 号

责任编辑: 吕建忠 陈砺川/责任校对: 柏连海

责任印制: 吕春珉/封面设计: 飞天创意

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>

印刷

科学出版社发行 各地新华书店经销

*

2004 年 11 月第一版 开本: 787×1092 1/16
2004 年 11 月第一次印刷 印张: 31 1/4
印数: 1 - 4 000 字数: 720 000

定价: 60.00 元

前 言

MSC.Marc 软件在我国的航空、航天、核工业、铁路运输业、石油化工、机械制造、能源、汽车、电子、土木工程、造船、生物医学、轻工、地矿、水利等领域得到广泛的应用,为各领域中产品设计、科学研究做出了很大贡献。该软件的功能在不断地改进,应用领域也在不断地扩展。

但由于实际问题的多样性,以及不同用户要求的特殊性,利用软件缺省的标准输入/输出有时并非不是最佳选择。另外,也可能存在标准程序尚不具备而特定用户需要某一方面功能的情形。对前一种情况,可通过 MSC.Marc 提供的大量用户子程序接口,将用户需要的输入/输出以最简便的方式定义,而无需受缺省输入/输出的限制。对于后者,在功能强大的通用软件框架下可以耦合进用户所需功能,使通用软件向特定领域的专用软件扩展。

MSC.Marc 的许多国外著名的用户已利用 MSC.Marc 提供的开放性求解问题的能力,并借助于用户子程序完成了众多高级复杂工程问题分析和学术研究。例如由美国宇航局牵头,美国政府、工业界和著名大学三方合作完成的 MHOST 计划,就涉及许多这样的 MSC.Marc 二次开发工作。而由 NASA 和 MSC.Marc 共同开发的新型可再用航天飞行器模型 X-33 的热翼面擅振分析也是一个扩展 MSC.Marc 软件分析功能的成功实例。

自从 1995 年以来, MSC.Marc 在国内得到了越来越广泛的应用。一些国内的用户利用二次开发扩展了程序功能。如高温结构和土木工程中的本构模型研究、复杂结构的热分析、金属成型过程中的组织演化、焊接过程中的热源控制、网格死活条件和自适应条件控制等。

为了使广大的用户更好地使用软件的二次开发功能,编写一本有关二次开发功能的书,一直是编者的一个愿望。编者从实际工程应用出发,结合使用软件的经验以及帮助用户解决问题的经验,根据 MSC.Software 公司的最新资料编写了本书。

本书共分 14 章。第 1 章介绍 MSC.Marc 二次开发的概况及基本过程;第 2 章介绍 Fortran 计算机语言的基本知识;第 3 章通过多个实例介绍有关加载及边界条件施加的用户子程序的使用;第 4 章通过多个实例介绍各向异性材料和本构关系用户子程序的使用;第 5 章介绍粘塑性用户子程序的使用方法;第 6 章介绍粘弹性用户子程序的使用方法;第 7 章介绍如何通过用户子程序修改单元几何形状;第 8 章介绍如何定义用户特殊的输出;第 9 章介绍有关滑动轴承分析的用户子程序的使用;第 10 章介绍如何对 MSC.Marc 的后处理文件进行重新处理;第 11 章介绍如何将用户与温度、应变率相关的材料弹塑性数据文件直接调到程序之中;第 12 章通过实例介绍如何将复杂的本构关系及其积分过程加到 MSC.Marc 之中;第 13 章通过实例介绍二次开发在大型复杂结构热分析中的应用;第 14 章介绍如何利用 Python 语言进行二次开发。

由于篇幅有限,本书介绍的仅是 MSC.Marc 二次开发功能中的一部分,读者如要做比较复杂的二次开发工作,还需要全面掌握有限元分析所需的相关知识,参考其他有关教程以及软件的英文文档,并在实践中不断学习与提高。也可以从 MSC.Software 公司

的网站（www.mscsoftware.com、www.marc.com 和 www.mscsoftware.com.cn）得到有关信息。

本书在编写过程中得到了 MSC.Software 公司中国首席代表李军毅先生的大力支持和指导，也得到了 MSC.Software 公司中国各办事处同事们的许多帮助。另外，张清泉、王刚、陆新征、董洪波等先生提供了书中一些用户子程序文件，科学出版社的有关工作人员对本书的出版做了大量的工作。编者在此向他们表示衷心的感谢。

由于编者水平有限，书中缺点和错误在所难免，敬请读者批评指正，也欢迎用户和读者来信来函共同探讨。

第一编者的 E-mail：huohong.chen@mscsoftware.com。

目 录

第 1 章	MSC.Marc 二次开发基础.....	1
1.1	概述	1
1.2	用户子程序分类及常用用户子程序.....	2
1.3	利用公共块进行数据传递	3
1.4	利用 ELMVAR 和 NODVAR 提取单元和节点变量	6
1.4.1	利用 ELMVAR 提取单元求解结果	6
1.4.2	利用 NODVAR 提取节点结果	7
1.5	利用内部子程序进行矩阵运算.....	8
1.6	用户子程序调用原理	9
1.7	应用举例——移动载荷.....	10
1.7.1	分析模型描述	10
1.7.2	用户子程序的选择及其格式	11
1.7.3	用户子程序代码.....	12
1.7.4	采用 Mentat 做前后处理.....	13
1.7.5	采用 Patran 做前后处理.....	19
第 2 章	Fortran 语言基础.....	31
2.1	Fortran 简介	31
2.1.1	Fortran 77 格式	32
2.1.2	程序实例	32
2.2	常用概念	34
2.2.1	常量	34
2.2.2	变量	35
2.2.3	常用 Fortran 函数	36
2.2.4	算术运算符.....	36
2.2.5	赋值语句.....	36
2.2.6	参数语句 PARAMETER.....	37
2.2.7	DATA 表达式定义变量初值	37
2.2.8	END 语句、STOP 语句和 PAUSE 语句.....	37
2.3	逻辑运算和选择结构	38
2.3.1	关系运算符.....	38
2.3.2	逻辑运算符.....	39
2.3.3	用 If-then-else 实现选择结构	39
2.4	循环结构的实现	40
2.4.1	用 Go to 语句实现循环	40
2.4.2	用 Do 语句实现循环	40
2.4.3	用 While 语句实现循环	41

2.4.4	Continue 语句	41
2.5	Fortran 的数据结构	41
2.5.1	双精度类型数据	41
2.5.2	复型数据	42
2.5.3	字符型数据	42
2.6	数据的输入/输出	43
2.6.1	格式化输入/输出	43
2.6.2	自由格式输出语句	44
2.7	数组	44
2.7.1	定义数组	44
2.7.2	在子程序中可变数组的定义	45
2.8	子程序	45
2.8.1	函数子程序的定义和调用	45
2.8.2	子例子程序的定义和调用	46
2.8.3	数据块子程序	46
2.9	数据存储方式	47
2.10	文件操作语句	48
2.11	用户子程序中的文件操作	48
2.11.1	将结果写入文件	48
2.11.2	从数据文件中读入数据	49
2.11.3	定义输入数据	50
2.11.4	重新定义重启动文件输出	51
2.11.5	退出计算运行	52
第 3 章	用户定义的加载、边界条件和状态变量	53
3.1	概述	53
3.2	热源控制	54
3.2.1	子程序格式	55
3.2.2	分析模型描述	56
3.2.3	用户子程序代码	61
3.2.4	计算结果	62
3.3	非线性弹簧	63
3.3.1	子程序格式	64
3.3.2	分析模型描述	64
3.3.3	用户子程序代码	71
3.3.4	计算结果	73
3.4	集中力跟随	75
3.4.1	子程序格式	75
3.4.2	分析模型描述	76
3.4.3	用户子程序代码	79

3.4.4	计算结果.....	85
3.5	分布力.....	87
3.5.1	子程序格式.....	88
3.5.2	分析模型描述.....	88
3.5.3	用户子程序代码.....	92
3.5.4	计算结果.....	93
3.6	对流边界和流动速度用户子程序及应用	95
3.6.1	子程序格式.....	95
3.6.2	分析模型描述.....	96
3.6.3	用户子程序文件.....	100
3.6.4	分析结果.....	102
3.7	集中弯矩	103
3.7.1	子程序格式.....	104
3.7.2	分析模型描述.....	104
3.7.3	用户子程序代码.....	106
3.7.4	计算结果.....	107
3.8	非均匀温度场分布.....	108
3.8.1	子程序格式.....	109
3.8.2	分析模型描述.....	109
3.8.3	用户子程序代码.....	114
3.8.4	计算结果.....	115
3.9	梁在温度梯度和热辐射影响下的蠕变.....	117
3.9.1	子程序格式.....	117
3.9.2	蠕变规律.....	118
3.9.3	分析模型描述.....	119
3.9.4	用户子程序代码.....	124
3.9.5	计算结果.....	127
3.10	位移响应谱的输入	128
3.10.1	子程序格式.....	129
3.10.2	分析模型描述.....	129
3.10.3	用户子程序代码.....	133
3.10.4	计算结果.....	134
第 4 章	用户定义的各向异性材料特性和本构关系	136
4.1	概述	136
4.2	线性平面应力三角形单元.....	137
4.2.1	子程序格式.....	138
4.2.2	分析模型描述.....	140
4.2.3	用户子程序代码.....	145
4.2.4	计算结果.....	150

4.3	邓肯 - 张非线性弹性 E-B 模型	151
4.3.1	子程序格式	152
4.3.2	模型描述	155
4.3.3	用户子程序代码	159
4.3.4	计算结果	163
4.4	梁单元	166
4.4.1	子程序格式	166
4.4.2	非线性弹性梁	167
4.4.3	非线性弹性梁分析模型描述	168
4.4.4	非线性弹性梁使用的用户子程序代码	172
4.4.5	非线性弹性梁计算结果	173
4.4.6	双折线弹塑性梁单元	174
4.4.7	双折线弹塑性梁分析模型描述	174
4.4.8	双折线弹塑性梁使用的用户子程序代码	178
4.4.9	双折线弹塑性梁计算结果	182
4.5	3D 复合材料铺层	183
4.5.1	用户程序 change.f 的使用说明	185
4.5.2	模型描述	186
4.5.3	用户子程序代码	190
4.5.4	计算结果	195
4.6	正交各向异性材料方向的定义	197
4.6.1	程序格式	197
4.6.2	模型描述	197
4.6.3	用户子程序代码	204
4.6.4	计算结果	206
4.7	各向异性材料定义	207
4.7.1	子程序格式	207
4.7.2	分析模型描述	209
4.7.3	用户子程序代码	213
4.7.4	计算结果	215
4.8	采用变形梯度定义非线性应力—应变关系	216
4.8.1	子程序格式	217
4.8.2	分析模型描述	218
4.8.3	用户子程序代码	223
4.8.4	计算结果	227
4.9	Rebar 单元属性定义	228
4.9.1	子程序格式	228
4.9.2	分析模型描述	230
4.9.3	用户子程序代码	236

4.9.4	计算结果.....	238
4.10	材料的硬化特性定义.....	239
4.10.1	子程序格式.....	239
4.10.2	分析模型描述.....	240
4.10.3	用户子程序代码.....	244
4.10.4	计算结果.....	245
第 5 章	粘塑性和广义塑性用户子程序.....	247
5.1	概述.....	247
5.2	隐式蠕变分析.....	247
5.2.1	用户子程序格式.....	248
5.2.2	分析模型描述.....	250
5.2.3	用户子程序代码.....	254
5.2.4	计算结果.....	257
第 6 章	粘弹性用户子程序.....	260
6.1	概述.....	260
6.2	玻璃热分析.....	260
6.2.1	热传导温度场分析模型描述.....	261
6.2.2	温度场计算结果.....	265
6.3	玻璃粘弹性分析.....	266
6.3.1	用户子程序格式.....	266
6.3.2	粘弹性分析模型描述.....	267
6.3.3	用户子程序代码.....	275
6.3.4	粘弹性分析结果.....	276
第 7 章	修改几何形状的用户子程序.....	279
7.1	概述.....	279
7.2	厚度变化.....	280
7.2.1	用户子程序格式.....	280
7.2.2	分析模型描述.....	280
7.2.3	用户子程序代码.....	282
7.2.4	计算结果.....	286
7.3	焊接死活单元.....	288
7.3.1	子程序格式.....	288
7.3.2	分析模型描述.....	289
7.3.3	用户子程序代码.....	296
7.3.4	计算结果.....	300
第 8 章	定义输出量的用户子程序.....	303
8.1	概述.....	303
8.2	晶粒大小.....	303
8.2.1	用户子程序格式.....	304

8.2.2	分析模型描述.....	305
8.2.3	用户子程序代码.....	313
8.2.4	计算结果.....	315
8.3	莫尔应力和 Tresca 应力输出.....	317
8.3.1	用户子程序格式.....	318
8.3.2	分析模型描述.....	318
8.3.3	用户子程序代码.....	321
8.3.4	计算结果.....	322
第 9 章	定义滑动轴承分析的用户子程序	324
9.1	概述	324
9.2	程序格式与实例分析	325
9.2.1	程序格式.....	325
9.2.2	模型描述.....	326
9.2.3	用户子程序代码.....	331
9.2.4	计算结果.....	332
第 10 章	程序 PLDUMP 2000.....	334
10.1	概述	334
10.2	文件格式及 PLOUMP 2000 的应用	334
10.2.1	后处理文件格式.....	335
10.2.2	PLDUMP 2000 概述.....	338
10.2.3	应用举例.....	340
10.2.4	运行结果.....	344
第 11 章	用户材料库定义	348
11.1	概述	348
11.2	用户材料库的定义.....	348
11.2.1	用户材料库的定义过程.....	348
11.2.2	应用举例.....	350
11.2.3	例题中流动应力文件.....	353
第 12 章	高温结构粘塑性分析	356
12.1	背景	356
12.2	Walker 模型及其修正模型	357
12.2.1	Walker 模型.....	357
12.2.2	Walker 模型的修正模型	358
12.3	有限元分析方法	358
12.4	用户子程序文件	364
12.5	程序验证.....	379
12.5.1	分析模型的建立.....	379
12.5.2	结果讨论.....	381

第 13 章 复杂结构热分析	383
13.1 背景	383
13.2 基本理论和方法	383
13.2.1 热传导分析单元	383
13.2.2 辐射边界条件的处理方法	384
13.2.3 对流换热系数的确定方法	385
13.2.4 辐射通量密度及辐射强度计算	386
13.3 自编程序及用户子程序文件	387
13.4 例题分析	400
13.4.1 模型与网格划分	400
13.4.2 计算工况的选取	400
13.4.3 边界条件的施加	400
13.4.4 计算及结果分析	402
第 14 章 MSC.Marc Python 指南	405
14.1 Python 简介	405
14.1.1 PyMentat 使用基础	406
14.1.2 PyPost 使用基础	407
14.1.3 Python 编程简介	408
14.2 一个简单实例	409
14.2.1 生成节点和单元的 Python 脚本代码	410
14.2.2 Python 的一些语法规则	411
14.2.3 运行脚本	411
14.3 如何通过 PyMentat 从 MSC.Mentat 中获取数据	412
14.3.1 为 MSC.Mentat 创建一个简单的 Python 脚本	412
14.3.2 py_get_int 和 py_get_float 方法	413
14.3.3 运行脚本	414
14.4 如何使用 Python 创建复杂几何体	414
14.4.1 板的属性	414
14.4.2 Python 脚本代码	415
14.4.3 运行脚本	417
14.5 如何使用 PyMentat 添加模型属性	417
14.5.1 板的属性	418
14.5.2 节点和单元标识符	420
14.5.3 运行脚本	422
14.5.4 提交作业	422
14.6 处理后处理结果文件	423
14.6.1 后处理 Python 脚本基础	423
14.6.2 标量值	425
14.6.3 运行脚本	426

14.7	如何使用组	426
14.7.1	组基础	426
14.7.2	一个简单例子	427
14.7.3	运行脚本	428
14.7.4	组标识符和组名	428
14.8	读取后处理结果文件	429
14.8.1	PyPost 模块基础知识	429
14.8.2	运行脚本	431
14.9	如何使用 PyPost 获取单元数据	431
14.9.1	处理单元数据	431
14.9.2	运行脚本	434
14.10	处理单元张量数据	435
14.10.1	后处理 Python 脚本基础	435
14.10.2	运行脚本	437
14.11	如何使用 py_connect 方法	438
14.11.1	为 MSC.Mentat 创建一个简单的 Python 脚本	438
14.11.2	py_connect 方法	439
14.11.3	处理 socket 错误	439
14.11.4	运行脚本	440
14.12	处理单元张量数据	441
14.12.1	面向 MSC.Mentat 的绘图模块 gdchart	441
14.12.2	gnuplot 模块	443
14.12.3	OpenGL 模块	446
14.13	Python 常用程序模块	452
14.13.1	简介	452
14.13.2	PyMentat 程序模块	453
14.13.3	PyPost 程序模块	456
14.13.4	算术和数据库函数	474
14.14	几何非线性悬臂梁优化实例	477
14.14.1	悬臂梁优化设计目标	478
14.14.2	优化设计思路	478
14.14.3	悬臂梁高度优化设计的 Python 脚本	479
14.14.4	运行脚本	484
参考文献		486

第 1 章 MSC.Marc 二次开发基础

本章讲述了 MSC.Marc 中二次开发的作用、用户子程序的类型、公共块的使用、调用的原理以及一些简单例子。

本章内容包括：

- 概述
- 用户子程序分类及常用用户子程序
- 利用公共块进行数据传递
- 利用 ELMVAR 和 NODVAR 提取单元和节点变量
- 利用内部子程序进行矩阵运算
- 用户子程序调用原理
- 应用举例——移动载荷

1.1 概 述

用有限元分析不同工程背景的实际问题都可归结为离散有限元模型、给定载荷、定义初始/边界条件和确定材料模型、分析求解、结果输出等几个步骤。作为一个在应用领域中通用的有限元软件，MSC.Marc 提供了方便灵活的输入/输出功能来建立有限元分析模型。MSC.Marc 软件以多样的单元类型、齐全的边界条件、广泛的材料模型、支持多种场问题及耦合场的有限元分析，可以输出各类分析结果。如果把程序提供的缺省单元类型、边界条件、材料模型定义方式称为标准输入，缺省的各种结果存储方式称为标准输出，那么绝大多数实际问题的分析模型都可通过 MSC.Marc 的标准输入和输出建立和完成。

但由于实际问题的多样性，不同用户要求的特殊性，利用缺省的标准输入/输出有时并非最佳选择。另外，也可能标准的程序尚不具备用户需要的某一方面功能。对前一种情况，可通过 MSC.Marc 提供的大量用户子程序接口，将用户需要的输入/输出以最简便的方式定义，而无需受缺省输入/输出的限制。对于后者，在功能强大的通用软件框架下可以耦合进用户所需功能，使通用软件向特定领域应用扩展。

MSC.Marc 的许多用户已利用 MSC.Marc 提供的开放性求解能力，借助于用户子程序完成了众多复杂工程问题的分析和学术研究。例如由美国宇航局牵头，美国政府、工业界和著名大学三方合作完成的 MHOST 计划，就涉及许多这样的 MSC.Marc 二次开发工作。该项目着重于通过实验、理论分析和数值模拟研究来开发新一代航空涡轮发动机高温部件。MSC.Marc 软件是 MHOST 项目采用的主要分析软件之一，许多高温材料本构模型，例如 Walker、Bonder-Partom、Miller 等统一粘塑性本构模型，都是通过亚弹性

材料模型用户子程序耦合进 MSC.Marc 软件,对这些高温部件的循环非线性分析及疲劳寿命评定,获得了与实验很接近的结果。

由 NASA 和 MSC.Marc 共同开发的新型可再用航天飞行器模型 X-33 的热翼面擅振分析也是一个扩展 MSC.Marc 软件分析功能的成功实例。NASA 认为这种新型航天飞行器有望作为更经济的空间登陆工具,具有快速返回、维护低廉的特点。

自 1995 年以来, MSC.Marc 在国内得到了越来越广泛的应用。一些国内的用户利用二次开发扩展了程序功能。如高温结构和土木工程中的本构模型研究、复杂结构的热分析、金属成型过程中的组织演化、焊接过程中的热源控制、网格死活条件和自适应条件控制等。

在 MSC.Marc 2003 版本中,软件提供了 328 个有名公共块和 130 个用户子程序接口。用户在用户子程序中调用这些公共块,可以提取所需数据或者以新的数据赋值给公共块变量和数值,进行数据交换。130 个用户子程序入口几乎覆盖了 MSC.Marc 有限元分析的所有环节。从几何建模、网格划分、边界条件定义、材料选择到分析求解、结果输出,用户除了无法更改求解方法外,在有限元分析的其他环节上 MSC.Marc 都预留了用户可以访问的子程序接口。

MSC.Marc 提供以 Fortran 语言编写的用户子程序模板文件。使用时,用户可以按 Fortran 编程规则,根据需要填充子程序模板文件。程序中包含一个或若干个用户子程序时,会以这些用户定义的子程序代替缺省的相应子程序,重新生成新的执行程序,使程序以用户期望的方式运行。

本章将列出用户子程序分类及常用用户子程序,讨论如何利用公共块进行数据传递,如何使用进行矩阵运算的内部子程序,最后给出受移动载荷作用的方板分析作为用户子程序应用实例。

1.2 用户子程序分类及常用用户子程序

MSC.Marc 的用户子程序大致可分为八类,下面为具体分类及常用用户子程序。

1. 用户定义的加载、边界条件和状态变量

利用此类用户子程序可以在结构分析或耦合分析时定义集中力、给定节点位移,定义分布机械载荷,定义 2D、3D 刚性接触体的速度,定义 2D、3D 接触分析的分离力,定义 2D、3D 接触分析按应力控制接触后分离的分离应力,定义用户建立的节点自由度间的约束矩阵,定义随时间、温度、接触压力、应力变化的摩擦系数。在简谐激励分析中,可以定义集中力或运动边界条件。在频谱分析中,定义位移谱密度函数。在热应力分析时,可以定义包含温度在内的状态变量。在热分析和耦合分析时,可以定义随时间和空间变化的对流边界,定义分布热流密度。

2. 用户定义的各向异性材料特性和本构关系

利用此类用户子程序可以定义各向异性或正交各向异性材料主轴的方向余弦、弹性应力—应变关系、热应变增量、热传导率或焦耳热分析中的电阻率,定义 Hill 各向异性或正交各向异性塑性的应力—应变关系,也可以定义蠕变应变率演化方程,定义非线性

应力—应变关系，采用形变梯度定义非线性应力—应变关系，定义梁单元的非线性广义应力—应变关系，定义 Mooney-Rivlin 材料参数随温度变化的函数，添加用户自定义单元；定义加工硬化或应变硬化曲线等。

3. 粘塑性和广义塑性用户子程序

利用此类用户子程序可以定义显式粘塑性模型的非弹性应变率，定义隐式粘塑性模型的非弹性应变率，定义各向同性或各向异性弹性材料进入塑性后的屈服应力，定义等效应力等。

4. 粘弹性用户子程序

利用此类用户子程序可以定义显式的广义 Kelvin 模型，定义各向异性粘弹性材料性质，定义热流变简单材料的平移函数等。

5. 修改几何形状的用户子程序

利用此类用户子程序可以定义加筋单元的轴向方向余弦及等效厚度，定义分析过程中需激活或删除的单元，定义用户确定的网格自适应控制准则及加密单元过程中新生节点的几何位置，定义单元连接关系的变化，定义网格重划过程中节点坐标的变化和单元连接关系的变化，定义壳元积分点处的厚度，定义节点自由度的局部变换矩阵等。

6. 定义输出量的用户子程序

利用此类用户子程序可以定义提取单元结果、积分点坐标、提取节点向量结果；将单元量输出，写到后处理结果文件中；在每个增量步开始时或结束时提取参数的值或对参数赋值；定义节点向量写入后处理文件，以便在 MSC.Mentat 中进行可视化后处理。

7. 定义滑动轴承分析的用户子程序

利用此类用户子程序可以定义油膜表面的方向，定义槽深，定义非均布约束器系数和冲击压力，定义润滑剂厚度，定义轴承表面节点速度等。

8. 其他用户子程序

PLDUMP 2000（在 MSC.Marc 2000 以前版本中为 PLDUMP）是 MSC.Marc 分析后处理结果文件的再处理器。用来访问、分析、转换或处理 MSC.Marc 后处理结果文件。通过 PLDUMP 2000 可将 MSC.Marc 计算结果文件逐一注释。一方面便于检查结果，MSC.Marc 将 PLDUMP 2000 的源程序免费提供给用户，用户可以重新更改以适合自己需要；另一方面，依照已知的结果输出格式和提供的 PLDUMP 2000 源程序，修改后可将 MSC.Marc 结果文件转存成其他软件可识别的结果文件进入其图形处理界面。

1.3 利用公共块进行数据传递

MSC.Marc 用户定义的子程序经常需要与不同模块进行数据传递。用户子程序调用

语句中的形参可以进行数据传递，但往往不够，可以用 MSC.Marc 软件提供的 328 个有名公共块进行数据传递。这些公共块在 MSC.Marc 的安装目录下，用户编程时可用“include”语句将它们包含进去。如

```
UNIX
include 'xxx/version/common/yyy'
NT
include '../common/yyy'
```

其中 xxx 为 MSC.Marc 安装的全路径，version 为 MSC.Marc 的版本，如 MSC.Marc 2003；yyy 为公共块名。

由于 MSC.Marc 程序均采用双精度，为了保证传递数据的相容性及所包含的公共块的正确性，对于除 Cray 机外的所有机器，每个用户子程序都需有下列双精度说明语句：

```
IMPLICIT REAL *8 (A-H, O-Z)
```

常用数据名及所在的公共块如下。

1) 增量步开始时间 CPTIM、时间增量 TIMINC，在所有用户子程序中均可由下列语句得到：

```
include 'path/common/creeps'
```

2) 增量步号 INC、子增量步号 INCSUB，在所有用户子程序中均可由下列语句得到：

```
include 'path/common/concom'
```

3) 网格单元总数 NUMEL、网格节点总数 NUMNP、每个节点的最大自由度数 NDEG、坐标方向的最大数 NCRD，在所有用户子程序中均可由下列语句得到：

```
include 'path/common/dimen'
```

4) 单元号 M 可在单元循环中调用的用户子程序中由下列语句得到：

```
include 'path/common/far'
```

5) 内部单元号 N、积分点号 NN、层号 KC，可在单元循环中调用的用户子程序中由下列语句得到：

```
include 'path/common/lass'
```

在 MSC.Marc 用户手册 D 卷的第一章中，还给出了如何得到积分点坐标、根据程序内部节点号（优化后节点号）得到外部节点号（优化前节点号）、根据程序外部节点号得到内部节点号、根据程序内部节点号得到坐标、根据程序外部节点号得到总位移、利用 MSC.Marc 内部子程序 ELMVAR 得到单元各种结果。

在 MSC.Marc 用户手册 D 卷的第一章中，还给出了最常用的三个公共块 MATDAT、CONCOM 和 ELMCOM 的所有变量说明。

在用户子程序 ANELAS、HOOKLW、ANPLAS、ANEXP、ANKOND、ORIENT、

CRPLAW 和 VSWELL 等中必须使用材料识别号 (如 1, 2, 3 等) 来找到或定义有关材料参数如 TEMPERATURE EFFECTS、WORK HARD 等。公共块 ELMCOM 包含有对每种材料的材料识别号 “ MATS ”。

另外, 在模型定义部分材料定义的材料特性的参考值可从公共块 MATDAT 得到 :

ET(3)	杨氏模量
XU(3)	泊松比
RHO	质量密度
SHRMOD(3)	剪切模量
COED(3)	热膨胀系数
YIELD(1)	屈服应力
YIELD(2)	ORNL 第 10 个循环屈服应力
YIELD(3)	ORNL 反向塑性屈服应力
YRDR(3)	各向异性塑性正向比例 (direct ratio's for Hill)
YRSR(3)	各向异性塑性剪切比例 (shear ratio's for Hill)
CONDU(3)	传导率
SPHT	比热
CONDV(3)	电阻率
RHOHT	热传导分析用质量密度
EMISV	发射率 (也称黑度系数)
COSTPV	单位体积费用
COSTPM	单位质量费用
PERMEAB(3)	磁透性
RELUCT(3)	磁阻
PERMAIR	气透性
PERMIT(3)	介电常数
ECOND(3)	导电性
VISCOSIT	粘性
TK21	热传导率 21
TK31	热传导率 31
TK32	热传导率 32
R21	电阻系数 21
R31	电阻系数 31
R32	电阻系数 32
C10	Mooney 材料参数 C10
C01	Mooney 材料参数 C01
C11	Mooney 材料参数 C11
C20	Mooney 材料参数 C20
C30	Mooney 材料参数 C30
BULK	体积模量(Mooney, Ogden, Arruda-Boyce, Gent)
AMOHR	Mohr-Coulom6 本构模型的参数
ENTHALPY	焓
PHPERC	相成分
FUTURE(38)	暂不用

1.4 利用 ELMVAR 和 NODVAR 提取单元和节点变量

1.4.1 利用 ELMVAR 提取单元求解结果

要提取单元求解结果，可以采用 ELMVAR 子程序，在一个单元循环之中的任何用户子程序都可以调用该子程序。ELMVAR 可与 MSC.Marc 单元变量、后处理代码联合使用返回计算的单元结果值。ELMVAR 调用语句如下：

```
CALL ELMVAR (ICODE,M,NN,KC,VAR)
```

其中，ICODE 为后处理代码，M 为单元号，NN 为积分点号，KC 为梁、板或壳单元使用的层号，VAR 为所求变量的当前值。

注意：如果要求一个张量，必须要在你的用户子程序中将 VAR 设为一个局部数组。

ICODE 的值在 MSC.Marc 手册 C 卷的模型定义选项部分的 POST 选项中有较详细的定义，因变量太多，本书不再列出。如果 ELMVAR 子程序被单元组集或应力计算阶段中的用户子程序调用，VAR 的值为当前迭代的值，而不一定是收敛的值。ELMVAR 可被以下用户子程序调用：

ANELAS、	ZFLUX、	PLOTV、	UDAMAG、	UMU、	USIGMA、
ANEXP、	FORCEM、	REBAR、	ELDAM、	UNEWTN、	USPCHT、
ANKOND、	GENSTR、	SINCER、	UELOOP、	UOGDEN、	UVOIDN、
ANPLAS、	HOOKLW、	TENSOF、	UENERG、	UPERM、	UVSCPL、
ASSOC、	HOOKVI、	TRSFAC、	UEPS、	UPOWDR、	VSWELL、
CRPLAW、	HYPELA、	UACTIVE、	UFAIL、	UPSTRECH、	WKSPL、
CRPVIS、	HYPELA2、	UACOUS、	UGENT、	URESTR、	YIEL、
CUPFLX、	INTCRD、	UADAP、	UFINITE、	URPFLO、	ZERO、
ELEVAR、	NASSOC、	UARRBO、	UHTCOE、	USELEM、	ELEVEC、
NEWSV、	UCOMPL、	UHTCON、	USHELL、	FILM、	ORIENT、
UCRACK、	UMOONY、	USHRET、			

例如在用户子程序 UADAP 中需要塑性应变张量用于用户定义的网格自适应准则。在此例中不存在壳单元，因而 KC=1 每个单元的积分点数为 4，因而 INTEL=4。在手册 C 卷中查到塑性应变张量后处理代码为 321。塑性应变存于局部数组 EPTEN 中。可以编写以下程序：

```
SUBROUTINE UADAP (M,XORD,DSXT,NCRDMX,NDEGMX,LM,NNODE,USER)
IMPLICIT REAL *8 (A-H, O-Z)
DIMENSION XORD(NCRDMX, *), DSXT(NDEGMX, *), LM(*)
DIMENSION EPTEN (6,28)
c 提取每个积分点上的塑性应变张量
KC=1
INTEL=4
ICODE=321
DO NN=1, INTEL
    CALL ELMVAR(ICODE,M,NN,KC,EPTEN(1,NN))
ENDDO
c 用户定义的网格自适应准则
USER CODE TO DEFINE USER
```

```
RETURN
END
```

1.4.2 利用 NODVAR 提取节点结果

NODVAR子程序可用于提取MSC.Marc 数据库中的节点结果,该子程序可被任何用户子程序调用。NODVAR的调用语句如下:

```
CALL NODVAR(ICOD,NODEID,VALNO,NQNCOMP,NQDATATYPE)
```

注意:如果要得到一个矢量,VALNO 应是一个大到能放下所有NQNCOMP矢量分量的数组。

其中,输入参数为:

NODEID 为节点号;

ICOD 为要求变量的代码。

0 = 坐标值	21 = 外部电流
1 = 位移	22 = 反电流
2 = 转动	23 = 空隙压力
3 = 外力	24 = 外部质量流动密度
4 = 外部弯矩	25 = 反质量流动密度
5 = 反力	26 = 轴承压力
6 = 反弯矩	27 = 轴承力
7 = 流体速度	28 = 速度
8 = 流体压力	29 = 转动速度
9 = 外部流体力	30 = 加速度
10 = 流体反力	31 = 转动加速度
11 = 声压	32 = 模态质量
12 = 外声源	33 = 转动模态质量
13 = 反声源	34 = 接触法向应力
14 = 温度	35 = 接触法向力
15 = 外部热流	36 = 接触摩擦应力
16 = 反热流密度	37 = 接触摩擦力
17 = 电势	38 = 接触状态
18 = 外部电荷	39 = 接触到物体
19 = 反电荷	40 = Herrmann 变量
20 = 磁势	

输出参数为:

VALNO 所求变量的当前值;

NQNCOMP 返回的分量数目;

NQDATATYPE 返回的数据类型:

- 0 = 缺省
- 1 = 模态
- 2 = 屈曲
- 3 = 谐响应实值
- 4 = 谐响应实值/虚值
- 5 = 谐响应幅值/相角

例如,想显示节点总的接触力,即将法向矢量和摩擦矢量相加成为总的接触力矢量。采用用户子程序UPSTNO来实现矢量相加并放在后处理文件中,程序如下:

```

      subroutine upstno(nqcode,nodeid,verno,nqncomp,nqtype,
      * nqaver,nqcomptype,nqdatatype,nqcompname)
      implicit real*8 (a-h,o-z)
      dimension verno(*)
      character*24 nqcompname(*)
c..... 用户代码开始
      dimension verno1(3),verno2(3)
      if (nqcode.eq.-1) then
c... 提取接触法向力矢量存储到数组 verno1
      call nodvar(35,nodeid,verno1,nqncomp,nqdatatype)
c... 提取接触摩擦力矢量存储到数组 verno2
      call nodvar(37,nodeid,verno2,nqncomp,nqdatatype)
c... 将接触法向矢量和摩擦矢量相加
      do 1 i = 1, nqncomp
      verno(i)=verno1(i)+verno2(i)
      1 continue
c... 标识得到的变量 verno 代表的是一个矢量
      nqtype=1
      end if
c..... 用户代码结束
      return
      end

```

1.5 利用内部子程序进行矩阵运算

如果在用户子程序中需要进行矩阵运算,可以调用内部子程序进行矩阵运算,既方便又效率高,下面介绍常用矩阵运算符程序:

- 1) 对矩阵 $A[m,n]$ 进行初始化,使 A 的所有项都等于 rp :

```

call scla(A,rp,m,n,0)
(A(ijk)=rp, ijk=1,m*n)

```

- 2) 对矩阵 $I[m,n]$ 进行初始化,使 I 的所有项都等于 ip :

```

call iscla(I,ip,m,n,0)
(I(ijk)=ip, ijk=1,m*n)

```

- 3) 对矩阵 $A[m,n]$ 和 $B[m,n]$ 进行相加,将结果存于 $C[m,n]$:

```

call gmadd(A,B,C,m,n)
(C(ijk)=A(ijk)+B(ijk), ijk=1,m*n)

```

- 4) 对矩阵 $A[m,n]$ 减去矩阵 $B[m,n]$,将结果存于 $C[m,n]$:

```

call gmsub(A,B,C,m,n)

```

$(C(ijk)=A(ijk)-B(ijk), \text{ijk}=1, m*n)$

5) 对矩阵 $B[m,n]$ 乘标量 s 后与矩阵 $A[m,n]$ 进行相加, 将结果存于 $C[m,n]$:

```
call gmad(A,B,C,s,m,n)
  (C(ijk)=A(ijk)+s*B(ijk), ijk=1,m*n)
```

6) 对矩阵 $A[m,n]$ 和 $B[m,n]$ 进行相乘, 将结果存于 $C[m,n]$:

```
call gmprd(A,B,C,m,n)
  (((C(i,j)=A(i,k)*B(k,j), k=1,n), j=1,l), i=1,m)
```

7) 对矩阵 $A[m,n]$ 进行转置, 将结果存于 $C[m,n]$:

```
call gmtra(A,C,m,n)
  ((C(i,j)=A(j,i), i=1,n), j=1,m)
```

8) 对矩阵 $A[m,n]$ 进行转置后与 $B[m,n]$ 进行相乘, 将结果存于 $C[m,n]$:

```
call gtprd(A,B,C,m,n,1)
  (((C(i,j)=A(k,i)*B(k,j), k=1,m), j=1,l), i=1,n)
```

9) 将矩阵 $A[m,n]$ 复制到矩阵 $C[m,n]$:

```
call mcpy(A,C,m,n,0)
  (C(ijk)=A(ijk), ijk=1,m*n)
```

10) 对矩阵 $B[m,n]$ 乘标量 s , 将结果存于 $C[m,n]$:

```
call smpy(A,s,C,m,n,0)
  (C(ijk)=s*A(ijk), ijk=1,m*n)
```

11) 对矩阵 $A[n,n]$ 进行求逆, 将矩阵 $A[n,n]$ 的行列式存于 d , 将 A 的逆 A^{-1} 存于 A 中, 将 $A^{-1}(n,n)*B(n,m)$ 存于 B 中。

注意, n 不能大于 15。如果 $m=0$, 只对矩阵 $A[n,n]$ 进行求逆。

```
call invert(A,n,B,m,d,n)
```

1.6 用户子程序调用原理

有些用户子程序需要在界面上指定, 如要用用户子程序指定分布载荷时, 在边界条件定义的菜单中要选上用户子程序。有些用户子程序则不需定义而直接调用即可。在用 Patran 作前处理时, 由于不能用菜单指定用户子程序名, 有时需要在界面上采用常规定义方式并给定一个载荷类型为假设值, 然后修改 MSC.Marc 输入文件将载荷类型改为通过用户子程序定义, 详见 1.7 节。

MSC.Marc 提供以 Fortran 语言编写的用户子程序模板文件。用户可以使用时, 按 Fortran 编程规则, 根据需要填充子程序模板文件。程序中包含一个或若干个用户子程序时, 会以这些用户定义的子程序代替缺省的相应子程序, 重新生成新的执行程序, 使程序以用户期望的方式运行。

在软件安装目录下有 MSC.Marc 的执行文件, 如 d:\msc\marc2003\marcs.exe。重新

生成执行代码，以用户子程序文件名来命名，如用户子程序文件名为 plate.f，编译产生的文件为 plate.obj，执行文件为 plate.exe。在缺省条件下，执行文件在作业结束后自动删除，用户可以选择保留执行文件或采用原已保留的文件进行运算。

用户可以直接在界面中运行程序，也可以用命令行运行，作用相同。下面为命令行中三种常见命令。

(1) run_MSC.Marc -jid e2x14 -user u2x14

输入数据文件为 e2x14.dat，执行文件为用户文件 u2x14.f 编译、连接形成 u2x14.MSC.Marc，不保留执行文件 u2x14.MSC.Marc。

(2) run_MSC.Marc -jid e2x14 -user u2x14 -save y

输入数据文件为 e2x14.dat，执行文件为用户文件 u2x14.f 编译、连接形成 u2x14.marc 并将执行文件 u2x14.marc 保留。

(3) run_MSC.Marc -jid e2x14a -prog u2x14

利用 (2) 已生成的执行文件 u2x14.marc 进行分析。

最好在安装 Marc 以前先安装 Fortran 编译器，不同的版本、不同平台支持的编译器是不同的，可以查看各版本的版本信息说明。对于 Marc 2003 版，其在 Windows 操作系统的编译器为 Compaq Virtual Fortran 6.0a (或 6.5)。

1.7 应用举例——移动载荷

1.7.1 分析模型描述

方板宽为 $L = 10.0$ ，厚为 $h = 0.1$ ，四周固支，承受位置随时间变化的分布载荷，分布载荷的幅值为 2，作用范围宽为 $b = 1.0$ 的小方块。分布载荷以 $\sqrt{2}$ 的速度沿板的对角线移动，如图 1.1 所示。设板的材料为线弹性，杨氏模量为 $E = 2.0 \times 10^5$ ，泊松比为 $\nu = 0.28$ 。分析涉及的总时间为 10，因而载荷从板的一端移至另一端。分 20 均匀增量步进行线弹性分析。采用 75 号壳单元。

移动载荷，也即非均匀分布载荷，在 MSC.Marc 中可以通过用户子程序 FORCEM 来实现。通过使用该子程序，用户可以指定一个随着位置坐标或者时间变化的载荷。

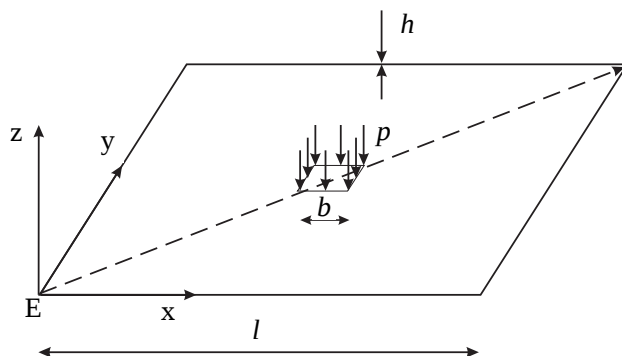


图 1.1 受移动载荷作用的方板

1.7.2 用户子程序的选择及其格式

对于二维的单元，子程序的格式如下：

```
SUBROUTINE FORCEM (P,X1,X2,NN,N)
IMPLICIT REAL *8 (A-H, O-Z)
DIMENSION N(7)
      user coding

RETURN
END
```

下面说明子程序输入和输出参数的意义，对于输入参数有：

X1	积分点的第一个坐标值
X2	积分点的第二个坐标值
NN	积分点号
N(1)	单元号
N(2)	载荷类型参数
N(3)	这里不用
N(4)	这里不用
N(5)	分布载荷序号
N(6)	这里不用
N(7)	内部单元号

输出参数：

P 用户定义的在要计算的积分点上的分布载荷的数值大小

对于三维的单元和类型为 22、49、72、75、138、139 和 140 的壳单元，子程序的格式如下所示：

```
SUBROUTINE FORCEM (P,X1,X2,NN,N)
IMPLICIT REAL *8 (A-H, O-Z)
DIMENSION X1(3), X2(3), N(7)
      user coding

RETURN
END
```

输入参数：

X1(3)	积分点的 x、y、z 坐标值
NN	积分点号
N(1)	单元号
N(2)	载荷类型参数
N(3)	这里不用
N(4)	这里不用
N(5)	分布载荷序号

输出参数：

P 用户定义的在要计算的积分点上的分布载荷的数值大小。在某些情况下需要指定载荷的方向（如壳单元或者梁单元）

X2(3) 载荷方向向量

1.7.3 用户子程序代码

从 path/user/目录中将 forcem.f 复制到当前工作目录中并改名为 plate.f, 在程序中对分布载荷的幅值和载荷方向进行合理的定义。变量为增量步开始时的时间和时间增量, 它们均是公共块 CREEPS 的一部分。FORCEM 可以编写如下:

```

      subroutine forcem(press,th1,th2,nn,n)
      c
      implicit real*8 (a-h,o-z)
      c
      c...   在单元上定义非均匀分布载荷
      c
      c...   press           分布载荷的数值大小
      c...   th1             积分点的坐标(x,y,z)
      c...   th2             载荷方向向量
      c...   nn              积分点号
      c...   n(1)            单元号
      c...   n(2)            载荷类型参数
      c...   n(3)            这里不用
      c...   n(4)            这里不用
      c...   n(5)            分布载荷序号
      c...   n(6)            这里不用
      c...   n(7)            内部单元号
      c
      c...   该程序是为了模拟一个移动分布载荷
      c...   假定载荷区域是一个正方形, 边长为 b
      c...   载荷作用区域中心以不变的速度 v 沿着
      c...   (x,y)平面上的直线 x=y 移动
      c...   在模型中选项 FOLLOW FORCE 和 AUTO LOAD 一起被激活,
      c...   所以在增量步的末尾必须定义总的载荷
      c
      dimension th1(3),th2(3),n(7)
      c
      c...   包含公共块 "creeps", 这样可以获得总的时间变量(cptim)
      c...   和增量步时间变量(timinc)
      include '../common/creeps'
      c
      c...   定义边长 b 和移动速度 v 的数值大小
      b=1.0d0
      v=sqrt(2.0d0)
      c
      c...   计算确定当前时刻载荷中心的位置
      distan=v*(cptim+timinc)
      xc=0.5d0*sqrt(2.0d0)*distan
      yc=0.5d0*sqrt(2.0d0)*distan
      c
      c...   计算确定当前时刻载荷作用区域在(x,y)平面上的边界位置

```

```

    xmin=xc-b/2.0d0
    xmax=xc+b/2.0d0
    ymin=yc-b/2.0d0
    ymax=yc+b/2.0d0
c
c...   判断积分点是否在载荷作用区域内
c...   如果是的话就在该积分点定义非零载荷
    press=0.0d0
    if (th1(1).le.xmax.and.th1(1).ge.xmin.and.
$      th1(2).le.ymax.and.th1(2).ge.ymin) press=2.0d0
c
c...   定义载荷作用的方向
    th2(1)=0.0d0
    th2(2)=0.0d0
    th2(3)=-1.0d0
c
    return
end

```

1.7.4 采用 Mentat 做前后处理

1. 几何建模和网格划分

该模型的有限元网格如图 1.2 所示,采用 10×10 的网格。激活并设置格栅尺寸,直接在格栅上取点产生一个四边形面,然后将面转换成单元,在转换之前先指定各个方向单元数目。重复节点由 SWEEP 选项删去,而后用 RENUMBER 选项给单元节点重新编号。操作过程如下:

```

MESH GENERATION
  GRID (on)
  COORDINATE SYSTEM SET:
    U DOMAIN 0 10
    U SPACE 1
    V DOMAIN 0 10
    V SPACE 1
  RETURN
  SRFS ADD:
    POINT (0.0,0.0,0.0)
    POINT (10.0,0.0,0.0)
    POINT (10.0,10.0,0.0)
    POINT (0.0,10.0,0.0)
  CONVERT
    SURFACE TO ELEMENTS
    ALL EXISTING
  RETURN
  GRID (off)
  SWEEP

```

```
ALL  
RETURN  
RENUMBER  
ALL  
RETURN  
MAIN
```

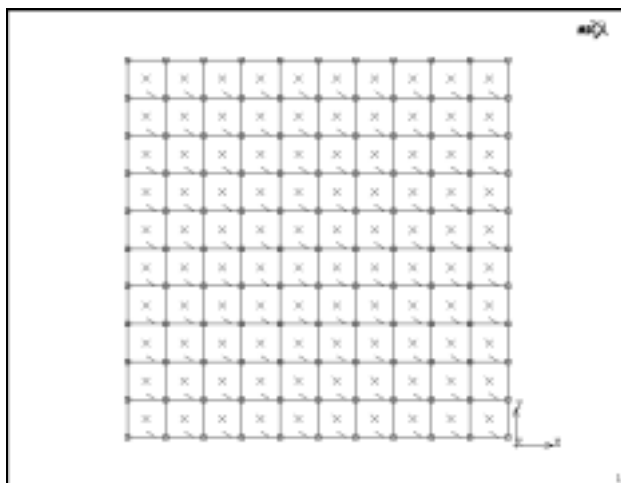


图 1.2 有限元网格

2. 定义边界条件

本模型的边界条件包括：

1) 位移边界条件：固定四周节点的平移和转动自由度，如图1.3所示。

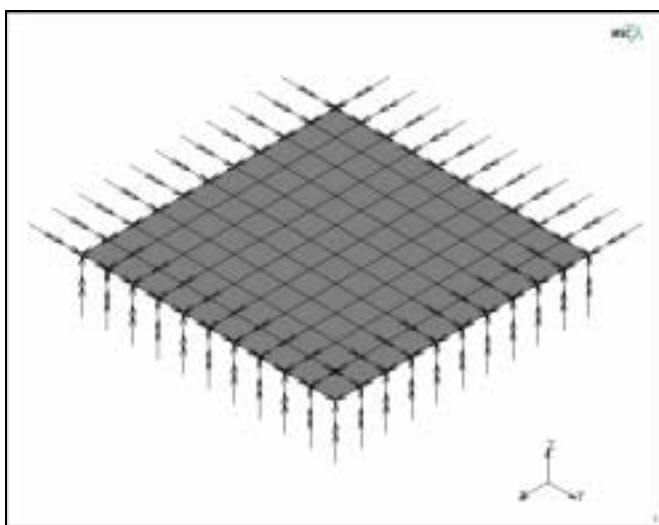


图 1.3 位移边界条件

2) 载荷边界条件：采用用户子程序FORCEM将压力载荷施加到所有单元上，如图1.4所示。

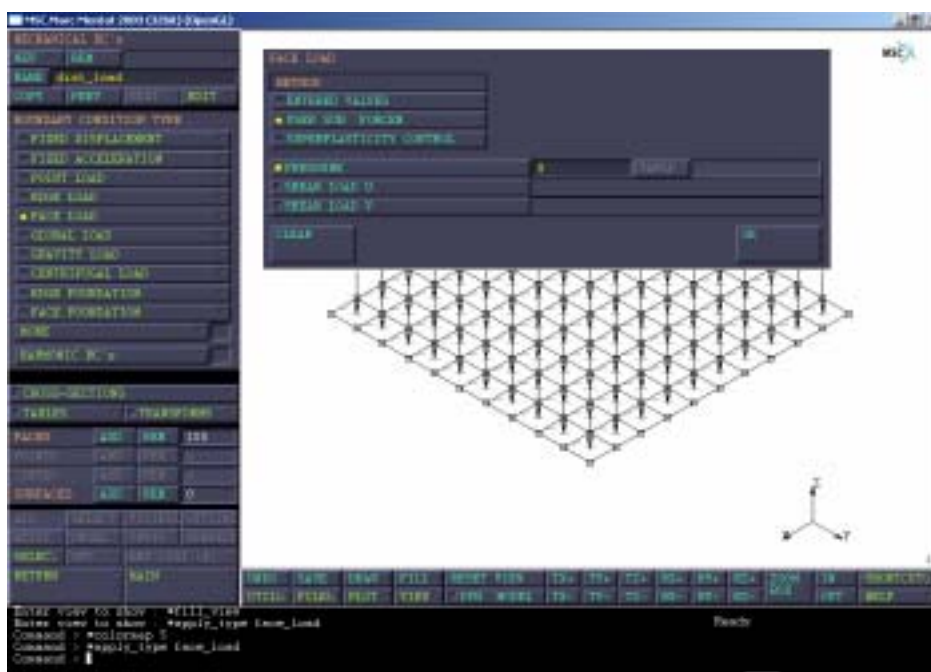


图 1.4 压力载荷

BOUNDARY CONDITIONS

MECHANICAL

FIXED DISPLACEMENT

ON DISPLACEMENT X

ON DISPLACEMENT Y

ON DISPLACEMENT Z

ON ROTATION X

ON ROTATION Y

ON ROTATION Z

OK

NODES ADD

(选择所有边界上的节点)

END LIST

NEW

FACE LOAD

USER SUB.FORCEM

ON PRESSURE

OK

FACES ADD

ALL EXISTING

RETURN

MAIN

3. 定义材料性质

板的材料为线弹性、各向同性材料。

```

MATERIAL PROPERTIES
  ISOTROPIC
    YOUNG'S MODULUS
      2.0E5
    POISSON'S RADIO
      0.28
    OK
  ELEMENTS ADD
    ALL EXISTING
  RETURN
MAIN

```

4. 定义几何特性

先定义 3D 壳单元板厚为 0.1m，然后施加到所有的单元上。

```

GEOMETRIC PROPERTIES
  3-D
    SHELL
      THICKNESS
        0.1
      OK
    ELEMENTS ADD
      ALL EXISTING
    RETURN
  RETURN
MAIN

```

5. 定义静力分析工况

```

LOADCASES
  MECHANICAL
    STATIC
      TOTAL LOAD TIME
        10
      FIXED #STEPS
        20
      OK
    RETURN
  RETURN
MAIN

```

6. 作业提交

单元类型为 75 号壳单元，激活跟随力选项。使用跟随力选项，用户能很容易地在单元上定义载荷；它取决于时间和积分点的位置，节点等效力就是根据积分来计算得到的。分布载荷的幅值在 FORCEM 中设为 0 或 2.0，如图 1.5 所示。



图 1.5 激活跟随力选项

JOBS

MECHANICAL

lcase1

ANALYSIS OPTIONS

NO FOLLOWER FORCE -> FOLLOWER FORCE

OK

OK

ELEMENT TYPES

MECHANICAL

3D MEMBERANE/SHELL

75

OK

ALL EXISTING

RETURN

RETURN

SAVE

RUN

USER SUBROUTINE FILE

(选择用户子程序文件, 如图 1.6 所示)

SUBMIT1

MONITOR

OK

RETURN

MAIN

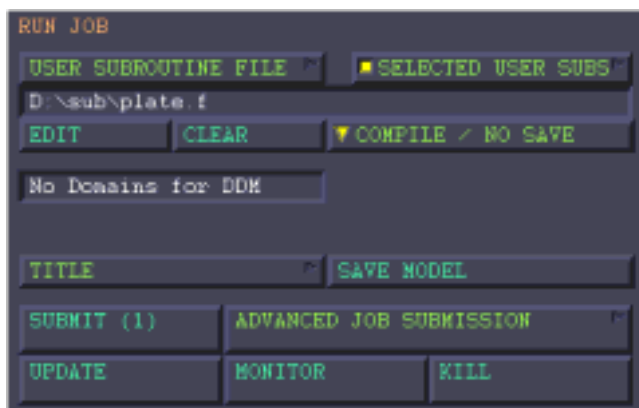


图 1.6 选择用户子程序文件

7. 计算结果

计算通过，可以进行后处理，图 1.7 为第 10 增量步的 Z 方向外力等值云图，图 1.8 为第 10 增量步的 Z 方向的位移等值云图。

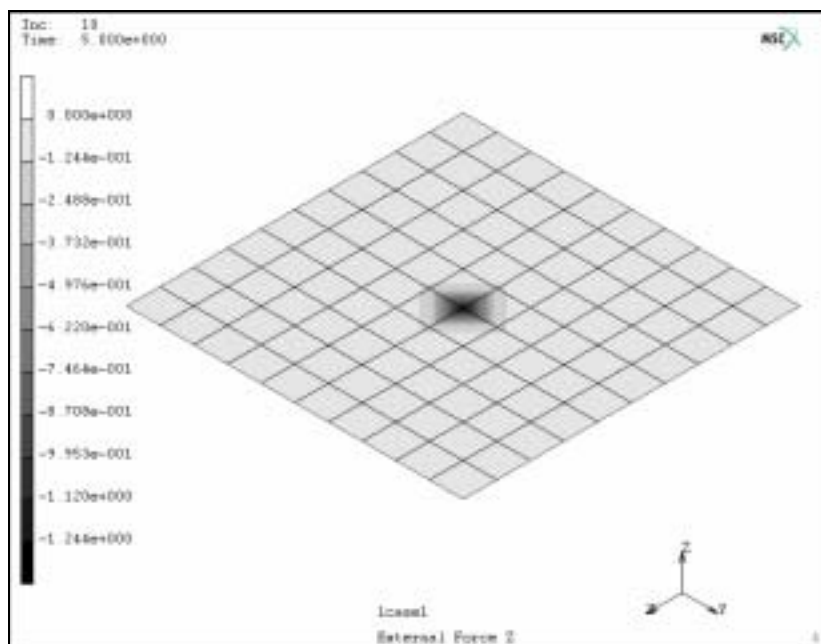


图 1.7 第 10 增量步的 z 方向外力算值云图

RESULTS

OPEN DEFAULT

DEF ONLY

COUTOUR BAND

PLOT

NODES (off)

RETURN