

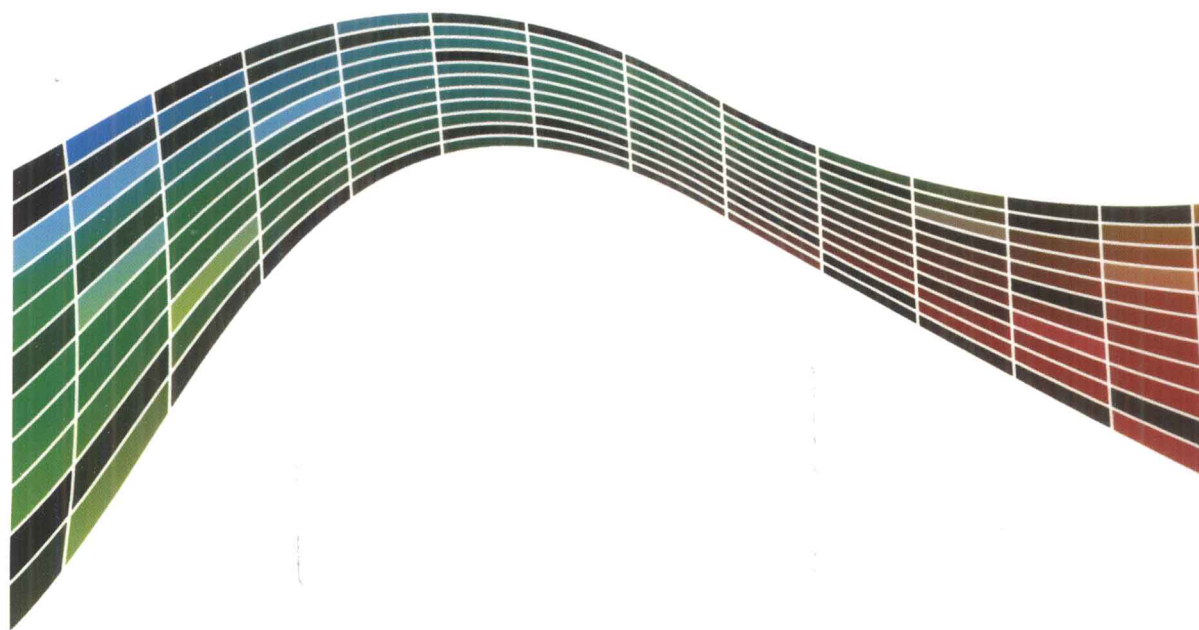
岩土力学与工程国家重点实验室开放课题
国家863机械系统动力学CAE平台项目



FEPG

有限元应用深入剖析

白冰 李小春 著



清华大学出版社

FEPG 有限元应用深入剖析

白 冰 李小春 著

清华大学出版社
北 京

内 容 简 介

本书围绕一个非线性热传导问题,详细介绍了FEPG的使用过程,并以研究和剖析的视角阐述了FEPG背后的机理。主要内容包括:FEPG的详细安装,实例的完整使用过程,FEPG与GID的接口,程序的自动生成过程,源代码详解以及有限元理论,脚本与源代码的对应关系。通过本书,揭开用户使用FEPG中的疑惑,帮助用户知其然也知其所以然,最终达到灵活、高效使用FEPG为科研服务的目的。

为方便读者使用,本书配有光盘,内含本书实例程序、FEPG一些批命令文件、FEPG网络版客户端程序以及一些对读者有用的参考文档。

本书适用于期待对FEPG技术有深入了解的用户,也可供高校理工类各专业教师和学生使用。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

FEPG 有限元应用深入剖析 / 白冰, 李小春著. —北京: 清华大学出版社, 2011.9
ISBN 978-7-302-25716-5

I. ①F… II. ①白… ②李… III. ①有限元法—程序设计 IV. ①O241.82-39

中国版本图书馆CIP数据核字(2011)第107508号

责任编辑:王峰松 战晓雷

责任校对:徐俊伟

责任印制:王秀菊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦A座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62795954,jsjic@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015,zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京鑫丰华彩印有限公司

装 订 者:三河市新茂装订有限公司

经 销:全国新华书店

开 本:185×260 印 张:14 字 数:349千字

附光盘1张

版 次:2011年9月第1版

印 次:2011年9月第1次印刷

印 数:1~4000

定 价:39.00元



研究是最好的学习

序

计算机硬件技术获得了极大发展，我国的硬件研究也已跃居世界一流水平。而计算软件和软件人才正成为瓶颈。作为计算科学重要组成部分的有限元软件目前被国外厂商垄断，国产有限元软件非常稀缺，严重制约了我国自主可控计算技术的发展。可喜的是，中科院数学与系统科学研究院的梁国平先生开发的具有完全自主知识产权的有限元自动生成系统 FEPG，给打破国外垄断带来了曙光，也为高性能计算软件稀缺提供了一个新颖、高效的解决途径。

传统通用有限元软件的“通用性”是通过事先维护庞大的方程库程序、算法库程序等方式实现的。但相对于实际问题的千变万化，这仍然不是本质的。FEPG 的特色在于以程序自动生成的方式高效构建新模型、实现新算法。即将实际问题在这两点上的多变性留给用户自己解决，而这样做所带来的重复编程负担则由程序自动生成机制化解。从这一点上说，FEPG 通过其程序自动生成技术的灵活性实现了有限元软件真正的通用性。对于科学研究中模型和算法多变的用户，熟练掌握 FEPG 尤其可能提高效率。

FEPG 的理论依据是有限元方法的一般数学基础，要求用户有较好的有限元理论储备。对于广大非计算数学专业的用户而言，深入掌握 FEPG 可能会有一定的困难。本书作者根据多年使用和研究 FEPG 的经验，尝试以研究和剖析的视角来阐述 FEPG 背后的机理，这在有限元软件类著作中立意颇为新颖。希望本书的出版对于促进国产高性能有限元软件的推广、提高用户素质起到一定的作用。是为序。

中科院数学与系统科学研究院研究员，中国科学院院士 林群

2011.2

前 言

2004 年，作者在中科院研究生院张怀教授处第一次接触 FEPG，便被其能够自动生成有限元源代码的特色所深深吸引，内心充满了激动。当时的想法主要有两点：第一，如果掌握了 FEPG，就可以控制其源代码，从而更加灵活自由地根据自己的意愿去修改源程序，毕竟无论怎样高度灵活的闭源软件，其灵活性都不可能同直接控制源代码的方式相比；第二，如果掌握了 FEPG，就可以利用它生成的源代码高效地开发自己的有限元软件，甚至销售它。我想这也是许多用户被 FEPG 吸引的重要原因。

FEPG 作为在源代码这个层次上的软件，在使用上与流行的大型商业有限元软件有较大区别。一般而言，使用 FEPG 进行自己的创新工作，要求用户自己提出问题的数学模型，设计算法，推导弱解积分形式，填写脚本文件，生成有限元代码，编译连接程序，前处理建模，运行程序进行计算以及后处理等较多环节，因此，对用户有较高的要求。当前许多用户是非计算数学专业（多是工科）的研究生，这些同志在本科阶段通常对自己专业领域的有限元法也仅有初步的了解，而对上述各环节尤其是有限元方法的数学基础则了解得更少，因此，在学习使用 FEPG 的过程中遇到困难就是自然的了。加之 FEPG 的使用过程不甚直观，初学者在一段时间后对其缺乏兴趣也是可以理解的。另一方面，即便是一些较为熟练的用户，在使用 FEPG 的过程中，也常有这样的困惑，就是在实现自己特定功能时，往往感觉许多环节总有难以把握的“谜”点，用户的实现究竟是对是错实难控制。人总有把握事物根源的冲动，并且只有把握了根源才能真正相信，然而这些若始终不可得，则内心必将大为困惑。

笔者认为，基于一个典型算例，将 FEPG 的代码生成过程、脚本文件、程序代码与有限元理论之间的对应关系理清楚了，就能够从根本上掌握和控制 FEPG，然后充满信心地去使用 FEPG 提供的各种工具高效、自由地开发自己的程序。然而要真正成为 FEPG 的高手，则有赖于用户在专业能力、有限元理论和算法、编程技术等各方面素质的普遍提高。笔者在这几年学习和使用 FEPG 的过程中深深感到，FEPG 在这些方面“逼着”自己往前走，促进了自身素质的进步，因此对 FEPG 也产生了深厚的感情。这几年积累了点滴经验，我们希望把这些经验集中起来与读者分享，共同切磋。书中谬误和不当之处在所难免，欢迎读者批评指正。

最后，我们感谢国家 863 项目（2009AA044501）和岩土力学国家重点实验室开放课题项目（Z110803）对本书出版的支持。对梁国平先生及夫人林继玲女士的热情指导与帮助，北京飞箭软件有限公司的大力支持，尤其是中国科学院林群院士百忙之中拨冗作序、关心后学表示衷心感谢。

作 者

2011.2 于湖北 武汉

目 录

第 1 章 FEPG 简介与安装	1
1.1 FEPG 是什么	1
1.2 学习 FEPG 的起点	1
1.3 为什么用 FEPG, 谁用 FEPG	2
1.4 FEPG 的特色及其应用模式	3
1.4.1 定量科学研究的数学模型范式	3
1.4.2 FEPG 的特色及应用模式比较	4
1.4.3 FEPG 自由模式及其文件准备	6
1.5 有限元语言	7
1.6 FEPG 的历史与版本	8
1.6.1 FEPG 的历史	8
1.6.2 FEPG 产品系列	9
1.6.3 FEPG .PC.V6 的软硬件环境	10
1.7 从安装开始	11
1.7.1 安装及配置经验	11
1.7.2 留意! 一个编译细节	13
1.7.3 FEPG 不必重新安装	14
1.8 安装目录及文件概览	14
1.9 网络版与单机版的比较及安装	14
1.9.1 网络版与单机版使用模式比较	14
1.9.2 网络版 IFEPG 客户端的安装	16
第 2 章 一个完整实例——基于用户文件法的开发	18
2.1 明确你的角色	18
2.2 实际问题	18
2.3 写出数学模型	19
2.4 推导弱形式	20
2.5 FEPG 实现	22
2.6 b2p 命令与元件程序的合并	27
2.7 第一次运行你的程序——程序应用阶段	28
2.8 程序的修改和调试	29
2.9 脱离 FEPG	30
第 3 章 前后处理与 pre 文件	33
3.1 GID 前处理及其界面命令	33

3.2	GID 后处理及其界面命令	35
3.3	用 GID 作通用前后处理器的一般原理	37
3.4	FEPG 中使用 GID 的设置	40
3.4.1	设置过程探秘	40
3.4.2	从 fepg1.bas 文件到 giddisp 文件	42
3.4.3	FEPG.BAT——食之无味，弃之“违法”	43
3.4.4	体验设置——换一种口味	44
3.5	pre 文件及其与 GID 界面的对应	45
3.6	计算命令流批处理文件探秘	47
3.6.1	批处理文件内容分析	47
3.6.2	批处理文件的来源	48
3.6.3	元件程序的输入输出文件——台面上	49
第 4 章	程序自动生成过程	51
4.1	程序的自动生成过程	51
4.2	Fortran 源程序的编译和连接	60
4.2.1	单元子程序的编译	60
4.2.2	骨干程序的编译和连接	62
4.3	FEPG 中的文件体系	68
4.3.1	文件总体分类	68
4.3.2	开发阶段文件	69
4.3.3	使用阶段文件	70
4.4	几点说明	71
第 5 章	理论准备——实例的 FEPG 有限元公式推导	72
5.1	强解积分形式	72
5.2	方程的线性化	72
5.3	弱解积分式	73
5.4	区域剖分（网格剖分或空间离散）	73
5.5	单元插值	73
5.6	单元有限元方程	74
5.7	时间离散化	75
5.8	本质边界条件处理、节点集中荷载（集中热源）、总体方程组右端项	76
5.9	总体合成——总体系数矩阵的叠加	77
5.10	线性代数方程组求解	78
5.11	非线性迭代的控制	78
5.12	时间步控制	78
5.13	解后计算	79

第 6 章 应用程序框架：鸟瞰	80
6.1 应用程序框架	80
6.2 Fortran 程序组成	81
6.2.1 Fortran 语言的程序组成	81
6.2.2 什么是“元件程序”	83
6.3 编译命令与文件依赖关系	85
6.4 源程序的输入/输出文件内容概貌	86
6.4.1 元件程序的输入/输出文件——台上台下	86
6.4.2 FEPG 内核输出文件	86
6.4.3 GID 的输出文件 np.dat	87
6.4.4 fgidpre.for 的输出文件	88
6.4.5 starta.for 的输出文件	91
6.4.6 bft.for 的输出文件	92
6.4.7 enpa.for 的输出文件	93
6.4.8 solve.for 的输出文件	95
6.4.9 unpa.for 的输出文件	95
6.4.10 gidpost0.for 的输出文件	96
第 7 章 应用程序框架：剖析	97
7.1 前接口程序 fgidpre	97
7.1.1 程序功能描述	97
7.1.2 调用关系图	97
7.1.3 进入源代码	98
7.1.4 关键技术详解	103
7.2 start 元件程序	104
7.2.1 程序功能描述	104
7.2.2 调用关系图	104
7.2.3 进入源代码	105
7.2.4 关键技术详解	110
7.3 bft 元件程序	113
7.3.1 程序功能描述	113
7.3.2 调用关系图	113
7.3.3 进入源代码	113
7.3.4 关键技术详解	119
7.4 e 元件程序	119
7.4.1 程序功能描述	119
7.4.2 调用关系图	119
7.4.3 进入源代码	121
7.4.4 关键技术详解	127

7.5 单元子程序 aeq4g2.for	133
7.5.1 程序功能描述	133
7.5.2 调用关系图	133
7.5.3 进入源代码	134
7.5.4 关键技术详解	140
7.6 单元子程序 agl2.for	150
7.6.1 程序功能描述	150
7.6.2 调用关系图	150
7.6.3 进入源代码	151
7.6.4 关键技术详解	151
7.7 库文件 CCSHAP.FOR	158
7.7.1 进入源代码	158
7.7.2 关键技术详解	161
7.8 solv 元件程序	163
7.8.1 程序功能描述	163
7.8.2 调用关系图	163
7.8.3 进入源代码	163
7.8.4 关键技术详解	169
7.9 u 元件程序	173
7.9.1 程序功能描述	173
7.9.2 调用关系图	173
7.9.3 进入源代码	173
7.9.4 关键技术详解	177
7.10 后接口程序 gidpost0	179
7.10.1 程序功能描述	179
7.10.2 调用关系图	180
7.10.3 进入源代码	180
7.11 串起代码之纲——批命令文件	185
7.12 FEPG 程序的内存分配	186
 第 8 章 理论-脚本-程序的对应	 188
8.1 理论-脚本-代码的对应	188
8.2 脚本说明	191
 第 9 章 多场耦合原理	 193
9.1 场与耦合	193
9.1.1 场与耦合概述	193
9.1.2 耦合的数学实质: 微分方程组	193
9.1.3 多场耦合实例	194

9.1.4 “FEPG 场”与耦合	194
9.2 微分方程组求解的有限元公式推导	196
9.3 多场问题的步骤	198
9.4 FEPG 对耦合的处理	198
9.5 FEPG 的文件命名规律	200
第 10 章 杂谈	202
10.1 FEPG 问题对策——错误定位技巧	202
10.2 FEPG 用户的“段位”	203
10.3 代码重用、自动生成与开发效率	203
10.4 FEPG 中一些关键字的含义	204
10.5 FEPG 问答	204
参考文献	208

第1章 FEPG 简介与安装

1.1 FEPG 是什么

或许无需赘言 FEPG 是什么，读者可能通过各种途径已经对 FEPG 有了一定的了解。但不能排除有些读者的确是第一次接触它。FEPG 是“Finite Element Program Generator”的缩写。顾名思义，它是有限元软件的生成器，因而称为“有限元程序自动生成系统”。简单地说，这是一套有限元软件，然而顾名思义未必准确，其实它还是一套支持有限体积法的软件。也就是说，这是一套可以生成有限元或有限体积法源程序的软件。这套系统是利用有限元方法或有限体积法进行数值计算的软件包。

该系统由中国科学院数学与系统科学研究院梁国平先生开发，是一套独具特色的国产高性能有限元数值计算软件。这些特色将随着本书的叙述逐渐得以展现并被用户体会到。笔者根据自己的理解和体会用下面几个软件开发领域流行的术语从不同的角度对其做总体概括。

FEPG 的总体技术特征可以概括为以下几点：

- CAG: Code Automatic Generator

FEPG 是代码生成器，是生成软件（源代码）的软件。

- IDE: Integrated Development Environment

FEPG 是一套有限元（有限体积）程序的集成开发环境。

- CAE: Computer Aided Engineering

FEPG 是一套计算机辅助工程软件。

- PSE: Problem Solving Environment

FEPG 对用户来说是一套集成的工作平台和问题求解环境。

- A reusable code repository

FEPG 是一套高效、可重用的代码仓库。

初学者可能对于上述概括不太理解，希望读者带着这些疑惑，在后续的学习中不断体会和领悟。鉴于 FEPG 的高度灵活性，对于 FEPG 究竟是什么这个问题，完全应由用户根据自己对其理解和体会归纳出来。用户体会越深，使用越灵活，越表明在使用 FEPG 上正在逐步达到较高层次。

1.2 学习 FEPG 的起点

严格来说，不同用户有不同的基础，也都有适合自己的学习方法。但也有不少用户反映 FEPG 难学，或者难以学得深入。甚至一些用户在使用一段时间后，感觉其纷繁复杂而无从下手，或者困惑于究竟是否要先完整学习有限元理论而裹足不前。笔者认为，拥有工

科本科的知识基础（有限元基础，计算机知识基础），再适当补充有限元的基本数学知识都是可以学好的，问题的关键是找到 FEPG 学习的起点。

我们的工作和学习常常因为没有积极的进步（如成果、成就感等）作为激励而缺乏兴趣、难以坚持。很多人执迷于电子游戏的主要原因正是可以随时获得“晋级”等“奖赏”作为动力。另一方面，这些“晋级”或“奖赏”所带来的成就感又逐渐成为游戏玩家的目标。这就提示我们，明确的目标和不断进步的正反馈是一个重要的动力之源。对于大多数用户来说，学习 FEPG 不是目标，使用 FEPG 解决专业问题才是目的。在学习 FEPG 时最好一开始就结合自己的研究任务，也就是直接把待解决的问题作为自己的学习实例，而不是一味地做书本上现成的简单实例。这样，使用 FEPG 就有了明确的目标。在解决问题这一目标的吸引下，边用边学，遇到问题随时思考、咨询。到了最后目标问题解决之日，FEPG 也就完全学会了。

当然，可能有些用户习惯于先把所有有限元知识都学懂了才敢学 FEPG，然后再去解决实际问题。但笔者认为，这不是研究阶段的工作方法。学习的主要目的还是为了解决问题，在问题没有解决的时候盲目地学习自己认为可能用到的知识将会花费过多的时间，最后可能会发现对解决问题并没有直接帮助。因此，效率最高的方式就是在有了必需的基础后，边干边学边思考，把学习 FEPG 贯穿于研究、解决具体的实际任务当中。

1.3 为什么用 FEPG，谁用 FEPG

应用有限元方法和程序，当前不外乎几种情况：一种是购买和使用大型商业有限元程序；一种是完全自己开发和使用有限元程序；第三种则是从别处（如导师或其他人）获得别人的有限元程序代码，在使用过程中加以修改丰富以满足自己的要求。另一方面，从需求者的角度而言，可以分为两类，一类是对灵活性要求较高的，比如科研应用；一类是对灵活性的要求不那么高的，比如多数工程领域的应用。

对用户而言，软件只是实现目标的工具。然而目前软件工具真可谓汗牛充栋，即便同一个领域，比如有限元领域，也存在非常多的选择。每个软件都有自己的特色，也都有自身的局限性。选择软件一看能否满足自己的需求，二看该软件能否更好地满足需求，比如能否更高效地、更高质量地完成自己的目标。即便如此，如果没有前人指导，面对众多软件做出一个合理的选择有时也的确是一件费心的事情。

当前国际上有大量功能强大的有限元软件包可供使用。ANSYS, ABAQUS, ADINA, NASTRAN 等鼎鼎大名的大型通用有限元软件占据了国内有限元软件的很大市场。在这种情况下，为什么国产的 FEPG 系统日益获得了较多的应用呢？事实上，许多学者在做研究时发现，尽管商业通用软件功能强大，使用方便友好，但是，在处理实际问题中的需求是千变万化的，很多情况下，这导致通用软件并不完全通用。比如，研究人员想修改程序的算法，或者想修改一下控制方程，甚至有时连想增加一个补充方程（如本构模型）都很难做到。这是由于商业软件不是完全开放的，用户不能直接修改程序的源代码。仅仅依靠用户子程序或内嵌语言等开放接口所提供的灵活性是远远不够的。

事实上，这些大型商业软件的特点是，通常只针对有限的几类特定问题（特定的数学模型），一旦编译封装成可执行代码，程序就无法改变，其功能便固定了。然而用户在处理实际问题中的需求是千变万化的，特别是在科学研究中，这种变化更是难以预测。尽管这

些商业软件提供了二次开发的接口,并且进行不断的升级,但是对用户而言它始终是黑匣子,因此仍然无法适应实际问题的多样性和多变性,尤其是控制方程发生变化的场合。程序的使用者无法完全地直接操纵软件内部代码是这类软件的主要缺点。事实上,从大型商业程序出现之前直至现在,国内外许多学者都在针对各自的研究领域编写自用的有限元程序。这样,在研究中可以随时根据需要而自由地修改代码,具有极大的灵活性和适用性。甚至一些具有共享精神的学者将程序源代码奉献出来,这就是开源有限元程序。当前,在网络的开源社区甚至一些机构将有限元程序源代码公布出来给大家免费使用、测试,有效地弥补了商业闭源软件的缺点,极大地促进了有限元法的研究和普及。然而这类开源程序往往注释不够,其他用户难以理解诸多代码细节,也给用户修改和使用程序造成了一定的困难。

FEPG 不仅提供源代码,而且不用像使用常规开源代码那样在纷繁芜杂的代码中穿梭,因为它提供了一套代码快速开发机制,用户只须填写少量的脚本语言文件,就可以自动生成完整的源代码。从这个角度讲, FEPG 实际上具有高效地帮助用户构建有限元程序系统的能力,并且十分容易维护和修改^[1]。因此,对于进行创新性科研并且对软件的灵活性要求较高的研究人员、有限元程序内核的开发人员、特别是有限元教学中使用 FEPG 是非常合适的。当然,并非只有这类人才适合用 FEPG,实际上,任何使用有限元方法进行数值计算和模拟的用户都可以使用 FEPG。

1.4 FEPG 的特色及其应用模式

1.4.1 定量科学研究的数学模型范式

科研讲求创新,研究新问题,提出新思想,利用新方法。这就决定了科研中的问题是千变万化的。定量科学研究是对事物由感性(定性)认识上升到理性(定量)认识的一个重要手段。定量科学研究的一个典型范式就是基于数学模型的研究方法,如图 1-1 所示。

当前大量的科学研究是基于上述范式的。这就意味着实际研究中具有大量有限元程序的应用需求。这些大量的需求又是千变万化的,很难有软件能满足全部学科领域的所有需求。即便是同一专业,具体研究也需要软件具有较大的灵活性。但有限元程序的编写是相对复杂的(依赖于人员素质,所采用的开发工具、投入等),即便一个很小的程序也可能花费一个人许多时间,这在当前讲究效率和速度的时代,要求每个学者都开发出一套自己的程序是不太现实的。

需要指出的是,许多研究人员,特别是硕士和博士研究生,几乎很少需要完整地进行上述各个环节的研究,由于时间或者其他因素,大多数只进行其中的一个

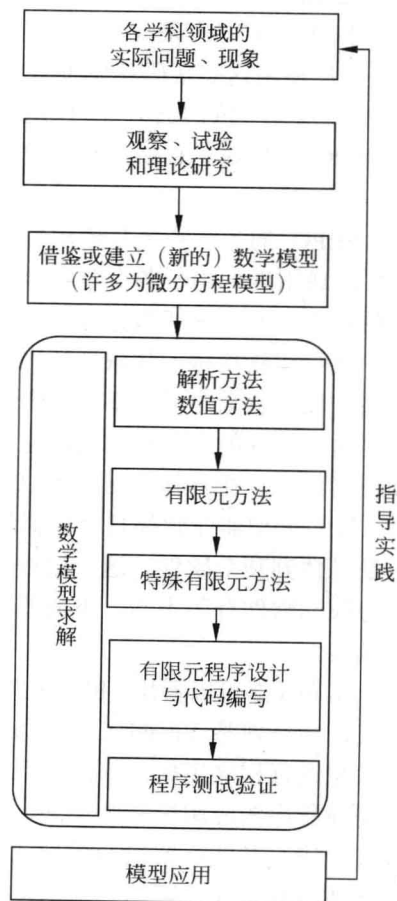


图 1-1 定量创新性科学研究的数学模型范式

或者几个环节，譬如有的专注于做试验，有的专注于理论研究，有的专注于数值计算方法，还有的专注于代码编写。而其中的程序编写环节可能是最耗时费力的，而这可能又不属于大多数人需要创新的地方。早期的学者由于既有的研究积累较少，可能每个环节都需要亲力亲为，但目前来看，即使有个别学者需要完成各个环节，但也不需要每一个环节都要有创新性的内容。这就意味着，在一个或者几个非创新性的环节上可以借鉴前人的研究基础，从而节省自己的研究时间。一些人员比较幸运，有前辈或者师兄师姐的工作可以作为基础，自己只要专注于某一项创新性工作即可，大大加快了自己的研究进程。然而，幸运的人毕竟有限，有大量研究人员尤其是研究生是从“一无所有”开始进行研究的。因此，熟练掌握一套有限元程序代码对提高创新性研究的效率是大有裨益的。

1.4.2 FEPG 的特色及应用模式比较

FEPG 区别于其他有限元软件的第一大特色便是开源，并且这种开源是通过代码自动地生成来实现的，可以随着用户需求自动地动态变化。笔者把这种开源称为“适应性开源”。然而这里所谓“开源”，并不是说 FEPG 这个软件本身是开放源代码的，而是说利用该软件开发的“产品”（有限元程序）就是源代码，并且版权属于开发者。拥有了代码的全部自主权，用户使用这些代码就像使用自己编写的程序一样，拥有极大的灵活性。从这个角度讲，用户可以把 FEPG 当作一个代码开发工具来使用，这一点和微软的 Visual C++ 有些相似，即 FEPG 和 Visual C++ 都是开发软件（或程序）的软件，并且给用户提供了一些高效的辅助开发工具。

FEPG 的另一特色是通用性。它是基于有限元方法的一般数学原理开发的，因此具有本质的通用性。由于各个学科领域的问题通常归结为微分方程数学模型，而 FEPG 恰恰是利用有限元方法求解微分方程的，因此，从理论上讲，任何学科领域的学者都可以使用它。

我们在新接触一款软件特别是大型有限元软件的时候，即使软件做得用户友好程度非常高，仍然感觉难以下手，或者说不知道这个程序有什么样的使用步骤和原则，这实际上就是这个软件的应用模式的问题。懂得了一个软件的使用模式就抓住了该软件的使用思想和使用哲学，在具体应用中可以发挥积极的指导作用，不至于只会照猫画虎而不会灵活运用。许多大型商业软件具有相似的使用模式，即编写输入文件，提交分析，获得分析结果。但是具体到某个软件，这个过程还是有差别的。鉴于 FEPG 的特色，其应用模式必然和其他有限元软件有所不同。下面通过图 1-2 中的对比来体会 FEPG 的这种由其特色决定的应用模式。

从图 1-2 可以看出，FEPG 作为一个软件开发工具，具有和 Visual C++ 极其相似的应用模式，如果熟悉 Visual C++ 开发，则很容易理解这一点。FEPG 可以随时随地对用户的数学模型进行程序生成、修改以实现用户的目的，它并不需要维护一个巨大的模型库。而通用商业程序的应用模式则不同，它恰恰需要维护一个庞大但有限的数学模型库，一旦软件开发完毕，其能够解决的模型就是有限且固定的，因此其应用模式也是单一的，即用户只需要按照其输入要求填写输入数据文件然后提交给程序分析即可。当然，这些商业软件通常还给用户提供了较为友好的使用界面以增加直观性，简化使用难度，例如给出良好的图形用户界面（GUI）等。但是单纯的图形用户界面往往又不能完全实现（描述）程序的功

能, 因此还是要借助于输入数据文件 (如 ABAQUS 中的 inp 文件) 才是最根本的。FEPG 的灵活性恰恰是因为增加了一个代码开发过程, 可以对各个学科出现的数学模型随时进行代码开发, 形成新的计算内核。这相当于 FEPG 不仅可以做改变计算参数和几何模型这类的计算, 还可以改变程序的控制方程, 只不过这种改变是以完全重新自动生成 (编写) 代码的方式来实现的。这种以代码自动生成技术为基础的以不变应万变的思想在灵活性方面是传统 CAE 软件难以企及的。

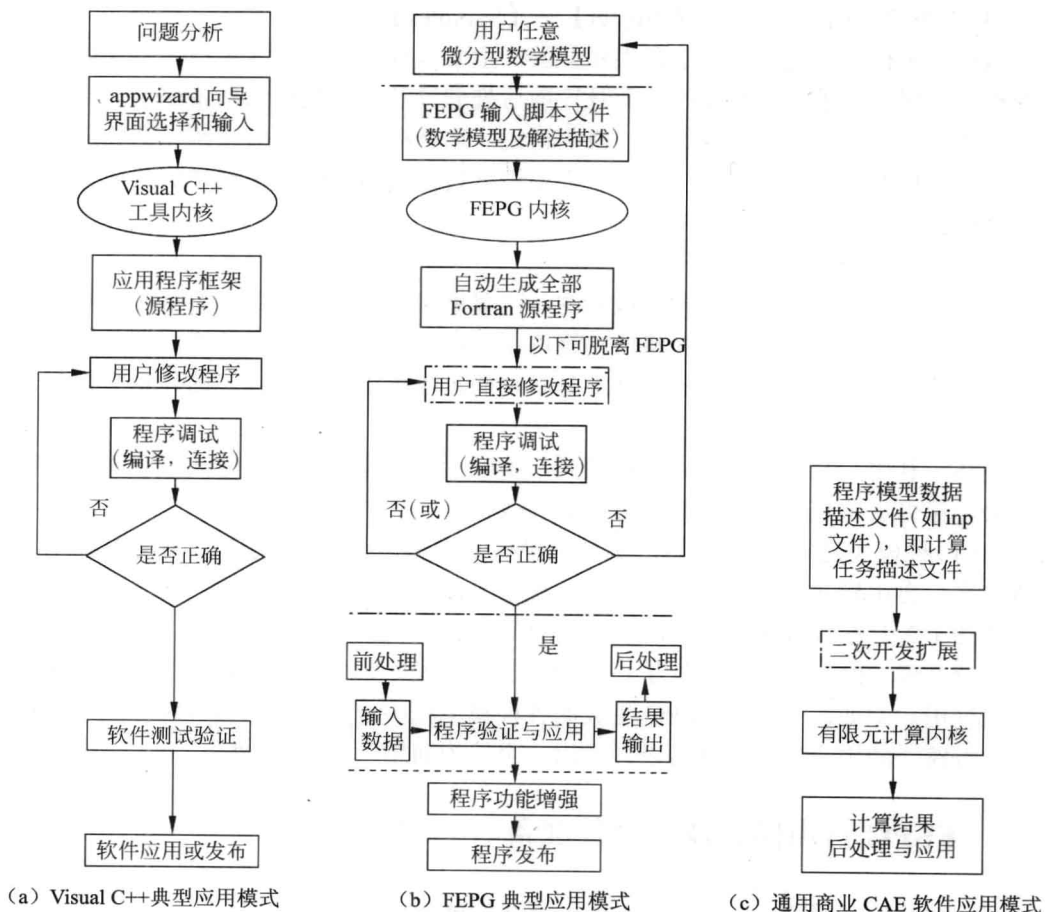


图 1-2 FEPG 与 Visual C++ 及通用商业 CAE 软件的应用模式对比

FEPG 主要有以下几种使用方法。

(1) 用户文件法 (笔者称之为“自由模式”)。

即用户填写 FEPG 所要求的全部脚本文件, 然后通过命令提交给 FEPG 内核进行程序生成。

(2) 公式库法 (或 Appwizard 向导方法, 【Project】→【Appwizard】)。

所谓公式库, 就是针对一些典型的应用问题, 软件已经为用户准备好了各个问题的脚本文件, 对于这些问题, 用户不必亲自去研究和编写脚本文件了, 而是直接使用公式库里的这些文件就可以工作了。而这里的“使用”, 仍然是通过向 FEPG 内核提交这些脚本文件,

然后生成程序。只是这些程序已经过严格的测试，程序生成后用户直接输入数据就可以进行数值计算了，这种方法与商业 CAE 的使用很类似（当然商业 CAE 没有代码生成过程）。另外，用户还会感觉到的一个区别是，在向内核提交这些文件时，使用的操作是不同的，公式库法是直接通过界面操作，而用户文件法是利用 GIO（或 MDI）命令。当然，这不是本质区别，对于 Appwizard 模式，由于其脚本文件都被置于安装文件下，也可以将这些脚本文件拿出来用 GIO（或 MDI）命令进行生成。

（3）“系统算例库”模式（【Project】→【Samples】）。

这种应用模式和公式库方法是很相似的，主要区别是，利用这种模式不仅生成了全部的源程序并编译成了可执行程序，而且连前后处理以及计算数据都准备好了，用户直接执行命令就可以计算并得到计算结果。换句话说，这种模式是 FEPG 系统在公式库方法的基础上又为用户增加了一些便利，用户只需要根据自己的问题修改数据和对几何模型进行前处理就可以了。

（4）“用户算例库模式”。

用户在使用 FEPG 过程中开发成功的一些算例可以保存在 FEPG 系统中，将来如果遇到类似的应用则可以直接使用（将成功算例保存在系统中的好处是从界面提供了一个使用自己历史资源的简单途径，当然这并不是必须的，还可以保存在自己的任意文件下，将来直接拿来使用）。

显然，利用（2）～（4）这几种模式，实际上是在利用系统、他人或者用户自己已有的成果，免去了重复劳动，显然具有很高的效率。

第（1）种模式是软件开发模式，其他几种模式则和传统 CAE 的应用模式类似。从 FEPG 本质来看，自由模式是本质方法，其他几种模式都要归结到这个方法上来。

从图 1-2 中还可以看到，FEPG 的使用模式实际上包容了通用商业 CAE 软件的应用模式。这就是说，FEPG 既可以作为一个软件开发工具来使用，也可以作为一个普通的 CAE 软件来使用。例如，只要对 FEPG 生成的各种模型程序进行适当的整理和打包，然后提供一个友好统一的用户界面，就可以开发出一个较为通用的 CAE 软件。

1.4.3 FEPG 自由模式及其文件准备

在上述的自由模式下，FEPG 的具体使用如图 1-3 所示^[3]。

FEPG 内核通过脚本文件来生成全部有限元源代码程序。但对于用户来说究竟哪些脚本文件是必须的，哪些是根据情况选择的，还是很复杂的。详见表 1-1。

表 1-1 程序生成时用户填写的脚本文件选择

版 本	脚本文件选择			备 注
	必 须	可 选	视情况选择	
FEPG 5.2	gcn, gio, pde(cde,vde), pre		fb, nfe, ges, glt, for	
FEPG 6.0	gcn, gio, pde(fde,cde,vde)	pre	fb, nfe, ges, glt, for	此三行分别为各自独立的选择
	gcn, mdi, pde(fde,cde,vde)	pre	fb, nfe, ges, glt, for	
	mds, mdi, pde(fde,cde,vde)	pre	fb, nfe, ges, glt, for	