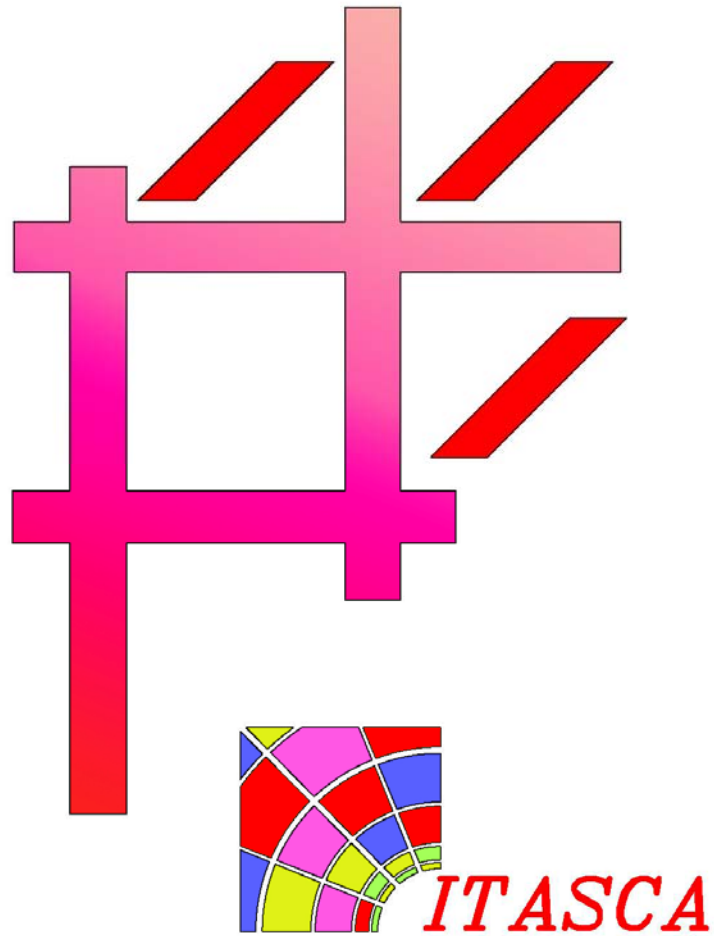


快速入门

(GETTING STARTED)



版本: **flac^{3d} 3.0版 (FTD127)**

翻译: 一米

2009.06

声 明

现在市面上关于FLAC^{3D}软件的教材寥寥无几,在学习的过程中,主要还是参考软件本身的使用手册,虽然读英文版手册有些吃力,但是它论述非常详细,我觉得是用户最好的教材。我在边看手册的时候边做了翻译,目前为止翻译完成了本部分的内容(略去了部分内容和例子),还翻译了命令手册的前半部分内容,等翻译完成了,也会和网友共享,但是像本人这类英语水平一般的人做这样的翻译工作是比较辛苦的,我也不确定是否有毅力完成命令手册下半部分的内容。虽然这样的工作比较艰难,但我觉得还是学到了不少东西,手册是最原始,最翔实的基础教材,看明白了手册,运用软件才会游刃有余。

由于本人专业水平和英语能力的限制,存在问题是在所难免的,有的地方甚至可能曲解了原意。考虑到时间因素,译文的措辞没有细细斟酌,还请网友谅解。如果发现译文中的错误,还请广大读者斧正。

一米

2 快速入门

这一部分将向初次使用 flac3d 的用户介绍软件的基本使用方法。主要有以下内容：软件的安装与启动；用软件分析解决问题的步骤，在每一步的操作中，都有简单例题来说明该步骤具体是如何操作的。

如果你对软件比较熟悉，但是现在很少用它来处理问题，那么这部分的内容（尤其 **2.7 节**）能很好的帮你回顾软件操作的要点。本部分 **3.3 节** 全面详细的介绍了如何进行问题的求解。

Flac3d 支持命令驱动和图形菜单驱动两种模式*。在本手册中大部分的算例都采用了命令驱动模式。我们认为这种模式能给用户提供操作软件最清晰的思路。在 **1.1 节** 中我们就已经提到了命令驱动模式使得 flac3d 在分析求解工程问题时成为了一个功能强大的“多面手”。然而这种模式让新用户，或者长时间未接触软件的老用户用起来有点不那么容易。命令行必须用键盘输入，可以直接输入到软件的命令窗口，或者先保存为数据文件，再通过软件的相关命令进行读取。Flac3d 能识别超过 40 个主命令和 400 多个附属的关键词。

本部分主要包括以下内容：

- 1 在 **2.1 节**，手把手的教你们如何在自己的电脑上安装和启动 flac3d 软件。
- 2 在 **2.2 节**，用一些简单的教学案例帮组用户熟悉一些常用的命令。
- 3 在用户建立自己的模型并进行分析计算之前，有必要先了解 flac3d 的一些基本知识。在 **2.3 节** 讲述了 flac3d 的基本术语；在 **2.4 节** 主要说明了有限差分网格的定义规则；而在 **2.5 节** 阐述了输入命令的基本句法。
- 4 在 **2.6 节**，阐述了 flac3d 的特点，比如创建、命名和使用对象，以方便用户进行问题的求解
- 5 在 **2.7 节**，一步步的指导用户如何建模和分析问题，每一个步骤都分开论述，并提供简单的例子帮助用户理解。
- 6 **2.8 节 - 2.10 节** 分别论述了系统的符号约定、单位体系和精度限制
- 7 **2.11 节** 说明了软件中各种类型文件的创建和使用。
- 8 **2.12 节** 对图形菜单操作模式进行了简介。

*：对于初级用户来说一般图形菜单驱动模式只进行图形输出或者文件操作。本章节的最后一部分将向用户展示如何使用图形菜单驱动模式来操作软件。

2.1 安装启动程序

2.1.1 系统要求

安装运行 flac3d 需要的系统最低配置如下：

处理器：时钟频率至少为 1GHZ，处理器的主频越高，那么 flac3d 的计算速度将越快。

硬盘：安装软件至少需要 12MB 的硬盘空间。如果装载了在线的用户手册，那么还需要 16MB 的空间。（注意默认情况下，安装软件时会自动装载用户手册）。除此之外，还需要至少 100MB 的硬盘空间来存储分析计算时生成的各种文件。

内存 - 启动软件至少需要 3MB 的内存。在建模过程中，软件所占用的内存，会不断的发生变化（见表 2.1）WINDOW 操作系统还限定了软件建模时占用的内存不能超过 2GB。

显示器：推荐 1024 × 768 分辨率，16 位彩色显示器。

操作系统：FLAC3D 是 32 位操作系统的应用程序，所以基于 intel 技术的 WINDOWS 98 及以上操作系统均支持软件的安装和使用。

输出设备：默认情况下，系统图形会输出到系统打印机上。也可以复制到剪贴板上，或者保存为格式化的文件，这里所说的格式包括：加强型图元文件格式和位图文件格式（PCX/BMP/JPEG）。用户可以使用 **set plot** 命令来指定输出的形式及格式。

2.1.2 软件的安装

(略)

2.1.3 组件

软件的可执行文件为“F3300.EXE”。FLAC3D 是使用 VC++ 7.0 编写的。除了可执行程序外，还需要两套动态链接库（DLL 文件），一套用来接入和存取各种各样的图形；另一套提供内置的各种本构模型。

2.1.4 应用程序和图形处理设备

在使用 FLAC3D 时，各种应用软件和图形处理设备会起到很大的辅助作用。

编辑器：任何以 ASCII 码为标准格式的文本编辑器都可以用来创建 FLAC3D 的数据文件。但是必须要注意一些“先进”的文档编辑器（如 WordPerfect, Word 等软件），这些编辑器会把格式说明信息编译成标准输出格式，这些说明信息并

不能被 FLAC3D 识别，所以导入这类文档时会出现错误。FLAC3D 输入的数据文件必须是标准 ASCII 码形式的文件。

图形输出设备：FLAC3D 支持很多种类型的图形处理设备，默认情况下，生成的图形可以用“**Plot hardcopy**”命令来连接到系统默认的打印机以便输出。（或者通过 FLAC3D 主窗口中 **FILE** 菜单栏下的 **print-view** 来设定）

“**Plot clipboard**”命令可以将显示的图形，存放到 WINDOWS 剪贴板上（没有任何文件生成）。该图形接着就可以以加强型图元文件格式被粘贴到其它兼容该格式的 WINDOWS 应用程序中去。

“**Set plot metafile**”命令可以将图形以加强型图元格式存盘，以便作为计算的参考或日后插入到文档中去。

通过命令：**Set plot** + 关键词（**pcx, bitmap, bmp** 或者 **jpg**）可以存储为许多图像格式（**pcx, bmp, jpeg** 等）。输出的这些位图的分辨率由命名行：**Set plot <output type> size** 来控制。当然也可以使用 **Set plot avi** 或者 **Set plot dcx** 以及 **Set plot movie** 命令将显示图形输出为视屏格式。

无论是黑白的还是彩色的 **postscript** 打印机，都需要通过“**Set plot postscript**”命令来指定。打印图形将存储为文件，这样支持 **postscript** 格式的图形处理程序就可以读入并进行修改了。

2.1.5 启动软件

双击可执行文件“F3300.EXE”便启动了程序，接着会弹出一个 FLAC3D 的主窗口。在主窗口的最下面附带了一个命令窗口，我们可以把命令直接输入到命令窗口中来执行相关命令，命令窗口最初显示的提示符为：“FLAC3D>”。

当软件启动后，它占用的系统内存是随着用户的操作而不断变化的（比如说，在建模过程中，系统所占用的内存会越来越多）。我们可以在命令窗口中输入 **print memory system** 命令来查看现阶段程序已占用的内存及操作系统还可为软件提供的总内存。如果你在操作过程中发现命令失效（并不是命令错误），那么一定是系统可分配的内存太少了，软件所占用的内存过多。这个时候，最好退出并重启软件，以释放内存。表 2.1 列出了一般建立摩尔库伦材料模型的单元数与软件占用的内存之间的大致对应关系。

表 2.1 FLAC3D 内存使用情况表

单元数	占用内存量 (MB)
20 ³	20
30 ³	68
40 ³	161
50 ³	314
60 ³	541

2.1.6 版本号说明

(略)

2.1.7 程序的初始化

刚打开FLAC3D软件，它首先会在当前文件夹下寻找“FLAC3D.INI”文件，如果没有找到，它就会到安装目录下寻找。它的作用是存放用户设定的程序初始化模式的命令。以便每次打开软件都载入用户的初始设置。

如果“FLAC3D.INI”文件不存在，软件继续运行而不会提示出错信息，注意一点：一些存储在“FLAC3D.INI”里的命令，如果并不是设置初始化的命令，有可能导致错误的信息。

2.1.8 运行FLAC3D

Flac3d命令驱动模式包括两种方式：交互模式（在命令窗口中输入命令行）；命令流模式（将命令行保存在数据文件中，通过读入该文件执行相关命令）。如果输入的命令存在错误，那么窗口中将会出现错误提示。命令流文件一般通过文本编辑器创建和修改（见2.14节），虽然命令流文件可以定义为任何文件名，但是最好设定其扩展名为“.dat”，以防止和flac3d其它类型的文件相混淆。

要读入命令流文件可以使用以下命令：**call file.dat**

其中，file.dat 指的是用户定义的命令流文件的文件名。一旦读入文件，你会发现软件会将当前在文件中读入的命令行，显示在屏幕上。如果命令流文件保存在当前文件夹下^{*}，那么在**call**命令后面只需输入完整的文件名即可，否则还应

^{*}笔者注：所谓的当前文件夹包括两种情况：1、没有读入任何数据时当前文件夹指的是软件应用程序所在的文件夹。2、如果已读入了数据，比如导入了模型信息文件（“.flac3d”文件），这时当前文件夹指的就是用户之前读入文件所在的文件夹。

在文件名前面加上文件的完整路径（比如：c:\我的文件夹\file.dat）。除这种方法外，我们也可以菜单操作读入文件：依次点**file-call**按钮（见2.12节）

为方便起见，我们可以为应用程序创建快捷键，右键点击“F3300.exe”不放，并拖动到相应的创建快捷键的位置，松手后会弹出一个对话框，选中“**在当前位置创建快捷方式**”，这样就生成了一个快捷方式。双击该快捷方式就可以启动软件。创建快捷方式的目的并不只在于方便打开应用程序，我们右键新创建的快捷方式，选择“**属性**”，接着在弹出的对话框中将“**起始位置**”这个文本框中内容删除并点击左下角的确定按钮。这样当你双击该快捷方式启动应用程序时，系统默认的“当前文件夹”就是快捷方式所在的文件夹了。我们可以将快捷方式和输入文件放在同一目录下，这样就方便了文件的输入。

2.1.9 装载测试文件

(略)

2.2 一个简单的计算教程——常用命令的使用

这一部分主要是为那些刚接触FLAC3D，跃跃欲试的新用户准备的。在这一部分，将通过一个简单的例子来帮助用户学习一些求解问题的基本知识。

例题的主要问题描述如下：在一块土体中一次性开挖一个 $2\text{m} \times 4\text{m} \times 4\text{m}$ 的沟渠，并对沟渠周围土体的变形作监测和分析。为了给用户提供方便，在安装目录中“\Tutorial\Beginner”文件夹下的“TUT.DAT”数据文件里包含了本例题使用的所有命令。

我们采用交互式的方法在FLAC3D中运行该算例。（即：用键盘输入命令行，在命令行输入完成后按回车键，然后直接查看输入命令后的结果）

首先让我们打开FLAC3D软件，打开后，便可以看到FLAC3D的主窗口，在主窗口下方，是命名窗口，命令窗口的初始提示符为“FLAC3D>”。

建立初始的有限差分网格的命令为**GENERATE***:

GEN ZONE BRICK SIZE 6 8 8

这个命令会在软件中建立一个X方向有6格，Y方向有8格，Z方向有8格的三维长方体网格。在我们建立的这个模型中，Z方向为竖直方向。

我们可以用**PLOT**命令进入绘图模式，在该模式中可以观察所建立的网格。一旦输入**PLOT**命令，那么接下来所要输入的命令都必须为**PLOT**子命令，绘图窗口默认定义为一个名为“BASE/0”的窗口，我们可以看到这时提示符变成了“Plot Base/0>”。

为了显示彩色的网格体和模型的坐标系，就需要输入如下的命令行:

例2.2 创建和使用新视窗

create Trench（创建了一个叫做“TRENCH”的视图窗口）

add surface yellow（表面显示黄色）

add axes black（显示黑色的坐标轴）

show（在屏幕上显示）

新创建的视窗自动的设置为当前活动视窗。

按键盘上的x、y、z键（小写状态）可使得当前视图绕着X Y Z轴旋转（大

*: 该命令的详细信息请参见命令手册部分的第一章节。注意命令可以缩写，变量与其对应的参数值之间可以试用“=”符号来分隔（见2.5节）。

写状态时旋转方向相反)使用**m(M)**键则可以对视图进行放大(缩小),方向键则可以控制视图进行平移。当然反方向旋转也可以使用**shift+**控制键实现(其原理就是大小写之分)对视图的其它一些操作和控制,可以参考命令手册部分的表1.2。

注意:我们也可以通过**PLOT**命令直接指定放大的倍数和视角。每一个视图窗口的左边都会有该视图的详细信息:**CENTER**(中心);**ROTATION**(旋转角度);**DIST**(距离);**MAG**(放大倍数)等。我们现在通过交互模式建立我们想要的视图,而这些视图的数据信息其实还可以转变成数据文件,这样在其它视图中想要得到类似的视图效果,就可以通过命令输入相同的数据。(实现过程见例题2.3)我们现在在例题2.2的基础上再建立一个视图窗口通过输入以下命令来创建一个新视图。

例2.3 指定视窗显示数据

```
create Trench2  
add surface yellow  
add axes black  
set rotation x y z  
set center x y z  
set dist d  
set magnification m  
show
```

新例题2.3中多输入了加了黑体字的部分(红色部分的数字等同于例题2.2中经用户调整后显示在屏幕左侧中对应的数据),该部分等同于用键盘控制视图,2.3和2.2的两个视图应该是一模一样的。

还是来看例子2.2:在Trench视窗中我们建立了一个 $6 \times 8 \times 8$ (m)坐标原点在(0, 0, 0)的网格。网格的尺寸,原点,以及网格的密度都是可以通过**GEN+关键词**的命令来更改的。这个问题将在稍后再作介绍。

现在我们要给整个模型空间定义本构模型和具体材料参数。在本例中我们定义为摩尔-库伦弹塑性本构模型。反回到“Flac3D>”命令模式(如在PLOT模式中,按回车键即可返回)并输入:

```
model mohr
```

这个命令就定义了该网格体的本构模型为摩尔 - 库伦模型。网格中的每块区域都可能需要定义成不同的本构模型和赋予不同的材料参数。但是如果 **MODEL** 命令后没有特别指定命令作用的区域，则系统默认为作用的范围为整个模型。

为了提高塑性解答的精度，每一个单元都会自动的被细分为两组四面体的子网格（参见**理论与背景**部分的第一章）。

想知道计算这个问题需要占用多少的内存，可以输入以下命令：

```
print mem sav mem sys
```

输入命令后软件便会提供一张在现阶段内存的统计列表。我们这个例子在现阶段大概需要占用0.6MB的内存来存储数据，大约需要4.2MB的内存来进行模型的计算。

接着在命令窗口中输入材料属性赋值命令：

```
prop bulk = 1e8 shear = 0.3e8 fric = 35
```

```
prop coh = 1e10 tens = 1e10
```

以上的命令定义了材料的体积模量（PA）、剪切模量、内摩擦角、粘聚力和抗拉强度。用户会发现粘聚力和抗拉强度值取得很大，这样取值的目的是为了模拟得到模型在自重作用下的初始应力状态^{*}。取值很大是防止了模型在初始加载分析时就达到塑性极限。一旦你拥有了很多的模拟经验，你就会知道这样做的原因是很明显的。

在该样例中，土体仅受自重，在命令行中输入如下命令来加载重力：

```
set grav 0, 0, -9.81（重力加速度,单位：m/s2）
```

```
ini dens = 1000（密度，单位：Kg/m3）
```

命令行中：**SET**命令设置了重力加速度，**INI**定义了初始网格的质量密度（只有考虑重力的情况下才需要定义密度，在静力问题中，如果我们忽略重力，那么就无需定义密度，这点与FLAC不同）。接着我们用如下命令来定义模型的边界条件：

```
fix x range x -0.1 0.1
```

^{*}：有许多方法都可以模拟达到模型的初始平衡状态，比如可以先给模型赋予弹性本构模型来计算初始应力状态，再重新定义为摩尔 - 库伦模型来分析计算模型的开挖、加载及各种条件的改变过程；还可以直接赋予摩尔 - 库伦模型，然后采用 **solve elastic** 命令来分阶段求解。

```
fix x range x 5.9 6.1
```

```
fix y range y -0.1 0.1
```

```
fix y range y 7.9 8.1
```

```
fix z range z -0.1 0.1
```

以上的命令固定了模型的五个面（连杆支承），边界一旦被“**FIX**”（固定）后，在被固定的方向就不会发生位移和产生速度。

FIX命令在例题中起到了如下作用：

- 1、固定了边界面 $X=0, X=6$ 上的所有节点 X 方向的位移，因为这两个边界面分别在`range`关键词所指的范围内；（命令的前两行）
- 2、固定了边界面 $Y=0, Y=8$ 上的所有节点 Y 方向的位移；（第三、第四行的命令）
- 3、命令的最后一行固定了底部边界面（ $Z=0$ ） Z 方向的位移。

如果我们想在迭代计算过程中记录一些有用的变量或参数的变化，从而来判断分析是否已经达到平衡状态或者模型已经发生垮塌，就需要使用到**HISTORY**（**HIST**）命令。

在本算例中，我们接着在命令窗口输入如下命令：

```
hist n= 5
```

```
hist unbal
```

```
hist gp zdisp 4,4,8
```

命令的第一行：指定了参数记录的步频： $N(Nstep)=5$ 指定了每迭代计算5次记录一次相关的值（如果不指定，则系统默认值为10）。命令的第二行指定记录最大不平衡力（the maximum unbalanced force）命令的第三行指定记录坐标值为（4，4，8）的节点 Z 方向的位移。在计算时记录最大不平衡力是一个很好的习惯，如果最大不平衡力接近一个很小的值，并且位移记录值不再发生变化，那就表明计算已经到达了平衡状态。

现在已经准备好了求解模型初始状态的必要条件。由于FLAC3D计算的结果为显式的动态解，我们通过设定计算所需的时间步来控制计算的进程，模型的动能会慢慢衰减为零，这样就得到了我们所要的静态解。为了使单元体获得重力场，我们需要设定计算的时间步来求解模型在自重的作用下初始平衡状态。软件中

SOLVE命令就是按一定精度自动求解平衡状态的命令。

了解了这些知识，我们接着回到例题中来，在命令窗口中输入如下命令：

```
set mech force=50
```

```
solve
```

这样，软件便开始了问题的求解，在求解过程中，会在程序窗口中显示计算的时间步数以及对应的最大不平衡力的值，当最大不平衡力小于所设定的上限值（本例题中我们用**SET**命令设定了最大不平衡力的上限值为50N，即：计算的最大不平衡力一旦小于此值，我们便人为模型达到了平衡状态。）求解过程终止*。因为我们之前没有关闭绘图窗口，你会发现显示在窗口中图像不断的被更新。如果我们先关闭了绘图窗口，那么求解所需的时间会更短。

在我们的模型中，计算停止在351步**，对于一般主频为1GHZ的奔腾电脑，完成这样的计算仅需要短短的几秒钟时间。

我们可以通过查看最大不平衡力以及所记录节点Z方向的位移采样图来分析，模型是否达到了平衡状态。

在命令窗口中输入：

```
plot
```

```
hist 1
```

输入完命令后便可以在窗口中显示**HIST 1**(最大不平衡力)的采样记录图。按回车键，再输入如下命令：

```
hist 2
```

可以查看所记录节点Z方向的位移采样图了。

最大不平衡力采样图（图2.3）中显示最大不平衡力接近于零，位移图（图2.4）中可以看出位移已经趋于一个固定值。这两个采样结果均表明系统已经达到了平衡状态。

上面两个采样图都是在FLAC3D默认视图窗口（BASE/0）中显示的。任何图形如果没有定义新的视图窗口，都会直接在默认视图中加以显示，这并不会对已建立的自定义视图窗口产生影响。默认视图窗口就像一个“抓拍”器，能立即

*：计算精度的有关内容将在2.7.4节中详细讨论。

**：由于用户处理器的不同，实际计算步数和给出的计算步数会有微小差别。

让你观察到你所需要的视图。

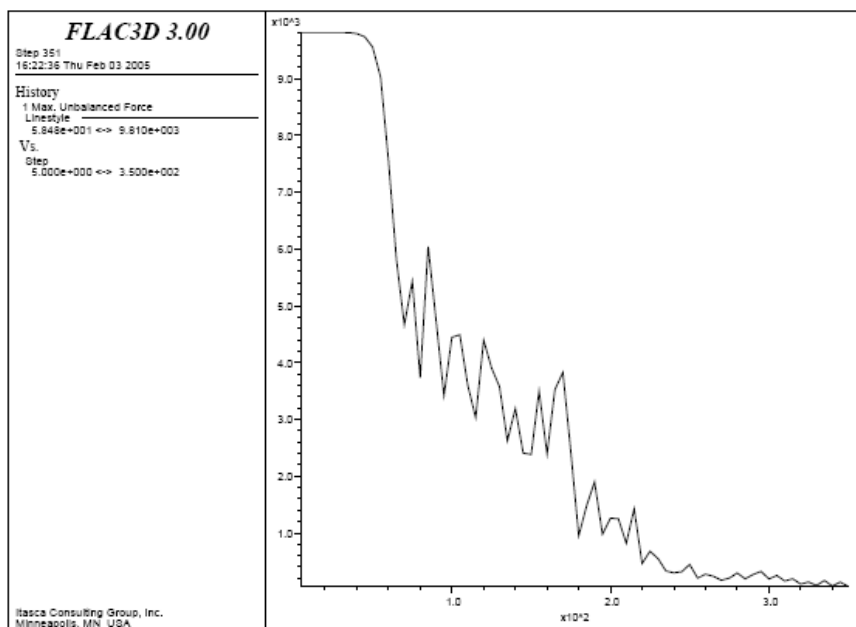


图2.3 最大不平衡力记录图

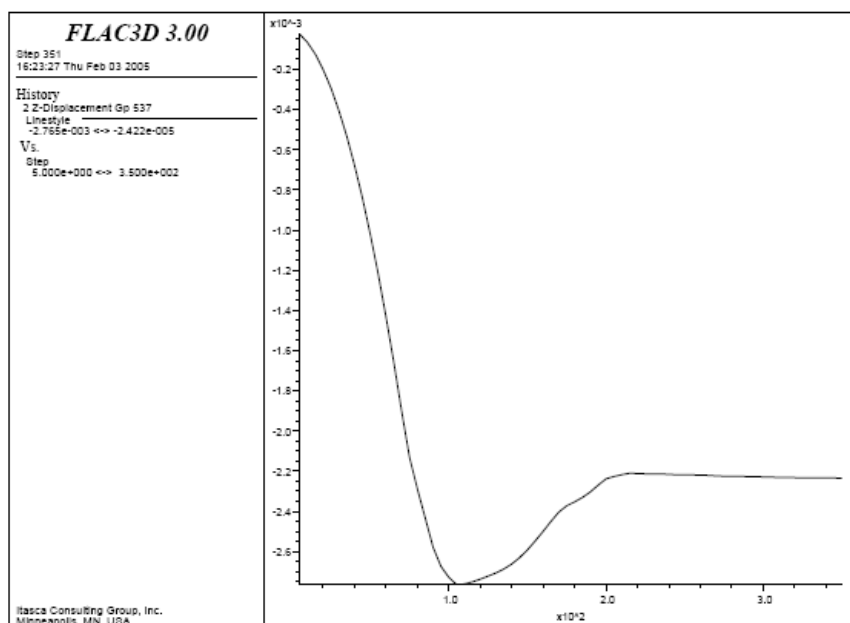


图2.4 节点 (4, 4, 8) z方向位移记录图

注意：输入的采样记录，系统按输入顺序从“1”开始一直往下编号，这就是为什么输入`hist 1`命令显示的是最大不平衡力的采样记录图，输入`hist 2`命令可显示Z方向的位图采样图的原因了。当采样参数较多，自己又不记得输入的顺序时，可以在“FLAC3D>”提示符状态下输入：`print hist`命令来显示所有采样参数对应的编号（称为id号）。

接着我们通过下面几行PLOT命令来建立一个比较复杂的新视图，以检查重力场是否已经施加到模型中：

```
plot create GravV
plot set plane dip=90 dd=0 origin=3,4,0
plot set rot 15 0 20
plot set center 2.5 4.2 4.0
plot add bound behind
plot add bcont szz plane
plot add axes
plot show
```

注意以上命令均在“FLAC3D>”提示符状态下输入，如果先输入`plot`命令，进入“plot”相关的提示符状态，那么命令中所有的`plot`均可删除。执行以上命令后，便会出现一个名叫GravV的视窗，并且系统指定它为当前显示窗口。我们在命令中已经定义了当前视图的剖面：倾角为90度（`dip=90`），倾向为0度（`dd=0`），通过点（3, 4, 0）。在命令中我们还定了视角（`plot set rot 15 0 20`；`plot set center 2.5 4.2 4.0`）并且使用`bound`（boundary）命令来增加边界线框，线框只在平面的后面显示（由关键字`behind`定义，注可简写为`BE`）。`plot add bcont szz plane`命令的作用是在当前平面中绘出垂直应力分布图，图形如图2.5所示。

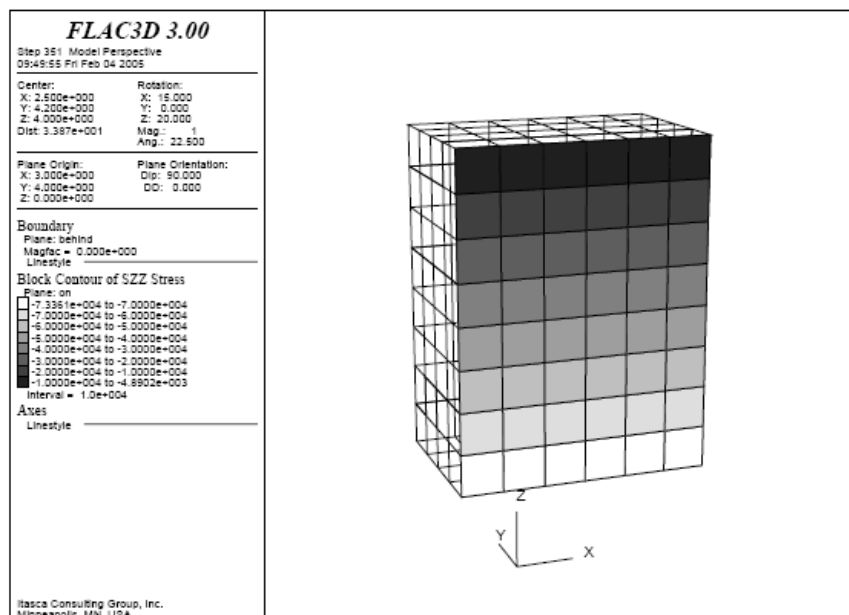


图2.5 初始平衡状态z向应力剖面图

现在，最好保存以上得出的初始状态，便于在将来任何时候返回到该状态，重新设定相关参数来做相应的研究。按回车键返回到“FLAC3D>”提示符状态，输入下面的命令：

save trench.sav

这样一个名叫trench的SAV类型文件就在当前文件夹下生成了。

到此，我们在模型中已经新建了两个视图窗口。我们可以用**plot print**命令来列出所有的视图名称，当前视图前会有“==>”这样的标志以示区别。也可以用**plot print view**命令来显示当前视窗的详细信息。在以上命令后加上视窗名或者其对应的id号，就可以显示相应视窗的详细信息，包括：所有的视图设置；记录了哪些参数。

如果想从GravV视图返回到早先创建的Trench视窗，那么输入如下命令：**plot current Trench**。实现以上目的还有另一种方法，输入：**plot show Trench**。只要你知道你想设置为当前视图的视图名，你就可以通过以上两种方法实现。当然也可以采用菜单操作：当鼠标输入光标指示在命令窗口时，在主窗口中有一个PLOT菜单，在该菜单中罗列了所有用户定义的视图，鼠标点到你想看到的视图，点选SHOW子选项，你就把该视图设置成了当前视图。

我们再进行下一步工作，在土体中开挖沟渠，首先输入命令：

prop coh=1e3 tens=1e3

这个命令重新定义了整个模型材料的粘聚力和抗拉强度为1000Pa, 这样的取值其实已经能够保证在初始状态中不出现错误了（未开挖前），但是我们应该总是要通过一些计算步数的计算来检查在初始状态可能出现的失稳垮塌。为了模拟开挖，我们只需把要开挖部分的网格体的本构模型设置成空模型（null）即可，命令如下：

model null range x=2,4 y=2,6 z=5,10

开挖部分（即模型材料为NULL）是通过限定x, y, z方向区域的办法来选取的。

由于粘聚力比较小，而且沟渠壁没有支护，因此开挖后应该会发生土体垮塌的现象。我们想要模拟的是现实过程，那么通过上面的简单分析，设定计算为大变形是合情合理的，设定的命令如下：

set large

我们需要看到的仅是开挖这个过程引起的位移变化。而不是从加载重力到开挖整个过程的位移变化，所以，系统中所有网格节点的位移应该全部清零*命令如下：

ini xdis=0 ydis=0 zdis=0

我们故意取了较小的粘聚力值，以保证模型发生破坏。这样一来，我们就不能用SOVE + 最大不平衡力上限值（判断达到平衡的条件）这样的命令方式来求解问题了，因为模拟的计算将永远不收敛而不能达到平衡状态。我们可以通过限定每次计算的时间步数来控制模拟过程，从而了解在坍塌发生时的有关情况，这里才真正体现了显式求解方法的精髓。计算过程永远不会慢慢趋于平衡，这也是有别于一般工程师所熟悉的隐式求解线性方程组的过程。

在FLAC3D中要求得这类解，就要用到**STEP**命令，在本例题中输入：**step 2000**，这样FLAC3D就会执行2000时间步的计算。（对于主频为1GHZ的奔腾电脑，这个过程都花不了一分钟的时间）

然后我们要通过查看相关图形，来检查到现阶段为止的计算结果。现在我们先来查看位移的等值线图，输入如下命令：

*：这个步骤不会影响到模型的计算，只是方便于用户查看需要的结果，因为在迭代计算过程中，没有用到位移。

plot

create DispCont

copy GravV DispCont settings

add cont disp plane behind shade on

add axes

show

在命令中我们拷贝了之前定义的GravV视图的设置为当前定义视图的设置（具体来说就是平面的定位设置）。在出现的位移等值线图中（图2.6）你会发现，在开挖部分的周围一些网格开始发生变形。在图中你还可以看出因开挖，地面发生沉降的区域*。

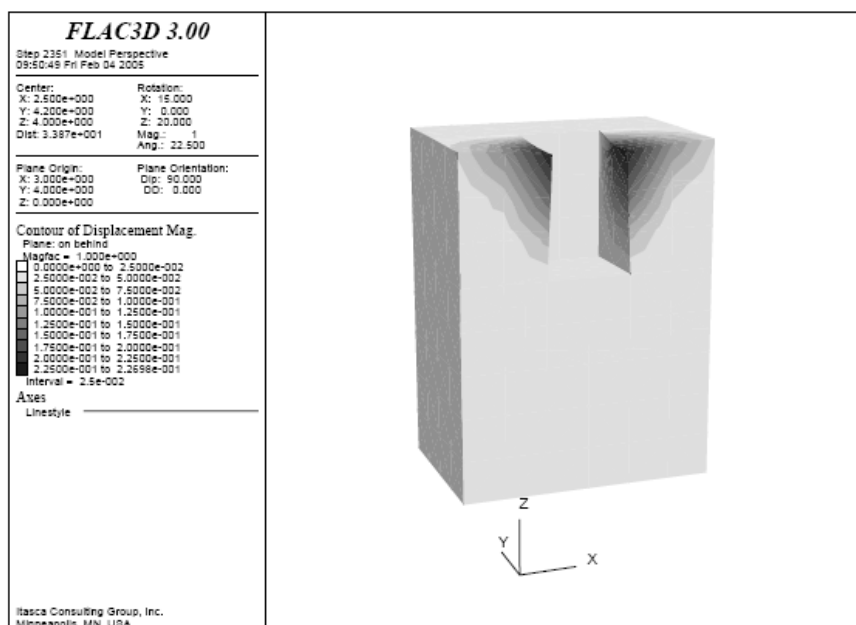


图2.6 开挖计算2000步后的位移等值线图

本例题暂时就到此结束，在接下来的内容中我们将介绍FLAC3D的其它一些知识，来解决本例题模型一些更复杂的问题，包括怎样对开挖的沟渠进行加固。我们建议初学者采用边看教学手册边操作的模式来学习接下来的知识。如果你对FLAC3D的命令已经比较熟悉了，那可以转到3.3章节来了解FLAC3D在求解问题时还需了解的其它细节信息。

*：考虑到加快例题的计算时间，我们取的开挖土体边界很小，在实际情况中，开挖可能是在很大的一块地面上进行的。那么自然的，我们认为定义的边界会对计算结果产生影响。

2.3 基本术语

FLAC3D使用的术语与一般的有限差分程序或者有限元程序是一致的，为了使用户能够清晰的了解这些术语，现图文并茂加以说明，图2.7中标示了flac3d中的基本术语。

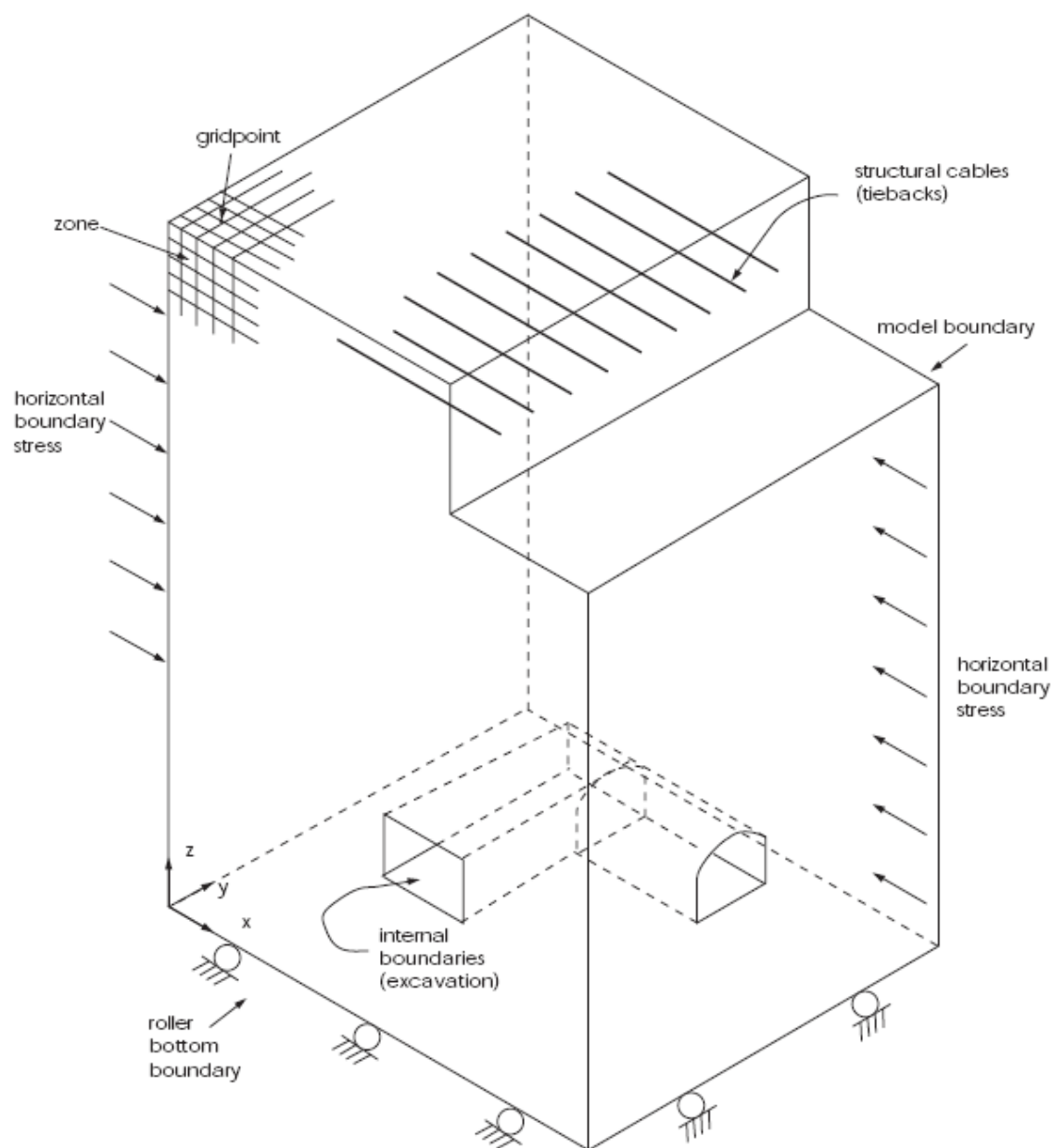


图2.7 基本术语说明图

FLAC3D MODEL——用户在FLAC3D软件里建立的一个物理模型，用户通过输入命令的方式来定义模型所处的各种状态，已完成对真实问题的数值模拟过程。

ZONE——单元体，也叫ELEMENT,是分析现象（如：应力应变关系）时发生变化的最小空间区域，空间单元可以有不同形状（如：六面体、楔体、锥体、四

面体等)，模型的网格是由这些最基本形状的单元体组成的。每一个单元都自动再细分为两套Sub-zone^{*}，这个和用户建模无关，一般不需要去深入了解。

GRIDPOINT——节点（单元体的角点），也叫node(或 nodal point)。任一个单元体都是由它上面的每个节点的三维坐标值而精确定位的。

FINITE DIFFERENCE GRID——有限差分网格，又叫MESH，是由各种形状的单元体组合而成的分析问题的区域，它也可以确定模型中变量的存储位置，所有的向量（力、速度、位移等）都存储在节点位置，所有标量和张量（应力、材料参数）都存储在单元体的中心位置。

MODEL BOUNDARY——即网格体外部边界和内部边界（网格体内的孔洞）

BOUNDARY CONDITION——边界条件，模型边界的约束或者控制条件（如位移的约束，在边界上施加的力，在渗流模型中的不透水边界，在热交换问题中的绝热边界等）

INITIAL CONDITIONS——初始条件，在加载和扰动前所有变量的初始状态。

CONSTITUTIVE MODEL——本构模型，也叫MATERIAL MODEL，用理论公式表示模型材料的变形和受力特性。软件中附带有许多本构模型以用来模拟各种岩土材料的各种问题，每个单元都可以单独的定义本构模型和相关模型的材料参数。

NULL ZONE——空单元，用来定义计算时不需赋材料的网格区域，通常用来模拟开挖或者暂时还未填筑的材料。

SUB-GRID——子网格，网格可以由子网格组成，子网格通常用来创建那些模型中不同形状的子区域（比如：在建立大坝模型时，可独立建立坝体和地基的子网格），子网格是各自生成后再通过移动和绑定操作来组装成整个模型的。

ATTACHED FACES——绑定面，是指子网格被连接和绑定在一起的网格面，两相互连接的子网格绑定面一定要共面和接触，但是不要求节点相互匹配，不同单元体划分密度的两个子网格也可以相互绑定在一起。

INTERFACE——接触面，是两个子网格间的一种连接面，通过这样的连接面连

^{*}笔者注：网上关于它的解析，subzone和flac3d的Mixed Discretization算法有关，比如说一个brick，计算时又自动划分为2套overlay，每套overlay中各有5个tetra（四面体）即subzone，据说这样可以更好地模拟材料的塑性变形。

接的子网格在计算中可以被分开（如产生相互滑动、开裂）接触面通常用来模拟物质的不连续性，比如断层、实际接触面、两种不同材料间的分界面。

RANGE——在FLAC3D中它用来定义所指向的区域，它限定了命令作用的区域。

GROUP——组，组是FLAC3D中用来定义单元体集合的术语，组只能唯一命名，不同的组要采用不同的组名，组用来限定命令有效性的空间域，命令后面加上了组名，就表示，该命令只对该组内的成员有效。

ID NUMBER——编号（id号），模型中的各个元素都有一个数字编号，同类元素之间就是以这个编号加以区分和识别的。下面这些元素都具有ID标识号：接触面（interface）、节点（gp）、单元（zone）、体(volume)、参考点（reference points）、历史记录项（history）、表（table）、绘图项（plot items）以及结构单元（包括：梁、索、桩、壳、衬砌、土工栅格等）可以用**PRINT**命令来得到元素的ID号。用户可以自行定义当前输入的接触面、参考点、历史记录项、表、绘图项的id号。每个结构单元中还有CID编号，CID号用来标识该结构体中各个元素，结构体中的节点，单元，以及之间联系部件等都有各自的CID号。

STRUCTURAL ELEMENT——结构单元，FLAC3D中有两种类型的结构单元，两节点的直线段用来模拟梁、索、桩，三节点的三角形用来模拟壳、衬砌、土工栅格。结构单元通常用来模拟土或岩石与支承结构的相互作用。这些单元是典型的非线性材料。每一种结构单元，都由以下三部分组成：节点、结构类型（SEs）、节点间的联系结构。

STEP——计算时间步，也叫time step或者cycle因为FLAC3D为显式码解法，求解一个问题时需要给定计算求解步数。一般简单典型的问题求解只需要2000到4000步，然而对于复杂的大型模型就可能需要上万步的求解才能达到稳定状态。如果是做动力分析，则必须在**STEP**命令后加上确定的求解步数。

STATIC SOLUTION——静态解，如果求解的相关变量的增量随着时间步的变化只发生微小的变化，而这个变化对于变量本身来说可以忽略不计，那么我们就人为模型已经达到了相对平衡状态，计算的结果便是所求的静态解。在静态解模式中，要么模型达到了平衡状态，要么模型外荷载作用部分区域或者全部区域下达到了相对稳定的塑性流动状态（在发生滑坡等失稳状态下）。静态解模式是

FLAC3D的默认求解模式*。静态解可以和地下不稳定渗流、热交换问题进行耦合。

UNBALANCED FORCE——不平衡力，也称为unbalanced或者out-of-balance force。是数值计算迭代过程中产生的系统内外力之差。在静态求解模式中用它来判断是否达到平衡状态（或者模型刚开始发生塑性变形的状态）如果模型网格的每个节点力都为0，那么模型应该达到了绝对的平衡状态。在FLAC3D中用STEP或SOLVE命令执行计算的过程中，系统会自动记录最大的节点力，并在屏幕上显示。最大的节点力也叫最大不平衡力，在数值分析中，最大不平衡力不可能为零，但是只要最大不平衡力与作用在体系上的外力相比小到可以忽略不计，我们便认为体系达到了平衡状态。如果在计算中，发现最大不平衡力趋于一个非零值，那么就可以肯定模型中发生了坍塌或者塑性变形。

DYNAMIC SOLUTION——动态解模式，用来求解动力学问题，一般适用于外力为高频率震荡且持续时间较短的情况（比如：地震或者爆破）。

LARGE STRAIN/SMALL STRAIN——大变形/小变形，在软件中，系统默认为小变形模式，即：系统中的节点坐标始终不发生变化，即使计算所得的应变比较大（与一般的单元尺寸相比）。在大变形模式中，可能发生几何非线性的变化。

2.4 有限差分网格

有限差分网格覆盖了整个所要分析的物理区域，最小的网格可以只包含一个单元，但是大部分模型的网格都是由成千上万个单元组成的。

有限元网格由全局的坐标系定义，所有的节点和单元体的重心都由（x,y,z）坐标值定义。节点和单元也可通过ID号加以识别。ID号对应的节点全局坐标，可以通过如下命令得到：

print gp position

ID号对应的单元重心全局坐标可以通过如下命令得到：

*:在一些有限元的文献中存在这样的错误观念：有限元解法能够精确符合支配方程组，动态解法与有限元解法相比不可能达到真实的平衡状态。事实上，两种解法都只是近似的满足支配方程组，但是剩余误差可以被控制在尽可能小的范围内。在FLAC3D中这个误差被量化为节点上的不平衡力与作用在节点上所有外力平均值的比值。这个误差的量度与有限元解法中的收敛准则其实是类似的。在两种算法中，只要误差小于期望值，计算就停止了。

print zone

在FLAC3D中网格的生成是为了调整和塑造各种形状的网络体以模拟现实问题的空间物理形状。

网格的生成是通过**GEN(GENERATE)**命令来实现的。网格的通用生成方法将在3.3章节部分加以表述。

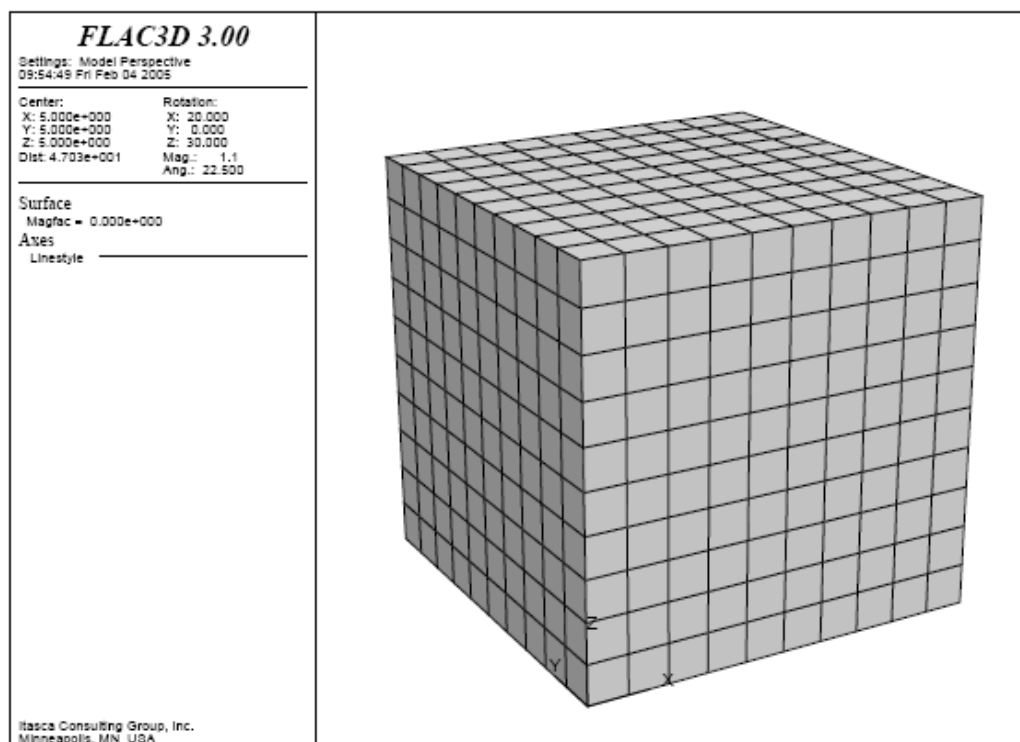


图2.8 1000个单元组成的有限差分网格

2.5 命令句法

所有在flac3d里输入的命令都有严格的语法约定，一般的形式为一个主命令后面跟一个或多个关键词以及关键词附带的参数。有些命令还有选择性，即关键词可以改变该命令的作用方式。每个命令都遵循如下的基本格式：

主命令 关键词 <参数值> ... <关键词 参数值 ...> ...

在上面的格式中，“<>”里的内容表示可选择部分（不一定要有），...表示多个参数（或关键词）。输入命令和关键词时可以输入整个单词也可只输入单词的缩写，其效果是一样的。（每个命令的缩写可参见**命令手册**部分）很多关键词都附带有参数，有的参数规定为实型数，有的则规定为整数。实型数如果没有小数位，则小数点及小数位（0）可以省略，但是小数点绝不能出现在整型参数

中。命令、关键词、参数间可以通过任意个空格间隔，也可以用以下符号分隔：

() , =

在命令行中有的时候还会出现分号(;)分号在程序中为注释符，分号开始到行末尾的内容均表示注释，在执行命令时，程序会忽略该部分的内容。注释在命令文件中是经常要用到的，有了注释命令文件才有更好的可读性，可以帮助用户提高查错、修改和理解命令文件的效率。

一个单一的命令行，包括注释部分，最多只能有80个字符。如果一个命令行的字符超出了该限定，那么可以将命令分为几行续写，除最后一行外，行的末尾要加上续行符(&)。续行符表示本行命令没有结束，下一行的内容和本行内容组成一个命令行。使用了续行符后，一个命令行所包含的字符也不能超过1024个。

2.6 对象的命名

Flac3d可以进行对象的创建、命名和使用。比如用户可以创建名为“Top_view”和“Velocity_Vectors”的视图窗口。在上面的例子2.2中我们就建立了一个新视窗，并且命名为“Trench”。我们还可以将部分的网格命名为诸如“Excavate_1”这样的名称。这样我们进行开挖时，就指定开挖的部分为“Excavate_1”对象里的所有网格就可以了。当然我们不创建开挖对象也可以完成相关的操作，但是肯定没有创建并使用对象那样来得方便。这部分主要告诉用户如何创建并使用对象，及各种对象之间的区别。

总的来说，对象分为两类：宏对象和模型对象。宏对象是先于被命令处理器执行之前被编译的；模型对象是在命令处理器中被编译的。用户输入的命令首先被命令预处理器接收，如果预处理器发现有宏对象，就进行宏的扩展。接着预处理器创建一个字符串输入到命令处理器中，处理器将字符串编译为可执行命令进行相关命名的操作。模型对象就是在这一步才被编译的。注意：宏对象名和模型对象名是区分大小写字母的。

(1) 宏对象

Macro命令可创建宏对象。在模型中如有多次重复使用且含有又长又复杂字符的对象时，使用宏命令就能带来不少方便。例题2.4中说明了怎么使用宏对象。注意宏嵌套的使用。

例题2.4 宏对象的使用

```
macro Pt0 'p0 0 0 0'
macro Pt1 'p1 add 10 0 0'
macro Pt2 'p2 add 0 10 0'
macro Pt3 'p3 add 0 0 10'
macro Model_Size 'size 4 5 6'
macro Big_Brick 'zone brick Pt0 Pt1 Pt2 Pt3 Model_Size'
gen Big_Brick
macro 'Pt0' 'p0 15 15 15'
gen Big_Brick
```

如果要重新定义一个宏，我们必须将宏名称用单引号括起来（见上例中倒数第二行），否则会得到不可预知的结果。预处理器在编译时会跳过引号，将其留到命令处理器中编译。

这个预处理有两个作用：

- （1）实现宏对象的嵌套（但不能是递归调用）。
- （2）预处理后宏对象名就不会出现在经过预处理器处理后产生的字符串中。

至于第二点作用再做一下说明：假如定义的宏对象名和命令或者模型对象名相同，那么如果在命令处理器中还出现宏对象名的话，那么相同名字的模型对象或命令将不能被识别。

还有一个特殊的宏对象：和符号（&），作用为续行。（在**2.5 命令句法**中已讲到）。具体使用方法见**例题2.5**

例2.5 使用续行符

```
gen zone brick p0 0 0 0 &
p1 10 0 0 &
p2 0 10 0 &
p3 0 0 10 &
size 4 5 6
```

该例中其实只有一个命令行：

```
gen zone brick p0 0 0 0 p1 10 0 0 p2 0 10 0 p3 0 0 10 size 4 5 6
```

一个宏对象不能采用多行陈述，也最好不要使用“&”符号来续行。如果避免不了要多行陈述，建议还是采用fish语言来实现相同的作用。

(2) 模型对象

模型对象包括空间区域的范围（range）、单元的组(group)、绘图视窗。用户可以对这些对象加以命名，并在操作时直接使用对象名来指定对应的对象。GROUP命令可以将部分的模型网格归为一组，这样就方便对这组网格进行开挖等操作了。例题2.6说明了如何创建和使用模型对象

例2.6 使用模型对象

```
gen zone brick size 6 6 6
group Tunnel range x 1 5 y 0 6 z 1 5
model mohr
...
...
model null range group Tunnel
```

空间区域的范围（range）、单元的组(group)是两个十分不同的模型对象。Range指的是空间区域的某一范围，而group指的是某一范围内的单元体，是有限差分网格单元的一个集合。

(3) 建议

首先建议用户这样理解对象：对象是可以被用户操作的一个实体，而不是对某一实体进行操作的一个动作指令。

用宏可以创建新命令，当然这里所指的命令并不是凭空创造的一个命令，可以参见例题2.9。

例题2.9 创建新命令

```
macro PA 'plot add'
...
...
PA history 1 ...
```

对象名最好采用大小写混合的形式，以避免和命令（关键词）发生意外的冲突。

2.7 用FLAC3D进行问题分析的过程

FLAC3D是基于命令驱动模式的软件，因为大多数的分析都需要用到输入文件。命令语句控制着分析的进程。当然用户交互式的图形控制界面在某些时候还是可以派上用场，比如在控制出图的时候会相对方便一些。这一部分主要讲解用FLAC3D进行简单计算所需要的基本命令的使用方法。如果你发现还是看不懂下面的内容，建议你回到2.2章节再仔细研究下例题中命令驱动方法，边看边操作着学习。

本章的教学案例——低粘聚力土体中沟渠的开挖问题。我们将用这个例题来讲述解决问题的基本过程。这个过程可以分为几个明显的阶段（参见图2.9）。每个阶段我们都分开来论述，每一个部分中都详述了所用到的各种命令，以使得用户更明白每个过程的操作。

要建立一个可以用FLAC3D来模拟计算的模型，必须要做以下三步工作：

- （1）建立模型的有限差分网格；
- （2）定义本构模型和赋材料参数；
- （3）定义边界条件，初始条件。

由网格来定义所要模拟的几何空间。由本构模型和材料参数来限定模型对于外界扰动做出的变化规律（比如开挖引起的变形反应）。由边界条件和初始条件来定义模型的初始状态（比如说模型在发生变化或扰动前的稳定状态）。

做好了以上三步工作，就可以进行模型初始平衡状态的计算了。接着对模型做一些变动（比如：开挖或者改变边界条件），然后再对改动后的模型进行计算。FLAC3D为采用显式解法的软件，它的实际求解过程不同于常规的隐式解法（具体参见第一章**理论和背景**中的背景部分）。FLAC3D是采用显式时间步推的方法来求解代数方程组的，通过一些时间步的计算，才会得到所要的计算结果。完成计算所需要的时间步可以由软件自动控制，也可以人为的指定计算步数。但最后，还是需要用户自己来判断进行了这些时间步的计算，模拟的问题是否已经得到了最终所要的解。至于如何加以判断，我们将在后面的2.7.4节介绍。

通过图2.9我们能够很清楚的了解FLAC3D是如何一步步的完成问题的模拟和求解的，那么接下来我们将叙述每一步是通过哪些命令来实现的。

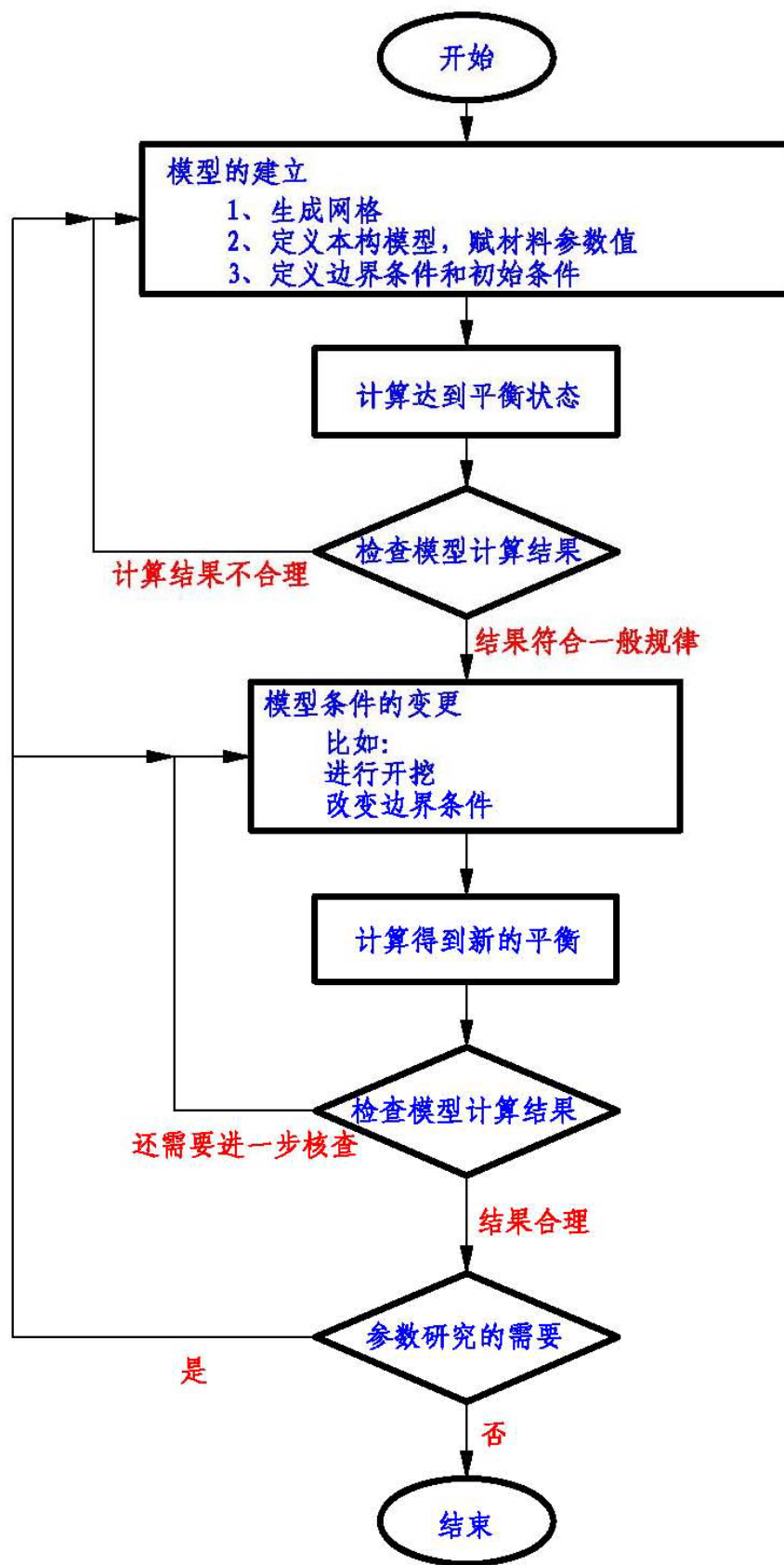


图2.9 FLAC3D基本求解过程

2.7.1 简单网格的生成

网格的生成是通过输入**GEN(GENERATE)+关键词**的命令来实现的。该命令既可以用来定义模型的网格数也可以规定网格的形状、大小来适应模型的空间区域。

在网格生成器中,提供了一些基本形状的网格以帮助用户快速的进行简单问题的建模。网格生成器中提供了以下一些基本形状:六面体网格,楔形网格,锥体网格,柱体网格。下面的**例子2.12**说明了怎样用命令来创建一个x方向划分成6格,y和z方向分别划分成8格的矩形网格体。

例题2.12 简单矩形网格的生成

gen zone brick size 6,8,8

plot surf

网格中的单元数由关键词**SIZE**来定义,网格的单元数不能信手定义,要把握好精度与计算用时之间的平衡尺度。虽然,数值模型的很多方面都会影响到计算的速度,但是一旦确定了模型的基本参数,计算速度就主要由模型的总单元数决定了。根据经验,一个大约由5000个单元组成的模型一般计算2000到4000步就可以完成求解。在主频为1GZ的奔腾电脑上,FLAC3D进行一个由5000个单元组成的模型的4000步计算大概也就5分钟的时间。在具体模型的计算中,你可以进行典型模型的测试,看看你的电脑计算速度有多快。

我们建议:最好开始的时候将网格剖分的相对稀疏一些(1000~1500个单元左右),先求解一下,看看结果,然后再根据大致的结果有针对性的增加模型的网格数,以得到更精确的解答。

GEN命令的一种最简单的形式可以在建立网格时给网格定义坐标系,比如,现假定开挖例题的模型在坐标系中的空间范围是这样的:x方向,-10~10;y方向,-10~10;z方向,-20~0。下面的例题2.3将告诉你这种情况的模型怎么定义。

例题2.13 定义全局坐标系

gen zone brick size 6,8,8 p0 -10, -10, -20 &

p1 10, -10, -20 &

p2 -10, 10, -20 &

p3 -10, -10, 0

plot surf

在例题中，**p0**、**p1**、**p2**、**p3**关键字代表了标准FLAC3D基本网格的四个角点（见图2.10），你会注意到在前三个命令行的最后出现了“&”符号，这表示，接下来输入的那行命令是本行命令的继续，也就是说由于命令太长了，一行写不下，只能分几行输入，那么命令行之间就是通过“&”符号来衔接的（前面的内容已经叙述了这方面的知识）。

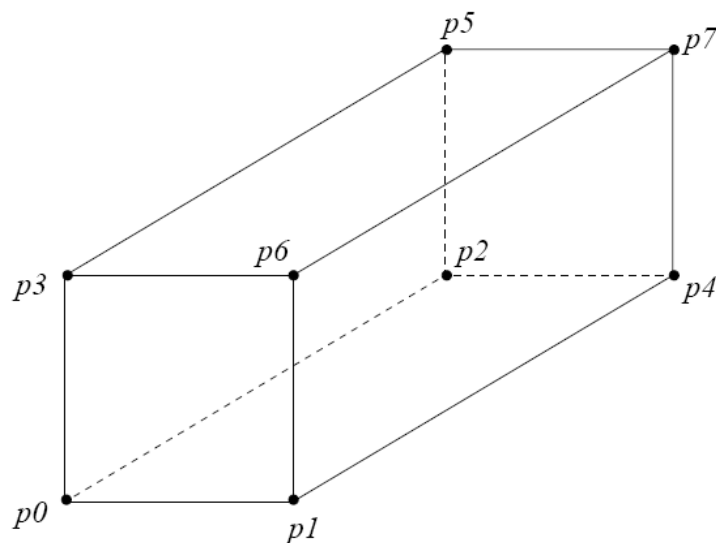


图2.10 标准FLAC3D网格的角点关键词位置

对于平行六面体网格，在命令中只需定义四个角点就可以建立网格了，对于不规则形状的网格，那就需要输入更多的角点信息来建立网格了。下面的例题将告诉你如何定义一个顶部为斜面的网格。

例2.14 顶部为斜面的网格的定义

```
gen zone brick size 6,8,8 p0 -10, -10, -20 &
```

```
p1 10, -10, -20 p2 -10, 10, -20 &
```

```
p3 -10, -10, 0 p4 10, 10, -20 &
```

```
p5 -10, 10, 10 p6 10, -10, 0 &
```

```
p7 10, 10, 10
```

```
plot surf
```

从理论上来说关键词**p0**、**p1**、**p2**、**p3... p7**在网格中对应的角点应该是任意的，但事实上，在软件中角点的定义必须满足图2.10中的排列方式（比如说：**p0**、**p1**、**p2**、**p3**必须满足右手法则）。

在之前的教学例题中我们知道，模型边界的不同会影响模型的计算结果。所以模型的边界应该要离开挖的沟渠足够的远，以减少边界对计算的影响。我们可以定义一种环绕的放射状网格。这种网格可以在不需大量增加网格数量的基础上将边界移到“更远”的地方。例如，我们可以使用 **GEN ZONE RADBRICK** 命令创建一个矩形外环绕放射状网格。具体命令的使用见例题2.15，在本例中，创建了这样一个混合网格：一个 $3 \times 5 \times 5$ 的矩形网格 + 在此网格外围创建的一个由7格单元组成的放射状网格。

例2.15 创建外环绕放射状网格

```
gen zone radbrick &  
p0 (0,0,0) p1 (10,0,0) p2 (0,10,0) p3 (0,0,10) &  
size 3,5,5,7 &  
ratio 1,1,1,1.5 &  
dim 1 4 2 fill  
plot surf
```

FLAC^{3D}中，三维坐标轴总是遵循右手法则的，并且系统在显示图形的时候，会将z轴默认为竖直方向。在例题2.14中我们就是采用了系统的默认设置，将Z轴设定为竖直方向，但这并不是规定死的，在例题2.15中，我们就定义了Y轴为竖直方向。只要是坐标系满足右手法则，我们可以在建立模型时任意定义坐标的方向。

在例题2.15中，**SIZE** 关键词定义了矩形网格的网格数和环绕其外的放射状网格的网格数（参见图2.11）；**RATIO**关键词定义了网格的间距，前三个值定义了矩形体三个坐标方向的比率，第四个值定义了环绕扩散单元的扩散比率，在上例中该值为1.5，即表示放射状网格单元的尺寸以1.5倍的倍率慢慢变大。**DIM**关键字定义了矩形单元的范围（在上例中为： $1\text{m} \times 4\text{m} \times 2\text{m}$ ）**FILL**命令定义在矩形区域填充单元，如果在DIM关键字没有FILL关键字，那么在长方体部分就不会生成单元。

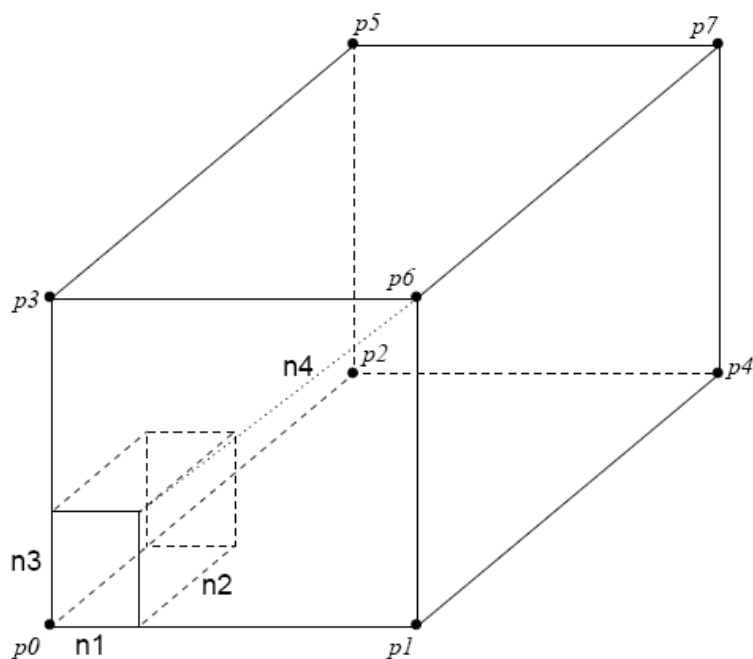


图2.11 六面体外环绕放射状网格的角点标号图

我们可以使用**GEN**命令来生成各种形状的网格，比如说**例题2.15**，用**GEN**命令创建了沟渠开挖问题1/4的网格。在接下来的**例题2.16**中，我们将告诉你怎么用一系列的**GEN**命令来创建整个模型。

例题2.16 GEN命令的用法

```
gen zone radbrick &
p0 (0,0,0) p1 (10,0,0) p2 (0,10,0) p3 (0,0,10) &
size 3 5 5 7 &
rat 1 1 1 1.5 &
dim 1 4 2 fill

gen zone radbrick &
p0 (0,0,0) p1 (0,0,10) p2 (0,10,0) p3 (-10,0,0) &
size 5 5 3 7 &
rat 1 1 1 1.5 &
dim 2 4 1 fill

gen zone radbrick &
p0 (0,0,0) p1 (-10,0,0) p2 (0,10,0) p3 (0,0,-10) &
size 3 5 5 7 &
```

```

rat 1 1 1 1.5 &
dim 1 4 2 fill
gen zone radbrick &
p0 (0,0,0) p1 (0,0,-10) p2 (0,10,0) p3 (10,0,0) &
size 5 5 3 7 &
rat 1 1 1 1.5 &
dim 2 4 1 fill
plot surf lblue

```

注意：SIZE RATIO DIM 都对应为由P0 P1 P2 P3 定义的局部网格坐标（见图2.11），并不对应模型整体坐标。最后由放射状网格建立的开挖模型见图2.12。

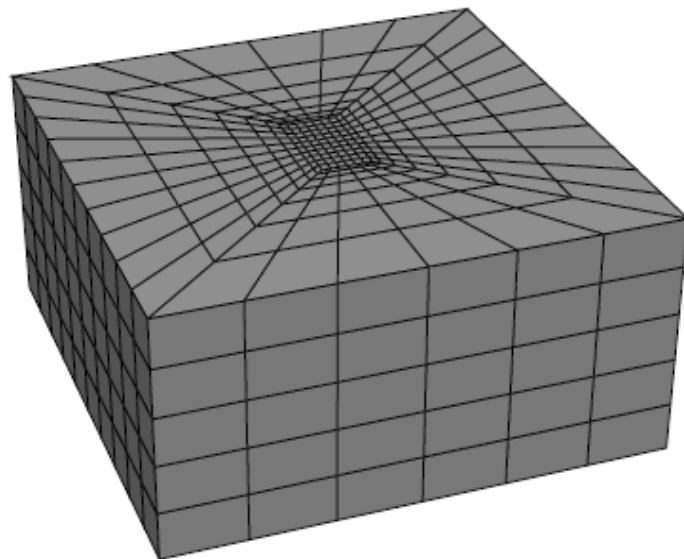


图2.12 放射状沟渠开挖模型

注意检查各网格之间的连接面上的节点是否相互匹配，如果两个连接面上的节点坐标不一致那么在计算时就会发生错误。我们将在第三章进一步讨论如何创建更复杂的网格以及网格节点的容许误差问题。

我们还可以利用模型的对称性，用另一种方法建立图2.12中的模型。我们可以建立模型的一部分，然后利用对称面将网格扩大到整个模型区域。例题2.17将告诉我们怎么实现这个过程。

例题2.17 利用对称面建立模型

```

gen zone radbrick &

```

```
p0 (0,0,0) p1 (10,0,0) p2 (0,10,0) p3 (0,0,10) &  
size 3,5,5,7 &  
ratio 1,1,1,1.5 &  
dim 1 4 2 fill  
gen zone reflect dip 0 dd 90  
gen zone reflect dip 90 dd 90  
plot surf
```

在确定模型的比率（ratio）时要考虑到以下一些准则：

- 1、在应力变化范围较大的情况下，好的网格形状会使得计算结果更加的准确。
- 2、单元各方向的比率越趋于一致，计算的精度就会越高。
- 3、如果在模型中需要有不同大小的网格，那么建模时最好是由最小的网格慢慢渐变到最大的网格，尽量避免两个相邻网格尺寸的突然变化。

当然，模型的网格形状越好，网格划分的越密，那么所占用的内存自然就越大，计算所需要的时间也就越长。我们可以通过调整单元尺寸的方法来优化我们的模型，这样就在尽可能少的占用系统内存，减少时间所需要的时间。当然我们需要花一些时间来设计和执行优化方案。在复杂问题中，解的精度越高，那么所占用的系统内存就会越多，所需要的计算时间也会越长，这是普遍的规律。生成网格的一些方法和技巧我们将在3.3节中加以讨论。

2.7.2 定义材料模型（本构模型）

网格建好后，接下来我们就需要给网格定义一种或几种本构模型以及其对应的材料参数了。这一步工作需要用到两个命令：**model**和**property**。FLAC3D内置了12种本构模型（在2.1 理论与背景那一节中已经叙述过了）。一般新用户在分析问题只要知到其中的三种就足够了：**model null**（空模型）、**model elastic**（弹性模型）、**model mohr**（摩尔 - 库伦模型）。

model null（空模型）：可以用来定义那些被移除或开挖掉的区域的材料；

model elastic（弹性模型）：用来模拟弹性材料；

model mohr（摩尔 - 库伦模型）；用来模拟摩尔 - 库伦塑性材料。

如果是定义了后两种本构模型，那么还需要通过**PROPERTY**命令给模型赋

予材料参数值。如果是弹性模型，那么只需要赋两个材料参数就可以了：**bulk**（体积模量）；**shear**（切变模量）。注意体积模量**bulk**（**K**）、切变模量（**G**）和杨氏模量（**E**）及泊松比（**ν**）有如下关系：

$$K = \frac{E}{3(1-2\nu)}$$

$$G = \frac{E}{2(1+\nu)}$$

或者：

$$E = \frac{9KG}{3K+G}$$

$$\nu = \frac{3K-2G}{2(3K+G)}$$

如果是摩尔-库伦模型，那么就需要输入以下的材料参数：

- （1）体积模量、切变模量
- （2）内摩擦角（**friction**）和剪胀角（**dilation**）
- （3）粘聚力（**cohesion**）
- （4）抗拉强度（**tension**）

如果没有定义其中的一种材料参数，系统会默认它的值为“0”。

在一个模型中，不同区域的网格可能需要定义成不同的本构模型。比如将上半部分10×10×10m的网格定义成弹性材料模型，将下面其它部分的网格定义成摩尔-库伦模型。**例题2.18**将告诉我们配合使用**range**关键词怎么通过命令实现这个过程。顺便提一下：我们在例题中用宏预先定义了不同的类型的土。当然宏里面本还可以定义该类土所在的区域，但是建议不要将材料参数和对应的区域定义在同一个宏里，我们应该养成这样的习惯。因为这样我们可以方便的借用宏将不同的区域定义成相同的材料，并赋予材料参数。

例题2.18 给一个简单的网格中不同的区域定义不同的本构模型

```
gen zone brick size 10,10,10
```

```
macro SiltySand 'bulk 1.5e8 shear 0.3e8'
```

```
macro ClayeyGravel ' bulk 1.5e8 shear 0.6e8 fric 30 coh 5e6 ten 8.66e6'
```

```
; 网格上半部分定义为弹性模型
```

```
model elas range z=5,10
```

```
prop SiltySand range z=5,10
```

；下半部分定义为摩尔 - 库伦模型

```
model mohr range z=0,5
```

```
prop ClayeyGravel range z=0,5
```

```
plot block model lmagenta lgreen
```

为了避免在输入命令时总要通过 **range + 全局坐标域** 的方法来限定该命令执行的区域，我们可以先通过 **range** 命令来给不同的区域命名，然后就可以用 **range + 区域名** 的方法来限定范围了。这样就避免了每次输入命令要限定在该区域执行时，都要输入一遍坐标范围的麻烦。具体如何给区域命名，以及日后如何用区域名来限定该区域，可以看下面这个例子：

例2.19 创建和使用区域名

```
gen zone radbrick &
```

```
p0 (0,0,0) p1 (10,0,0) p2 (0,10,0) p3 (0,0,10) &
```

```
size 3,5,5,7 &
```

```
ratio 1,1,1,1.5 &
```

```
dim 1 4 2 fill
```

```
gen zone reflect dip 0 dd 90
```

```
gen zone reflect dip 90 dd 90
```

```
;
```

；定义开挖的范围

```
range name trench x=-1,1 y=0,4 z=-2,2
```

```
;
```

；进行开挖

```
model null range trench
```

```
plot surf lblue
```

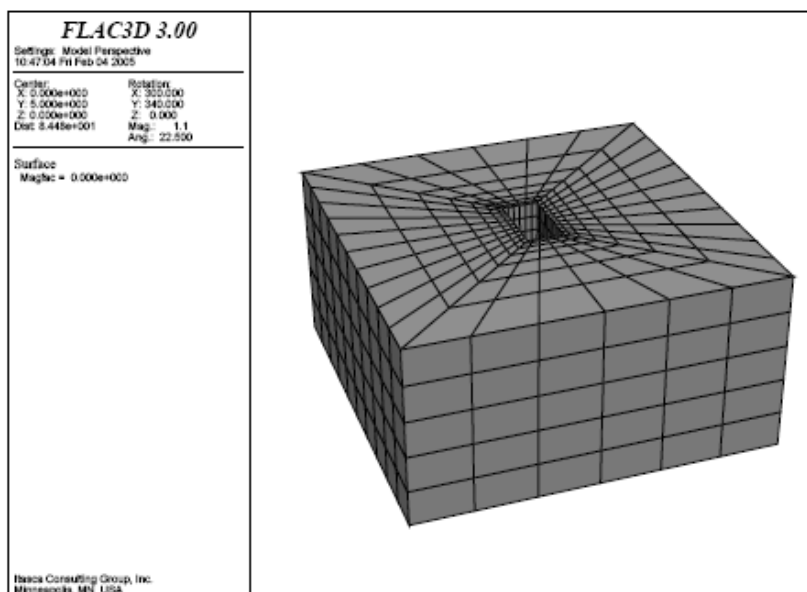


图2.13 放射状沟渠模型的开挖

2.7.3 设定边界和初始条件

网格生成后还需要设定模型的边界和初始条件。边界条件的定义通常要用到 **apply**、**fix**、**free** 这三个命令；初始条件的定义则要用到 **ini (initial)** 命令。表2.3、表2.4罗列了这些命令的所有关键词，以及这些关键词的作用。

在数值模拟中，边界条件包括了一切可以用具体变量值来描述的网格边界情况。注意：一旦给模型定义了边界条件，那么在FLAC3D计算模拟的过程中，边界条件始终不会发生变化（除非用户在此后做了修改）。这里我们所说的边界可以是真实边界，也可以人为边界。真实边界是模型在现实状况中实际存在的边界，但人为边界是用户为了使模型封闭而假定的边界。

人工边界又分为两类：对称面和切断面。对称面边界是利用了模型及其所受荷载沿一个或多个面的对称性而设定的；切断面边界是由于实际问题的边界广阔无边，或者相对于所要关注的区域来说非常的大而在建模时考虑到计算和内存的要求，只取实际区域的一部分来模拟问题而产生的边界面。当然在建模的时候必须要保证切断面离所要关注的区域足够的远。至于到底取多远，也是一个十分重要的问题，在具体问题中我们应该了解模型的切断面边界对所关注区域的应力，应变有什么样的影响，影响有多大，再确定切断边界面的位置。这样建模时才可能尽量减少切断面边界对所关注区域的影响。

表2.3 常用APPLY、FIX命令关键词说明表

命令		作用
节点边界条件		
APPLY	Xforce (xf)	施加节点力X方向的分量
	Xvelocity(xvel)	施加节点速度的X方向分量
	yforce (yf)	施加节点力y方向的分量
	yvelocity(yvel)	施加节点速度的y方向分量
	zforce (zf)	施加节点力z方向的分量
	zvelocity(zvel)	施加节点速度的z方向分量
面边界条件		
	sxx	施加边界面上应力张量x方向的分量
	sxy	施加应力张量xy方向的分量
	sxz	施加应力张量xz方向的分量
	syy	施加应力张量y方向的分量
	syz	施加应力张量yz方向的分量
	szz	施加应力张量z方向的分量
FIX	x	固定边界节点X方向的速度分量
	y	固定边界节点Y方向的速度分量
	z	固定边界节点Z方向的速度分量

注意:

- 1、Free命令是用来释放由“Fix”设置的约束。
- 2、如果想要约束边界上的位移，只需用到fix命令就可以了，因为fix命令约束了该节点的速度为“0”，没有速度，那么自然不会有位移。
- 3、见第一章 命令手册，来查看所有APPLY的关键词。
- 4、如果是压应力，那么赋值的时候应该是负值，可参见2.8节来了解符号的规定。

表2.4 常用INITIAL命令关键字说明表

命令		作用
节点条件		
INITIAL	Xdis	设定节点x方向位移初始值
	Ydis	设定节点Y方向位移初始值
	Zdis	设定节点Z方向位移初始值
	Xvel	设定节点x方向速度初始值
	Yvel	设定节点Y方向速度初始值
	Zvel	设定节点Z方向速度初始值
单元条件		
	Density	设定单元的密度
	SXX	设定单元体的应力xx方向分量初始值
	SXY	设定单元体的应力xy方向分量初始值
	SXZ	设定单元体的应力xz方向分量初始值
	SYX	设定单元体的应力yx方向分量初始值
	SYZ	设定单元体的应力yz方向分量初始值
	SZZ	设定单元体的应力zz方向分量初始值

FLAC3D中，系统默认情况下，边界条件完全自由，不受应力或者位移等条件的约束，需要通过**APPLY**命令给所有边界或者部分边界加上外力或者应力。各个方向应力分量的施加所用到的关键字可以参考表2.3。比如说在X = 10这个边界面上现在要施加一个x方向的常压应力，应力值为10MP，那么输入的命令应该这样：

```
apply sxx -10e6 range x 9.9 10.1
```

记住，压应力在输入的时候应该是负值。在命令行中，使用了range关键字来规定**apply**命令所作用的范围。在刚才那个命名行中，**apply**命令所作用的范围就是夹在平面x=9.9和x=10.1之间的所有边界面。

如果要设定一个y方向的压应力，并且应力值是线性变化的，我们就需要用**gradient (grad)**关键字。

```
apply syy = -20e6 grad 0,0,20e5 range y -20.1,-19.9 z 0,10
```

这个命令将应力施加到了夹在 $y=-20.1$ 和 $y=-19.9$ 平面间的边界面上，并且在 $z=10$ 处应力值为“0”， $z=0$ 处应力值为 20×10^6 （负号说明是压应力）。

在使用关键字`grad`时，值的线性变化遵循下面的公式：

$$S=S^{(0)}+g(x)x+g(y)y+g(z)z$$

式中： $S^{(0)}$ 为变量在全局坐标系原点处的值，是在命令中，紧随在关键字`syy`后的值； $g(x)$ 、 $g(y)$ 、 $g(z)$ 分别指变量在 x 、 y 、 z 方向的变化率。这三个值在命名行中应接在关键字`grad`后面。

在FLAC3D中，用户不能直接控制位移，事实上，在计算中根本没有涉及位移。为了给边界施加一个已知的位移值，那么就必须定义在计算达到指定步数的时候边界处的速度值。比如说已知的位移值为 D ，那么所设定的计算步数（ n ）和速度（ v ）就应该有这样的关系： $n=D/v^*$ 。在实际问题中，速度应该取得很小，而时间步数应该取得比较大，以尽量减小在初始化的时候对模型的扰动。

下面的例子将告诉我们如何用`fix`命令约束 $x=0$ 的边界面上的位移值为“0”：

fix x range x -0.1 0.1

这个命令相当于定义了一个连杆支承的边界，连杆固定了 x 方向的位移，但是在 y 、 z 两个方向是自由的。

如果是要约束边界的几个方向（比如 x y z 三个方向全部受约束），那么类似的，用下面的命令就可以做到了：

fix x y z range x -0.1 0.1

对于真实边界，一般很容易确定一个边界到底是位移边界还是应力边界。但是对于人工边界，比如切断面边界，那么到底应该算位移边界还是应力边界就不那么好判断了，现在告诉你一般判断的经验：

- （1）位移边界：忽略了人工边界上应力和位移；
 - （2）应力边界：强调和夸大了人工边界上应力和位移。
 - （3）真实解应该位于这两者之间，所以有必要的話，可以将以上两种情况都进行计算，并最终综合平均两个计算结果，对真实情况做一个估算。
- 初始条件是为了模拟模型达到初始状态而设置的一些变量的初始值，这些值

*:速度在静态解模式中，指每个计算步产生的位移；在动态解模式中，指单位时间产生的位移。

可以在计算的过程中加以更改。比如在岩土材料的模拟中，岩土材料在开挖或者扰动前，本身具有初始的地应力。因为初始地应力会影响到模型以后（加载或者扰动）的变化，所以我么就需要设置一个初始条件来使模型的网格单元在初始状态下就获得初始地应力。（我们将在**3.3节**详细介绍会有怎样的影响）

为了描述初始状态应力，我们举例来说明INI命令的使用：

```
ini sxx -50e6 syy -40e6 szz -10e6
```

这个命令使得模型中所有网格在初始状态就获得了压应力，其中：x方向的压应力大小为 50×10^6 ，y方向的压应力为 40×10^6 ，z方向的压应力为 1×10^6 。我们可以用INI命令来定义所有六个应力分量的初始值，也可以像前面定义边界条件那样配合使用grad关键字来定义线性变化的初始应力。如果初始应力只是由于重力引起，那么我们可以通过：set gravity命令来施加应力，例如：

```
set gravity = 0, -9.81, 0
```

在上面的命令中，我们在y负方向定义了重力加速度值为 9.81m/s^2 ，如果指定了重力加速度，那么还必须指定模型材料的密度，定义密度可以使用INI density(de)命令。

现在让我们利用已掌握的知识对在2.2节中介绍过的开挖模型重新进行建模。这个问题中我们可以定义两个通过模型对称中心的对称面人工边界，一个是x-y平面，另一个是y-z平面（可参见图2.12）。在下面的例子中，我们利用对称面，只建立了1/4的模型，并定义模型为摩尔-库伦材料模型，赋予相应的材料属性值。还定义了边界条件以及考虑自重的初始条件，模型见图2.14。

例题2.20 开挖模型——利用对称面建模

```
gen zone radbrick &
p0 (0,0,0) p1 (10,0,0) p2 (0,10,0) p3 (0,0,10) &
size 3,5,5,7 &
ratio 1,1,1,1.5 &
dim 1 4 2 fill
mo moh
prop bulk 1e8 shear 3e8 fric 35 coh 1e3 tens 1e3
fix x range x -0.1 0.1
fix z range z -0.1 0.1
```

```
fix y range y 9.9 10.1
set grav 0 10 0
ini dens 1000
ini syy 0.0 grad 0 -10000 0
ini sxx 0.0 grad 0 -5000 0 ; （取值为原来的一半）
ini szz 0.0 grad 0 -5000 0 ; （取值为原来的一半）
apply sxx 0.0 grad 0 -5000 0 range x 9.9 10.1
apply szz 0.0 grad 0 -5000 0 range z 9.9 10.1
```

在这个例子中，我们定义x方向和z方向正应力时，应力值和变化增量值取为原来整个模型中的一半。初始化应力值可以定义成任何大小（应与实际相符），但是我们必须应该检查初始化后的应力水平，以保证没有违背模型的屈服准则（在我们这个问题中，模型为摩尔-库伦模型）。我们将在下一部分详细讨论模型的初始平衡状态。

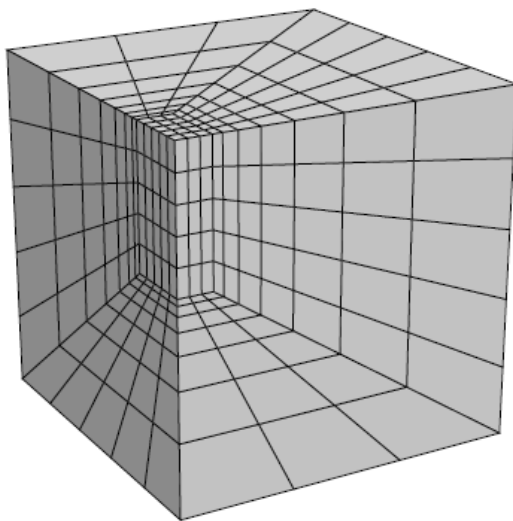


图2.14 利用对称面建立的开挖模型

2.7.4 计算以达到平衡状态

FLAC^{3D}中，在模型发生变化以前必须使模型先达到一个初始的力的平衡状态。如果是简单的模型，我们可以直接定义模型的边界条件和初始条件，模型可能自然的已经达到了初始的平衡状态。但是，在大多数情况下，模型需要在给定边界条件和初始条件的前提下，进行一些计算步的计算来使模型达到初始平衡状态，尤其在模型比较复杂或者模型中有多种材料的时候。这时就需要用到step或

者`solve`命令了。如果使用`step`命令，那么用户需要指定计算步数，以使得模型在进行了这些步的计算后达到初始平衡状态。如果网格每个节点的节点力都为“0”，那么表示模型达到了绝对平衡状态。（见第一章**理论背景**中的介绍）。最大的节点力的值（也叫最大不平衡力；不平衡力）在使用`step`命令进行计算时，会自动的被软件监视，并在计算过程中，显示在屏幕上。这样，用户就可以估计计算何时会达到平衡状态。

在数值模拟中，最大不平衡力是不可能为“0”的。但是当模型中的最大不平衡力与典型的单元力相比可以忽略不计时，我们便认为模型已经达到了平衡状态。比如在未初始化前，模型的最大不平衡力是1MN，计算后，它的值下降到了100N，那么我们可以认为模型已经达到了平衡状态，因为计算后的值是原来值的0.01%。这是FLAC^{3D}数值计算中很重要的一个知识点。用户必须确定模型在计算后已经达到了平衡状态。在FLAC^{3D}中，有一些条件可以帮助用户判断模型是否达到了平衡状态。最大不平衡力随着计算步的发展历程可以用以下命令记录下来：

```
hist unb
```

顺便说一句：任何变量都可以用`hist`命令来记录它随计算步的变化过程（比如某节点的速度、位移等），可以举个例子来说：

```
hist gp xvel 5,4,4
```

```
hist gp zdisp 0,0,8
```

命令的第一行，记录了（5，4，4）这个节点些x方向的速度值；第二行记录了（0，0，8）节点z方向的位移值。

进行了上百步，或者上千步的计算后，我们可以绘制出这些被记录变量随时间变化的曲线图，并用它们来判断模型是否达到了平衡。**例题2.21**告诉我们怎么来完成这个过程

例题2.21 计算达到初始平衡状态

```
gen zone brick size 6 8 8
```

```
mo elas
```

```
pro bulk 1e8 shear 0.3e8
```

```
fix x range x -0.1 0.1
```

```
fix x range x 5.9 6.1
```

```
fix y range y -0.1 0.1
fix y range y 7.9 8.1
fix z range z -0.1 0.1
apply szz -1e6 range z 7.9 8.1
hist unb
hist gp xvel 3 4 4
hist gp zdisp 0 0 8
step 1500
; solve ( 可以用step命令也可以用solve命令 )
```

模型最开始的最大不平衡力为1MN，经过1500步计算后，最大不平衡力降到了大概270N，通过绘制第一个历史记录变量图，我们可以看到，最大不平衡力已经接近于“0”了。输入下面的命令，就能在FLAC3D中显示图2.15中的图像：

```
plot hist 1
```

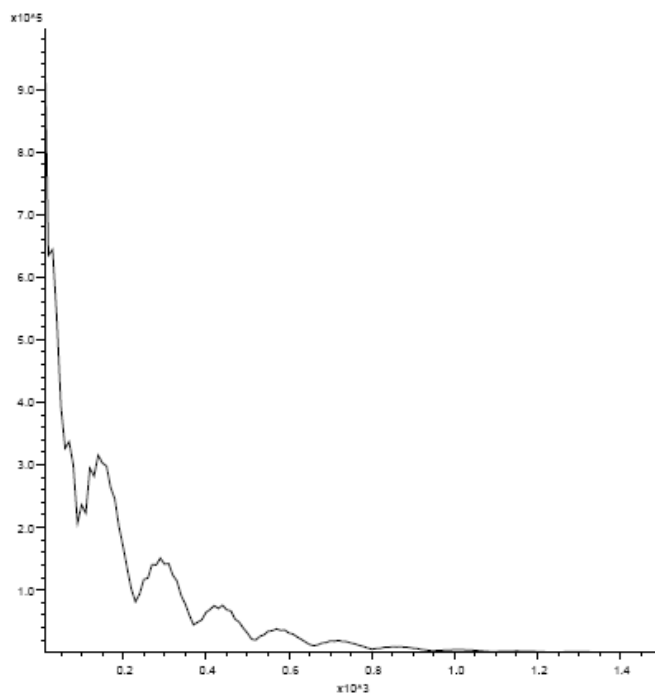


图2.15 最大不平衡力记录图

输入：

```
plot hist 2
```

```
plot hist 3
```

可以分别看到所记录节点的速度记录图（见图2.16）和位移记录图（见图

2.17)。在图2.16中可以看到速度值已经接近于“0”；在图2.17中我们也可以看到位移值已接近某一固定值。

以上这些条件都说明一点：模型已经达到了初始平衡状态。

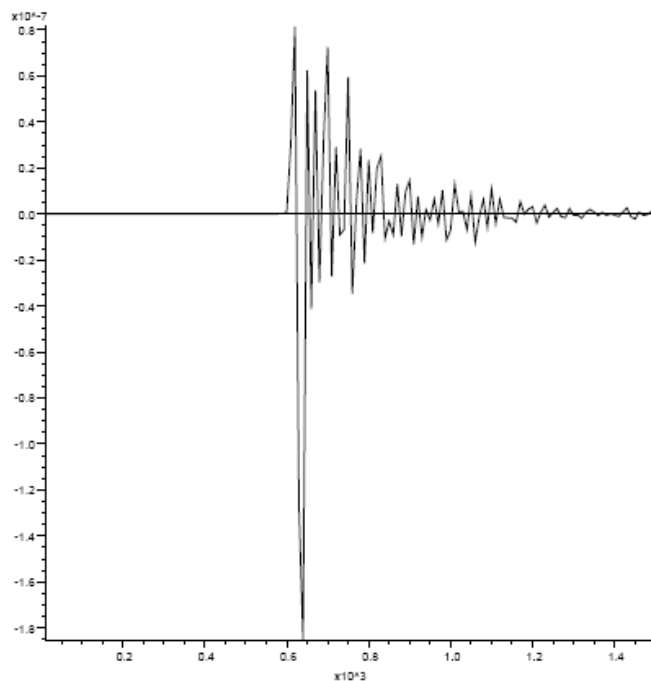


图2.16 节点（3，4，4）x方向速度记录图

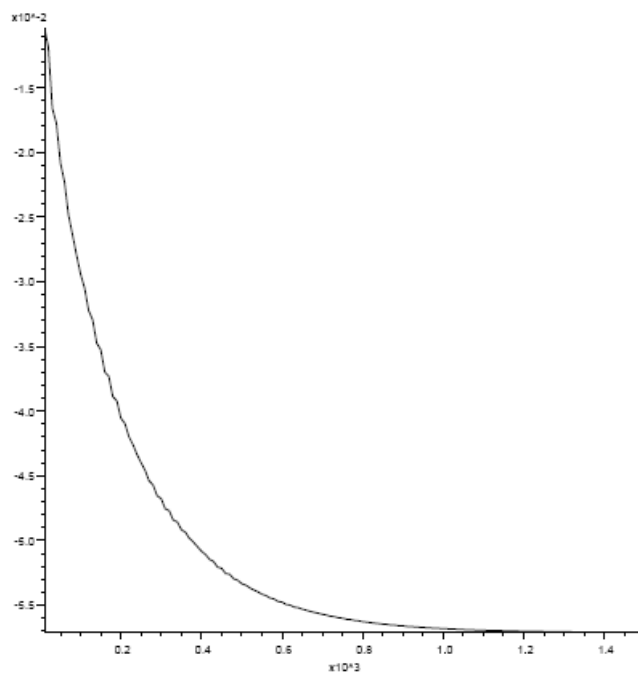


图2.17 节点（0，0，8）z方向位移记录图

如果用户想让FLAC3D自动来控制何时计算结束（当最大不平衡力小于某一限定值时），那么可以不用`step`命令而改用`solve`命令。在上面的例子中，可以用`solve`来取代`step 1500`。这次，计算将在1650步停止。如果还记录以上那些变量的历史记录，那么绘出的图形总体上应该还是和前面的三张图一样的。

如果我们使用了`solve`命令，默认情况下，系统通过最大不平衡力比率来控制计算过程，当最大不平衡力与最初施加的节点力的平均值之比小于 1×10^{-5} ，计算就会停止。我们也可以在输入`solve`命令之前通过输入以下命令人为的设置这个比率：

```
set mech ratio = f
```

此处，f是一个用户给定的比率限定值。

回过来看**例题2.21**，我们可以只用`INI`命令来给模型进行初始化，而不用经过任何计算步就使模型达到初始平衡状态：`ini sxx=-1e6 syy=-1e6 szz=-1e6`现在最大不平衡力就已经接近“0”了*。如果不信，我们可以用`solve`命令来证实一下。注意，在这种情况下，模型初始位移自然而然的为“0”。

我们再回到开挖问题中，从**例题2.20**中的命令来看，模型应该处于应力平衡的状态。我们在输入**例题2.20**的命令后再输入：

```
step 1
```

你会发现最大不平衡力大概是7000N，与这个问题中最大的自重应力（大概 2.7×10^5 N）比起来已经很小了。当然因为采用了环绕扩散网格的原因，模型中还是存在不平衡力。如果模型都采用相同尺寸的单元体网格，那么最大不平衡力应该几乎等于“0”了。如果在模型中网格单元有梯度变化，大小不一，那么我们就需要通过一些计算步的计算，使模型达到初始平衡状态。我们如果在**例题2.20**中输入`solve`命令，那么应该计算应该在1040步时停止，这时最大不平衡力比率应该小于 1×10^{-5} 。

在问题的模拟分析中，要确保模型在变动前达到初始平衡状态，这一点非常重要。我们应该通过`history`命令来记录一些变量随计算步的变化过程，以帮助我们判断模型是否达到了平衡状态。如果计算步数过少，模型其实没有达到平衡状

*:不可能精确等于“0”，因为在计算中引进了单精度和双精度运算。

态，那么对问题的分析会产生较大的影响。相反的，计算过多的步数（模型可能早已达到平衡状态了），对问题的分析丝毫没有影响（应该说反而使得解更精确了），但是花费了较多的时间。

在FLAC3D计算过程中的任何时候，我们可以按下键盘上的“ESC”键，来暂停计算。这经常会在step命令求解的情况下用到，尤其设定的求解步数非常多。我们可以按下“ESC”键，来暂停计算，然后查看历史记录项，再可以恢复计算，直到计算达到平衡状态。

2.7.5 模型条件的变更

FLAC3D在计算的任何时候都允许用户更改模型条件，这些条件包括：

- （1）材料的开挖
- （2）增加或删除节点处的外荷载或者应力
- （3）改变模型中任何单元的本构模型或者改变材料参数。
- （4）固定或者释放节点处的约束

开挖是通过model null命令来执行的；在任何节点上施加节点力可以用apply xforce(yforce、zforce)命令；边界应力的改变之前也已经叙述过了，可以使用apply命令；本构模型和材料参数值的改变可以使用model和property命令；约束（或释放）节点，可以使用fix(或free)命令。很显然，我们可能需要重复使用某些命令来模拟模型条件的变化。

如果模型中包括塑性材料本构模型（比如摩尔－库伦模型），那么可能在模型条件发生变更后，模型达不到平衡状态，换言之，部分或全部模型的最大不平衡力将会趋近于某一非零值，而不是趋于“0”。这就表明模型达到了一种稳定的塑性流动状态。（比如部分或全部模型正在发生垮塌）

在我们的开挖沟渠模型中，我们利用例2.20中的命令来建立和开挖模型，然后再利用下面例2.22中的命令来监测模型做出的反应。

例2.22 开挖沟渠并监测模型反应

```
range name trench x=0,1 y=0,4 z=0,2
model null range trench
set large
hist gp xdisp 1,0,0
step 1000
```

输出的位移等值线图(参见图2.18)和开挖边坡位移历史记录图(参见图2.19) (位移一直在增加) 加都表明了模型正在发生垮塌。

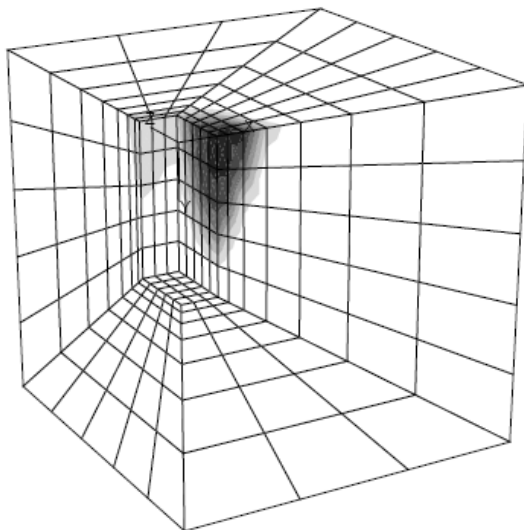


图2.18 开挖模型位移等值线图

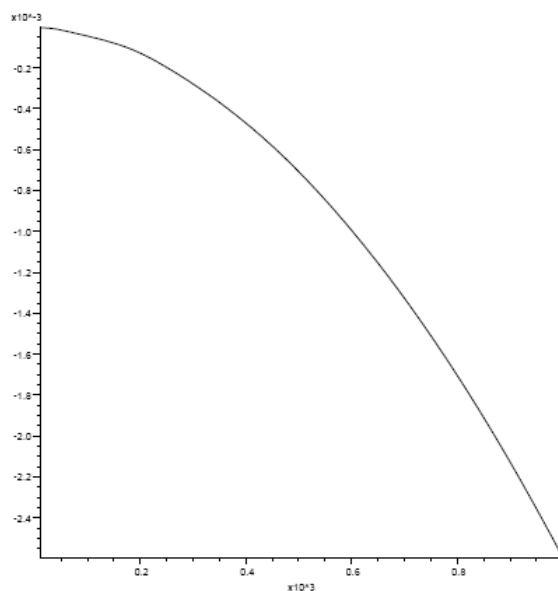


图2.19 开挖边坡的位移历史记录图

2.7.6 保存和恢复模型的某个状态

在模型的分析阶段，还有两个命令对我们有很大的帮助，即：**SAVE**命令和**RESTORE**命令。（也可通过FILE下拉菜单中相应的子选项执行）计算达到某种状态后（比如初始平衡状态），这个状态可通过以下命令来进行保存（生成一个

二进制文件）。

save file.sav

在此，file为用户自定义的一个文件名；扩展名sav表示该文件是一个用户保存文件（见2.11节）。在以后，用户想要模型回到这个状态，就可以输入以下命令：

rest file.sav

我们没有必要每改变一次状态就建一次模型，那样的话太费时费力了，我们完全可以通过存储状态的方法，以便日后对模型状态更改时可以调用更改前的模型状态。

拿我们的沟渠开挖模型来说，我们可以先保存初始平衡状态；然后分步进行沟渠的开挖直至挖深为2m，并保存各个开挖状态，那么我们就可以分析到底在哪个阶段沟渠开始垮塌，并进行支护（比如锚杆和锚钉的锚固）的加固分析。

我们再回过来看例2.20，这一次我们与例2.20不同，我们采取分步开挖，并保存初始平衡状态和每个开挖状态。具体命令见例2.23。

例2.23 开挖模型——分步开挖和支护

```
gen zone radbrick &
p0 (0,0,0) p1 (10,0,0) p2 (0,10,0) p3 (0,0,10) &
size 3 5 5 7 &
rat 1 1 1 1.5 &
dim 1 4 2 fill
model mohr
pro bu 1e8 sh .3e8 fric 35
pro co 1e3 ten 1e3
fix x range x -0.1 0.1
fix z range z -0.1 0.1
fix y range y 9.9 10.1
set grav 0 10 0
in dens 1000
in syy=0.0 grad 0,-10000,0
in sxx=0.0 grad 0,-5000,0
```

```
in szz=0.0 grad 0,-5000,0
apply sxx=0.0 grad 0,-5000,0 range x 9.9,10.1
apply szz=0.0 grad 0,-5000,0 range z 9.9,10.1
set large
hist unb
solve
save tr_eq.sav
; monitor trench excavation ( 监测开挖过程 )
ini xdis=0 ydis=0 zdis=0
h gp xdisp 1,0,0
h gp zdisp 0,0,2
; excavation steps ( 分期开挖 )
range name exc1 x 0 1 y 0.0 0.8 z 0 2
range name exc2 x 0 1 y 0.8 1.6 z 0 2
range name exc3 x 0 1 y 1.6 2.4 z 0 2
range name exc4 x 0 1 y 2.4 3.2 z 0 2
range name exc5 x 0 1 y 3.2 4.0 z 0 2
; excavation step 1 ( 一期开挖 )
model null range exc1
step 1000
save exc1.sav
; excavation step 2 ( 二期开挖 )
model null range exc2
step 1000
save exc2.sav
; excavation step 3 ( 三期开挖 )
model null range exc3
step 1000
save exc3.sav
```

```
; excavation step 4 ( 四期开挖 )
model null range exc4
step 1000
save exc4.sav

; excavation step 5 ( 五期开挖 )
model null range exc5
step 1000
save exc5.sav

; install cable support ( 进行锚索支护 )
rest exc2.sav

sel cable beg 1.0,0.4,1.5 end 5.0,0.4,1.5 nseg 4
sel cable beg 1.0,0.4,0.5 end 5.0,0.4,0.5 nseg 4
sel cable beg 1.0,1.2,1.5 end 5.0,1.2,1.5 nseg 4
sel cable beg 1.0,1.2,0.5 end 5.0,1.2,0.5 nseg 4
sel cable prop emod 2e9 ytens 1e8 xcarea 1.0 &
gr_coh 1e10 gr_k 2e9 gr_per 1.0
model null range exc3
step 1000
save cab3.sav

sel cable beg 1.0,2.0,1.5 end 5.0,2.0,1.5 nseg 4
sel cable beg 1.0,2.0,0.5 end 5.0,2.0,0.5 nseg 4
model null range exc4
step 1000
save cab4.sav

sel cable beg 1.0,2.8,1.5 end 5.0,2.8,1.5 nseg 4
sel cable beg 1.0,2.8,0.5 end 5.0,2.8,0.5 nseg 4
model null range exc5
step 1000
save cab5.sav

plot create TrenchSupport
```

```

plot set rot 290,350,0
plot set cent 2.579,3.855,6.1
plot set mag 1.56
plot set dist 32.65
plot add cont disp out on
plot add sel cable force
plot show

```

运行以上命名到 “; install cable support（进行锚索支护）” 之前一步时，我们可以用 **rest** 命令来回到各个状态观察历史记录项，通过观察我们发现模型在二期开挖时发生垮塌（见图2.20 x方向和z方向的历史记录图）那么我们就需要回到二期开挖状态进行锚索加固措施，所以接着输入命令，使模型回到二期开挖状态。我们在这个状态先进行支护，然后接着开挖（参见**命令手册**第一章中有关 **sel cable** 命令的介绍）。经过支护处理，我们发现开挖边坡已经稳定了，不再发生垮塌现象（见图2.21）。锚索的受力状态和沟渠附近的位移可以见图2.22。

通过保存和恢复状态操作，我们可以在模拟过程中，在任何状态改用不同的模型状态进行问题的研究分析，然后进行有关参数及问题的研究。比如上个例子中我们可以改变锚索的间距和材料参数，以进行锚索加固的优化研究。

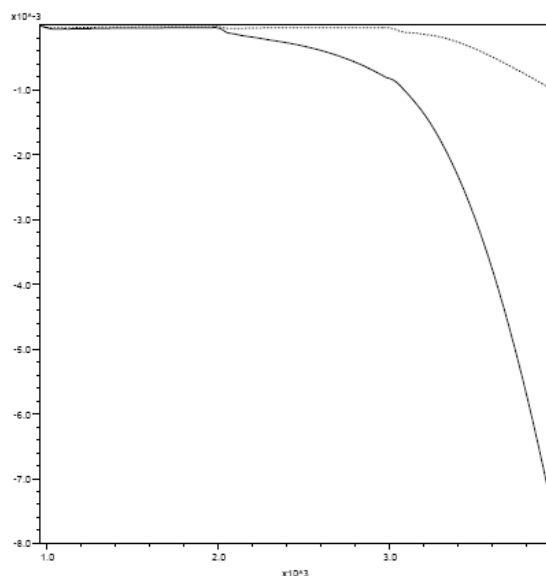


图2.20 三期开挖发生垮塌的位移历史记录图*

*笔者注：要在一个图中显示多个历史记录项，可以在 **hist** 后输入多个历史记录项的编码，比如上图是通过输入：**hist 2 3** 得到的。

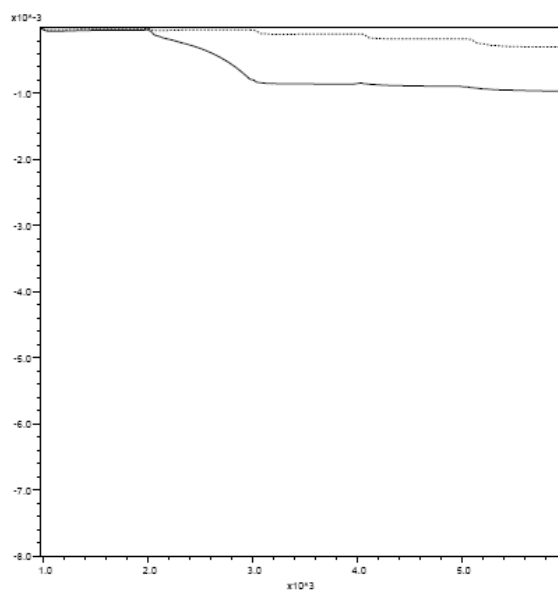


图2.21 锚索加固后的位移历史记录图

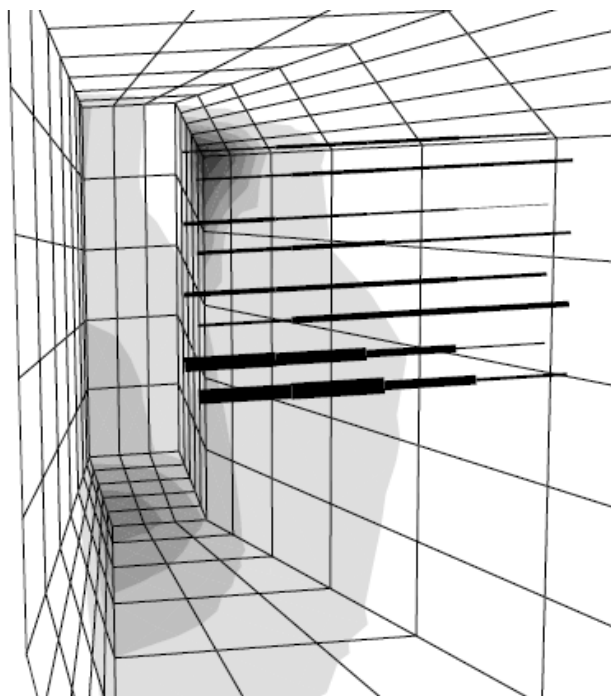


图2.22 锚索受力和沟渠周围土体位移图

2.7.7 常用命令总结

现在把2.7这部分使用到的主要命令罗列在表2.5中。表中命令是按分析问题的先后顺序从上到下排列的。这些命令都是FLAC3D分析问题时必须要用到的。

我们使用这些命令来做测试时(比如简单的开挖加固问题的分析或者单轴压缩试验的分析),最好再回到**命令手册**部分第一章查看一下这些命令的具体使用方法以及细节问题,然后试着给模型施加更复杂的条件,加以分析。在进行具体分析模拟之前,我们建议你先学习下**3.3节**,它论述了分析问题的基本方法。

表2.5 FLAC3D常用命令表

功能	命令
网格生成	gen
本构模型定义	mo
材料赋值	pro
定义边界条件/初始条件 (设定重力加速度)	apply
	Fix
	in
	Set grav
得到初始平衡状态	step
	solve
进行条件变更	mo
	pro
	apply
	fix
	free
	Sel cable
保存/恢复状态	Save
	rest

2.8 符号约定

本部分将介绍FLAC3D中有关参数的符号约定规则,用户在输入相应变量或者对结果进行评估分析时必须牢牢记在心里。

正应力: 拉为正; 压为负。

切应力: 可参考图2.23, 如果一个面的外法线方向与坐标轴正方向一致, 在

该面上，若剪应力指向其第二个下标对应的坐标正方向，那么该剪应力为正，若指向其第二个下标对应的坐标负方向，那么该剪应力为负；相反的，如果一个面的外法线方向与坐标轴负方向一致，在该面上，若剪应力指向其第二个下标对应的坐标负方向，那么该剪应力为正，若指向其第二个下标对应的坐标正方向，那么该剪应力为负。图2.23中所标示的剪应力均为正，应力张量是对称分布的。

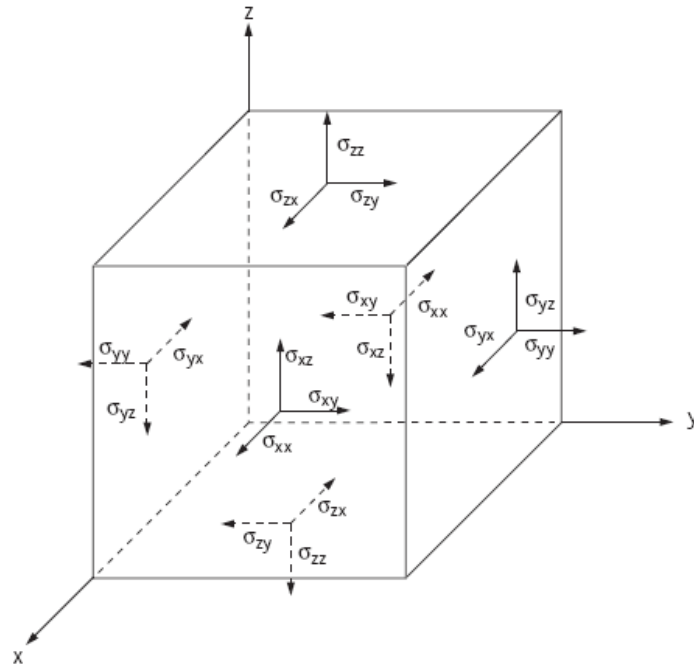


图2.23 应力分量正方向示意图

正应变：拉应力对应的应变为正；压应力对应的应变为负。

切应变：正的切应力对应的切应变为正，负的切应力对应的切应变为负。

压力：指向面的内部为正，指向外的为负，见图2.24。

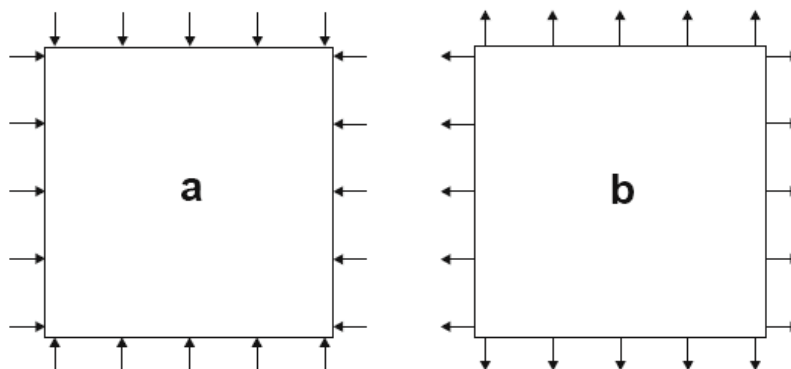


图2.24 压力符号约定（图a为正压力，b为负压力）

孔隙压力：压缩时的孔隙压力为正，反之为负。

倾角、倾向：假定了x轴方向为“东”，y轴方向为“北”，z轴方向指向“上”。倾角指的是，在x-y平面下半空间区域部分与x-y平面的夹角。倾向指投影在x-y平面上，以y轴为基准，绕着顺时针方向转，与y轴之间的夹角。指向坐标轴方向的矢量，比如力，位移，速度，只要指向坐标正方向即为正。

2.9 单位系统

FLAC3D接受任何工程单位制。下面的表2.6和表2.7，例举了一些基本参数在不同工程单位制中的单位。用户如若要从一种单位制转到另一种单位制，那么就必须要小心操作，不要弄混了。《石油工业杂志》（1977.12）中详细论述了各种单位制度以及英制（IMPERIAL）和国际标准单位制（SI）之间的转换关系。它是在我们在变换单位制时非常有用的参考资料。

表2.6 力学单位系统

名称	国际标准单位制				英制	
长度	m	m	m	cm	ft	in
密度	Kg/m ³	10 ³ Kg/m ³	10 ⁶ Kg/m ³	10 ⁶ g/cm ³	Slugs/ft ³	Snails/in ³
力	N	kN	MN	Mdynes	Lbf/ ft ³	Lbf
应力	pa	KPa	Mpa	bar	Lbf	psi
重力	m/s ²	m/s ²	m/s ²	cm/s ²	ft/s ²	In/s ²

注：1 bar=10⁶dynes/cm²=10⁵N/m²=10⁵pa;

1 atm=1.013bars=14.7psi=2116lbf/ ft²=1.01325 × 10⁵pa;

1 slug=1 lbf · s²/ft=14.59kg;

1 snail=1 lbf · s²/in;

1 gravity=9.81m/s²=981cm/s²=32.17ft/s²。

表2.6 地下水流单位系统

名称	国际标准单位制		英制	
水的体积模量	pa	bar	Lbf/ ft ²	psi
水的密度	10 ³ Kg/m ³	10 ⁶ g/cm ³	Slugs/ft ³	snails/in ³
渗透率	m ³ s/kg	10 ⁻⁶ cm s/g	ft ³ s/slug	in ³ s/snail
内渗透性	m ²	cm ²	ft ²	In ²
渗透系数	m/s	cm/s	ft/s	In/s

注意：FLAC3D渗透性恒等于内渗透性 (cm^2) $\times 9.9 \times 10^{-2}$

在国际标准单位制中恒等于渗透系数 (cm/s) $\times 1.02 \times 10^{-6}$

FLAC3D渗透性是一个流度系数（达西定律中所说的孔隙压力系数）

结构单元和热传导问题的有关参数的单位制可分别参见第一章的**结构单元**和**可选模式**部分。

2.10 计算精度

当选择好了统一的单位系统，就要注意另外一个问题：要防止计算过程中计算精度接近电脑的最大计算精度。如果计算的精度超过了电脑本身最大的计算精度，那么可能计算会发生崩溃，至少，在模型中肯定存在某些不易被识别和观察到的假象。

2.11 文件

FLAC3D可以调用或者创建六种类型的文件。它们有不同的扩展名。下面分别对它们进行阐述：

1、初始化系统配置文件

“FLAC3D.ini”文件，是一个由用户创建的ASCII类型的文件。在用户打开软件或者输入`new`命令时，系统会自动读取该文件。

2、数据文件

用户可以交互式的操作FLAC3D，也可以通过输入数据文件（也叫批处理文件）的方式进行有关的操作。它是由用户创建的，包括分析问题所需要的命令流的ASCII类型文件。一般来说，采用调用数据文件的形式进行有关操作是最高效的方法。用户若还不太清楚怎么调用数据文件，可以参见**命令手册**第一章中介绍`call`命令的部分。数据文件可以是任何文件名和扩展类型的文件。当然最好还是限定扩展名，以便可以与其它类型的文件加以区别。（比如把命令流文件保存为以“.dat”后缀的文件；将FISH语言文件保存为“.fis”文件）

3、保存文件

“.Sav”后缀的文件是用使用`save`命令而创建的文件。使用了`save`命令，程序会在当前文件夹下创建一个默认名为“FLAC3D.SAV”的保存文件。当然用户也可通过`save`命令来指定所要保存的文件名。“FLAC3D.SAV”保存了两部分信息：所有状态变量的值和用户定义的各种边界条件。保存文件最主要的作用是方便用

户调参，并研究有关参数对问题的影响。保存文件的存在，就避免了每改变一次参数就要重头开始分析、计算问题的麻烦。我们可以使用 **restore** 命令来调用 **sav** 文件恢复到问题分析的某一已存储阶段（具体参见**命令手册**第一部分有关 **restore** 命令的详细说明）。通常来说，计算达到某一状态后便存储一个相应的 **sav** 文件，是一个比较好的习惯。

4、日志文件

如果用户输入 **set logo on** 命令，那么在问题的分析过程中，软件会生成一个“FLAC3D.log”文件。它是一个格式化的ASCII文件，默认的名为“FLAC3D.log”，用户也可以使用 **set logfile + 文件名** 命令来指定所生成的log文件的名称。当使用了 **set logo on** 命令后，接下来在软件分析过程中在屏幕上出现的所有文本信息都会被保存在日志文件中。

5、历史记录文件

“FLAC3D.his”文件是用户输入 **history write n** 命令创建的一个历史记录文件。命令中的 **n** 指的是历史纪录项的记录序号（参见**命令手册**第一章中有关 **history** 命令的说明）。它也是一个格式化的ASCII文件。系统默认的文件名为“FLAC3D.his”。用户也可以通过在 **history write** 命令的后面加上关键词 **file 文件名.his** 的方法来指定历史纪录文件的文件名。用户可以在文件中看到有关历史记录项随计算步变化的值。

6、打印文件

打印文件是用户通过在命令窗口中输入 **plot hardcopy** 命令或者在菜单窗口中点 **file-print** 生成的。生成文件后，显示图形就会和打印机进行连接。用户可以使用 **set output 文件名.ps** 命令来指定打印文件的文件名。其它图形输入选项已经在 2.1.4 中做了介绍。

7、影片文件

DCX MOVIE: 这个文件是用户使用 **set movie dcx** 和 **movie** 命令生成的一个影片文件。它的作用是通过软件提供的影片播放器（**MOVIE.EXE**, 在安装目录 **shared\utility** 文件夹下）来回放捕捉下来的图形。如果用户没有指定文件名，那么在退出 **FLAC3D** 时，会在当前文件夹下生成 **FLAC3D.DCX** 文件。用户也可以通过命令：**set movie file 文件名** 指定文件名。具体命令的使用可以参见**命令手册**第

一章中有关movie和set movie 命令的说明。

AVI MOVIE: 这个文件是用户使用set movie AVI和movie命令生成的一个影片文件。它的作用是通过操作系统提供的媒体播放器来回放捕捉下来的图形。默认的文件名为FLAC3D.avi。同样的用户也可以通过set movie file 文件名命令来指定文件名。

8、FISH 输入输出流文件

指的是FISH语言创建的或读入、写入数据的文件。有两种格式：二进制文件和ASCII文件。

2.12 FLAC3窗口操作界面介绍

这一部分将简要介绍窗口交互界面。窗口交互界面主要是为了加快我们对于图像和文件的操作速度，我们只要点击鼠标就能完成本来需要输入命令行才能完成的操作。

当打开flac3d时，在屏幕上首先就会显示软件的主界面（图2.25）。最顶端的标题栏上显示了软件名称及版本号；标题栏下方为菜单栏；在底部有一个带有“FLAC3D>”提示符的命令窗口；命令窗口中可以输入任何命令来执行相关操作。在具体命令的操作中命令窗口输入框的上方会显示你输入或者输入的信息。

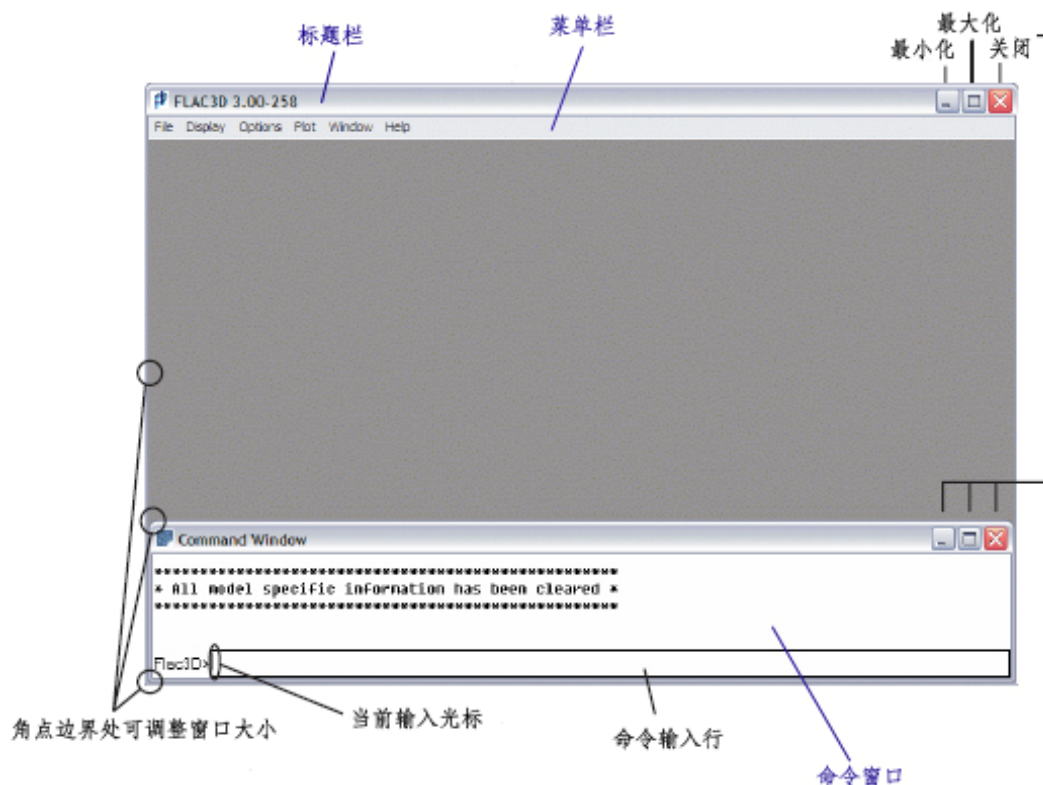


图 2.25 FLAC3D 主窗口

菜单栏上各选项按钮的单词第一个字母都有下划线，表示这个字母为该按钮的热键，只要在键盘上按住 ALT+该单词就可以启动该选项。当选择了其中一个选项时，便会弹出其子菜单，同样，子菜单中的各选项也有热键，子菜单的选项的启动只需直接在键盘上敲击下划的字母即可。按 ESC 键退出当前子菜单回到主窗口。



图 2.26 菜单栏

我们可以来尝试一下，打开软件，显示主窗口后，依次按下 ALT+F 键，C 键，这时会弹出一个如图 2.27 所示的对话框来选择要读入的数据文件。我们也可以依次选择 File-Call 选项执行该命令。

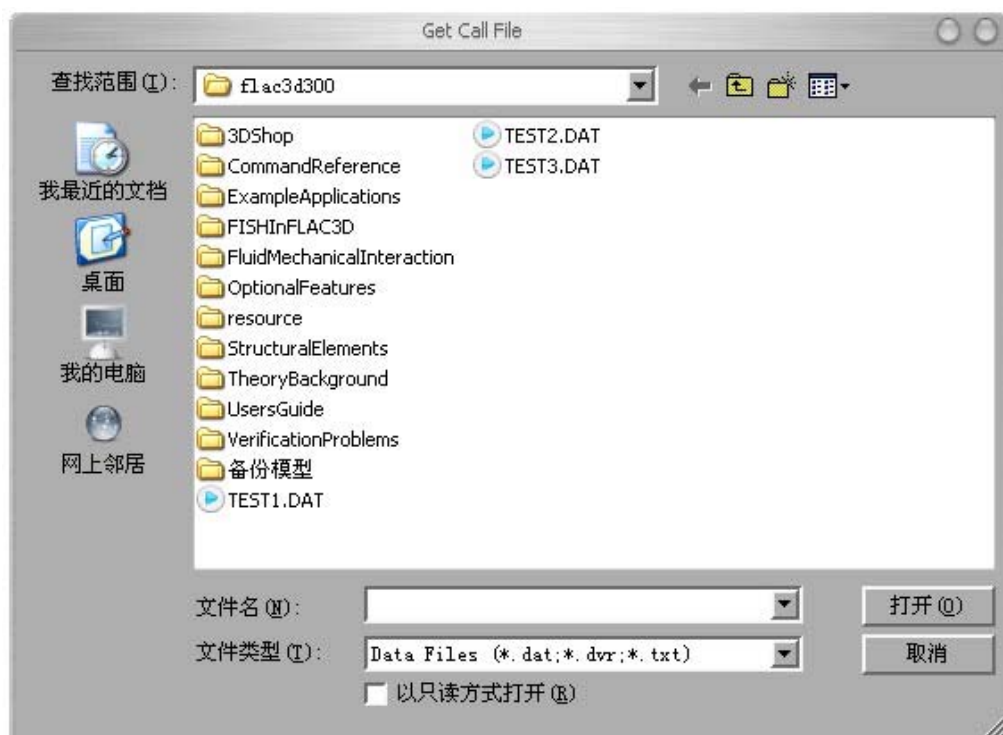


图 2.27 CALL 选项窗口

如果我们已有了自己的数据文件就可以找文件的路径然后读入数据文件了。如果不读入数据文件，那么按 ESC 键便可退出该窗口。

如果我们已经读入了之前的算例文件，那么可以看到，在主窗口上多了一个视图窗口（如图 2.28）。

用户可以在窗口中查看各选项的子菜单，自己试着做一些操作，慢慢熟悉软件的各项功能。

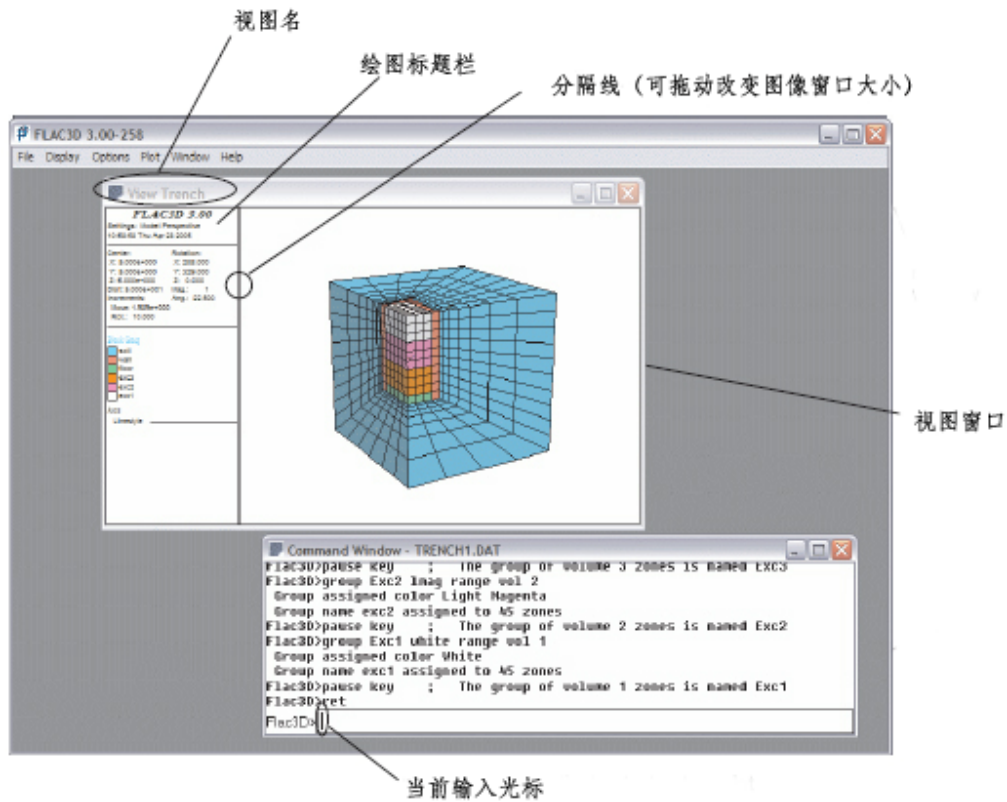


图 2.28 视图窗口（绘图窗口）

本部分内容就到此为止。